J Neurophysiol 103: 2912–2930, 2010.
First published December 23, 2009; doi:10.1152/jn.00518.2009.

Generation of Spatiotemporally Correlated Spike Trains and Local Field Potentials Using a Multivariate Autoregressive Process

Diego A. Gutnisky[1] **and Krešimir Josić**[2]

[1]*Department of Neurobiology and Anatomy, University of Texas–Houston Medical School; and* [2]*Department of Mathematics, University of Houston, Houston, Texas*

Submitted 15 June 2009; accepted in final form 21 December 2009

Gutnisky DA, Josić K. Generation of spatiotemporally correlated spike trains and local field potentials using a multivariate autoregressive process. *J Neurophysiol* 103: 2912–2930, 2010. First published December 23, 2009; doi:10.1152/jn.00518.2009. Experimental advances allowing for the simultaneous recording of activity at multiple sites have significantly increased our understanding of the spatiotemporal patterns in neural activity. The impact of such patterns on neural coding is a fundamental question in neuroscience. The simulation of spike trains with predetermined activity patterns is therefore an important ingredient in the study of potential neural codes. Such artificially generated spike trains could also be used to manipulate cortical neurons in vitro and in vivo. Here, we propose a method to generate spike trains with given mean firing rates and cross-correlations. To capture this statistical structure we generate a point process by thresholding a stochastic process that is continuous in space and discrete in time. This stochastic process is obtained by filtering Gaussian noise through a multivariate autoregressive (AR) model. The parameters of the AR model are obtained by a nonlinear transformation of the point-process correlations to the continuous-process correlations. The proposed method is very efficient and allows for the simulation of large neural populations. It can be optimized to the structure of spatiotemporal correlations and generalized to nonstationary processes and spatiotemporal patterns of local field potentials and spike trains.

INTRODUCTION

The generation of artificial spike trains with predetermined spatiotemporal characteristics is of considerable interest in both theoretical and experimental neuroscience. In recent years there have been significant advances in simultaneous recording of neural activity at multiple sites. The resulting data provide strong evidence of spatial and temporal correlations in neural tissue (Bair et al. 2001; Gray et al. 1989; Heck et al. 2002; Kohn and Smith 2005; Zohary et al. 1994). Complex spatiotemporal patterns, such as synchronous and phase-locked oscillatory activity, have been observed experimentally and shown to play an important role in information processing (Engel et al. 2001; Fries et al. 2001; Gray et al. 1989; Kohn and Smith 2005; MacLeod and Laurent 1996; Riehle et al. 1997; Steinmetz et al. 2000; Vaadia et al. 1995; Wehr and Laurent 1996; Womelsdorf et al. 2006). However, the functional role of correlated activity is not fully understood. Theoretical tools can guide our efforts to understand the impact of spatiotemporal activity structure on neural coding (Averbeck and Lee 2006; de la Rocha et al. 2007; Gutnisky and Dragoi 2008; Markowitz et al. 2008; Moreno et al. 2002; Poort and Roelfsema 2009; Romo et al. 2003; Salinas and Sejnowski 2000, 2001; Shea-Brown et al. 2008).

Artificially generated spike trains with predetermined statistical properties could be used to study and manipulate cortical neurons in silico, in vitro, and in vivo *(*Netoff et al. 2005). For instance, optical recording and controlled stimulation of single neurons at many different locations along their dendritic tree is now a possibility (Callaway and Katz 1993; Duemani Reddy et al. 2008; Gasparini and Magee 2006; Huber et al. 2008). These techniques are dramatically increasing our understanding of the computational capabilities of single neurons. Advances in two-photon stimulation (Nikolenko et al. 2007, 2008) allow neurotransmitter uncaging at different synaptic locations in a precisely determined spatiotemporal pattern. Optical stimulation of neuroengineered microbial opsins can be used to activate and inactivate specific neural populations (Boyden et al. 2005; Han and Boyden 2007; Zhang et al. 2006, 2007a,b). These approaches will offer unprecedented insights into the input–output properties of single cells and neuronal networks. However, because the number of potential activation patterns is practically infinite, it will be possible to characterize them only statistically. The generation of neural activity patterns with a given statistical structure is therefore essential in the design of such experiments.

We propose a method to generate spike trains with predetermined mean firing rates and cross-correlations. These measures are typically used to characterize data from multiple electrode recording experiments. The cross-correlogram is widely used to describe the degree of interdependence between the activity of two neurons (Perkel et al. 1967) and study aspects of cortical processing. For instance, the presence and strength of monosynaptic connections can be usually inferred from the cross-correlogram (Alonso and Martinez 1998; Alonso et al. 2001; Fetz and Gustafsson 1983; Usrey et al. 1999; Veredas et al. 2005), whereas periodically recurring peaks in the cross-correlogram indicate the presence of oscillatory neural activity (Muresan et al. 2008). Furthermore, cross-correlograms have been used to study how functional connectivity among neurons depends on stimulus selectivity (Bair et al. 2001; Gray et al. 1989; Gutnisky and Dragoi 2008; Kohn and Smith 2005; Smith and Kohn 2008), task demands (Cohen and Newsome

Address for reprint requests and other correspondence: D. A. Gutnisky, Janelia Farm, 19700 Helix Drive, Ashburn, VA 20147 (E-mail: gutniskyd@janelia.hhmi.org).

2008; Poort and Roelfsema 2009; Riehle et al. 1997; Steinmetz et al. 2000; Vaadia et al. 1995), previous stimulation (Gutnisky and Dragoi 2008), and cognitive processing (Engel et al. 2001; Fries et al. 2001; Womelsdorf et al. 2006). Despite some efforts (Bohte et al. 2000; Kuhn et al. 2003; Niebur 2007) there are no general methods to efficiently generate a large number of spike trains with predetermined firing rates and arbitrary spatiotemporal correlations [but see the DISCUSSION section and similar methods proposed independently in Brette (2009), Krumin and Shoham (2009), and Macke et al. (2009)].

The proposed algorithm generalizes the following simple idea—a Poisson spike train can be approximated by discretizing time into bins of length Δt and assuming a spike in each bin occurs with probability $r\Delta t$ independently of all other events. More generally, for a collection of spike trains with complex spatial and temporal structure, this probability becomes dependent on past and current spikes in all the spike trains. To capture this structure we generate a continuous stochastic process[1] by filtering Gaussian noise through a multivariate autoregressive model (AR). We threshold this process to generate the desired collection of spike trains. The parameters of the AR are obtained by transforming the spike train correlations to continuous-process correlations. This approach allows us to optimize our algorithm to efficiently simulate spatiotemporal activity patterns in large groups of neurons. The ideas are easily extended to nonstationary statistics and simulations of neural populations with predetermined spike-triggered averages (STAs) of the local field potential (Fries et al. 2001). The algorithms described are implemented in Matlab (freely available at: http://www.math.uh.edu/~josic/code).

We first illustrate these ideas in the case of a single spike train. We next extend the approach to generate correlated spike train pairs. Several examples are provided that illustrate the method: single spike trains with almost absolute refractory period, populations with oscillatory cross-correlations, and experimentally constrained spike trains. We demonstrate how the algorithm can be extended to generate spike trains with time-varying statistics and spike-field coherence. Finally, we show how the level of synchrony in a presynaptic neural population can affect the firing rate of a conductance-based neural model.

METHODS

Setup

The dependencies of a collection of spike trains are frequently characterized by covariance density functions (cross-correlation function). If $N_i(a, b)$ denotes the number of spikes fired by neuron i in the time interval (a, b), then the *covariance density* is defined by (Cox and Isham 1980)

$$\mathrm{cov}\,[N_i(t, t + \Delta t), N_j(t + \tau, t + \tau + \Delta t)] = c_{i,j}(\tau)\Delta t^2 + o(\Delta t^2) \quad (1)$$

where $o(\Delta t^2)$ represents terms that decay faster than Δt^2.

In trying to statistically describe a collection of spike trains, one is faced with the problem of estimating $c_{i,j}(t)$ (Bair et al. 2001; Brody 1999a,b; Perkel et al. 1967). Our goal is the opposite—we prescribe $c_{i,j}(t)$ and aim to simulate a collection of spike trains with these statistics.

We start by discussing the second-order characterization of a collection of spike trains. Next we review the theory of autoregressive (AR) processes and discuss how to transform an AR process to a spike train using thresholding. Finally, we describe how to find an AR process that produces a collection of spike trains with the desired statistics after thresholding.

Covariance matrix of univariate and multivariate point process

A spike train $x_i(t)$ is a sequence of 1s and 0s and the occurrence of a spike between the times t_0 and $t_0 + \Delta t$ is denoted by $x_i(t_0) = 1$. The magnitude of Δt determines the resolution at which the simulation is performed.

We assume that each spike train $x_i(t)$ is stationary, with fixed rate r_i (generalizations are discussed in a following subsection). The probability of firing during a time interval Δt is therefore $\Pr\,[x_i(t) = 1] = r_i\Delta t$ and $\langle x_i(t)\rangle = r_i\Delta t$, with the expectation $\langle\cdot\rangle$, taken over all time bins. The covariance between the number of spikes in time bins that are separated by τ units of time is given by

$$\begin{aligned} c_{i,j}(\tau) &= \langle x_i(t)x_j(t + \tau)\rangle - \langle x_i(t)\rangle\langle x_j(t)\rangle \\ &= \langle x_i(t)x_j(t + \tau)\rangle - r_ir_j\Delta t^2 \end{aligned} \quad (2)$$

Note that $c_{i,i}(0) = \langle x_i(t)^2\rangle - \langle x_i(t)\rangle^2 = r_i\Delta t + O(\Delta t^2)$ is the variance of $x_i(t)$.

To simplify notation, in the following we measure time in discrete increments of Δt, so that $t = n$ refers to the time bin of width Δt, starting at time $n\Delta t$. Since $c_{i,j}(\tau)$ is the covariance of $x_i(t)$ and $x_i(t + \tau)$, we see that the Toeplitz matrix

$$C_i = \begin{pmatrix} 1 & c_{i,i}\,(1) & c_{i,i}\,(2) & \cdots & c_{i,i}\,(L-3) & c_{i,i}\,(L-2) & c_{i,i}\,(L-1) \\ c_{i,i}\,(1) & 1 & c_{i,i}\,(1) & \cdots & c_{i,i}\,(L-4) & c_{i,i}\,(L-3) & c_{i,i}\,(L-2) \\ c_{i,i}\,(2) & c_{i,i}\,(1) & 1 & \cdots & c_{i,i}\,(L-5) & c_{i,i}\,(L-4) & c_{i,i}\,(L-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ c_{i,i}\,(L-3) & c_{i,i}\,(L-4) & c_{i,i}\,(L-5) & \cdots & 1 & c_{i,i}\,(1) & c_{i,i}\,(2) \\ c_{i,i}\,(L-2) & c_{i,i}\,(L-3) & c_{i,i}\,(L-4) & \cdots & c_{i,i}\,(1) & 1 & c_{i,i}\,(1) \\ c_{i,i}\,(L-1) & c_{i,i}\,(L-2) & c_{i,i}\,(L-3) & \cdots & c_{i,i}\,(2) & c_{i,i}\,(1) & 1 \end{pmatrix} \quad (3)$$

is the covariance matrix of the collection of random variables $x_i(t)$, $x_i(t + 1), \ldots, x_i(t + L - 1)$. Since C_i must be positive definite, the covariances $c_{i,i}$ cannot be arbitrary.

We denote the binary vector of events at time t by $\vec{x}(t) = [x_1(t), x_2(t), \ldots, x_N(t)]$, where N is the number of spike trains. The matrices $C(\tau)$ are the covariance matrix of the random variables $\vec{x}(t)$ and $\vec{x}(t + \tau)$,

[1] All the processes that we use in simulations are discrete in time, although some are continuous and some discrete in space. Rather than using the phrases "continuous in space," "discrete in time," or "discrete in space and time," we will refer to them as continuous and discrete processes, respectively.

$$C(\tau)=\begin{pmatrix} c_{1,1}(\tau) & c_{1,2}(\tau) & \cdots & c_{1,N-1}(\tau) & c_{1,N}(\tau)\\ c_{2,1}(\tau) & c_{2,2}(\tau) & \cdots & c_{2,N-1}(\tau) & c_{2,N}(\tau)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ c_{N-1,1}(\tau) & c_{N-1,2}(\tau) & \cdots & c_{N-1,N-1}(\tau) & c_{N-1,N}(\tau)\\ c_{N,1}(\tau) & c_{N,2}(\tau) & \cdots & c_{N,N-1}(\tau) & c_{N,N}(\tau)\end{pmatrix} \qquad (4)$$

The covariance matrices obtained from the collection of random variables $\vec{x}(t), \vec{x}(t+1), \ldots, \vec{x}(t+L-1)$ are $C(0), C(1), \ldots, C(L-1)$ (as defined by *Eq. 4*). We can define C as the $(NL) \times (NL)$ block–Toeplitz matrix

$$C=\begin{pmatrix} C(0) & C(1) & \cdots & C(L-2) & C(L-1)\\ C(1) & C(0) & \cdots & C(L-3) & C(L-2)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ C(L-2) & C(L-3) & \cdots & C(0) & C(1)\\ C(L-1) & C(L-2) & \cdots & C(1) & C(0)\end{pmatrix} \qquad (5)$$

The spatiotemporal correlations of N spike trains are described by C and $\vec{r} = (r_1, r_2, \ldots, r_N)$. Such matrices completely describe the second-order statistics of spike counts over windows of arbitrary size (Cox and Isham 1980).

If the cross-covariances are equal for every neuron pair, that is $c_{cross}(\tau) = c_{i,j}(\tau)$ for all (i, j), where $i \neq j$, and autocovariances are equal for every neuron, $c_{auto}(\tau) = c_{i,i}(\tau)$ for all i, we can write $C(\tau)$ as the Kronecker tensor product ($\otimes$)

$$C = C_{off} \otimes 1_N + C_{diag} \otimes I_N$$

where I_N is the identity matrix of size N, 1_N is a matrix of 1s, and C_{off} and C_{diag} are $L \times L$ Toeplitz matrices, defined as

$$C_{off}=\begin{bmatrix} c_{cross}(0) & c_{cross}(1) & \cdots & c_{cross}(L-2) & c_{cross}(L-1)\\ c_{cross}(1) & c_{cross}(0) & \cdots & c_{cross}(L-3) & c_{cross}(L-2)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ c_{cross}(L-2) & c_{cross}(L-3) & \cdots & c_{cross}(0) & c_{cross}(1)\\ c_{cross}(L-1) & c_{cross}(L-2) & \cdots & c_{cross}(1) & c_{cross}(0)\end{bmatrix}$$

and

$$C_{diag}=\begin{bmatrix} c_{auto}(0)-c_{cross}(0) & c_{auto}(1)-c_{cross}(1) & \cdots & c_{auto}(L-2)-c_{cross}(L-2) & c_{auto}(L-1)-c_{cross}(L-1)\\ c_{auto}(1)-c_{cross}(1) & c_{auto}(0)-c_{cross}(0) & \cdots & c_{auto}(L-3)-c_{cross}(L-3) & c_{auto}(L-2)-c_{cross}(L-2)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ c_{auto}(L-2)-c_{cross}(L-2) & c_{auto}(L-3)-c_{cross}(L-3) & \cdots & c_{auto}(0)-c_{cross}(0) & c_{auto}(1)-c_{cross}(1)\\ c_{auto}(L-1)-c_{cross}(L-1) & c_{auto}(L-2)-c_{cross}(L-2) & \cdots & c_{auto}(1)-c_{cross}(1) & c_{auto}(0)-c_{cross}(0)\end{bmatrix}$$

We next review the theory of autoregressive processes.

