Input nonlinearities can shape beyond-pairwise correlations and improve information transmission by neural populations

Joel Zylberberg

Department of Applied Mathematics, University of Washington, Seattle, Washington 98195, USA

Eric Shea-Brown[*]

Department of Applied Mathematics, Program in Neuroscience, Department of Physiology and Biophysics,
University of Washington, Seattle, Washington 98195, USA

While recent recordings from neural populations show beyond-pairwise, or *higher-order*, correlations (HOC), we have little understanding of how HOC arise from network interactions and of how they impact encoded information. Here, we show that input nonlinearities imply HOC in spin-glass-type statistical models. We then discuss one such model with parametrized pairwise- and higher-order interactions, revealing conditions under which beyond-pairwise interactions increase the mutual information between a given stimulus type and the population responses. For jointly Gaussian stimuli, coding performance is improved by shaping output HOC only when neural firing rates are constrained to be low. For stimuli with skewed probability distributions (like natural image luminances), performance improves for all firing rates. Our work suggests surprising connections between nonlinear integration of neural inputs, stimulus statistics, and normative theories of population coding. Moreover, it suggests that the inclusion of beyond-pairwise interactions could improve the performance of Boltzmann machines for machine learning and signal processing applications.

I. INTRODUCTION

The number of neurons for which activities can be simultaneously recorded is rapidly increasing [1]. We thus have an advancing understanding of the statistics of population activities, like the relative frequencies of coactive neural pairs, triplets, etc. In particular, much work has investigated the distributions of simultaneously recorded retinal ganglion cell "words" (patterns of binary neural activities). For some population sizes and stimuli, these distributions are well fit by pairwise maximum entropy (ME) models [2–4], while in other cases beyond-pairwise interactions are evident in the data and models incorporating higher-order correlations (HOC) are needed [5,6]. Cortical studies yield similar observations [7–9].

How do these correlations affect *population coding*? Much work has investigated how pairwise correlations impact a population's ability to transmit information [10–19]. In particular, Ref. [14] identifies optimal pairwise interactions that maximize encoded information in a setting very similar to that we will study below. Coding studies of higher-order correlations (HOC) are limited, but empirical work shows that, in some cases, including HOC allows a decoder to recover the stimulus presented to a neural population three times faster than a decoder with access only to pairwise statistics [5]. Intriguingly, other work [7] shows that HOC reduce the mutual information (MI) between the stimuli and resultant population responses. A recent theoretical investigation [20] outlines principles by which higher-order correlations can impact the discriminability of pairs of nearby stimuli. There, second-order response statistics (pairwise correlations) were held fixed at identical values for each stimulus, and the authors showed how triplet correlations could make population

responses more discriminable by skewing the distributions of population responses to the two stimuli away from each other.

This prior work is intriguing, but begs several questions. How do HOC impact population coding of multiple stimuli, or continuous families of stimuli—and not just the stimulus pairs of [20]? What network mechanisms can generate the underlying HOC? Here, we address both of these questions, identifying when HOC may improve coding performance and how those performance gains come about. Aside from its relevance for neurobiology, our study suggests that incorporating higher-order interaction terms may improve the performance of Boltzmann machines (including deep belief networks), which are promising algorithms for machine learning applications [21,22].

II. RESULTS

A. Encoding model

We generalize the approach of Tkačik *et al.* [14], and model the activity of a population of neurons by a triplet-wise ME distribution. Within this model, the stimuli affect neural responses via *bias terms* $h_i = h_i^s + h_i^0$ [Fig. 1(a)]. Here, h_i is the bias to neuron i, h_i^0 is the stimulus-independent bias, and h_i^s is the stimulus-dependent bias. As discussed below, the stimuli can be interpreted as additive inputs in a linear-nonlinear neural model, and we define the stimulus distribution by the joint distribution over h_i^s [14].

Once the biases are specified, the neural activities $\{\vec{\sigma}\}$ ($\sigma_i \in \{0,1\}$ is the silent versus spiking state of neuron i) are distributed as

$$p(\vec{\sigma}|\vec{h})$$

$$= \frac{1}{Z} \exp\left[\beta\left(\vec{h} \cdot \vec{\sigma} + \sum_{i<j} J_{ij}\sigma_i\sigma_j + \sum_{i<j<k} \gamma_{ijk}\sigma_i\sigma_j\sigma_k\right)\right].$$
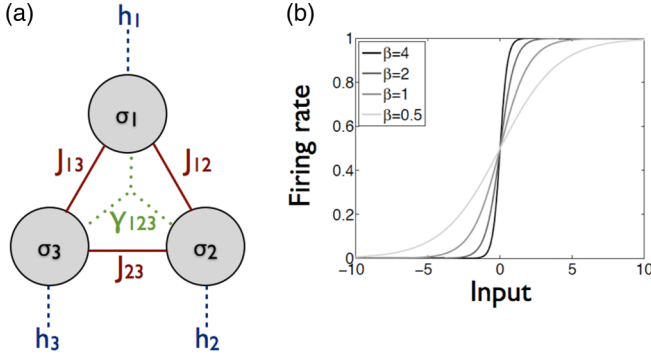
(1)

*etsb@washington.edu

FIG. 1. (Color online) Encoding model. (a) Each model neuron has a bias h_i (blue) determined by external stimuli. Recurrent pairwise (J_{ij}, red) and triplet (γ_{ijk}, green) interactions further modify the output statistics. Illustrative schematic shown for $N = 3$ neurons; models can have arbitrary N. (b) The firing rate (spiking probability) of each neuron varies sigmoidally with the strength of its input ($x_i = h_i + \sum_{j\neq i} J_{ij}\sigma_j + \sum_{j<k;j,k\neq i} \gamma_{ijk}\sigma_j\sigma_k$); see text. The steepness of the sigmoid depends on β.

Here, β specifies the distribution's width, defining neural reliability [14,23] analogous to the inverse temperature of an Ising spin-glass model. The parameters J_{ij} and γ_{ijk} describe the pairwise interactions and triplet interactions, respectively. The *partition function $Z = \sum_{\{\vec{\sigma}\}} \exp[\beta(\vec{h}\cdot\vec{\sigma} + \sum_{i<j} J_{ij}\sigma_i\sigma_j + \sum_{i<j<k} \gamma_{ijk}\sigma_i\sigma_j\sigma_k)]$* is the normalizing constant that ensures that all probabilities sum to 1. This distribution is the one that specifies the means, covariances, and 3-pt correlations of the activity distribution, while making the fewest possible assumptions about the distribution overall [2,3,24–26]. Later, we will optimize over the interaction parameters (and thus the moments of the response distribution), for different stimulus distributions. In so doing, we will identify conditions under which triplet interactions improve stimulus encoding.

As emphasized by [14], this parametrization of $p(\vec{\sigma}|\vec{h})$ can be interpreted as a static nonlinear input-output neural model, within a network with symmetrical connections between units. Consider the probability of one neuron firing (having $\sigma_i = 1$), conditioned on the states of the other neurons and the biases:

$$p(\sigma_i = 1|\{\sigma_{j\neq i}\},\vec{h})$$

$$= g\left[\beta\left(h_i + \sum_{j\neq i} J_{ij}\sigma_j + \sum_{j<k;j,k\neq i} \gamma_{ijk}\sigma_j\sigma_k\right)\right], \quad (2)$$

where $g(\beta x_i) = (1 + e^{-\beta x_i})^{-1}$ is a sigmoidal function [Fig. 1(b)]. To obtain Eq. (2) from Eq. (1), consider the conditional probability distribution $p(\sigma_i|\{\sigma_{j\neq i}\},\vec{h})$, where σ_i can take on one of two values: either 0 or 1. The exponential "Boltzmann factors" for each of the two states, obtained by using $\sigma_i = 0$ or 1 in Eq. (1), while keeping everything else fixed, are $\exp[\beta(\sum_{j\neq i}\sigma_j h_j + \sum_{j<k;j,k\neq i} J_{jk}\sigma_j\sigma_k + \sum_{j<k<l;j,k,l\neq i} \gamma_{jkl}\sigma_j\sigma_k\sigma_l)]$ and $\exp[\beta(h_i + \sum_{j\neq i} J_{ij}\sigma_j + \sum_{j<kj;j,k\neq i} \gamma_{ijk}\sigma_j\sigma_k) + \beta(\sum_{j\neq i}\sigma_j h_j + \sum_{j<k;j,k\neq i} J_{jk}\sigma_j\sigma_k + \sum_{j<k<l;j,k,l\neq i} \gamma_{jkl}\sigma_j\sigma_k\sigma_l)]$ respectively. Summing these two Boltzmann factors, we get the conditional partition

function, and dividing the appropriate Boltzmann factor by this conditional partition function, we obtain Eq. (2).

The firing probability in one discrete time bin is akin to the mean firing rate. Since firing rates that vary sigmoidally with synaptic input are commonly encountered [27–29], we interpret the argument (x_i) of the sigmoid as the input to a linear-nonlinear model neuron. With no beyond-pairwise interactions ($\gamma_{ijk} = 0$), the bias h_i and recurrent inputs to the neuron $\{J_{ij}\sigma_j\}$ add; the sigmoidal function of that sum determines the firing rate. If $\gamma_{ijk} > 0$, then when neurons j and k are coactive, the recurrent input to neuron i is $J_{ij} + J_{ik} + \gamma_{ijk}$, which is larger than the sum of the contributions observed when only one recurrent input is active at a time ($J_{ij} + J_{ik}$); these inputs combine superlinearly. Conversely, for $\gamma_{ijk} < 0$, they combine sublinearly. Thus the way that synaptic inputs combine maps onto triplet interactions in statistical models of population activity, shaping beyond-pairwise correlations. If the recurrent input to neuron i is an arbitrary function of the activities of the other neurons, $x_i = h_i + f(\{\sigma_{j\neq i}\})$, triplet interactions come from the first nonlinear terms in the series expansion of $f(\cdot)$ (see Appendix A).

We note that, in this mechanistic interpretation of our probability model, both the recurrent connections J_{ij}, and the super- or sublinear integration (given by γ_{ijk}) are symmetrical in their indices. While this symmetry is physically realizable in neuronal networks, it is not the most general possible configuration. We will later return to possible biophysical mechanisms behind such nonlinearities. We further note that, in our model, these "recurrent" interactions are instantaneous, which is not true for physical neurons. Studies of the present mechanism for HOC in recurrent dynamical models, as in Ref. [30], are an intriguing area for future work. Finally, we note that, in addition to recurrent coupling, interaction terms can also come from common noise inputs to multiple neurons [30–32], and that, in this case, the interactions are likely to be symmetrical, as in Eq. (1).

B. When do HOC improve coding? Analytical results

Having motivated our probability model, we ask when triplet interactions improve coding. To do this, we use the framework introduced by Tkačik *et al.* [14] to study population coding with pairwise interactions. For a given stimulus distribution and parameter set $\{\beta, \{h_i^0\}, \{J_{ij}\}, \{\gamma_{ijk}\}\}$, we compute the mutual information (MI) between stimuli $\vec{h^s}$ and the responses $\vec{\sigma}$:

$$I(\vec{\sigma};\vec{h^s}) = -\sum_{\{\vec{\sigma}\}} p(\vec{\sigma})\log[p(\vec{\sigma})] + \int d\vec{h^s} p(\vec{h^s})$$

$$\times \sum_{\{\vec{\sigma}\}} p(\vec{\sigma}|\vec{h^s})\log[p(\vec{\sigma}|\vec{h^s})]. \quad (3)$$

The first term is the *response entropy*, and the second term is (minus) the mean entropy of the response conditioned on the stimulus (*noise entropy*).

