LETTER **Communicated by Emilio Salinas**

Exact Solutions for Rate and Synchrony in Recurrent Networks of Coincidence Detectors

Shawn Mikula
samikula@ucdavis.edu
Center for Neuroscience, University of California, Davis, CA 95618, U.S.A.

Ernst Niebur
niebur@jhu.edu
Krieger Mind/Brain Institute, Johns Hopkins University, Baltimore, MD 21218, U.S.A.

We provide analytical solutions for mean firing rates and cross-correlations of coincidence detector neurons in recurrent networks with excitatory or inhibitory connectivity, with rate-modulated steady-state spiking inputs. We use discrete-time finite-state Markov chains to represent network state transition probabilities, which are subsequently used to derive exact analytical solutions for mean firing rates and cross-correlations. As illustrated in several examples, the method can be used for modeling cortical microcircuits and clarifying single-neuron and population coding mechanisms. We also demonstrate that increasing firing rates do not necessarily translate into increasing cross-correlations, though our results do support the contention that firing rates and cross-correlations are likely to be coupled. Our analytical solutions underscore the complexity of the relationship between firing rates and cross-correlations.

1 Introduction

Neuronal codes have been the focus of considerable past work at both the experimental and theoretical levels, which has underscored the importance of firing rate and cross-correlation, though their exact roles in information processing and representation remain to be elucidated fully (Abeles, 1991; Alonso, Usrey, & Reid, 1996; Merzenich et al., 1996; Stevens & Zador, 1998; Steinmetz et al., 2000; Eagleman and Sejnowski, 2000; Niebur, 2002; Tomita and Eggermont, 2005). Given the importance of rate and cross-correlation in neural coding, the determination of how they differentially affect neuronal responses assumes significance if the impact of these different neural codes is to be fully appreciated.

Much of the theoretical basis for understanding information processing and neural coding in complex biological systems is based on computational

modeling—numerical solutions of the underlying model equations. While this approach has proven extremely useful and is the only practical one in many cases, analytical solutions nearly always would be preferable if they were available. Analytical solutions for neuronal coding would be especially useful to determine the relative contributions of different types of neuronal codes in network activity. It is the derivation and exploration of these types of solutions that forms the motivation for our current work.

In this letter, we present analytical solutions for recurrent networks of particularly simple model neurons, coincidence detectors, with arbitrary connectivity and thresholds. By "arbitrary connectivity," we mean that the network can have an arbitrary number of loops, the case of networks without loops having been solved by Mikula and Niebur (2005). However, we require that the network be strongly connected (i.e., it cannot be subdivided in subnetworks), as discussed below. Synapses can be of arbitrary strengths, they can be excitatory and inhibitory, and they can be between any two neurons, with or without loops. The input to the network is characterized in terms of the average firing rate. We derive exact closed-form solutions for all neurons (and pairs) in the network in the same form for the steady state. The model is based on our previous analytical solutions for the output firing rate of an individual coincidence detector receiving an arbitrary number of excitatory and inhibitory inputs, in both the presence and absence of synaptic depression (Mikula & Niebur, 2003a, 2003b, 2004), and on our solutions for multilayer feedforward networks of coincidence detectors (Mikula & Niebur, 2005).

After defining methods and notations in section 2, we present our main result, the closed-form expressions for steady-state firing rates and cross-correlations, in section 3. Several examples are studied in section 4 and compared with numerical solutions in section 5. Limitations of the model and implications regarding neural coding are discussed in section 6. The notation used is summarized in Table 1.

2 Methods

2.1 Model Neurons: Coincidence Detectors. The model neurons utilized in this study are coincidence detectors, also known as linear threshold gates, McCulloch-Pitts neurons receiving weighted inputs, or Perceptron units (McCulloch & Pitts, 1943; Rosenblatt, 1958; Rojas, 1996). A coincidence detector is a computational unit that fires at time t if the weighted sum of its inputs received within the window $(t, t - \delta t)$ equals or exceeds the threshold $\boldsymbol{\theta}$. This is a very simplified model of a neuron, but it is analytically tractable, and there is considerable experimental evidence indicating that at least under certain conditions, such as high background synaptic activity, neurons can function as coincidence detectors (Abeles, 1982; Wörgötter, Niebur, & Koch, 1991; König, Engel, & Singer, 1996; Destexhe, Contreras, & Steriade, 1998; Kempter, Gerstner, & van Hemmen, 1998; Destexhe &

Table 1: Notation Used.

Symbol	Description	Dimension/Range/ Number/Value
$\Delta(,)$	Generalized Krönecker δ	$\{0,1\}^n \times \{0,1\}^n \to \{0,1\}$
C	Connectivity (or adjacency) matrix	$\mathbb{R}^n \times \mathbb{R}^n$
I	Set of all input vectors	Total: 2^n
$I(t)$	Input at time t	$\{0,1\}^n$
I_i	Input vector number i	$\{0,1\}^n$
n	Number of neurons	$\mathbb{N}_+$
N	Number of system states	$N := 2^n$
$\boldsymbol{\Omega}$	Transition matrix	$[0,1]^N \times [0,1]^N$
$P(I_i)$	Probability of input vector i	$[0,1]$
$\pi(t)$	State probabilities of system at time t	$[0,1]^N$
π	Steady-state probabilities of the system ($t \to \infty$)	$[0,1]^N$
π_i	Component i of π (i.e., steady-state probability of state i)	$[0,1]$
p_i	Mean rate of input to neuron i	$[0,1]$
$p(i)$	Firing rate of neuron i	$[0,1]$
$\psi(t)$	State of system at time t	$\{0,1\}^n$
ψ_i	State vector number i	$\{0,1\}^n$
$\boldsymbol{\Psi}$	Matrix of all states	$\{0,1\}^N \times \{0,1\}^n$
$\boldsymbol{\theta}$	Threshold vector	$\mathbb{R}^n_+$
$\boldsymbol{\Theta}()$	Component-wise Heaviside function	$\mathbb{R}^n \to \{0,1\}^n$

Note: $[0,1]$ is the (closed) interval from 0 to 1, and $\{0,1\}$ denotes the binary pair of values 0 and 1.

Pare, 1999). Thus, even though our model neuron is very simple, it carries biological significance and may be considered biologically realistic under certain experimental conditions. We also point out that our formalism is applicable to the larger class of sigma-pi type of model neurons (Mel, 1993). A sigma-pi unit is a model neuron that sums contributions over clusters of input synapses, and the resulting sums are then multiplied. Optionally, a nonlinearity can be applied to the sum of products.

In many cases, it makes sense to think of δt as of a period on the order of 5 to 10 ms. This is the timescale of fast ionic synaptic conductances, and it is at this timescale that synaptic events superpose and interact. We do not, however, make use of this specific setting in our analysis other than requiring that it is sufficiently small that a maximum of one spike can be generated in a period of this length. An example neuron is shown in Figure 1, which also introduces some of the notation used.

2.2 Network Architecture. We define our network of n coincidence detectors as a pair, $(\boldsymbol{C}, \boldsymbol{\theta})$, where $\boldsymbol{C}$ is a connectivity matrix (also known as an adjacency matrix) whose (i, j)th entry, C_{ij}, is the numerical value of the connection from the ith coincidence detector to the jth coincidence detector, and where the threshold vector, $\boldsymbol{\theta}$, whose ith element, denoted θ_i, is the

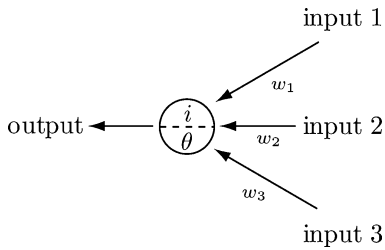

Figure 1: Model neuron used in this study, with three inputs in this case. The coincidence detector (circle in the center), with index, i, produces an output spike (left) when the sum over weighted binary inputs (right) in any given time bin is equal to or above threshold, θ. The thresholding operation is symbolized by the Heaviside function $\mathbf{\Theta}$ (see equation 2.1). The convention of representing the threshold, θ, in the lower half of the neuron, and the index, i, in the upper half, will be used in all figures.

nonnegative threshold for the ith coincidence detector. For a network of n neurons, the size of $\boldsymbol{C}$ is n^2, and the values of the connectivity matrix are real numbers—positive for excitatory connections and negative for inhibitory connections. For reasons that will become apparent in section 3.2, we require that our network be strongly connected in the graph-theoretic sense; that is, it is necessary that all nodes be reachable from every other node by at least one direct or indirect path. Whether a graph has this property can be tested efficiently (Corman, Leiserson, Rivest, & Stein, 2001); it is a very weak constraint and likely fulfilled for any biological neural network.