Univariate autoregressive processes

A univariate autoregressive (AR) process of order L is defined as

$$y(t)=\sum_{k=1}^{L} a_k \cdot y(t-k)+n(t) \qquad (6)$$

where the sequence $y(t)$ is a collection of continuous random variables, each with 0 mean and variance $R(0)$, $n(t)$ are independent univariate Gaussian random variables with 0 mean and variance σ^2, and the constants a_k characterize the AR process. Using arbitrary initial values $y(0)$ to $y(L)$, and iterating *Eq. 6* starting at time $t = L$, we see that for subsequent times each $y(t)$ is a sum of Gaussian random variables and thus is itself Gaussian. Multiplying by $y(t-\tau)$ and taking the expected value of both sides in *Eq. 6*, we obtain

$$\langle y(t)\cdot y(t-\tau)\rangle=\sum_{i=1}^{L} a_i\cdot\langle y(t-i)\cdot y(t-\tau)\rangle+\langle n(t)\cdot y(t-\tau)\rangle$$

Defining $R(\tau) = \langle y(t)\cdot y(t-\tau)\rangle$ we obtain

$$R(\tau)=\sum_{i=1}^{L} a_i\cdot R(|\tau-i|) \quad \text{for} \quad \tau>1$$

and

$$R(0)=\sum_{i=1}^{L} a_i\cdot R(i)\;+\;\sigma^2 \quad \text{for} \quad \tau=0 \qquad (7)$$

The variance is given by

$$\sigma^2=R(0)-\sum_{i=1}^{L} a_i\cdot R(i) \qquad (8)$$

Equations 7 and *8* are known as the Yule–Walker equations (Shanmugan and Breipohl 1988) and can be expressed in a matrix form as

$$\underbrace{\begin{bmatrix} R(1)\\ R(2)\\ \vdots\\ R(L-1)\\ R(L)\end{bmatrix}}_{\vec{R}} = \underbrace{\begin{bmatrix} R(0) & R(1) & \cdots & R(L-2) & R(L-1)\\ R(1) & R(0) & \cdots & R(L-3) & R(L-2)\\ \vdots & \vdots & \ddots & \vdots & \vdots\\ R(L-2) & R(L-3) & \cdots & R(0) & R(1)\\ R(L-1) & R(L-2) & \cdots & R(1) & R(0)\end{bmatrix}}_{T} \underbrace{\begin{bmatrix} a_1\\ a_2\\ \vdots\\ a_{L-1}\\ a_L\end{bmatrix}}_{\vec{A}} \qquad (9)$$

The parameters needed to obtain an AR process with a given autocovariance $R(\tau)$ can therefore be obtained by

$$A = T^{-1}\cdot R \qquad (10)$$

To define a Gaussian process in discrete time with prescribed autocovariance we simply follow the steps described earlier in reverse. Given an autocovariance function $R(\tau)$ defined for $\tau \in \{0, 1, \ldots, L\}$, we find coefficients a_k from *Eq. 10* and the variance of the Gaussian noise $n(t)$ from *Eq. 8*. The AR process obtained by iterating *Eq. 6* has the desired statistics.

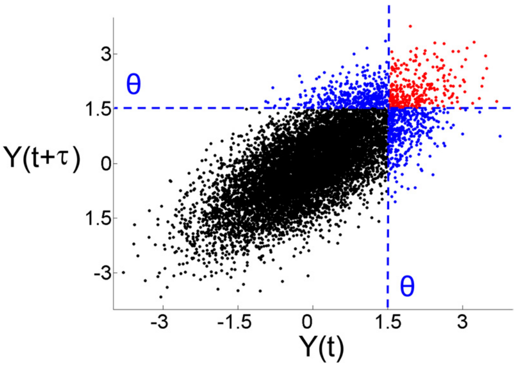

FIG. 1. Transformation of a bivariate random Gaussian variable to a bivariate point process. Thresholding the random variables $y(t)$ and $y(t + \tau)$ at θ divides the plane into 4 regions. Red dots represent a spike at time t followed by a spike at time $t + \tau$. Blue dots represent a spike either at time t (*top left*) or at time $t + \tau$ (*bottom right*). Black dots represent the absence of spikes at either time. The correlation between $y(t)$ and $y(t + \tau)$ determines the probability of a spike at time t and at time $t + \tau$, i.e., the probability of choosing a point in the red portion of the plane (*Eq. 13*).

Thresholding the AR process to obtain a univariate spike train

The simplest way of mapping continuous to binary variables is by assigning the value "1" whenever a threshold is exceeded and "0" otherwise. This operation can be used to convert the AR process described in the previous section to a binary sequence, i.e., a spike train. In particular, if $y(t)$ is defined using an AR process (*Eq. 6*), we define the spike train $x(t)$ as

$$x(t) = \begin{cases} 0 & \text{if} \quad y(t) \le \theta \\ 1 & \text{if} \quad y(t) > \theta \end{cases} \tag{11}$$

so that $x(t) = H_\theta[y(t)]$, where H_θ is the Heaviside function. For each t, the random variable $y(t)$ is Gaussian with zero mean and variance $R(0)$. Therefore the probability that $x(t) = 1$—i.e., the firing rate of the spike train obtained by thresholding—is

$$r = \frac{1}{\sqrt{2\pi R(0)}} \int_\theta^\infty e^{-\left[\frac{y^2}{2R(0)}\right]} \mathrm{d}y \tag{12}$$

The threshold θ divides the $y(t)$–$y(t + \tau)$ plane into four regions (Fig. 1) corresponding to a spike at time t and time $t + \tau$ (red), no spikes (black), or a spike either at t or $t + \tau$ (blue). The probability that both $y(t)$ and $y(t + \tau)$ jointly exceed the threshold is given by the integral of their joint density over the upper right quadrant (red region) in Fig. 1. The dependencies between $y(t)$ and $y(t + \tau)$ [i.e., $R(\tau)$] imply dependencies between $x(t)$ and $x(t + \tau)$. We define $p(\tau)$ as the mass of the bivariate Gaussian distribution above the threshold θ. Defining[2] $a = y(t)$ and $b = y(t + \tau)$

$$p(\tau) = \int_\theta^\infty \int_\theta^\infty \frac{1}{2\pi\sqrt{1 - R(\tau)^2}} \quad \exp\left\{-\frac{a^2 + b^2 - 2R(\tau)ab}{2[1 - R(\tau)^2]}\right\} \mathrm{d}a\mathrm{d}b \tag{13}$$

For the spike train $x(t)$, the covariance is determined by the probability of having two spikes τ time units apart, since $c(\tau) = \langle x(t)x(t + \tau)\rangle - r^2$. Thresholding therefore results in a spike train with autocovariance function related to $p(\tau)$ by

$$p(\tau) = \langle x(t)x(t + \tau)\rangle = c(\tau) + r^2 \tag{14}$$

Generating a univariate spike train

We now return to the problem of generating a univariate spike train with a prescribed covariance function. This can be achieved by inverting the steps in the previous section—we start with the firing rate r of a neuron. Since we assumed that $R(0)$—the variance of $y(t)$—is unity, we can invert *Eq. 12* to obtain θ in terms of r. Similarly, the autocovariance function $c(\tau)$ is related to $p(\tau)$ via *Eq. 14*. We can therefore use *Eq. 13* to determine $R(\tau)$ from the threshold θ and $c(\tau)$.

In particular, fixing θ and solving for $R(\tau)$ in terms of $p(\tau)$ using numerical integration gives a monotonic function that can be used to convert the autocovariance function of the spike train into the autocovariance of a continuous stochastic process (Fig. 2).

The resulting autocovariance $R(\tau)$ determines the coefficients a_k of the AR process (*Eqs. 9* and *10*). The continuous variable $y(t)$ obtained by filtering the Gaussian noise $n(t)$ using the filter defined by the coefficients a_k has autocorrelation $R(\tau)$, which equals the autocovariance when the variance of $y(t)$ is unity. By construction, applying the Heaviside function with threshold θ to $y(t)$ generates a spike train $x(t)$ with mean firing rate r and autocovariance $c(\tau)$.

Extension to a multivariate process

In this section we give a brief overview of the extension to populations of spike trains with firing rates $\vec{r} = (r_1, r_2, \ldots, r_N)$ and cross-correlation functions $c_{i,j}(\tau)$. The multivariate point process is thus defined through the $(N \times L) \times (N \times L)$ block–Toeplitz matrix C (*Eq. 5*). A vector of thresholds $\vec{\theta} = \{\theta_1, \theta_2, \ldots, \theta_N\}$ is calculated using *Eq. 12* based on the vector of firing rates $\vec{r}$. The covariance matrix of the continuous stochastic process is obtained by applying *Eq. 13* to each pair (i, j) for all time lags $\vec{\tau} = (\tau_1, \tau_2, \ldots, \tau_{L-1})$ to get the covariances $R_{i,j}(\tau)$. We obtain the following $(N \times L) \times (N \times L)$ block–Toeplitz covariance matrix

$$T = \begin{bmatrix} R\,(0) & R\,(1) & \cdots & R\,(L-2) & R\,(L-1) \\ R\,(1) & R\,(0) & \cdots & R\,(L-3) & R\,(L-2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ R\,(L-2) & R\,(L-3) & \cdots & R\,(0) & R\,(1) \\ R\,(L-1) & R\,(L-2) & \cdots & R\,(1) & R\,(0) \end{bmatrix}$$

where each $R(\tau)$ is an $N \times N$ matrix [note that $R_{i,j}(\tau) = R_{j,i}(-\tau)$]

$$R(\tau) = \begin{bmatrix} 1 & R_{1,2}\,(\tau) & \cdots & R_{1,N-1}\,(\tau) & R_{1,N}\,(\tau) \\ R_{2,1}\,(\tau) & 1 & \cdots & R_{2,N-1}\,(\tau) & R_{2,N}\,(\tau) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ R_{N-1,1}\,(\tau) & R_{N-1,2}\,(\tau) & \cdots & 1 & R_{N-1,N}\,(\tau) \\ R_{N,1}\,(\tau) & R_{N,2}\,(\tau) & \cdots & R_{N,N-1}\,(\tau) & 1 \end{bmatrix}$$

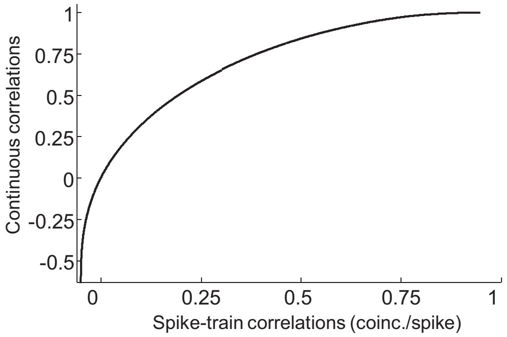

FIG. 2. Correspondence between the spike-train correlations and the Gaussian process. The transformation from point-process correlations to continuous correlations is nonlinear and depends on the threshold θ. This example is for a neuron firing at 50 Hz.

[2] For simplicity we fix the variance of y to be 1 because it is just a scaling factor.

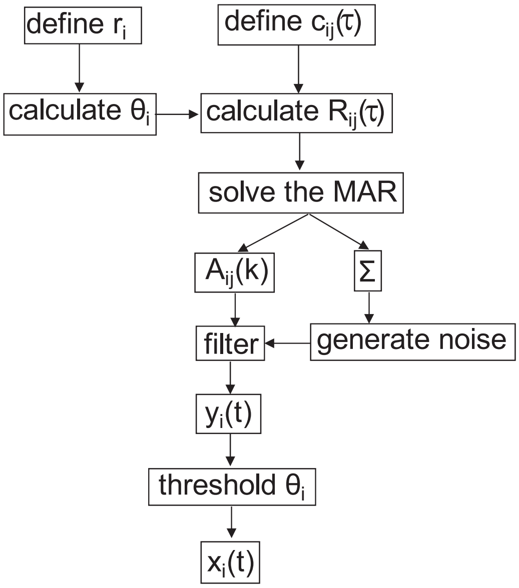

FIG. 3. Description of the multivariate spike-train generation algorithm. The inputs to the algorithm are the neural firing rates r_i and the pairwise cross-correlations $c_{ij}(\tau)$. The thresholds θ_i depend on r_i and the time discretization (Δt). The next step is the calculation of the continuous cross-correlations $R_{ij}(\tau)$ and the solving of the multivariate autoregressive (AR) model equations. Gaussian noise is generated with covariance Σ and filtered using the AR matrix parameters $A_{ij}(k)$. Finally, the output spike trains $x_i(t)$ are obtained after thresholding the continuous variables $y_i(t)$ at θ_i.

A multivariate autoregressive process of order L is defined as

$$\vec{y}(t) = \sum_{i=1}^{L} A_k \cdot \vec{y}(t-i) + \vec{n}(t) \tag{15}$$

where each A_k is a symmetric $N \times N$ matrix. We again have to solve

$$A = T^{-1} \cdot R \tag{16}$$

with A and R given by

$$A = \begin{bmatrix} A_1 \\ A_2 \\ \vdots \\ A_{L-1} \\ A_L \end{bmatrix} \quad R = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(L-1) \\ R(L) \end{bmatrix} \tag{17}$$

The covariance matrix of the multivariate Gaussian random variable $\vec{n}(t)$ is

$$\Sigma = R(0) - \sum_{i=1}^{L} A_i \cdot R(i) \tag{18}$$

Finally, to obtain the spike trains $\vec{x}(t)$ the vector variable $\vec{y}(t)$ is thresholded using the vector of thresholds $\vec{\theta}$.

Summary of the algorithm to generate a multivariate process

We summarize the steps used to generate spike trains with given spatiotemporal structure. The inputs to the algorithm are the desired cross-correlations $c_{i,j}(\tau)$ for every pair of neurons (i, j) for a finite number of time lags $\tau \in \{0, 1, \ldots, L\}$ and the firing rates of each neuron r_i. An outline schematic of the algorithm is shown in Fig. 3.

In the simulations, the spike-train covariances were normalized by the geometric mean of the firing rates

$$c_{ij}(\tau) = \frac{\langle x_i(t)x_j(t-\tau)\rangle - \langle x_i(t)\rangle\langle x_j(t)\rangle}{\sqrt{\langle x_i(t)\rangle\langle x_j(t)\rangle}}$$

For the continuous process we used $R(0) = 1$ (i.e., the continuous variables will be Gaussian with unit variance). Thus the covariances equal the correlations and are measured in units of coincidences/spike. Using this normalization, the probability of having coincident spikes in two bins τ time units apart for two neurons i, j is $p_{i,j}(\tau) = c_{i,j}(\tau)\sqrt{r_i \cdot r_j} + r_i r_j$ (for simplicity we set $\Delta t = 1$ in all simulations). We chose this normalization because it is one of the most commonly used in experimental work (e.g., Bair et al. 2001; Kohn and Smith 2005). Using *Eqs. 12* and *13* we calculate the thresholds θ_i. The continuous-process correlations $R_{i,j}(\tau)$ are obtained by finding bivariate Gaussian distributions with mass above θ equal to $p(\tau)$ (*Eq. 13*).

To obtain the parameters of the multivariate autoregressive process, $A_{i,j}(k)$ and Σ, we solve the linear system in *Eqs. 16* and *18*.

