EVOKED PATTERNS OF OSCILLATORY ACTIVITY IN MEAN-FIELD NEURONAL NETWORKS

by

Jason E. Pina

B.S. in Mathematics, B.S. in Physics

George Mason University, 2012

Submitted to the Graduate Faculty of

the Kenneth P. Dietrich School of Arts and Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2018

UNIVERSITY OF PITTSBURGH

KENNETH P. DIETRICH SCHOOL OF ARTS AND SCIENCES

This dissertation was presented

by

Jason E. Pina

It was defended on

November 30, 2018

and approved by

G. Bard Ermentrout, Distinguished University Professor, Mathematics

Brent Doiron, Professor, Mathematics

Anne-Marie Oswald, Associate Professor of Neuroscience

Jonathan Rubin, Chair and Professor, Mathematics

Dissertation Director: G. Bard Ermentrout, Distinguished University Professor, Mathematics

EVOKED PATTERNS OF OSCILLATORY ACTIVITY IN MEAN-FIELD NEURONAL NETWORKS

Jason E. Pina, PhD

University of Pittsburgh, 2018

Oscillatory behaviors in populations of neurons are oberved in diverse contexts. In tasks involving working memory, a form of short-term memory, oscillations in different frequency bands have been shown to increase across varying spatial scales using recording methods such as EEG (electroencephalogram) and MEG (magnetoencephalogram). Such oscillatory activity has also been observed in the context of neural binding, where different features of objects that are perceived or recalled are associated with one another. These sets of data suggest that oscillatory dynamics may also play a key role in the maintenance and manipulation of items in working memory.

Using similar recording techniques, including EEG and MEG, oscillatory neuronal activity has also been seen to occur when certain images that cause aversion and headaches in healthy human subjects or seizures in those with pattern-sensitive epilepsy are presented. The images most likely to cause such responses are those with dominant spatial frequencies near 3–5 cycles per degree, the same band of wavenumbers to which normal human vision exhibits the greatest contrast sensitivity.

We model these oscillatory behaviors using mean-field, Wilson-Cowan-type neuronal networks. In the case of working memory and binding, we find that including the activity of certain long-lasting excitatory synapses in addition to the usual inhibitory and shorter-term excitatory synaptic activity allows for bistability between a low steady state and a high oscillatory state. By coupling several such populations together, both in-phase and out-of-phase oscillations arise, corresponding to distinct and bound items in working memory, respectively. We analyze the network's dynamics and dependence on biophysically relevant parameters using a combination of techniques, including numerical bifurcation analysis and weak coupling theory. In the case of spatially resonant responses to static simtuli, we employ Wilson-Cowan networks extended in one and two spatial dimensions. By placing the networks near Turing-Hopf bifurcations, we find they exhibit spatial resonances that compare well with empirical results. Using simulations, numerical bifurcation analysis, and perturbation theory, we characterize the observed dynamics and gain mathematical insight into the mechanisms that lead to these dynamics.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

1.1 MEAN-FIELD MODELS

The mathematical modeling of neuronal data began as early as 1907, when Lapicque modeled data obtained by stimulating the nerve in a frog's leg as a capacitor, thereby presaging the integrate-and-fire model [1, 2, 3]. While much work continued in the same vein of capturing the activity of single neurons, perhaps best highlighted by Hodgkin's and Huxley's seminal biophysical model [4], the computational intractability of using such equations to capture the dynamics of many interacting neurons drove efforts to find simple, averaged equations that could do so.

In 1956, Beurle made perhaps the first foray into such mean-field models, introducing and analyzing a spatially-extended neural field model composed of excitatory neurons [5]. Wilson and Cowan followed up on this work by including inhibitory populations of neurons in non-spatially extended [6] and spatially-extended [7] models, allowing for excitatory-inhibitory dynamics. Using u, e to denote excitatory populations and indices, respectively, and v, i to denote inhibitory populations and indices, respectively, these networks may be written as

$$\begin{aligned} \tau_e \frac{\partial}{\partial t} u(x,t) &= -u + f_e(C_{ee} * u - C_{ei} * v) \\ \tau_i \frac{\partial}{\partial t} v(x,t) &= -v + f_i(C_e * u - C_{ii} * v), \end{aligned} \tag{1.1}$$

where $*$ denotes spatial convolution ($f * g(x,t) = \int_{-\infty}^{\infty} f(x-x',t)\, g(x',t)\, dx'$), and $C_{\beta,\alpha}(x)$, $\alpha,\ \beta \in \{e,\ i\}$, are connectivity functions from population α to population β. We can rewrite these functions to explicitly show the maximum connectivities and the normalized kernels: $C_{\beta\alpha}(x) = a_{\beta\alpha} K_{\beta\alpha}(x)$, where $\int_{-\infty}^{\infty} K_{\beta\alpha}(x')dx' = 1$. For simplicity, we generally suppose the kernels are determined by the presynaptic population, so that $K_{\beta\alpha}(x) = K_\alpha(x)$, and are generally chosen to be monotonically decaying with respect to distance. Two common kernel choices are Gaussian, $K(x) \sim \exp\left(\frac{-|\boldsymbol{x}|^2}{\sigma^2}\right)$, and exponential, $K(x) \sim \exp\left(\frac{-|\boldsymbol{x}|}{\sigma}\right)$. In the case that we are considering isolated populations of neurons (i.e., not spatially extended), we can instead replace the convolutions by weight matrices that encode the connection strengths between populations. If the network has, say, N_e excitatory populations and N_i inhibitory populations, these weight matrices will have N_e^2, N_e^2, or $N_e \cdot N_i$ elements. The activation, firing rate, or transfer functions $f_\alpha(w)$ are generally taken to be sigmoidal, although other nonlinear monotonic functions may be used as well to better fit the data or other spiking neuronal models, as we do in Chap 2.

While work has continued to add further realism to mean-field models [8, 9, 10], Wilson-Cowan-type equations have been profitably employed in the decades since their introduction to model many different experimentally observed neuronal network phenomena, including waves in cortex [11, 12], oscillating activity in seizures [13], working memory dynamics [14][15] and visual hallucinations [16, 17, 18]. Thus, we use Wilson-Cowan-type mean-field equations to capture oscillatory dynamics that may contribute to our understanding of working memory and sensitivities to spatially periodic stimuli, as we describe below.

1.2 OSCILLATIONS IN WORKING MEMORY

Memory serves vital functions in animals. By forming stable internal representations of the external world, animals can recognize, process, and learn from various types of patterns they encounter. While long-term memory comprises a highly stable form of storage that forms on long timescales through synaptic plasticity [19], animals also need a way to form stable short-term representations of objects in order to perform real-time computations so that they might appropriately interact with their environment. These items can either be ones that have been perceived in a local spatiotemporal frame or else ones that they have accessed from long-term memory. For example, a mouse might encounter a scent that activates circuits corresponding to the neural correlates that represent a predator; this representation should then be active for real-time processing so that the mouse might move away from the source of the scent. Perhaps the mouse encounters this scent in a novel environment, and must recall whence she came to determine a safe route by which to retreat. By having a representation of the most recent steps in her course active in short-term memory, she may rapidly choose an appropriate escape path by simply doubling back. Thus, animals need a way to very rapidly form, combine, change, and terminate internal representations of information linked to objects both perceived and stored.

The form of memory that has been found to enable such dynamic representations and real-time computations is known as working memory. By forming short-term instantiations of relevant information and actively processing this information, working memory was hypothesized within experimental cognitive psychology as a multi-component model to explain humans' ability to guide their actions based on combinations of perceived and remembered items [20, 21, 22]. One fundamental limitation of working memory appears to be its capacity: only a finite amount of information, usually quantified as $\approx 3 - 5$ items, can be held active at once. Thus, working memory trades off the accessibility and processing speed of relevant information for the quantity it can maintain.

While the processing of the information necessarily involves complex, difficult-to-trace network processes, the substrate of the active memories need not be so complicated. Thus, much of the research within neuroscience that has involved working memory has focused on how these memories are encoded. In 1971, Fuster and Alexander produced the first seminal work in this vein [23]. Rhesus monkeys performed a delayed response task involving a visual cues. While the specific setup has changed over the years, the basic form of the experiment has stayed the same: Monkeys are briefly provided a visual cue, which is then removed. After a delay period (18s in the original study), the monkey is asked to recall the location of the cue (e.g., by reaching towards the location of the cued object, or simply moving its eyes to point to the remembered location). Fuster and Alexander reported neurons that fired at higher rates during the delay period, but not before or afterwards. Numerous other studies [24, 25, 26, 27] have since confirmed this basic fact: selected networks of neurons increase their rate of firing during the short-term retention of information. How exactly this increased activity persists over several seconds after the stimulus has been removed, however, is still not definitively known. The activation of glutamatergic (excitatory) NMDA (N-methyl-D-aspartate) receptors may be key to the maintenance of working memory items [28]. These receptors allow positively-charged ions to pass through if (1) the proper neurotransmitters (glycine and glutamate) are bound and (2) the membrane is sufficiently depolarized, thus releasing bound Mg^{2+} ions and allowing Na^+, K^+, and Ca^{2+} to flow across the channel. These voltage- and ligand-gated receptors have slower activation and inactivation timescales, allowing excitatory currents to persist for longer periods than if (excitatory) AMPA receptors alone were present [19]. They have long been implicated in long-term memory, since they allow the secondary messenger Ca^{2+} to pass through the membrane, driving synaptic plasticity through LTP (long-term potentiation) [29]. However, Adler et al. found evidence that links NMDA receptors to the maintenance of short-term memory as well [30]. Therefore, computational spiking models of working memory have been devloped that, along with the usual AMPA and GABA synapses, incorporate NMDA receptors as well [28, 29, 31]. These models have shown that, indeed, such synapses can allow for active states to persist well after the stimulus has been removed.

Even though an increase in firing rates of networks of neurons that persists for several seconds following stimulus removal has been established as the neural correlate of working memory, exactly how the information is encoded is still an open question, in part because the variation in the rates of neurons is more complicated than a simple increase. Different neurons have been been shown to increase, decrease, oscillate, or maintain their firing rates during short-term retention, e.g. [25, 32, 26]. Other work has shown that populations of neurons oscillate in their activities, including in LFP (local field potential), EEG, and MEG recordings [26, 33, 34]. In particular, increases in frequencies in the gamma and alpha ranges [33, 35] have been measured during delayed response working memory experiments. These population-level oscillations may play a critical role in the coding, maintanence, and processing of items in working memory. For example, Lisman and Idiart proposed that theta-gamma cross-frequency coupling may allow multiple items to be stored based on their relative phases, while also accounting for the finite capacity of working memory [36].

While the early working-memory experiments focused on the representation of elementary, possibly atomic, attributes such as location, in reality visual objects have multiple features associated with them, such as contours, colors, and orientation. Other, non-visual attributes might include the temperature, sound, and scent that are associated with an object. The same is true of more abstract objects as occur in language, so that values can be attached to general variables (e.g., "the runner is fast"). How such attributes are assigned to objects is known as "the neural binding problem" [37]. In 1981, von der Malsburg proposed that the encoding of multiple features of an object might occur through neuronal synchrony (see [38]; note, this was originally published in 1981 as an internal report for the Max Planck Institute). This hypothesis was lent empirical credence in 1989, when the Singer lab found evidence for synchronous oscillatory activity associated with the stimulus characteristics in cat visual cortex [39, 40].

Together, this suggests that relative phase timings in the oscillatory activity of populations of neurons equipped with NMDA receptors might be responsible for the maintenance, segregation, and conjunction of information held in short-term memory. In particular, the phase information of such oscillatory ensemble dynamics is a natural candidate to distinguish between bound (synchronous oscillations) and distinct (out-of-phase oscillations) items held actively in working memory, while excitatory synapses that are slow to activate and slow to decay can maintain the high activity after the stimulus has been shut off or removed. We thus use a set of Wilson-Cowan-type equations to model the activities of AMPA, GABA, and NMDA synapses that have oscillatory dynamics, and find that the NMDA components allow for the persistence of high-activity states. This is instantiated as bistability between a low baseline firing rate and a high, oscillatory firing rate, where the transition occurs as a result of a pulse of excitatory current to a selected population.

Coupling several of these populations together to form a firing rate network allows for competitive oscillatory dynamics, whereby different populations may be pairwise synchronous or out-of-phase, corresponding to bound or distinct items in memory, respectively. We find that up to 3 populations may oscillate out-of-phase with plausible model connectivities and parameter values, a result that is consistent with working memory capacity. The formulation of the model allows us to better examine from a dynamical systems perspective how these states arise as bifurcations of steady states and periodic orbits. In particular, we look at the ranges of coupling strengths and synaptic timescales that allow for synchronous and out-of-phase attracting states. We also explore how varying patterns of selective stimuli can produce and switch between rich sets of dynamics that may be relevant to working memory states and their transitions.

1.3 SENSITIVITIES TO STATIC, SPATIALLY PATTERNED STIMULI

In 1950, a 6 year old boy, recovering from an illness involving fever, began to experience absence (petit mal) seizures, and occasional tonic-clonic (grand mal) seizures when he would look at finely-striped patterns, such as corduroy or copper mesh [41]. This is the first documented case of so-called “pattern-sensitive epilepsy” (PnSE). PnSE is a subset of photosensitive epilepsy, in which patients experience seizure due to patterns of light, especially the rate at which it flashes [42]. While thought to be extraordinarily rare, PnSE was later found to be quite prevalent among patients who already experienced seizures induced by light flicker, with 11 of the 32 such patients in the study shown to exhibit epileptiform activity when exposed to stationary patterns [43].

Soon after, the effects of different pattern characteristics were studied in some detail [44]. The authors found that the luminance contrast and the spatial frequency in particular had the greatest effects: The greater the contrast, the more likely a pattern was to produce epileptiform activitiy, while frequencies between 3 and 5 cycles per degree (cpd) were most likely to trigger such EEG readouts. They found this band of wavenumbers matched with the most sensitive wavenumbers in contrast sensitivity tests in healthy subjects quite well, where the sensitivity was measured as the inverse of the contrast necessary for subjects to detect bars in periodic grating patterns [45][46].

Moreover, Wilkins found that, while showing these patterns to patients, family members with no history of seizures reported distinct discomfort when viewing the same images [42]. Further study revealed that striped patterns in the same sensitive band as found for the induction of seizures in those with PnSE were most likely to induce headaches, the feeling of unpleasantness, and visual illusions. Illusions include the appearance of additional colors and geometries (such as rhombuses) and the breakup of the lines to appear curvilinear or even as like chain links [42].

Such reactions to images extend to more complex images as well. Several examples are given in Fernandez and Wilkins, including reactions to art work in a 1971 exhibition made in the so-called Op Art style (Optical Art, generally highly geometric art made to induce optical illusions such as the false perception of movement and flickering) that reportedly caused guards to suffer headaches, and to art work displayed at a hospital in London that gave some staff members migraines in 2005 [47]. By examining such complex images, including abstract artwork and noisy images, Fernandez and Wilkins found that the subjective reporting of the level of discomfort correlated strongly with the wavenumbers of the Fourier components. Even with such complex stimuli, the dominance of modes within the same sensitive 3–5cpd band of spatial frequencies predicted the level of discomfort experienced by subjects.

The pathological responses to these stimuli have been found to have oscillatory correlates via MEG [48], motivating us to seek a simple model that exhibits oscillatory dynamics when presented with a static, spatially periodic stimulus. By using a Wilson-Cowan neural field network and adjusting system parameters such as the amount of recurrent excitation, we may place the stimulus-free system near a so-called Turing-Hopf bifurcation, where the uniform steady state is spontaneously lost to temporally and spatially periodic patterns with wavenumber m^*. We study the model extended in both one and two spatial dimensions, employing simulations, numerical bifurcation analysis, and perturbation theory to characterize the network behaviors.

Our results suggest that different parameter values involving, e.g., connectivity strengths, could cause neuronal networks to exhibit a natural sensitivity to particular spatial frequencies. The spatially-resonant behaviors observed in pattern-sensitive epilepsy and with aversive images may then be explained within the framework of dynamical systems, as spatially periodic stimuli with components near the resonant wavenumber push the system past a symmetry-breaking bifurcation into a spatiotemporal pattern forming regime.

1.4 OSCILLATIONS IN NEURONAL MODELS

To model the neuronal dynamics that we describe above, we wish to use parameters that allow for oscillatory solutions. Indeed, many of the types of neuronal dynamics of interest involve temporally or spatiotemporally complicated behaviors. When modeling this activity in the context of dynamical systems, such oscillatory activity often arises as limit cycles or bifurcations from limit cycles. While establishing the existence for periodic orbits in many planar ODE systems is often feasible, even determining how many limit cycles exist in such simple systems is an open problem (cf. Hilbert's 16th problem), let alone in ODE systems of 3 or more variables. A number of analytical and numerical methods have been developed to determine in certain circumstances whether a given system has stable limit cycles (see [49] for a review). One method involves finding bifurcations in a given system that are known to give rise to limit cycles, especially co-dimension 1 bifurcations (those that occur as a single parameter in the network changes). The Hopf (or Andronov-Hopf) bifurcation is a common co-dimension 1 bifurcation to use to this end. As this is the mechanism that underlies many of the oscillations in this dissertation, we briefly describe the features of this bifurcation below.

1.4.1 *The Hopf bifurcation*

In a Hopf bifurcation, stable or unstable limit cycles emerge from a fixed point as a system parameter is varied. Generically, this occurs when a pair of complex-conjugate eigenvalues of the linearized system cross the imaginary axis at nonzero imaginary values. That is, consider the dynamical system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x};\ \zeta), \qquad \text{with } \boldsymbol{x} \in \mathbb{R}^N \text{ and } \zeta \in \mathbb{R}, \tag{1.2}$$

where $\dot{\boldsymbol{x}}$ denotes the time derivative of $\boldsymbol{x}$, $N \in \mathbb{N}$, $\boldsymbol{f}$ is smooth (at least C^3), and ζ is a parameter. The linearized system is then $\dot{\boldsymbol{x}} = J(\zeta)\boldsymbol{x}$, where J is the Jacobian matrix, $J(\zeta) = D_{\boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x};\ \alpha)|_{\boldsymbol{x}_0}$, where $\boldsymbol{x}_0$ is the fixed point of Eq (1.2) (i.e., $\boldsymbol{f}(\boldsymbol{x}_0;\ \zeta) = 0$). If $J(\zeta_0)$ admits purely imaginary eigenvalues $\pm\lambda_{1,\,2} = \pm i\,\eta_0 \neq 0$, then, generically, the system Eq

(1.2) will admit limit cycles that bifurcate from x_0, ζ_0 [50]. The bifurcating limit cycles can be stable (supercritical Hopf) or unstable (subcritical Hopf). In the former case, small, stable limit cycles bifurcate from the steady state. The amplitudes go to zero as $\zeta \to \zeta_0$ from the direction of the unstable steady state. In the latter case, the branch of limit cycles may fold back, leading to (i) bifurcation of the steady state to a large limit cycle, and (ii) a region of bistability between the steady state and a large limit cycle. We illustrate these properties with a simple planar system that, as we will see, is in some sense the prototye for Hopf bifurcations. The example and discussion closely follow [51], Sec 3.4.

Consider the system:

$$\begin{aligned} \dot{x} &= a\,x - y + (b\,x + c\,x^3)(x^2 + y^2) \\ \dot{y} &= x + a\,y + (b\,y + c\,y^3)(x^2 + y^2) \end{aligned} \tag{1.3}$$

Since the Jacobian evaluated at the fixed point (0, 0) is

$$J(a) = \begin{bmatrix} a & -1 \\ 1 & a \end{bmatrix},$$

the eigenvalues as a function of the parameter a are $\lambda_{1,2}(a) = \frac{-a}{2} \pm \sqrt{\frac{a^2}{4} - (a^2+1)}$, which are purely imaginary at $a = 0$: $\lambda_{1,2}(0) = \pm i$. We can determine the behavior of the system by transforming it into polar coordinates. We do so by letting $z = x + iy$, finding that

$$\dot{z} = (a + i)z + b\,z|z|^2 + \mathcal{O}(5), \tag{1.4}$$

where $\mathcal{O}(5)$ indicates quintic terms in z and $\overline{z}$. We note that these terms are 0 in the case that $c = 0$; by making c negative, we ensure that solutions remain bounded. Below we discuss this further. By rewriting z in polar coordinates, as $z = r\ \exp(i\theta)$, and differentiating, we find

$$\dot{z} = \dot{r}\exp(i\theta) + i\dot{\theta}\,r\ \exp(i\theta). \tag{1.5}$$

By setting Eq (1.4) equal to Eq (1.5), we obtain Eq (1.3) in polar coordinates (up to 3rd order terms):

$$\begin{aligned} \dot{r} &= a\,r + b\,r^3 \\ \dot{\theta} &= 1. \end{aligned} \tag{1.6}$$

Since such a change of variables is always possible for Hopf bifurcations, Eq (1.6) represents their *normal form.* We see that there are two steady state r values: 0 and $\sqrt{\frac{-a}{b}}$. Differentiating the right-hand side of the r equation wrt r and evaluating at these fixed points reveals that $r = 0$ is stable for $a < 0$ and unstable for $a > 0$, while $r = \sqrt{\frac{-a}{b}}$ exists and is stable for $a > 0,\ b < 0$, and exists and is unstable for $a < 0,\ b > 0$. Hence, the former corresponds to a supercritical Hopf bifurcation, and the latter to a subcritical Hopf bifurcation. Both of these situations are shown in the bifurcation diagrams generated in AUTO in Fig 1. In the subcritical case in Fig 1B, we see that, in addition to the unstable limit cycles that the preceeding local bifurcation analysis revealed, there are stable limit cycles that arise as a saddle-node bifurcation of limit cycles. Thus, as we noted above, there is an interval of the parameter a wherein the system is bistable between the $r = 0$ fixed point and large-amplitude limit cycles. Additionally, the $r = 0$ steady state bifurcates to large limit cycles as a becomes positive, compared to the bifurcation to small limit cycles whose amplitudes limit to zero in the supercitical case. The large limit cycles in the subcritical case do not arise if we set $c = 0$. However, we have included the terms involving nonzero c in our example since the large-amplitude limit cycles that bifurcate from the unstable limit cycles at the saddle-node of periodic orbits are common in systems of Wilson-Cowan equations, where the firing rate function helps keep solutions bounded.

1.4.2 *Inhomogeneous oscillations*

Suppose we have a spatially continuous neural field model given by Eq (1.1). If such a system goes through a Hopf bifurcation from spatially uniform resting state, the oscillations that arise can be either spatially homogeneous (i.e., bulk oscillations of the media) or inhomogeneous. If the latter, a particular wavenumber, m^*, will be dominant near the bifurcation, so

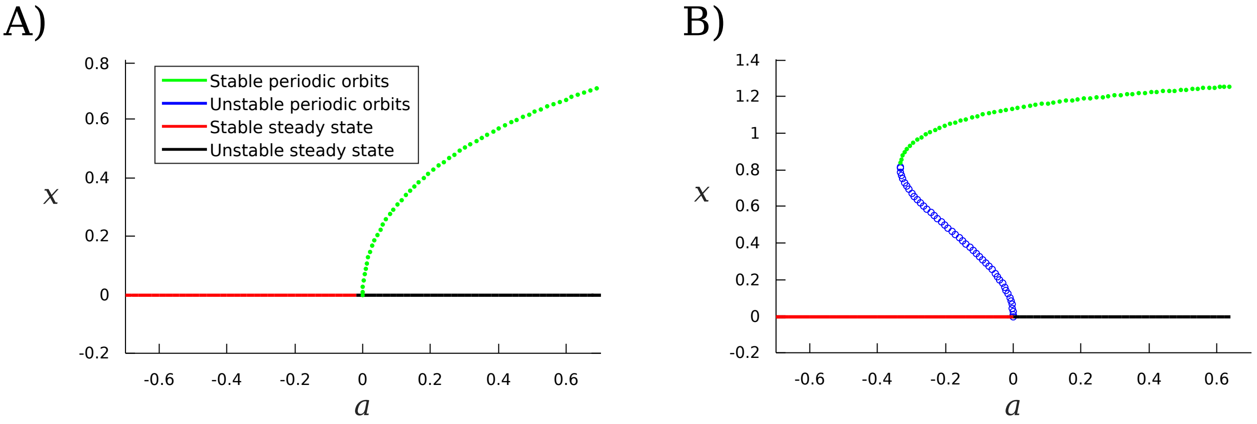

Figure 1: Hopf bifurcations of Eq (1.3). As the parameter a increases from negative to positive values, the steady state at $(x, y) = (0, 0)$ destabilizes. It does so as a pair of complex conjugate eigenvalues cross the imaginary axis. (A) Supercritical Hopf bifurcation. In this case, $b < 0$, leading to stable limit cycles that emanate from the bifurcation point, as can easily be seen by a linear stability analysis of the system in polar coordinates, Eq (1.6). (B) Subcritical Hopf bifurcation. $b > 0$, leading to unstable limit cycles stemming from the bifurcation point (again, as can be determined from Eq (1.6)). By letting $c < 0$ in Eq (1.3), solutions remain bounded, and stable limit cycles bifurcate from the unstable limit cycles at a saddle-node of periodic orbits, leading to bifurcation to large-amplitude limit cycles once a becomes positive and to an interval of bistability between the steady state at (0, 0) and large-amplitude limit cycles. This is commonly observed in subcritical Hopf bifurcations in Wilson-Cowan networks.

that the oscillatory activity will also display a spatial breakup into m^* regions in the case that Eq (1.1) is extended in one spatial dimension. To distinguish the two cases, the latter (the bifurcation of a uniform steady state to a spatially inhomogeneous oscillatory state) is often known as a Turing-Hopf bifurcation. We show how such dynamics arise and explore the complex behaviors as applied to spatial resonances that appear in visual contexts (discussed below in Sec *1.3*) in Chap 3. We also explore how the unstable modes of the system interact with a spatially periodic stimulus with an adjustable wavenumber. The behaviors available to such a system are complicated; we thus introduce the ideas here in a simplified discrete Wilson-Cowan network. We note that this example is chosen to illustrate the relevant dynamics; the parameters used do not match those in the model presented in Chap 3, but the structure of the system is constructed analogously to the model we use later.

Consider the following 4-component Wilson-Cowan network:

$$\begin{aligned} \dot{u}_1 &= -u_1 + f_e\left(a_{ee}\,\bar{u}_1 - a_{ei}\,\bar{v}_1 + q\,s\right) \\ \tau\,\dot{v}_1 &= -v_1 + f_i\left(\,a_{ie}\;\bar{u}_1 -\; a_{ii}\;\bar{v}_1 + q\,r\,s\right) \\ \dot{u}_2 &= -u_2 + f_e\left(a_{ee}\,\bar{u}_2 - a_{ei}\,\bar{v}_2 + q\,s^2\right) \\ \tau\,\dot{v}_2 &= -v_2 + f_i\left(\,a_{ie}\;\bar{u}_2 -\; a_{ii}\;\bar{v}_2 + q\,r\,s^2\right), \end{aligned} \tag{1.7}$$

where

$$\begin{aligned} \bar{u}_1 &= (1-\sigma_e)u_1 + \sigma_e\cdot u_2 \\ \bar{v}_1 &= (1-\sigma_i)v_1 + \sigma_i\cdot v_2 \\ \bar{u}_2 &= (1-\sigma_e)u_2 + \sigma_e\cdot u_1 \\ \bar{v}_2 &= (1-\sigma_i)v_2 + \sigma_i\cdot v_1. \end{aligned} \tag{1.8}$$

The firing rate functions are given by the sigmoids $f_{e,i}(w) = \frac{1}{1+\exp(-w+\theta_{e,i})} - \frac{1}{1+\exp(\theta_{e,i})}$. s and s^2 are the stimuli, where $s \in \{-1, 1\}$. This allows us to stimulate the network with either a constant ($s = 1$) or alternating ($s = -1$) stimulus. The stimulus strength is controlled by q and $q\,r$, where r, the ratio of the strength of the inhibitory stimulus to that of the excitatory stimulus, remains fixed. The parameters of the network are $\tau = 4$, $\sigma_e = 0.015$, $\sigma_i = 0.25$, $a_{ee} = 20$, $a_{ei} = 16$, $a_{ie} = 20$, $a_{ii} = 7$, $\theta_e = 0.2$, $\theta_i = 0.3$, $r = 0.5$.

Surprisingly, this minimum Wilson-Cowan network can demonstrate spatiotemporal pattern formation (over discretized space). We note, similarly to the modes available to the stimulus mentioned above, there are two symmetric oscillatory modes available to the network: synchronous and alternating. With other parameters fixed and no stimulus applied ($q = 0$), increasing a_{ee} beyond a certain critical level, a^*_{ee}, causes the network to bifurcate from the uniform steady state when $a_{ee} < a^*_{ee}$ (of $(u_{1,2},\ v_{1,2}) = (0,\ 0)$ ((i) in Fig 2) to alternating oscillatory solutions when $a_{ee} > a^*_{ee}$: u_1 and u_2 oscillate in antiphase with each other ((ii) in Fig 2). Thus, this is a sort of discrete version of the Turing-Hopf bifurcation.