2.3 Input: Binomial Spike Trains with Specific Cross-Correlations. The inputs to our network are represented by the set I of all possible input combinations, $\boldsymbol{I}_k, k = 1, \ldots, 2^n$, and their corresponding probabilities, $P(\boldsymbol{I}_k)$. In previous reports (Mikula & Niebur, 2003a; Niebur, 2007), we introduced a systematic method for the generation of an arbitrary number of spike trains with specified firing rates (and also with specified pair-wise mean cross-correlations; only uncorrelated spike trains are used as input in the examples used in this letter). Action potentials are distributed according to binomial counting statistics in each spike train. A physiologically important special case is obtained if the rate of incoming spikes is low and convergence is high; the binomial statistics that governs the spikes generated by a coincidence detector can then be approximated by Poisson statistics. We further note that throughout this letter, we often refer to the probability that a bin contains a spike simply as an input or output firing rate, with the understanding that the actual firing rate is obtained by dividing the probability by the length of the time bins, δt.

2.4 Network Dynamical Equation. Synthesizing what we have stated above, the equation for updating the recurrent network $(\boldsymbol{C}, \boldsymbol{\theta})$ is given by

$$\boldsymbol{\psi}(t+1) = \boldsymbol{\Theta}(\boldsymbol{\psi}(t)\boldsymbol{C} + \boldsymbol{I}(t) - \boldsymbol{\theta}), \tag{2.1}$$

where $\boldsymbol{\psi}(t)$ is a binary row vector denoting the network state at time t and $\boldsymbol{I}(t)$ is a binary row vector denoting the input at time t. The symbol $\boldsymbol{\Theta}()$ represents the component-wise Heaviside function, that is, the Heaviside step function (zero for negative arguments, unity for zero or positive arguments) applied componentwise to the n-tuple, which is its argument.

3 Results

In this section, we derive the main results of this letter: the exact steady-state solutions for mean firing rates and cross-correlations in a recurrent network of coincidence detectors receiving rate- and cross-correlation modulated binomial inputs. Toward this end, we recast our network model in terms of a Markov chain.

3.1 Markov Chain Transition Matrix. Let $\boldsymbol{\Psi}$ be an enumeration of all 2^n network states such that each row contains a unique state; thus, row i of the matrix $\boldsymbol{\Psi}$ contains the state $\boldsymbol{\psi}_i$. To compute the Markov chain transition matrix, $\boldsymbol{\Omega}$, we note that it is a matrix of size $2^n \times 2^n$, whose ith row tells us how the ith network state, $\boldsymbol{\psi}_i$, probabilistically transforms into other network states in the next time step of the discrete network dynamics. That is, entry (i, j) in this matrix is the probability that state $\boldsymbol{\psi}_i$ goes to state $\boldsymbol{\psi}_j$ in the next time step, $\boldsymbol{\Omega}_{ij} = P(\boldsymbol{\psi}_i^t \rightarrow \boldsymbol{\psi}_j^{t+1})$. We use equation 2.1 to compute the network states at time $t+1$ for different probabilistically occurring inputs, $\boldsymbol{I}_k$, and thus obtain

$$\boldsymbol{\Omega}_{ij} = P\left(\boldsymbol{\psi}_i^t \rightarrow \boldsymbol{\psi}_j^{t+1}\right) = \sum_k P(\boldsymbol{I}_k) \times \Delta(\boldsymbol{\psi}_j, \boldsymbol{\Theta}(\boldsymbol{\psi}_i\boldsymbol{C} + \boldsymbol{I}_k - \boldsymbol{\theta})), \tag{3.1}$$

where the sum is over all 2^n input states and where $\Delta(\,,\,)$ is a generalized Kronecker-δ function that takes two network states as input and yields unity if the states are identical and zero otherwise.

3.2 Steady-State Vector of the Markov Chain Transition Matrix. Let $\boldsymbol{\pi}(t)$ be the row vector of size 2^n whose ith component denotes the probability that the network is in state i at iteration t. Using the elements of $\boldsymbol{\Psi}$ as indices for the Markov chain transition matrix $\boldsymbol{\Omega}$, it follows from elementary properties of Markov chains that

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\boldsymbol{\Omega}^t. \tag{3.2}$$

The long-term probabilities of finding the network in each of its possible states are found from

$$\boldsymbol{\pi}\boldsymbol{\Omega} = \boldsymbol{\pi}, \tag{3.3}$$

subject to the normalization condition,

$$\sum_{i=1}^{N} \pi_i = 1, \tag{3.4}$$

since each component of this vector is the probability of finding the network in the corresponding system state.

In words, equations 3.3 to 3.4 say that $\boldsymbol{\pi}$ is the eigenvector of the transition matrix with eigenvalue 1 and of unit length. Because the graph describing the network states is, by assumption, strongly connected, the transition matrix $\boldsymbol{\Omega}$ is irreducible (Graham, 1987).

Since the elements of this matrix are transition probabilities, they are nonnegative real numbers. According to the Perron-Frobenius theorem (Graham, 1987), the largest eigenvalue of the matrix (the so-called Perron root) is then positive and real. Its value is bounded from below and above by the smallest and largest row sums (sums of all elements of one row), and since $\boldsymbol{\Omega}$ is a (right) stochastic matrix, all row sums are unity (each term in a given row is the probability that a given state goes to one of the system states, and the sum of these probabilities has to be unity). The value of the Perron root must therefore be unity, satisfying the condition that the eigenvalue of $\boldsymbol{\Omega}$ is indeed 1.

So far, we have shown that equations 3.3 and 3.4 have a solution. In the final step, we show that it is unique. Indeed, the Perron-Frobenius theorem asserts that the Perron root is a simple eigenvalue, that is, it is a simple root of the characteristic polynome of $\boldsymbol{\Omega}$, and the eigenspace corresponding to this eigenvalue is therefore one-dimensional. The normalization condition that the sum over all its elements is unity, equation 3.4, determines the remaining degree of freedom and makes this normalized eigenspace therefore the unique solution of equations 3.3 and 3.4. We note in passing that the Perron-Frobenius theorem also guarantees that all elements of the eigenvector corresponding to the Perron root are nonnegative, as is required for their interpretation as probabilities of the system's states.

In practice, an efficient way to compute the steady-state solution is as follows. Let $\boldsymbol{I}$ be the $2^n \times 2^n$ identity matrix (this matrix $\boldsymbol{I}$ should not be confused with the set I of all possible inputs) and define $\boldsymbol{Q} = \boldsymbol{\Omega} - \boldsymbol{I}$. Furthermore let $\boldsymbol{e}$ be the 2^n-vector of all 1's, and $\boldsymbol{b}$ be the $(2^n + 1)$-vector with a 1 in position $2^n + 1$ and 0 elsewhere. The ith element of the steady-state vector of the Markov chain, $\boldsymbol{\pi}$, is the steady-state probability for the corresponding network state $\boldsymbol{\psi}_i$ or, equivalently, the ith row of $\boldsymbol{\Psi}$. The vector

$\boldsymbol{\pi}$ is obtained as solution of the following linear equation (Bolch, Greiner, de Meer, & Trivedi, 1998),

$$(\boldsymbol{Q} \mid \boldsymbol{e})^{\mathrm{T}} \boldsymbol{\pi}^{\mathrm{T}} = \boldsymbol{b}. \tag{3.5}$$

Appending $\boldsymbol{e}$ to $\boldsymbol{Q}$ and a final 1 at the end of the zero vector on the right-hand side ensures that normalization—that the solution vector $\boldsymbol{\pi}$ has components summing to 1.

3.3 Firing Rates for the Network. To obtain $p(i)$, the mean firing rate of the ith neuron, we sum over the probabilities for all those network states in which this neuron fires (i.e., is in state 1). Given that the components of the vector $\boldsymbol{\pi}$ are the steady-state probabilities and that the ith column of the network state matrix $\boldsymbol{\Psi}$ enumerates the activity states (0 or 1) of the ith neuron for all inputs, we obtain

$$p(i) = \sum_k \pi_k \Psi_{ki}, \tag{3.6}$$

with the sum running over all 2^n input states.