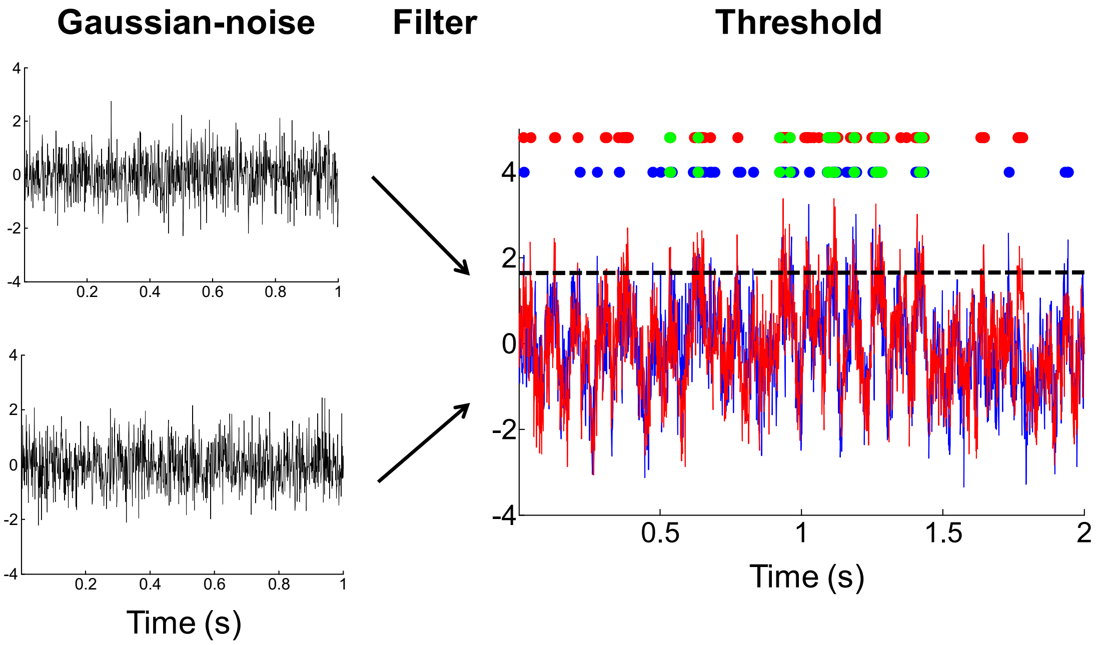

FIG. 4. Example of filtering and thresholding a neural pair. Two Gaussian random vectors are filtered using an AR model. The resulting random variables (blue and red curves) are temporally and spatially correlated. The green dots represent the simultaneous firing of the 2 neurons.

To obtain the continuous variables $y_i(t)$, we filter the time-independent Gaussian random noise $n_i(t)$ with covariance matrix Σ, through the AR with parameters $A_{i,j}(k)$ using *Eq. 15* (Fig. 4). Finally, the population spike trains $x_i(t)$ are obtained by thresholding $y_i(t)$ with θ_i for each neuron i.

Notes on the implementation: complexity and memory requirements

We next discuss the computational costs and optimization strategies available in certain special cases. After converting the block–Toeplitz covariance matrix C into the continuous-process covariance matrix T, we have to solve a linear system $A = T^{-1} \cdot R$. The matrix T is $NL \times NL$, whereas A and R are $NL \times L$ (N is the number of neurons and L is the order of the AR). The Cholesky factorization of T is $O(N^3L^3)$, but the subsequent operations needed to solve the linear system are only $O(N^2L^2)$. More efficient implementations are available by taking advantage of the block–Toeplitz structure of T (it has only N^2L different entries) and many algorithms are applicable to positive-definite block–Toeplitz matrices (Akaike 1973; Ammar 1996; Ammar and Gragg 1988; Kamm and Nagy 2000) and the special case of Toeplitz–block–Toeplitz linear systems (Wax and Kailath 1983).

After obtaining the parameter matrix A we have to generate Gaussian noise with covariance Σ and apply the autoregressive equations (*Eq. 15*). Defining M as the number of time steps that will be computed, the number of computations needed to obtain $\vec{y}(t)$ is $O(N^2LM)$. Therefore the computational time is linear in the spike-train length but quadratic in the number of neurons. Alternatively, the AR process can be written as an infinite-impulse response (IIR) filter (Shanmugan and Breipohl 1988). In this case, the computation of $\vec{y}(t)$ is $O(N^2M \log M)$ for $L \ll M$. The last equation indicates that the computational time is independent of the maximum correlation length L when using the frequency-domain approach. With programs such as Matlab the use of loops may be less efficient than implementing the filter approach, especially when $L \gg \log(M)$. It is also possible to obtain $\vec{y}(t)$ using a moving-average process (FIR filter) instead of an AR process (Krumin and Shoham 2009).

Optimizations

In many practical cases the spatiotemporal covariances have additional structure. An important case is when the covariance is the product of a function of the neuron number and a function of the time lag

$$R_{i,j}(\tau) = f(i,j)g(\tau) \quad \text{with} \quad f(i,j) = f(j,i) \quad i \neq j$$

$$R_{i,i}(\tau) = d(\tau)$$

Note that R will be symmetric since $R_{ij}(\tau) = R_{ji}(-\tau)$ implies $g(\tau) = g(-\tau)$. The last equation allows us to have a different time dependence for the cross-correlations and the autocorrelations. Using the Kronecker tensor product we can write T compactly

$$T = G \otimes F + (D - G) \otimes I_N$$

where G and D are $L \times L$ Toeplitz matrices with row entries $g(\tau)$ and $d(\tau)$, respectively ($\tau = 0, 1, \ldots, L - 1$), and F is an $N \times N$ symmetric matrix with values $[F]_{ij} = f(i,j)$. We also define the matrices R_{off} and R_{diag} as the $L \times L$ Toeplitz matrices, with row entries $g(\tau)$ and $d(\tau)$, respectively, with $\tau = 1, 2, \ldots, L$.

We can factorize $F = P\Lambda P^t$ with the columns of P having the eigenvectors of F and Λ the eigenvalues (λ_i) on its diagonal. We define the $N \times N$ matrices Λ_i to have a single nonzero entry $[\Lambda_i]_{ii} = \lambda_i$. The linear system $TA = R$ takes the form

$$\left[\sum_{i=1}^{N}(G \otimes P\Lambda_i P^t) + (D - G) \otimes I_K\right]\left[\sum_{i=1}^{N}[A_{off}(i) \otimes P\Lambda_i P^t] + A_{diag} \otimes I_K\right] = \left[\sum_{i=1}^{N}(R_{off} \otimes P\Lambda_i P^t) + R_{diag} \otimes I_K\right]$$

where $A_{off}(i)$ and $A_{diag}(i)$ are the parameter matrices to be determined. Using the fact that $PP^t = P^tP = I_N$ and $\Lambda_i\Lambda_j = 0$ for $i \neq j$

$$A_{diag} = (D - G)^{-1}R_{diag} \tag{19}$$

$$A_{off}(i) = (\lambda_i G + D - G)^{-1}(R_{off} - GA_{diag}) \tag{20}$$

Instead of having to solve one $NL \times NL$ linear system we have to solve $N + 1$ Toeplitz-structured systems of size $L \times L$. For such systems the calculations are linear in the number of neurons and the memory requirement for the AR parameters drop to $(N + 1)L$.

We also consider the special case of a population consisting of N_{sub} subpopulations, where subpopulation i contains n_i neurons. The cross-correlations for neurons within a subpopulation are identical, whereas neurons in different subpopulations are uncorrelated. The matrix F can then be written as

$$F = \begin{pmatrix} 1_{n_1} & & & \\ & 1_{n_2} & \ddots & \cdots \\ & \cdots & 1_{n_{N_{sub}-1}} & \\ & & & 1_{n_{N_{sub}}} \end{pmatrix}.$$

In this case all the eigenvalues of F are 0 except for the N_{sub} eigenvalues $(\lambda_1, \lambda_2, \ldots, \lambda_{N_{sub}}) = (n_1, n_2, \ldots, n_{N_{sub}})$. In this way, *Eq. 20* reduces to

$$A_{off}(i) = [G(n_i - 1) - D]^{-1}(R_{off} - GA_{diag}) \quad \text{for} \quad i = 1, 2, \cdots, N_{sub} \tag{21}$$

There are $N_{sub} + 1$ equations to solve and thus the number of linear systems depends on the number of subpopulations but does not depend on the number of cells within each subpopulation.

Generalization to nonstationary spike trains

In the previous sections we assumed that the firing rate and the correlations do not change in time. A spike train with time-dependent firing rate $r(t)$ can be obtained by using a time-dependent threshold and solving *Eq. 12* at each time step. The continuous-process autocovariance $R(t_1, t_2) = \text{cov}\,[y(t_1), y(t_2)] = \langle y(t_1) \cdot y(t_2) \rangle$ is obtained by solving a time-dependent version of *Eq. 13*

$$p(t_1, t_2) = \int_{\theta(t_1)}^{\infty}\int_{\theta(t_2)}^{\infty} \frac{1}{2\pi\sqrt{1 - R(t_1, t_2)^2}}\, e^{-\left\{\frac{a^2 + b^2 - 2R(t_1,t_2)ab}{2[1 - R(t_1,t_2)^2]}\right\}} \mathrm{d}a\mathrm{d}b$$

$$p(t_1, t_2) = \langle x(t_1)x(t_2) \rangle = c(t_1, t_2) + r(t_1)r(t_2)$$

The stationary autocovariance is a special case: $R(t_1, t_2) = R(t_2 - t_1) = R(\tau)$. Note that $R(t_1, t_2)$ may be time dependent, even if the point-process autocovariance $c_{i,i}(\tau)$ is not because $R(t_1, t_2)$ depends on the threshold $\theta(t)$.

The next step is to generate a continuous process with zero mean and covariance $R(t_1, t_2)$. We generate the nonstationary continuous process by means of a time-varying autoregressive process

$$y(t) = \frac{1}{a_0(t)}\left[-\sum_{k=0}^{L} a_k(t) \cdot y(t - i) + n(t)\right] \tag{22}$$

where the parameters $a_k(t)$ are time dependent. For a given, nonstationary cross-covariance $R(t_1, t_2)$, these parameters can be obtained by solving the nonstationary Yule–Walker equations (Hallin and Ingenbleek 1983)

$$A(t) = T(t)^{-1} \cdot V(t) \tag{23}$$

$$A(t) = a_0(t) \begin{bmatrix} a_1(t) \\ a_2(t) \\ \vdots \\ a_L(t) \end{bmatrix} \quad \text{with} \quad a_0(t) = \sqrt{1 - A(t)^t V(t)} \tag{24}$$

First we solve *Eq. 23* to obtain $A(t)$ and then calculate $a_0(t)$ in *Eq. 24* to get the parameters $a_1(t), a_2(t), \ldots, a_L(t)$. The matrices $T(t)$ and $V(t)$ are defined as

$$[T(t)]_{ij} = R(t, t-i)R(t, t-j) - R(t-i, t-j) \quad \text{for} \quad i, j = 1, 2, \cdots, L \tag{25}$$

$$[V(t)]_i = R(t, t-i) \quad \text{for} \quad i = 1, 2, \cdots, L \tag{26}$$

Note that in the stationary case we used $a_0 = 1$ and found the SD using *Eq. 8*. For consistency with Hallin and Ingenbleek (1983), here we let $a_0(t)$ be a normalizing factor and let the SD of $n(t)$ equal 1.

Generalization to STAs of the LFP

The generation of continuous random variables $\vec{y}(t)$ with given spatiotemporal correlations is an intermediate step in our algorithm to create spike trains. However, the vector of continuous variables can also be used to directly simulate a collection of local field potentials (LFPs) with specific spectral components. In this section we show how we can extend our algorithm to simulate given spike-triggered averages (STAs) of the LFP.

Let $y_{LFP}(t)$ be the LFP signal and $x(t)$ be the spike train associated with the continuous variable $y_{spike}(t)$. As before, $x(t)$ is obtained by thresholding $y_{spike}(t)$ at θ (*Eq. 11*). The STA is defined as

$$STA(\tau) = \frac{\sum_{t_k} y_{LFP}(t_k + \tau)}{N_{spikes}} \quad \text{for} \quad k = 1, 2, \cdots N_{spikes}$$

where t_k is the time of each spike in $x(t)$. We can find the correlation $R(\tau)$ between the bivariate Gaussian distribution $[y_{LFP}(t), y_{spike}(t)]$ to obtain a desired value of $STA(\tau)$. We have to find the distribution of $y_{LFP}(t + \tau)$ given that a spike occurred at time t, i.e., $y_{spike}(t) \geq \theta$ (Cox and Wermuth 1999). The corresponding density is given by

$$f_{STA}(x) = \frac{\int_{\theta}^{\infty} f_{y_{spike}}(\phi) f_{y_{LFP}|y_{spike} \geq \theta}(x) \mathrm{d}\phi}{\int_{\theta}^{\infty} f_{y_{spike}}(x) \mathrm{d}x} \tag{27}$$

where f_{STA}, $f_{y_{spike}}$, and $f_{y_{LFP}|y_{spike} \geq \theta}$ are Gaussian probability density functions (pdfs) of the corresponding random variables. The conditional pdf $f_{y_{LFP}|y_{spike} \geq \theta}$ has mean $\mu_{y_{LFP}|y_{spike} \geq \theta}$ and variance $\sigma^2_{y_{LFP}|y_{spike} \geq \theta}$

$$\mu_{y_{LFP}|y_{spike} \geq \theta} = \mu_{y_{LFP}} + R(\tau)\sigma_{y_{LFP}}(\theta - \mu_{y_{spike}}) \tag{28}$$

$$\sigma^2_{y_{LFP}|Y_{spike} \geq \theta} = \sigma^2_{y_{LFP}}[1 - R(\tau)^2] \tag{29}$$

We use $\sigma_{y_{spike}} = 1$. Calculating the integral in *Eq. 27* we obtain

$$\mu_{STA(\tau)} = R(\tau)\sigma_{y_{LFP}} K(\theta) \quad K(\theta) = \frac{e^{-\frac{\theta^2}{2}}}{\int_{\theta}^{\infty} e^{-\frac{x^2}{2}} dx} \tag{30}$$

In this way, we can generate the STA of the LFP (*Eq. 30*) by choosing an appropriate $\sigma_{y_{LFP}}$ and finding the correlation function $R(\tau)$.

Conductance-based model

We implemented a standard conductance-based integrate-and-fire model to study the impact of correlated input on the firing properties of downstream neurons. The membrane voltage $V(t)$ is described as follows

$$C\frac{\mathrm{d}}{\mathrm{d}t}V(t) + [V(t) - V_r]G_l + [V(t) - V_e]G_e(t) + [V(t) - V_i]G_i(t) = 0 \tag{31}$$

where C is the membrane capacitance; G_l is the leakage conductance; V_r is the resting potential; and V_e and V_i are the excitatory and inhibitory reversal potentials, respectively. The excitatory and inhibitory conductances $G_e(t)$ and $G_i(t)$ are sums of unitary synaptic events, $g_e(t)$ and $g_i(t)$

$$G_e(t) = \sum_j g_e(t - t_j) \quad G_i(t) = \sum_k g_i(t - t_k)$$

for spikes arriving at t_j and t_k.