The network also destabilizes when a stimulus of this same alternating mode is applied; i.e., by letting $s = -1$, thereby applying the stimulus $[-q,\ q]^T$ to the excitatory populations (and $[-r\,q,\ r\,q]^T$ to the inhibitory populations). When we first turn the stimulus on (r fixed, let $|q| > 0$), when $a_{ee} < a^*_{ee}$, the uniform steady state, (i), is immediately lost to a nonuniform steady state, (iii). Further increasing the stimulus results in a Hopf bifurcation of the nonuniform state, leading to an asymmetric alternating oscillation, (iv). The curve (a) in Fig 2, obtained by following the Hopf bifurcation in XPP-AUTO [16] in the two parameters a_{ee} and q, is the stability boundary of the inhomogeneous oscillatory solution: below the curve, we obtain stable steady-state dynamics; above the curve, we obtain spatiotemporal pattern formation, with spatially and temporally periodic solutions. Since the bifurcation that occurs with no stimulus (i.e., the bifurcation of the steady state, (b) in Fig 2) is the only bifurcation wherein the spatially uniform steady state bifurcates to a spatiotemporal patterns, this is the only analog of the Turing-Hopf bifurcation. The other spatiotemporal patterns that form as the stability boundary, (a), is crossed with increasing q are simply Hopf bifurcations that arise from the nonuniform steady states. We note that the symmetry observed across the q axis occurs due to the periodic boundary conditions implicit in Eq (1.8). Many of the essential behaviors that we will see in the continuous spatial mean-field networks in Chap 3 are analogous to the behaviors we see in this simple network.

1.5 OUTLINE

We begin in Chap 2 by looking at oscillations in a working memory model. After reviewing the relevant literature, we introduce the model and stimulation protocol we use to activate

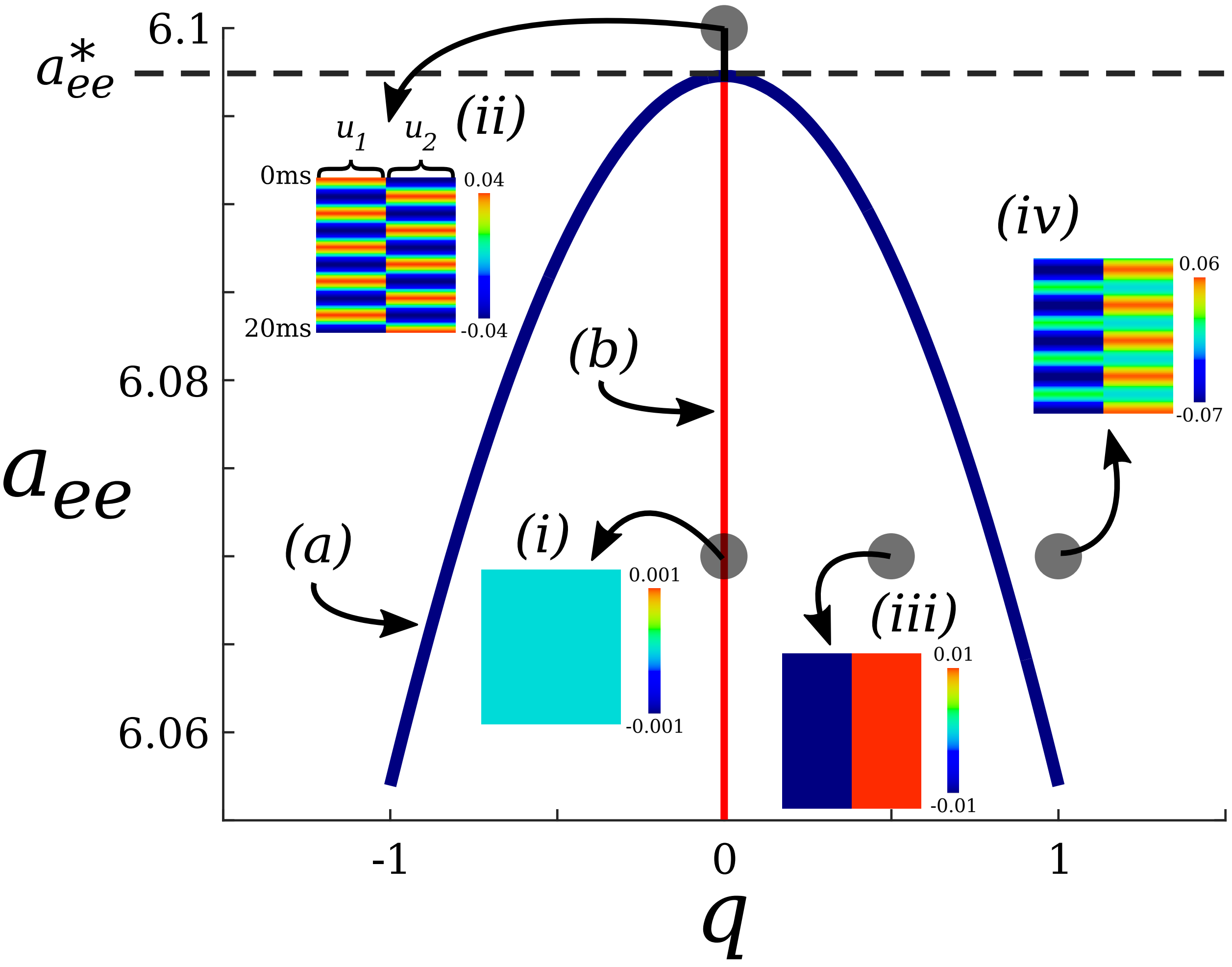

Figure 2: Bifurcations of Eq (1.7). Note that for the images (i)–(iv) that show the dynamics of the network, the axes are all the same as those indicated in (ii). We first focus on the stimulus-free network, so that $q = 0$, corresponding to the line (b). For $a_{ee} < a_{ee}^*$, the uniform steady state is stable (red color), as shown in (i). This state destabilizes at a_{ee}^*, so that for a_{ee} above this critical value, the network destabilizes to alternating oscillations, as shown in (ii). This corresponds to a discrete analog of a Turing-Hopf bifurcation. If the stimulus with the same alternating mode is turned on (delivering $[q, -q]^T$ to u_1 and u_2 and $[rq, -rq]^T$ to v_1 and v_2), the uniform steady state immediately bifurcates to an alternating steady state, (iii). If the amplitude q increases enough, this state is then lost through a Hopf to alternating oscillations, (iv). Thus, the stability curve (a) divides steady state from nonuniform oscillatory solutions.

selected populations. We then present the relevant synchronous and asynchronous oscillatory dynamics we achieve, and analyze the dependence of these solutions on the parameters. In Chap 3, we look at a model to capture the sensitivity to spatial patterns that people can experience. After introducing the biological background, we explore how a neural field model extended in one and two spatial dimensions can capture many of the relevant resonant behaviors that have been found psychophysically. Finally, in the appendices, we cover some of the more technical material, such as the working memory model derivation and the linear stability calculation for determining the stability boundary at which spatiotemporal pattern formation can occur.

2.0 OSCILLATIONS IN WORKING MEMORY AND NEURAL BINDING: A MECHANISM FOR MULTIPLE MEMORIES AND THEIR INTERACTIONS

This chapter is based on [52].

2.1 INTRODUCTION

The neuronal substrate of working memory is thought to be persistent elevated firing rates of neurons that have been found in numerous physiological and imaging studies across widely-varying scales, from single neurons up to neuronal populations and networks [23, 28, 53, 26, 25, 54]. Moreover, for working memory to function properly these populations must be able to be activated rapidly by external or internal input corresponding to aspects of memoranda, that activation must be selective and stable, and the elevated firing rates must be able to return to background levels when the information is no longer needed. Cognitive and behavioral considerations further imply that mechanisms must exist for rapidly transitioning between sequences of active memories, and that multiple (and possibly overlapping) selective populations and networks can be simultaneously active. These factors apply both to the case of neural binding, in which the activity of disparate neuronal populations or networks must be combined and maintained (e.g., corresponding to different aspects of a particular item in memory or working memory task), and for the case in which multiple different items are simultaneously maintained in working memory. In this latter case, there are well established approximate upper limits to the number of separate objects that can be simultaneously maintained. Indeed, one of the fundamental properties of working memory is that it has a limited capacity, possibly limited to three to five objects [55, 20, 56, 57, 58].

Binding refers to how items encoded in distinct brain circuits or neural populations can be combined for perception, decisions, and action, and can be partitioned to encompass multiple situations [37]. These different binding types include feature binding, which involves the association of different characteristics to an object and how we make and then unravel these associations, and variable binding, which arises, for example, in language and other symbolic thought (e.g., the binding of a variable name and its value). In every instance, some form of synchronization of neural activity has been proposed as the underlying mechanism (e.g., [38, 59, 60, 61, 62, 63, 37]). Binding is a key aspect in working memory, as most objects we encounter, physical or symbolic, are multi-featured. There is also a limited capacity to the number of features that may be represented for objects; however, it is unclear at present exactly how this feature capacity relates to the aforementioned working memory capacity [56, 64, 65].

In many studies, oscillatory dynamics have been identified with cognitive function in general and, increasingly, with working memory in particular (e.g., [66, 67, 33]). While the presence of different patterns of oscillations is well documented, the specific roles they play are not well understood. Recent work has suggested, however, that oscillations in various frequency bands and coupled or nested oscillations could play a fundamental role in the functioning of different aspects of working memory [35, 54, 68, 69, 70]. Oscillatory dynamics may also play a critical generic role in facilitating the range of dynamics and optimized conditions required for working memory function as described above.

In the present work, the networks are capable of producing a range of oscillatory frequencies implicated in working memory and consistent with these other studies. However, we did not focus on particular frequency bands in terms of their implications for a working memory code, but rather on the role they can play in facilitating the essential functions of working memory. In particular, we examine the role oscillatory states can serve as an underlying mechanism to allow for multiple stable, active items in memory, to establish binding via synchronous relationships, to transition between different active working memory states, and to rapidly activate and terminate activity in those networks as required by the needs of cognition and thought. We show how oscillatory dynamics may facilitate these potentially competing requirements, and identify and discuss critical network parameters involved in achieving these dynamics. Indeed, although all of the network's parameters play a part in the allowed dynamics, we observed that different parameters played roles of varying importance to each of the states. We thus found it fruitful to first explore some of the features and constraints of the asynchronous and the synchronous states individually, and then to see how these attracting states naturally combine to form more complex activity involving both synchronous and asynchronous oscillations that may represent multiple rich, multi-feature memories. We found that the attracting synchronous and asynchronous states allowed for the representation of bound and distinct items in memory, for multiple items to be bound or to be maintained as distinct, representing different network capacities, and for rapid transitions between the different states as is required during cognition.

2.2 METHODS

Oscillations during memory task delays have been seen in population-level activity, including in local field potential recordings and human EEG traces [25, 26]. This motivates us to model working memory oscillatory dynamics at the ensemble level, using Wilson-Cowan type equations for the firing rates (or synaptic activities) of each local circuit that will comprise the model. Here, we provide the outline of the model. In Appendix A we develop the model in greater detail, beginning with a quadratic integrate-and-fire network. Since previous experimental and computational studies suggest NMDA receptors may be crucial to the persistent elevated firing rates associated with proper working memory function in experimental and computational studies, we model the effect of NMDA receptors as a separate component [29, 30, 71, 28, 31].

We note that many population models only include the excitatory and inhibitory dynamical variables of the network, here denoted as u and v. Thus, the models generally appear as coupled first-order equations,

$$\tau_e\, u' = -u + f_e(u,\, v)$$
$$\tau_i\, v' = -v + f_i(u,\, v),$$

where $\tau_{e,i}$ are the time constants of the excitatory and inhibitory components, respectively, and $f_{e,i}$ are transfer functions describing the response of the network to the u and v variables, and are often sigmoidal in shape.

The variables u, v are variously interpreted as the firing rates, "activities", or synaptic drives of the network [17, 72, 73, 74]. In Pinto et al. [73], a population model was developed with the goal of making quantitative comparisons to data taken from rat whisker barrels in the primary somatosensory cortex, and was found to match the general form of the Wilson-Cowan equations, allowing for physical quantities to be explicitly assigned to the equations' variables. In particular, it was determined that u corresponds to the excitatory synaptic drive, v corresponds to the inhibitory synaptic drive, and the transfer functions $f_{e,i}$ correspond to the actual firing rate of the network. Thus, u and v may be called the "activities" of the network and represent low-pass filters of the network firing rate [17]. This formulation allows for the inclusion of other relevant synaptic activity that will then appear as additional dynamical variables to the system. Here, for example, by including NMDA receptors we end up with a third dynamical variable that we denote as n. Hence, Wilson-Cowan networks are better understood as modeling the synaptic activities of the underlying neurons, only indirectly representing the activities of the excitatory and inhibitory neurons that comprise the medium. This is made explicit in our development from a QIF network in Appendix A. The system of equations that result from this development are:

$$\begin{aligned} u_j' &= -u_j + f(a_{ee}\cdot \widetilde{u}_j - a_{ei}\cdot \widetilde{v}_j + a_{en}\cdot \widetilde{n}_j - \theta_e + s_j(t)) \\ \tau_i \cdot v_j' &= -v_j + f(a_{ie}\cdot u_j - a_{ii}\cdot v_j + a_{in}\cdot n_j - \theta_i) \\ \tau_n \cdot n_j' &= -n_j + a_n \cdot u_j^p(1-n_j), \end{aligned} \tag{2.1}$$

with $j \in \{1, ..., N\}$, where N is the number of interconnected populations, or u-v-n triplets, where each such triplet represents a tightly recurrently connected population of excitatory and inhibitory neurons with fast AMPA synapses (u_j), a slow NMDA component of the excitatory synapse (n_j), and slow GABA synapses (v_j). $a_{\gamma\delta}$, for $\gamma,\ \delta \in \{e,\ i,\ n\}$, give the coupling strengths between different components of each population (these values do not vary between the populations in the model). $\widetilde{u}_j$, $\widetilde{v}_j$, and $\widetilde{n}_j$ represent coupled populations:

$$\widetilde{\alpha}_j = \left(\alpha_j + c_z \sum_{k\neq j} \alpha_k\right) \Big(1 + c_z\ (N-1)\Big)^{-1}, \tag{2.2}$$

where $\alpha \in \{u,\ v,\ n\}$, and c_z represents the relevant coupling from population k to population j: c_e for NMDA (n) and AMPA (u) synapses and c_{ei} for the inhibitory (v) synapses. The denominators ensure the excitation and inhibition remain bounded. Thus, c_e and c_{ei} give interpopulation coupling strengths, while a_{ee}, a_{ei}, a_{ie}, and a_{ii} give intrapopulation coupling strengths. Fig 3 shows a schematic of the connectivities. The interpopulation excitatory coupling c_e represents overlap between populations. The mutual inhibition, c_{ei}, represents the coupling strength from v_j to u_k ($j \neq k$), resulting in reciprocal inhibitory connections between pairs of populations (e.g., populations 1 and 2 inhibit each other when c_{ei} has a nonzero value). Thus, c_{ei} allows for competitive dynamics between populations, as would the value of c_{ie}, the interpopulation coupling strength from u_j to v_k ($j \neq k$). However, we found that both of these connections provide similar dynamics to the results described in Sec *2.3* (see, e.g., Table 1 in the weak coupling case in Appendix C); thus, we have set $c_{ie} = 0$ for simplicity.

We assume coupled populations reside in neighboring areas, so that delays need not be considered, and that $\tau_n > \tau_i > 1$, where we have rescaled time so that the timescale is 1 for the fast excitation. An input stimulus, $s_j(t)$, allows the system to be activated. This stimulus is a square wave, for which we may adjust the amplitude, A_j, as well as the width, w_j, and onset time, t_{0j}. The firing rate is given by $f(x) = \sqrt{\frac{x}{1-\exp(-\beta \cdot x)}}$. We have chosen this firing rate since it gives an approximation of the firing rate of a noisy quadratic-integrate-and-fire spiking neuron [72, Chap 10]. In Appendix A, we show an example simulation of the spiking model and compare it to the one-population version of the Wilson-Cowan equations.

Since we do not explicitly model membrane potential or spiking, there is no voltage dependence to the NMDA component; rather, it acts as a slow excitatory input. The NMDA

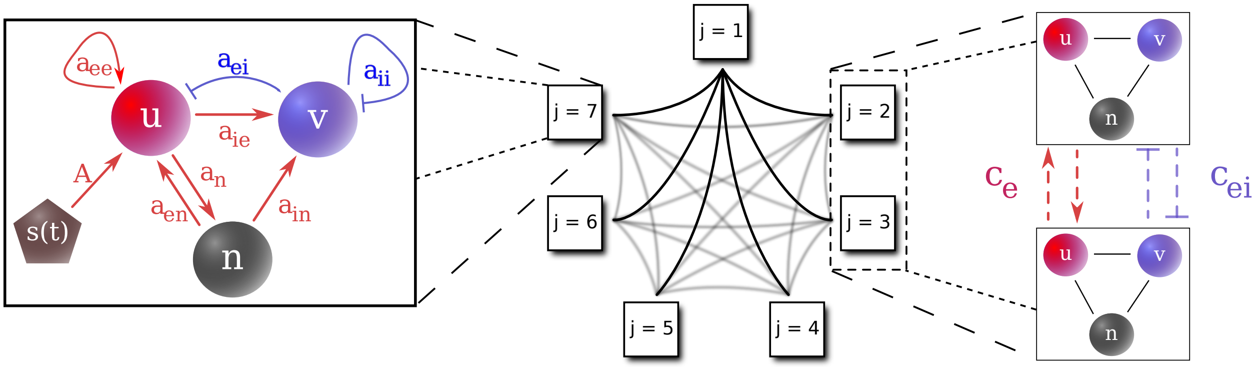

Figure 3: Model connectivity. Left: One population as described in Eq (2.1). u is the fast excitatory AMPA synaptic activity, v the inhibitory GABA activity, and n the slow excitatory NMDA activity. Feedforward excitation to the AMPA synapses (u) triggers activity in the system. Middle: An example network with $N = 7$ populations. The elements are coupled in an all-to-all fashion as shown on the right. Right: Connectivity between two populations, each with three components as shown in the left panel. The populations are connected with excitatory (c_e) and inhibitory (c_{ei}) coupling as described in Eq (2.2).

component also saturates at 1, while the faster variables, u and v, are not constrained. The parameters a_n and p in the NMDA equation allow us to adjust the average NMDA level when that component is active; larger p also results in faster NMDA activation since u is typically larger than 1 when activated. Note that the behavior is fairly insensitive to p; the $n-u$ null surface (where $n' = 0$) is sigmoidal and p increases the gain of this sigmoid.

Unless otherwise specified, the parameters used are $N = 5$, $\tau_i = 12$, $\tau_n = 144$, $c_e = 0.001$, $c_{ei} = 0.03$, $a_{ee} = 14$, $a_{ei} = 10$, $a_{en} = 4$, $\theta_e = 6$, $a_{ie} = 20$, $a_{ii} = 8$, $a_{in} = 0.1$, $\theta_i = 5$, $a_n = 2$, $\beta = 1$, $p = 2$. These parameters allow for tristability among three behaviors of interest: a low nonzero steady state, a low-amplitude oscillation around this low steady state, and a large-amplitude oscillation. In our model, "active populations" refer to populations that are engaging in large-amplitude oscillations. Note that we vary N, τ_i, τ_n, c_e, and c_{ei} to study their effects below. In particular, we find that there are open sets of these parameters that allow for the dynamics of interest, so that the precise values used are not critical to obtain the results shown.

Our protocol for simulating working memory is to load a memorandum via a square wave pulse stimulus ($s_j(t)$ in Eq (2.1)). Each stimulus is associated with the excitatory component of a specific population, so that a particular population may be selected for a memorandum. Multiple populations may be selected and provided stimuli either simultaneously or serially. Varying each of the amplitude A_j, onset time t_{0j}, and width w_j may produce different activation patterns, so that the observed activity is a direct result of the selected parameters and sequence of feed-forward inputs.

Each stimulus corresponds to a feature of an item present in the environment, so that persistent activation of a population selected by a stimulus corresponds to that feature's representation within the neuronal network, where as mentioned above we allow for some overlap of excitatory populations, as represented by a nonzero c_e value. We then study how the system responds to patterns of stimuli, allowing their amplitudes, widths, and onset times to vary. We are particularly interested in phase synchrony and asynchrony of the large amplitude oscillations across the excitatory components of the populations. In our model, synchrony corresponds to different features that are associated with one another held active in working memory, while asynchrony corresponds to independent features held active. These behaviors correspond to binding and working memory capacity, respectively. We further study the existence and stability of these states while varying key parameters using XPP-AUTO [75].

However, since we obtain these network behaviors through a feedforward excitatory impulse to the excitatory components, we are further interested in which states are in fact accessible from some other given state. In systematically studying what patterns may be obtained from other patterns following a selective stimulus, the combinatorics involved quickly threaten to make the problem intractable. Thus, we limit our study to just two and three active populations, which we will refer to as diads and triads, respectively. We begin with either a diad or triad, and then use the protocol as described above, providing a single, selective stimulus of a chosen amplitude and width either to an active population or to an inactive one, and allow the network to evolve for some time afterwards. The resultant pattern is then determined to be accessible from the initial pattern. The outcome of stimulating the system is phase-dependent, as is the case in many oscillatory systems. Since we are interested in behaviors that may be more robust, we narrow our results to only include those that contain multiple-millisecond intervals of time that produce the same resultant activity for a stimulus of fixed width and amplitude. All of the included results may be obtained with a stimulus of fixed width and amplitude over intervals no smaller than 6ms, consistent with intervals associated with gamma frequency phase locking that some pyramidal neurons and interneurons display, as described in Senior et al. [76] The authors found that these neurons locked their firing to two phases within one gamma (30–80Hz) period. Thus, the smallest interval of time that these neurons might be able to distinguish is 6–7ms. Additionally, synchronization between neurons has been shown to occur within a time window of around 10ms [77]. For simplicity, we only consider a network with $N = 5$ when we explore the accessibility of operations involving diads and triads.

2.3 RESULTS

The model produces persistent elevated firing states (above baseline levels) in response to inputs to selected populations in networks consisting of multiple populations, consistent with what has been observed in neurophysiological studies (e.g., [25]). This persistent working memory activation may occur either as steady-state or oscillatory firing rates, depending on the value of the relative speed of the inhibitory synapses. For the case of oscillatory dynamics, analogues of several critical features of working memory arise naturally and robustly, which, in addition to the persistent elevated activity, include working memory capacity and binding. In contrast, for the steady-state case, it is difficult to obtain multiple populations active at once, and in the case of uniform connectivity, the activity would be indistinguishable (i.e., they would all be active with the same firing rate). An analysis and description of the mechanisms underlying the oscillatory behavior of an isolated $u-v-n$ triplet is presented in Appendix B.

We found that our model allows for stable oscillatory states involving a single population (SO: Single Oscillator) or multiple coupled populations. In the latter case, three basic types of oscillations may occur: out-of-phase (OP), synchronous (S), and mixed phase (MP). By OP, we mean that if there are k active populations, then each of the k occupy separate parts of a cycle. For example, if $k = 2$, then the two populations oscillate a half cycle apart, and if $k = 3$, each population oscillates a third of a cycle apart. In our model, OP populations represent single-feature, distinct items in memory. S means all populations fire in-phase with one another and represent bound items in memory, while MP means that some populations are synchronized and some are out of phase, corresponding to bound and distinct items in memory, respectively. For example, it could be the case with three populations that two are synchronized while the third fires out of phase, corresponding to two distinct objects held in memory: one with two features and one with a single feature.

Although all of the network's parameters play a part in the allowed dynamics, we observed that different parameters played roles of varying importance to each of the states. We thus found it fruitful to first explore some of the features and constraints of the S and especially the OP states individually, and then to see how these attracting states naturally combine to form more complex MP activity that may represent multiple rich, multi-feature memories.

Organization of results

In Sec *2.3.1* we look at OP oscillations in isolation. We look at both example simulations (Sec *2.3.1.1*) and some of the central results on constraints on the number of populations that may oscillate OP (Sec *2.3.1.2*).

In Sec *2.3.2*, we turn our attention to S oscillations. Again, we examine example dynamics (Sec *2.3.2.1*) and limits on how many populations may oscillate synchronously (Sec *2.3.2.2*).

In Sec *2.3.3*, we see how OP and S oscillations may both occur and can lead to MP oscillations, which are a central feature of the model. We look at example activities (Sec *2.3.3.1*) and then try to gain insight into how we might attain the different oscillatory modes by looking at two different simplifying assumptions. First we see how the coupling strengths allow for the different states in the reduced case of just 2 coupled populations (Sec *2.3.3.2*), and then turn to the case of weak coupling (Sec *2.3.3.3*) and examine the basins of attraction for S, OP, and MP modes with just 3 populations.

Finally, in Sec *2.3.4*, we examine the model in relation to biological and cognitive considerations. We first focus on one of the main features of the model by exploring how the network can rapidly form and transition between a rich variety of MP relationships in the context of binding (Sec *2.3.4.1* and Sec *2.3.4.2*). We then more carefully examine the transitions the model network may make (Sec *2.3.4.3*), and compare some of the frequency relationships in the model to experimental results (Sec *2.3.4.4*).

2.3.1 *Out-of-phase oscillations and distinct memoranda*

OP oscillations of the interacting populations in the network (Fig 4) display the fundamental signature of working memory in the model. Each distinct memorandum being held active is associated with a different phase, similar to what has been proposed in previous work (e.g., [70, 36, 78, 79]). In contrast to some of this previous work, however, the present firing rate networks are self-contained. That is, they do not explicitly receive any external signal to organize their relative phase timings. Instead, mutual inhibition allows for competition between the populations so that only one population (or group of synchronously firing populations as discussed in Sec *2.3.3.1*) is active during any given portion of a cycle. Thus, for a given cycle duration (determined by conditions discussed in detail in Sec *2.3.1.2* and Sec *2.3.3.2*), there is a limit to the number of OP populations that the network may support. In the model this corresponds to working memory capacity. We will also refer to this capacity as the network capacity, as it is a fundamental property of the networked populations.

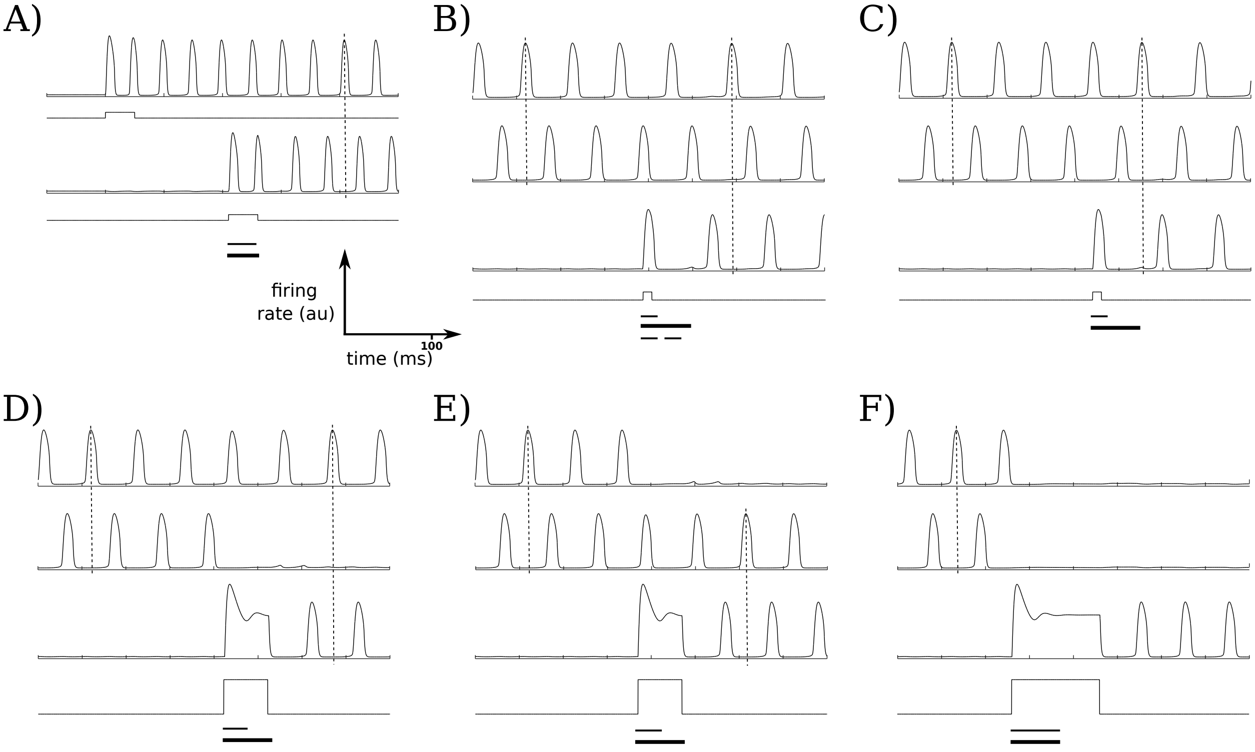

Figure 4: OP dynamics for up to 3 active populations. For a network size of 5 populations ($N = 5$), all combinatorial possibilities for up to 3 active OP populations may be realized. Square waves in these and similar figures indicate the stimuli given to the population just above the given wave, and the vertical dashed lines in each plot allow for phase comparisons across different active populations. The first bar below each group of traces shows the interval of starting times in which the same stimulus (fixed amplitude and width) will produce the same result; the second bar shows the length of the period of the oscillation that is active before the stimulus is applied (note: the period may change after the stimulus is applied); the third bar in (B) is explained below. (A) The network starts at a nonzero, nonactive baseline firing rate. The first stimulus selectively activates the first population, while the other populations remain inactive with low firing rates. A second population is then activated; for these parameters and stimulus strength, almost any stimulus onset time will induce the OP state with 2 active populations, as the bars show. (B, C) Stimulating a third population with a short stimulus induces the OP state with 3 active populations. Either activation ordering may occur, depending on the phase the stimulus is presented; (B) and (C) show

the two different orderings. The third, dashed, bar below the stimulus trace in (B) shows the interval of onset times that induce the OP state with three active populations; the first interval shows the onset times that produce (B) while the second interval shows the onset times that produce (C). (D, E) Larger and wider stimuli may deactivate either of the active populations, so that the network remains in the OP state but with different active populations (WTS scenario). (F) Maintaining the amplitude of the stimulus from the WTS case but increasing the stimulus width allows the third selected population to deactivate both active populations and become the only active population (WTA scenario).