3.4 Cross-Correlations. The Pearson cross-correlation coefficient between neurons i and j is defined, as usual, as

$$q(i, j) = \frac{E(i, j) - E(i)E(j)}{\sqrt{E(i, i) - E(i)^2}\sqrt{E(j, j) - E(j)^2}}, \tag{3.7}$$

where $E()$ is the expectation value, calculated again as usual—$E(i, j) := E(\psi_{ki}\psi_{kj}) = \sum_k \pi_k \psi_{ki}\psi_{kj}$ and $E(i) := E(\psi_{ki}) = \sum_k \pi_k \psi_{ki} = p(i)$ as in equation 3.6. Given that a neuron state takes only values 0 and 1, we have $E(\psi_{ki}^2) = E(\psi_{ki})$ and

$$q(i, j) = \frac{\boldsymbol{\Theta}(\boldsymbol{\psi}^{(i)} + \boldsymbol{\psi}^{(j)} - 2)\boldsymbol{\pi}^{\mathrm{T}} - p(i)p(j)}{\sqrt{p(i) - p(i)^2}\sqrt{p(j) - p(j)^2}}. \tag{3.8}$$

4 Examples

4.1 Mutual Inhibition ($n = 2$). Let us consider the simple two-neuron recurrent network shown in Figure 2a with thresholds set equal to $+1$. Connection weights shown as edge labels are equal to -1 between the neurons and 1 for the inputs; thus, simultaneous input to a neuron from the other neuron and its external input will be subthreshold and produce no output. Recalling from section 2.2 that the (k, l)th entry of the connectivity

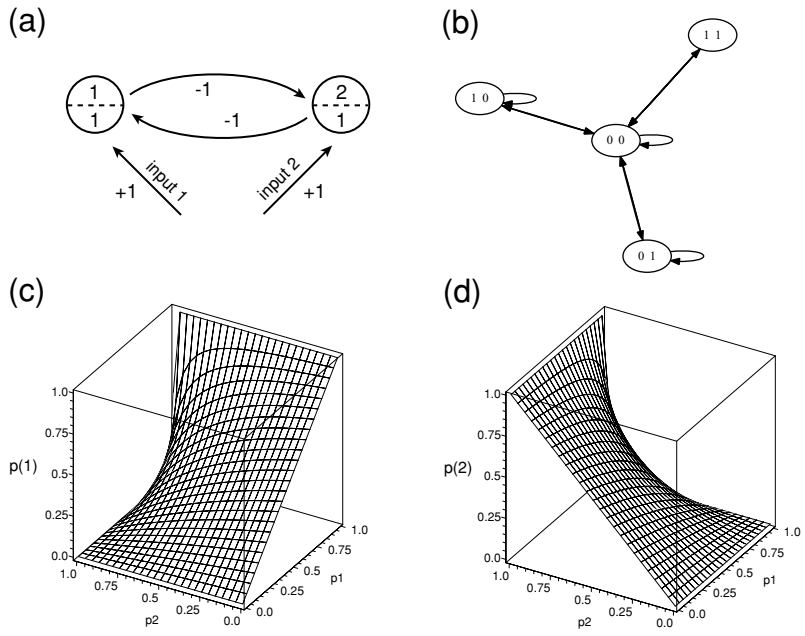

Figure 2: (a) A simple two-neuron recurrent network with mutual inhibition receiving two excitatory binomial inputs. Both thresholds are equal to 1. (b) State transition diagram for simple two-neuron recurrent network in *a*. Network states are 2-tuples, read from left to right; for example, 10 denotes that neuron 1 produces an output of 1, whereas neuron 2 produces no output. (c, d) Firing rates plotted as a function of input firing rate for this network.

matrix is defined as the weight of the connection to the kth neuron from the lth neuron, we obtain a connectivity matrix given by the following:

$$C = \begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}. \tag{4.1}$$

There are two neurons and thus 2^2 input states. The resulting network state matrix, Ψ, is

$$\Psi = \begin{pmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}. \tag{4.2}$$

The first row of Ψ is the network state for having zero output spikes for both neurons, the second row is the network state for having neuron 2 have one output spike and neuron 1 has zero, and so on for all of the four rows of Ψ. Figure 2b illustrates the network state space.

We now proceed to construct the Markov chain transition matrix, Ω, using the rows of Ψ as corresponding indices. Using equation 3.1, we obtain the following:

$$\Omega = \begin{pmatrix} (1-p_1)(1-p_2) & (1-p_1)p_2 & p_1(1-p_2) & p_1 p_2 \\ 1-p_2 & p_2 & 0 & 0 \\ 1-p_1 & 0 & p_1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \tag{4.3}$$

where p_1 is the spiking probability of the input to neuron 1 and p_2 is the spiking probability of the input to neuron 2. See Figure 2b for the corresponding state diagram.

From section 3.1, we obtain π, the steady-state vector of the Markov chain transition matrix, Ω:

$$\pi^T = \begin{bmatrix} \dfrac{p_1 p_2 + 1 - p_2 - p_1}{1 - 2p_1 p_2 + {p_1}^2 {p_2}^2} \\ \dfrac{({p_1}^2 - 2p_1 + 1)p_2}{1 - 2p_1 p_2 + {p_1}^2 {p_2}^2} \\ \dfrac{({p_2}^2 - 2p_2 + 1)p_1}{1 - 2p_1 p_2 + {p_1}^2 {p_2}^2} \\ \dfrac{p_1 p_2 (p_1 p_2 + 1 - p_2 - p_1)}{1 - 2p_1 p_2 + {p_1}^2 {p_2}^2} \end{bmatrix}, \tag{4.4}$$

where the ith element of π is the steady-state probability for the corresponding network state, the ith row of Ψ.

The mean firing rate $p(i)$ of the ith neuron in the network is given in equation 3.6 as the sum over all those network states in Ψ in which the ith neuron has output unity, times the corresponding probabilities for the network states, given by π. From that equation and equation 4.4, we obtain

$$p(1) = \pi_3 + \pi_4 = \frac{(1-p_2)p_1}{1-p_1 p_2} \tag{4.5}$$

$$p(2) = \pi_2 + \pi_4 = \frac{(1-p_1)p_2}{1-p_1 p_2}, \tag{4.6}$$

where, as a reminder, π_i is the ith component of vector π.

Plots of equations 4.5 and 4.6 as functions of the input rates, p_1 and p_2, are shown in Figures 2c and 2d. One of the defining properties of the firing rates is the behavior close to the point where both neurons receive continuous input, $p_1 = p2 = 1$. The expressions in equations 4.5 and 4.6 are not defined here, and they cannot be continued into this point because different limits are reached along different trajectories in the p_1, p_2 plane. This can be most clearly seen on the axes $p_1 = 1$ and $p_2 = 1$. In the former case, neuron 1 fires continuously and neuron 2 never, and equations 4.5 and 4.6 yield $p(1) = 1$, $p(2) = 0$. The opposite occurs in the latter case and $p(1) = 0$, $p(2) = 1$ is obtained. Other functional dependencies between p_1 and p_2 yield other limits (not shown). No steady state is defined for the system in this limiting case.

The computation of the cross-correlation using equation 3.8 yields

$$q(1,2) = \frac{\boldsymbol{\pi}_4 - p(1)p(2)}{\sqrt{p(1) - p(1)^2}\sqrt{p(2) - p(2)^2}} = 0. \tag{4.7}$$

Although one might intuitively expect negative correlation between the two mutually inhibitory neurons, this intuition is not correct in the case we consider here. Each neuron inhibits its partner in the next time step of length δt since the input to each neuron is collected over this time period and the decision of whether to fire is made at its end. Since the inputs to the neurons are not correlated in time, the cross-correlation between the activity of the neurons is identically zero, as computed explicitly in equation 4.7. These results from the analytical solution are confirmed by simulation (see section 5).