The unitary synaptic events were modeled by alpha-functions with peak conductance B_e and B_i and time constants τ_e and τ_i

$$g_{e,i}(t) = B_{e,i} e \frac{t}{\tau_{e,i}} e^{\frac{t}{\tau_{e,i}}} H(t) \tag{32}$$

where $H(t)$ is the Heaviside step function. The parameters and the model are the same as those in Kuhn et al. (2004): $C = 250$ pF; $G_l = 1/60$ ms; $V_r = -70$ mV; $V_e = 0$ mV; $V_i = -75$ mV; $B_e = 7.1$ ns; $B_i = 3.7$ ns; $\tau_e = 0.2$ ms; $\tau_i = 2$ ms. An action potential is elicited whenever the membrane voltage $V(t)$ crosses the threshold $V_{th} = -50$ mV. An absolute refractory period is obtained by holding $V(t)$ at qj$V_{reset} = -60$ mV for $\tau_{refr} = 2$ ms after a spike. We used a time step of 0.1 ms for the numerical simulations.

RESULTS

To test our algorithm we generated a variety of spike trains with different spatiotemporal structures. We first present single spike trains with prescribed autocorrelations. These are followed by examples of multivariate spike trains that illustrate the impact of pairwise correlation on the behavior of the neural population.

Single-cell examples

We briefly summarize the steps used to generate a single spike train with a given cross-correlation (for details see METHODS). In the simplest case, a Poisson spike train with mean firing rate r can be obtained by discretizing time into bins of length Δt. A spike is generated with probability $r\Delta t$ in each bin independently of all other events. We can implement a Poisson spike train by thresholding a continuous-in-space, discrete-in-time Gaussian stochastic process $y(t)$. A spike is elicited whenever the Gaussian random variable is higher than a threshold θ (*Eq. 11*). Using *Eq. 12* we can obtain the appropriate θ for a spike train with mean firing rate r.

More generally, for spike trains with complex temporal structure, the probability of having a spike in a given time bin $r\Delta t$ becomes dependent on past spikes. To capture this structure we have to generate a stochastic process $y(t)$ with specific autocovariance function. After finding the threshold (θ) we have to map the desired spike-train autocovariance, $c(\tau)$, into a continuous process covariance function $R(\tau)$. This transformation is computed for every time lag in such a way that the application of the threshold θ to a pair of continuous variables $[y(t), y(t + \tau)]$ with a bivariate Gaussian distribution gives the desired point-process autocovariance $c(\tau)$ (Figs. 1 and 2; *Eq. 13*).

The second step consists of generating a continuous random variable $y(t)$ with prescribed correlation $R(\tau)$. This random variable follows a Gaussian distribution with mean 0 and is thus fully described by $R(\tau)$. We have to design a filter to transform independent random Gaussian noise $n(t)$ to $y(t)$. This is accomplished by generating an AR process (*Eq. 6*). The parameters of the AR process are calculated using *Eqs. 8* and *10*. Finally, $y(t)$ is obtained by applying the AR process equations (*Eq. 6*).

The last step involves the only nonlinear operation in the algorithm: we assume a spike occurs at time t, so that $x(t) = 1$, whenever $y(t) > \theta$. In this way, we obtain a spike train $x(t)$ with the specified firing rate r and correlation $c(\tau)$. The explicit representation using a continuous function $y(t)$ and a threshold θ is similar to the ubiquitous integrate-and-fire model: $y(t)$ plays the role of the scaled membrane potential, which elicits a spike when it crosses the threshold θ.

To demonstrate the algorithm we provide several illustrative examples. First, we generated a spike train with vanishing autocorrelation, except at time $T = 10$ ms (for all simulations $\Delta t = 1$ ms; see Fig. 5*A*). In Fig. 5*B* we present a collection of simulated spike trains aligned to the spike occurring at the origin. Trials were sorted by the time of the second spike (red dots) and an increase in spiking probability at $T = 10$ ms is clearly visible. This is precisely what we expect, in that the autocorrelation is directly related to the conditional probability of a spike at time $T = 10$ ms, given a spike at *time 0*.

In the second example we examined how the cross-correlation estimated from the simulated spike trains converges to the predetermined cross-correlation with an increase in the length of the simulation and the amount of data generated. We simulated a spike train with exponentially decaying autocorrelation and set the probability of coincident spikes at $t \pm 1$ ms to 10% of chance (see following text for an example of absolute refractory period). In Fig. 5*C*, the red and black curves correspond to the autocorrelation function obtained from simulations (6.4×10^6 bins) and theory, respectively. To assess the precision of the algorithm we calculated the mean squared error (RMS) for spike trains of different lengths (Fig. 5*D*). The error decayed as $1/\sqrt{N}$, as expected for an unbiased estimator of the correlation due to the central limit theorem (Brody 1999a). This indicates that only the finiteness of the data accounts for the observed RMS error.

In the third example, we approximated a spike train with absolute refractory period of 1 ms. Our algorithm does not include any mechanism to remember when a spike was elicited to prevent the firing of another action potential during the absolute refractory period. However, under certain conditions, such as short refractory periods or low firing rates, we can obtain a good approximation to absolute refractory periods. To ensure the absence of pairs of spikes in adjacent time bins, it is necessary that $R(\pm 1) = -1$ for the corresponding AR process. However, unless $R(\pm 1) > -0.5$ the covariance matrix of $y(t)$, $y(t + 1), \ldots, y(t + L - 1)$ may not be positive definite and thus the AR process cannot be generated. However, it is

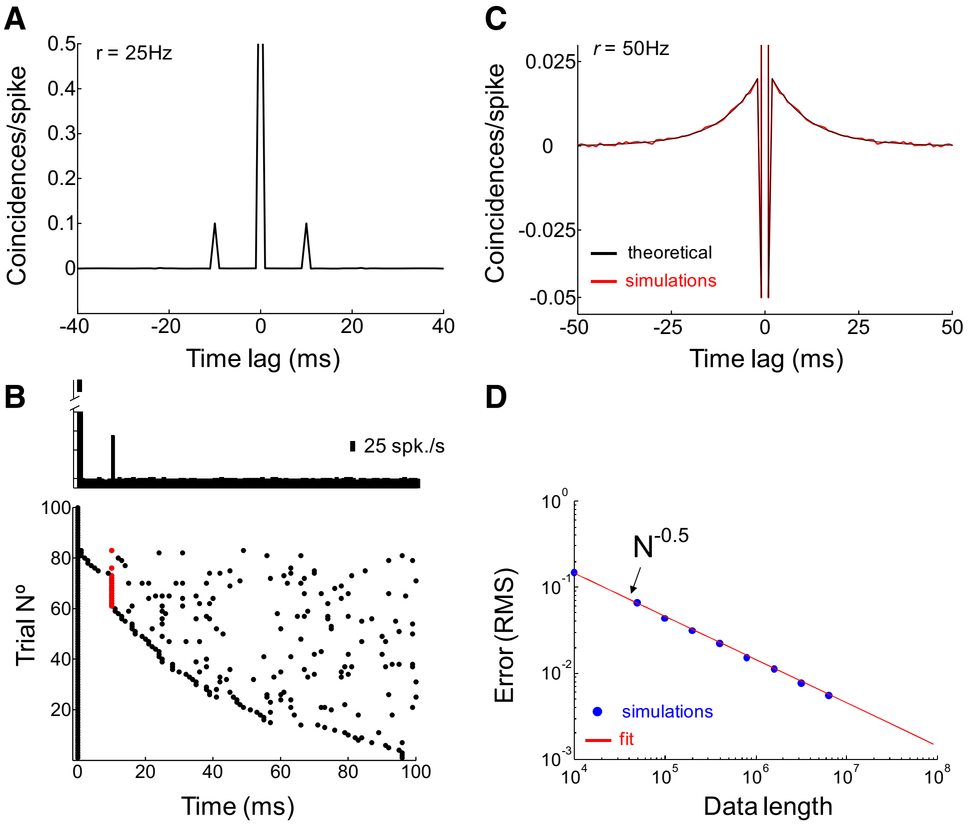

FIG. 5. Examples of individual spike trains with specified autocorrelations. *A* and *B*: the autocorrelation is 0 everywhere except at a time $T = 10$ ms. *B*: peristimulus time histogram and raster sorted by the occurrence of the first spike in each trial. The red dots correspond to spikes occurring 10 ms after the first spike. The probability of spikes occurring at time t and $t + \tau$ is at chance, except for $T = 10$ ms. *C*: the convergence of the simulated to the desired autocorrelation was examined by generating a spike train with exponentially decaying autocorrelation function ($\tau_{decay} = 10$ ms), peak correlation of 0.02 coincidence/spike, a firing rate of 50 Hz, and a decreased chance of firing at ± 1 ms. The black curve corresponds to the desired autocorrelation function; the red curve was obtained by simulation. *D*: the root-mean-squared (RMS) error as a function of data length. The RMS decays as $1/\sqrt{N}$ (red line), which is equal to the expected RMS in the presence of finite data due to the central limit theorem. This indicates that there are no systematic errors in the algorithm.

possible to generate processes with large relative refractory periods. The degree of refractoriness that can be achieved depends on the firing rate of the neuron and the discretization Δt. For a Poisson spike train with a firing rate of 25 Hz the probability of having two consecutive spikes is $(0.025)^2 = 6.25 \times 10^{-4}$ (Fig. 6*A*). We generated spike trains where this probability is reduced to about 4×10^{-6}, a decrease of nearly two orders of magnitude compared with chance. We show that the distribution of the ISI of an inhomogeneous Poisson spike train with absolute refractory period of 1 ms fits the empirical histogram well (Fig. 6*B*; the *inset* is a zoom of the histogram in the main figure).

We explored how well the algorithm could generate longer refractory periods. We derived the minimum correlation coefficient of the continuous process that could be achieved as a function of the length of the refractory period (Fig. 6*C*; see the APPENDIX). The minimum correlation coefficient $R_{\min}$ is inversely related to the length of the refractory period m, as $R_{\min} = -0.5/m$. The probability of having a spike during the refractory period depends not only on $R_{\min}$, but also on the firing rate of the neuron. We defined the refractoriness index as the ratio of the probability of having a spike during the refractory period to the probability of the same event for a homogeneous Poisson process. In Fig. 6*D* we plot the refractoriness index for different firing rates and refractory period lengths. The algorithm is better at generating refractory periods for low firing rates and short refractory periods.

Cortical responses are typically nonstationary. In sensory cortices, neurons respond to external stimulation with a fast increase in the firing rate followed by a slower adaptation response (e.g., Muller et al. 1999). To demonstrate that our algorithm can generate such responses we simulated a neuron with a nonstationary firing rate (Fig. 7*A*), but stationary cross-correlation function. The simulation was started at a baseline firing rate $r_{baseline} = 5$ Hz. After 50 ms, the firing rate was first set to increase exponentially with rate $\tau_{rise} = 10$ ms and then set to exponentially decay back to baseline with rate $\tau_{decay} = 30$ ms. In between, the peak firing rate was set at $r_{peak} = 30$ Hz. The firing rate was therefore described as

$$r(t) = \begin{cases} r_{baseline} & 0 \le t < 50 \text{ ms} \\ r_{baseline} + (r_{peak} - r_{baseline})z\left[1 - e^{-\left[\frac{(t-50)}{\tau_{rise}}\right]}\right]e^{\left[\frac{(t-50)}{\tau_{decay}}\right]} & 50 \le t < 300 \text{ ms} \end{cases}$$

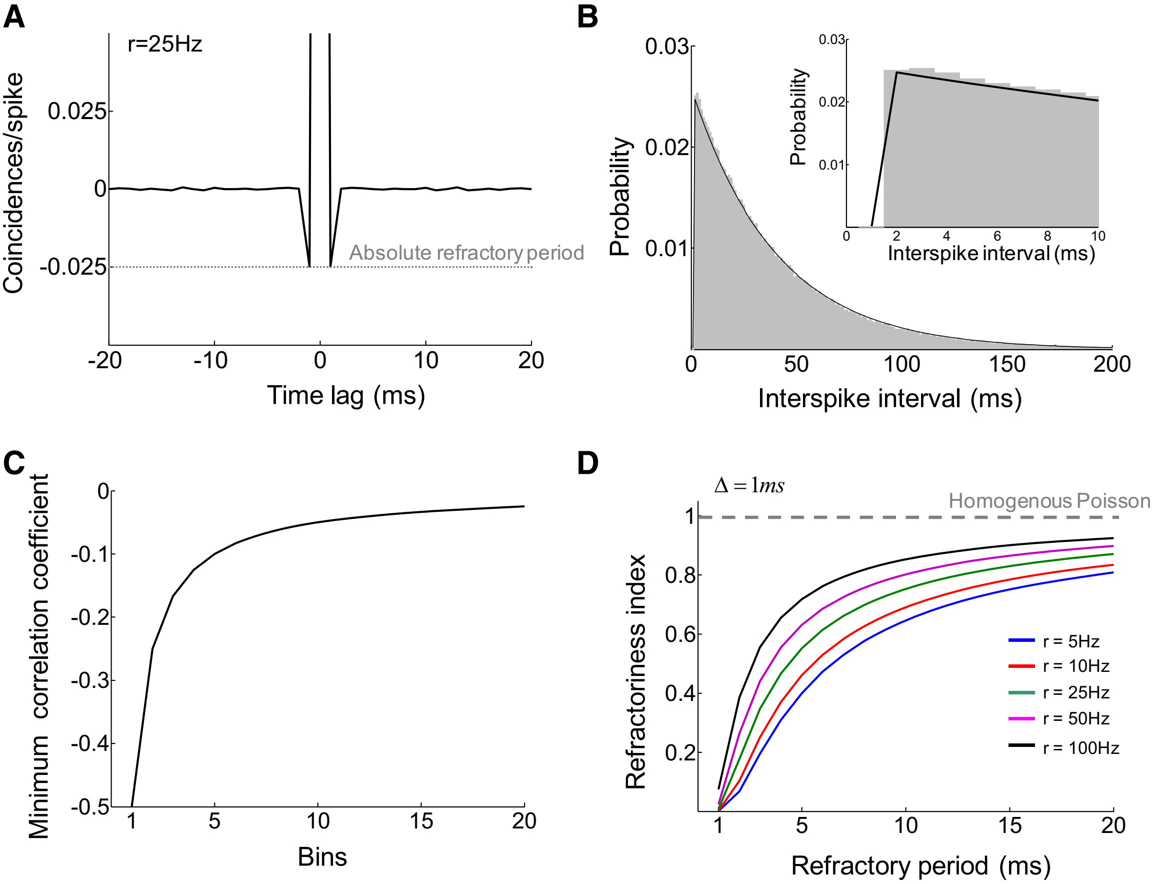

FIG. 6. Simulation of inhomogeneous Poisson trains with absolute refractory periods. *A*: autocorrelation function of a simulated spike train with an absolute refractory period of 1 ms and a firing rate of 25 Hz. *B*: histogram of interspike intervals (ISIs, gray bars) and theoretical fit (black lines). The *inset* is a zoom of the main figure. *C*: the continuous-process minimum correlation coefficient as a function of the length of the refractory period (i.e., number of bins with reduced spiking probability after a spike occurred at time *t*). The target autocorrelation is different from zero only in the first *m* bins. For a given number of bins we find the minimum correlation that yielded a positive definite correlation matrix. The equation of the curve is $-0.5/m$. *D*: refractoriness index as a function of the refractory period length for different firing rates. The refractoriness index is defined as the ratio of the probability of having a spike during the refractory period to the probability of the same event in a homogeneous Poisson spike train. The refractoriness index is 0 for an absolute refractory period and 1 for a homogeneous Poisson spike train.