We will first discuss analytical results, obtained in the limit of weak stimulus dependence of the neural responses—corresponding to small β—for arbitrary $\{\{h_i^0\}, \{J_{ij}\}, \{\gamma_{ijk}\}\}$. We will then show numerical results indicating that our qualitative findings persist over a range of stimulus-coupling

strengths. The analytical calculations are quite tedious, and so we describe them briefly here, and show all the details in Appendix B. For the analytical investigation, we rewrite our probability distribution as

$$p(\vec{\sigma}|\vec{h}) = \frac{1}{Z} \exp\left[\epsilon \vec{h}^s \cdot \vec{\sigma} + \beta \left(\vec{h}^0 \cdot \vec{\sigma} + \sum_{i<j} J_{ij}\sigma_i\sigma_j + \sum_{i<j<k} \gamma_{ijk}\sigma_i\sigma_j\sigma_k \right) \right], \qquad (4)$$

where ϵ parametrizes the strength of the stimulus dependence of the neuronal responses [33]. We then expand the MI [Eq. (3)] in powers of ϵ. This expansion yields

$$I(\vec{\sigma};\vec{h}^s) = \frac{\epsilon^2}{2}\left[\sum_i \langle (h_i^s)^2 \rangle \text{var}(\sigma_i) + \sum_{i\neq j} \langle h_i^s h_j^s \rangle \text{cov}(\sigma_i,\sigma_i) \right] + \epsilon^3 \left[\sum_i \frac{\mu_i}{3}\left(1 + 2\mu_i^2 - 3\mu_i\right)\langle h_i^3 \rangle + \sum_{i\neq j} \text{cov}(\sigma_i,\sigma_j)(1-2\mu_i)\langle h_i^2 h_j \rangle \right]$$

$$+ \epsilon^3 \left[\sum_{i\neq j\neq k} \frac{1}{3}(\tau_{ijk} + 2\mu_i\mu_j\mu_k - 3\mu_k\pi_{ij})\langle h_i h_j h_k \rangle \right] + O(\epsilon^4), \qquad (5)$$

where $\mu_i = \mathbb{E}[\sigma_i|\vec{h}^s = \vec{0}]$, $\pi_{ij} = \mathbb{E}[\sigma_i\sigma_j|\vec{h}^s = \vec{0}]$, $\tau_{ijk} = \mathbb{E}[\sigma_i\sigma_j\sigma_k|\vec{h}^s = \vec{0}]$, angled brackets denote expectations over the stimulus distribution, and $\text{cov}(\sigma_i,\sigma_i) = \pi_{ij} - \mu_i\mu_j$ and $\text{var}(\sigma_i) = \pi_{ii} - \mu_i^2$ are moments of the *spontaneous activity distribution* (obtained when no stimulus is present, and thus $h_i^s = 0 \,\forall i$).

The form of Eq. (5) emphasizes that the MI depends on the relationships between the moments of the stimulus distribution, and the moments of the response distribution in a very specific way. In particular, it is the moments of the *spontaneous activity distribution* (obtained when no stimulus is present, and thus $h_i^s = 0 \,\forall i$) that determine the MI. This effect arises because the Boltzmann factors (the exponentials in the numerator of) Eq. (4) factorize into a stimulus-dependent part $\exp(\epsilon \vec{h}^s \cdot \vec{\sigma})$, and a stimulus-independent part. In the case where the stimulus is set to zero, the stimulus-dependent part is unity, and one obtains the Boltzmann factors for the spontaneous activity distribution. Thus the Boltzmann factors consist of a stimulus-dependent term multiplied by the Boltzmann factors for the spontaneous activity distribution. Consequently, the moments of the spontaneous activity distribution determine the statistical interactions of the encoding model, and thus the MI. Our analytical expansion exploits this multiplicative structure to obtain Eq. (5).

Because Eq. (5) yields the MI as a function of moments of the spontaneous activity distribution—and the interaction terms in Eqs. (1) and (4) determine those moments—we can use Eq. (5) to understand the conditions under which different interactions will improve MI.

Our first observation is that, in the case of stimulus distributions that are symmetric about their mean (like, for example, Gaussian distributions), the odd moments vanish, and thus so do the $O(\epsilon^3)$ terms in Eq. (5). In this case, at least up to $O(\epsilon^4)$, the MI depends only on the first and second moments of the response distribution. Consequently, in this case, adjusting the HOC (by having $\gamma \neq 0$) will not impact the MI [at least up to $O(\epsilon^4)$]. Because the inclusion of HOC lowers the response entropy—which can reduce the MI—we conjecture that the optimal population code in this case will have no HOC. This conjecture is supported by our numerical investigation [below, Figs. 3(a) and 3(c)].

A second prediction is that, for this case of symmetric stimulus distributions, the best MI is obtained when the pairwise interactions between neurons—which determine the signs of $\text{cov}(\sigma_i,\sigma_i)$ in Eq. (5)—have the same signs as the correlations between their stimulus-dependent gains $\langle h_i^s h_j^s \rangle$. This finding is consistent with the numerical results obtained by [14], in the case of small β (which, in their work, parametrizes both the strength of the stimulus coupling and of the interactions between cells). We note that, because our analytical expansion [Eq. (5)] holds only in the limit of small β, we cannot use this expansion to understand the numerical results that [14] obtained in the limit of large β.

In the case of skewed stimulus distributions, the $O(\epsilon^3)$ terms in Eq. (5) are nonzero, and thus changes to the HOC—by adjusting the third-order interaction γ—can increase the MI: this is confirmed by the numerical results shown in Figs. 3(b), 3(d), and 3(f) (below). Unfortunately it is not straightforward to determine the optimal sign of γ from Eq. (5) in the case of unskewed stimulus distributions, because triplet moments in the response distribution will be simultaneously determined by the J and γ. However, Eq. (5) does allow us to understand how HOC might impact coding in large neural populations: in a population of N neurons, there are $O(N^3)$ of the cubic terms in Eq. (5) (that depend on HOC), but only $O(N^2)$ of the quadratic terms (that do not depend on the HOC). Consequently, in the limit of large N, the HOC could have a large impact on the MI.

C. When do HOC improve coding? Numerical results

Our analytical investigation identified situations—of skewed stimulus distributions—in which we expect HOC to enhance the population code. To verify this finding (and investigate the extent to which it applies to stronger stimulus-coupling regimes than could be considered by our series expansion), we numerically seek the h_i^0, J_{ij}, and γ_{ijk} in Eq. (1) that maximize the MI [Eq. (3)]. We repeat this optimization for different values of β and thus different strengths of stimulus coupling: note that, in Eq. (1), β multiplies the stimulus, and thus it sets the strength of the stimulus coupling.

To simplify our numerical investigation, we consider homogeneous parameter values: $h_i^0 = h^0$, $J_{ij} = J$, and $\gamma_{ijk} = \gamma \forall i,j,k$. For consistency with this, we use permutation-symmetric stimulus distributions. However, for any given stimulus example $\vec{h}^s$, the conditional response distribution will not necessarily be permutation symmetric. For a given set of model parameters, we numerically compute the MI using Monte Carlo methods, and we optimize over the parameters using gradient ascent (see Appendix C).

As a control, to see if and when the HOC really do lead to better MI than could be obtained by a model with no higher-order correlations, we also optimize MI over h^0 and J while imposing the constraint $\gamma = 0$, the *triplet forbidden* case. In this case, the conditional response distribution becomes the pairwise maximum entropy model, which is the situation considered by [14] (here, with homogeneous parameters). Comparing the maximum attainable MI with triplet interactions allowed or forbidden, we ascertain when, and how much, their presence improves coding. This is related to third-order connected information [24], where one fits both second- and third-order ME models to the stimulus-conditioned response distributions and compares the resulting MI. In the case of [24] (and in the recent work of [20]), the second-order moments are held fixed as the third-order moments are varied, whereas we separately *optimize* the interaction parameters controlling those moments in the two cases. Because we separately optimize the second- and third-order models, we obtain more conservative estimates of how MI increases due to the third-order interactions than did [20].

We note that, when triplet interactions are allowed, the optimum can still occur for $\gamma = 0$. In this case the maximal MI will be equal for networks allowing and forbidding triplet interactions. Thus, because the triplet-allowed model space is a superset of the triplet-forbidden one, it will never be the case that the optimized triplet-allowed network performs worse than the optimized triplet-forbidden one.

We begin with the case of jointly Gaussian stimuli [Fig. 2(a)] of varying levels of correlation ρ. Recall that our analytical results [Eq. (5), above] suggest that, in this case, it is only the second-order moments of the response distribution that affect the MI, and that we conjectured based on this that HOC confer no coding benefit. This is borne out by our numerical analysis [Figs. 3(a) and 3(c), $N = 10$ neurons]: even when triplet interactions are allowed, the optimal encoder has $\gamma = 0$ [Fig. 4(a)]. Here, the stimulus distribution is symmetric about the mean, and the optimal encoder is on-off symmetric with $\langle \sigma \rangle = 0.5$. That symmetry—with neurons being in either the "spiking" (1) or "nonspiking" state (0) with equal probability—maximizes response entropy. When the stimuli are drawn from discrete binary distributions with equal probabilities for the two states, we also find that triplet interactions confer no coding advantage (data not shown). These observations support our analytical finding that, for unskewed stimulus distributions, with on-off symmetric response distributions, triplet interactions are not useful for coding. Moreover, these numerical results hold over a range of β values, and thus a large range of stimulus-coupling strengths (and not just the weak ones considered in our analytical calculation).

To seek situations when triplet interactions might be beneficial to the population code, we break the on-off symmetry.
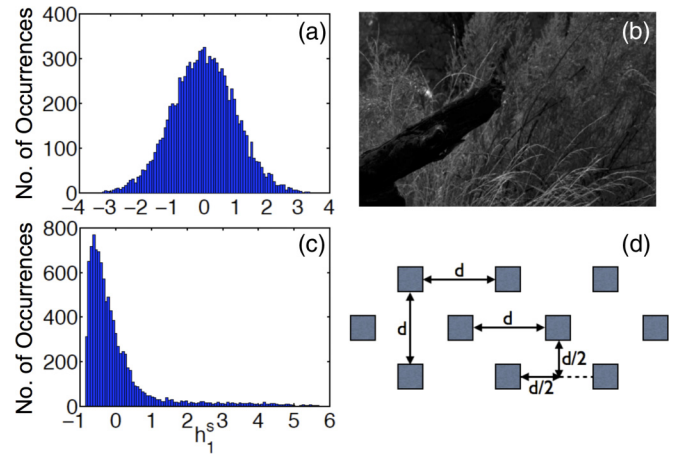

FIG. 2. (Color online) Stimuli considered in this work. (a) Marginal histogram of one stimulus, h_1^s, for the jointly Gaussian ensemble. (b) One of the natural luminance images used in this work, from the database of Tkačik *et al.* [36]. The marginal histogram of pixel values for this set of images (normalized to have zero mean and unit variance) is skewed (c). To generate our naturalistic stimulus ensemble, we randomly draw dectuplets of pixel values with spacing d pixels by placing the template (d) at a random location on a randomly chosen image, and setting stimulus values h_i^s to match the pixel values falling under each square. To maintain permutation symmetry of our stimulus ensemble, we permute which square corresponds to which stimulus index i for each draw. The marginals [(a),(c)] are the same for all stimulus indices i.

We do this in two different ways, the first of which is to consider skewed stimulus distributions. The case of skewed stimulus distributions is potentially important because the distribution of membrane potentials (neuronal inputs) in rat auditory cortical neurons is skewed [34], as are the spike-count distributions in dichotomized Gaussian models of neural population activity [31]. Moreover, naturalistic stimuli (like pixel values in real-world images) have skewed distributions [Figs. 2(b) and 2(c)]. Consequently, the inputs to real biological neurons—either from external stimuli, or from other neurons within the brain—may show nonzero skew [35]. We use as stimuli calibrated luminance images from the database of Tkačik *et al.* [36].

It is worth noting that images of natural scenes (as one might collect with a digital camera in, say, a forest) have statistical properties that differ markedly from purely random white-noise images. Much theoretical and empirical work has investigated these properties [37–43]. Of particular note are the rich correlation structure, and the power-law power spectra: natural images have autocorrelation functions (Fourier transform of the power spectra) that decrease with distance [39], in a manner that is surprisingly independent of the occlusion property of objects in those images [42,43]. By drawing groups of pixels (Fig. 2) with variable spacing d, we vary the level of correlation between stimulus values. Since luminance (or photon count) is non-negative, but can be arbitrarily large, this distribution is skewed [Fig. 2(c)].