4.2 Feedback Inhibition ($n = 3$). Let us now consider the three-neuron recurrent network receiving two uncorrelated and differentially weighted inputs, shown in Figure 3a, with thresholds of neurons 2 and 3 set equal to $+1$, and neuron 1 set equal to $+3$, and connection weights shown as edge labels. Recalling from section 2.2 that the (k, l)th entry of the connectivity matrix is defined as the weight of the connection to the kth neuron from the lth neuron, we obtain a connectivity matrix given by

$$\boldsymbol{C} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 2 & -1 & 0 \end{pmatrix}. \tag{4.8}$$

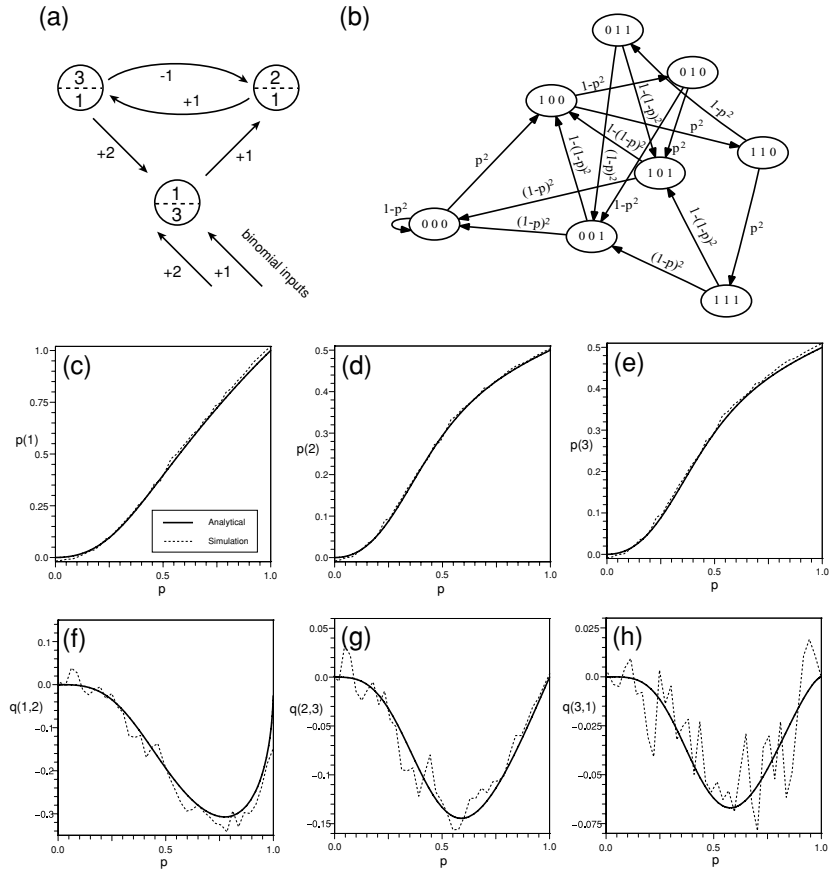

Figure 3: (a) A simple three-neuron recurrent network receiving two differentially weighted binomial input. Thresholds for neurons 2 and 3 are unity, and for the threshold for neuron 1 is 3. Edge values are connection weights. (b) State transition diagram for the network in *a*. Network states are 3-tuples, read from left to right; for example, 100 denotes that neuron 1 produces an output of 1, whereas neurons 2 and 3 produce no output (i.e., each has output 0). Edge values are network state transition probabilities. Note that the outgoing probabilities at individual nodes sum to unity. (c–h) Firing rates and cross-correlations plotted as a function of input firing rate for the network shown in Figure 3a. (c–e) Output firing rates of the network as a function of input rate, p. (c) $p(1)$ versus p, (d) $p(2)$ versus p, (e) $p(3)$ versus p. (f–h) Cross-correlations in the network as a function of input rate, p. (f) $q(1,2)$ versus p, (g) $q(2,3)$ versus p, (h) $q(3,1)$ versus p. Exact solutions from equations 4.12 to 4.14 and 3.8 shown as solid black lines, and simulation results as dotted lines.

Now there are three neurons, thus 2^3 input states, and the resulting network state matrix, $\mathbf{\Psi}$, is

$$\mathbf{\Psi} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix}. \tag{4.9}$$

Thus, the first row of $\mathbf{\Psi}$ is the network state with zero output spikes for all three neurons, the second row is the network state for neuron 3 having one output spike and neurons 1 and 2 have zero, and so on for all the eight rows of $\mathbf{\Psi}$. Figure 3b illustrates the network state space.

We now proceed to construct the Markov chain transition matrix, $\mathbf{\Omega}$, using equation 3.1. That is, we determine how network states at time t get mapped to network states at time $t + 1$. For instance, the first row of $\mathbf{\Psi}$, the state [000], gets mapped to the state [100] with probability $p_{in,1}\,p_{in,2}$, and to state [000] with probability $1 - p_{in,1}\,p_{in,2}$, where $p_{in,i}$ is the spike density for the ith binomial input. For simplicity, we assume $p_{in,1} = p_{in,2}$, which will be denoted as p. Continuing in this manner for all 16 network states yields the following:

$$\mathbf{\Omega} = \begin{pmatrix} 1-p^2 & 0 & 0 & 0 & p^2 & 0 & 0 & 0 \\ (1-p)^2 & 0 & 0 & 0 & 1-(1-p)^2 & 0 & 0 & 0 \\ 0 & 1-p^2 & 0 & 0 & 0 & p^2 & 0 & 0 \\ 0 & (1-p)^2 & 0 & 0 & 0 & 1-(1-p)^2 & 0 & 0 \\ 0 & 0 & 1-p^2 & 0 & 0 & 0 & p^2 & 0 \\ (1-p)^2 & 0 & 0 & 0 & 1-(1-p)^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1-p^2 & 0 & 0 & 0 & p^2 \\ 0 & (1-p)^2 & 0 & 0 & 0 & 1-(1-p)^2 & 0 & 0 \end{pmatrix}. \tag{4.10}$$

See Figure 3b for the corresponding state diagram.

From section 3.1, we obtain $\boldsymbol{\pi}$, the steady-state vector of our Markov chain transition matrix, $\boldsymbol{\Omega}$:

$$\boldsymbol{\pi}^T = \begin{bmatrix} \dfrac{1-2p+p^2}{1+p^4+4p^2-2p} \\ \dfrac{p^2\left(2p^4-2p^3-p^2+1\right)}{1+p^4+4p^2-2p} \\ -\dfrac{\left(p^2-1\right)p^2}{1+p^4+4p^2-2p} \\ -\dfrac{p^4\left(p^2-1\right)}{1+p^4+4p^2-2p} \\ \dfrac{p^2}{1+p^4+4p^2-2p} \\ -\dfrac{\left(2p^2-2p-1\right)p^4}{1+p^4+4p^2-2p} \\ \dfrac{p^4}{1+p^4+4p^2-2p} \\ \dfrac{p^6}{1+p^4+4p^2-2p} \end{bmatrix}, \tag{4.11}$$

where the ith element of $\boldsymbol{\pi}$ is the steady-state probability for the corresponding network state, the ith row of $\boldsymbol{\Psi}$. One notable result is that the first component is identically zero, indicating that the probability of finding the system in the fully quiet state (no neuron firing) is nil. This might be a counterintuitive result since one might expect absence of firing in all neurons to be a common state, in particular for very low input rates, that is, for $p \to 0$. Figure 3b shows why this intuition is wrong. The quiet state (000) can be reached only from itself. Therefore, a single spike will propel the system out of this state, and it will never return to it again. Therefore, the probability of finding this state in the steady-state solution vanishes. In the limit of vanishing firing rates, the steady-state solution is the sequence of states $100 \to 010 \to 001 \to 100$ (loop at bottom of Figure 3b). In the limit of low input, equation 3.2 yields probabilities of one-third each for these three states and zero for all others, in agreement with this observation.