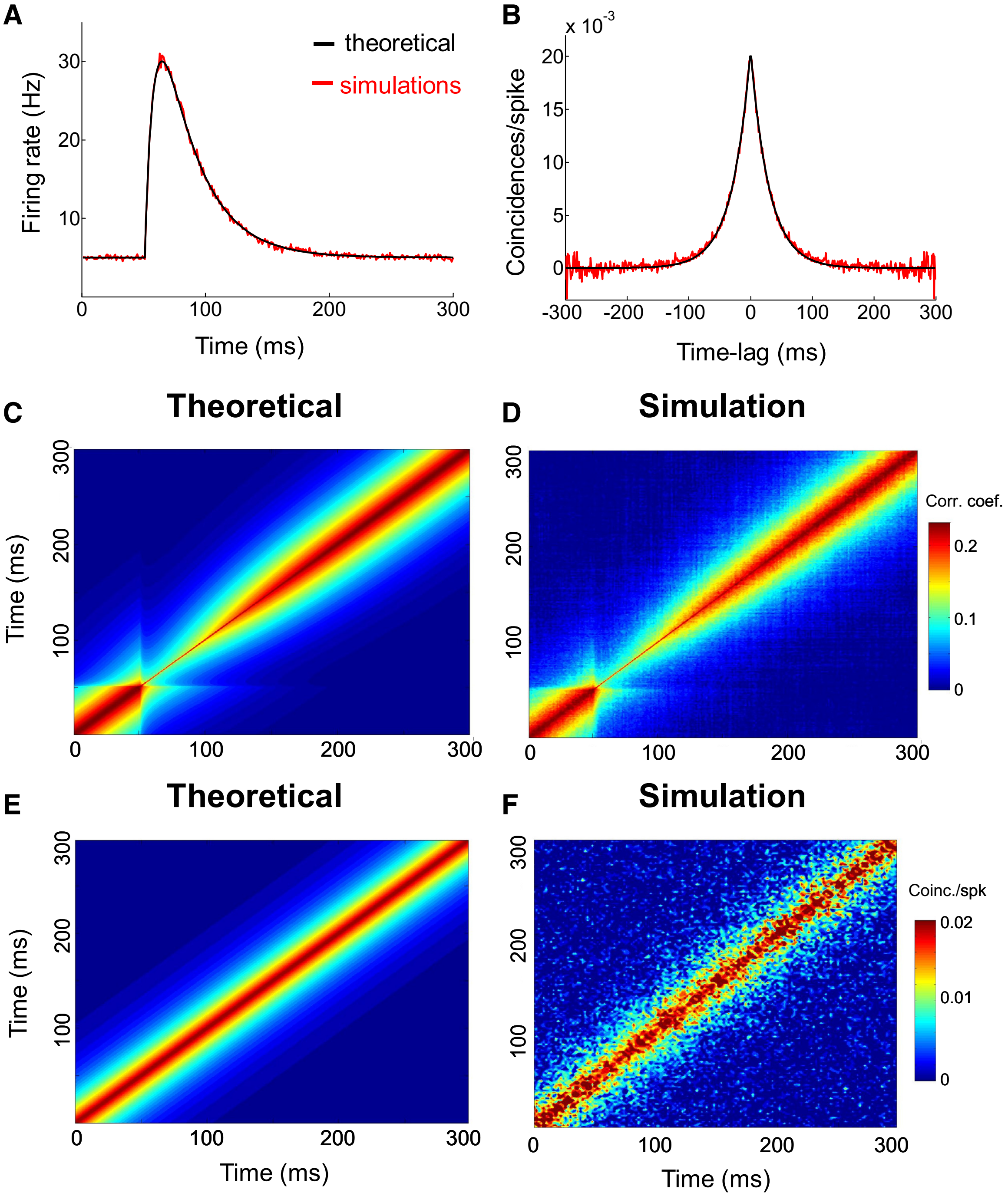

FIG. 7. Example of a neuron with time-varying firing rate and specified autocorrelation. *A*: simulated and theoretical firing rate as a function of time. *B*: simulated and theoretical autocorrelation. The autocorrelation was time independent: $C(t_1, t_2) = C(|t_2 - t_1|) = C(\tau)$. *C* and *D*: autocorrelation function $R(t_1, t_2)$ for the continuous nonstationary process (*C*: theoretical, *D*: simulation). Since the continuous process depends on the nonstationary firing rate, the autocorrelation function $R(t_1, t_2)$ is nonstationary. *E* and *F*: autocorrelation function $C(t_1, t_2)$ for the point-process stochastic process (*E*: theoretical; *F*: simulation).

$$z = \frac{\tau_{decay}}{\tau_{decay} + \tau_{rise}} \left(\frac{\tau_{rise}}{\tau_{decay} + \tau_{rise}} \right)^{\frac{\tau_{rise}}{\tau_{decay}}}$$

where z is a normalization constant. This behavior emulates a typical neural response evoked by a stimulus at 50 ms: a fast phasic increase in firing rate followed by a slower adapted response.

The autocorrelation function was chosen to be time independent, with a peak of 0.02 coincidence/spike and an exponential decay rate of 30 ms (Fig. 7*B*). To generate nonstationary spike trains we used a time-dependent threshold $\theta(t)$ so that the underlying continuous Gaussian process, $R(t_1, t_2)$, was also time dependent. The main difference with the stationary process algorithm is the determination of time-dependent parameters of the autoregressive process. In Fig. 7, *C* and *D* we show that we can generate a nonstationary continuous Gaussian process using *Eqs. 22–26* (see METHODS). As in the case of the stationary process, we obtained spike trains by thresholding the continuous process at $\theta(t)$. Because we chose the autocorrelation to be stationary, after thresholding we obtained the prescribed spike-train autocorrelations $C(t_1, t_2)$ (defined as $C(t_1, t_2) = \text{cov}\,[x(t_1), x(t_2)]$; Fig. 7, *E* and *F*).

Multiple-cell examples

We next provide examples of both homogeneous and nonhomogeneous population activity. In the first two examples we focus on the temporal aspects of the correlation structure. In the last example, we illustrate the impact of spatial correlations in the activity of a presynaptic population on the firing of an afferent neuron. In all simulations we used an optimized version of the algorithm applicable when the spatial and temporal correlations are "decoupled" (see METHODS).

We first illustrate how correlations between cells can affect population activity in ways that are difficult to detect at the level of small groups of neurons, but may be significantly amplified in larger populations. To this effect, we simulated the activity of homogeneous neural populations of 200 neurons with exponentially decaying autocorrelations (Fig. 8*A*). In contrast, we chose the cross-correlations to be oscillatory, with frequency varying between 20 and 150 Hz (Fig. 8, *B–D*). As a consequence, no periodic patterns were observable in the activity of individual cells, whereas the pooled activity of many neurons clearly exhibited rhythmic activity at the predetermined frequency. As expected, the activity of individual cells displayed approximately constant power at all frequencies. However, the population activity (i.e., the summed response of all neurons) displayed a distinguishable peak at the chosen frequencies (Fig. 9*A*). We computed STAs of the pooled activity to determine whether single neurons were synchronized to the population activity. We illustrate for an example of an oscillatory frequency of 50 Hz that the STA exhibits oscillations at the frequency of the cross-correlation functions. To further characterize this behavior we computed the power spectrum of the STA to find peaks at the specified frequencies (Fig. 9*C*).

We note that LFPs are believed to represent the pooled activity of large populations. LFPs show strong oscillations in different frequency bands depending on brain area, behavioral state, and stimulus (e.g., Engel et al. 2001; Fries et al. 2007; Schroeder and Lakatos 2009). On the other hand, there are many electrophysiological reports (e.g., Bair et al. 2001; Kohn and Smith 2005) of single units that failed to find oscillatory cross-correlations. As this example illustrates, it is possible that oscillations in spike-train cross-correlograms are simply difficult to detect with limited data. However, even weak, oscillatory features that are swamped by noise and undetectable at the single-cell or small population level could be amplified at the level of larger populations and are thus detectable in the LFP.

We also studied the impact of temporal correlation in the activity of a presynaptic population on the firing rate of a postsynaptic neuron. In this simulation the autocorrelations and cross-correlations of spike trains in the simulated presynaptic population exhibited exponential decay in time (auto- and cross-correlations had the same parameters). We kept the area of the auto- and cross-correlogram fixed at 0.2 while varying the exponential decay constant between 1 and 100 ms (we used $\Delta t = 0.1$ ms). The area under the cross-correlogram is proportional to the Pearson spike-count correlation between two neurons (Bair et al. 2001; Cox and Isham 1980). Therefore the Pearson correlation coefficient was held constant, whereas the precision of synchrony between the cells was varied. The postsynaptic neuron also received inhibitory Poisson input at 1,000 Hz. Figure 10*A* shows some of the cross-correlation profiles of single cells in the presynaptic population. The pooled population activity ($n = 100$) displayed autocorrelation similar to that of individual cells (Fig. 10*B*), but

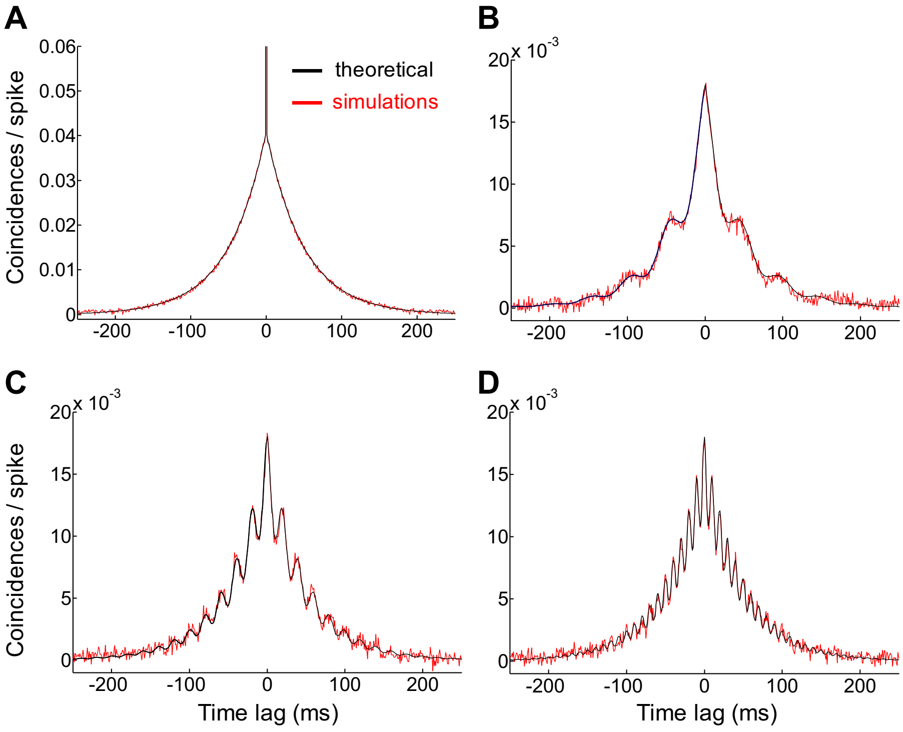

FIG. 8. Example of populations of artificial spike trains with exponentially decaying oscillatory cross-correlations, but only exponentially decaying autocorrelations. We generated spike trains for 200 neurons with exponentially decaying autocorrelations (τ_{decay} = 50 ms) and cross-correlations with different oscillatory frequencies ranging from 20 to 150 Hz, with a firing rate of 50 Hz. *A*: exponentially decaying autocorrelation function. *B–D*: cross-correlation functions for different oscillatory frequencies (*B*: 20 Hz; C: 50 Hz; *D*: 100 Hz). Theoretical curves are in black and simulated in red.

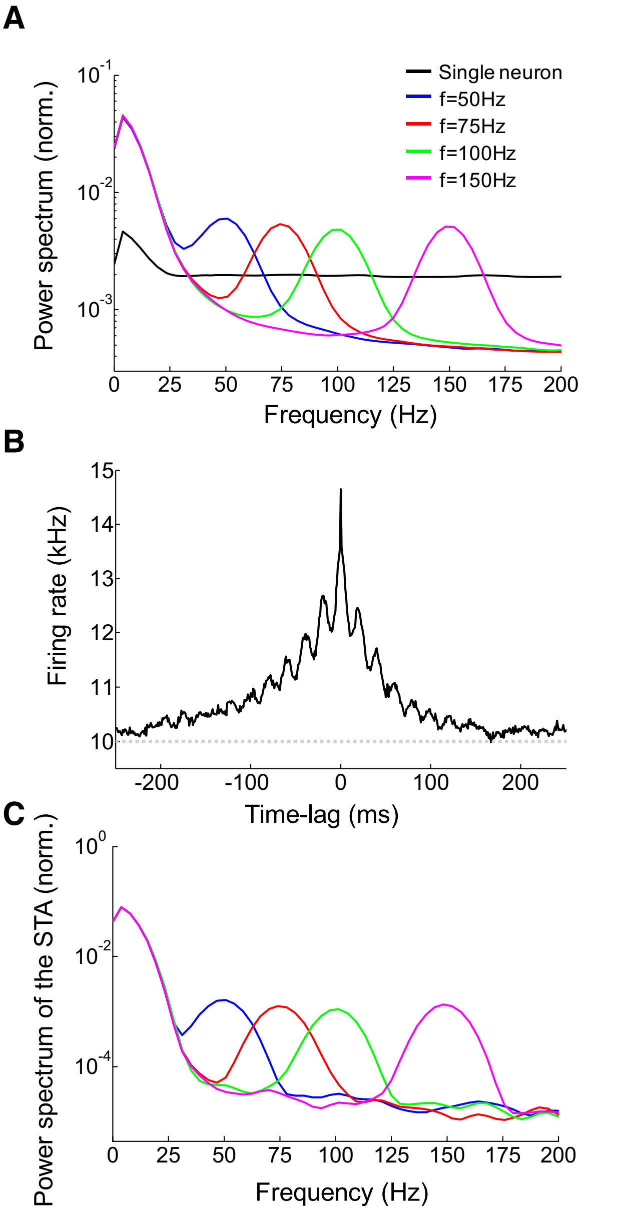

FIG. 9. Pooled activity of a neuronal population with exponentially decaying and oscillatory cross-correlations but without oscillatory autocorrelations. *A*: power spectrum of the population (pooled) activity. The power spectra of individual cells are flat (black line). However, the power spectrum analysis shows that the pooled activity has peaks at the oscillatory frequency of the cross-correlations. *B*: spike-triggered average (STA) of the pooled activity (frequency = 50 Hz). Although individual neurons display no oscillatory behavior in their cross-correlation, their spikes are synchronized to the pooled activity. The gray dotted line represents the population average firing rate. *C*: the power spectrum of the STA of the pooled activity shows peaks at each of the oscillatory frequencies. This analysis indicates that individual neurons fire in synchrony with the population at the cross-correlation oscillatory frequency.

increased magnitude. This increase was not due to an increased signal strength (i.e., increase in firing rates), but an amplification caused by effective averaging over correlated variables. The amplification of the autocorrelation function is explained by the fact that a random variable that is the sum of small correlated variables exhibits an autocorrelation approximately Nc, with N the number of variables and c the correlation magnitude.