2.3.1.1 ***OP dynamics*** Depending on the timing and strength of the stimulus given to each population, a selected population may either oscillate out-of-phase with currently active populations in the network cycle, or it may compete with one or more populations, quenching their activity while remaining active itself. More concretely, consider the case of a 5-population network ($N = 5$) with a network capacity of 3 active OP populations. If no input is given, all of the populations fire at low, but non-zero, rates. Selectively stimulating first one of the populations, and at a later time a second population, allows for these two populations to be simultaneously active, exhibiting OP oscillations as shown in Fig 4A. The result of subsequently selectively stimulating a third population depends on the strength, width, and onset time of the stimulus. Stimuli of sufficient width and amplitude can result in a winner-take-all (WTA) scenario (Fig 4F), where in the 3-population-capacity network the activation of the third population suppresses the first two populations sufficiently to become the only active population. Weaker stimuli allow for a winner-take-some (WTS) scenario (Fig 4D and 4E); in this case, the third population quenches one population (whose activity returns to baseline level) and becomes persistently active, leaving two OP populations. Both of these cases could represent selective forgetting due to interference. For example, they could correspond to situations in which attention is shifted to one item at the expense of other attended memoranda, effectively resulting in the forgetting of those items that had been held active in working memory.

If the stimulus is of moderate strength (that is, of sufficient amplitude and duration to activate the population, but not sufficient to quench another active population), then, depending on the onset time of the stimulus, the selected population may become interleaved with the other active populations in the current cycle (Fig 4B and 4C). In this case, all populations may fire OP, with the ordering of the newly activated population with respect to the already-active populations within the cycle determined by the stimulus onset time (e.g., Fig 4B vs Fig 4C).

2.3.1.2 ***OP oscillations and working memory capacity*** As mentioned above, for any set of nontrivial parameters (e.g., $c_{ei} \geq \epsilon > 0$) there is a maximum number of OP populations that may be active, resulting in a finite network capacity.

We note that a working memory capacity is entailed either by nonzero coupling (i.e., one of c_e, c_{ei}, c_{ie} is positive), or by the temporal resolution of the network. That is, in a biological network, we would not expect perfect synchronization to occur with any regularity; rather, the detection of synchrony by the network would need to operate within certain temporal bounds, so that oscillations that are within some threshold time measure of being synchronous are in fact read out as synchronous. Indeed, the same is true numerically in our model simulations, if for very narrow temporal bounds. In either case, there is a minimum distance, say $\eta > 0$, that the excitatory peaks, e.g., may be from each other (in the first case, the minimum distance may arise due to the attracting or repelling basins of the synchronous state that result from nonzero coupling strengths, for example). Thus, if T is the period of the network oscillation, the maximum working capacity would be $\frac{T}{\eta}$, which is finite for any fixed set of parameters.

Both the mutual inhibition, c_{ei}, and the timescales, τ_i and τ_n, are critical in determining this capacity. We expect the value of c_{ei} to play a strong role in determining the network capacity since the OP dynamics fundamentally arise from mutual inhibition between the populations in the network (see Sec *2.2* and Sec *2.3.3.2*). The effect the timescales have on the capacity is somewhat more nuanced. The timescale of inhibition works differentially on the excitatory populations, depending on their phases. Briefly, we find that, in particular, larger values of τ_i increase the time the excitatory populations spend near zero more than the time they spend away from zero, providing a greater window of opportunity for other populations to be active. We now look at this in greater detail.

We observe that there appear to be two qualitatively different features to the excitatory oscillation: (1) a pulse-like portion, where the activity rapidly increases and rapidly decreases, which we will call the "active phase"; (2) a portion where the firing rate stays very close to zero, which we will call the "quiescent phase". Although the division between these phases is necessarily arbitrary, a natural threshold choice is the low fixed point, u_* (the baseline firing rate). Thus, for some fixed j, the active phase of a given oscillation, T_a, is defined as the interval of time such that $u_j > u_*$, while the quiescent phase of an oscillation, T_q, is the interval of time such that $u_j \leq u_*$. Thus, T_a is a measure of the width of the pulse, while T_q is a measure of the time it spends near zero. Note that $T_a + T_q = T$, the period of the oscillation. T_a is mostly determined by the rise time of the inhibition, which is strongly affected by the excitatory population. In contrast, T_q is mostly determined by the decay time of the inhibition. See Appendix B for more detail. Thus, while both T_a and T_q increase as τ_i increases, we expect $\frac{T_q}{T_a}$ to increase as well. Indeed we find this to be the case; for example, with our original parameters (including $\tau_i = 12$) and one active population, $T \approx 50$ms, and $\frac{T_q}{T_a} \approx \frac{28}{22} \approx 1.3$. If we increase the timescale of inhibition up to $\tau_i = 20$, $T \approx 76$ms, and $\frac{T_q}{T_a} \approx \frac{48}{29} \approx 1.7$. Thus, the ratio $\frac{T_q}{T_a}$ has increased by a factor of about 1.3. We therefore might intuit that more populations can be active OP as τ_i increases, since more pulses from other populations may "fit" between the pulses of an already-active population. That is, since there is a proportionally larger interval of time the population spends near zero than the interval of time it is active, there is greater opportunity for other populations to be active during this quiescent phase. We also note that while it is important for τ_n to be large enough so that the NMDA population may reactivate the excitatory population (see Appendix B), it otherwise does not affect the period of the oscillation very much. For example, with our original parameters (including $\tau_n = 144$) and one active population, recall that $T \approx 50$ms; increasing τ_n to 240 slightly *decreases* the period to $T \approx 49$ms. (We note that it is not surprising for τ_n to have an inverse relationship with the period, since τ_n mostly controls the decay time of the NMDA; thus, n stays slightly higher for larger τ_n, resulting in slightly faster activation times for u and v. Since τ_n has little effect on the decay time of inhibition, which is mostly determined by the time constant τ_i, we see that increasing τ_n may in fact decrease the period.)

We may follow the oscillatory solutions in AUTO to determine exactly how the network capacity depends on the mutual inhibition and the system's timescales. The OP states are lost as folds of limit cycles or destabilize via torus or period-doubling bifurcations as τ_i increases or decreases with τ_n fixed (Fig 5A illustrates this for the 3-OP state). In particular, we found that for $N = 5$ or 10, the solutions were lost as folds of limit cycles as in Fig 5A, while for $N = 20$ the oscillations sometimes first lost stability via the torus or period-doubling bifurcations (see Fig 5D). For $N = 5$ or 10, the 1-OP state (i.e., just one active oscillating population) is also lost as a fold of limit cycles (in τ_i), suggesting we may gain intuition into why the M-OP states exist for certain τ_i and τ_n values for $M > 1$ by examining the 1-OP case in more detail, which we do at the end of Appendix B by examining how the separation of timescales allows for the existence of these solutions (i.e., why the ratio $\frac{\tau_i}{\tau_n}$ cannot be too large or too small). While there are certain trends in the behaviors of the solutions, we did not observe any significant changes in the dynamics near bifurcation. In the case of the inhibition, as τ_i increases, the period increases (as discussed at the beginning of this section), more than doubling (from 50 to 117 ms), and the maxima of the excitatory populations increase by about 25%, while the minima of the NMDA populations decrease by $\approx 33\%$ (note that the NMDA generally peaks at or near saturation at 1, so this does not tend to change as we adjust parameters) and the maxima of the inhibitory populations decrease only very slightly, staying virtually constant (a change of $\approx 5\%$). We note that not all of these behaviors (or those described below for the case of varying τ_n) are strictly monotonic, but rather they indicate the trends. For example, the amplitude of the inhibition increases very slightly with τ_i before it decreases; however, up to two significant figures it remains at 6.8. If τ_i increases beyond the high fold, the solutions tend to change from M-OP to $(M-1)$-OP; e.g., 3-OP solutions are lost to 2-OP solutions. If τ_i decreases beyond the low fold, we get slightly different behavior. In this case, the solutions tend to change from M-OP to M-MP; i.e., one of the populations will synchronize with another one. In both cases, we see that the number of mutually OP groups decreases from M to $M-1$.

For NMDA, as τ_n increases, the period decreases (as discussed at the beginning of this section) from 77 to 63 ms, the maxima of the excitation and inhibition stay virtually constant (changing by less than 1%) and the minima of the NMDA populations increase by $\approx 50\%$ (since, of course, τ_n increases). As we can see from Fig 5D, there is a fold of limit cycles (for $N = 5$ and $N = 10$) for low τ_n; increasing τ_n does not appear to cause the loss of existence or stability of the solutions. This is as expected, since in the limit $\tau_n \to \infty$, NMDA will simply stay high (once activated) as it acts essentially as a parameter. Thus, as we explore in more detail in Appendix B, the NMDA will always outlast the downstroke of the inhibition, allowing the excitation to reactivate. If we decrease τ_n to below the fold, we again can lose the M-OP state to the M-MP state, as one of the active populations will tend to synchronize with another one.

By following the above bifurcations, we may partition $\tau_i - \tau_n$ space based on how many OP populations can be active for fixed c_{ei} (Fig 5B), or on the maximum number of OP populations that may be active for different values of c_{ei} (Fig 5C). In Fig 5 we see that we have open sets of τ_i and τ_n values that lie within physiological ranges where the network capacity is 3, matching experimental ranges of a capacity of 3 – 5 items in working memory. As we increase the number of populations, the capacity of the system remains more or less the same (Fig 5D), suggesting the capacity is only weakly dependent on the network size.

2.3.2 *Synchronous oscillations and binding*

Different types of binding occur within the context of working memory. In all examples of binding, temporal synchronization of firing may play a significant role as the underlying mechanism. There are several different ways in which our model may achieve such synchrony between different populations in the network.

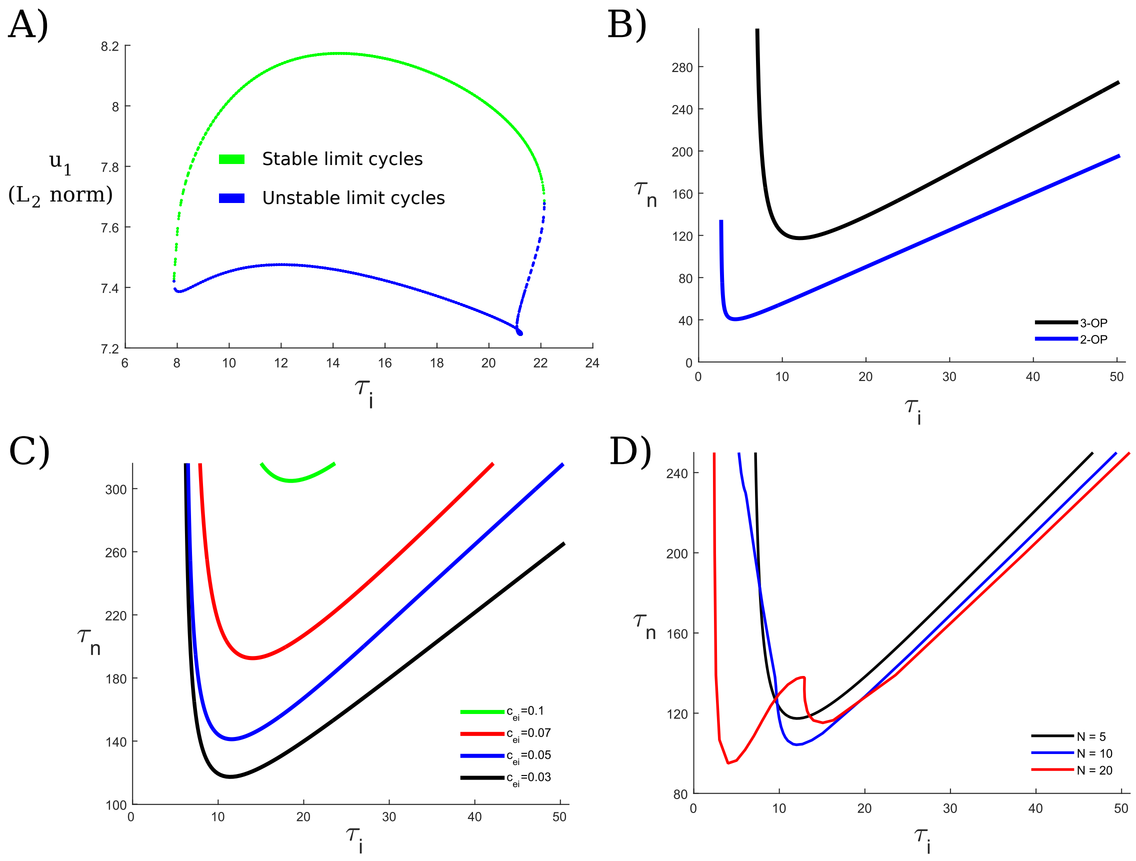

Figure 5: Stability of OP oscillations and working memory capacity. For fixed τ_n, there is a window of τ_i values for which oscillations exist. We examine the existence and stability of the active out-of-phase populations, corresponding to distinct, single-featured memoranda. (A) Here, we look at the 3-OP state for a network of $N = 5$ populations, with τ_n fixed at 144 and the mutual inhibition c_{ei} fixed at 0.03. For τ_i too large or too small, the oscillations are lost as folds of limit cycles (for larger N, they may also be lost through torus or period-doubling bifurcations). (B) By following the fold points in (A) and keeping N fixed at 5, we may examine the dependence of the oscillatory states on both timescales, τ_n and τ_i, for different OP states. Thus, we see how the capacity of the system depends on the timescales. Each curve is a curve of the limit points as shown in (A). Thus, the OP state with 2 active populations exists stably above the blue curve, and the 3-OP state exists stably above the black curve. (C) We may further examine how the capacity is affected by the strength of the mutual inhibition, c_{ei}. Here, the 3-OP state exists above each curve for different c_{ei} values as indicated. As the mutual inhibition increases, the minimum τ_n value that supports the

3-OP state within physiologically realistic synaptic timescales, but high enough to allow for the WTA state. (D) If we fix $c_{ei} = 0.03$, we may further explore how the network size N affects the 3-OP state. Overall, as N increases, the set of timescales that supports the 3-OP state does not change very much, generally increasing slightly. The bifurcation structure for $N = 20$ changes somewhat as well, so that the 3-OP state may destabilize through a torus or period-doubling bifurcation for lower τ_i values as well.

2.3.2.1 ***S dynamics*** In Fig 6, we see three such possible paths based on the pattern of inputs. Unsurprisingly, if two inactive populations, say populations 1 and 2 without loss of generality, are stimulated with similar stimulus parameters (onset time, amplitude, and duration), they will oscillate S (Fig 6A). If two populations are active OP, we may apply a stimulus to either one so that they synchronize (Fig 6B). Finally, if population 1, for example, is oscillating in isolation, population 2 may also synchronize with population 1 if given a small stimulus within a certain time window of the phase of population 1 (in particular, close to when the excitatory component has an upstroke) (Fig 6C). The latter two cases illustrate the attractive nature of S oscillations in the model, which we further confirm through numerical continuation. We note there are other ways in which populations may pairwise oscillate S in the network, some of which are explored in Sec *2.3.3* and Sec *2.3.4*.

2.3.2.2 ***Maximum S populations*** We first observe from Eq (2.1) that if all N populations are active S in a given network, they oscillate as a single population with inhibition given by a_{ei}. Thus, we might expect the maximum number of populations that can oscillate synchronously to depend more on the intrapopulation parameters, such as a_{ei}. Indeed, this solution is eventually lost as a fold of limit cycles as a_{ei} increases. However, the interpopulation coupling values, such as c_{ei}, do affect the stability of this oscillating solution. We have observed these solutions to be lost as branch points (with no stable alternate branch)

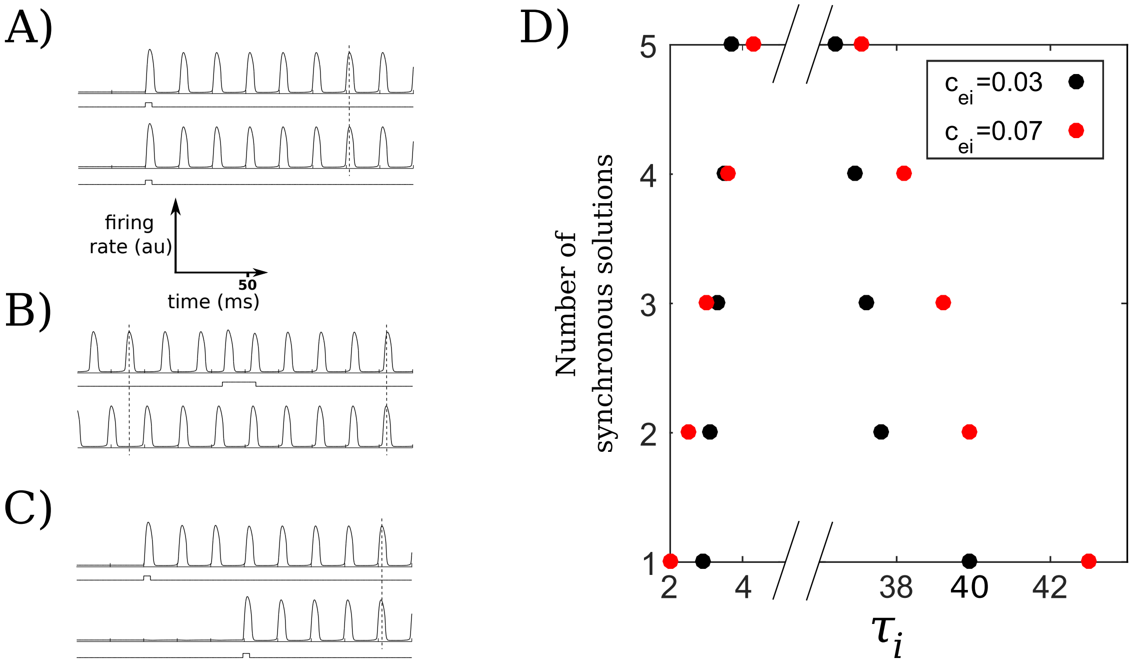

Figure 6: S dynamics and capacity. (A) Simultaneous stimuli can cause the selected populations to exhibit S oscillations. (B) If two populations are pairwise OP, selectively stimulating one can alter its relationship so that the two populations subsequently oscillate pairwise S. (C) Sequential stimuli of the right timing may also cause the selected populations to dynamically bind and oscillate S. (D) Synchronous capacity as a function of τ_i and c_{ei}. For $M = 1$, ..., N (here $N = 5$), the M-S solution is stable for all τ_i values between each respective pair of (same-colored) points. For example, the 5-S solution with $c_{ei} = 0.03$ is stable for $3.7 < \tau_i < 36.4$. The synchronous solutions are not stable outside of these intervals and are lost to different solutions, as indicated in the text.

or period doubling bifurcations. Although the types of bifurcation may change, the types of changes the parameters cause match our expectations: As c_{ei} increases, the a_{ei} value at which the solution destabilizes decreases. Increasing the c_e value correspondingly increases the a_{ei} value at which the bifurcation occurs. Furthermore, decreasing the number of active populations increases the a_{ei} values at which the stable oscillations are lost, so that, for example, with the same intrapopulation and interpopulation coupling strengths, 4 S populations remain stable for higher a_{ei} values than 5 S populations do.

For other parameters, the behavior may be somewhat more complicated. For example, consider the range of τ_i values that allows for the stable existence of M-S solutions for $M \in \{1, \dots N\}$ (of course the 1-S oscillation is just a single oscillator, SO). As we see in Fig 6D, as M increases from 1 to 5 with c_{ei} fixed at 0.03 or 0.07, the interval of τ_i that allows for the stable existence of the M-S oscillations narrows monotonically, resulting in a nest of intervals, each of which is strictly contained in the interval below it. However, the result of increasing c_{ei} from 0.03 to 0.07 varies somewhat on the number of synchronous oscillating populations. The right boundary, corresponding to the largest τ_i that supports the synchronous solution, increases with increasing c_{ei}, especially for fewer oscillating populations. The left boundary, corresponding to the smallest τ_i that supports the synchronous solution, increases for larger M ($M = 4$, 5) and decreases for smaller M ($M = 1$, 2, 3). As mentioned above, when $M = N$, the whole network oscillates as a single group, as can be seen with the appropriate substitutions in Eq (2.1); thus, while varying c_{ei} does change the stability of this oscillation, as can be seen for $M = 5$ in Fig 6D, the other characteristics, such as the amplitudes and period of oscillation, do not change at all. However, these can and do change for $M < N$; in particular, increasing c_{ei} tends to increase the amplitudes of the excitatory and inhibitory populations. We explore these changes in dynamics and explore the changes we see in Fig 6D in more detail in Appendix D.

2.3.3 ***Mixed-phase oscillations: synchronous and out-of-phase***

We have seen above in Sec *2.3.1* and Sec *2.3.2* that the network allows for both S and OP oscillatory states as attractors. When such states coexist in the network (i.e., at least two populations are pairwise OP and at least two populations are pairwise S), we say the populations oscillate MP (mixed-phase). An important result in our current work is that the model network can support a large number of attracting MP states, where the pairwise OP populations correspond to distinct items in memory and the pairwise S populations correspond to bound features of a memory.

Thus, multiple items in memory, each with multiple features, can be represented within our framework. Indeed, we found a range of combinatorial arrangements and dynamics emerge in the model network that are rich enough to map onto or underlie many of the aspects of binding. We explore some of these dynamics below, and other examples that may be relevant to feature and variable binding in Sec *2.3.4.1* and Sec *2.3.4.2*.

2.3.3.1 ***MP dynamics*** When a quiescent or active population is stimulated, it may oscillate pairwise OP or S with currently active populations. Here we see three ways whereby we can obtain MP oscillations with a selective stimulus, so that a feature is added to an item in memory.

In Fig 7A–C, there are initially 3 active OP populations (say populations 1–3). If population 3, for example, receives a stimulus, then its timing may be readjusted to synchronize with population 2. Alternatively, a quiescent population (say population 4) may be stimulated. Since the capacity of the network is reached with 3 OP populations (or really, 3 groups of mutually OP populations; that is, populations within each group oscillate S and populations in different groups oscillate mutually OP), in order for population 4 to remain active, either at least one of populations 1–3 must be quenched or at least two of populations 1–4 must synchronize. In Fig 7A, activating population 4 with a somewhat weaker stimulus quenches the activity of population 1 as might occur in forgetting, suggesting independently of the numerical analysis in Sec *2.3.1.2* that the capacity of the network is indeed 3. Further stimulating population 4 causes it to synchronize with population 2, in a similar way that occurs in Fig 6C when sequential stimuli cause two populations to oscillate S. However, since all of the populations are coupled to one another, more complicated effects may occur as well. For example, in Fig 7C, once population 4 is stimulated, it also synchronizes with population 2 as in Fig 7B; however, populations 1 and 3 are perturbed enough to synchronize as well. We note that the only differences between Fig 7B and Fig 7C are the onset times of the stimulus to population 4. More generally, the stimulus parameters determine which populations oscillate S and which do not. We also see that in both Fig 7B and Fig 7C four populations are active following the stimulus. Thus, the number of active populations depends in part on how many OP populations may be active (see Sec *2.3.1.2*) and in part on how many S populations may be active (see Sec *2.3.2.2*). We consider additional exmaples of MP states in Sec *2.3.4.1* and Sec *2.3.4.2*.

2.3.3.2 ***Two populations: effects of coupling strengths*** Having seen that our model supports both OP and S dynamics that naturally combine to allow for richer MP dynamics, we look at the reduced case of two populations to distinguish how the interpopulation coupling affects the existence of the OP and S states, as well as the case where only a single oscillator (SO) is active. We note that the stable existence of the SO state is necessary for WTA dynamics; however, the SO state may also stably exist with lower coupling values that do not allow for WTA, and so it is not sufficient for WTA dynamics. The coupling strengths are crucial in being able to maintain any of these oscillatory dynamics. We see that in the coupling-strength space in Fig 8D, only the region $*$ supports all three oscillatory states. Both SO (Fig 8B) and OP (Fig 8C) states exist stably in bounded sets, so that if either c_e or c_{ei} values become too large, the states will be lost. In contrast, there is no indication that the S state only exists in a bounded region (Fig 8A). If c_{ei} values are large enough with c_e small, the S state may lose stability (to the right of (i) in Fig 8A). However, we see from Eq (2.2) that as $c_e \to \infty$, each excitatory population only gets excitation from the complementary population, so we expect synchrony to continue to stably exist for arbitrarily large c_e values. All three oscillatory states appear to exist stably when either c_e or c_{ei} is zero and as the other value approaches zero (i.e., along one of the axes near the origin), suggesting that weak coupling is sufficient for these states to exist. In Appendix C, we explore the attracting states in the weak coupling limit.

Although the general features described above for Fig 8 match our intuitions, some of the particular features are unforeseen and may warrant further scrutiny. We now look more carefully at the regions that the S, SO, and OP states stably exist in coupling-strength

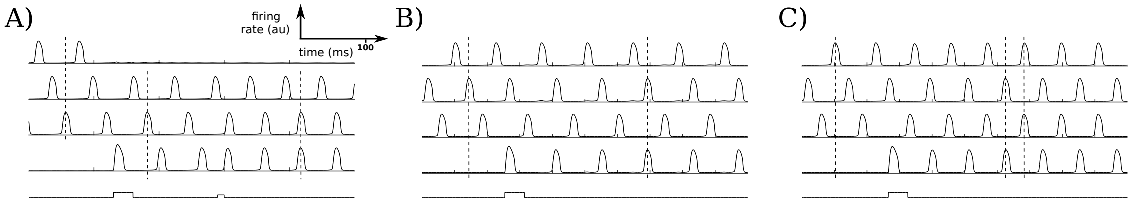

Figure 7: MP dynamics. (A) Stimulating an inactive population when the network is already at capacity (3 OP populations here) may cause one of the active populations to become quiescent, even without a strong stimulus. A subsequent stimulus produces similar behavior as in Fig 6B, so that stimulating an active population may allow it to change its relationship from pairwise OP with both other active populations to oscillate S with one of the two other active populations, and OP with the second. (B – C) Stimulating a quiescent population can cause various dynamic bindings and MP dynamics. For example, changing only the timing of the stimulus can alter the resulting patterns of synchronization. In (B), the stimulated population synchronizes with one of the three OP populations. (C) Adjusting the onset time of the stimulus may result in additional interactions, so that the network transitions from 3 OP populations to 2 OP pairs, where both populations in each pair oscillate S.