To obtain $p(i)$, the firing rate of the ith neuron in our recurrent network, we use equation 3.6. That is, we sum over all network states in $\boldsymbol{\Psi}$, with the ith neuron output unity, times the corresponding probabilities for the network states, given by $\boldsymbol{\pi}$. Explicit solutions for the firing rates for the

three neurons comprising our network are as follows:

$$p(1) = -\frac{p^2\left(-1 + p^4 - 2\,p^3 - 2\,p^2\right)}{1 + p^4 + 4\,p^2 - 2\,p} \tag{4.12}$$

$$p(2) = \frac{p^2\left(1 + p^2\right)}{1 + p^4 + 4\,p^2 - 2\,p} \tag{4.13}$$

$$p(3) = p(2). \tag{4.14}$$

Plots of equations 4.12 to 4.14 as a function of input rate, p, are shown in Figures 3c to 3e. In agreement with the discussion of equation 4.11, the case $p \to 0$ discussed above yields mean firing rates of one-third for all three neurons. We also note in Figure 3c that in the opposite extreme of input firing rate (i.e., for $p \to 1$), the networks cycles exclusively through the sequence of states $100 \to 110 \to 111 \to 101 \to 100$ (loop around the center of the figure). Since neuron 1 fires in all four of these states, its firing rate must be unity in this case; neurons 2 and 3 both fire in exactly two of the states, and therefore their firing rates must be one-half. This result is confirmed by direct evaluation of equations 4.12 to 4.14 for $p \to 1$.

The cross-correlations for pairwise neurons in Figure 3a, obtained using equation 3.8, p, contain too many terms to display here; they are shown, plotted as a function of input rate, in Figures 3f to 3h.

Just as for the solutions for firing rates discussed earlier, naive intuition can be deceiving. While the negative correlation between neurons 2 and 3 shown in Figure 3g may be expected since these neurons are connected by a (one-way) inhibitory synapse, it may seem surprising that the correlation between neurons 1 and 2 (see Figure 3f) and that between neurons 2 and 3 (see Figure 3h), which are coupled by excitatory connections, are also negative for all input frequencies. The reason for the observed anticorrelation is related to the discussion following equation 4.11. As was observed there, in the case of vanishing input, the network will cycle through the three states in which each of the neurons 1, 2, 3 are activated in order, one at a time. Therefore, at a given time, exactly one of these neurons is active, while the other two are consistently inactive; this results in negative cross-correlations. As p increases, this relationship loses consistency, and for $p \to 1$, the system locks into another loop in which neuron 1 is always active and neurons 2 and 3 are firing in two of the four states. Inspection of Figure 3c shows that each of the pairs of neurons is not correlated in this loop: if the state of one neuron is 1, it is equally likely that that of the other neuron is 0 or 1).

4.3 Cortical Microcircuit ($n = 4$). Next we turn to the derivation of exact solutions for a simple model inspired by the canonical structure of cortical microcircuits (Callaway, 1998; Binzegger, Douglas, & Martin, 2004;

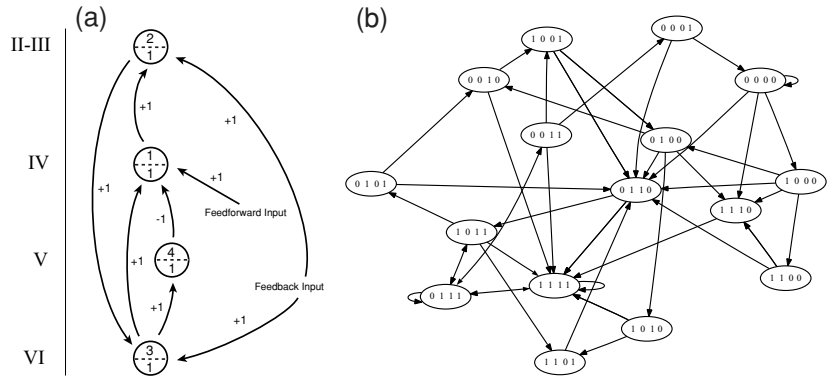

Figure 4: A cortical microcircuit. (a) Anatomy. A four-neuron recurrent network receiving two excitatory rate-modulated feedforward and feedback inputs targeting different cortical layers. Edge values are connection weights. Neurons 1 to 3 are excitatory, whereas 4 is inhibitory. All neurons have unitary threshold. (b) State diagram. Inputs are two rate- and cross-correlation-modulated binomial spike trains characterized by firing rate p and cross-correlation q. Network states are 4-tuples, read from left to right; for example, 1000 denotes that neuron 1 produces an output of 1, whereas neurons 2 to 4 produce no output (i.e., each has output 0). Edge values (not shown) are state transition probabilities given in Table 2. Note that both the number of inputs and outputs for a given state are a power of 2.

Douglas & Martin, 2004), which has recently been gaining popularity for use in modeling studies (Grossberg & Howe, 2003; Raizada & Grossberg, 2003; Destexhe & Sejnowski, 2003; Grossberg & Swaminathan, 2004; Haeusler & Maass, 2006). Let us consider the four-neuron recurrent network shown in Figure 4. The connectivity matrix is given by

$$\boldsymbol{C} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \end{pmatrix}. \tag{4.15}$$

The threshold matrix is given by

$$\boldsymbol{\theta} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \tag{4.16}$$

Note that, as defined in section 2.1, a neuron fires if its input equals or exceeds the threshold $\boldsymbol{\theta}$. An input of size 1 will thus fire the neurons with the thresholds given in equation 4.16.

There are two independent, rate-modulated binary inputs for our example. The first one consists of feedforward (FF) inputs and targets layer IV (neuron 1). The second, referred to as feedback (FB) input, modulates the superficial layers II/III (neuron 2) and the deep layer VI (neuron 3). The FF input firing rate is denoted p_{ff}, whereas the FB input firing rate is denoted p_{fb}. To ease notation, we define $\overline{p_{ff}} = (1 - p_{ff})$ and $\overline{p_{fb}} = (1 - p_{fb})$.

The total number of states in the system is 64 (2^4 neuron states times 2^2 input states), and the complete transition matrix Ω therefore has a dimension of 64×64. Most of its elements vanish and rather than proceeding by a straightforward enumeration of all 4096 states, we now describe a more efficient way for finding the steady-state solution. Define the network states as 4-tuples, read from left to right; for example, 1000 denotes that neuron 1 produces an output of 1, whereas neurons 2 to 4 produce no output (i.e., each has output 0). Consider now the neural state diagram in Figure 4b, where the edge values are transition probabilities between the neural states. Note that the number of neural states (disregarding input states) is only 16, and the dimension of the transition matrix between neural states therefore is 16×16.

These transition probabilities are obtained from equation 3.1 and depend on the possible input configuration probabilities, that is, from considering the different ways and associated probabilities that a given neural state is transformed in one time step. For example, network state 0000 is transformed to 1000 only if there is a spike in the feedforward input but not in the feedback input. For instance, we find that the probability that 0000 is transformed to 1000 in one time step ($0000 \to 1000$) is $p_{ff}\overline{p_{fb}}$, and this is the entry in Table 2 for the matrix element with index $0000 \to 1000$. By repeating this process for each network state, the Markov state transition matrix is obtained. Most entries of the 16×16 matrix vanish; all nonzero entries are listed in Table 2. We thus find that the matrix of transitions between the system states is indeed quite sparse; only about 1% of its entries are nonzero (41/4096).

We obtain the steady-state vector per section 3.2. If specific values for p_{ff} and p_{fb} are chosen, a numerical solution of the linear system of equations, equation 3.5, with a 17×16 matrix is easily computed. Our interest here, however, is in the analytical solution of the system. It is obtained in straightforward though tedious computations for which symbolic equation solvers (like Maple 10; Waterloo Maple) are well suited. The solution contains hundreds of terms and is too verbose to show here. Firing rates and cross-correlations are computed from the solution by using equations 3.6 and 3.8, respectively. Again, each of the analytical expressions for the firing rates and cross-correlations contains hundreds of terms. They are not listed here explicitly; instead, they are shown in graphical form in Figures 5 and 6.

Table 2: State Transition Table for the Recurrent Network Shown in Figure 4.