Although the mean population firing rates were equal for all decay constants, the instantaneous population firing rate variability was greatly increased when the decay rate was smaller (i.e., the population was more synchronized; Fig. 10*C*). As expected, this caused an increase in the firing rate of the postsynaptic neuron (Fig. 10*D*; Salinas and Sejnowski 2000). Moreover, changes in the decay rate of input correlations influenced the variability of the output of the afferent neuron. As shown in Fig. 10*E* slower decay rates led to outputs that are more bursty. Indeed, an increased correlation timescale in the input led to broader synchronous events in the input and an increased probability that a threshold crossing will be closely followed by others in the afferent cell. We quantified the variability in the ISIs by computing their coefficient of variation (CV). We found that the output CV was close to 1 for tightly synchronous inputs. Longer decay rates in input correlations led to increased output burstiness and thus a higher CV. Although the impact of changes in decay rate on input CV (i.e., single-cell input) was negligible, the effect on output CV was considerable (Fig. 10*F*).

In the next example we generated different examples of heterogeneous populations to demonstrate the impact of variations in spatial correlation profile on the output of a postsynaptic neuron. We simulated several populations of 128 neurons, comprised of a different number of homogeneous subpopulations. The neurons within a subpopulation shared the same auto- and cross-correlations but were uncorrelated with cells in other subpopulations. The number of subpopulations ranged between 1 (a single subpopulation with statistically identical neurons and neuron pairs) and 128 (all neurons uncorrelated). The mean firing rate of each neuron was 20 Hz, whereas the auto- and cross-correlations were exponentially decaying functions with τ_{decay} = 10 ms, with the area of the cross-correlogram equal to 0.4. The structure of these populations allowed for the use of very efficient versions of the algorithm (see METHODS).

As shown previously (Fig. 10), neurons are sensitive to highly synchronized inputs (Kuhn et al. 2003; Salinas and Sejnowski 2000). However, if the input contains a high number of synchronous spikes, adding further synchronous spikes does not cause an increase in the firing rate of the output neuron (Kuhn et al. 2003). The distance of the resting membrane voltage to the threshold determines the optimal synchronization level at which the number of spikes in a synchronous volley is exactly sufficient to cross threshold and no spikes are "wasted." Another factor that affects the firing rate is the frequency of highly synchronous events. In homogeneous populations, the level of synchronization depends on the level of correlation between neurons. To show the interplay between synchronized neural populations and the frequency of synchronized events we generated eight different populations by modifying the number of subpopulations (from 1 to 128 in steps determined by powers of 2) and used their output to drive a conductance-based neural model. Increasing the number of subpopulations caused a higher rate of synchronized events because the synchronous events in each subpopulation were independent. On the other hand, there were also fewer neurons in each subpopulation causing a reduction in the impact of the individual synchronous events.

In this way, we kept the same mean population firing rate, but altered the probability and the number of neurons involved

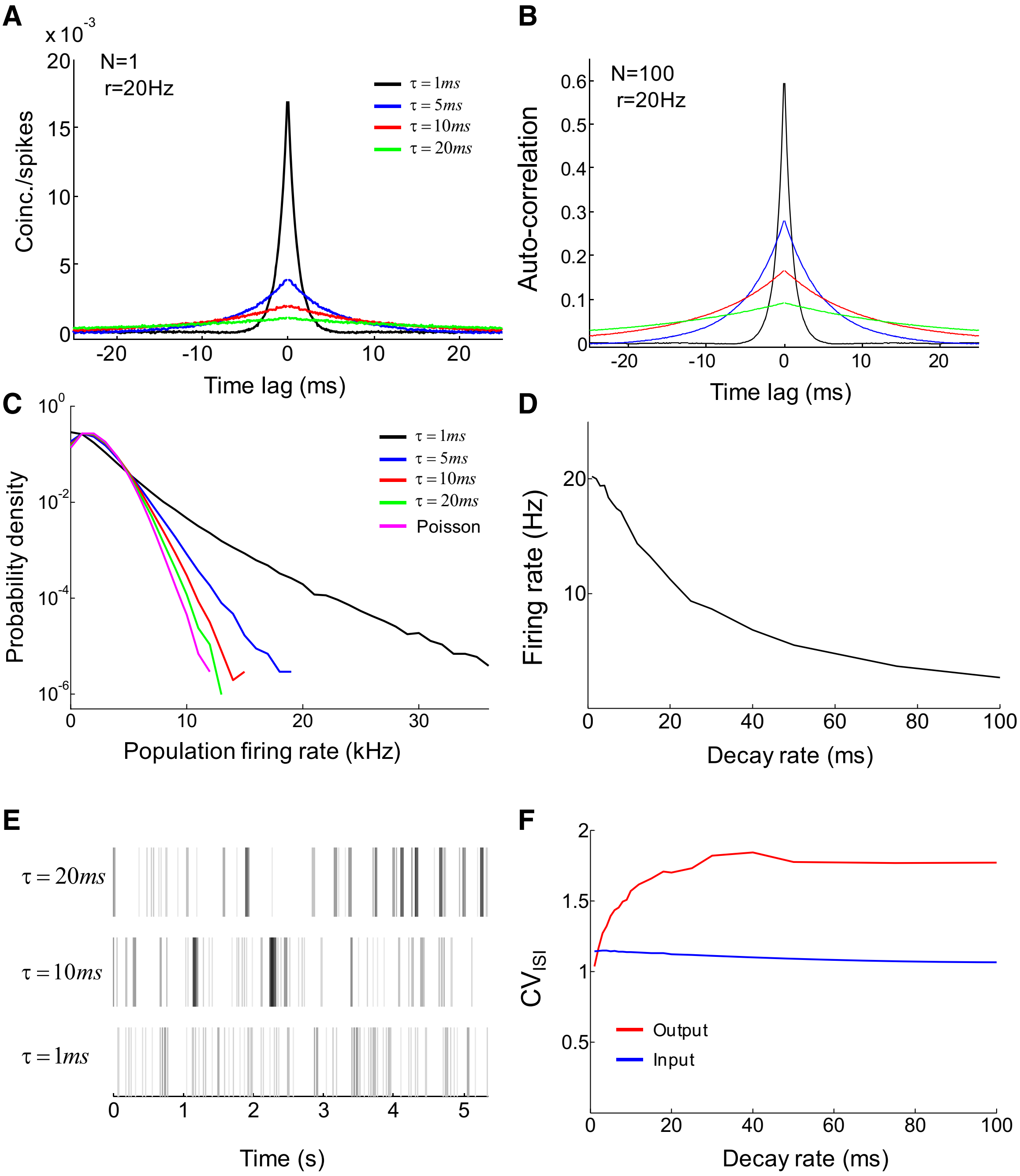

FIG. 10. The impact of temporal correlations on the firing rate of an afferent neuron. We generated a set of homogeneous presynaptic populations consisting of cells firing at 20 Hz with the same auto- and cross-correlations. The different populations differed in the temporal decay rate of the auto- and cross-correlations of the constituent cells. Whereas the temporal decay rate of the auto- and cross-correlations was varied between 1 and 100 ms, the area of the auto- and cross-correlograms was held fixed at 0.2. Therefore the Pearson correlation coefficient between the spike counts of the cells in the presynaptic population was held constant. *A*: auto- and cross-correlations for different temporal decay constants show that the cells in the presynaptic pool become less tightly synchronized with an increase in the temporal decay rate. *B*: the autocorrelation of the pooled activity is amplified with respect to the autocorrelations of the single cells. *C*: the probability density function of the population firing rate is affected by the synchronization level. We computed the probability of the population firing rate for each temporal decay rate. Whereas the mean population firing rate was fixed at 2 kHz, the firing rate variability was strongly modulated by the temporal decay rate. Highly synchronous populations (i.e., shorter temporal decay rates) are linked to a significant increase in the probability of high firing rate events. *D*: the firing rate of the afferent neuron is strongly modulated by the decay rate in the auto- and cross-correlation. As cells in the presynaptic pool become less tightly synchronized, synchronous events that drive the afferent cell to spike become more rare. *E*: the output of the postsynaptic neuron became more bursty for inputs with longer correlation decay rates. The time for each raster plot was rescaled to display the same number of spikes under each condition. *F*: as a consequence, the coefficient of variation of the ISIs of the output spike train is strongly modulated by the tightness of input synchrony.

in synchronous events. In addition, we manipulated the background excitation of the output neuron. We used positive rates to indicate excitatory inputs and negative ones to indicate inhibitory inputs. We found that the optimal number of subpopulations depended on the level of inhibition to the cell (Fig. 11). For a highly inhibited cell, a critical factor to make it fire was

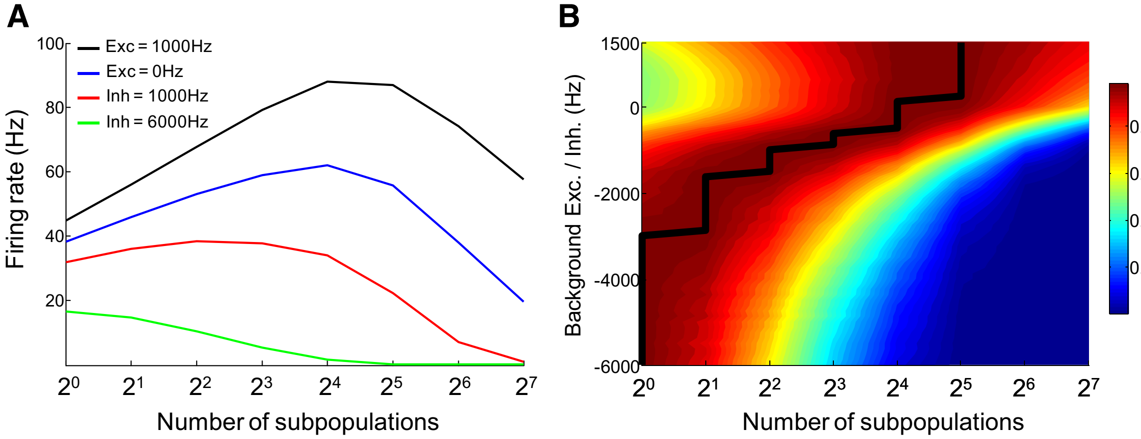

FIG. 11. The impact of the number of synchronized subpopulations on the firing rate of a conductance-based neural model. *A*: average firing rate of the downstream neuron as a function of the number of subpopulations (from 1 to 128 in powers of 2) and the level of background excitation or inhibition. *B*: normalized firing rate of the downstream neuron as a function of the number of subpopulations and the level of background excitation (positive rates) or inhibition (negative rates). The firing rate as a function of the number of subpopulations was normalized to the peak firing rate at each background excitation/inhibition level. The black line indicates the optimal number of subpopulations that maximize the firing rate at a given background activity level.

to have highly synchronized events. This is because to drive the neuron to fire, a large number of cells have to be active at the same time to cross the threshold. For less-inhibited cells, less synchronization was needed (i.e., it is more efficient to have more synchronous events of smaller amplitudes). Populations with fewer subpopulations were less efficient in driving the neuron because the synchronous events involved a larger number of neurons than necessary. Thus the optimal number of subpopulations depended on the background inhibitory input to the output neuron.

Finally, we tested whether our algorithm could generate spatiotemporally correlated spike trains that follow experimentally observed statistics. We used spike trains from a publicly available database (http://crcns.org/data-sets/pvc/pvc-3/). The data were recorded from cat area 17 (primary visual cortex) using multielectrode silicon probes. Anesthetized cats were stimulated with drifting gratings for 722.8 s. We computed the mean firing rate and covariance functions of four selected neurons that had a relatively large number of spikes. In this example we did not normalize the covariance function to display the raw number of spike coincidences. Next, we generated spike trains with the measured first- and second-order statistics obtained from experimental data. Figure 12 shows an excellent agreement between our simulations (red curves) and experimental data (black curves). Our algorithm was able to generate spike trains with irregular peaks in the auto- and cross-covariance functions. Interestingly, the algorithm was able to capture the strongly asymmetric covariance functions of cell 1 and cell 2.

LFPs and STA of the LFP

In Fig. 13 we show how the present algorithm can be used to simulate LFPs and spike trains synchronized to the LFP. We simulated a particular case where the spike train had a constant firing rate with no oscillatory autocorrelation, although it was synchronized to one of the frequency components of the LFP. The spike train had a constant firing rate of 20 Hz and an exponentially decaying autocorrelation (Fig. 13*A*; τ_{decay} = 30 ms). The LFP had a longer time constant (τ_{decay} = 70 ms) and spectral peaks at 30 and 80 Hz (Fig. 13*B*). The nonnormalized spike-field coherence (Fig. 13*C*, *inset*) shows that the spike train was synchronized only to the 80-Hz component of the LFP.

Here we showed an example of a single neuron spike train synchronized to an LFP. In the general case of multiple neurons the LFPs are not required to be associated one to one to each of the spike trains. For instance, we can define fewer LFPs than neurons.

DISCUSSION

We have presented a conceptually simple and computationally efficient method that uses thresholding to translate a multivariate AR process to a multivariate point process with predetermined spatiotemporal statistics. The parameters of the AR process are determined by the desired statistical properties of the spike trains. The algorithm is particularly efficient when simulating homogeneous populations or when the spatial and temporal dependencies of the correlations can be decoupled. Moreover, the Yule–Walker equations, used to determine the parameters of the AR, can be naturally extended to nonstationary first- and second-order statistics and to spike-field coherence simulations.

We next compare our algorithm to previously proposed methods and discuss the limitations, statistical properties of the generated spike trains, and possible applications of the algorithm.

Relation to alternative methods

In most previous efforts to devise algorithms to generate spike trains with specified statistics the proposed methods did not allow a full control of mean firing rates and spatiotemporal correlations (Bohte et al. 2000; Kuhn et al. 2003; Mazurek and Shadlen 2002; Niebur 2007; Oram et al. 1999; Song et al. 2000). However, approaches similar to the present method were very recently developed in parallel (Brette 2009; Krumin and Shoham 2009; Macke et al. 2009). We first describe some generalities of the previously developed algorithms and then relate these new algorithms to our own approach.