For the natural image luminance stimuli, we find that triplet interactions indeed confer a coding advantage. For $N = 10$ cells, this is a 5%–10% improvement in MI compared to the
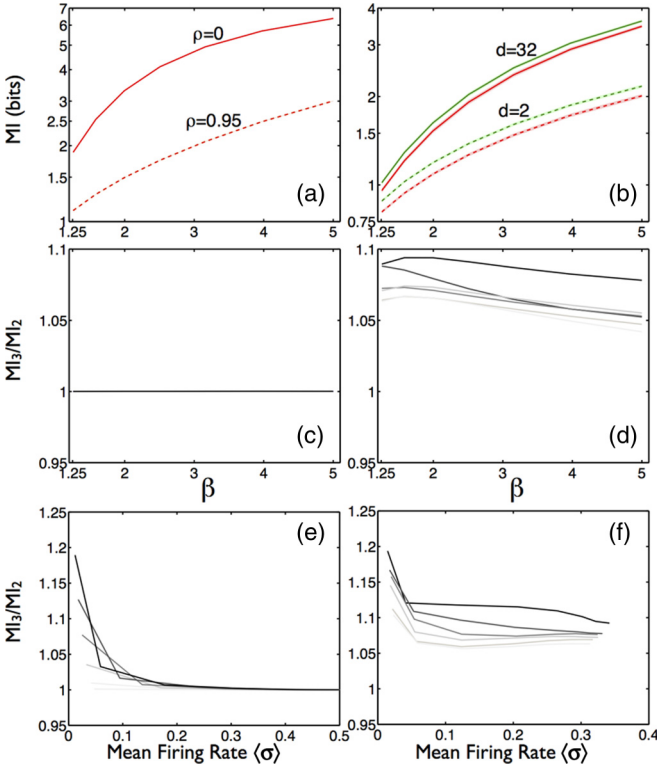
FIG. 3. (Color online) Triplet interactions can improve encoding of stimuli. Panels (a),(c),(e) show results for Gaussian-distributed stimuli, whereas panels (b),(d),(f) are for natural image pixel stimuli, which have skewed distributions. (a) For jointly Gaussian stimuli with pairwise correlation coefficients of 0.95 (dashed lines) or 0 (solid lines), encoders with triplet interactions allowed (green) or forbidden (red, $\gamma = 0$) have the same coding performance, which increases with neural reliability β. (b) For natural image stimuli with pixel spacings of $d = 32$ (solid lines) or $d = 2$ (dashed lines), the triplet-allowed encoder (green) performs better. The shaded regions (similar in thickness to the lines) around the lines in panels (a),(b) show the standard deviation of the mean MI over five repeats of the optimization procedure with different sets of random stimuli. (c),(d) To summarize how performance gains vary with correlation level, we plot the ratio of the MI for the optimal triplet-allowed networks (MI_3) to the one for triplet-forbidden networks (MI_2, $\gamma = 0$) as a function of β. The darkest curve corresponds to the largest correlation [$\rho = 0.95$ for the Gaussian (c) and $d = 2$ for the natural image stimuli (d)], the lightest curve corresponds to the smallest correlation ($\rho = 0$ for the Gaussian and $d = 32$ for the natural image stimuli), and intermediate shades correspond to intermediate levels of correlation. (e),(f) For $\beta = 1.5$, we similarly plot the performance ratio as a function of mean firing rate for Gaussian (e) and natural image (f) stimuli, in cases with constrained firing rates (see text). In order to make a fair comparison, we estimate the MI of the triplet-allowed and triplet-forbidden networks at the same firing rate (see Appendix C for details). $N = 10$ neurons for all cases.

optimized purely pairwise encoder [Figs. 3(b) and 3(d)]. The advantage is largest for close-by sampled pixels (small d), and at relatively low values of β (i.e., relatively unreliable neurons). Natural images have rich beyond-pairwise statistics [38]. Is that why triplet interactions improve encoding for natural image stimuli? No: repeating these optimization experiments
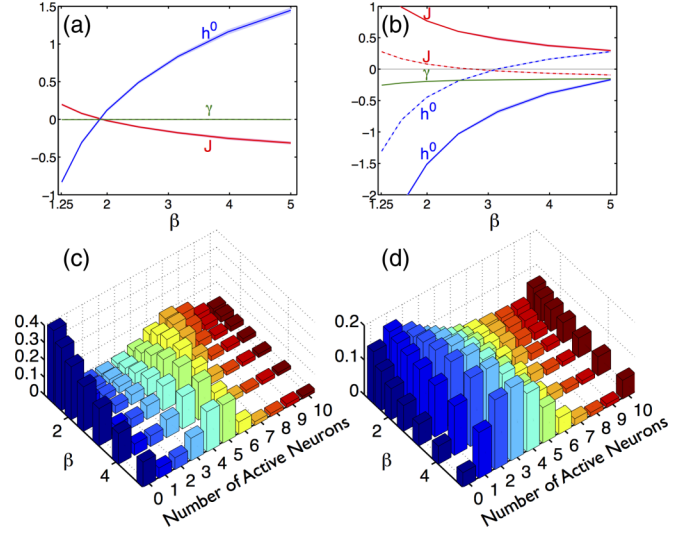

FIG. 4. (Color online) Triplet interactions, when they are beneficial, sparsify the neural representation of stimuli. (a) For jointly Gaussian stimuli ($\rho = 0.95$) and no firing rate constraint, optimal encoders have no triplet interaction (γ, green). At low β, these optimal encoders have pairwise interactions (J, red) that enhance the input correlations, whereas at high β they oppose them [14]. The optimal biases h^0 (blue) cancel the effective mean-field bias from pairwise interactions, $h_{\mathrm{eff}} = (N - 1)J\langle\sigma\rangle$. (b) For the natural image pixel stimuli ($d = 2$), the optimal encoders have negative triplet interactions (when triplet interactions are allowed: solid lines) for all β—the encoder parameters do not change sign. When triplet interactions are forbidden (dashed lines), the magnitudes of the pairwise interaction and bias are smaller, and do change sign with increasing β. The solid horizontal line (at zero) is to guide the eye. The shaded regions (similar in thickness to the lines) around the lines in panels (a),(b) show the standard deviation of the mean parameter values over five repeats of the optimization procedure with different sets of random stimuli. For other levels of stimulus correlation, we find qualitatively similar results (not shown). A comparison of the response distributions of the optimal encoders for the natural image pixel stimuli ($d = 2$) with triplet interactions allowed (c) or forbidden [(d), $\gamma = 0$] shows that the triplet interactions sparsify the responses by reducing the probability of the state in which all neurons are active.

using linear mixtures of variables from skewed Pearson-system marginal distributions as stimuli, we also observed that triplet interactions improved coding (data not shown). These results support our analytical findings that, in the case of skewed stimulus distributions, triplet interactions can improve coding performance.

While our analytical results focused on the statistics of the stimulus distribution, one can also break the on-off symmetry of our problem by restricting the firing rates (FR's) of the neurons in our networks. Thus far, they have been allowed to take arbitrary values. Empirically, however, neurons are seen to fire infrequently [2,5,9,44,45], with mean FR's of a few Hz: for 10–20 ms time bins [2,5,9] this yields $\langle\sigma\rangle \sim 0.01$–0.1. To incorporate this constraint, we maximized a Lagrange function $\mathcal{L} = MI - \lambda\langle\sigma\rangle$ that disfavors high FR's [14,46], similar to the notion of *sparse coding* [46–50]. By varying λ, we alter the mean FR of the optimal network [14]. We ask how the

MI for these optimal networks varies as a function of their mean FR for networks with triplet interactions either allowed or forbidden. For these investigations, we restricted ourselves to $\beta = 1.5$.

Intriguingly, for jointly Gaussian stimuli triplet interactions improve coding performance at sufficiently low firing rates [Fig. 3(e)]. The improvement is larger for stronger stimulus correlations. This effect is somewhat surprising because our analytical results [Eq. (5)] show that, up to $O(\epsilon^3)$, MI is not increased by the inclusion of HOC when the stimulus distribution is unskewed. We thus ascertain that the results in Fig. 3 (e) arise at higher-order terms in the series expansion. The key intuition is that, when the encoding problem is sufficiently symmetric (i.e., the stimulus distribution is symmetric about the mean *and* no firing rate constraint prevents the response distribution from being on-off symmetric), the best encoders have no triplet interactions.

For natural image pixel stimuli (with skewed distributions), restricting firing rates lead to further benefits of triplet interactions, over and above those already seen with unconstrained firing rates. In particular, for low firing rates the benefits of triplet interactions can be as large as 15%–20% [Fig. 3(f)].

The results shown herein are for networks of $N = 10$ neurons. This is as large as we can consider while being able to numerically optimize our MI function with reasonable speed (see Appendix C for methods). However, we emphasize that our analytical results [Eq. (5)] hold for arbitrary N, and in fact suggest the effect of higher-order interactions on MI may grow with population size (see discussion of analytic calculations above).

D. How do HOC improve coding?

To understand how the HOC improve the population code, we investigated the parameters of the optimized models: the J, h^0, and γ that maximized the MI in different cases.

For skewed natural image pixel stimuli, the negative ($\gamma < 0$) triplet interactions we observed at optimality [Fig. 4(b)]*sparsify* neural responses by reducing the frequency of multispike synchrony in which many neurons fire simultaneously. This sparsifying role of triplet interactions agrees with experimental findings [9] and mechanistic modeling [30,31,51]. Importantly, $\gamma < 0$ is optimal even in the absence of a FR constraint, pointing to a richer role in shaping response distributions.

Following [14], we first note that (at least at small β when $\gamma = 0$, and for all β when $\gamma \neq 0$) the optimal encoders have positive J [Figs. 4(a) and 4(b)]. Thus pairwise network interactions reinforce the positive correlations already present in the stimulus, which [14] interpret as an error-reducing property: responses tend to be constrained to a smaller set of possibilities. This effect can be beneficial only up to a point: for very large positive J, neurons would all fire synchronously regardless of the stimulus, sharply reducing response entropy. There is therefore a trade-off between the desiderata of error reduction (J reinforces correlations) and high response entropy (J opposes correlations).

Triplet interactions impact the tradeoff in a novel way: $\gamma < 0$ combats multispike synchrony, so that response entropy can be maintained even with J reinforcing stimulus correlations

[52]. Triplet interactions are more suited than pairwise ones at specifically suppressing multispike states, in line with the observation that $\gamma < 0$ and $J > 0$ at optimality, and not vice versa (see Appendix D). The response distributions of optimal encoders with triplets allowed [Fig. 4(c)] or forbidden [Fig. 4(d)] support this notion: even with no constraints on the FR, the triplet-allowed network makes less use of the state in which all neurons are active. Moreover, with triplet interactions forbidden, the optimal encoders have smaller J [Fig. 4(b)], also consistent with the interpretation above. Finally, we observed $\gamma < 0$ to be optimal when we constrained the firing rates as well (data not shown).

Allowing nonzero triplet interactions yields optimized network parameters J and γ that do not change sign as β is varied. This stands in contrast to the case of $\gamma = 0$, for which the encoder parameters change sign as β is varied (Fig. 4): at low β, they reinforce the stimulus correlations, while at high β, they oppose them. This behavior is dictated by the trade-off between noise and response entropies described above [14].

III. DISCUSSION AND CONCLUSIONS

We have shown that, in the case of skewed stimulus distributions, or of low neural firing rates, neural population coding efficacy can be enhanced by third-order correlations between neurons.

A. What about fourth-order and higher interactions?

While we have herein restricted ourselves to third-order interactions, the same methods would apply equally well to higher-order models [with fourth- or fifth-, or higher-order terms in the exponential in Eq. (1)]. This naturally begs the question: "What order is high enough?" In other words, what is the highest order of interactions that one must consider in order to understand information transmission in neural systems? Evidence from available experiments seems to suggest an encouraging answer. For recordings of ≈ 100 cells from retina being stimulated with naturalistic movies [5] it appears that, even at fourth order, the number of nonzero interactions is quite small, and by sixth order, there are none. Recordings from somatosensory cortex [7] also suggest that the order of needed interactions will be much smaller than the recorded number of neurons. To summarize, the type of approach used in this paper could be extended to higher orders, and available experiments suggest that such a venture will have a well-defined stopping point: once we understand how interactions up to order five or six affect coding, we could be largely done.

B. Potential biophysical origins of beyond-pairwise interactions

In our model, higher-order interactions can arise from the nonlinear combination of recurrent inputs. Neurobiology provides several processes which can affect such nonlinear combinations. Even for passive single-compartment neurons (no dendrites), inputs can combine sublinearly, as follows. Synaptic inputs open ion channels, moving the membrane potential towards that ion's reversal potential [53]. Opening subsequent channels creates less current as there is less driving force pushing ions through the channel [54]. Dendrites have additional properties that yield nonlinearities [54–57].

This allows flexible higher-order interactions: both super- and sublinear dendritic summation are observed when two inputs impinge on the same dendritic branch [58], while inputs to separate branches combine linearly. For strong dendritic inputs, the observed integration properties are sublinear [58], corresponding to negative triplet interactions, similar to what we observed (Fig. 4) for optimal coding.

We emphasize that, in our probability model, the nonlinear integration is symmetric for the cell triplets [Eq. (1)]. In other words, if neuron i superlinearly combines the inputs from neurons j and k, then neuron j superlinearly combines the inputs from cells i and k, etc. While this is a highly constrained case, it is physically realizable. At the same time, the nonlinear statistical interactions in our probability model can also arise from common noise inputs to multiple neurons [30–32]. In that case, the interactions will naturally be somewhat symmetrical.

We note that caution is warranted when making mechanistic interpretations of statistical parameters observed in neural data. Pairwise interactions in those data do not necessarily reflect synaptic couplings, as there may be common input to both neurons from unobserved cells that are the cause of the correlation. Similar remarks apply to higher-order interactions, which can also be driven by "hidden" (unobserved) cells [59], spike-generating nonlinearities [30,31,51], and other mechanisms [24,30].