State Transition	Probability
$0000 \to 0000$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$0000 \to 0110$	$\overline{p_{ff}}\,p_{fb}$
$0000 \to 1000$	$p_{ff}\,\overline{p_{fb}}$
$0000 \to 1110$	$p_{ff}\,p_{fb}$
$0001 \to 0000$	$\overline{p_{fb}}$
$0001 \to 0110$	p_{fb}
$0010 \to 1001$	$\overline{p_{fb}}$
$0010 \to 1111$	p_{fb}
$0011 \to 0001$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$0011 \to 0111$	$\overline{p_{ff}}\,p_{fb}$
$0011 \to 1001$	$p_{ff}\,\overline{p_{fb}}$
$0011 \to 1111$	$p_{ff}\,p_{fb}$
$0100 \to 0010$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$0100 \to 0110$	$\overline{p_{ff}}\,p_{fb}$
$0100 \to 1010$	$p_{ff}\,\overline{p_{fb}}$
$0100 \to 1110$	$p_{ff}\,p_{fb}$
$0101 \to 0010$	$\overline{p_{fb}}$
$0101 \to 0110$	p_{fb}
$0110 \to 1011$	$\overline{p_{fb}}$
$0110 \to 1111$	p_{fb}
$0111 \to 0011$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$0111 \to 0111$	$\overline{p_{ff}}\,p_{fb}$
$0111 \to 1011$	$p_{ff}\,\overline{p_{fb}}$
$0111 \to 1111$	$p_{ff}\,p_{fb}$
$1000 \to 0100$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$1000 \to 0110$	$\overline{p_{ff}}\,p_{fb}$
$1000 \to 1100$	$p_{ff}\,\overline{p_{fb}}$
$1000 \to 1110$	$p_{ff}\,p_{fb}$
$1001 \to 0100$	$\overline{p_{fb}}$
$1001 \to 0110$	p_{fb}
$1010 \to 1101$	$\overline{p_{fb}}$
$1010 \to 1111$	p_{fb}
$1011 \to 0101$	$\overline{p_{ff}}\,\overline{p_{fb}}$
$1011 \to 0111$	$\overline{p_{ff}}\,p_{fb}$
$1011 \to 1101$	$p_{ff}\,\overline{p_{fb}}$
$1011 \to 1111$	$p_{ff}\,p_{fb}$
$1100 \to 0110$	$\overline{p_{ff}}$
$1100 \to 1110$	p_{ff}
$1101 \to 0110$	1
$1110 \to 1111$	1
$1111 \to 0111$	$\overline{p_{ff}}$
$1111 \to 1111$	p_{ff}

Of note are the "spikes" in the cross-correlations, Figure 6, as both input probabilities approach unity. It is probable that these instabilities arise from vanishing denominators in equation 3.8; note that the mean rates $p()$ of all four neurons go toward unity in the limit $p_{ff}, p_{fb} \to 1$ (see Figure 5). The

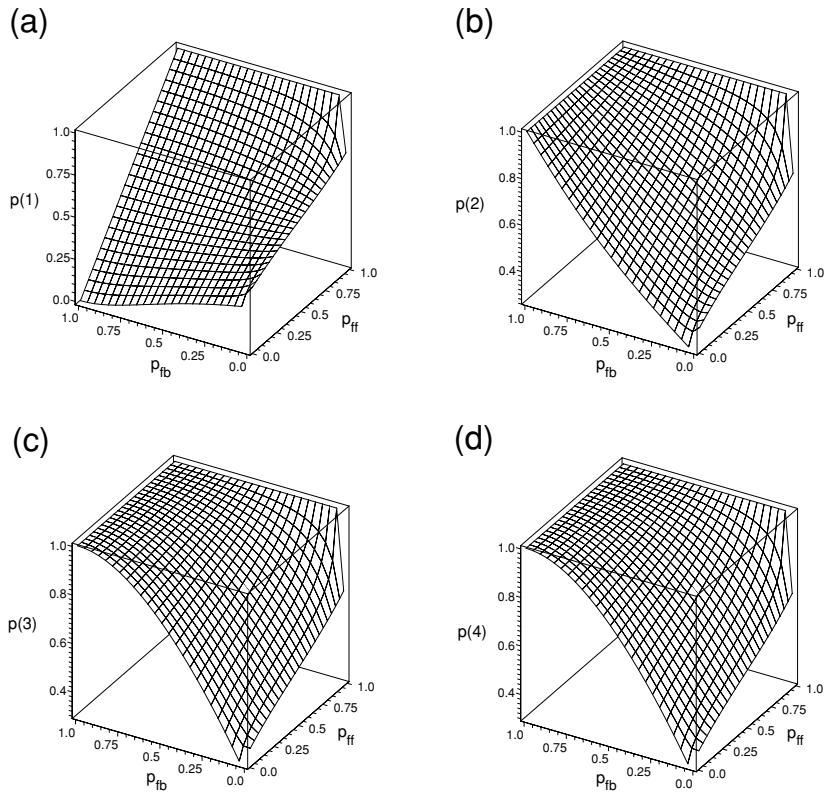

Figure 5: Firing rates plotted as a function of feedforward input firing rate, p_{ff}, and feedback input firing rate, p_{fb}, for the four-neuron recurrent network shown in Figure 4. (a) $p(1)$, (b) $p(2)$, (c) $p(3)$, and (d) $p(4)$. These plots are based on the exact solutions.

resulting simultaneously vanishing denominators and numerators in equation 3.8 clearly pose difficulties for the numerical evaluation routines whose results are shown in Figure 6. Comparison with simulations confirms the validity of the analytical solutions in general and, in particular, that these "spikes" in the cross-correlation solutions are likely artifacts of the equation solver (see section 5).

5 Numerical Simulations

Simulations of the three networks discussed in section 4 were run in Matlab. The initial states of each network were chosen randomly, and each network was then iterated through 5000 iterations of its basic dynamics. The first

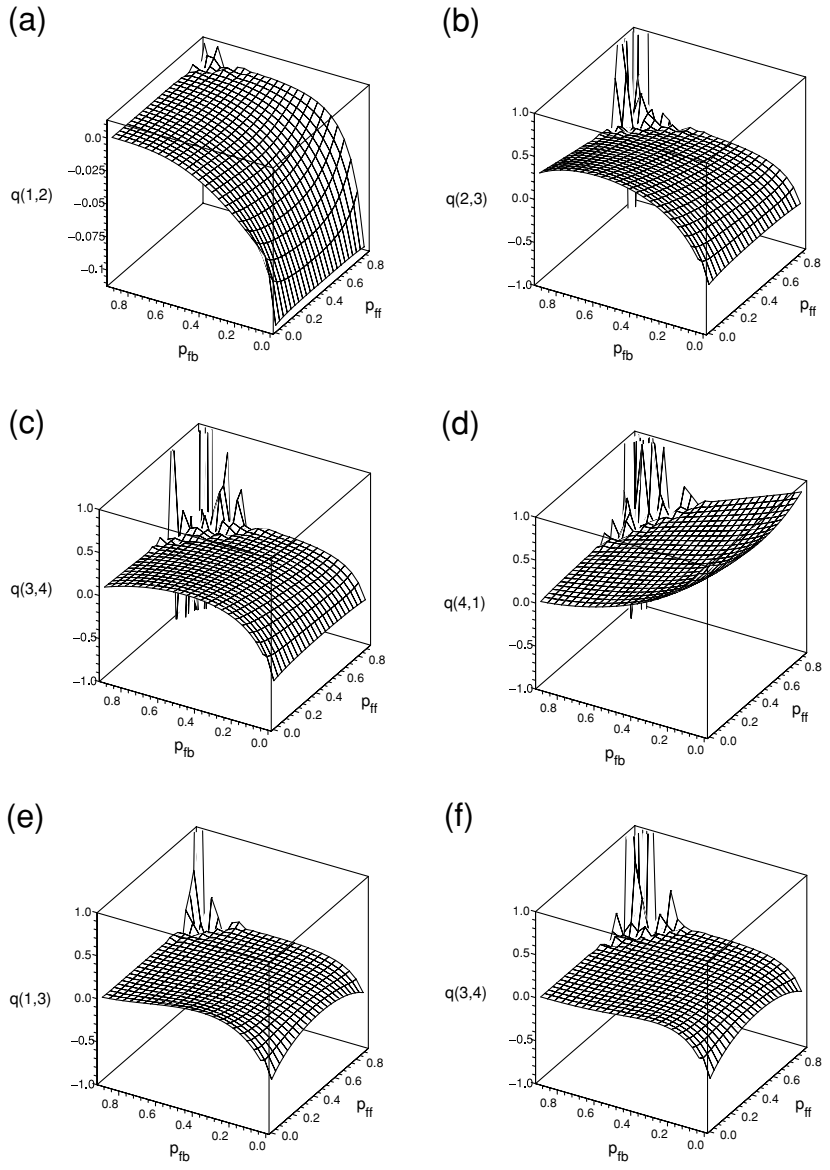

Figure 6: Cross-correlations plotted as a function of feedforward input firing rate, p_{ff}, and feedback input firing rate, p_{fb}, for the four-neuron recurrent network shown in Figure 4. (a) $q(1,2)$, (b) $q(2,3)$, (c) $q(3,4)$, (d) $q(4,1)$, (e) $q(1,3)$, and (f) $q(3,4)$. These plots are based on the exact solutions. Note the instabilities involved with evaluating the exact solutions for some large values of p_{ff} and p_{fb}.