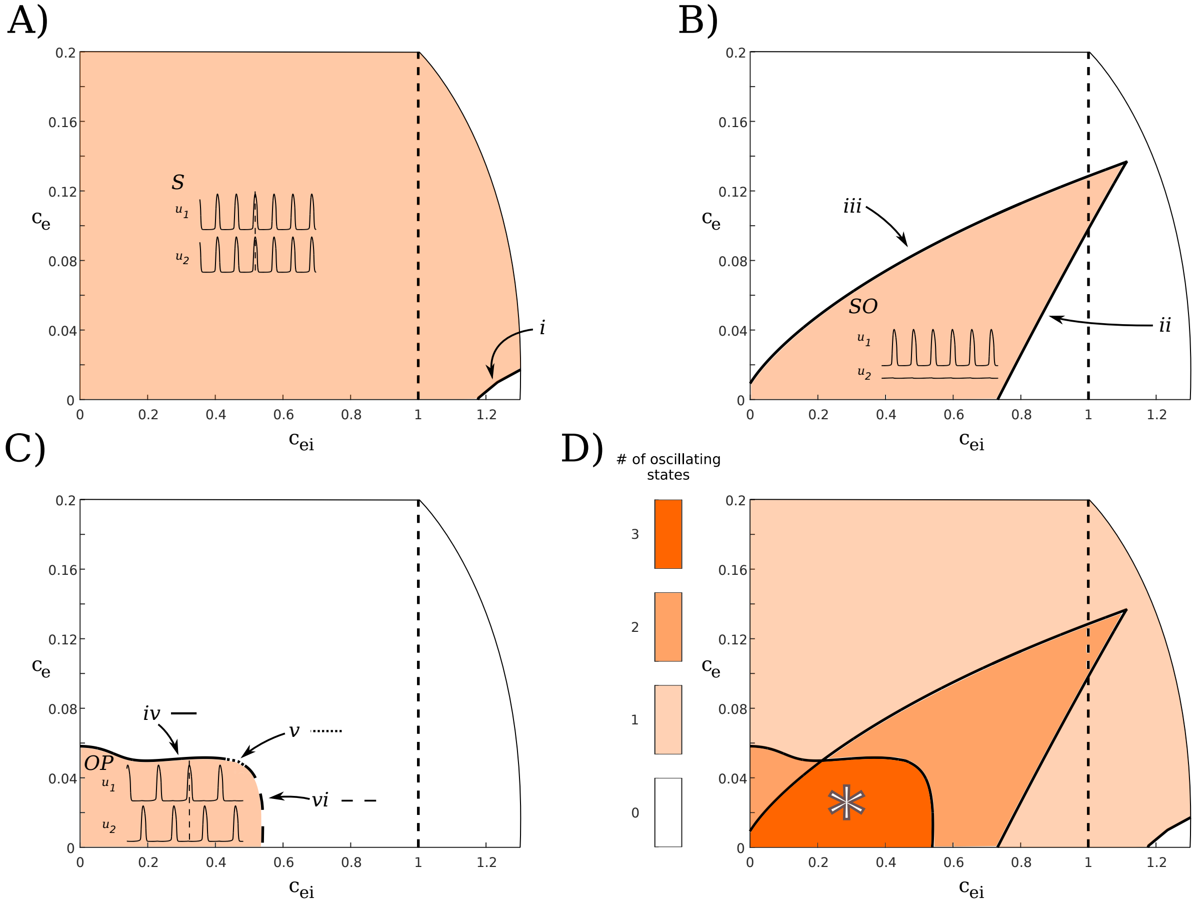

Figure 8: Stability as a function of coupling strengths for $N = 2$. The dashed line at $c_{ei} = 1$ indicates the interpopulation and intrapopulation inhibition are the same. All of the dynamics of interest exist stably when the interpopulation coupling is less than the intrapopulation coupling (to the left of the dashed line), as desired. (A–C) Colored regions indicate areas that the oscillatory states shown stably exist. Roman numerals refer to the boundary curves of the regions as indicated. (A) The S state exists to the left of (i), which is a curve of branch points of limit cycles (BPLCs). (B) The SO state exists to the left of (ii), a curve of folds of limit cycles (FLCs) (these folds are the only ones in this diagram that arise from a subcritical Hopf bifurcation) and to the right of (iii), another curve of FLCs. (C) The OP state exists below (iv), a curve of FLCs, and (v), a curve of torus bifurcations, and to the left of (vi), a curve of BPLCs. The line style for each curve is shown to the right of each roman numeral for clarity. (D) All of the regions in (A–C) superimposed. ($*$) indicates the region of interest, where all three oscillating states (S, SO, and OP) exist stably.

space in turn, describing the curves of bifurcations and the behavior of the oscillations with changing coupling strengths.

The S state (Fig 8A) exists for all relevant coupling strengths. Neither the period nor the amplitude of the oscillations change as c_e and c_{ei} change. However, if c_{ei} increases beyond (i), a curve of branch points of limit cycles (BPLCs), only the low and high steady states are stable.

The SO state (Fig 8B) lies to the right of (iii), a curve of folds of limit cycles (FLCs). As we can see in Fig 8D, the oscillatory states that are available for lower c_{ei}, to the left of (iii), are either S or, for sufficiently small c_e values, OP. The SO state is bounded on the right by (ii). Curve (ii) is also a curve of FLCs; however, unlike the other such curves in Fig 8, these folds arise from a subcritical Hopf bifurcation.

For fixed c_e and increasing c_{ei}, the SO solutions show increases in the period (from 48 to 74 ms) and the maxima of both the excitatory (increasing by $\approx 20\%$) and inhibitory ($\approx 40\%$) oscillations. These changes continue until the SO state is lost to a high fixed point. As we might expect from a subcritical Hopf (in particular, one that involves a fixed point away from zero), this change happens rapidly, with no changes before bifurcation suggestive of the loss of stability of the solution. For fixed c_{ei} and increasing c_e, the SO solutions show slight decreases in the period (from 53 to 51 ms) and the maxima of the excitatory and inhibitory populations (both decrease by $\approx 12\%$ or less).

We note that once c_{ei} increases beyond the subcritical Hopf bifurcation (curve (ii)), the system displays steady state bistability between a down state and an up state with the same inhibitory timescale as for the case of oscillations. However, a much larger input is required to switch between active populations in this case. For example, suppose $c_e = 0.001$ and population 1 is active. If $c_{ei} = 0.7$ ("after" the Hopf, so that population 1 has a large-amplitude oscillation), stimulating population 2 with a width of 50ms and an amplitude of 3 (arbitrary units) is sufficient to allow population 2 to become active and quench the activity of population 1. However, if $c_{ei} = 0.8$ ("before" the Hopf, so that population 1 is in the up state), a stimulus to population 2 with the same width requires an amplitude of greater than 36 in order for population 2 to activate, sending population 1 back to the down state. This up state is only stable for smaller c_e values; for larger c_e the only stable state of the system is S activity. We see in Fig 8D that the S state exists to the right of (ii) for low c_e values as well.

The region of definition of the OP state is more complicated. It is defined by (iv), a curve of FLCs, (v), a curve of torus bifurcations, and (vi), a curve of BPLCs. These curves do not decrease in c_e monotonically as c_{ei} increases. We see that that there is a small interval of c_e values around 0.06 where the OP state exists stably for small c_{ei} values, but is lost for higher values (e.g., c_{ei} not quite 0.2). Surprisingly, once the OP state is lost here, we initially can only get the S state. As c_{ei} increases further, we may also obtain the SO state. Perhaps the most unanticipated feature of the OP region occurs near the local minimum of (iv). There is a very small c_e interval, near $c_e = 0.05$, where the OP state is lost (to a region where S and, for larger c_{ei}, SO both exist, as seen in Fig 8D) and then regained as c_{ei} increases, before finally being lost again for large c_{ei}. We explore the behaviors of the OP state further in Appendix D, including changes in the dynamics that occur with increasing c_e or c_{ei} that may lead to the bifurcations we observe.

2.3.3.3 ***Weak coupling*** Surprisingly, both S and OP oscillations are stable as the coupling strengths c_e and c_{ei} approach 0 with $N = 2$, as we saw above in Fig 8. Although we have chosen coupling strengths that allow for stronger interactions between populations (allowing for, e.g., WTA and WTS dynamics), we may gain some insight into our system by examining the limit of weak coupling. In this case we first assume that the populations are oscillating.

In Appendix C, we outline how we may use weak coupling theory to reduce our system of N coupled populations from a $(3 \cdot N)$-dimensional system to an $(N-1)$-dimensional system involving the phase differences. Briefly, we restrict our attention to weakly-coupled oscillatory populations. Since each 3-dimensional population (involving u, v, and n variables) is assumed to be on a limit cycle, its state can be referenced by its phase on that limit cycle, θ_j, where j is the population index. We can further reduce the number of dimensions by one by referencing each population's phase relative to θ_1, the first population's phase. Thus, we are left with *phase differences* ψ_j. Since $\psi_1 \equiv 0$, we can examine the attractor structure of three weakly-coupled populations in the plane.

With just EI or IE coupling, we can obtain all three relevant states of interest (OP, S, and MP). In Fig 9, we see the basins of attraction with only c_{ei} coupling (cf. Fig 2 in Horn and Opher 1996 [80]). Adding EE to EI coupling ($c_e = 0.1\,c_{ei}$) predictably increases the basin of attraction of the synchronous state. Based on the geometry we see in Fig 9, we might also expect transitions between regions that share boundary curves to be easy to realize (e.g., the S and MP regions), and transitions between regions that only share isolated boundary points (the S and OP regions) to be more difficult to realize, if even possible. In fact, we found just such a correspondence when we more thoroughly examined accessible states (see Sec *2.3.4.3*).

2.3.4 *Biological considerations and applications*

In Sec *2.3.3*, we showed that our model is able to support complex and stable oscillatory modes that may be relevant to different scenarios in working memory. Here, we explore further aspects of these oscillations that may be relevant to cognitive processes and empirical findings in working memory tasks. For example, in working memory, rapid transitions between activated states must occur in order to establish and adjust the relationships of items that are being held in memory. Thus, we describe how the patterns of synchronization that emerge from the model are rich enough to accommodate many aspects of the current understanding of both feature and variable binding from a working memory standpoint. We then examine in greater detail what transitions are in fact possible within the model. Finally, we describe the frequency relationships that naturally emerge in the model and how these compare to experimental findings of power spectra dynamics in large-scale recordings during working memory tasks.

2.3.4.1 ***Feature binding*** Dynamic binding is required in working memory in order to quickly associate and dissociate different elements together [81], such as different features of an item in the environment. In the context of our model, this means that items must be able to be both synchronized and desynchronized in response to different stimuli. We have seen above in Fig 6 how synchronization may occur via either simultaneous (Fig 6A) or

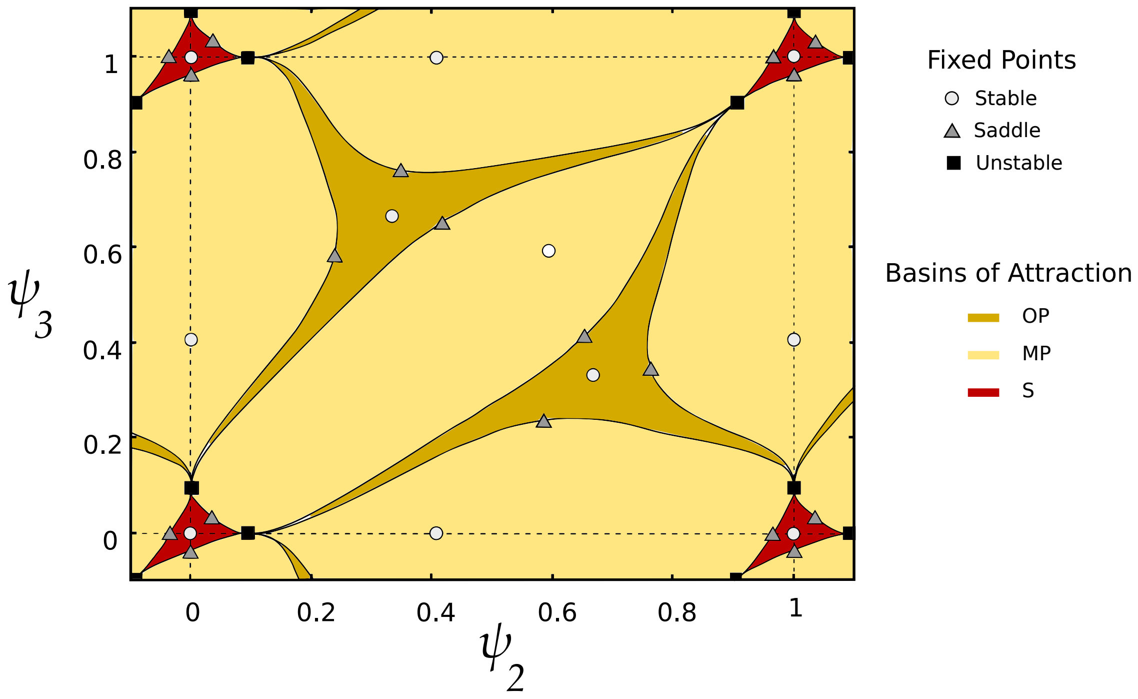

Figure 9: Basins of attraction for the case of weak EI coupling with 3 active populations. When three populations are active (note, ψ_i is the phase of population i relative to θ_1), OP, S, and MP states all have open sets as basins of attraction, as we found in the full model. These basins are defined by the stable manifolds of saddle points, as shown.

sequential (Fig 6B and 6C) selective stimuli. Such synchronization allows different features to bind together into one item in memory. In Fig 10, we see two situations in which the patterns of synchrony change as the associations between different features change. For example, if a red traffic light turns green (Fig 10A), *red light* can unbind from *stoplight* (and indeed may become quiescent, as in this example), while *green light* then binds to *stoplight* as the two associated neuronal populations synchronize. Alternatively, an observer may perceive the static environment in a unified manner, so that the car at the red light and the street, for example, are bound together (Fig 10B). Once the light turns green, the car begins to move, so that *car* and *street*, for example, unbind and are represented as distinct items in memory. Thus, attentional mechanisms or environmental changes may act to change the patterns of synchronization, and thus of the binding of different items in working memory, by stimulating either quiescent or already-active populations.

2.3.4.2 ***Variable binding*** The combinatorial dynamics emerging within the model are rich enough to produce variable binding as is characteristic, for example, of language and abstract reasoning within the context of working memory. Below, we illustrate some simple examples from the model consistent with the most expected and widespread approaches to variable binding and language (e.g., the SHRUTI inference network [82]). These utilize temporal phase binding and represent simple predicate calculus rules. Consider the predicate calculus rule

$$\text{purchases}(x,\, y) \Longrightarrow \text{owns}(x,\, y),$$

where x and y represent variables in the above rule, and a Boolean query (for example, as presented in Feldman [37]), such as “Does Tom (x) own a book (y)”. In most common inference network models, for example, SHRUTI [83], separate clock phases are assigned to the pairings of *Tom* with *owns* and *book* with *owns*. *Tom* here represents a variable agent who could come to own something (a *book*, for example) through *purchases*. There are other possibilities, of course, and they could all be linked or synchronized with the *owns* relation. We note that while a mediator circuit is implicated in the SHRUTI inference network, here the relevant association may occur in a single step as a result of synchronous firing due

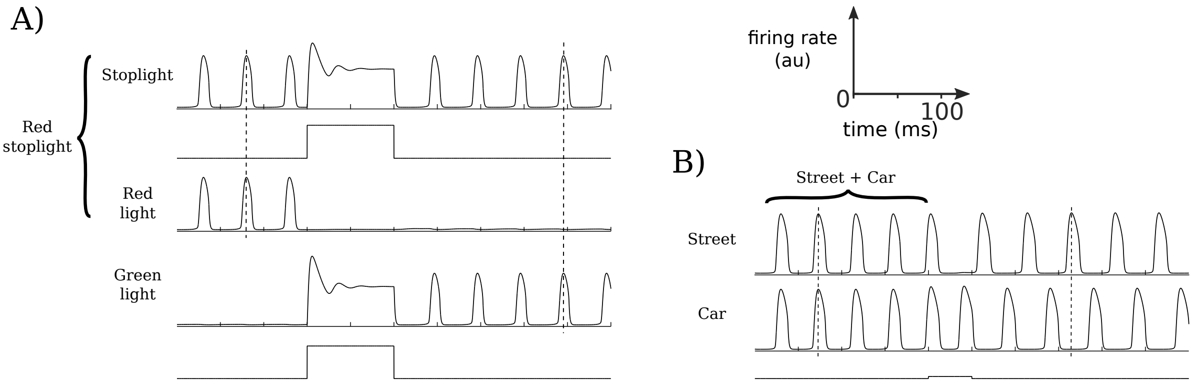

Figure 10: Feature binding examples. For simplicity, memoranda that may require several different bound features are represented as a single oscillating population (e.g., a street). (A) Feature binding can be achieved through the synchronous firing of the individual features as in the left half of (A). A new input causes an unbinding and a new binding of an object with a modified feature. For example, the initial binding might represent a *red stoplight.* The light changes to green, causing the *stoplight* to bind with *green light.* (B) Feature binding in which a new input (e.g., focus) results in the decoupling of a single memorandum (e.g., some background such as a *street* and stationary *car*). For example, the *car* is observed to begin moving, and so is now perceived as a separate object from the *street.*

to temporal associations and the pattern of input, as shown in Fig 11. Once the query mentioned above, "Does Tom (x) own a book (y)," is made, two populations become active (shown in Fig 11 as the already-active populations): one for a node for *owns* (e.g., the active population in line 1 of Fig 11A), and one for a node for *Tom* (as an agent node; e.g., the active population in line 2). The active populations of *Tom* and *owns* then become associated and paired as a result of the activation of a node or population representing *purchases* in working memory (active population in line 4 of Fig 11A). Then, over several clock cycles "Tom owns a book" becomes established, or possibly instantly as here with the activation of the *purchases* node (i.e., population). Additional populations could become activated and linked via synchronization, such as "Tom purchases *The Awakening*" where *The Awakening* would become synchronized with a *book* node. We note that, whereas *Tom* requires two populations to be active in this example in order to form distinct bindings with both *purchases* and *owns*, different, $n : m$ locking ratios or aperiodic oscillatory dynamics could allow for such unambiguous bindings while just one population is maintained active for *Tom*, as we touch on at the end of Sec *2.4*.

We can also consider how the combinatorics allowed through the present model's dynamics may apply in language by illustrating how they may facilitate the presence of grammars. We note that the predicate-argument structure used above (e.g., owns(x, y)) fits well within some dependency grammars. However, the model is agnostic to any particular type of grammar, and we now consider an example as might be implemented within a phrase structure grammar, illustrated in Fig 11B and 11C. The words are represented by the activation of populations (presumably stored in the structure of networks in long-term memory) and become synchronized with the appropriate activated nodes. For clarity, we have simplified the example so that the nodes, and therefore the bindings of words to nodes, are not displayed. For example, a determiner node is first activated and then bound to *the*, but in Fig 11B we only show the activity of the population corresponding to *the*. A noun node is then activated, which is bound to the variable *woman*. A verb node is activated next, which is bound to *writes*. Finally a second noun node is activated and then bound to *music*. This results in the binding of the determiner node and noun node of the subject into a determinate noun phrase (displaying the phrase structure rule D + N → DNP), and within a single clock cycle

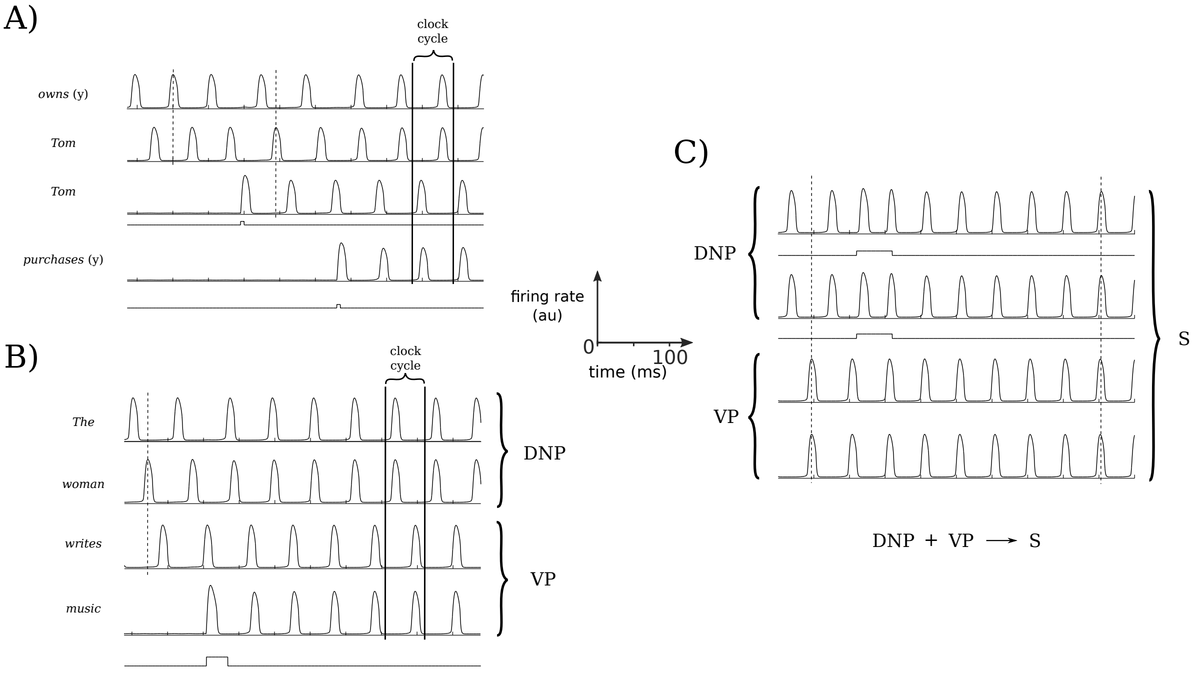

Figure 11: Variable binding and sentence construction examples. Specific sequential inputs result in a cascade of bindings that emerge, following one another within a single clock cycle. (A) A fragment of the working memory system demonstrating how an inference emerges and is established in working memory. Here, the predicate calculus rule examined is purchases$(x,\ y) \implies$ owns$(x,\ y)$ (from an example in Feldman [37]). After a query is made ("Does Tom own *The Awakening*?"), a statement is provided, "Tom purchased *The Awakening*". This statement first activates a second instantiation of *Tom*, and then the verb *purchases*(y), causing *owns*(y) and *Tom* to synchronize so that the inference is made. (B) An illustration of that same combinatorial structure is shown as it could apply towards a mechanistic realization of a phrase structure grammar. This could be based upon an already established or innate structure in the cortex. Upon reading the sentence, the words are sequentially input and the appropriate nodes are activated and bound in working memory, forming a determinate noun phrase (DNP) and a verb phrase (VP). For clarity, bindings between nodes and variables (i.e., words here) are not explicitly illustrated. (C) A final binding occurs, as the components of the DNP receive selective and equal simultaneous stimuli, binding all of the components together to form a sentence (S).

the dynamic binding of the verb with the direct object to form a verb phrase (V + N → VP). These states are activated sequentially within a single clock cycle, and thus could be recognized as a grammatical sentence maintained in working memory based on the phrase structure rule that a DNP followed by a VP produces a grammatical sentence S. Alternatively, the sequential DNP and VP representations could be input with a subsequent final synchronization of the DNP and VP taking place to produce S, and recognized as a grammatical sentence that is maintained in working memory, as illustrated in Fig 11C.

Ungrammatical sentences could be recognized when bindings occur that do not correspond to valid production rules. We do not consider here the specific mechanism or details by which the particular variables become bound, but rather show how they can dynamically emerge within working memory representations. In principle this could arise via some mapping and closeness of the associations in that cortical map, "hardwired" architectural associations in long-term memory, or some combination.

2.3.4.3 ***Accessible operations*** We have shown that the dynamics of our model map onto a number of working memory and binding examples. Indeed, it appears that the relevant attractor patterns are exactly the combinatorial possibilities, limited by the network size and the number of MP patterns available. Beyond ascertaining what states are available to the network, we are further interested in how the network can transition from one pattern to another. For example, in Fig 4B we see that if a third, inactive population is stimulated, the network can transition from 2 OP populations to 3 OP populations. We will consider this to be an operation available to the network; i.e., the operation of transitioning from 2 OP populations to 3 OP populations by providing a stimulus of the right strength and timing (see Sec *2.2* for further protocol detail). Different patterns may require a different number of operations to attain from a given starting pattern. We refer to a transition that requires n selective, sequential stimuli to instantiate as an nth-order operation. We may assess the capabilities, and thus to some degree the plausibility, of the model more systematically by delimiting the available operations. Determining whether a pattern is accessible through a first-order or a higher-order operation may also allow for predictions to be made from the model, such as the timings involved in certain cognitive processes. We restrict the number of active populations to either two (diads) or three (triads). Our parameter choices are as above, so that WTA is always an accessible state. Thus, if we begin with a diad or triad, the only resultant patterns of interest are also diads and triads.

Fig 12 shows what operations related to diads and triads are available to the network, under the limitations described above and in Sec *2.2*. In each gray circle, 1st-order operations are displayed; we may determine available higher-order operations by stringing together sequences of 1st-order operations. For example, we may transition from an OP diad to an S triad as a 2nd-order operation by stimulating an inactive population so that the network first transitions to an OP triad, and then stimulating an active population so that the network then transitions to an S triad. In particular, examining Fig 12B and 12C reveals that the network may transition from any diad or triad to any other (through transitions involving only other diads and triads) using no higher than a 3rd-order operation (i.e., no more than 3 selective, sequential excitatory stimuli are required).

Since the stimulus is excitatory, the stimulated population will remain active. Thus, every beginning pattern will have at least two resultant patterns that will not be accessible for a given selected population that receives a stimulus (e.g., beginning with O and stimulating population 1 will ensure that s_{23} and o_{23} are not accessible – see Fig 12A for a notation key). By changing the population that receives the stimulus, other patterns become available in a way that is obvious by inspecting Fig 12. For example, we see in Fig 12B that an MP triad and OP diads and triads can transition to most of the remaining diads and triads: Discounting order, there are 5 possible triads and 6 possible diads; disregarding which population receives the stimulus, we see that both an MP and an OP triad can directly transition to 6 of the 10 patterns that are different from themselves. By contrast, the S populations have fewer populations accessible by 1st-order operations. This is in part an outcome of maintaining a uniform connectivity. In order to differentially affect two S populations, one of them must receive a stimulus, as any other population will affect the two S populations symmetrically. So, for example, it is as expected that the only accessible patterns for 3 S populations are the ones consisting of 3 MP populations where the selectively stimulated population oscillates out-of-phase with the remaining two. The results for s_{12} as shown are likewise entirely expected.

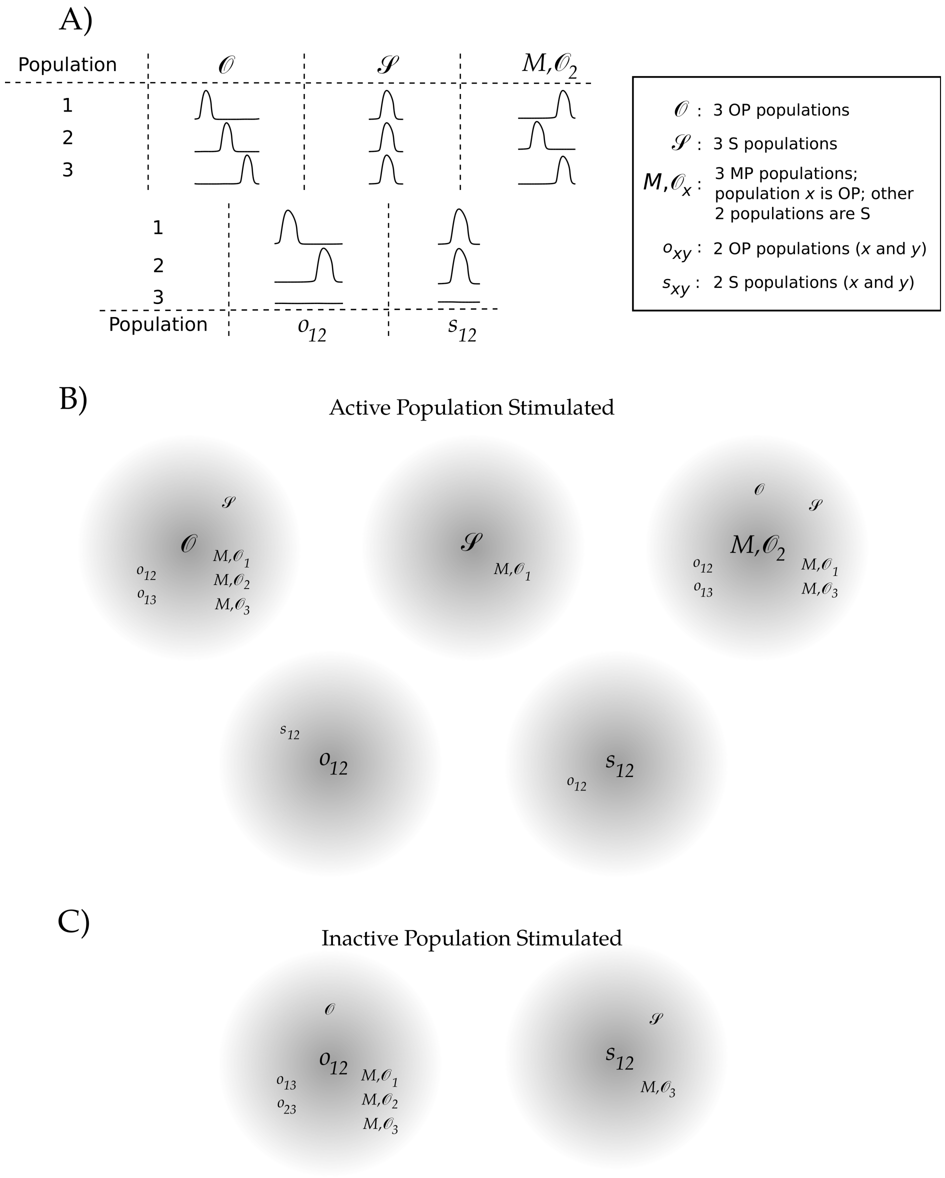

Figure 12: Accessible diad and triad operations. (A) Using $N = 5$, a stimulus is given to each of the activity patterns shown. For (B) and (C), the starting activity pattern is indicated in the center of the gray circles. For (B), one of the already-active populations, population 1, was stimulated, whereas an inactive population, population 3, was stimulated for (C). The observed resultant activity patterns are indicated in smaller text around the circumference of the gray circles.