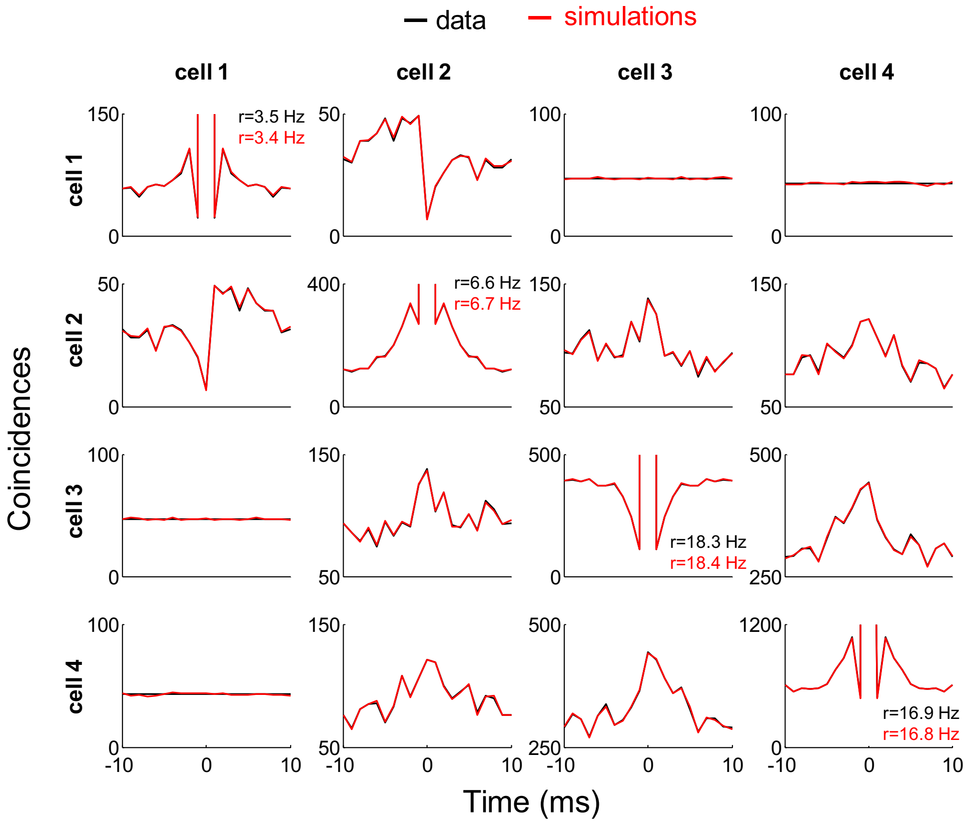

FIG. 12. Simulation of spike trains with first- and second-order statistics corresponding to experimentally measured firing rates and covariance functions in cat area 17 (primary visual cortex). We computed the mean firing rate and covariance of 4 neurons from a publicly available extracellular recording database (http://crcns.org/data-sets/pvc/pvc-3/; drifting bar data set: neurons t00, t02, t23, t27) and generated spike trains with the first- and second-order statistics obtained from the experimental data. The autocovariance functions are plotted in the diagonal along with the experimental (black) and simulated firing rates (red). The *y*-axis indicates the total number of counts during the 722.8 s of recording. The simulation was able to capture the temporally asymmetric irregular and covariance functions. Black curves correspond to simulated data; red curves to experimental data.

Of particular interest is the "thinning and shifting" process described in Bauerle and Grubel (2005). In a strong mathematical sense, this algorithm can be used to generate any stationary multivariate point process in which the constituent point processes are marginally Poisson. Since this algorithm contains a number of others as special examples (method II of Brette 2009; Kuhn et al. 2003; Niebur 2007), we provide a brief description and discussion.

To generate N spike trains start with defining a probability distribution on the $2^N - 1$ nonempty subsets of the set of numbers from 1 through N. Generate a "mother" Poisson spike train and assign to each spike one of the $2^N - 1$ subsets with the given probability. Now generate N daughter processes by creating train i from spikes that have been marked by subsets containing the number i. For instance, when $N = 3$, the set $\{1, 3\}$ could have probability $p_{1,3} = 0.1$, so that spikes in the mother spike train are marked with 1,3 with probability 0.1. A spike at time t marked by $\{1, 3\}$ is copied to spike trains 1 and 3.

The final step of the algorithm is to choose a vector $\vec{v}^i$ from a distribution Q on $\Re^N$ for each spike t_i in the mother spike train. Now shift (or jitter) the spike at time t_i in daughter train j by the amount v_j^i. Note that the spike t_i appears only in some daughter trains and only those will be affected.

Although this procedure results in Poisson spike trains with correlations of all orders easily determined from Q, it also has several shortcomings. First, only Poisson spike trains preserve their characteristics under thinning and jittering. Thus the procedure is difficult to generalize to cases when the individual spike trains are not marginally Poisson. Second, the cross-correlation functions are given in terms of double integrals of the density of Q. Thus to obtain the density Q that produces a desired cross-correlation one would have to solve a set of nontrivial integral equations.

The most recent methods described in Brette (2009), Krumin and Shoham (2009), and Macke et al. (2009) are more closely related to our approach. The method proposed by Macke et al. (2009) also uses the thresholding of Gaussian processes to generate spatial correlations (see also Qaqish 2005). However, in contrast to the present algorithm, this method samples a multivariate Gaussian distribution conditioned on the last K samples to generate temporal correlations. We instead used a multivariate AR that we generalized to the nonstationary case and provided extensive simulations of large neural populations exploiting the structure of the covariance matrix.

Krumin and Shoham (2009) proposed generating a doubly stochastic Poisson process (i.e., a Cox process) with the prescribed mean firing rates and cross-correlations by nonlinearly transforming a Gaussian processes into a nonnegative rate process. They start by choosing an appropriate nonlinear transformation (exponential, square, or absolute value) that is applied to a correlated Gaussian process to yield a continuous nonnegative process that represents the instantaneous firing rate of each neuron. The final step is to use the Gaussian process to determine the intensity of a doubly stochastic Poisson process. The correlated Gaussian process is generated by applying a finite-impulse response (FIR) filter to an uncorrelated Gaussian process (in contrast to the use of AR in our case). The parameters of the FIR are derived from the nonlinear transformation of the desired firing rates and the desired correlation structure. One advantage of the algorithm is that the transformation of the desired firing rates and correlations can be directly calculated, whereas in our case this requires numerical evaluation. Their implementation is slower because, first, a

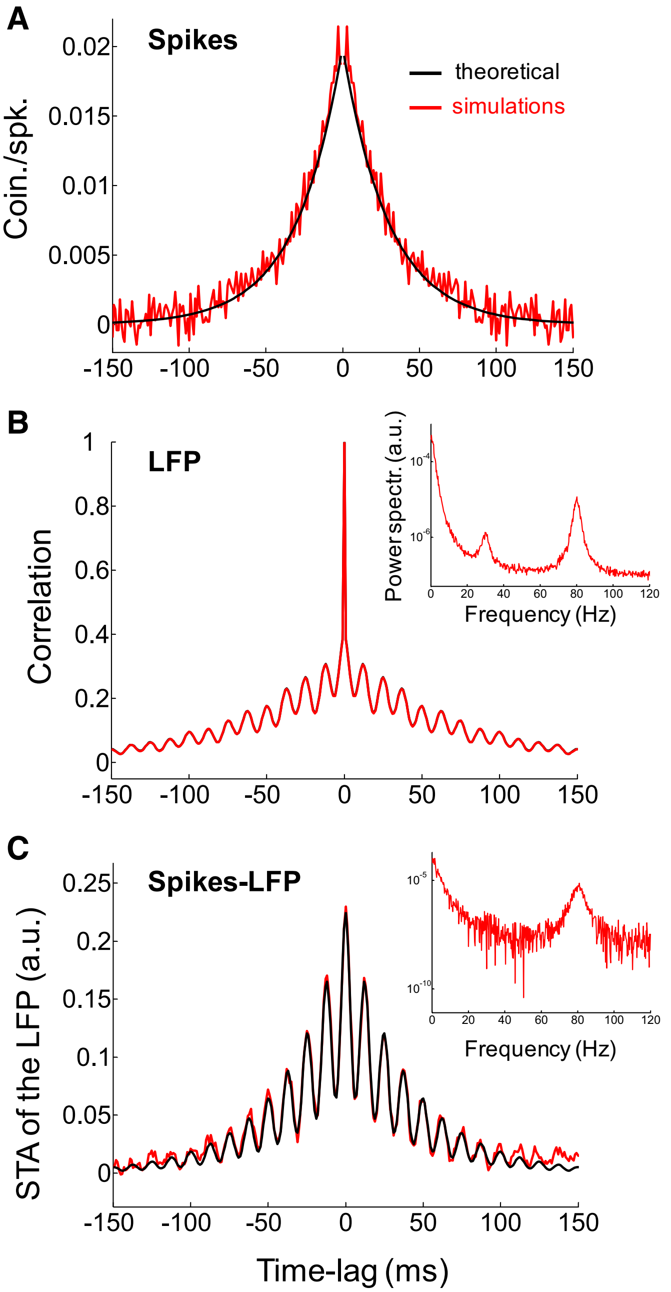

FIG. 13. Example of a spike train synchronized to the local field potential (LFP). *A*: the autocorrelation of the spike train is exponentially decaying with a τ_{decay} = 30 ms and peak correlation of 0.02 coincidence/spike. *B*: the autocorrelation of the LFP is exponentially decaying with τ_{decay} = 70 ms and 2 frequency components at 30 and 80 Hz. The peak correlation was 0.4. *Inset*: power spectrum of the LFP. Two peaks are noticeable at 30 and 80 Hz. *C*: STA of the LFP. The spike train was synchronized only to the 80-Hz component of the LFP. *Inset*: the power spectrum of the STA. Black lines correspond to theoretical curves and red lines represent simulated data.

nonnegative process needs to be generated by filtering Gaussian noise (similarly to our case), but in addition the resulting rates need to be used to generate a point process.

Brette (2009) proposed two algorithms. One was an extension of the method of "thinning and shifting" point processes discussed earlier, whereas another involved doubly stochastic processes, with the analysis restricted to exponential correlations. The stochastic rate is obtained by generating an Ornstein–Ulhenbeck process. The rate process has to be nonnegative, although the Ornstein–Ulhenbeck process is not. To generate a nonnegative rate the realization of the Ornstein–Ulhenbeck process is rectified with a Heaviside step function. Brette (2009) notes that this transformation implies that spike trains with strong correlations will have vanishing rates most of the time. Thus this method is not suitable for the generation of strongly correlated spike trains on fine timescales.

Krumin and Shoham (2009) explained how to generate nonstationary firing rates and cross-correlations explicitly, whereas Brette (2009) described time-varying firing rates. In addition, Brette's method had stronger restrictions on the possible cross-correlation functions. As far as we know, the present approach is the first to exploit the covariance matrix structure for efficient simulations of large neural populations with a natural extension to generate LFPs and STA of the LFP.

Statistical properties of the generated spike trains

The statistical structure of a collection of spike trains is not fully characterized by the first two moments. The structure of higher-order spatiotemporal correlations may have a significant impact on the neural code (Kuhn et al. 2003). It is therefore desirable that the generated spike trains have the least possible amount of structure beyond the first two moments that are specified. In other words, the generated spike trains should have the maximal entropy among all the spike trains with the same first- and second-order statistics (Schneidman et al. 2006). As shown in Macke et al. (2009), point processes generated by thresholding multivariate Gaussian distributions appear to have close to maximal entropy in some parameter regimes (Bethge and Berens 2008).

Limitations

One requirement of our method is that the continuous-process covariance matrix has to be positive definite. In general, a positive-definite point-process covariance matrix *C* does not imply a correspondent positive-definite continuous-process covariance matrix *R*. This implies that point processes with covariance matrices with strong negative correlations will not be able to be generated with our method. This could potentially be circumvented by using an AR process with other nonlinearities or alternative spike-generation mechanisms (Krumin and Shoham 2009). In the APPENDIX, we derive the conditions under which we can obtain a spike train with a prescribed refractory period.

Applications

Multielectrode recording technology, calcium imaging (Greenberg et al. 2008), and voltage-sensitive dyes (Chen et al. 2006) offer unprecedented insights into the structure of correlated activity in neuronal populations. Because the role of correlations in the neural code is difficult to intuit, theoretical tools that aid in the exploration of their impact will be of significant importance. The techniques we have presented will aid this study in at least two ways: *1*) the algorithm offers a way to generate a hypothesis about the role of spatiotemporal structure in spiking activity; and, similarly, *2*) it can be used to test the validity of reduced models that have been proposed to explain this role.

The algorithm also has direct experimental applications. Since there are thousands of synapses per cell, the space of input patterns can be best described using statistical measures. Spatial and temporal correlations between synaptic inputs on the dendritic tree are examples of such measures. New methods developed for random access microscopy (Duemani Reddy et al. 2008) can be used to experimentally simulate almost arbitrary spatiotemporal patterns of synaptic input to a single neuron. The present algorithm provides a natural way of generating such patterns from their statistical description. With the possibility of generating controlled and complex spatiotemporal input patterns we could in principle study more in depth what makes a more realistic, multicompartment neuron model to fire and to assess whether the same conclusions would hold for simpler neuron models.

Simulated spike trains could also be useful in conjunction with more traditional electrophysiological techniques. For instance, the spatiotemporal structure of the inputs plays a significant role in the response of ubiquitous cortical microcircuits (Cruikshank et al. 2001; Pouille and Scanziani 2001). Synthetic spike trains could be used to probe the impact of spatiotemporal structure in such microcircuits and to experimentally study the propagation of coherent behavior through layered neural networks (Reyes 2003).

Local field potential oscillations in particular frequency bands are implicated in a variety of sensory, motor, and cognitive processes (Engel and Singer 2001; Engel et al. 2001; Fries 2005; Fries et al. 1997, 2001; Gray et al. 1989; Rickert et al. 2005; Roelfsema et al. 1997; Schoffelen et al. 2005; Siegel et al. 2008; Womelsdorf and Fries 2007; Womelsdorf et al. 2006, 2007). However, the sources of the LFP activity are usually unknown. In this way, it is usually difficult to understand the role of spike–LFP synchronization in information processing. We believe that progress can be made by simulating the experimentally observed statistical relationship between spikes and LFP. For instance, we could test specific conditions for efficient information transmission by parametrically manipulating the synchronization of spikes and LFPs and studying the impact in the firing patterns of neural networks. Specific hypotheses about the role of spike–LFP synchronization in different frequency bands could be obtained by comparing the simulations results with experimental data.

APPENDIX

Our method of generating spike trains requires that the continuous process covariance matrix be positive definite. Spike trains with absolute refractory periods cannot be generated with our algorithm because the associated continuous-process covariance matrix is non-positive definite. However, the algorithm can generate relative refractory periods, close to absolute ones, for short refractory periods (see Fig. 6). Here, we derive the maximum possible refractoriness (i.e., how much the probability of firing a spike is reduced after a spike was elicited in the past) as a function of the length of the refractory period. The maximum possible refractoriness for a fixed refractory period length is obtained when the continuous-process covariance matrix is positive definite but it is as close as possible to be singular. This means that we have to minimize the correlation of the continuous variable $y(t)$ and $y(t + \tau)$, with τ shorter than or equal to the refractory period length.

Let m be the length of the refractory period and a the minimum correlation value. The Toeplitz continuous-process covariance matrix R of size $N \times N$ is a banded matrix, where

$$R_{ij} = \begin{cases} 0 & \text{if } j < i - m \text{ or } j > i + m \\ 1 & \text{if } j = i \\ a & \text{otherwise} \end{cases}$$

The definition of a positive definite matrix requires that $\vec{x}R\vec{x}^t > 0$ for every

$\vec{x} \in \Re^{1 \times N}, \vec{x} \neq 0$

$$\vec{x}R\vec{x}^t = \sum_{i=1}^{N} x_i^2 + a \sum_{i=1}^{N} x_i \left(\sum_{j=i-m}^{i-1} x_j + \sum_{j=i+1}^{i+m} x_j \right) > 0$$

We want to find a condition that does not depend on x_i. If we take $x_i = x$ for $\forall i$

$$\vec{x}R\vec{x}^t = Nx^2 + ax^2[m(m-1) + 2m(N-m)] > 0$$

$$a > -\frac{N}{m(m-1) + 2m(N-m)}$$

This last equation represents a necessary condition. When $N \rightarrow \infty$, the condition for the matrix to be positive definite is $a > -1/(2m)$. This indicates that for longer refractory periods, the restriction of possible covariance matrices becomes tighter. Next, we demonstrate that $a > -1/(2m)$ is also a sufficient condition when $N \rightarrow \infty$. We can rewrite $\vec{x}R\vec{x}^t > 0$ as

$$\vec{x}R\vec{x}^t = k + \sum_{i=m+1}^{N-m} \sum_{\substack{j=i-m \\ j \neq 1}}^{i+m} ax_ix_j + \frac{x_i^2}{4m} + \frac{x_j^2}{4m} > 0$$

The factor k incorporates the terms corresponding to the boundaries of the Toeplitz matrix, becoming negligible when $N \rightarrow \infty$. If every term in the sum is >0, then $\vec{x}R\vec{x}^t > 0$

$$ax_ix_j + \frac{x_i^2}{4m} + \frac{x_j^2}{4m} > 0$$

$$x_i^2 + x_j^2 + 4\,ma\,\mathrm{x_i}\,\mathrm{x_j} > 0$$

and the condition $a > -1/(2m)$ ensures that this inequality holds for every x_i, x_j.