C. Implications for machine learning and computer vision

In both the mammalian visual system, and in Boltzmann-type computer vision algorithms [21,22], continuous-valued pixel intensities are encoded into binary. These codewords describe the states of all computational units in the system, each of which are either "on" or "off" at any given moment. This binarization naturally leads to a loss of information about the image, potentially hindering these systems' abilities to perform the hard computational task of object recognition. Herein, we investigated the features that can minimize the information loss in binary image processing systems. Our results indicate that pairwise and beyond-pairwise statistical interactions between computational units can improve the performance of Boltzmann-type image encoders, like deep belief networks [21]. In other words, we identify and begin to explain an information-theoretic role for recurrent interactions *within* such network layers. At the same time, these higher-order interactions might be difficult to optimize (i.e., to find suitable learning rules).

D. Summary and conclusions

We have demonstrated that input nonlinearities can generate beyond-pairwise interactions in spin-glass statistical models of neural population activity, and observed that—under biolog-

ically relevant conditions—these nonlinearities can improve population coding. In particular, we find that the third-order interactions can improve coding when the stimulus distribution is skewed, and/or when the neurons are restricted to have reasonably low firing rates. Normative theories might thus predict differences in the summation properties of neurons in networks that are evolved (or adapted) to encode different types of stimuli, or in networks with different pressures to regulate firing rates.

APPENDIX A: TRIPLET INTERACTIONS ARISE FROM THE FIRST NONLINEAR TERMS IN A SERIES EXPANSION OF THE INPUT TO OUR MODEL NEURON

If we let the recurrent input to neuron i be an arbitrary function of the activities of the other neurons, $x_i = h_i + f(\{\sigma_{j \neq i}\})$, triplet interactions arise from the first nonlinear terms in the series expansion $x_i = h_i + \sum_{j \neq i} a_{ij}\sigma_j + \sum_{j \neq i} b_{ij}\sigma_j^2 + \sum_{j,k \neq i} c_{ijk}\sigma_j\sigma_k + \cdots$, where a_{ij}, b_{ij}, c_{ijk} are the series coefficients, and we have omitted the constant in the expansion. Since $\sigma_j \in \{0,1\}$, $\sigma_j^2 = \sigma_j$, and the $b_{ij}\sigma_j^2$ terms can be grouped with the $a_{ij}\sigma_j$ ones, this yields $x_i = h_i + \sum_{j \neq i} J_{ij}\sigma_j + \sum_{j,k \neq i} \gamma_{ijk}\sigma_j\sigma_k + \cdots$, where $J_{ij} = a_{ij} + b_{ij}$ and $\gamma_{ijk} = c_{ijk}$.

APPENDIX B: ANALYTIC CALCULATIONS OF MUTUAL INFORMATION

This section is organized as follows: we first parametrize the strength of the stimulus-coupling (ϵ) in the probability model. Next, we rewrite the mutual information in a more convenient form. Then, we exploit that rewrite to expand the MI in powers of ϵ.

1. Problem (re)statement

We want to compute the mutual information (MI) between neural responses $\vec{\sigma}$, where $\sigma_i \in \{0,1\}$ is the spiking or not spiking of a neuron in a given time bin, and the stimuli $\vec{h^s}$. We do this by writing down the conditional PDF, and computing conditional and marginal entropies. Following the main paper [Eq. (4)], let's write down the conditional PDF as

$$p(\vec{\sigma}|\vec{h^s}) = \frac{1}{Z(\vec{h^s})} \exp\left[\epsilon\vec{h^s} \cdot \vec{\sigma} + \beta\left(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k \right) \right], \tag{B1}$$

where the pairwise and triplet interactions are given by J and γ, and β describes the strength of those interactions. Note that the diagonal elements of J can be thought of as the bias terms (h_i^0 in the main paper) for the neurons. As in Eq. (4) of the main paper, parameter ϵ describes the strength of the coupling between stimuli and neural responses. $Z(\vec{h^s})$ is the conditional partition

function,

$$Z(\vec{h^s}) = \sum_{\{\vec{\sigma}\}} \exp\left[\epsilon \vec{h^s} \cdot \vec{\sigma} + \beta\left(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k\right)\right].$$ (B2)

The mutual information can then be computed as

$$I(\vec{\sigma}; \vec{h^s}) = -\sum_{\{\vec{\sigma}\}} p(\vec{\sigma})\log[p(\vec{\sigma})] + \int d\vec{h^s}\, p(\vec{h^s})\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}|\vec{h^s})\log[p(\vec{\sigma}|\vec{h^s})].$$ (B3)

Note that these two terms look like log(p) averaged over some distributions, which requires that we know the partition function (i.e., the $\log[Z(\vec{h^s})]$ terms). Computing partition functions is *hard*, since it requires a sum over all states, and there are 2^N states of a system with N neurons. The "α method," described below, gets around this difficulty.

2. "α method"

a. Rewriting the conditional PDF

Looking at the conditional PDF, we notice that (ignoring for now the partition function), it factorizes into two parts. First, the stim-coupling part $\alpha = \exp[\epsilon(\vec{h^s} \cdot \vec{\sigma})]$. And, next, the "network interactions" part $\psi = \exp[\beta(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k)]$. Consider, for now, the partition function when there is no stimulus present, so that $\vec{h^s} = \vec{0}$. In that case, $\alpha = 1$, and the partition function becomes

$$Z(\vec{h^s} = 0) = \sum_{\vec{\sigma}} \exp\left[\beta\left(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k\right)\right]$$

$$= \sum_{\vec{\sigma}} \psi.$$ (B4)

Call this "zero-stim" partition function Z_0. We can then notice that PDF of the spontaneous activity distribution (or zero-stim distribution) is

$$p(\vec{\sigma}|\vec{h^s} = 0) = \frac{1}{Z_0}\exp\left[\beta\left(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k\right)\right]$$

$$= \frac{\psi}{Z_0}.$$ (B5)

And, accordingly, the non-zero-field partition function is

$$Z(\vec{h^s}) = \sum_{\vec{\sigma}} \exp\left[\epsilon(\vec{h^s} \cdot \vec{\sigma}) +\right.$$

$$\left. \times \beta\left(\sum_{ij} J_{ij}\sigma_i\sigma_j + \sum_{ijk} \gamma_{ijk}\sigma_i\sigma_j\sigma_k\right)\right]$$

$$= \sum_{\vec{\sigma}} \alpha\psi$$

$$= \sum_{\vec{\sigma}} \alpha Z_0 p(\vec{\sigma}|\vec{h^s} = 0) = Z_0 \mathbb{E}_{SAD}[\alpha],$$ (B6)

where $\mathbb{E}_{SAD}[\cdot]$ represents an expectation over the spontaneous activity distribution (SAD). Finally, using this bit of notation, we can rewrite our conditional PDF as

$$p(\vec{\sigma}|\vec{h^s}) = \frac{1}{Z(\vec{h^s})}\alpha\psi$$

$$= \frac{\alpha\psi}{Z_0 \mathbb{E}_{SAD}[\alpha]}$$

$$= \frac{\alpha p(\vec{\sigma}|\vec{h^s} = 0)}{\mathbb{E}_{SAD}[\alpha]}.$$ (B7)

This, on its own, doesn't look like much of a simplification. However, notice that we no longer have a conditional partition function to evaluate. In its stead, we have the zero-field partition function, and an expectation of α over the SAD. The zero-field partition function will be the same for the conditional and marginal distributions, and will thus cancel when we compute mutual information (below).

b. Simplifying the mutual information

We use another form for the mutual information, that is equivalent to Eq. (B3), but a bit more convenient for our purposes:

$$I(\vec{\sigma}; \vec{h^s}) = \int d\vec{h^s}\, p(\vec{h^s})\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}|\vec{h^s})\log[p(\vec{\sigma}|\vec{h^s})/p(\vec{\sigma})].$$ (B8)

To evaluate this, we need to know the marginal distribution over responses, which is

$$p(\vec{\sigma}) = \int d\vec{h^s}\, p(\vec{h^s})p(\vec{\sigma}|\vec{h^s})$$

$$= p(\vec{\sigma}|\vec{h^s} = 0)\mathbb{E}_{\vec{h^s}}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right],$$ (B9)

where $\mathbb{E}_{\vec{h^s}}[\cdot]$ is an expectation over the stimulus distribution. Using all of our above observations, we can rewrite the mutual information as

$$I(\vec{\sigma}; \vec{h^s})$$

$$= \mathbb{E}_{\vec{h^s}}\left[\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}|\vec{h^s} = 0)\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\log\left(\frac{\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}}{\mathbb{E}_{\vec{h^s}}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]}\right)\right]$$

$$= \mathbb{E}_{\vec{h^s}}\left[\mathbb{E}_{SAD}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\log\left(\frac{\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}}{\mathbb{E}_{\vec{h^s}}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]}\right)\right]\right],$$ (B10)

where the factors of $p(\vec{\sigma}|\vec{h}^s = 0)$ in the logarithm cancel because they appear in both the numerator and denominator of the fraction, and the second line follows from the first because the sum over all states reduces to an average over the spontaneous activity distribution. We now observe that the problem of computing mutual information, which previously required a *hard* partition function calculation, has been reduced to computing expectations of $\alpha = \exp(\epsilon \vec{h}^s \cdot \vec{\sigma})$ over both the spontaneous activity distribution, and over the stimulus distribution.

Finally, notice that, while we have used a specific functional form for $p(\vec{\sigma}|\vec{h}^s)$ in our calculations, the same logic applies to any conditional PDF of the form $p(\vec{\sigma}|\vec{h}^s) = \alpha(\vec{h}^s, \vec{\sigma})\psi(\vec{\sigma})$, with the condition $\alpha(\vec{h}^s = 0, \vec{\sigma}) \neq 0$. Intuitively, the variability in responses due to $\psi(\vec{\sigma})$ is the same for all stimuli, so it provides the same contribution to the marginal and conditional entropies, and that contribution cancels when we compute mutual information.

3. Analytically relating MI to moments of the stimulus and response distributions

Our MI calculation reduced to Eq. (B10) expectations over the SAD and the stimulus distribution. We know that we can write out the exponentials, logarithms, and ratios as power series in $\epsilon \vec{h}^s \cdot \vec{\sigma}$, and the MI is an average of those things over the SAD and stimulus distributions. This means that we can analytically write down the MI as a function of moments of the

SAD and stimulus distributions. Let's do that, but first break up Eq. (B10) into a few easier-to-manage terms:

$$
\begin{aligned}
I(\vec{\sigma}; \vec{h}^s) &= \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\log\left(\frac{\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}}{\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]}\right)\right]\right] \\
&= \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\log(\alpha)\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right] \\
&\quad - \mathbb{E}_{\vec{h}^s}\left[\log\left(\mathbb{E}_{SAD}[\alpha]\right)\right] \\
&\quad - \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\right]\right].
\end{aligned}
\tag{B11}
$$

a. Dealing with the third term in the above equation

The first few terms are (relatively) simple, so let's focus for now on the third term $(-T_3)$ in the above equation, with

$$
T_3 = \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\right]\right]. \tag{B12}
$$

We'll expand to third order in ϵ, and thus restrict ourselves to small ϵ. In so doing, we will find that $T_3 = 0 + O(\epsilon^4)$, and hence that we can ignore it in the small ϵ limit. To simplify our notation, let $s = \vec{\sigma} \cdot \vec{h}^s$ (the s is for "sum"), such that $\alpha = \exp(\epsilon s)$. We first note that $T_3(0) = 0$, because $\alpha(\epsilon = 0) = 1 \forall s$, and thus the ratio in the logarithm is unity. Since they will keep coming up in our calculation, we'll note straight away that

$$
\frac{\partial}{\partial \epsilon}\left(\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right) = \frac{s\alpha E_{SAD}[\alpha] - \alpha E_{SAD}[s\alpha]}{(E_{SAD}[\alpha])^2}, \quad \frac{\partial^2}{\partial \epsilon^2}\left(\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right) = \Omega,
$$

$$
\Omega = \frac{s^2\alpha(\mathbb{E}_{SAD}[\alpha])^2 - \alpha\mathbb{E}_{SAD}[s^2\alpha]\mathbb{E}_{SAD}[\alpha] - 2s\alpha\mathbb{E}_{SAD}[s\alpha]\mathbb{E}_{SAD}[\alpha] + 2\alpha(\mathbb{E}_{SAD}[s\alpha])^2}{\mathbb{E}_{SAD}^3[\alpha]}, \tag{B13}
$$

where we have defined Ω for our convenience. Then,

$$
T_3(\epsilon) = T_3(0) + \epsilon\left.\frac{\partial T_3}{\partial \epsilon}\right|_{\epsilon=0} + \frac{\epsilon^2}{2}\left.\frac{\partial^2 T_3}{\partial \epsilon^2}\right|_{\epsilon=0} + \frac{\epsilon^3}{6}\left.\frac{\partial^3 T_3}{\partial \epsilon^3}\right|_{\epsilon=0} + O(\epsilon^4). \tag{B14}
$$