One of the patterns that is difficult to directly obtain from a diad or triad is an S diad (e.g., s_{12}). We were only able to obtain this pattern by stimulating one of the active populations in an OP diad (e.g., o_{12}). Thus, going from any other pattern to an S diad requires at least a second order operation. For example, transitioning from an S triad to an S diad is at least a third order operation: With the first stimulus the network may transition to an MP triad, with the second the network may transition to an OP diad, and with the third the network may finally transition to an S diad. We note that there may be multiple paths that allow one pattern to evolve to another pattern. For example, an alternative route from the S triad to diad would be through a WTA scenario: A strong first stimulus can cause the selected population to quench the activity of the other populations, and a second stimulus to a nonactive population could cause the network to transition to an S diad.

These results show that there are a number of options to get from one pattern to another for two and three active populations, even with strong limitations on the network architecture and the stimulus protocol. For the operations that take triads to triads (as shown in Fig 12B, top), we note that the accessible operations are almost exactly what we would have predicted from the weak coupling analysis. That is, as we mention in Sec *2.3.3.3*, we expect transitions between basins of attraction with boundaries given as curves in Fig 9 (i.e., between OP and MP basins and between MP and S basins) to be easier to realize, and transitions between basins of attraction with boundaries given as points (as in the unstable fixed points separating OP and S basins) to be more difficult to realize. Indeed, we found it easy to transition between OP and MP states and between S and MP states, and found it more difficult to transition between OP and S states (note that we were unable to transition from S to O within the constraints given in Sec *2.2*, while there is only a very narrow 7ms window that allows us to transition from O to S, just within our stated protocol). It may be of interest in future work to loosen some of the above restrictions (for example, examining heterogeneous networks or having the stimulus also drive the inhibitory components), and to further quantify the level of difficulty for a particular transition (e.g., some transitions are much less dependent on the stimulus parameters than others).

2.3.4.4 ***Frequencies*** In constructing our model network, we chose parameters that were biologically plausible and further found constraints on these parameters that allowed OP, S, and MP modes and transitions between them that are relevant to working memory situations. While we did not tune parameters in order to fit certain frequency bands that have been found to be of potential relevance to working memory and binding, we indeed observed some correspondence between these and the frequency relationships in the model network. For example, if we have populations that oscillate OP, then as each population becomes active, the oscillation frequency of the network as a whole (or, equivalently, of each individual population) decreases, as illustrated in Fig 13. If the number of active OP populations remains constant, the frequency of the network oscillations also decreases with increasing network size. Thus, for example, if there are three active populations with OP dynamics, each population oscillates at just over 15 Hz for a network with $N = 3$, and at approximately 13 Hz for a network with $N = 20$. We see in Fig 13 that if either N or the number of active populations corresponding to distinct memoranda increases, the frequency of oscillations for any individual population trends downward towards the alpha band.

Since we are considering OP dynamics, we also note that the period between successive peaks in the case when, for example, three populations are active, is one third the overall period of the network. We may refer to the associated measure as the "interpopulation period/frequency". As an example, let us consider a network size of $N = 20$ with three active OP populations. As mentioned above, the network oscillation, and thus the oscillation of any particular population, is 13 Hz. However, since three populations are active OP, the interpopulation frequency is 39 Hz. Depending on the distances between these active populations of neurons and the spatial resolution of the measurement in an experimental setting, then, increased activity near the alpha and/or gamma band may be detected. Greater numbers of populations that oscillate S may cause increased power near the alpha spectrum, since the interpopulation frequency would not increase. Increases in power in gamma frequencies and in frequencies near alpha are both consistent with neurophysiological experiments that have reported increases in gamma or alpha contributions with working memory tasks and increasing working memory load [33, 84, 68, 35, 54, 85, 86, 87, 88]. In addition, increased alpha-band oscillations have been reported to play an important role in mental processes

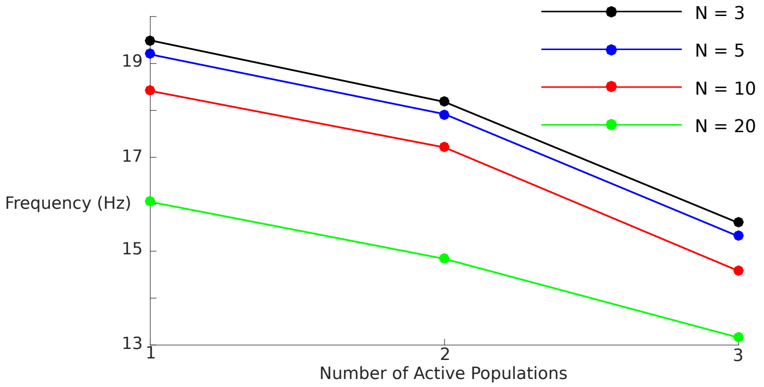

Figure 13: Frequency of population oscillation for different network sizes N. We examine the effect of increasing both system size and number of active OP populations on the oscillation frequency. In both cases, the frequency decreases monotonically. The frequency of interest may also be that based on the time between peaks of active populations (e.g., the time between the peak firing rates of population 1 and population 2 in a given network). This "interpopulation frequency" may be obtained by multiplying the given population by the number of active populations. For example, the interpopulation frequency for a network size of 20 when 2 populations are active would be approximately 30 Hz.

related to attention and memory [92, 93, 94, 89, 90, 91].

2.4 DISCUSSION

We presented an oscillatory firing rate network model that can represent both distinct and bound items in working memory. The oscillations provide a mechanism to separate or bind these items by utilizing the phase information of the oscillations, resulting in groupings of synchronous and asynchronous populations that represent multi-item, multi-feature memories. By framing our mean-field model within a dynamical systems context, we were able to study how the oscillatory solutions arise as functions of key network parameters, such as the coupling strengths and synaptic timescales. We were also able to rapidly simulate various scenarios of interest, which allowed us to explore different oscillatory states and transitions between them that may be relevant to working memory and binding.

Beginning with a spiking network, we derived a simple mean-field model. In addition to the fast AMPA and slow GABA synapses that are normally modeled, we included slower excitatory NMDA synapses, which have been implicated in the persistent activation of neurons during information retention in working memory tasks. The NMDA synapses allow for bistability in our model, so that only transient stimuli, applied to the excitatory components, are required to trigger the active oscillatory states, consistent with experimental findings and other modeling efforts that have employed NMDA. While others have examined oscillatory models in working memory and binding [95, 96, 97, 36, 98, 99, 79, 100, 101, 102, 103, 106, 104, 15, 105, 80], to our knowledge none have employed NMDA as a mechanism for bistability, used numerical continuation to define regions of parameter space that allow for the oscillations of interest, or explored in detail the combinatorially rich oscillatory states and transitions between them.

We found attracting states involving synchronous (S) and out-of-phase (OP) oscillations, as well mixed-phase (MP) states where some populations oscillated S and others OP. Numerical continuation defined regions in parameter space where they exist stably. The mutual inhibition between populations (c_{ei}) both facilitated OP states through competition and also limited the number of populations that could oscillate. We found a wide range of biologically plausible parameter values that allow three populations to oscillate mutually OP, in line with the $3-5$ item working memory capacity found across different modalities[37, 107, 55]. While c_{ei} strongly affected the number of populations that could oscillate OP, it had only a weak effect on the number of populations that could oscillate S. The model provides a natural explanation for the capacity of working memory, apart from the system's bifurcations discussed above in Sec *2.3.1*, in that the excitatory coupling c_e can greatly increase the basin of attraction of synchronous states, as we show explicitly in the case of weak coupling. Thus, two populations may only get so close before they will tend towards synchrony, implying there may only be a finite number of OP populations. However, for c_e too large, the network is only attracted to a bulk oscillatory state. While synchrony is desirable in some binding situations, too much synchrony can be pathological, so it is necessary to maintain low-level excitatory connections between distinct populations.

We explored a number of different scenarios by adjusting the stimuli parameters. We found by stimulating either inactive or active populations that the network could rapidly transition between accessible oscillatory states. Thus, depending on the strength and timing of the stimuli, populations could quickly synchronize, desynchronize, or become quiescent. All of these operations are necessary for binding and cognitive processes within working memory. We found many different sets of dynamics that are relevant to binding, and the flexibility of the network lends itself to linguistic demands. Thus, our network may provide a framework to realize certain ideas in grammars, sentence construction, and simple predicate calculus. In particular, the attracting S, OP, and MP states and the response speeds to selective stimuli allow our network to provide a versatile underlying skeleton to realize such binding scenarios as outlined in a connectionist setting in, for example, Shastri et al. [82]

More generally, the oscillatory activity provides basic properties by which executive processes can direct ongoing cognitive operations through the control of the contents of declarative or procedural working memory. The dynamics meet the potentially opposing demands of this control by providing a mechanism for quickly establishing and maintaining new structures or representations by both forming strong bindings that are resistant to interference and retaining the ability to rapidly dissolve those bindings in order to update the contents of working memory, or remove content from working memory that is no longer relevant. Thus, a primary result in this work is that binding via synchronization between populations that represent working memory elements, the number of arrangements of which appears limited only by the number of permutations of active populations, may be rapidly and stably formed as well as dissolved to form new structures from other populations that represent elements in long-term memory. However, while all arrangements of the active populations appear possible, different numbers of operations (e.g., activations or deactivations) are necessary to reach particular states from each other. This could be a factor involved in different lag times in the performance of different types of working memory tasks, as observed in numerous studies [108, 109, 62, 110, 111].

We note that in our model, working memory and binding depend on the onset time or phase presentation of "stimuli". However, these stimuli do not necessarily represent direct stimuli, but rather could be inferred to be filtered through some control circuitry not explicitly addressed in the present model. Nonetheless, our model does require precise phase timing control of selective stimuli that can affect whether items are perceived as distinct or bound. The existence of such precise timing control is essentially a prediction of the model. However, there is some evidence that indicates such finely tuned temporal control is plausible. As we mentioned in Sec *2.2*, some pyramidal neurons and interneurons lock to two different phases of a gamma oscillation, allowing as little as 6–7ms temporal resolution [76], and synchronization between neurons has been shown to occur within a time window of around 10ms [77]. Even for direct external inputs in working memory and binding, it has been shown that sequential inputs of stimuli, in conjunction with other stimulus characteristics (e.g., spatial characteristics), lead to increased misbinding if presented with a sufficiently short duration. While the functioning of visual working memory has been shown to be independent of the cortical spacing between memoranda [112, 113], studies of multidimensional perceptual interaction have shown that presenting stimuli in short time windows results in different perceptions and bindings. For example, the pitch and loudness of a 200ms tone is experienced differently from a 50ms tone of the same pitch or loudness [114, 115]. Furthermore, a significant amount of work has shown that binding and differentiation of distinct items in memory depend in part on timings associated with the gamma band [33, 68, 70, 78, 77, 34].

These considerations make the existence of such short phase precision plausible.

Oscillatory models have been developed both in the context of working memory [95, 15, 97, 36] and of binding [98, 99, 79, 100], and models – often with an eye towards image processing – have employed the distinction between bound and distinct objects as synchronous or asynchronous oscillations [79, 100, 101, 102, 103, 106]. These models tend to be spiking networks, appeal to cross-frequency coupling (e.g., theta-gamma codings), provide unrealistic connections (e.g., delayed self-inhibition for excitatory elements), use delays or constant inputs to produce persistent oscillatory activity, or employ structured architectures (e.g., using Hopfield networks, Hebbian rules, or pre-wired assemblies).

The line of work that is the closest to the present study also utilized Wilson-Cowan-type networks [104, 15, 96, 105, 80]. By incorporating common inhibition, dynamic thresholds, and sustained inputs, they could obtain asynchronous oscillations. The dynamic thresholds (which could instead be interpreted as linear inhibitory neurons) induced oscillatory activity, global (nonlinear) inhibition allowed for competitive dynamics that led to asynchronous oscillations, and continuous inputs kept the selected populations active (or quiescent at or near a low fixed point for negative-valued inputs). In Horn and Usher [15] and Horn and Opher [105] in particular, correlated noise was added to the inputs to obtain synchronous oscillations (uncorrelated noise resulted in asynchronous oscillations). In Horn and Opher [80], an explanation that may be relevant to the network we consider was provided for the limit to the number of active asynchronous populations in terms of the subharmonic solutions that could be obtained by driving a single oscillator at greater frequencies. However, the mechanisms our model employs for oscillatory dynamics, sustained activity, and competitive asynchronous and synchronous oscillations are all distinct from this interesting series of papers.

Indeed, our model differs in important aspects from all of the above-cited body of work. We draw on evidence that implicates NMDA in the persistent activity in neurons associated with working memory, incorporating simplified NMDA synapses to produce persistent activation [29, 30, 71, 28, 31]. Thus, transient inputs result in sustained activity, in line with neurophysiological studies. Our network has strong local and weaker global inhibition and excitation with simple uniform all-to-all connectivity that, importantly, allows any of the populations to be in- or out-of-phase with each other. Finally, we consider the ensemble activity through a mean-field model that we develop from a spiking quadratic integrate-and-fire network. On the one hand, this allows us to relate the mean-field dynamics to those of the underlying spiking network. On the other hand, the mean-field model has the advantage of being posed in a more mathematically tractable form than spiking models, allowing us to analyze the dependence of states on different network parameters through numerical continuation methods. The reduced computational requirements of the mean-field network also free us to explore more of the model's rich dynamical behaviors, and we have included several examples of interesting and relevant working memory and binding activities.

Increased memory loads during working memory tasks have been associated with increases in the spectral power in both gamma and the lower (e.g., alpha) frequency bands in, for example, EEG and MEG recordings [33, 84, 68, 35, 54, 85, 86, 87, 88]. Neuronal networks may select for (coherent) oscillatory signals in the gamma range in particular [116]. In our model, the frequency of the oscillations depends both on system parameters and network-level interactions. For most parameter values, the peak-to-peak period of oscillations within a given working memory cycle corresponds to either gamma or beta band oscillations. As more populations become active OP, however, while the overall peak-to-peak frequency remains in the higher frequency bands (gamma or high beta) in a working memory cycle, the peak-to-peak frequency of each separately active population firing within the cycle decreases towards lower bands. As more populations become active S, they may increase the contribution of either lower or higher frequencies. Thus, depending on the spatial distribution of the populations active in the working memory cycle, either gamma power may increase or else a "downshift" in measured frequencies (e.g., an increase in activity in low beta or near the alpha band) may be expected with increasing working memory load.

We focused on the simplest cases of uniform all-to-all coupling and periodic oscillators, allowing for clear network state classification. It would be interesting to explore more realistic heterogeneous coupling, including distance-based. Preliminary results have shown interesting phase relationships can develop, such as 2:1 frequency locking. Placing the oscillators into parameter regimes that allow for quasiperiodic or chaotic regimes (as seen to exist in E-I Wilson-Cowan networks in [117]) may also allow for even richer phase relationships as in the spiking networks in Raffone and van Leeuwen [100]. For example, a population may be in some sense partially synchronous with several other populations so that a feature (such as a color) could be associated with several distinct objects that themselves oscillate asynchronously.

3.0 TEMPORALLY AND SPATIALLY OSCILLATORY NEURONAL RESPONSES TO STATIC, SPATIALLY VARYING STIMULI

3.1 INTRODUCTION

Populations of neurons comprise excitable media that can support a rich variety of spontaneous and stimulus-induced spatiotemporal dynamics, including distributed oscillations of various frequencies [67, 66], waves [11], and localized spatial structures [121]. Some particularly interesting types of responses that can occur are *spatiotemporal resonances*. For example, certain spatially uniform, temporally-varying stimuli that oscillate in a narrow frequency band can cause spatial structures in the neuronal responses that do not exist in the stimuli, an illusion known as flicker phosphenes [122]. Conversely, there is evidence for spatial resonances that can produce temporal structures in the neuronal respones that do not exist in the stimuli when the static stimuli have dominant wavenumbers in a narrow frequency band. The two major examples that have been known and studied for decades are pattern-sensitive epilepsy (PnSE) and cases of so-called visual discomfort [123, 48, 47, 124, 42]. PnSE is a subset of photosensitive epilepsy. The most common trigger of seizures for those who suffer from photosensitive epilepsy involves diffuse flickering light; in PnSE, seizures may be brought on by static images with spatial modes within a narrow spatial frequency of around 3–5 cycles per degree (cpd) [47]. Even individuals who do not experience seizures in response to such images may still experience headaches, dizziness, illusions, general discomfort, and other symptoms when shown such images [42]. Both of these phenomena have been shown to correlate with increased neural oscillations, particularly in the gamma band in primary visual cortex (V1) in humans [48]. The co-occurrence of discomfort in some individuals and seizures in others to the same type of visual stimuli suggests a common underlying cause.

Indeed, it has been known that even a normal, healthy response to stripes involves a resonant response: humans have a much higher contrast sensitivity to stripes of around 3–5 cpd, exhibiting a lower contrast detection threshold for such stimuli [45, 46]. Thus, all three of these phenomena show almost exactly the same resonance patterns, as strikingly shown in Fig 4 in [47].

Earlier work by Rule et al. [17] described and analyzed a neural field model that produces spatiotemporally resonant behaviors to model flicker phosphenes. We use a similar neural field approach in order to model the spatiotemporally resonant activity observed in the above motivating examples. We set up a spatially-extended, two-population Wilson-Cowan model to which we provide temporally static, spatially varying stimuli and study how the network's responses depend on the stimulus wavenumber and amplitude. We hypothesize that the relevant parts of the brain, such as V1, may be somewhat near (and, indeed, nearer in the case of PnSE patients) to a so-called Turing-Hopf bifurcation, a bifurcation in which a homogeneous steady state can lose stability to spatially and temporally oscillatory patterns. By poising the system near such a bifurcation, we find parameters such that the network is sensitive to a narrow range of stimulus wavenumbers, exhibiting spatially heterogeneous temporal patterns to these stimuli even with low applied amplitudes, as we would expect in both PnSE and visual discomfort. Additionally, by increasing the excitability of the network, we find that partial-field stimuli can trigger full-network oscillatory activity, akin to the spread of a seizure in PnSE. Finally, we find that for the parameters that we use for our study, the stimulus amplitudes must increase beyond some nonzero threshold value before the network evolves into spatiotemporal patterns. For stimulus amplitudes below this threshold value, the network only evolves to spatially heterogeneous steady states, regardless of the stimulus wavenumber, whose spatial patterns mirror the applied stimulus pattern. However, for a fixed stimulus amplitude below this threshold value, we find that when we sample the maximum values of the excitatory populations in the stable steady state for different wavenumbers, we find a similar resonance pattern as described above for spatiotemporal pattern formation. In this case, the same wavenumbers that require the lowest stimulus amplitudes to trigger spatiotemporal patterns are the same wavenumbers that correspond to the largest maximum activity values of the excitatory populations. We hypothesize that this

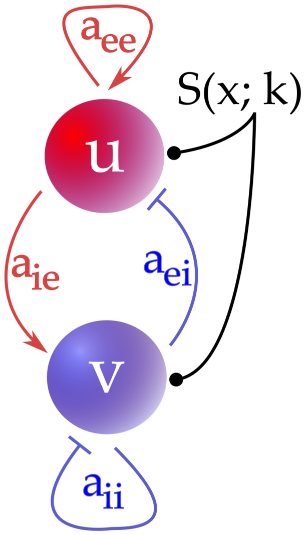

Figure 14: Connectivity diagram for Eq (3.1).

increase in the excitatory activity "before" the Hopf bifurcations occur underlies normal, healthy contrast sensitivity. Thus, we provide a model that directly connects three of the main types of spatiotemporal resonance that occur in neuronal networks.

3.1.1 *Model*

We use a two-population neural field model extended in space with periodic boundary conditions, where the spatial variable x can be in $\mathbb{R}^1$ for the network on the ring or in $\mathbb{R}^2$ for the network on the torus. A schematic showing the connectivity is shown in Fig 14. The equations are:

$$\begin{aligned} \tau_e \; u(x,\; t)_t &= -u + f_e(J_{ee} * u - J_{ie} * v + q \; S(x;\, k)) \\ \tau_i \; v(x,\; t)_t &= -v + f_i(J_{ei} * u - J_{ii} * v + q \, r \; S(x;\, k)) \end{aligned} \tag{3.1}$$

where u and v are excitatory and inhibitory synaptic activities, respectively, and provide a low-pass filter on the firing rate, τ_e and τ_i are the timescales associated with the excitatory and inhibitory synapses, respectively, "$*$" denotes spatial convolution, and $J_{\beta\alpha} = a_{\beta\alpha} K_\alpha$ for $\alpha,\ \beta \in \{e,\ i\}$ and the $a_{\beta\alpha}$ are the maximum connection strengths from α to β.

$K_{e,i} = \dfrac{1}{2\sigma_{e,i}\sqrt{\pi}}\ \exp\left(\dfrac{-|x|^2}{2\sigma_{e,i}^2}\right)$ is a gaussian kernel, $S(x;\, k)$ is our stimulus, parameterized by the wavenumber k and generally taking the form $\cos(\frac{2\pi\, k\, x}{N})$, where N is the number of excitatory or inhbitory neuronal populations being modeled for the network on the ring, or the square root of the number of excitatory or inhibitory populations in for the network on the torus, q modulates the stimulus strength to both u and v, while r is fixed in our study at 0.8 and represents the inhibitory-to-excitatory stimulus strength ratio. Letting

$$f(w) = \frac{1}{1 + \exp(-4w)},$$

the firing rate functions, $f_{e,i}$ can be either

$$f_{e,i}(w) = f(w - \theta_{e,i}) \quad \text{or} \tag{3.2}$$

$$f_{e,i}(w) = f(w - \theta_{e,i}) - f(-\theta_{e,i}), \tag{3.3}$$

where $\theta_{e,i}$ are the excitatory and inhibitory thresholds, respectively. The advantage of the first formulation is that the activity variables u, v are always positive, resulting in values that can be more directly interpreted physically. The advantage of the second formulation is that it sets the steady state of our network, Eq (3.1), to $(u, v) = (0, 0)$, facilitating further calcualtions. In this formulation, we interpret (0,0) as the baseline firing rate, and a positive or negative values of u or v as an increase or decrease from this baseline, respectively. Importantly, we can easily go back and forth between these two systems using the a simple change of variables. For this change of variables, how we choose the parameters, and the parameter values used, see Sec *3.4.1*. While most of the parameters will remain fixed, we vary the values of a_{ee}, k, and q to see the effects on the network's dynamics.

3.1.2 ***Outline***

The organization of the chapter is as follows. First, we look at our model in two spatial dimensions (i.e., the network is on the torus), approximating a sheet of cortex receiving full-field visual stimuli. We examine both simple cosine stripes and more complex noisy images as stimuli. With both simple and complex stimuli, we find that our model demonstrates spatiotemporal resonance, showing spatially heterogeneous oscillatory activity with weak stimuli for some wave numbers, requiring stronger stimuli for others, and showing no sustained oscillatory activity outside of a narrow band of wave numbers.

We then look at simpler modeling assumptions to gain insight into the network's resonant responses. First, we restrict our model to be extended in only one spatial dimension (i.e., the network is on the ring), providing a more amenable formulation for further analysis. Using the numerical tool XPP-AUTO [75], we can analyze the network's dependence on various parameters, defining the regions of instability as functions of, e.g., the recurrent excitation and spatial frequency of the stimulus. In particular, using similar parameters as for the network on the torus, we find the same qualitative responses as on the torus across different stimulus spatial frequencies, demonstrating the utility of reducing our spatial dimensions.

We also study the behavior of the network in response to partial-network stimuli; that is, to stimuli that only project to part of the network. We find parameter regimes that show the spread of large-amplitude oscillatory activity, dynamics that may be highly relevant to PnSE. We then study the network on the ring with stimuli involving the superposition of two spatial modes, allowing for both a more realistic scenario and a still-tractable system to analyze numerically.

Using a perturbation calculation, we can then compute the stability boundaries found above (as a function of the network's recurrent excitation, a_{ee}, and the stimulus amplitude, q, for different stimulus wavenumbers k) near the onset of instability for small-amplitude stimuli of different wave numbers, finding very good agreement with the numerically-computed regions. These calculations lend further insight into the network's responses to different stimuli. We review the results and relate them to the relevant biological and psychophysical findings in Sec *3.3*, before finally detailing the protocols used to produce the results in Sec *3.4*.

3.2 RESULTS

The network extended in both two and one spatial dimensions with periodic boundary conditions imposed (i.e., on a torus and the ring, respectively) displays a strong spatial resonance, showing high sensitivity to stimuli with modes in a narrow band and low or no sensitivity to modes outside of this band, consistent with experimental findings in contrast sensitivity, PnSE, and visual discomfort.

The network on the torus and the ring both give rise to rich and interesting spatial and spatiotemporal patterns that may be relevant to experimental setups that are able to attain sufficient spatial resolution, as well as of mathematical interest. Complex patterns can form and evolve as the result of the interplay of wavelike and more spatially-isolated oscillatory activities. We thus describe the spatial and spatiotemporal patterns that we observe, but we leave a more thorough analysis of such behaviors as future work.

3.2.1 *Network on the torus*

Having chosen our parameters as described above, we simulated our system using custom CUDA C code on a desktop equipped with a GPU on the torus (square domain with periodic boundary conditions) with 512x512 excitatory and 512x512 inhibitory populations (see Sec *3.4.3* and Appendix F).

3.2.1.1 ***Bifurcations on the torus*** We found that as we increased the recurrent excitatory strength to above a critical value, a_{ee}^*, the network spontaneously broke up into spatially heterogeneous oscillatory activity, suggestive of a Turing-Hopf (TH) bifurcation. This activity could evolve into alternating spots or stripes with wavelike motion along the stripes; in either case, the wavenumber (counted as the number of areas of high — or equivalently, low — activity along a fixed horizontal or vertical line across the domain) was approximately 9, the critical wavenumber of the network through the calculation outlined in Sec *3.4.1*. As we mentioned in Sec *3.4.1*, we decreased the recurrent excitatory strength (a_{ee}) below this critical value (a_{ee}^*) so that the system was not very close to such spontaneous pattern formation, as well as to ensure that the network would be more sensitive to stimuli with nonzero wave numbers than to constant stimuli (see Sec *3.2.2.3*).

Beginning with a simple cosine stimulus and varying the wavenumber, k, and the amplitude, q, we simulated the network from uniform random initial conditions in the interval [0, 0.02] and found a contiguous region of $m - k$ space for which the network exhibits spatiotemporal pattern formation, as we see in Fig 15A (see Sec *3.4.3* for a more detailed protocol). This figure demonstrates a clear resonance: spatial stimuli with wavenumbers near 15, corresponding to around 3cpd (see Sec *3.4.2* for the conversion of simulation wavenumbers to physical spatial dimensions), required the smallest amplitudes before the network bifurcated into oscillations. For each k, there is a range of q values that allow for oscillatory activity defined by the boundary curve (see Sec *3.4.3*), as we would expect for Hopf bifurcations. Indeed, we see in Sec *3.2.2.1* that for the model extended in one spatial dimension, oscillations that result from increasing q for a fixed stimulus wavenumber k result from (Turing-) Hopf bifurcations. Thus, we will refer to the bifurcations delineated by the boundary in Fig 15A as Hopf bifurcations. We will use the notation q_* to refer to the lowest observed q value for which the network undergoes spatiotemporal pattern formation, corresponding to the first Hopf bifurcation, and q^* to refer to the highest such observed q value, corresponding to the second Hopf bifurcation. We note that q_* and q^* are functions of k, but we suppress this dependence in the notation when it is contextually clear for simplicity.

For k values for which the network undergoes Hopf bifurcations, the network evolves from a low steady state for $q < q_*$, to spatiotemporal patterns for $q_* \leq q \leq q^*$, to a high steady state for $q^* < q$. For $q < q_*$, the network evolves to a low-activity steady state with vertical stripes that correspond to the stimulus. Similarly, for $q^* < q$, the network is attracted to a high-activity steady state that reflects the stimulus. For $q_* \leq q \leq q^*$ — i.e., q values between the Hopf bifurcations — the network can form complicated spatiotemporal patterns. Below, we describe spatial and temporal aspects of these patterns that we observed.