ACKNOWLEDGMENTS

We thank R. Rosenbaum, H. Rey, and P. Berens for comments on the manuscript.

GRANTS

This work was supported by a Texas Advanced Research Program/Advanced Technology Program grant to K. Josić and National Science Foundation Grants DMS-0604429 and DMS-0817649.

REFERENCES

Akaike H. Block Toeplitz matrix inversion. *SIAM J Appl Math* 24: 234–241, 1973.

Alonso JM, Martinez LM. Functional connectivity between simple cells and complex cells in cat striate cortex. *Nat Neurosci* 1: 395–403, 1998.

Alonso JM, Usrey WM, Reid RC. Rules of connectivity between geniculate cells and simple cells in cat primary visual cortex. *J Neurosci* 21: 4002–4015, 2001.

Ammar GS. Classical foundations of algorithms for solving positive definite Toeplitz equations. *Calcolo* 33: 99–113, 1996.

Ammar GS, Gragg WB. Superfast solution of real positive definite Toeplitz-systems. *SIAM J Matrix Anal Appl* 9: 61–76, 1988.

Averbeck BB, Lee D. Effects of noise correlations on information encoding and decoding. *J Neurophysiol* 95: 3633–3644, 2006.

Bair W, Zohary E, Newsome WT. Correlated firing in macaque visual area MT: time scales and relationship to behavior. *J Neurosci* 21: 1676–1697, 2001.

Bauerle N, Grubel R. Multivariate counting processes: copulas and beyond. *Astin Bull* 35: 379–408, 2005.

Bethge M, Berens P. Near-maximum entropy models for binary neural representations of natural images. In: *Advances in Neural Information Processing Systems 20: Proceedings of the 2007 Conference*, edited by Platt JC, Koller D, Singer Y, Roweis S. Cambridge, MA: MIT Press, 2008, p. 97–104.

Bohte SM, Spekreijse H, Roelfsema PR. The effects of pair-wise and higher-order correlations on the firing rate of a postsynaptic neuron. *Neural Comput* 12: 153–179, 2000.

Boyden ES, Zhang F, Bamberg E, Nagel G, Deisseroth K. Millisecond-timescale, genetically targeted optical control of neural activity. *Nat Neurosci* 8: 1263–1268, 2005.

Brette R. Generation of correlated spike trains. *Neural Comput* 21: 188–215, 2009.

Brody CD. Correlations without synchrony. *Neural Comput* 11: 1537–1551, 1999a.

Brody CD. Disambiguating different covariation types. *Neural Comput* 11: 1527–1535, 1999b.

Callaway EM, Katz LC. Photostimulation using caged glutamate reveals functional circuitry in living brain slices. *Proc Natl Acad Sci USA* 90: 7661–7665, 1993.

Chen Y, Geisler WS, Seidemann E. Optimal decoding of correlated neural population responses in the primate visual cortex. *Nat Neurosci* 9: 1412–1420, 2006.

Cohen MR, Newsome WT. Context-dependent changes in functional circuitry in visual area MT. *Neuron* 60: 162–173, 2008.

Cox DR, Isham V. *Point Processes.* New York: Chapman & Hall, 1980.

Cox DR, Wermuth N. Likelihood factorizations for mixed discrete and continuous variables. *Scand J Stat* 26: 209–220, 1999.

Cruikshank SJ, Killackey HP, Metherate R. Parvalbumin and calbindin are differentially distributed within primary and secondary subregions of the mouse auditory forebrain. *Neuroscience* 105: 553–569, 2001.

de la Rocha J, Doiron B, Shea-Brown E, Josic K, Reyes A. Correlation between neural spike trains increases with firing rate. *Nature* 448: 802–806, 2007.

Duemani Reddy G, Kelleher K, Fink R, Saggau P. Three-dimensional random access multiphoton microscopy for functional imaging of neuronal activity. *Nat Neurosci* 11: 713–720, 2008.

Engel AK, Fries P, Singer W. Dynamic predictions: oscillations and synchrony in top-down processing. *Nat Rev Neurosci* 2: 704–716, 2001.

Engel AK, Singer W. Temporal binding and the neural correlates of sensory awareness. *Trends Cogn Sci* 5: 16–25, 2001.

Fetz EE, Gustafsson B. Relation between shapes of post-synaptic potentials and changes in firing probability of cat motoneurones. *J Physiol* 341: 387–410, 1983.

Fries P. A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. *Trends Cogn Sci* 9: 474–480, 2005.

Fries P, Nikolic D, Singer W. The gamma cycle. *Trends Neurosci* 30: 309–316, 2007.

Fries P, Reynolds JH, Rorie AE, Desimone R. Modulation of oscillatory neuronal synchronization by selective visual attention. *Science* 291: 1560–1563, 2001.

Fries P, Roelfsema PR, Engel AK, Konig P, Singer W. Synchronization of oscillatory responses in visual cortex correlates with perception in interocular rivalry. *Proc Natl Acad Sci USA* 94: 12699–12704, 1997.

Gasparini S, Magee JC. State-dependent dendritic computation in hippocampal CA1 pyramidal neurons. *J Neurosci* 26: 2088–2100, 2006.

Gray CM, Konig P, Engel AK, Singer W. Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. *Nature* 338: 334–337, 1989.

Greenberg DS, Houweling AR, Kerr JN. Population imaging of ongoing neuronal activity in the visual cortex of awake rats. *Nat Neurosci* 11: 749–751, 2008.

Gutnisky DA, Dragoi V. Adaptive coding of visual information in neural populations. *Nature* 452: 220–224, 2008.

Hallin M, Ingenbleek JF. Nonstationary Yule–Walker equations. *Stat Probabil Lett* 1: 189–195, 1983.

Han X, Boyden ES. Multiple-color optical activation, silencing, and desynchronization of neural activity, with single-spike temporal resolution. *PLoS ONE* 2: e299, 2007.

Heck D, Kummell F, Thach WT, Aertsena A. Dynamic correlation of neuronal activity in rat cerebellar cortex modulated by behavior. *Ann NY Acad Sci* 978: 156–163, 2002.

Huber D, Petreanu L, Ghitani N, Ranade S, Hromadka T, Mainen Z, Svoboda K. Sparse optical microstimulation in barrel cortex drives learned behaviour in freely moving mice. *Nature* 451: 61–64, 2008.

Kamm J, Nagy JG. Optimal Kronecker product approximation of block Toeplitz matrices. *SIAM J Matrix Anal Appl* 22: 155–172, 2000.

Kohn A, Smith MA. Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *J Neurosci* 25: 3661–3673, 2005.

Krumin M, Shoham S. Generation of spike trains with controlled auto- and cross-correlation functions. *Neural Comput* 21: 1642–1664, 2009.

Kuhn A, Aertsen A, Rotter S. Higher-order statistics of input ensembles and the response of simple model neurons. *Neural Comput* 15: 67–101, 2003.

Kuhn A, Aertsen A, Rotter S. Neuronal integration of synaptic input in the fluctuation-driven regime. *J Neurosci* 24: 2345–2356, 2004.

Macke JH, Berens P, Ecker AS, Tolias AS, Bethge M. Generating spike trains with specified correlation coefficients. *Neural Comput* 21: 397–423, 2009.

MacLeod K, Laurent G. Distinct mechanisms for synchronization and temporal patterning of odor-encoding neural assemblies. *Science* 274: 976–979, 1996.

Markowitz DA, Collman F, Brody CD, Hopfield JJ, Tank DW. Rate-specific synchrony: using noisy oscillations to detect equally active neurons. *Proc Natl Acad Sci USA* 105: 8422–8427, 2008.

Mazurek ME, Shadlen MN. Limits to the temporal fidelity of cortical spike rate signals. *Nat Neurosci* 5: 463–471, 2002.

Moreno R, de la Rocha J, Renart A, Parga N. Response of spiking neurons to correlated inputs. *Phys Rev Lett* 89: 288101.1–288101.4, 2002.

Muller JR, Metha AB, Krauskopf J, Lennie P. Rapid adaptation in visual cortex to the structure of images. *Science* 285: 1405–1408, 1999.

Muresan RC, Jurjut OF, Moca VV, Singer W, Nikolic D. The oscillation score: an efficient method for estimating oscillation strength in neuronal activity. *J Neurophysiol* 99: 1333–1353, 2008.

Netoff TI, Banks MI, Dorval AD, Acker CD, Haas JS, Kopell N, White JA. Synchronization in hybrid neuronal networks of the hippocampal formation. *J Neurophysiol* 93: 1197–1208, 2005.

Niebur E. Generation of synthetic spike trains with defined pairwise correlations. *Neural Comput* 19: 1720–1738, 2007.

Nikolenko V, Poskanzer KE, Yuste R. Two-photon photostimulation and imaging of neural circuits. *Nat Methods* 4: 943–950, 2007.

Nikolenko V, Watson BO, Araya R, Woodruff A, Peterka DS, Yuste R. SLM microscopy: scanless two-photon imaging and photostimulation with spatial light modulators (Abstract). *Front Neural Circuits* 2: 5, 2008.

Oram MW, Wiener MC, Lestienne R, Richmond BJ. Stochastic nature of precisely timed spike patterns in visual system neuronal responses. *J Neurophysiol* 81: 3021–3033, 1999.

Perkel DH, Gerstein GL, Moore GP. Neuronal spike trains and stochastic point processes. II. Simultaneous spike trains. *Biophys J* 7: 419–440, 1967.

Poort J, Roelfsema PR. Noise correlations have little influence on the coding of selective attention in area V1. *Cereb Cortex* 19: 543–553, 2009.

Pouille F, Scanziani M. Enforcement of temporal fidelity in pyramidal cells by somatic feed-forward inhibition. *Science* 293: 1159–1163, 2001.

Qaqish BF. A family of multivariate binary distributions for simulating correlated binary variables with specified marginal means and correlations *Biometrika* 90: 455–463, 2005.

Reyes AD. Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro. *Nat Neurosci* 6: 593–599, 2003.

Rickert J, Oliveira SC, Vaadia E, Aertsen A, Rotter S, Mehring C. Encoding of movement direction in different frequency ranges of motor cortical local field potentials. *J Neurosci* 25: 8815–8824, 2005.

Riehle A, Grun S, Diesmann M, Aertsen A. Spike synchronization and rate modulation differentially involved in motor cortical function. *Science* 278: 1950–1953, 1997.

Roelfsema PR, Engel AK, Konig P, Singer W. Visuomotor integration is associated with zero time-lag synchronization among cortical areas. *Nature* 385: 157–161, 1997.

Romo R, Hernandez A, Zainos A, Salinas E. Correlated neuronal discharges that increase coding efficiency during perceptual discrimination. *Neuron* 38: 649–657, 2003.

Salinas E, Sejnowski TJ. Impact of correlated synaptic input on output firing rate and variability in simple neuronal models. *J Neurosci* 20: 6193–6209, 2000.

Salinas E, Sejnowski TJ. Correlated neuronal activity and the flow of neural information. *Nat Rev Neurosci* 2: 539–550, 2001.

Schneidman E, Berry MJ 2nd, Segev R, Bialek W. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature* 440: 1007–1012, 2006.

Schoffelen JM, Oostenveld R, Fries P. Neuronal coherence as a mechanism of effective corticospinal interaction. *Science* 308: 111–113, 2005.

Schroeder CE, Lakatos P. Low-frequency neuronal oscillations as instruments of sensory selection. *Trends Neurosci* 32: 9–18, 2009.

Shanmugan KS, Breipohl AM. *Random Signals: Detection, Estimation, and Data Analysis.* New York: Wiley, 1988.

Shea-Brown E, Josić K, de la Rocha J, Doiron B. Correlation and synchrony transfer in integrate-and-fire neurons: basic properties and consequences for coding. *Phys Rev Lett* 100: 108102, 2008.

Siegel M, Donner TH, Oostenveld R, Fries P, Engel AK. Neuronal synchronization along the dorsal visual pathway reflects the focus of spatial attention. *Neuron* 60: 709–719, 2008.

Smith MA, Kohn A. Spatial and temporal scales of neuronal correlation in primary visual cortex. *J Neurosci* 28: 12591–12603, 2008.

Song S, Miller KD, Abbott LF. Competitive Hebbian learning through spike-timing-dependent synaptic plasticity. *Nat Neurosci* 3: 919–926, 2000.

Steinmetz PN, Roy A, Fitzgerald PJ, Hsiao SS, Johnson KO, Niebur E. Attention modulates synchronized neuronal firing in primate somatosensory cortex. *Nature* 404: 187–190, 2000.

Usrey WM, Reppas JB, Reid RC. Specificity and strength of retinogeniculate connections. *J Neurophysiol* 82: 3527–3540, 1999.

Vaadia E, Haalman I, Abeles M, Bergman H, Prut Y, Slovin H, Aertsen A. Dynamics of neuronal interactions in monkey cortex in relation to behavioural events. *Nature* 373: 515–518, 1995.

Veredas FJ, Vico FJ, Alonso JM. Factors determining the precision of the correlated firing generated by a monosynaptic connection in the cat visual pathway. *J Physiol* 567: 1057–1078, 2005.

Wax M, Kailath T. Efficient inversion of Toeplitz–Block Toeplitz matrix. *IEEE Trans Acoust Speech Signal Process* 31: 1218–1221, 1983.

Wehr M, Laurent G. Odour encoding by temporal sequences of firing in oscillating neural assemblies. *Nature* 384: 162–166, 1996.

Womelsdorf T, Fries P. The role of neuronal synchronization in selective attention. *Curr Opin Neurobiol* 17: 154–160, 2007.

Womelsdorf T, Fries P, Mitra PP, Desimone R. Gamma-band synchronization in visual cortex predicts speed of change detection. *Nature* 439: 733–736, 2006.

Womelsdorf T, Schoffelen JM, Oostenveld R, Singer W, Desimone R, Engel AK, Fries P. Modulation of neuronal interactions through neuronal synchronization. *Science* 316: 1609–1612, 2007.

Zhang F, Aravanis AM, Adamantidis A, de Lecea L, Deisseroth K. Circuit-breakers: optical technologies for probing neural signals and systems. *Nat Rev Neurosci* 8: 577–581, 2007a.

Zhang F, Wang LP, Boyden ES, Deisseroth K. Channelrhodopsin-2 and optical control of excitable cells. *Nat Methods* 3: 785–792, 2006.

Zhang F, Wang LP, Brauner M, Liewald JF, Kay K, Watzke N, Wood PG, Bamberg E, Nagel G, Gottschalk A, Deisseroth K. Multimodal fast optical interrogation of neural circuitry. *Nature* 446: 633–639, 2007b.

Zohary E, Shadlen MN, Newsome WT. Correlated neuronal discharge rate and its implications for psychophysical performance. *Nature* 370: 140–143, 1994.