To proceed further, we need to evaluate the derivatives in Eq. (B14):

$$
\begin{aligned}
\frac{\partial T_3}{\partial \epsilon} &= \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\frac{\alpha}{\mathbb{E}_{SAD}}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}}\right]\right)^{-1}\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right]\right] \\
&\quad + \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right].
\end{aligned}
\tag{B15}
$$

Conveniently, the first term in Eq. (B15) (above) vanishes. To see this, notice that $(\mathbb{E}_{\vec{h}^s}[\frac{\alpha}{\mathbb{E}_{SAD}}])^{-1}$ and $\mathbb{E}_{\vec{h}^s}[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}]$ are independent of $\vec{h}^s$, being already expectation values over $\vec{h}^s$. If we change the order of our expectations over SAD and $\vec{h}^s$, then $\frac{\alpha}{\mathbb{E}_{SAD}}$ and $(\mathbb{E}_{\vec{h}^s}[\frac{\alpha}{\mathbb{E}_{SAD}}])^{-1}$ cancel. What remains is (again, swapping the order in which we compute expectations)

$$
\begin{aligned}
\mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right] &= \mathbb{E}_{\vec{h}^s}\left[\frac{\mathbb{E}_{SAD}[s\alpha]\mathbb{E}_{SAD}[\alpha] - \mathbb{E}_{SAD}[\alpha]\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right] \\
&= 0.
\end{aligned}
\tag{B16}
$$

With this simplification, we have

$$
\frac{\partial T_3}{\partial \epsilon} = \mathbb{E}_{SAD}\left[\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right], \tag{B17}
$$

which is zero when evaluated at $\epsilon = 0$, due to the ratio of α's in the logarithm. There is thus no first-order contribution to T_3. We now require the second (and eventually third) derivative in our Taylor series:

$$\frac{\partial^2 T_3}{\partial \epsilon^2} = \mathbb{E}_{SAD}\left[\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}[\Omega]\right] + \zeta,$$

$$\zeta = \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-1}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)^2\right]. \tag{B18}$$

As with the first derivative, when we evaluate this at $\epsilon = 0$, the term with the logarithm vanishes, leaving

$$\left.\frac{\partial^2 T_3}{\partial \epsilon^2}\right|_{\epsilon=0} = \zeta\Big|_{\epsilon=0}$$

$$= \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}[s] - \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}[s]\right]\right)^2\right]$$

$$\Rightarrow \left.\frac{\partial^2 T_3}{\partial \epsilon^2}\right|_{\epsilon=0} = \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}[s]\right)^2\right] - \left(\mathbb{E}_{SAD,\vec{h}^s}[s]\right)^2. \tag{B19}$$

The second derivative was messy to calculate [Eq. (B18)], and now we require the *third* derivative, which is even worse. We'll save ourselves a bit of effort by noting that the term that contains a logarithm will vanish when evaluated at $\epsilon = 0$, and so we won't compute it at all. With that in mind,

$$\frac{\partial^3 T_3}{\partial \epsilon^3} = \mathbb{E}_{SAD}\left[\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}\left[\frac{\partial\Omega}{\partial\epsilon}\right]\right]$$

$$+ \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-1}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}[\Omega]\right] + \frac{\partial\zeta}{\partial\epsilon}, \tag{B20}$$

where Ω and ζ are as defined in Eqs. (B13), (B18), and

$$\frac{\partial\zeta}{\partial\epsilon} = 2\mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-1}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}[\Omega]\right]$$

$$- \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-2}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)^3\right]. \tag{B21}$$

Putting all of these pieces together, we observe that

$$\frac{\partial^3 T_3}{\partial \epsilon^3} = \mathbb{E}_{SAD}\left[\log\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}\left[\frac{\partial\Omega}{\partial\epsilon}\right]\right] + 3\mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-1}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)\mathbb{E}_{\vec{h}^s}[\Omega]\right]$$

$$- \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}\left[\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right)^{-2}\left(\mathbb{E}_{\vec{h}^s}\left[\frac{s\alpha\mathbb{E}_{SAD}[\alpha] - \alpha\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]\right)^3\right], \tag{B22}$$

where (again) Ω is as defined in Eq. (B13). Let's now evaluate this at $\epsilon = 0$:

$$\left.\frac{\partial^3 T_3}{\partial \epsilon^3}\right|_{\epsilon=0} = 3\mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}[s] - \mathbb{E}_{SAD,\vec{h}^s}[s]\right)\mathbb{E}_{\vec{h}^s}[\Omega|_{\epsilon=0}]\right]$$

$$- \mathbb{E}_{SAD}\left[\left(\mathbb{E}_{\vec{h}^s}[s] - \mathbb{E}_{SAD,\vec{h}^s}[s]\right)^3\right]. \tag{B23}$$

This looks pretty hairy, but we'll note that all of our terms are multiplied by either $\mathbb{E}_{\vec{h}^s}(s)$ or $\mathbb{E}_{SAD,\vec{h}^s}(s)$. These will vanish for zero-mean stimulus distributions. Since one can compensate for nonzero mean along any stimulus distribution by changing the bias for the corresponding neuron [the diagonal elements J_{ii} in Eq. (B2)], then WLOG we can assume

the stimuli are zero mean. This means that

$$\mathbb{E}_{\vec{h}^s}[s] = \mathbb{E}_{\vec{h}^s}\left[\sum_i h_i\sigma_i\right]$$

$$= \sum_i \mathbb{E}_{\vec{h}^s}[h_i]\sigma_i$$

$$= 0\forall\vec{\sigma}. \tag{B24}$$

Thus the third-order term vanishes. Looking back at Eq. (B19), one can similarly note that the second-order term vanishes, and (at least up to third order in ϵ), T_3 vanishes.

b. Looking back at the other terms in our MI formula

Noting (from above) that we can ignore T_3 up to third order in ϵ, we find that

$$
I(\vec{\sigma}; \vec{h^s}) = \mathbb{E}_{\vec{h^s}}\left[\mathbb{E}_{SAD}\left[\log(\alpha)\frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right]
$$
$$
- \mathbb{E}_{\vec{h^s}}\left[\log(\mathbb{E}_{SAD}[\alpha])\right] + O(\epsilon^4), \quad \text{(B25)}
$$

where we must now evaluate the two terms in Eq. (B25) in order to proceed. Using the fact that $\alpha = \exp(\epsilon s)$, and calling these terms T_1 and T_2,

$$
T_1 = \mathbb{E}_{\vec{h^s}}\left[\mathbb{E}_{SAD}\left[\epsilon s \frac{\alpha}{\mathbb{E}_{SAD}[\alpha]}\right]\right],
$$
$$
T_2 = \mathbb{E}_{\vec{h^s}}\left[\log\left(\mathbb{E}_{SAD}[\alpha]\right)\right]. \quad \text{(B26)}
$$

c. Expanding the first term, T_1

We'll start by expanding the T_1 term up to third order in ϵ, but first note that it already has a factor of ϵ in front of it [from the $\log(\alpha)$], so we will need do a second-order Taylor series of $\tilde{T}_1 = T_1/\epsilon$, and just multiply that whole thing by ϵ. Thus

$$
T_1(\epsilon) = \epsilon[\tilde{T}_1(\epsilon)]
$$
$$
= \epsilon\left[\tilde{T}_1(\epsilon = 0) + \epsilon\frac{\partial\tilde{T}_1}{\partial\epsilon}\bigg|_{\epsilon=0} + \frac{\epsilon^2}{2}\frac{\partial^2\tilde{T}_1}{\partial\epsilon^2}\bigg|_{\epsilon=0}\right] + O(\epsilon^4). \quad \text{(B27)}
$$

The zeroth-order term in the expansion vanishes [$\tilde{T}_1(\epsilon = 0) = \mathbb{E}_{SAD,\vec{h^s}}[s] = 0$] for the same reason our T_3 vanished [Eq. (B25)]. Again, we require derivatives to proceed, and [Eq. (B13)]

$$
\frac{\partial\tilde{T}_1}{\partial\epsilon} = \mathbb{E}_{SAD,\vec{h^s}}\left[s\frac{s\alpha E_{SAD}[\alpha] - \alpha E_{SAD}[s\alpha]}{(E_{SAD}[\alpha])^2}\right]. \quad \text{(B28)}
$$

Evaluating this at $\epsilon = 0$, we observe that

$$
\frac{\partial\tilde{T}_1}{\partial\epsilon}\bigg|_{\epsilon=0} = \mathbb{E}_{SAD,\vec{h^s}}[s(s - E_{SAD}[s])]
$$
$$
= \mathbb{E}_{SAD,\vec{h^s}}[s^2] - E_{\vec{h^s}}[(E_{SAD}[s])^2], \quad \text{(B29)}
$$

which does not necessarily vanish for zero-mean stimulus distributions. Carrying on to get the second derivative,

$$
\frac{\partial^2\tilde{T}_1}{\partial\epsilon^2} = \mathbb{E}_{SAD,\vec{h^s}}[s\Omega], \quad \text{(B30)}
$$

where Ω [Eq. (B13)] has made a reappearance. Evaluating Ω for $\epsilon = 0$, we find that

$$
\frac{\partial^2\tilde{T}_1}{\partial\epsilon^2}\bigg|_{\epsilon=0} = \mathbb{E}_{SAD,\vec{h^s}}\big[s\big(s^2 - \mathbb{E}_{SAD}[s^2] - 2s\mathbb{E}_{SAD}[s]
$$
$$
+ 2\big(\mathbb{E}_{SAD}[s]\big)^2\big)\big]
$$
$$
= \mathbb{E}_{SAD,\vec{h^s}}[s^3] - \mathbb{E}_{SAD,\vec{h^s}}\big[s\mathbb{E}_{SAD}[s^2]\big]
$$
$$
- 2\mathbb{E}_{SAD,\vec{h^s}}\big[s^2\mathbb{E}_{SAD}[s]\big]
$$
$$
+ 2\mathbb{E}_{SAD,\vec{h^s}}\big[s\big(\mathbb{E}_{SAD}[s]\big)^2\big]. \quad \text{(B31)}
$$

Thus, combining Eqs. (B27), (B29), and (B31), T_1 [Eq. (B26)] is

$$
T_1 = \epsilon^2\big(\mathbb{E}_{SAD,\vec{h^s}}[s^2] - E_{\vec{h^s}}[(E_{SAD}[s])^2]\big)
$$
$$
+ \frac{\epsilon^3}{2}\big(\mathbb{E}_{SAD,\vec{h^s}}[s^3] - \mathbb{E}_{SAD,\vec{h^s}}\big[s\mathbb{E}_{SAD}[s^2]\big]
$$
$$
- 2\mathbb{E}_{SAD,\vec{h^s}}\big[s^2\mathbb{E}_{SAD}[s]\big] + 2\mathbb{E}_{SAD,\vec{h^s}}\big[s(\mathbb{E}_{SAD}[s])^2\big]\big). \quad \text{(B32)}
$$

So, the first term in our MI function has been reduced to moments of the stimulus and spontaneous activity distributions, which we will later simplify.

d. Expanding the second term, T_2

Let's go over our second term now [T_2 in Eq. (B26)]:

$$
T_2(\epsilon) = \mathbb{E}_{\vec{h^s}}\big[\log\big(\mathbb{E}_{SAD}[\alpha]\big)\big]
$$
$$
= T_2(0) + \epsilon\frac{\partial T_2}{\partial\epsilon}\bigg|_{\epsilon=0} + \frac{\epsilon^2}{2}\frac{\partial^2 T_2}{\partial\epsilon^2}\bigg|_{\epsilon=0} + \frac{\epsilon^3}{6}\frac{\partial^3 T_2}{\partial\epsilon^3}\bigg|_{\epsilon=0}
$$
$$
+ O(\epsilon^4). \quad \text{(B33)}
$$

The zeroth-order term $T_2(0)$ vanishes due to the log. Let's compute the derivatives one by one, starting with the first derivative:

$$
\frac{\partial T_2}{\partial\epsilon} = \mathbb{E}_{\vec{h^s}}\left[\frac{\mathbb{E}_{SAD}[s\alpha]}{\mathbb{E}_{SAD}[\alpha]}\right]. \quad \text{(B34)}
$$