3.2.1.2 ***Spatiotemporal patterns of simple stimuli on the torus*** Along with the network's temporally oscillatory activity, we observe spatial patterns beyond those directly induced by the stimulus. In Fig 15A, we see an example spatiotemporal pattern that appears

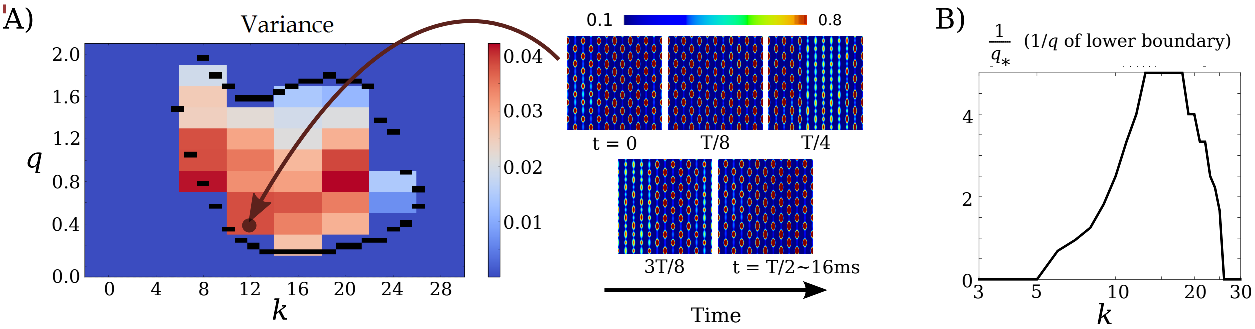

Figure 15: Bifurcations on the torus for cosine stimuli. (A) Left: We probe the dependence of the network on characteristics of the stimulus by varying the wavenumber, k, and amplitude, q, of a simple cosine-stripe stimulus and measuring the variance over an interval of time after letting transients pass, averaged over all of the excitatory elements. The variance is a measure of the oscillatory behavior of the network. Above a threshold (defined as 0.001; see Sec *3.4.3.1*), the network is defined to be in the spatiotemporal pattern regime. This defines the boundary, shown in black. Right: An example of the types of spatiotemporal patterns that can form. We see that spatial patterns form vertically down the stimulus stripes (which are just vertical stripes). The resultant checker-like spots alternate temporally in a wavelike pattern that travels horizontally across the medium. Note: for all simulations shown (in one and two spatial dimensions), only the excitatory populaions (u_j) are shown. (B) Since the network is more sensitive to modes that require lower stimulus amplitudes to trigger pattern formation, inverting the q values of the lower boundary in (A) provides a natural measure of sensitivity. The resultant sensitivity curve highlights the resonance of the network to spatial frequencies near $k = 15$, corresponding to $\approx$ 3cpd, the same resonant mode found for the human visual system.

periodic, or nearly so. Even though the stimulus only consists of vertical stripes (12 vertical stripes, since $k = 12$), the excitatory populations exhibit not only the expected horizontal spatial variation reflective of the horizontal variation of the stimulus, but also vertical spatial variation along the stimulus stripes. Each vertical stripe breaks up into 6 spot-like, localized regions of higher activity. These active regions oscillate in a wave-like pattern, such that each group of vertically aligned active regions oscillate synchronously, or nearly so, and active regions that are not vertically aligned oscillate in a sequential manner as the wavelike pattern propagates in a horizontal direction. One half-period is shown, so that the pattern shown at the end can be seen to be the "alternated" version of the pattern at the beginning; i.e., the active regions from the $t = 0$ frame are inactive in the $t = T/2$ frame, and vice versa.

We observe such vertical spatial variation for all k and q values that trigger spatiotemporal pattern formation with this simple cosine-stripe stimulus. The patterns that form may or may not alternate spatially along the horizontal dimension. The number of spatial regions (which may appear either as spots or more as bulges) that form along the stripes varied based on both k and q in a way that does not depend in an obvious way on either parameter. We will refer to this number as m_v, the vertical wavenumber such spots can be seen in Fig 15A, right). Thus, based on the preceeding discussion, $m_v = 0$ for $q < q_*$ and $q^* < q$

For example, keeping q fixed at 0.4 and letting $k = 12$, 16, 20, and 24, we found $m_v = 6$, 4, 1–2 (some stripes showed one bulge, some two), and 1, respectively. For q fixed at 0.6 and $k = 12$, 16, and 20, $m_v = 6$, 4, and 8, respectively. Fixing k also resulted in a number of spatial regions along the stripes that did not vary monotonically with q. For example, for k fixed at 12 and $q = 0.4$, 0.8, 1.2, and 1.4, $m_v = 6$, 8, 6, and 1, respectively. However, for any k, we observed that just before bifurcation to the high steady state, m_v decreased to 1. Thus, the transition from oscillatory activity to (high) stationary activity is also marked by a transition from $m_v = 1$ to $m_v = 0$.

The spatiotemporal patterns that arise can vary from wavelike patterns, to spatially periodic oscillations with no wavelike behavior, to complicated patterns that may involve interactions of these two types of behaviors. Examples of wavelike and non-wavelike patterns are seen above in the spontaneous case (i.e., $a_{ee} > a_{ee}^*$, $q = 0$). Most of the stimulus-driven activity that occurs includes aspects of both of these wavelike and spatially-alternating oscillations. For example, in Fig 15A (right panel), we see highly structured activity that alternates along the stimulus stripes and appears to spread between the stimulus stripes. Similarly, wavelike behavior can be more akin to that seen in the spontaneous case, where waves seem to travel along the stripes. More complicated behaviors can occur as well that are not obviously spatiotemporally periodic.

3.2.1.3 ***Resonance on the torus*** Since the network is most sensitive to wavenumbers that require the smallest amplitude to trigger oscillations, we can invert this amplitude to obtain a natural measure of the network's sensitivity to different stimulus wave numbers. That is, suppose the network bifurcates to spatiotemporal oscillations at spatial frequency k_0 with amplitude $q_*(k_0)$. Then the network's sensitivity to spatial frequency m_0 is $\frac{1}{q_*(k_0)}$. Thus, inverting the lower boundary in Fig 15A provides the sensitivity curve of the network, shown in Fig 15B. The network is very sensitive to wavenumbers near 15 ($\approx$ 3cpd), and drops off rapidly, showing no sensitivity to frequencies one octave above or below this value. We note that this figure compares favorably with Fig 4 in [47], supporting the hypothesis that the resonances they describe there arise as a result of spatially heterogeneous Hopf bifurcations in the visual cortex. We also note that while $m^* = 9$, the network was most sensitive to wavenumbers near twice this value. We explore this in much greater detail when we turn to the network on the ring.

3.2.1.4 ***Noisy stimuli*** Most stimuli are much more complex than simple cosines, and in particular noisy images with larger amplitudes in spatial modes near 3–5cpd have been shown to cause visual discomfort [47]. Here we look at the network's responses to full-field stimuli (images) (512×512 elements) that were generated by filtering gaussian white noise by low-pass filtering the noise and increasing the spatial components of wavevectors k with wavenumbers $|\mathbf{k}|$. We describe how we generate the stimuli in detail in Sec *3.4.3.1*. Even though these stimuli are quite complicated, the network's responses are very similar to the responses to simple cosine stimuli with the same modes, as we see in the bifurcation diagrams in Fig 16. Note that we tested the network with two uncorrelated stimuli (Pearson correlation coefficient < 0.001; see Sec *3.4.3.1*) to ensure robustness of the network's responses. As we see by comparing Fig 16A and Fig 16B, the original random image that is then filtered to have the dominant wavenumbers $|\mathbf{k}|$ has little impact on the network's response. Indeed, even though the network rapidly evolves to a high steady state with increasing q relative to the cosine stimuli (note the upper bound for q in these diagrams is half of that in Fig 15A), the general resonance pattern is left intact, so that the network responds strongly (i.e., oscillates heterogeneously) to wave numbers near $|\mathbf{k}| = 5$ and weakly or not at all to wavenumbers outside of this band. Thus, the presence of additional, "weak" modes in the stimulus may change the network dynamics, but does not change the network's resonant response.

3.2.2 *Network on the ring*

While cortex is often approximated as a sheet, the facts that (i) the 1-D linearization calculation correctly predicts the critical wave number for the 2-D system and (ii) that the bifurcation diagrams for the simple striped cosine stimuli and those for the noisy stimuli show the same qualitative behavior suggest that we can gain further understanding of our system by studying the neural field model extended in only one spatial dimension with periodic boundary conditions (i.e., on a ring domain).

Indeed, we find that when we use the same parameters as for the 2-D case (up to a scale factor for the spatial kernels; see Sec *3.4.1*), we obtain the same qualitative resonance behavior. In fact, we found the neural field model extended in one spatial dimension on a ring exhibits many of the same spatiotemporal behaviors that the network does on the torus. Studying the network extended in just one spatial dimension is more numerically and mathematically tractable, allowing us to use XPP-AUTO to further analyze the system and to perform a perturbation analysis of the network near the onset of bifurcation, providing further insight into the network's dynamics. We note that we have transformed the coordinates so that the stimulus-free steady state is at 0 (see Sec *3.4.1*) to simplify the calculations. Thus, the firing rate is measured relative to baseline, so that positive or negative values correspond to firing rates above or below baseline, respectively.

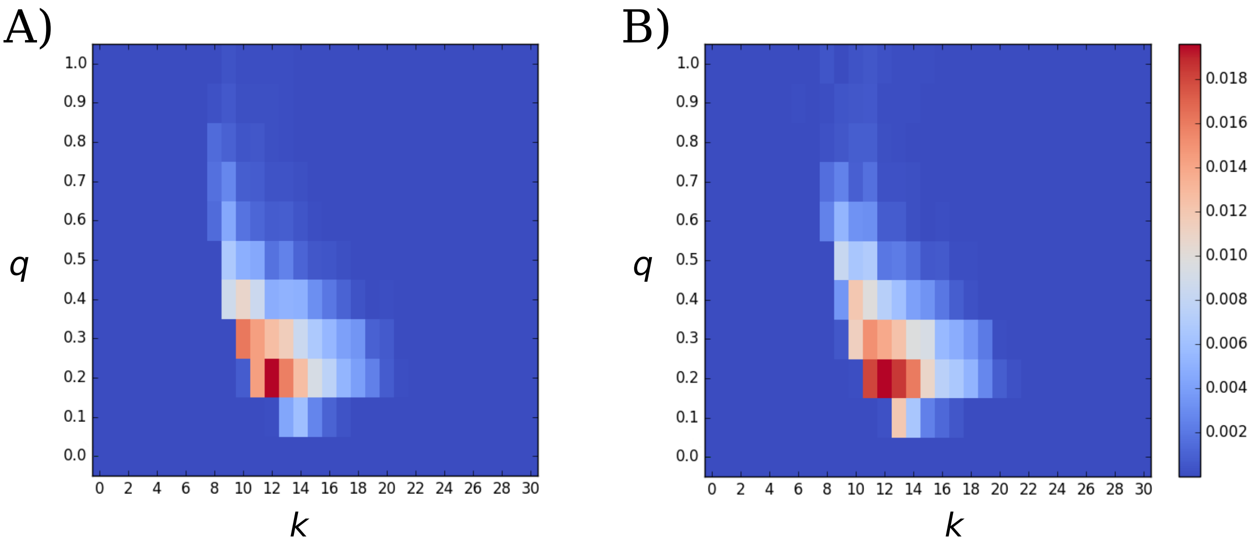

Figure 16: Bifurcations on the torus for noisy stimuli. By filtering two uncorrelated randomly-generated stimuli to have dominant modes from $k = 0$ to 30, as described in Sec *3.4.3.1* and in Fig 26, we test the network's resonance to much more complex stimuli than the simple cosine stimuli shown in Fig 15 by measuring the variance of the network as in the case for cosine stimuli. One seed stimulus is used in (A), and a different one is used in (B). Even though the Pearson correlation coefficient between the two stimuli is less than 0.001, the network responds nearly identically, requiring nearly identical q for each mode $|\mathbf{k}|$ to engage in oscillatory activity. The network shows nearly the same sensitivity to the modes $|\mathbf{k}|$ for these randomly-generated images as it does for simple sinusoidal stimuli (cf. Fig 26).

3.2.2.1 ***Bifurcations on the ring*** The linearization calculation in Sec *3.4.1* predicts the critical network parameters $m^* = 5$ and $a^*_{ee} \approx 7.3746$ (using the "corrected" a^*_{ee} value: the value the trace crosses 0 for a whole wavenumber, in this case 5; the uncorrected a^*_{ee} value, the value for which the determinant exactly equals 0 at a possibly non-integer wavenumber, was ≈7.3741). Thus, we expect the stimulus-free network to have a stable low steady state (at $(u, v) = (0, 0)$) that will lose stability to spatiotemporal oscillations with a wavenumber of 5 at the above a^*_{ee} value.

By following the low steady state in AUTO, we find it loses stability at a Hopf bifurcation at $a^*_{ee} \approx 7.3746$ (Fig 17A). In fact, for this set of parameters the analytically and numerically computed values differ by less than 1×10^{-10}. We note that since the homogeneous network loses stability to nonzero modes, this bifurcation is known as a Turing-Hopf bifurcation. Just above this value (in particular, e.g., at $a_{ee} = 7.367$), we find a variety of stable spatiotemporal patterns of different wave numbers to which the network evolves by beginning with either random initial conditions or initial conditions given by cosines (e.g., $0.1 \cos 2\pi lj/N$ for u_j and v_j, $j \in (1, ..., N)$). Note, we use l to differentiate from both the stimulus wavenumber k and the network wavenumber m. While we tried all of the permissible wavenumbers $l \in P$ (see Sec *3.2.2*) with a range of amplitudes, the network only evolved to spatiotemporal patterns with wavenumbers $m = 3, 4, 5$, and a sort of wave-like pattern involving a combination of wavenumbers 2 and 3, described below. These are shown in Fig 17A. Note that the wavenumber of the network, m, is measured by counting the number of high-activity (or low-activity) regions there at any given time slice (horizontal slices in the images in Fig 17A). Since the low steady state is lost to oscillatory activity at non-zero wavenumbers m, including $m* = 5$, this Hopf bifurcation corresponds to a TH bifurcation, as expected from the system linearization.

There are several patterns that occur with a_{ee} just above a^*_{ee} that involve the critical mode, $m^* = 5$. First, we find two highly-structured stable low-amplitude patterns – (i) and (ii) in Fig 18A. (i) looks like a checker board, with alternating rectangular regions of high and low activity with central regions exhibiting the highest (or lowest) activity. (ii) is a wave pattern with 5 (high-activity) stripes. We note the obvious similarities to the two spontaneous patterns the network on the torus exhibits; however, we caution that even

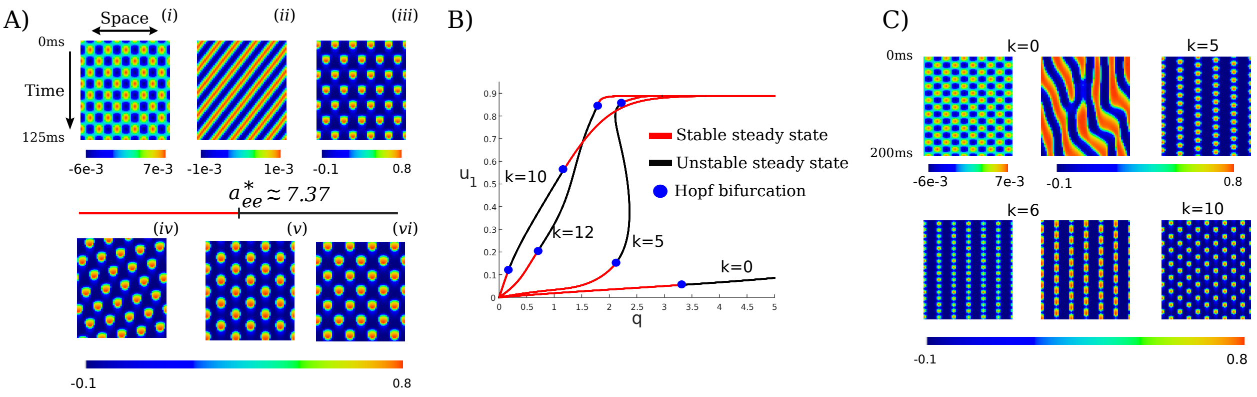

Figure 17: Bifurcations on the ring. The network, extended in one spatial dimension and with periodic boundary conditions imposed, shows similar behaviors as for the network on the torus. Using XPP-AUTO, we are able to precisely identify bifurcations. (A) Above a critical value $a_{ee} = a^*_{ee} \approx 7.37$, the low steady state of the stimulus-free network is lost to spatiotemporal patterns. The red line to the left of a^*_{ee} indicates the steady state is stable, while the black line to the right of a^*_{ee} indicates the steady state is unstable. Various patterns found for $a_{ee} \gtrapprox a^*_{ee}$ are shown. Three patterns at the critical wavenumber 5 are found (where we determine the wavenumber by counting the number of high or low regions of activity for a given time, corresponding to horizontal slices in the figures shown): (i) small-amplitude alternating checkerboard oscillations, (ii) small-amplitude waves, and (iii) large-amplitude alternating oscillations. The other oscillatory solutions found were: (iv) staggered oscillations that alternated between a wavenumber of 2 and 3, and alternating oscillations with (v) wavenumber 3, and (vi) wavenumber 4. (B) Using a simple cosine stimulus of wavenumber k, the steady states of the network are lost at Hopf bifurcations for different values of $q(k)$. (C) Different spatiotemporal patterns can form, depending on the values of k, q. We see that the stimulus wavenumber, k, need not match the wavenumber of the network pattern that forms, m. For constant stimuli ($k = 0$), either small-amplitude checkerboard oscillations of wavenumber $m = 5$ or more fluid, wavelike large-amplitude patterns can form; here, these vary between wavenumber $m = 5$ at the top of the figure to $m = 4$ at the bottom. For $k = 5$, staggered oscillations with wavenumber m varying between 2 and 3. $k = 6$ stimuli can form non-alternating patterns with $m = 6$ (left) or alternating patterns with $m = 3$ (right). $k = 10$ stimuli tend to form alternating oscillations with $m = 5$.

though these and other spatiotemporal patterns shown in 2-D images from the network on a ring look very similar to snapshots of the network on the torus, these images of course include time. Small amplitude (e.g., 0.01 or 0.1) random initial conditions tended to initially look like the small-amplitude checkers, eventually evolving to the small-amplitude waves. We also find larger-amplitude oscillations: When given large (e.g., amplitude 1) cosine initial conditions, the network evolved to the alternating oscillations shown in (*iii*). If we instead provide random initial conditions distributed on a larger interval (e.g., [-1, 1]), we can get (*iv*), a staggered, wave-like large-amplitude alternating oscillation that is more difficult to classify. It appears like a slanted, wave-like alternating oscillation of $m = 4$ or 5; however, counting across a particular slice of time as we have for the other patterns results in 2-3 regions of high activity. We believe this is characteristic of the critical wavenumber 5, for reasons we explore below when we look at patterns induced when we turn on the spatial stimulus. Hereafter, we refer to this pattern as the $(m =)2/3$ pattern. Thus, already we see that the network exhibits tendencies to engage in wavelike, non-wavelike, and combinations of wavelike and non-wavelike behaviors. The other patterns we find just above a^*_{ee} are large-amplitude alternating oscillations at $m = 3$ and 4.

We now study the patterns that form in the network for $a_{ee} = 5.8 < a^*_{ee} \approx 7.4$ by applying a cosine stimulus with wavenumber k and amplitude q (see Sec *3.2.2*). For most permissible k (see Sec *3.2.2*), the network displayed spatiotemporal patterns for $q_* \leq q \leq q^*$ (we use the same notation as in Sec *3.2.1.1*, where it is explained in more detail). By following the steady state in q with AUTO, we see that Hopf bifurcations give rise to the patterns, all with nonzero modes m, as we see in Fig 17C. Thus, on the ring, $q_*(k)$ and $q^*(k)$ correspond to Hopf bifurcations, lending support for our analogous classification of q_* and q^* on the torus as Hopf bifurcations.

The network on the ring shows very similar spatiotemporal dynamics in the three different stimulus amplitude regimes that the network on the torus does. For $q < q_*$, the low steady state is stable, and we observe low contrast stripes with wavenumber k, corresponding to the peaks and troughs of the stimulus. Similarly, for $q > q^*$, the high steady state is stable, leading to high contrast stripes of wavenumber k. In-between the TH bifurcations ($q_* \leq q \leq q^*$), these steady states lose stability to oscillatory modes that depend on k.

We sampled the spatiotemporal modes by beginning the network with random initial conditions in a small interval of $[w_j^{ss} - 0.01,\, w_j^{ss} + 0.01]$ for $w \in \{u,\, v\},\ j \in \{0, ..., N\}$, where w_j^{ss} is the steady state value computed for a particular k, q combination (and is unstable in between the Hopf bifurcations). For example, the bifurcation diagrams in Fig 17B show u_0^{ss} for $k \in \{0,\, 5,\, 10,\, 12\}$ and $q \in [0,\, 1]$. The network then evolved to steady spatially heterogeneous oscillatory modes, some of which are shown in Fig 17C. We find several aspects of interest within this set of patterns. We note that these patterns are meant to be representative but not exhaustive.

For a constant stimulus ($k = 0$, Fig 17C), we find the network can evolve into two disparate types of patterns, one highly structured and one more amorphous and wave-like. The low-amplitude, highly structured pattern appears nearly identical to (i). The large-amplitude, amorphous pattern shows the proclivity of the network to engage in wave-like behaviors as in (ii) and (iv). Note, too, the break up in the wave and the change in wavenumber m from 5 initially (i.e., at the top of the figure) to 4 (at the bottom of the figure, both of which are quite reminiscent of (iv). Of course both patterns are also very similar to those we observed for $k = 0$ on the torus. We note also that driving the network with a stimulus of wavenumber $k = 0$ results in the most dramatic examples in which k and m differ. Otherwise, a k-stimulus tends to drive the network into patterns of either $m = k$ (Fig 17C, $k = 6$, left) or $m = k/2$ (Fig 17C, $k = 6$, right), depending both on k and q. For example, all of the oscillatory patterns that we observed for stimuli with $k = 10$ and different q were alternating oscillations of mode $m = 5$. In contrast, $k = 6$ stimuli could drive the network into either alternating or non-alternating oscillations. $k = 5$ stimuli drive the network into more complicated patterns; in particular, they look like the alternating, staggered pattern $m = 2/3$ that we encountered above in (iv) (Fig 17A). Thus, we hypothesize that this pattern (here and in iv) arises due to the tendency of the network to engage in alternating oscillations and in wavelike patterns; since 5 is odd, the network settles on a wavelike, staggered oscillatory pattern with wavenumbers of 2 or 3.

3.2.2.2 ***Resonance on the ring***

In Fig 17B, we can clearly see a similar resonant behavior in the example bifurcation curves that are plotted. That is, we see an ordering

$q_*(10) < q_*(12) < q_*(5) < q_*(0)$. To visualize this resonance, we construct the network's sensitivity curve, $S(k)$, in the same way as we did for the torus domain, by inverting the lower Hopf values: $S(k) = \frac{1}{q_*(k)}$. In Fig 18A, we see a strong resonance near $k = 10$, corresponding to $\approx$ 3cpd (see Sec *3.4.2*). Indeed, the $S(k)$ on the ring plotted in Fig 18A is qualitatively similar to the $S(k)$ on the torus plotted in Fig 15B, finally confirming the utility in using the simplified ring network to better understand the dynamics of the full network on the torus.

As we mentioned in Sec *3.1*, we hypothesize that the spatially-heterogeneous oscillatory activity physiologically corresponds to neuronal activity related to PnSE and visual discomfort, and that higher neuronal activities correspond to contrast sensitivity. Fig 18B provides the underlying connection between these two related resonances: the peak activities of the populations at different k values (note that the excitatory population u_0 is located at a peak of the stimulus) appear to be directly related to $S(k)$. Indeed, if we take the steady-state values of u_0 for different k at some q_0 such that $q_0 < \min\limits_{k \in P} q_*(k)$, where P is the set of permissible k values, we observe a very similar resonance as that obtained with $S(k)$. For the set of parameters we are using, we have $\min\limits_{k \in P} q_*(k) \approx 0.17$. We therefore take $q_0 = 0.1$, and let $C(k)$ be the steady state values of u_0 when the network is given a stimulus with mode k and amplitude $q = 0.1$ (see Fig 18B). Fig 18C shows $C(k)$; apart from $C(15)$, the relative values match those of $S(k)$ well, so that both curves show strong resonances near $k = 10$, and low sensitivity 1-2 octaves away from this value, in agreement with our results on the torus and with experimental results [47].

3.2.2.3 ***Local numerical stability analysis*** Since the network loses stability to small-amplitude alternating and wavelike oscillations with $m^* = 5$ at a^*_{ee} as predicted by our original linearization (see Sec *3.4.1*), we expected that for a_{ee} near but below a^*_{ee} the stimulus-free network would be very sensitive to stimuli with k near 5, and less so for stimuli away from 5. Based on this, we might (and initially did!) naïvely expect that if we placed the stimulus-free network at a^*_{ee} and turned on the stimulus, the steady state would destabilize to $m = 5$ patterns for arbitrarily small q.

However, even for $a_{ee} = 7$, we see in Fig 19A that $S(k)$ is very similar to that in Fig 18A for $a_{ee} = 5.8$. Effectively, increasing a_{ee} narrows the band of sensitive k to just $k = 10 = 2m^*$

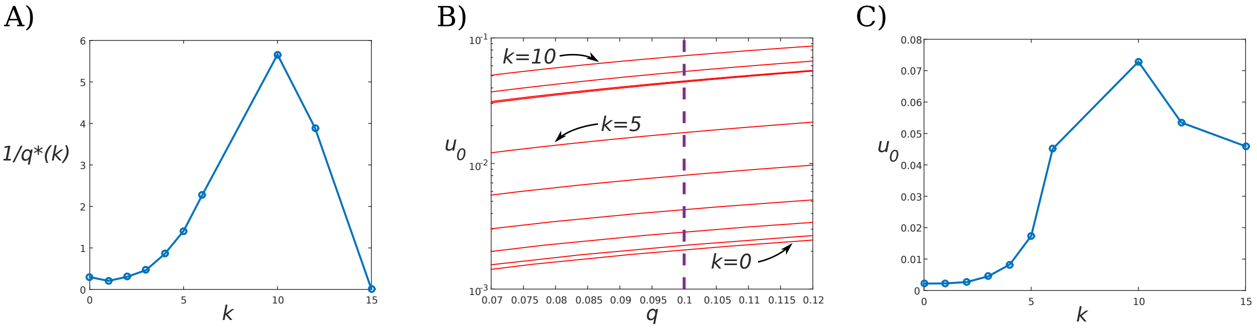

Figure 18: Sensitivities of the network on the torus. (A) As with the network on the torus, inverting $q_*(k)$, the value of the low Hopf bifurcation found in Fig 17B, results in a very similar resonance curve as for the torus domain. Note that the resonant frequency of 10 corresponds to a physical mode of $\approx$ 3cpd (see Sec *3.4.2*). (B) The steady state values of u_0 driven with different stimulus modes $k \in P$. Note, this figure captures the same steady state activity near the origin shown in Fig 17, rescaled to a semi-log plot and constrained to a narrower range of q and u_0 values, and made to include the steady state responses to all k modes, with only three specific modes labeled for clarity. At q values lower than the lowest q_* value of the network ($\approx 0.17 = q_*(10)$), the maximum steady state responses of the network vary according to k. We sample the u_0 values closest to $q = 0.1$, corresponding to the vertical dashed line, and plot them in (C). The resulting curve in (C) is qualtitatively very similar to (A). In our model, the sensitivity curve in (A), associated with temporal activity that does not exist in the stationary stimulus, corresponds to the aberrant responses in seizures and visual discomfort, while the sensitivity curve in (C), associated with measures of stationary responses to stationary stimuli, corresponds to normal, healthy contrast sensitivity in human vision.

while relatively increasing the sensitivity of the network to $k = 0$ (we note that as a_{ee} increases further, the sensitivity to $k = 0$ increases dramatically; we will see why this is the case below). We also recall from our above observations (Fig 17C) that the network tends to engage in *alternating* oscillations, so that for many $(k,\ q)$ values, the network exhibits alternating oscillations with wavenumber $k/2$. Thus, we might instead revise our expectation, so that at a^*_{ee} the steady state would be lost to m^* oscillatory patterns for $k = 2m^*$ stimuli with arbitrarily small q. We can examine this and the more general question of how the stability of the steady state is affected by different k stimuli by probing the dependence of the TH bifurcation of the homogeneous network on q for $k = 10$, and more generally for $k \in P$. We do so by following the TH bifurcation of the homogeneous network (Fig 17A) in the parameters a_{ee} and q for $k \in P$ in AUTO.

Fig 19B shows the interesting and surprising results. Before examining these in more detail, we first observe that the symmetries in the stability diagrams are not surprising, but rather are as expected. For $k \neq 0$, the network's dynamics should be equivalent up to a shift in the coordinates of the neuronal populations due to homogeneity of the stimulus-free network and the symmetry of the cosine stimulus $k \neq 0$ (i.e., changing the sign of q is equivalent to a shift of the stimulus on the ring), as mentioned in Sec *3.2.2*. Thus, for such stimuli, the stability of the network should be independent of the sign of q, leading to the reflective symmetries about the a_{ee} axis seen in the plots for $k = 5$ and 12, e.g.. Since changing the sign of q for $k = 0$ stimuli is not equivalent to shifting the stimulus, but rather causes the stimulus to be either excitatory or inhibitory for all of neuronal populations, we expect positive and negative stimuli to have opposite effects on the stability of the network. This observation directly leads to the odd symmetry about $(q,\ a_{ee}) = (0,\ a^*_{ee})$ seen in the $k = 0$ stability curve.