2000 iterations were discarded to remove effects due to transient network activity, and the resulting steady states are characterized in this section.

All simulations were run for finite input probabilities (as discussed, formally obtained results for vanishing p and p_{ff}, p_{fb} may depend on the initial state and are not valid). Although small variations (jagged curves) are noted due to the finite lengths of the simulation runs, particularly for the correlation functions for which the total number of contributing events is lower than that for the mean firing rates, overall the agreement with the analytical solutions is excellent.

The simulation results for the firing rates confirm the exact solutions from section 4.2, shown in Figures 2c and 2d.

Simulation results for the firing rates and cross-correlations also confirmed the validity of the the exact solutions in Figures 3c to 3h. As discussed, the finite length of the simulation runs leads to small variations around the exact solutions, and this effect is more pronounced for correlations than for the mean firing rates because the average is over larger numbers of events (spikes) in the latter case than in the former (coincidences of spikes).

Our analytical results for the firing rates and cross-correlations of the cortical microcircuit from section 4.3, shown in Figures 5 and 6, were also corroborated by the numerical work. Again, the same "jaggedness" noted previously is observed for the cross-correlations. On the other hand, the numerical instabilities in the analytical solution close to the point $p_{fb} = p_{ff} = 1$ ("spikes" in Figure 6), which are due to the simultaneously vanishing numerators and denominators of the analytical solutions, are not observed in the simulations; this confirms that they are due to instabilities of the symbolic equation solver used and not properties of the system.

6 Discussion

This letter extends our previous analytical results (Mikula & Niebur, 2003a, 2003b, 2004, 2005) for an individual coincidence detector and a feedforward network of coincidence detectors to a recurrent network of coincidence detectors. The limitations of using coincidence detectors as model neurons, mainly targeting biological plausibility, have been discussed previously (Mikula & Niebur, 2003a). We note that our derivation is valid only for steady-state neuronal responses and does not inform us about transient responses, which are likely to be of importance in many cases.

The extension of our analytical methods to arbitrary networks of coincidence detectors reveals two additional limitations: combinatorial explosion and algebraic intractability. The combinatorial explosion limits the size of the networks that may be analytically solved since the computations scale as 2^n for n neurons. In practice, this limits our solutions to moderately sized networks of neurons, which is still useful for analyzing systems like the canonical microcircuit discussed here. In addition to the networks described

in section 4, we have solved systems with up to $n = 6$ neurons (results not shown).

The second limitation, analytical intractability, is a problem arising from symbolical evaluation of expressions containing hundreds or thousands of terms. Related to this problem, but not to the validity of the analytical solutions themselves, are issues related to the numerical evaluation for the purpose of plotting these analytical solutions. Where this is most evident in the results presented here is in Figure 6, where numerical instabilities appear during the evaluation of the exact solutions for some large values of p_{ff} and p_{fb}. As discusssed, these are most likely caused by simultaneously vanishing denominators and numerators in equation 3.8; note that the mean rates $p()$ of all four neurons go toward unity in the limit $p_{ff}, p_{fb} \to 1$ (see Figure 5). Simultaneously vanishing denominators and numerators in equation 3.8 clearly pose difficulties for the numerical evaluation routines whose results are shown in Figure 6. Comparison with the network simulation (not shown) confirms that these instabilities are artifacts of the numerical evaluation.

Simple examples yielding insight into the system's behavior are the cases when all inputs to the canonical microcircuit have either very high frequency or very low frequency. Figure 5 shows that in the former case ($p_{ff}, p_{fb} \to 1$), the firing rates of all four neurons approach unity, as one might have expected. In the opposite case ($p_{ff}, p_{fb} \to 0$), the same figure shows that all four neurons again approach a common firing rate, which is now one-third. This arises because for low input probabilities, the steady state of the system is the cycle $0100 \to 0010 \to 1001 \to 0100$ (in the upper center part of Figure 4b). From inspection (or from formal evaluation of equation 3.6), it is clear that the mean firing rates of all three neurons are one-third while in this cycle. While this numerical result could have been obtained from simulation of the system, the systematic evaluation of the analytical solution provides a much more principled approach.

It is interesting to compare the derivation of recurrent network solutions with the derivation of feedforward network solutions (Mikula & Niebur, 2005). We note that the role of the Markov chain transition matrix in the recurrent network solution is analogous to the role of the truth table in the feedforward network solution and that the computational complexity for recurrent network solutions scales as 2^n, where n is the number of neurons in the network, whereas for feedforward network solutions, the computational complexity scales as 2^m, where m is the number of inputs. While it might appear from these numbers that the complexity of recurrent networks may be smaller than that of feedforward nets (for $m > n$), this is not the case. The discrepancy is resolved by incorporating the multiplicative complexity of the inputs into the recurrent network solutions: while the number of neural states of the recurrent network is 2^n and the Markov transition matrix therefore has $2^n \times 2^n$ elements, each of these elements may consist of 2^m terms—one for each input configuration. This yields a modified

computational complexity of 2^{n+m}, thereby making the feedforward network solutions considerably more efficient than the recurrent network solutions and underscoring the fact that recurrent networks are exponentially more complicated than feedforward networks.

What is the relationship of our model, involving coincidence detectors with weighted connections and receiving probabilistic inputs of varying rates and cross-correlations, to finite automata (also known as finite state machines)? If our model has no probabilistic inputs, then it reduces to a finite automaton consisting of coincidence detectors, which have long been recognized as useful for pattern recognition (Keller, 1961). The incorporation of differentially rate modulated inputs increases the dynamic complexity of the system and at the same time increases the relevance of the model for neuroscience studies where the circuits of interest are open systems that receive known inputs that are characterized in terms of their rates. Thus, we believe our model formulation, while simplistic, nonetheless bears relevance for theoretical and computational studies of small to modest-sized neuronal circuits, where exact solutions are desirable.

In our analysis, we focused on firing rates and cross-correlations, leaving aside the issue of higher-order correlations. These may well be of importance, but they are more difficult to analyze, visualize, and interpret, and there are also many fewer data available to compare theoretical predictions to experimental results (but see Gerstein & Clark, 1964; Abeles & Goldstein, 1977; Abeles & Gerstein, 1988; Abeles, 1991; Martignon, Von Hasseln, Grün, Aertsen, & Palm, 1995; Riehle, Grün, Diesmann, & Aertsen, 1997, for experimental studies of higher order correlations). Additional analysis techniques may be useful for understanding higher-order correlations, for instance, snowflake plots (Czanner, Grün, & Iyengar, 2005), and may be an interesting direction for further development of the methods described in this letter.

What do our results say about the relationship between firing rates and cross-correlations? As is evident in Figures 5 and 6, the relationship is invariably nonlinear and potentially counter-intuitive. For example, Figure 5d shows that the firing rate of neuron 4 increases as the rate of the feedback input is increased, yet from Figure 6d, the cross-correlation between neurons 4 and 1, q(4,1), decreases as the rate of the feedback input is increased. Either of these results is consistent with our understanding of the network dynamics: increasing feedback input leads to increased firing rates to neuron 3 and subsequently, via an excitatory synapse, to increased firing of neuron 4. At the same time, the inhibitory synapse from neuron 4 to neuron 1 *may* lead to low correlation between these two neurons, and it does, in this situation. Together, these observations demonstrate that increasing firing rates do not necessarily translate into increasing cross-correlations, though our results do support the contention that firing rates and cross-correlations are likely to be coupled. The derivation of analytical solutions underscores the complexity of the relationship between firing rates and cross-correlations.