When evaluated at $\epsilon = 0$, this derivative vanishes for zero-mean stimulus distributions:

$$
\frac{\partial T_2}{\partial\epsilon}\bigg|_{\epsilon=0} = \mathbb{E}_{\vec{h^s}}\big[\mathbb{E}_{SAD}[s]\big] = 0, \quad \text{(B35)}
$$

where, in the second step, we swap the order in which we take the averages, and recall Eq. (B24). Moving on to the second derivative,

$$
\frac{\partial^2 T_2}{\partial\epsilon^2} = \mathbb{E}_{\vec{h^s}}\left[\frac{\mathbb{E}_{SAD}[\alpha]\mathbb{E}_{SAD}[s^2\alpha] - \mathbb{E}_{SAD}^2[s\alpha]}{\mathbb{E}_{SAD}^2[\alpha]}\right]. \quad \text{(B36)}
$$

When evaluated at $\epsilon = 0$, we find that

$$
\frac{\partial^2 T_2}{\partial\epsilon^2}\bigg|_{\epsilon=0} = \mathbb{E}_{\vec{h^s}}\big[\mathbb{E}_{SAD}[s^2] - \big(\mathbb{E}_{SAD}[s]\big)^2\big]. \quad \text{(B37)}
$$

Finally, we require the third derivative,

$$
\frac{\partial^3 T_2}{\partial \epsilon^3} = \mathbb{E}_{\vec{h}^s} \left[\frac{-2\mathbb{E}_{SAD}[\alpha]\mathbb{E}_{SAD}[s\alpha](\mathbb{E}_{SAD}[\alpha]\mathbb{E}_{SAD}[s^2\alpha] - \mathbb{E}_{SAD}^2[s\alpha])}{\mathbb{E}_{SAD}^4[\alpha]} \right]
$$
$$
+ \mathbb{E}_{\vec{h}^s} \left[\frac{\mathbb{E}_{SAD}^2[\alpha](-\mathbb{E}_{SAD}[s\alpha]\mathbb{E}_{SAD}[s^2\alpha] + \mathbb{E}_{SAD}[\alpha]\mathbb{E}_{SAD}[s^3\alpha])}{\mathbb{E}_{SAD}^4[\alpha]} \right], \tag{B38}
$$

which could be simplified, but will be easier to simplify once we evaluate it at $\epsilon = 0$:

$$
\left. \frac{\partial^3 T_2}{\partial \epsilon^3} \right|_{\epsilon=0} = \mathbb{E}_{\vec{h}^s} \left[2\left(\mathbb{E}_{SAD}[s]\right)^3 - 3\mathbb{E}_{SAD}[s]\mathbb{E}_{SAD}[s^2] + \mathbb{E}_{SAD}[s^3] \right]. \tag{B39}
$$

Assembling the pieces [Eqs. (B33), (B35), (B37), and (B39)], we find that

$$
T_2 = \frac{\epsilon^2}{2} \mathbb{E}_{\vec{h}^s} \left[\mathbb{E}_{SAD}[s^2] - \left(\mathbb{E}_{SAD}[s]\right)^2 \right] + \frac{\epsilon^3}{6} \mathbb{E}_{\vec{h}^s} \left[2\left(\mathbb{E}_{SAD}[s]\right)^3 - 3\mathbb{E}_{SAD}[s]\mathbb{E}_{SAD}[s^2] + \mathbb{E}_{SAD}[s^3] \right] + O(\epsilon^4). \tag{B40}
$$

e. Assembling to terms in the MI

We can now take the individual terms we calculated [Eqs. (B32) and (B40) for terms T_1 and T_2], and assemble them to find the MI [Eq. (B25)]

$$
I(\vec{\sigma}; \vec{h}^s) = \epsilon^2 \left(\mathbb{E}_{SAD,\vec{h}^s}[s^2] - E_{\vec{h}^s}[(E_{SAD}[s])^2] \right)
$$
$$
+ \frac{\epsilon^3}{2} \left(\mathbb{E}_{SAD,\vec{h}^s}[s^3] - \mathbb{E}_{SAD,\vec{h}^s}\left[s\mathbb{E}_{SAD}[s^2]\right] - 2\mathbb{E}_{SAD,\vec{h}^s}\left[s^2\mathbb{E}_{SAD}[s]\right] + 2\mathbb{E}_{SAD,\vec{h}^s}\left[s\left(\mathbb{E}_{SAD}[s]\right)^2\right] \right)
$$
$$
- \frac{\epsilon^2}{2} \mathbb{E}_{\vec{h}^s} \left[\mathbb{E}_{SAD}[s^2] - \left(\mathbb{E}_{SAD}[s]\right)^2 \right] - \frac{\epsilon^3}{6} \mathbb{E}_{\vec{h}^s} \left[2\left(\mathbb{E}_{SAD}[s]\right)^3 - 3\mathbb{E}_{SAD}[s]\mathbb{E}_{SAD}[s^2] + \mathbb{E}_{SAD}[s^3] \right] + O(\epsilon^4). \tag{B41}
$$

Grouping the ϵ^2 and ϵ^3 terms together,

$$
I(\vec{\sigma}; \vec{h}^s) = \frac{\epsilon^2}{2} \left(\mathbb{E}_{SAD,\vec{h}^s}[s^2] - E_{\vec{h}^s}[(E_{SAD}[s])^2] \right)
$$
$$
+ \frac{\epsilon^3}{3} \left(\mathbb{E}_{SAD,\vec{h}^s}[s^3] - \mathbb{E}_{\vec{h}^s}\left[\left(\mathbb{E}_{SAD}[s]\right)^3\right] \right)
$$
$$
+ \epsilon^3 \left(\mathbb{E}_{\vec{h}^s}\left[\left(\mathbb{E}_{SAD}[s]\right)^3\right] \right.
$$
$$
\left. - \mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}[s^2]\mathbb{E}_{SAD}[s]\right] \right) + O(\epsilon^4). \tag{B42}
$$

f. Computing the relevant moments of the SAD and stimulus distributions

To proceed, we need to compute the expectation values in Eq. (B42). Let's start with the ϵ^2 terms, and notice that

$$
s^2 = \sum_{ij} \sigma_i \sigma_j h_i h_j
$$
$$
= \sum_i \sigma_i h_i^2 + \sum_{i \neq j} \sigma_i \sigma_j h_i h_j, \tag{B43}
$$

where we have used the fact that, for $\sigma_i \in \{0,1\}$, $\sigma_i^2 = \sigma_i$. Averaging these over the SAD, with $\mathbb{E}_{SAD}[\sigma_i] = \mu_i$, $\mathbb{E}_{SAD}[\sigma_i \sigma_j] = \pi_{ij}$ (the π is for "pair"), we see that

$$
\mathbb{E}_{SAD}[s^2] = \sum_i \mu_i h_i^2 + \sum_{i \neq j} \pi_{ij} h_i h_j. \tag{B44}
$$

Averaging this quantity over the stimulus distribution, recalling that the stimuli are zero mean, and denoting the elements of the covariance matrix by $v_{ij} = \mathbb{E}_{\vec{h}^s}[h_i h_j]$ (we can't use

σ_{ij} to denote covariance since that already describes the spin states),

$$
\mathbb{E}_{SAD,\vec{h}^s}[s^2] = \sum_i \mu_i v_{ii} + \sum_{i \neq j} \pi_{ij} v_{ij}. \tag{B45}
$$

Similarly, we can note that $E_{\vec{h}^s}[(E_{SAD}[s])^2] = E_{\vec{h}^s}[\sum_i \mu_i^2 h_i^2 + \sum_{ij} \mu_i \mu_j h_i h_j]$. Carrying out the average over the stim distribution, we find that

$$
E_{\vec{h}^s}[(E_{SAD}[s])^2] = \sum_i \mu_i^2 v_{ii} + \sum_{i \neq j} \mu_i \mu_j v_{ij}. \tag{B46}
$$

Then, the ϵ^2 term in Eq. (B42) is

$$
\frac{\epsilon^2}{2} \left[\sum_i \mu_i v_{ii} + \sum_{i \neq j} \pi_{ij} v_{ij} - \sum_i \mu_i^2 v_{ii} - \sum_{i \neq j} \mu_i \mu_j v_{ij} \right]
$$
$$
= \frac{\epsilon^2}{2} \left[\sum_i v_{ii} \mu_i (1 - \mu_i) + \sum_{i \neq j} v_{ij} (\pi_{ij} - \mu_i \mu_j) \right]. \tag{B47}
$$

The first of these terms contains $\mu_i(1 - \mu_i) = \mathrm{var}(\sigma_i)$, and the second one contains $(\pi_{ij} - \mu_i \mu_j) = \mathrm{cov}(\sigma_i, \sigma_j)$, where these moments (recall) are computed over the spontaneous activity distribution, and thus $\mathrm{cov}(\sigma_i, \sigma_j)$ is best related to the J_{ij} terms in Eq. (B2), and $\mathrm{var}(\sigma_i)$ is best related to the mean-determining bias term J_{ii} in Eq. (B2). If we ignore the higher-order terms in the MI [the third moments come in at $O(\epsilon^3)$], and restrict ourselves momentarily to $O(\epsilon^2)$, we notice

that

$$I(\vec{\sigma};\vec{h}^s) = \frac{\epsilon^2}{2}\left[\sum_i v_{ii}\operatorname{var}(\sigma_i) + \sum_{i \neq j} v_{ij}\operatorname{cov}(\sigma_i,\sigma_i)\right]$$

$$+ O(\epsilon^3). \tag{B48}$$

We now make our first observation, namely, that for maximum MI, we require that v_{ij} and $\operatorname{cov}(\sigma_i,\sigma_i)$ should have the same sign (both positive or both negative). Since the correlations in the SAD are controlled by (and have the same

sign as) J_{ij}, we find that J_{ij} should have the same sign as v_{ij} (the stimulus covariances), which agrees with our numerical experiments, and those of Tkacik *et al.* We also notice that, to maximize $\operatorname{cov}(\sigma_i,\sigma_i)$ and $\operatorname{var}(\sigma_i)$, one should choose $\mu_i = 1/2$, which again agrees with our numerical experiments with unskewed stimuli. If there are skewed stimuli, and thus the third-order terms are to be considered, these conclusions may no longer hold, so to make more progress, we must consider the ϵ^3 terms in Eq. (B42).

As with the ϵ^2 terms, we will consider the ϵ^3 terms one by one, starting with

$$\mathbb{E}_{SAD,\vec{h}^s}[s^3] = \mathbb{E}_{SAD,\vec{h}^s}\left[\sum_{ijk}\sigma_i\sigma_j\sigma_k h_i h_j h_k\right]$$

$$= \mathbb{E}_{\vec{h}^s}\left[\sum_i \mu_i h_i^3 + 3\sum_{i \neq j}\pi_{ij}h_i^2 h_j + \sum_{i \neq j \neq k}\tau_{ijk}h_i h_j h_k\right], \tag{B49}$$

where we have used the fact that, for $\sigma_i \in \{0,1\}$, $\sigma_i^3 = \sigma_i^2 = \sigma_i$, and defined $\tau_{ijk} = \mathbb{E}_{SAD}[\sigma_i\sigma_j\sigma_k]$ (the τ is for "triplet") in going from the first line to the second. Carrying out the average over the stimulus distribution, we observe that

$$\mathbb{E}_{SAD,\vec{h}^s}[s^3] = \sum_i \mu_i\langle h_i^3\rangle + 3\sum_{i \neq j}\pi_{ij}\langle h_i^2 h_j\rangle + \sum_{i \neq j \neq k}\tau_{ijk}\langle h_i h_j h_k\rangle, \tag{B50}$$

where we have used triangle brackets $\langle\cdot\rangle$ to denote an expectation value over the stimulus distribution. Moving on to the next ϵ^3 term in Eq. (B42),

$$\mathbb{E}_{\vec{h}^s}\left[\left(\mathbb{E}_{SAD}[s]\right)^3\right] = \mathbb{E}_{\vec{h}^s}\left[\sum_{ijk}\mu_i\mu_j\mu_k h_i h_j h_k\right] \tag{B51}$$

$$= \mathbb{E}_{\vec{h}^s}\left[\sum_i \mu_i^3 h_i^3 + 3\sum_{i \neq j}\mu_i^2\mu_j h_i^2 h_j + \sum_{i \neq j \neq k}\mu_i\mu_j\mu_k h_i h_j h_k\right]$$

$$= \sum_i \mu_i^3\langle h_i^3\rangle + 3\sum_{i \neq j}\mu_i^2\mu_j\langle h_i^2 h_j\rangle + \sum_{i \neq j \neq k}\mu_i\mu_j\mu_k\langle h_i h_j h_k\rangle.$$