With $k = 0$, we encounter a basic classification of the stimuli based on their stability curves: stimuli can be *stabilizing*, *destabilizing*, or both. *Destabilizing stimuli* the steady state at a^*_{ee} to become unstable (from being neutrally stable at $(0,\ a^*_{ee})$ since the real parts of the eigenvalues are 0) for arbitrarily small q (i.e., $|q| \to 0$). For a_{ee} values close to but below a^*_{ee}, destabilizing stimuli require q that are very small but bounded away from 0 to do so ($|q| > \epsilon(k,\ a_{ee}) > 0$, where $\epsilon(k,\ a_{ee}) \to 0$ as $a_{ee} \uparrow a^*_{ee}$). In contrast, the steady state

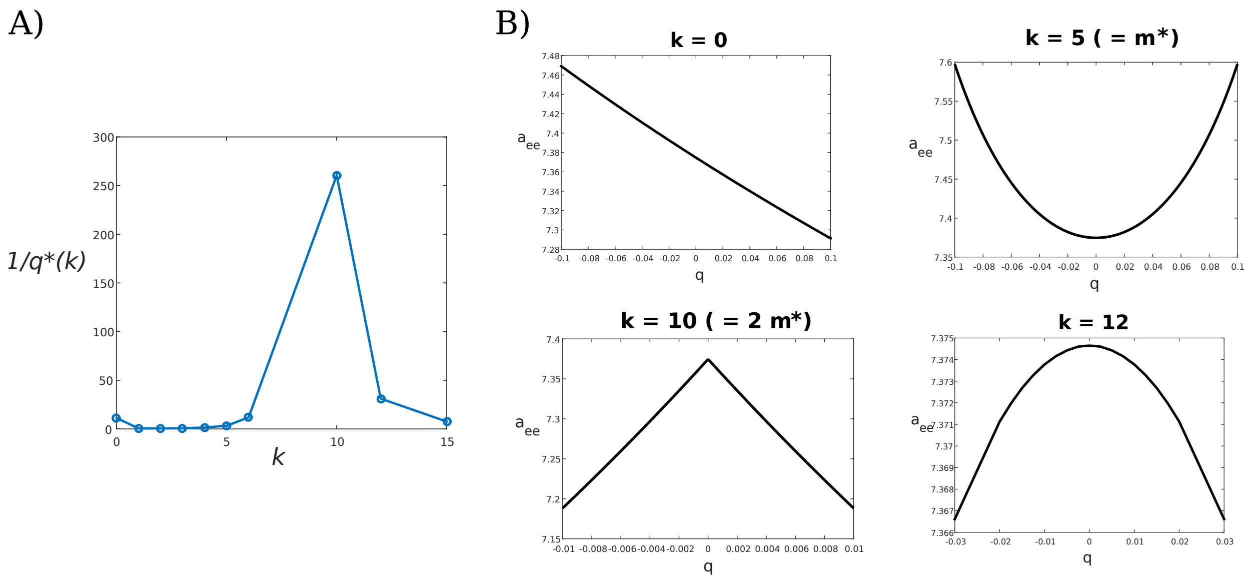

Figure 19: Local bifurcation structure. Bifurcations near a^*_{ee} for different stimulus modes. (A) Near $a{*}_{ee} \approx 7.37$ (here $a_{ee} = 7.3$), the network sensitivity curve is similar to lower a_{ee} values (cf. Fig 18A), with a more pronounced peak near $k = 10$. Therefore, even though we have been using lower a_{ee} values, identifying the local bifurcation structure near a^*_{ee} may be of use in understanding the resonance observed at lower a_{ee} values. (B) We see several qualitatively different bifurcation structures. Each curve partitions the area of the graph shown (i.e., the partitions are only local) into an area where spatiotemporal patterns form (above the curves) and an area where the steady state is stable and no stable oscillations occur (below the curves). As expected, all of the bifurcation diagrams for $k > 0$ are even-symmetric about $q = 0$ and that for $k = 0$ is odd- symmetric about $q = 0,\ a_{ee} = a^*_{ee}$ due to the associated symmetries with respect to the amplitude q in the stimulus. The stimuli can either *destabilize* the steady state at a^*_{ee} when q is turned on as for $k = 10$ and 12, *stabilize* it, as for $k = 5$, or both, as for $k = 0$ (which is stabilizing for negative q and stabilizing for positive q).

at $(0,\ a^*_{ee})$ becomes stable with *stabilizing stimuli* for arbitrarily small $|q|$, while requiring small $|q|$ bounded away from 0 to stabilize the unstable steady state for a_{ee} greater than, but arbitrarily close to, a^*_{ee}.

In Fig 19B, we see that for the various k the dependence of the stability curves on q appear to be either linear or q raised to an even power. In fact, in Sec *3.2.2.6* below, we will see that the dependence are exactly either linear or quadratic for any k. We note that all of the remaining k lead to stability curves that appear like those for $k = 5$ and 12. In fact, this observation points out a surprising fact: there appears to be *nothing special* about driving the network at $k = m^*$. Indeed, for the parameters we've used, $k = m^*$ stimuli are not even destabilizing, but rather are *stabilizing*. Though this result is not general and can depend on the network parameters, the quadratic dependence on q for $k \neq 0,\ 2m^*$ is a general result (independent of the parameters chosen) that we find below in Sec *3.2.2.6*. In contrast, we find special roles for $k = 0$ and $2m^*$: the stability curves for these are linear.

Together with the quadratic dependence for all other k, this means that *regardless of the k considered* the network will always be most sensitive to $k = 0$ and $k = 2m^*$ stimuli near a^*_{ee} if these stimuli are at least in part stabilizing, as is the case here; we examine this assumption further in Sec *3.2.2.6*. Similarly, this suggests that if we lower a_{ee} sufficiently, the network might still be sensitive to $k = 2m^*$ and perhaps other k with quadratic stabilizing curves, while being less sensitive to $k = 0$. However, since we are only analyzing the local stability near $(0,\ a^*_{ee})$, this is purely suggestive. But we did find such results above by setting $a_{ee} = 5.8$, so that this reasoning may provide some helpful intuition for the sensitivity curve in Fig 18A.

3.2.2.4 ***Partial-field stimuli*** In experimental situations, stimuli only occupy a portion of the visual field. We found that the network still exhibits similar resonance patterns when we present stimuli on some of the populations, but not others. In particular, we studied stimuli over a continuous interval of 39 populations (so a continuous interval of 21 populations received no stimulus) (see Sec *3.2.2.4*).

For lower a_{ee} values, the resultant spatiotemporal patterns did not spread, while for higher values (but below a^*_{ee}), the patterns could spread from the stimulus region to the stimulus-free region. However, the patterns in the two regions could be quite different from each other. Such a distinction between types of dynamics (localized activity vs. spreading activity) may underlie the distinction between experiencing discomfort vs. experiencing a seizure due to an image [48].

We found that the network can display similar sensitivity to patterns in both cases. Since in the case of spreading activity we were interested not in when oscillatory activity occurred, but when it spread to the stimulus-free region, we tested for when large-amplitude oscillations occurred in the stimulus-free region by beginning with small random initial conditions and observing the network's behavior (see Sec *3.4.4.1* for further detail). This afforded us an opportunity to test our using the destabilization of the network's low steady state through a Hopf bifurcation as our metric of the sensitivity to different stimuli. In general, we found very good agreement between the two measures; that is, q values that caused the network to engage in large-amplitude oscillatory activity in the stimulus-free region for a particular stimulus wavenumber k were very close to the q values at which the low steady-state destabilized through a Hopf bifurcation, as seen in Fig 20A.

The presence of oscillatory activity in the stimulus-free region suggests the network is multistable: depending on initial conditions, the network may evolve to different stable dynamic states. In particular, the large-amplitude oscillatory states that emerge suggest that there are several subcritical Hopf bifurcations, leading to multistability between the low steady state and several different oscillatory modes. Indeed, we found that when we turned off the stimulus (setting q to 0) after the large-amplitude oscillations had spread to the stimulus-free region (see Sec *3.4.4.1* for details), the network continued to oscillate, stabilizing at one of three patterns, regardless of the stimulus wavenumber k. The modes that emerged were checker patterns with either $m = 3$, $m = 4$, or else the $m = 2/3$ pattern (Fig 20B).

Going back to the full network, we found that all three of these patterns were stable for higher a_{ee} values. We were able to follow the checker patterns with m =3, 4, and found that indeed these corresponded to subcritical Hopf bifurcations in a_{ee}, as seen in Fig 20C. Which pattern emerges after turning off the stimulus depends on m, q, and the time at which it is turned off.

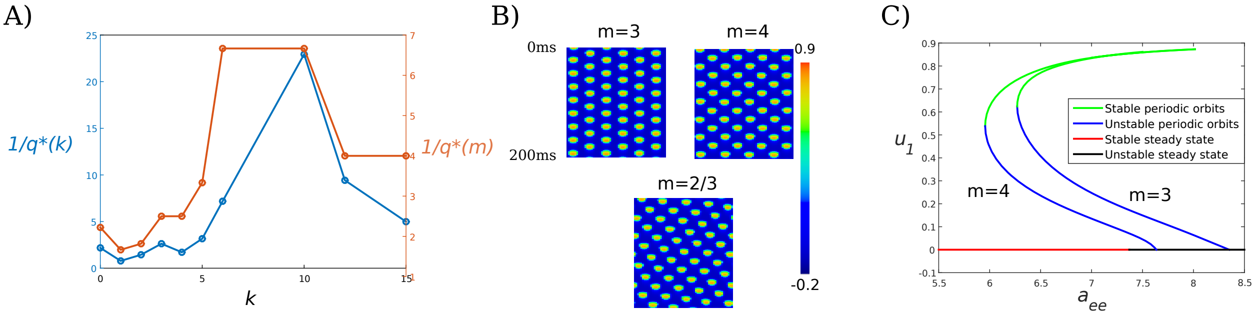

Figure 20: Partial-field stimuli bifurcations. By setting a_{ee} to 7 and stimulating only 39 of the 60 excitatory and inhibitory populations we test to see what q values are required for each k for the patterns spread to the full network. (A) Simulations that begin with random initial conditions (red curve) give a very similar sensitivity curve as found by determining the q values of the Hopf bifurcations (blue curve). (B) The modes that grow in the stimulus-free region (cf. Fig 21 and 22) tend to be $m = 3$, $m = 4$, and $m = 2/3$ (shown in (C)). Following the regularly oscillating modes 3 and 4 shows why these modes appear: they arise as subcritical Hopf bifurcations that persist into below the a_{ee} value used in the stimulus-free network.

For example, when $k = 2$ with a higher $q = 1.5$, the pattern that restults once the stimulus is turned off depends on the extant network state. In Fig 21A, we see that with the stimulus on, the network seems to transition through one pattern and stabilize at another. When the stimulus is shut off during the first pattern, the network settles into the $m = 4$ state (Fig 21B, bottom), while when the stimulus is shut off during the second pattern, the network settles into the $m = 2/3$ state (Fig 21B, top). More dramatic changes are possible as well. For example, with $k = 4$ and $q = 1$, the time at which the stimulus is turned off ("phase dependence") determines whether or not the network maintains its activity (Fig 22B).

The stereotypical behavior we observed was that once the stimulus-free region engaged in oscillatory activity, the network then continued oscillating in one of the three patterns mentioned ($m = 3$, $m = 4$, or $m = 4/5$) once the stimulus was removed. Interestingly, as for most of the wavenumbers presented, the network could even evolve into any of the three patterns after stimulus removal for stimulus wavenumbers $k = 3$, 4, or 5, depending on q (we did not observe a phase dependence). In contrast, for $k = 10$, the only resultant pattern observed following stimulus removal was $m = 2/3$.

Some interesting nonstereotypical behaviors in addition to a dramatic phase dependence were observed as well, generally at higher q values. For example, the application of the stimulus could initially lead to either oscillatory or high steady-state activity only in the stimulus region, but upon removal of the stimulus the network could then remain active with large-amplitude oscillations (Fig 17B, right). Counterintuitively, the stimulus could instead lead to large-amplitude oscillatory activity in the *stimulus-free* region, activity that would then spread to the remainder of the network following stimulus removal (Fig 22A). Such interesting examples show that the rich dependences of the resultant full-network pattern (on, e.g., k, q, the number of populations covered and not covered by the stimulus, and the time we set $q = 0$) and the complex driven dynamics result in a complicated setup to explore, one that may warrant further systematic study.

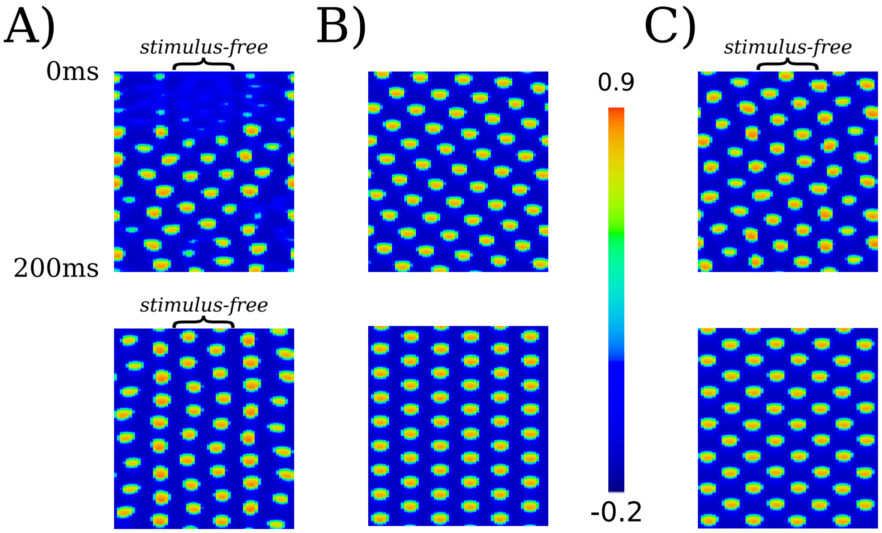

Figure 21: Typical dynamics for partial-field stimuli. Recall that k refers to the mode of the stimulus, while m refers to the (spatial) mode of the network, which may be different from k. (A) $k = 4$, $q = 0.4$, where the stimulus is applied everywhere except the indicated *stimulus-free* region. After the pattern spreads to the stimulus-free region (top), the network then evolves into an irregular full-field oscillatory pattern (B). If the stimulus is removed during the transient pattern shown at the end of (A, top), the network evolves into one of the two regular oscillatory patterns in (B) (top: $m = 2/3$; bottom: $m = 3$. If the stimulus is removed during the stable patterns shown in (A, bottom), the network evolves into the pattern shown in (B, top). (C) $k = 4$, $q = 0.4$, where the stimulus is applied everywhere except the indicated *stimulus-free* region. The network evolves into the pattern shown on top with the partial-field stimulus. After removing the stimulus, it is then attracted to the $m = 4$ pattern, below.

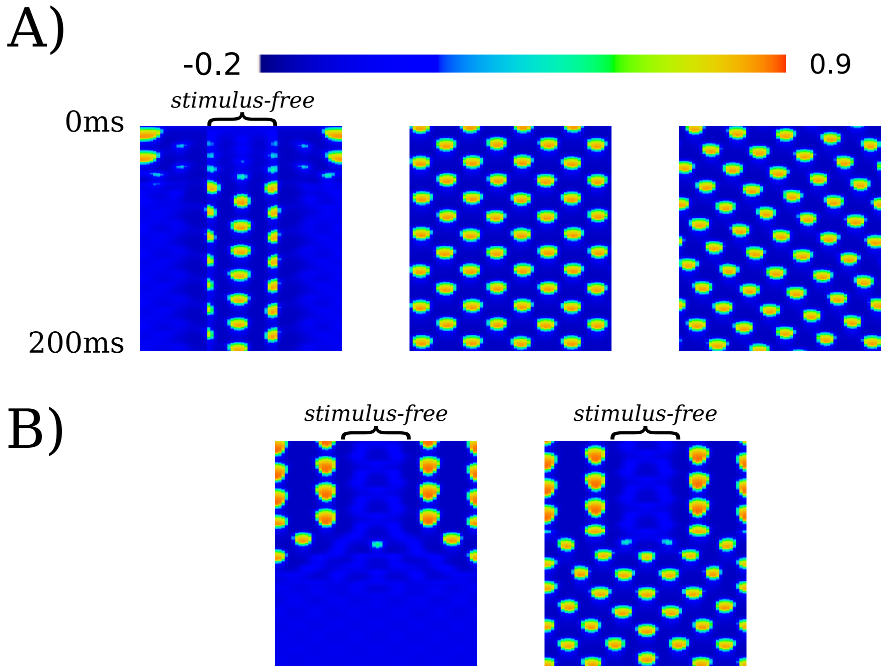

Figure 22: Atypical dynamics for partial-field stimuli. For (A, left) and (B), the stimulus is applied everywhere except the indicated *stimulus-free* region. Again, recall that k refers to the mode of the stimulus, while m refers to the (spatial) mode of the network, which may be different from k. In particular, once the stimulus is removed, the stimulus mode k no longer directly influences the network pattern. (A) $k = 2$, $q = 0.4$. Left: Even though the stimulus is applied to the outside populations (i.e., the middle 21 populations are receiving no stimulus), large-amplitude oscillations only form in the stimulus-free region. When the stimulus is removed, the full network continues oscillating with either the $m = 4$ (middle) or $m = 2/3$ (right) patterns. (B) $k = 4$, $q = 1.0$. Here, the pattern does not spread to the stimulus-free region. Once the stimulus is removed, the full network dynamics can either die down to a low steady state (left) or continue oscillating with $m = 4$ (right), depending on the phase of the oscillation at which the stimulus is removed. There is only a 5ms difference in times at which the stimulus is removed between the left and right plots.

3.2.2.5 *Mixed-mode stimuli* Naturally occurring stimuli are very rarely composed of a single spatial frequency. This is even the case for most human-made objects and images. A natural question that arises is: How do neuronal networks respond to multiple stimulus modes? This is a broad question, and so we will narrow the question considerably. Indeed, we have already encountered two situations in which we explored the interaction of different modes. In Sec *3.2.1.4*, we provided the network a stimulus that was the sum of hundreds of wavevectors with one dominant mode, $|\mathbf{k}|$and saw that the network can still respond to the stimuli in a very similar way as for simple cosine stimuli of the same wavenumber $|\mathbf{k}|$. In Sec *3.2.2.3*, although we only provided a single stimulus, we saw a somewhat more interesting interaction of modes near a^*_{ee}. For example, the network is most sensitive to stimuli with wavenumbers k that are twice the natural mode of the network (m^*), and in fact such stimuli induce alternating oscillations with wavenumber $k/2 = m^*$.

As we describe in Sec *3.4.4.2*, we limit ourselves here to the question of how a stimulus with modes $k_1,\ k_2$ and amplitudes $q_1,\ q_2$ affect the network's stability, where we let $k_1 \in P$ and $k_2 = 2m^* = 10$ (see Eq 3.14 for the form of the stimulus). To see how the network responds to the mixed-mode stimuli, first recall that earlier we found there is an associated amplitude, $q_*(k)$, at which the network exhibits a Hopf bifurcation for each wavenumber $k \in P$, except for $k = 15$. In order to test how the sensitivity of the network changes as a result of adding a second mode, we can follow the Hopf point of one mode as a function of the amplitude of the second mode, restricting ourselves to positive amplitudes without loss of generality due to the symmetry considerations discussed in Sec *3.2.2.3*. That is, we expect the network to destabilize through a Hopf bifurcation from the steady state for different combinations of $q_1,\ q_2$. This stability boundary should trace out a closed curve in the q_1-q_2 quarter-plane, connecting $(q_1,\, q_2) = (0,\, q_*(k_2))$ to $(q_1,\, q_2) = (q_*(k_1),\, 0)$. We will denote this boundary by $q_*(k_1,\, k_2) = 0$, shown in Fig 23A as the black curves. The boundary curve partitions the upper-right quadrant in the q_1-q_2 quarter-plane into two regions: an inner region, the region closer to the origin, where the steady state is stable, and an outer region, the region above and to the right of the boundary, where spatiotemporal patterns form.

These curves give us information about how two modes interact, and we will see that the modes can either be *facilitating* or *depressing* relative to each other, by which we mean that if we increase the amplitude of one mode, the network might be more likely (*facilitating* modes) or less likely (*depressing* modes) to exhibit spatiotemporal patterns. We can more precisely classify these behaviors based on whether, beginning on the stability boundary $q_*(k_1, k_2) = 0$, increasing q_1 or q_2 destabilizes (*facilitating* modes) or stabilizes (*depressing* modes) the network.

To illustrate this, consider the 3 branches of the curve for $k_1 = 2$ ((*i*)–(*iii*) in Fig 23A). For the point shown on (*i*), we see that if we increase either q_1 or q_2, then we move from a point of neutral stability on the boundary into the pattern-forming region. Thus, since keeping the amplitude of either mode fixed and increasing the amplitude of the other mode destabilizes the network, for any point along (*i*), $k_1 = 5$ facilitates $k_2 = 10$, and vice versa. In contrast, consider the point on (*ii*). In this case, increasing q_2 destabilizes the network, whereas increasing q_1 *stabilizes* the network, since the point moves into the steady-state region. Hence, for any point along (*ii*), $k_2 = 10$ facilitates $k_1 = 5$, while $k_1 = 5$ depresses $k_2 = 10$. Similarly, for the point on (*iii*), we see that increasing q_1 destabilizes the network, while increasing q_2 stabilizes the network. Thus, for any point along (*iii*), $k_1 = 5$ facilitates $k_2 = 10$, while $k_2 = 10$ depresses $k_1 = 5$.

From this example, we can draw several general conclusions. (1) Boundaries with negative derivatives (as in (*i*)) correspond to modes that are *mutually facilitating*: k_1 and k_2 facilitate each other; (2) boundaries with positive derivatives at all points (as in (*iii*) with the possible exception of one point with zero derivative but positive second derivative (as appears to occur at $(0, q_*(k_2))$, the leftmost point of (*ii*), correspond to *one mode that is depressing and one mode that is facilitating* (i.e., one mode depresses the second, while the second mode facilitates the first); (3) there must be at least one part of the boundary where k_1 facilitates k_2, and vice versa. This third point follows from the fact that we cannot have a boundary curve that is a function of either q_1 or q_2 with derivative everywhere positive that connects $(0, q_*(k_2))$ to $(q_*(k_1), 0)$. The best we could have would be a boundary composed of two branches with derivatives everywhere positive that come together in a cusp (consider, for example, a boundary curve like that for $k_1 = 12$, except where the left branch is concave up).

The boundary for $k_1 = 5$ in Fig 23A is similar to those for $k_1 = 3$, 4, and 6. These boundaries are composed of a left branch wherein $k_2 = 10$ facilitates $k_1 = 5$ while $k_1 = 5$ depresses $k_2 = 10$, and a right branch wherein $k_2 = 10$ and $k_1 = 5$ mutually facilitate each other. Thus, while $k_1 = 5$ may either facilitate (for higher q_1 values) or depress (for lower q_1 values) the $k_2 = 10$ mode, $k_2 = 10$ facilitates $k_1 = 5$ along the entirety of the boundary. We note that these statements similarly hold true for $k_1 = 3$, 4, and 6. These observations lead to a fourth general conclusion: (4) Boundaries composed of one branch of positive derivative almost everywhere and one branch of negative derivative almost everywhere correspond to one mode that is *everywhere facilitating* and a second mode that is depressing along the branch with positive derivative.

This same situation obtains for $k_1 = 12$, with the roles of k_1 and k_2 reversed. That is, $k_1 = 12$ facilitates $k_2 = 10$ along the entirety of the boundary, while $k_2 = 10$ facilitates $k_1 = 12$ for higher q_2 values (the upper branch in the plot), and depresses $k_1 = 12$ for lower q_2 values (the lower branch in the plot).

Finally, we note that there is a way in which modes might interact in an even more synergistic fashion. First, consider a case where two modes are *interchangeable*. Then, if we are at a point along the boundary curve and increase, say, q_1 by $\Delta(q_1)$, we would expect that we would need to *decrease* q_2 by $\Delta(q_1)$ in order to stay on the boundary curve. Or, more generally, we would expect that we would need to decrease q_2 by $c \cdot Delta(q_1)$ for some fixed $c > 0$ for *all* points on the boundary curve. That is to say, we would expect the boundary curve to *lie along the line that connects* $(0, q_*(k_2))$ *to* $(q_*(k_1), 0)$. We will denote this line as $q_2 = L(q_1)$. Unsurprisingly, this is the exact situation we find when we let $k_1 = 10$, as we see in Fig 23A. However, this also suggests that there is a special region in the q_1-q_2 quarter-plane: That is, if the network destabilizes for (q_1, q_2) values that lie below $L(q_1)$, this demonstrates that the network is *more sensitive* to the corresponding modes with these amplitudes than to either pure mode (where the second amplitude is set to 0). Indeed, we find that the *entire* stability boundary for $k_1 = 0$ lies below $L(q_1)$. Hence, not only are $k_1 = 0$ and $k_2 = 10$ mutually facilitating along the entire boundary, but mixing the two modes results in a stimulus to which the network will exhibit greater sensitivity than to either mode by itself.

We note that, while many of the specific results we describe can only be found by following the Hopf points in two parameters as we did, many of the general features are exactly what we would expect based on previous results described above, while others are surprising. In particular, since we found that the network was especially sensitive to $k_2 = 10$, we expect $k_2 = 10$ to facilitate other modes. Indeed, we find this to be the case, with only the few exceptions we enoutered earlier; i.e., on the right and lower branches of the boundaries for $k_1 = 2$ and 12, respectively, where $k_2 = 10$ is depressing. Since we also found that the network was nearly as sensitive to the $k = 12$ mode at $a_{ee} = 5.8$ Fig 18A and to the constant mode near bifurcation Fig 19B, we might expect these modes to facilitate $k = 10$ for certain (q_1, q_2) values. We find exactly this, though somewhat surprisingly these modes in fact facilitate the $k = 10$ mode along the entirety of the boundary curves. In contrast, the modes $k_1 = 2$, 3, 4, 5, and 6 all have branches along which k_1 depresses $k_2 = 10$. While we might naìvely have expected these modes to depresses the $k_2 = 10$ mode throughout the region of interest, we saw that this cannot be the case. That is, the argument leading up the general conclusion (4) above demonstrates that a mode cannot be *purely depressing*, whereas, as we saw in the first and fourth general conclusions, (1) and (4) above, a mode may in fact be *purely facilitating.*

The network's oscillatory activities that arise with multiple stimuli can be quite interesting as k_1, k_2, and m^* all interact to produce the resultant patterns. We can see two such patterns in Fig 23B. Overall, we see that, even in limiting the number of stimulus modes to 2, rich and interesting dynamics and network behaviors can arise. The setup is also simple enough to allow us to numerically analyze the stability as a function of the stimulus amplitudes. It may be of further interest physiologically to study the interaction of additional stimuli, at least through simulations. Such studies might allow us to come closer to making more precise predictions about how neuronal populations might behave when exposed to richer, more realistic stimuli.