An example of mixed codes involving firing rates and cross-correlations in complex nervous systems might be the representation of selective attention in the primate cortex (Niebur & Koch, 1994). Selective attention has been shown in electrophysiological studies to be correlated both with rate changes as well as with changes in the fine temporal structure (on the order of milliseconds or tens of milliseconds) of neural activity (Moran & Desimone, 1985; Steinmetz et al., 2000; Fries, Reynolds, Rorie, & Desimone, 2001; Niebur, 2002; Saalmann, Pigarev, & Vidyasagar, 2007). It will take more experimental as well as theoretical work to come to a conclusive answer which of the proposed neural coding schemes are used by the different nervous systems.

Acknowledgments

We thank Yi Dong for a careful reading of the manuscript. This work was supported by NIH grants 5R01EY016281-02 and R01-NS40596.

References

Abeles, M. (1982). Quantification, smoothing, and confidence limits for single-units' histograms. *J. Neurosci. Methods*, *5*(4), 317–325.

Abeles, M. (1991). *Corticonics: Neural circuits of the cerebral cortex*. Cambridge: Cambridge University Press.

Abeles, M., & Gerstein, G. L. (1988). Detecting spatiotemporal firing patterns among simultaneously recorded single neurons. *J. Neurophysiol.*, *60*(3), 909–924.

Abeles, M., & Goldstein Jr, M. (1977). Multispike train analysis. *Proceedings of the IEEE*, *65*(5), 762–773.

Alonso, J. M., Usrey, W. M., & Reid, R. C. (1996). Precisely correlated firing in cells of the lateral geniculate nucleus. *Nature*, *383*(6603), 815–819.

Binzegger, T., Douglas, R. J., & Martin, K. A. (2004). A quantitative map of the circuit of cat primary visual cortex. *J. Neurosci.*, *24*(39), 8441–8453.

Bolch, G., Greiner, S., de Meer, H., & Trivedi, K. (1998). *Queueing networks and Markov chains: Modeling and performance evaluation with computer science applications*. New York: Wiley.

Callaway, E. M. (1998). Local circuits in primary visual cortex of the macaque monkey. *Annu. Rev. Neurosci.*, *21*, 47–74.

Corman, T., Leiserson, C., Rivest, R., & Stein, C. (2001). *Introduction to algorithms* (2nd ed.). Cambridge, MA: MIT Press, and New York: McGraw-Hill.

Czanner, G., Grün, S., & Iyengar, S. (2005). Theory of the snowflake plot and its relations to higher-order analysis methods. *Neural Computation*, *17*(7), 1456–1479.

Destexhe, A., Contreras, D., & Steriade, M. (1998). Mechanisms underlying the synchronizing action of corticothalamic feedback through inhibition of thalamic relay cells. *J. Neurophysiol.*, *79*(2), 999–1016.

Destexhe, A., & Pare, D. (1999). Impact of network activity on the integrative properties of neocortical pyramidal neurons in vivo. *J. Neurophysiol.*, *81*(4), 1531–1547.

Destexhe, A., & Sejnowski, T. J. (2003). Interactions between membrane conductances underlying thalamocortical slow-wave oscillations. *Physiol. Rev.*, *83*(4), 1401–1453.

Douglas, R. J., & Martin, K. A. (2004). Neuronal circuits of the neocortex. *Annu. Rev. Neurosci.*, *27*, 419–451.

Eagleman, D. M., & Sejnowski, T. J. (2000). Motion integration and postdiction in visual awareness. *Science*, *287*(5460), 2036–2038.

Fries, P., Reynolds, J. H., Rorie, A. E., & Desimone, R. (2001). Modulation of oscillatory neuronal synchronization by selective visual attention. *Science*, *291*, 1560–1563.

Gerstein, G., & Clark, W. (1964). Simultaneous studies of firing patterns in several neurons. *Science*, *143*(3612), 1325.

Graham, A. (1987). *Nonnegative matrices and applicable topics in linear algebra*. Chichester: Ellis Horwood Limited.

Grossberg, S., & Howe, P. D. L. (2003). A laminar cortical model of stereopsis and three-dimensional surface perception. *Vision Res.*, *43*(7), 801–829.

Grossberg, S., & Swaminathan, G. (2004). A laminar cortical model for 3D perception of slanted and curved surfaces and of 2D images: Development, attention, and bistability. *Vision Res.*, *44*(11), 1147–1187.

Haeusler, S., & Maass, W. (2006). A statistical analysis of information-processing properties of lamina-specific cortical microcircuit models. *Cerebral Cortex*, *17*(1), 149–162.

Keller, H. (1961). Finite automata, pattern recognition and perceptrons. *Journal of the ACM*, *8*(1), 1–20.

Kempter, R., Gerstner, W., & van Hemmen, J. (1998). How the threshold of a neuron determines its capacity for coincidence detection. *Biosystems*, *48*(1–3), 105–112.

König, P., Engel, A. K., & Singer, W. (1996). Integrator or coincidence detector? The role of the cortical neuron revisited. *Trends Neurosci.*, *19*(4), 130–137.

Martignon, L., Von Hasseln, H., Grün, S., Aertsen, A., & Palm, G. (1995). Detecting higher-order interactions among the spiking events in a group of neurons. *Biol. Cybern.*, *73*(1), 69–81.

McCulloch, W., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biology*, *5*(4), 115–133.

Mel, B. W. (1993). Synaptic integration in an excitable dendritic tree. *J. Neurophysiol.*, *70*(3), 1086–1101.

Merzenich, M. M., Jenkins, W. M., Johnston, P., Schreiner, C., Miller, S. L., & Tallal, P. (1996). Temporal processing deficits of language-learning impaired children ameliorated by training. *Science*, *271*(5245), 77–81.

Mikula, S., & Niebur, E. (2003a). The effects of input rate and synchrony on a coincidence detector: Analytical solution. *Neural. Comput.*, *15*(3), 539–547.

Mikula, S., & Niebur, E. (2003b). Synaptic depression leads to nonmonotonic frequency dependence in the coincidence detector. *Neural Comput.*, *15*(10), 2339–2358.

Mikula, S., & Niebur, E. (2004). Correlated inhibitory and excitatory inputs to the coincidence detector: Analytical solution. *IEEE Trans. Neural. Netw.*, *15*(5), 957–962.

Mikula, S., & Niebur, E. (2005). Rate and synchrony in feedforward networks of coincidence detectors: Analytical solution. *Neural Comput.*, *17*(4), 881–902.

Moran, J., & Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex. *Science*, *229*, 782–784.

Niebur, E. (2002). Electrophysiological correlates of synchronous neural activity and attention: A short review. *Biosystems*, *67*(1–3), 157–166.

Niebur, E. (2007). Generation of synthetic spike trains with defined pairwise correlations. *Neural Computation*, *19*(7), 1720–1738.

Niebur, E., & Koch, C. (1994). A model for the neuronal implementation of selective visual attention based on temporal correlation among neurons. *J. Comput. Neurosci.*, *1*(1–2), 141–158.

Raizada, R. D. S., & Grossberg, S. (2003). Towards a theory of the laminar architecture of cerebral cortex: Computational clues from the visual system. *Cereb. Cortex*, *13*(1), 100–113.

Riehle, A., Grün, S., Diesmann, M., & Aertsen, A. (1997). Spike synchronization and rate modulation differentially involved in motor cortical function. *Science*, *278*(5345), 1950–1953.

Rojas, R. (1996). *Neural networks: A systematic introduction*. Berlin: Springer.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, *65*(6), 386–408.

Saalmann, Y. B., Pigarev, I. N., & Vidyasagar, T. R. (2007). Neural mechanisms of visual attention: How top-down feedback highlights relevant locations. *Science*, *316*(5831), 1612–1615.

Steinmetz, P. N., Roy, A., Fitzgerald, P. J., Hsiao, S. S., Johnson, K. O., & Niebur, E. (2000). Attention modulates synchronized neuronal firing in primate somatosensory cortex. *Nature*, *404*(6774), 187–190.

Stevens, C. F., & Zador, A. M. (1998). Input synchrony and the irregular firing of cortical neurons. *Nature Neuroscience*, *1*(3), 210–217.

Tomita, M., & Eggermont, J. J. (2005). Cross-correlation and joint spectro-temporal receptive field properties in auditory cortex. *J. Neurophysiol.*, *93*(1), 378–392.

Wörgötter, F., Niebur, E., & Koch, C. (1991). Isotropic connections generate functional asymmetrical behavior in visual cortical cells. *J. Neurophysiol.*, *66*(2), 444–459.

Received July 11, 2007; accepted January 29, 2008.