We can assemble these two terms, to get the $\frac{\epsilon^3}{3}$ term in Eq. (B42):

$$\frac{\epsilon^3}{3}\left(\sum_i \mu_i\left(1 - \mu_i^2\right)\langle h_i^3\rangle + 3\sum_{i \neq j}\left(\pi_{ij} - \mu_i^2\mu_j\right)\langle h_i^2 h_j\rangle + \sum_{i \neq j \neq k}(\tau_{ijk} - \mu_i\mu_j\mu_k)\langle h_i h_j h_k\rangle\right). \tag{B52}$$

What remains is to compute the ϵ^3 term (with no factor of 1/3), and we'll again do that term by term, noting that the first part $(\mathbb{E}_{\vec{h}^s}[(\mathbb{E}_{SAD}[s])^3])$ is already known [Eq. (B51)], leaving us to compute

$$\mathbb{E}_{\vec{h}^s}\left[\mathbb{E}_{SAD}[s^2]\mathbb{E}_{SAD}[s]\right] = \mathbb{E}_{\vec{h}^s}\left[\left(\sum_i \mu_i h_i^2 + \sum_{i \neq j}\pi_{ij}h_i h_j\right)\sum_k \mu_k h_k\right]$$

$$= \mathbb{E}_{\vec{h}^s}\left[\sum_i \mu_i^2 h_i^3 + \sum_{i \neq j}\mu_i\mu_j h_i^2 h_j + 2\sum_{i \neq j}\mu_i\pi_{ij}h_i^2 h_j + \sum_{i \neq j \neq k}\mu_k\pi_{ij}h_i h_j h_k\right]$$

$$= \sum_i \mu_i^2\langle h_i^3\rangle + \sum_{i \neq j}\mu_i\mu_j\langle h_i^2 h_j\rangle + 2\sum_{i \neq j}\mu_i\pi_{ij}\langle h_i^2 h_j\rangle + \sum_{i \neq j \neq k}\mu_k\pi_{ij}\langle h_i h_j h_k\rangle, \tag{B53}$$

where we have used Eq. (B44) for $\mathbb{E}_{SAD}[s^2]$, and the factor of 2 on the second line comes in because k could be either j or i. We can assemble this with Eq. (B51) to get the ϵ^3 term (with no factor of $1/3$) in our MI formula:

$$\epsilon^3 \left(\sum_i \mu_i^2 (\mu_i - 1) \langle h_i^3 \rangle + \sum_{i \neq j} \mu_i (3 \mu_i \mu_j - \mu_j - 2\pi_{ij}) \langle h_i^2 h_j \rangle + \sum_{i \neq j \neq k} \mu_k (\mu_i \mu_j - \pi_{ij}) \langle h_i h_j h_k \rangle \right)$$

$$= \epsilon^3 \left(-\sum_i \mu_i \mathrm{var}(\sigma_i) \langle h_i^3 \rangle + \sum_{i \neq j} \mu_i (\mu_i \mu_j - \mu_j - 2\, \mathrm{cov}(\sigma_i, \sigma_j)) \langle h_i^2 h_j \rangle - \sum_{i \neq j \neq k} \mu_k \mathrm{cov}(\sigma_i, \sigma_j) \langle h_i h_j h_k \rangle \right).$$

Assembling the pieces [above and Eqs. (B42), (B48), and (B52)], we observe that

$$I(\vec{\sigma}; \vec{h}^s) = \frac{\epsilon^2}{2} \left[\sum_i v_{ii} \mathrm{var}(\sigma_i) + \sum_{i \neq j} v_{ij} \mathrm{cov}(\sigma_i, \sigma_i) \right]$$

$$+ \frac{\epsilon^3}{3} \left(\sum_i \mu_i (1 - \mu_i^2) \langle h_i^3 \rangle + 3 \sum_{i \neq j} (\pi_{ij} - \mu_i^2 \mu_j) \langle h_i^2 h_j \rangle + \sum_{i \neq j \neq k} (\tau_{ijk} - \mu_i \mu_j \mu_k) \langle h_i h_j h_k \rangle \right)$$

$$+ \epsilon^3 \left(-\sum_i \mu_i \mathrm{var}(\sigma_i) \langle h_i^3 \rangle + \sum_{i \neq j} \mu_i (\mu_i \mu_j - \mu_j - 2\, \mathrm{cov}(\sigma_i, \sigma_j)) \langle h_i^2 h_j \rangle - \sum_{i \neq j \neq k} \mu_k \mathrm{cov}(\sigma_i, \sigma_j) \langle h_i h_j h_k \rangle \right) + O(\epsilon^4).$$

We can simplify this a bit further by grouping together the ϵ^3 terms to yield

$$I(\vec{\sigma}; \vec{h}^s) = \frac{\epsilon^2}{2} \left[\sum_i v_{ii} \mathrm{var}(\sigma_i) + \sum_{i \neq j} v_{ij} \mathrm{cov}(\sigma_i, \sigma_i) \right] + \epsilon^3 \left(\sum_i \frac{\mu_i}{3} (1 + 2\mu_i^2 - 3\mu_i) \langle h_i^3 \rangle + \sum_{i \neq j} \mathrm{cov}(\sigma_i, \sigma_j)(1 - 2\mu_i) \langle h_i^2 h_j \rangle \right)$$

$$+ \epsilon^3 \left(\sum_{i \neq j \neq k} \frac{1}{3} (\tau_{ijk} + 2\mu_i \mu_j \mu_k - 3\mu_k \pi_{ij}) \langle h_i h_j h_k \rangle \right) + O(\epsilon^4). \tag{B54}$$

Recall that $\mu_i = \mathbb{E}_{SAD}(\sigma_i)$, $\pi_{ij} = \mathbb{E}_{SAD}(\sigma_i \sigma_j)$, $\tau_{ijk} = \mathbb{E}_{SAD}(\sigma_i \sigma_j \sigma_k)$, $v_{ij} = \mathbb{E}_{\vec{h}^s}(h_i h_j)$, angled brackets denote averages over the stimulus ensemble, and $\mathrm{cov}(\sigma_i, \sigma_i)$ and $\mathrm{var}(\sigma_i)$ are moments of the spontaneous activity distribution. Cleaning up our notation to match the main paper, we thus find that

$$I(\vec{\sigma}; \vec{h}^s) = \frac{\epsilon^2}{2} \left[\sum_i \langle (h_i^s)^2 \rangle \mathrm{var}(\sigma_i) + \sum_{i \neq j} \langle h_i^s h_j^s \rangle \mathrm{cov}(\sigma_i, \sigma_i) \right] + \epsilon^3 \left[\sum_i \frac{\mu_i}{3} (1 + 2\mu_i^2 - 3\mu_i) \langle h_i^3 \rangle + \sum_{i \neq j} \mathrm{cov}(\sigma_i, \sigma_j)(1 - 2\mu_i) \langle h_i^2 h_j \rangle \right]$$

$$+ \epsilon^3 \left[\sum_{i \neq j \neq k} \frac{1}{3} (\tau_{ijk} + 2\mu_i \mu_j \mu_k - 3\mu_k \pi_{ij}) \langle h_i h_j h_k \rangle \right] + O(\epsilon^4). \tag{B55}$$

APPENDIX C: NUMERICAL METHODS

1. Monte Carlo methods and optimization

The mutual information between the stimuli and responses,

$$I(\vec{\sigma}; \vec{h}^s) = -\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}) \log[p(\vec{\sigma})]$$

$$+ \int d\vec{h}^s p(\vec{h}^s) \sum_{\{\vec{\sigma}\}} p(\vec{\sigma} | \vec{h}^s) \log[p(\vec{\sigma} | \vec{h}^s)], \quad \text{(C1)}$$

involves the sum over all 2^N possible population states, and an integral over the stimulus distribution of another such sum. This function is not analytically tractable for $N = 10$ (the network size considered in this work) and/or for continuous stimulus distributions. Instead, we use Monte Carlo methods to compute the MI. In particular, we define the (un-normalized)

frequency function

$$\phi(\vec{\sigma} | \vec{h}^s) = \exp \left[\beta \left(\vec{h}^s \cdot \vec{\sigma} + h^0 \sum_i \sigma_i \right. \right.$$

$$\left. \left. + J \sum_{i < j} \sigma_i \sigma_j + \gamma \sum_{i < j < k} \sigma_i \sigma_j \sigma_k \right) \right]. \quad \text{(C2)}$$

This (log-polynomial) function can be very quickly evaluated, and to compute the MI, we take a large number of stimuli $\vec{h}^s$ from the appropriate distribution, and evaluate the frequencies of each of the 2^N states for each of the stimuli. We then divide the frequencies for each state and stimulus by the sum of the frequencies over all states for that stimulus, to get (normalized)

conditional probabilities:

$$p(\vec{\sigma}|\vec{h}^s) = \phi(\vec{\sigma}|\vec{h}^s) \Big/ \sum_{\{\vec{\sigma}\}} \phi(\vec{\sigma}|\vec{h}^s). \qquad \text{(C3)}$$

This normalizing operation can be done quickly using matrix operations in MATLAB [60]. Note that, if one instead defined the conditional probability for each state (instead of frequencies), then one would need to evaluate the partition function (costly) in the calculation of the probability of each of the 2^N states. Using the approach of first computing frequencies, we evaluate the partition function only once for each stimulus value, saving $2^N - 1$ evaluations of the partition function for each stimulus example. Given the conditional probabilities, we then compute the conditional entropy (for each stimulus),

$$H(\vec{\sigma}|\vec{h}^s) = -\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}|\vec{h}^s) \log[p(\vec{\sigma}|\vec{h}^s)]. \qquad \text{(C4)}$$

Averaging these values over the set of stimuli from our distribution we get the noise entropy [H_{noise}, which is minus the second term in Eq. (C1)]. Similarly, we can average the conditional probabilities across all stimulus examples to get the (marginal) response distribution

$$p(\vec{\sigma}) = \langle p(\vec{\sigma}|\vec{h}^s) \rangle_{\vec{h}^s}. \qquad \text{(C5)}$$

Finally, we compute the entropy of the response distribution

$$H_{\text{resp}} = -\sum_{\{\vec{\sigma}\}} p(\vec{\sigma}) \log[p(\vec{\sigma})] \qquad \text{(C6)}$$

and subtract the noise entropy to get the MI: $I(\vec{\sigma};\vec{h}^s) = H_{\text{resp}} - H_{\text{noise}}$.

Note that, since we are using Monte Carlo integration, each evaluation of the MI function involves a (potentially) different set of stimuli, and thus a potentially (slightly) different result, even for identical network parameters. This noise makes gradient-based optimization methods highly error prone. We avoid this pitfall by using exactly the same set of stimuli in subsequent calls to the MI function during the optimization. This *common random number* approach makes the MI a smooth function of our parameters, allowing us to use gradient-based optimization techniques; see [61] for an overview of optimization methods for noisy functions. For the optimization itself, we use the open-source MINFUNC package [62] from Mark Schmidt. We found that MINFUNC was much faster and more reliable than the minimizers in the MATLAB optimization toolbox.

In this paper, we have used ensembles of 1000 stimulus examples in evaluating the MI function. We repeated the optimization five times, with different sets of stimuli each time, and found that the results were highly reproducible: the standard deviation of the mean MI achieved over those five trials is small [Figs. 3(a) and 3(b) of the main paper]—it is comparable to, or in many cases less than, the linewidth on the plots—as is the standard deviation of the mean parameter values obtained at optimality [Figs. 4(a) and 4(b) of the main paper].

The expressions herein (and in the main paper) do not specify the base in which the logarithm is computed. For MI values in bits, those logarithms are to base 2.

2. Comparing optimal networks with constrained firing rates

When we use Lagrange multipliers for optimizing MI with constrained firing rates (see main paper), the exact functional relationship between Lagrange multiplier λ and firing rate is unknown: although higher Lagrange multipliers lead to lower firing rates, we cannot easily specify what value of λ is needed to achieve a given firing rate. We use the same values of the Lagrange multipliers when we optimize with triplet interactions either allowed (TA), or forbidden (TF), resulting in (slightly) different mean firing rates for the optimal TA and TF networks. The reason for this difference is easy to understand, as they have different MI values, and thus the optimal trade-off between MI and firing rate in the Lagrange function $\mathcal{L} = MI - \lambda \langle \sigma \rangle$ will be slightly different.

We use linear interpolation to estimate the MI of the TF network at the *exact mean firing rate of the TA network*: since we have several points on the curve of MI versus mean firing rate for the TF network, this interpolation is easy to implement. Finally, we take the ratio of the MI value for the TA network to the (interpolated) one for the TF network at the same firing rate to create the data in Figs. 3(e) and 3(f).

APPENDIX D: TRIPLET INTERACTIONS ARE BETTER THAN PAIRWISE ONES AT SUPPRESSING MULTISPIKE STATES; HENCE THE OBSERVATION THAT $\gamma < 0$ AND $J > 0$ AT OPTIMALITY, AND NOT VICE VERSA

Consider the contribution C of the recurrent connections to the log-polynomial probability distribution over network states, $C = J \sum_{i<j} \sigma_i \sigma_j + \gamma \sum_{i<j<k} \sigma_i \sigma_j \sigma_k$. When this number is large, the state is favored, and vice versa. The first term ($J \sum_{i<j} \sigma_i \sigma_j$) is J times the number of active neural pairs, which is $O(\alpha^2)$, where α is the number of coactive neurons, while the second term ($\gamma \sum_{i<j<k} \sigma_i \sigma_j \sigma_k$) is $O(\alpha^3)$.

Let us consider situations in which it is desirable for neurons to act cooperatively, while not always firing synchronously.

If we choose positive J and (small) negative γ—which is what we observe at optimality: see Fig. 4(b) of the main
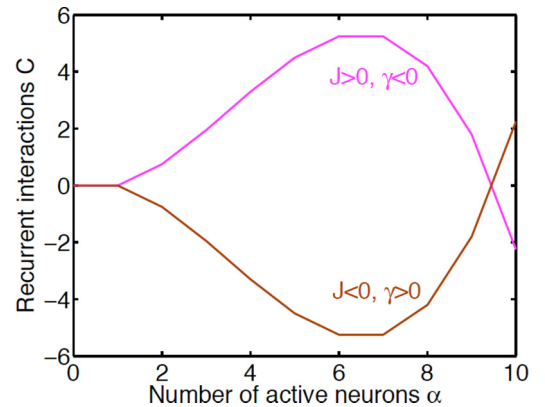

FIG. 5. (Color online) Negative triplet interactions are better at suppressing the all-on state than are negative pairwise interactions. The plots here show the function $C = J\alpha(\alpha - 1)/2 + \gamma\alpha(\alpha - 1)(\alpha - 2)/6$ for non-negative integer values of $\alpha \in [0,10]$. For $J = 0.75$ and $\gamma = -0.3$ (upper, pink curve), the states with many coactive neurons are suppressed by the recurrent interaction. For $J = -0.75$ and $\gamma = 0.3$ (lower, brown curve), the opposite is true.

paper—then neurons are encouraged to be coactive by the positive J: they cooperate. If one considers states with many spikes (large α), however, we can see that the effects of the triplet interaction, which are $O(\alpha^3)$, can exceed the pairwise ones, which are $O(\alpha^2)$. In other words, $C \sim J O(\alpha^2) + \gamma O(\alpha^3)$ is a unimodal function with a positive peak, such that for large α, states are strongly suppressed, while for intermediate α, they may be facilitated [Fig. 5: upper (pink) curve]. This acts somewhat like negative feedback: for small α, the effects of J (larger in magnitude than γ) dominate, pushing the network towards having coactive cells, while for large α, the effects of γ push the network away from having too many coactive cells.

Thus the network produces cooperative responses but has a diminished probability of having all of the neurons coactive.

Now consider the opposite situation, with negative J, and positive γ. In this case C is unimodal with a negative peak, and the larger-α states are progressively more facilitated by recurrent interactions [Fig. 5: lower (brown) curve]. This is reminiscent of positive feedback, and leads to heavy usage of the all-neurons-on state.

Of course, if we allow fourth-order terms in the probability model, then one could have positive γ, while still avoiding epileptic levels of synchrony, by having negative fourth-order interactions, for example.

[1] I. H. Stevenson and K. P. Kording, Nat. Neurosci. **14**, 139 (2011).

[2] E. Schneidman, M. J. Berry, R. Segev, and W. Bialek, Nature (London) **440**, 1007 (2006).

[3] J. Shlens *et al.*, J. Neurosci. **26**, 8254 (2006); J. Shlens *et al.*, *ibid.* **29**, 5022 (2009).

[4] E. Granot-Atedgi, G. Tkačik, R. Segev, and E. Schneidman, PLoS Comput. Biol **9**, e1002922 (2013).

[5] E. Ganmor, R. Segev, and E. Schneidman, Proc. Natl. Acad. Sci. U.S.A. **108**, 9679 (2011).

[6] G. Tkacik, O. Marre, D. Amodei, E. Schneidman, W. Bialek, and M. J. Berry II, PLoS Comput. Biol. **10**, e1003408 (2014).

[7] F. Montani *et al.*, Philos. Trans. R. Soc. A **367**, 3297 (2009).

[8] S. Yu, H. Yang, H. Nakahara, G. Santos, D. Nikolić, and D. Plenz, J. Neurosci. **31**, 17514 (2011).

[9] I. E. Ohiorhenuan, F. Mechler, K. P. Purpura, A. M. Schmid, and J. D. Victor, Nature (London) **466**, 617 (2010).

[10] B. Averbeck, P. E. Latham, and A. Pouget, Nat. Rev. Neurosci. **7**, 358 (2006).

[11] J. Cafaro and F. Rieke, Nature (London) **468**, 964 (2010).

[12] P. E. Latham and Y. Roudi, in *Principles of Neural Coding*, edited by S. Panzeri and R. Quian Quiroga (CRC Press, Boca Raton, RF, 2013), pp. 121–138.

[13] E. Zohary, M. N. Shadlen, and W. T. Newsome, Nature (London) **370**, 140 (1994).

[14] G. Tkačik, J. S. Prentice, V. Balasubramanian, and E. Schneidman, Proc. Natl. Acad. Sci. U.S.A. **107**, 14419 (2010).

[15] J. Beck, V. R. Bejjanki, and A. Pouget, Neural Comput. **23**, 1484 (2011).

[16] A. S. Ecker, P. Behrens, A. S. Tolias, and M. Bethge, J. Neurosci. **31**, 14272 (2011).

[17] M. Shamir and H. Sompolinsky, Neural Comput. **16**, 1105 (2004).

[18] K. Josic, E. Shea-Brown, J. de la Rocha, and B. Doiron, Neural Comput. **21**, 2774 (2009).

[19] Y. Hu, J. Zylberberg, and E. Shea-Brown, PLoS Comput. Biol. **10**, e1003469 (2014).

[20] N. A. Cayco-Gajic, J. Zylberberg, and E. Shea-Brown, Frontiers Comput. Neurosci. **9**, 57 (2015).

[21] R. Salakhutdinov and G. E. Hinton, *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, Vol. 5 of JMLR (Clearwater Beach, Florida, 2009), pp. 448–455.

[22] G. E. Hinton and T. J. Sejnowski, *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (IEEE, New York, 1983), p. 448.

[23] Our β parameter is effectively equivalent to double that of Tkačik *et al.* [14]: we use neurons with $\sigma_i \in \{0,1\}$, and they use ones with $\sigma_i \in \{-1,1\}$. The conversion is $\sigma^{\{-1,1\}} = 2(\sigma^{\{0,1\}} - 1/2)$.

[24] E. Schneidman, S. Still, M. J. Berry, and W. Bialek, Phys. Rev. Lett. **91**, 238701 (2003).

[25] E. T. Jaynes, Phys. Rev. **106**, 620 (1957).

[26] S.-I. Amari, IEEE Trans. Inf. Theory **47**, 1701 (2001).

[27] W. Maass, in *Pulsed Neural Networks*, edited by W. Maass and C. M. Bishop (MIT Press, Cambridge, MA, 1999), pp. 55–85.

[28] D. R. Brillinger and J. P. Segundo, Biol. Cybern. **35**, 213 (1979).

[29] N. Hô and A. Destexhe, J. Neurophysiol. **84**, 1488 (2000).

[30] A. K. Barreiro, J. Gjorjieva, F. Rieke, and E. Shea-Brown, Frontiers Comput. Neurosci. **8**, 10 (2014).

[31] J. H. Macke, M. Opper, and M. Bethge, Phys. Rev. Lett. **106**, 208102 (2011).

[32] D. Leen and E. Shea-Brown, J. Math. Neurosci. **5**, 17 (2015).

[33] We use this rewrite so that we can expand the MI in powers of the stimulus-coupling strength. This expansion involves taking derivatives of the MI with respect to the stimulus-coupling parameter. In the case where the same coupling parameter multiplies the stimulus strength and the interaction parameters, this expansion becomes much more involved.

[34] M. R. DeWeese and A. M. Zador, J. Neurosci. **26**, 12206 (2006).

[35] We note that, if the photoreceptors perform a logarithmic transformation on the luminance values that they sample from the environment, then the downstream inputs (say, to retinal ganglion cells) may have much less skewed distributions than do the raw luminance values. In that case, our arguments about skewed inputs may not apply directly to retinal ganglion cells. At the same time, if there are firing rate constraints on retinal ganglion cells, we expect that beyond-pairwise correlations could enhance their information coding ability regardless of the skewness of their inputs [Fig. 3(e)].

[36] G. Tkačik *et al.*, PLoS One **6**, e20409 (2011).

[37] H. B. Barlow, in *Sensory Communication*, edited by W. A. Rosenblith (MIT Press, Cambridge, MA, 1961), pp. 217–234.

[38] E. P. Simoncelli and B. A. Olshausen, Annu. Rev. Neurosci. **24**, 1193 (2001).

[39] D. L. Ruderman and W. Bialek, Phys. Rev. Lett. **73**, 814 (1994).

[40] G. J. Stephens, T. Mora, G. Tkacik, and W. Bialek, Phys. Rev. Lett. **110**, 018701 (2013).

[41] D. Ruderman, Vision Res. **37**, 3385 (1997).

[42] W. H. Hsiao and R. P. Milane, J. Opt. Soc. Am. A **22**, 1789 (2005).

[43] J. Zylberberg, D. Pfau, and M. R. DeWeese, Phys. Rev. E **86**, 066112 (2012).

[44] T. Hromádka, M. R. DeWeese, and A. M. Zador, PLoS Biol. **6**, e16 (2008).

[45] R. Baddeley *et al.*, Proc. R. Soc. London B **264**, 1775 (1997).

[46] Y. Karklin and E. P. Simoncelli, in *Advances in Neural Information Processing Systems*, edited by J. Shawe-Taylor *et al.* (MIT Press, Cambridge, MA, 2011), Vol. 24.

[47] J. Zylberberg, J. T. Murphy, and M. R. DeWeese, PLoS Comput. Biol. **7**, e1002250 (2011).

[48] B. A. Olshausen and D. J. Field, Nature (London) **381**, 607 (1996).

[49] P. D. King, J. Zylberberg, and M. R. DeWeese, J. Neurosci. **33**, 5475 (2013).

[50] J. Zylberberg and M. R. DeWeese, PLoS Comput. Biol. **9**, e1003182 (2013).

[51] S.-I. Amari, H. Nakahara, S. Wu, and Y. Sakai, Neural Comput. **15**, 127 (2003).

[52] We note that, for a given set of moments, the pairwise maximum entropy model (with $\gamma = 0$) maximizes the entropy of the responses. Thus it may seem puzzling to think of the inclusion

of $\gamma \neq 0$ increasing response entropy. In this situation here, we are not holding the response statistics fixed. The optimization procedure covaries J and γ to maximize the MI, but the response statistics (correlations, etc.) can vary. Consequently, it is possible for optimal $\gamma \neq 0$ models (triplet allowed) to have larger response entropy than optimal $\gamma = 0$ (triplet forbidden) ones.

[53] P. Dayan and L. F. Abbott, *Theoretical Neuroscience* (MIT Press, Cambridge, MA, 2001).

[54] A. T. Gulledge, B. M. Kampa, and G. J. Stuart, J. Neurobiol. **64**, 75 (2005).

[55] B. W. Mel, J. Neurophysiol. **70**, 1086 (1993).

[56] M. T. Harnett, J. K. Makara, N. Spruston, W. L. Kath, and J. C. Magee, Nature (London) **491**, 599 (2012).

[57] M. London and M. Häusser, Annu. Rev. Neurosci. **28**, 503 (2005).

[58] A. Polsky, B. W. Mel, and J. Schiller, Nat. Neurosci. **7**, 621 (2004).

[59] U. Koster, J. Sohl-Dickstein, C. M. Gray, and B. A. Olshausen, PLoS Comput. Biol. **10**, e1003684 (2014).

[60] MATLAB version 2012a, The MathWorks Inc., Natick, MA, 2012.

[61] G. Deng, Ph.D. thesis, University of Wisconsin, Madison, 2007.

[62] M. Schmidt, minFunc: unconstrained differentiable multivariate optimization in Matlab, 2005, http://www.cs.ubc.ca/schmidtm/Software/minFunc.html.