3.2.2.6 ***Stability Analysis*** The bifurcation analysis above suggests that near a_{ee}^*, a small-amplitude stimulus $q \cos(2\pi kx/N)$ may either destabilize or stabilize the low steady state. In order to gain some analytical insight into this question, we consider the original system Eq (3.1) extended in one spatial dimension. We would like to know whether, near

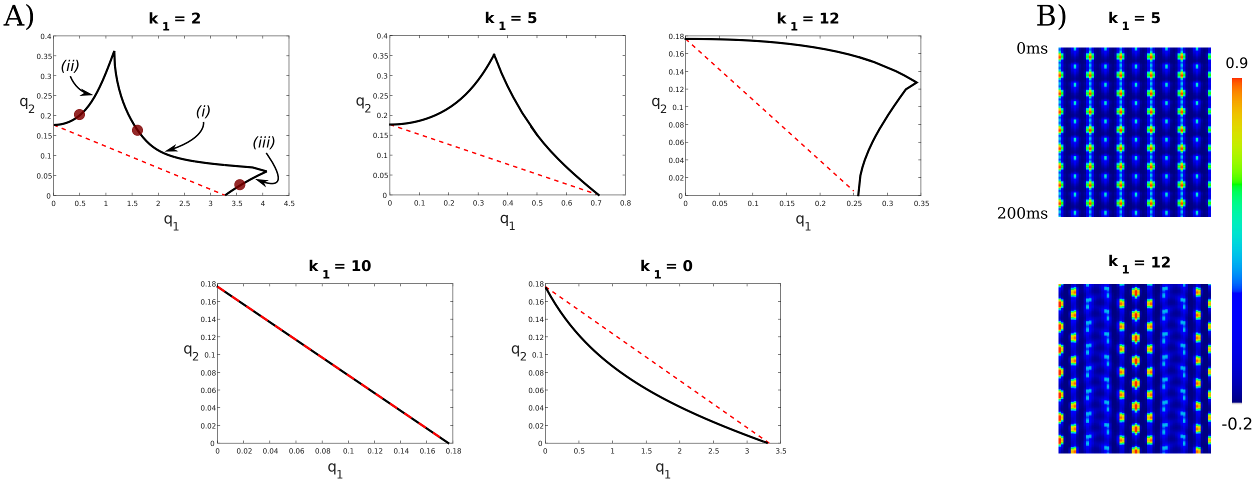

Figure 23: Stability curves for stimuli with two modes. (A) We examine the effect on the pattern formation of the network by providing stimuli that are mixtures of one mode, $k_1 \in P$, and the most sensitive mode, $k_2 = 10$. Black curves are curves of Hopfs that correspond to the stability boundary of the network, $q_*(k_1, k_2) = 0$. Points "inside" of these curves (in the regions of the quarter-plane closer to the origin) correspond to stable steady states. Points "outside" of these curves (in the regions of the quarter-plane above and to the right of the curves) correspond to spatiotemporal pattern formation. As described fully in the text, branches of a boundary curve with negative derivative correspond to where the two modes k_1 and k_2 are mutually facilitating, while branches with positive slope correspond to where one mode facilitates the second, while the second mode depresses the first. In the case that $k_1 = 10$, the two modes are interchangeable, so that increasing the amplitude of one stimulus is equivalent to decreasing the amplitude of the second stimulus by a constant times the increase of the first; thus, such boundary curves lie along the line connecting the two Hopf points corresponding to stimuli of the respective pure modes. This line, $L(q_1)$, is the red dashed line in the plots. Since the boundary curve for $k_1 = 12$ lies below $L(q_1)$, the network will be more sensitive to a stimulus with modes 0 and 10 than to a stimulus of either mode by itself. We note that the modes can be significantly depressing relative to $k = 10$; e.g., providing the modes 2 and 5 can more than double the amplitude of $k = 10$ required to destabilize the steady state from ≈ 0.17 to above 0.35. (B) Example spatiotemporal patterns found when $k_2 = 10$ and $k_1 = 5$ or 12 (cf. the patterns found with just one stimulus in Fig 17).

a^*_{ee}, perturbing the network with a small-amplitude stimulus $q\,\cos(2\pi kx/N)$ will destabilize the low steady state, potentially resulting in spatiotemporal patterns as we have studied above. Therefore, we perform a perturbation calculation, using q as the small parameter that we expand with. Here, we present an outline of this calculation; for more detail, please see Appendix E.

We first solve for our nonhomogeneous steady state, $(u_{ss},\, v_{ss})$, where, e.g., $u_{ss} = q\,u_0 + q^2\,u_1$ by Taylor-expanding Eq (3.1) using Eq (3.3) about the homogeneous steady state (0,0) and substituting it for $(u,\, v)$. We then linearize around the nonhomogeneous steady state $(u_{ss},\, v_{ss})$, providing a linear set of ODEs:

$$\Omega_t = L_0\Omega + qL_1\Omega + q^2L_2\Omega \tag{3.4}$$

where $\Omega(x,t) = [\widetilde{u}(x,t),\, \widetilde{v}(x,t)]^T$ and the linear operators L_1 and L_2 contain terms involving $\exp(\pm i\mu x)$, where $\mu = 2\pi k/N$. We then look for separable solutions, $\Omega(x,t) = Z(x)e^{\lambda t}$, and expand both Z and λ in powers of q. Plugging this back into Eq 3.4 and collecting terms of like powers of q provides a series of linear equations, as many as the number of powers of q in our expansion.

The real parts of the λ_is determine the stability of the solutions; since we set our system near a Hopf bifurcation, and the 0th order equation is our original linearization, Eq (3.10), $\mathbb{R}(\lambda_0) = 0$. We therefore need to continue up to at least $\mathcal{O}(q)$, and as it turns out, $\mathcal{O}(q^2)$. Thus, our sequence of equations looks like $\sum\limits_{j=0}^{i}(\lambda_j I - L_j)Z_{i-j} = 0$ for $i \in (0,1,2)$. We can then apply the Fredholm alternative to the second and third equations to find λ_1 and λ_2 in order to determine the stability of the system to small perturbations of the form $\cos(\mu x)$.

For $\mathcal{O}(q)$ and letting $\omega^* = 2\pi m^*/N$, the Fredholm alterntative solvability condition gives

$$\lambda_1 \int\limits_0^1 \cos\omega^* x\, \cos\omega^* x\, dx = c \int\limits_0^1 \cos\mu x \cos^2\omega^* x\, dx, \tag{3.5}$$

where c is a nonzero constant determined from the prior computations. The right hand side is nonzero only when $\mu = 0$ or $2\omega^*$; for other stimulus wavenumbers, $\lambda_1 = 0$, so that we must solve the $\mathcal{O}(q^2)$ equation for λ_2. These equations are similar to Eq 3.5, involving constants multiplying integrals that evaluate to 1 or 0 (due to orthogonality). Thus, we can obtain information about the stability of the network in response to a cosine stimulus. Furthermore, we can build on these results to find the local stability boundaries in the $q - a_{ee}$ plane and compare these analytical results to those found in Sec *3.2.2.3* through numerical bifurcation methods.

To do so, we expand λ in q for small q, recalling that λ_1 is nonzero for $\mu = 0$ or $2\omega^*$:

$$\lambda = \lambda_0 + \lambda_1\, q + \lambda_2\, q^2$$

Recall that $\mathbb{R}(\lambda) = 0$ when $a_{ee} = a_{ee}^*$, giving the curve:

$$\mathbb{R}(\lambda)(a_{ee}, q) = 0 = \alpha(a_{ee} - a_{ee}^*) + \beta\, q + \gamma\, q^2 \tag{3.6}$$

where $\beta,\ \gamma = \mathbb{R}(\lambda_1),\ \mathbb{R}(\lambda_2)$. Letting $\eta = a_{ee} - a_{ee}^*$, we have from Eq 3.6 $\alpha = \frac{d}{d\eta}\mathbb{R}(\lambda)|_0$, which we compute through the eigenvalue equation

$$A(\eta)\, \Phi(\eta) = \lambda(\eta)\, \Phi(\eta).$$

This provides the linear equation

$$(\alpha\, \mathrm{Id} - A'(0))\Phi_0 = (A(0) - \lambda_0\, \mathrm{Id})\Phi'(0), \tag{3.7}$$

where some of the previous variables we have used correspond to quantities here; i.e., $\lambda_0 = \lambda(0)$, $\Phi_0 = \Phi(0)$. We may then apply the Fredholm alternative to Eq 3.7 in order to find α, specifying the stability boundary curve in Eq 3.6. For more detail, please see Appendix E. For small q, our theoretical curves match the numerical boundary curves very well (Fig 24).

This calculation shows that, indeed, we should always obtain linear curves in the $q - a_{ee}$ plane for $k = 0$ or $2\, m^*$, and quadratic curves for all other k. Thus, near a_{ee}^*, the network counterintuitively will always be more sensitive to $k = 0$ and $k = 2m^*$ stimuli (as long as the curve for $k = 2m^*$ is concave down) than to $k = m^*$, regardless of the concavity of the curve at m^*. For most applications, the greater sensitivity to m^* near a_{ee}^* serves as a constraint: As we see in, subjects do not appear in general to be sensitive to uniform (spatially and temporally) light in comparison to patterns near 3–5cpd; thus, if we set a_{ee} low enough (e.g., below the point of intersection of the quadratic and linear curves in the case that any

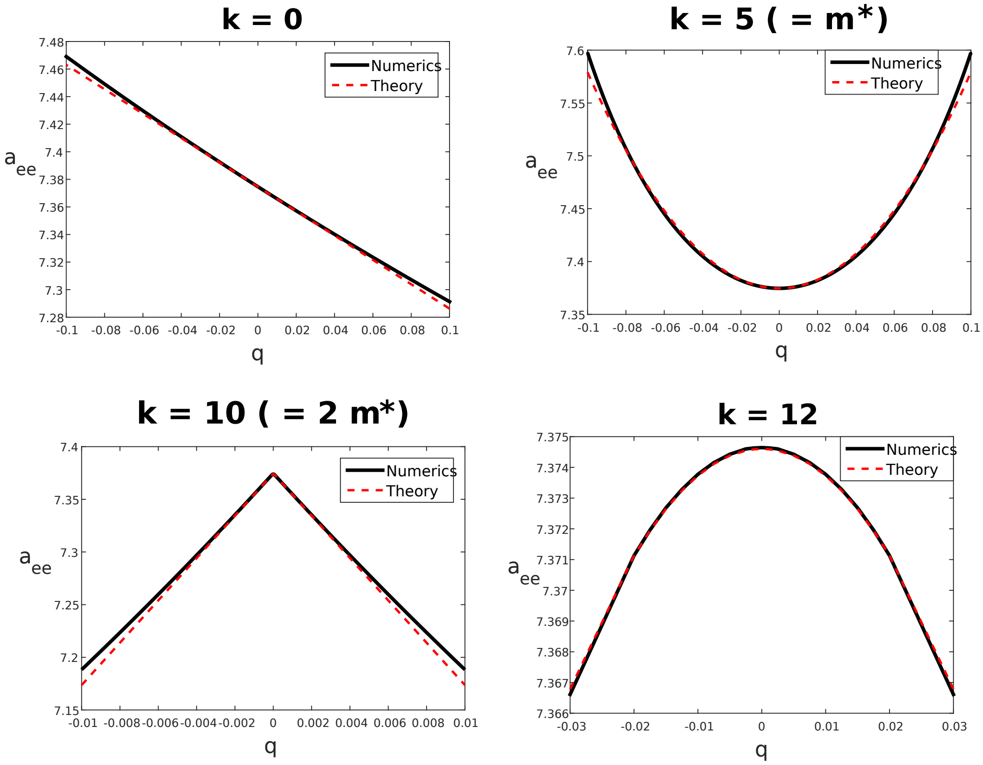

Figure 24: Comparison of theoretical and numerical stability analyses. Performing a perturbation calculation provides local stability curves (red dashed lines) near $(q,\ a_{ee}) = (0,\ a^*_{ee})$ that very closely match those computed numerically in Fig 19B for all $k \in P$. Note again that the pattern forming regimes locally lie above each curve; below these curves the steady state is stable. The calculation gives insight into the qualitative features previously observed; in particular, the high sensitivity observed to stimuli with $k = 0$ and $k = 2m^* = 10$ near a^*_{ee} come about due to the linear nature of these curves, as revealed through the calculation. The larger sensitivity to $k = 10$ compared to $k = 0$ is also predicted, as the $k = 10$ line has twice the slope of the $k = 0$ line. All other modes are shown to have locally quadratic curves that can be either concave up, so that such modes *stabilize* the steady state above but near a^*_{ee}, or concave down, in which case the stimuli *destabilize* the steady state below but near a^*_{ee}. Surprisingly, with this set of parameters, $k = m^* = 5$ is a *stabilizing* stimulus mode.

quadratic curves are concave down), the network may be more sensitive to nonzero wave numbers. Indeed, this was our intuition in setting $a_{ee} = 5.8$ (recall, $a^*_{ee} \approx 7.37$). However, even though most cases of PnSE are linked to patterns with modes in the 3–5cpd range, there are in fact rare cases where people have been documented to experience epileptic seizures when exposed to constant bright light [127]. Our model provides an explanation for such sensitivity .

3.3 DISCUSSION

Neuronal tissue can exhibit spatiotemporal resonances: either strong spatial responses to a narrow band of stimulus temporal frequencies, or strong temporal responses to a narrow band of stimulus spatial frequencies. Examples of the former include flicker phosphenes and photosensitive epilepsy, and were modeled in Rule et al. using a neural field approach [17]. In this paper we tackle the latter resonance, using a neural field model to capture resonant responses as observed in contrast sensitivity, pattern sensitive epilepsy, and visual discomfort [47]. Additionally, our model may encompass the rare condition of pattern sensitive epilepsy (PnSE) that occurs in response to bright, static, spatially uniform stimuli like constant bright lights [127]. Using biologically plausible excitatory spatial profiles result in model resonant modes that map directly onto the resonant modes psychophysically observed in the above-mentioned cases: 3–5cpd.

In addition to resonant responses to simple full-field stimuli in neural field models with periodic boundary conditions extended in one (i.e., on the ring) and two (i.e., on the torus) spatial dimensions, we found the network on the torus showed similar sensitivity to noisy stimuli with modes that were near the resonant wavenumbers found when simple cosine stimuli were used. Even though the individual dynamical behaviors for similar stimulus parameters (the dominant mode $|\mathbf{k}|$ and amplitude q) varied, the stability response was very similar, matching experimental findings in [47].

We also studied partial-field and dual-mode stimuli in the network on the ring. By making the network more excitable, as is believed to be the case in the brains of subjects with epilepsy, we found that partial-field stimuli could spread to the full network and show very similar resonance as in a healthy brain. Thus, we hypothesize that visual discomfort is associated with oscillatory activity, caused by spatially heterogeneous Hopf bifurcations, that stays localized to the area of cortex to which the stimulus projects, while PnSE is associated with such activity that further excites adjacent and other connected brain areas due to an increased neuronal excitability. We further hypothesize that contrast sensitivity is linked to these Hopf bifurcations, as the resonant modes excite the neuronal populations more than the nonresonant modes do, leading to higher peak (but stationary) firing rates before the Hopf bifurcations for the resonant modes.

We would be interested to see data of greater spatial resolution than is available thus far. Such detailed spatiotemporal data might match or be at odds with the dynamics observed in our model networks; either result would be welcome, leading to further work towards unlocking the mechanisms of spatiotemporal neuronal resonances, potentially leading to successful interventions of the less desirable consequences such as PnSE.

3.4 METHODS

3.4.1 *Model considerations*

To change the network Eq (3.1) from, e.g., using the firing rate formulation in Eq (3.2) to using the formulation in Eq (3.3), we change coordinates by letting $u(x,t) = \widetilde{u}(x,t) + f(-\widetilde{\theta}_e)$ and similarly for v. After some algebra and letting

$$\begin{aligned} h(w) &= \frac{1}{4}\ln\left(\frac{a_{ei}}{a_{ee}f(-w) + w - \theta_e} - 1\right) \\ g(w) &= a_{ie}f(-w) - a_{ii}f(-h(w)) + h(w) - \theta_i, \end{aligned} \tag{3.8}$$

we find that the new threshold $\widetilde{\theta}_e$ is given implicitly by setting $g(w) = 0$, while $\widetilde{\theta}_i = h(\widetilde{\theta}_e)$.

For the network on the torus, we simulate the network using formulation (3.2), while for the network on the ring, we use the formulation in (3.3) to facilitate our calculations. For both networks, we use equivalent parameters for comparison purposes (when $a_{ee} = 5.8$), using Eq (3.8).

As noted in Sec *3.1*, in experiments the occurrence of aberrant oscillatory neuronal activity is dependent on the spatial frequency of the stimulus image. Thus we wish for a range of input wave numbers to produce temporal oscillations and for the system to remain quiescent when input wave numbers outside of this range are given. We thus take our homogeneous system (i.e., that with no stimulus present) and set it near a so-called Turing-Hopf bifurcation, so that the homogeneous steady state bifurcates to a branch of stable limit cycles at a critical nonzero wave number, m^*. To do so, we linearize the system on the ring and find parameters that allow for purely imaginary eigenvalues at a nonzero mode m, or equivalently, for the trace of the linearized system to be zero with a positive determinant.

Our homogeneous system is:

$$\begin{aligned} \tau_e \; u_t(x,\; t) &= -u + f_e(J_{ee} * u - J_{ei} * v) \\ \tau_i \; v_t(x,\; t) &= -v + f_i(J_{ie} * u - J_{ii} * v), \end{aligned}$$

where $f_{e,i}(w)$ is given by Eq (3.3).

Our steady state solution is $(u_0, v_0) = (0, 0)$. Linearizing thus provides

$$\begin{aligned} \tau_e \; \bar{u}_t(x,\; t) &= -\bar{u} + \alpha_1(J_{ee} * \bar{u} - J_{ei} * \bar{v}) \\ \tau_i \; \bar{v}_t(x,\; t) &= -\bar{v} + \beta_1(J_{ie} * \bar{u} - J_{ii} * \bar{v}), \end{aligned} \tag{3.9}$$

where $\alpha_1 = f_e'(0)$ and $\beta_1 = f_i'(0)$. This is a linear autonomous system on a periodic domain, so that we obtain solution forms:

$$\begin{aligned} \bar{u}(x,t) &= \bar{u}_0 e^{\lambda t} e^{imx} \\ \bar{v}(x,t) &= \bar{v}_0 e^{\lambda t} e^{imx} \end{aligned}$$

This allows us to significantly simplify Eq 3.9 by using Fourier transforms to rid ourselves of the convolutions: e.g., $J_{ee} * e^{imx} = \widehat{J}_{ee}(m) \cdot e^{imx}$. We are then left with a simple 2×2 linear system, that, importantly, is parameterized by the wavenumber m:

$$\begin{gathered} \lambda \begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} = A \begin{bmatrix} \bar{u} \\ \bar{v} \end{bmatrix} \\ \text{where A } = \begin{bmatrix} \frac{-1+\alpha_1 \widehat{J}_{ee}(m)}{\tau_e} & \frac{-\alpha_1 \widehat{J}_{ie}(m)}{\tau_e} \\ \frac{\beta_1 \widehat{J}_{ei}(m)}{\tau_i} & \frac{-1-\beta_1 \widehat{J}_{ii}(m)}{\tau_i} \end{bmatrix} \end{gathered} \tag{3.10}$$

for $m \in \mathbb{Z}$. We now seek parameters to set the system near criticality by placing it by a Hopf bifurcation, where the eigenvalues cross the imaginary axis. That is, we find $a_{ee},\ a_{ie},\ a_{ei},\ a_{ii},\ \sigma_{e,i}$, and $\tau_{e,i}$ so that Det(A)>0, and Tr(A)=0 for some $m^* > 0$ (or in practice Tr$(A) \geq 0$ for a small interval $[m_l,\ m_r]$) and Tr(A)<0 for all other m >0. In particular, we fix all of the parameters except a_{ee} and $\sigma_{e,i}$. Increasing a_{ee} tends to have the effect of translating the trace curve upwards, so that we may adjust it until it hits the m axis (at which point $a_{ee} = a_{ee}^*$). The ratio $\frac{\sigma_e}{\sigma_i}$ tends to shift the peak of the determinant left or right, thereby determining m^*. For the choice of parameters shown below for the ring domain (except for the value of a_{ee}), the resultant disperison relation is shown in Fig 25. Note that $m^* \approx 5$ and $a_{ee}^* \approx 7.3741$. However, since the network can only experience bifurcations at whole wave numbers, we set $m^* = 5$ and provide a correction to the value of a_{ee}^* by increasing it until the curve of the determinant passes through the m-axis arbitrarily close to $m = 5$. We will see below in results that the corrected value of $a_{ee}^* \approx 7.3746$ matches that found numerically beyond single float machine precision. We also note that, since we have set the network up at a Hopf bifurcation at wavenumber $m^* = 5$, the network will lose the stability of the homogeneous rest state to oscillations at wavenumber 5 as a_{ee} increases beyond a_{ee}^*, so that at $a_{ee} = a_{ee}^*$, the network is poised at a Turing-Hopf bifurcation.

Unless otherwise noted in the text or methods, we use the following parameters for both networks (on the torus and on the ring):$a_{ee} = 5.8,\ a_{ei} = a_{ie} = 10,\ a_{ii} = 8,\ \tau_e = 3,\ \tau_i = 6.6$. For the torus domain, we use $\sigma_e = 8,\ \sigma_i = 16,\ \theta_e = -1.07,\ \theta_i = -0.3597$, while for the ring domain, we use $\sigma_e = 1.75,\ \sigma_i = 3.5,\ \theta_e = 0.518,\ \theta_i = 0.311$. We note that a_{ee} is well below the critical value a_{ee}^* found above; we explain the reasoning for this value in Sec *3.2.2.3*.

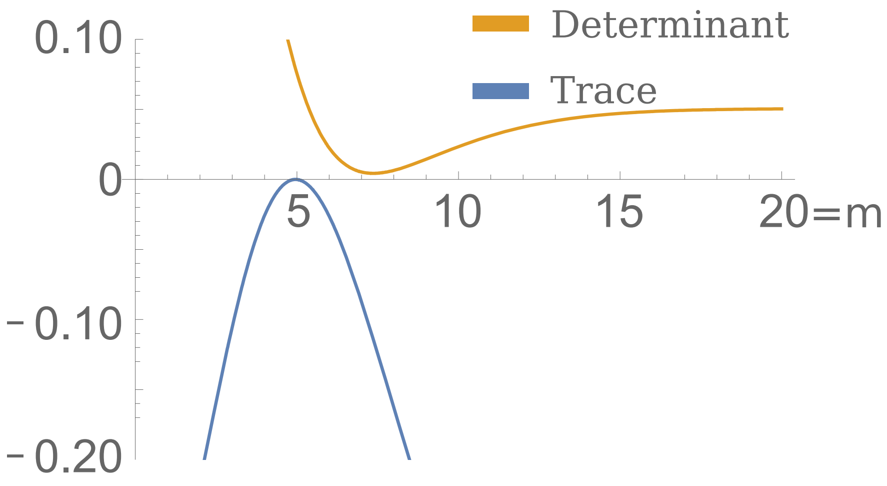

Figure 25: Dispersion relation for Eq (3.10). Since, for a 2×2 linear system as for A in Eq (3.10) the eigenvalues are given by the trace, T, and the determinant, D, as $-T/2 \pm \sqrt{T^2/4 - D}$, we can see the parameters have been chosen so that the system will have purely imaginary eigenvalues for $m = 5$ and be stable for other m.

3.4.2 *Physical dimensions*

As mentioned above, the apparent band of sensitive wavenumbers as measured psychophysically in several contexts is ≈ 3 cycles per degree (cpd). To see if our network models provide biologically plausible results, we use an approximation of the retinotopic map for foveal vision to find the cpd that corresponds to a particular stimulus mode k [125]:

$$X(\epsilon) \approx 12 \ln(1 + \frac{\epsilon}{\epsilon_0}), \tag{3.11}$$

where $X(\epsilon)$ corresponds to the cortical distance in mm as a function of the viewing angle subtended between two points, ϵ, and ϵ_0 is 1° (thus, ϵ/ϵ_0 is simply dimensionless degrees). For stripes that are 3cpd apart, $\epsilon/\epsilon_0 \approx 0.33$. Since $\sigma_e = 8$ for the network on the torus and most of the area covered by the gaussian is within $\sqrt{2}\sigma_e$ units of its peak, fixing the size of the domain to be 52×52mm implies that the $\sigma_e = 52\text{mm}/\frac{512}{\sqrt{28}} \approx 1.15\text{mm}$, matching empirical measurements for the spatial profile of pyramidal cells in layers 2/3 [126]. We find below in results that the most sensitive stimulus mode is around $k = 15$, so that the cortical distance between stripes is $\frac{52\text{mm}}{15} \approx 3.47\text{mm}$. Inverting Eq 3.11 above then provides the corresponding cpd for $k = 15$: $\exp \frac{52\text{mm}}{15 \cdot 12\text{mm}} - 1 \approx 0.33 \approx 3\text{cpd}$. A similar calculation for the network on the ring, fixing the length of the ring at 35mm and using $\sigma_e = 1.75$ implies that the physical dimensions of σ_e are approximately 1.44mm, and the sensitive stimulus mode of $k = 10$ that we find again corresponds to $\approx 0.33 \approx 3\text{cpd}$. Thus, biologically plausible spatial profiles for the excitatory populations map the resonances that we find in the model networks very closely to those found in psychophysical experiments.

3.4.3 *Methods for the network on the torus*

We simulated a network of 512×512 excitatory and 512×512 inhibitory populations with periodic boundary conditions that evolve according to Eq (3.1) and (3.2). We found that by performing the simulation on custom GPU code using CUDA C on an NVIDIA GTX970, that we were able to run the simulations in real-time, changing the parameters on the fly to observe the effects of, e.g., different coupling strengths, stimulus modes and amplitudes. In fact, we found this code to be ≈ 3.5 times faster than the identical code written purely in C. We used a simple Euler stepping scheme with $\Delta t = 0.1$ (we found the dynamics to be consistent with those found with Δt as low as 0.01).

The simulations are fast enough to systematically test for different network behaviors with different parameters; i.e., we are able to create an approximate bifurcation diagram by running the network for a sufficiently long time and determining whether the network is still oscillating or has died down towards a steady state. The parameters of interest were those of the stimulus: the amplitude, q, and the mode, k.

We determine the state of the network (oscillatory or not) by measuring the variance of the excitatory populations of the network (letting $U(t) = [u_0(t), \ ..., \ u_{N^2}(t)]^T$) after some transient time t_{trans} (letting $t \geq t_{trans} + 1ms$ to avoid terms that can be excessively large near t_{trans}:

$$\mathrm{Var}(U(t)) = \frac{1}{N^2} \sum_{j=1\ldots N^2} \left(\frac{1}{t - t_{trans}} \sum_{t>t_{trans}} u_j(t)^2 - \left(\frac{1}{t} \sum_{t>t_{trans}} u_j(t) \right)^2 \right) \tag{3.12}$$

Of course, the averages in the continuous setting would instead involve integrals over the time interval of interest. Plotting the variance as done in Sec *3.2.1.1* generally leads to very visible differences between states that settle down towards a steady state and those that continue oscillating if t_{trans} and t were large enough. To determine bounds that seemed to work well for our purposes, we created an interactive bifurcation diagram that allowed us to continue running the network from the last integration time step, so that we could determine whether or not the network was still in a transient state. We found that a transient time of about 16s and a recording time of about 1s allowed for robust determination of the network state. We note that this "eyeball" metric worked well in part because the Hopf bifurcations were subcritical, often leading to either large oscillations or none at all.

Using this method, we probed the sensitivity of the network by beginning with a simple cosine stimulus that spatially varies horizontally, corresponding to vertical stripes (using x as the abscissa and y as the ordinate, the stimulus to the excitatory population is simply $q \cos \frac{2\pi k x}{N}$). For each wave number k, we increased the amplitude, q, up to a value of 2. For a contiguous set of $(k,\ q)$ values, the steady state destabilized, resulting in spatially heterogeneous, temporally oscillatory patterns. We first simulated the network using a coarser Δq, and then reduced Δq near the onset of oscillatory activity to find the spatiotemporal pattern formation boundaries as a function of k and q. To determine these lower boundary boundaries of the q vs k bifurcation diagrams, we reduced Δq to 0.05 and stipulated a threshold variance so that for the network is defined to be in an oscillatory regime only if $\mathrm{Var}(U(t)) \geq \mathrm{Var}_{thresh}$. Once again, though Var_{thresh} is necessarily arbitrary, we found, again by testing with the interactive bifurcation diagram, that a value of 0.001 corresponded well with the existence or nonexistence of widespread oscillatory patterns. Of course, the term "widespread" suggests the additional problem that $\mathrm{Var}(U(t))$ is a *global* measure of the network's activity, since we are interested in whether the entire network might be thought of as being in a pattern-forming regime. Thus, in some cases, especially when we project noisy stimuli onto the network (discussed below), the variance might be globally low but locally high. As there exists no principled reason to make any particular value or spatial constraints for Var, we found this heuristic "eyeball" method to be warranted and sufficient for the present study.

3.4.3.1 ***Noisy stimuli methods*** We used the same method as outlined above to examine the network's responses to noisy stimuli. We created the stimuli in MATLAB by first creating 512×512 stimuli of Gaussian white noise with variance 1, as seen in Fig 26A. We then filtered the stimulus by first putting it in Fourier space (Fig 26B), where circles of radius p pixels correspond to wavevectors $\mathbf{k}$ with wavenumber $|k| = p$. The angle from the center to the pixel indicates the direction of the wavevector $\mathbf{k}$. We note that since the network only responds to stimuli with $k < 30$, the presence of high spatial frequencies tends to cause the network to die down to a steady-state. Thus, we apply a low-pass filter in Fourier space by using a circular logistic function (like the firing rate function): $F(r) = \frac{1}{1+\exp -g(r-r_0)}$, and used the gain $g = 0.1$. r_0 defines the cutoff radius (i.e., wavenumber) of the filter (Fig 26B, bottom). We used $r_0 = 70$ to obtain our results below (note, the maximum wavenumber of the system, ≈ 362, is found by dividing the the number of elements along the diagonal of the stimulus by 2. We then increased the contribution of a particular wavevector $\mathbf{k}$ with wavenumber $|\mathbf{k}| = p$ by multiplying all of the pixels that were p units away from the center by a parameter A; we used $A = 50$ for the results below. We then renormalized the Fourier transform to have the same energy as the original transform (and stimulus), and finally inverse-Fourier-transformed the stimulus and examined the power spectrum (Fig 26C). Using the same random image as a seed, we applied this algorithm for all $p \in \{0,\ ...,\ 30\}$, forming a set of stimuli that we could then input to our system, adjusting the overall stimulus q and the dominant mode $|\mathbf{k}|$ just as we did for the simple cosine stimulus.

We then generated a new set of stimuli for all $p \in \{0,\ ...,\ 30\}$ from a newly-generated random stimulus in order to check for repeatability in the network's response to such random images. In order to ensure that we would not obtain similar responses due to similarities (i.e., correlations) in the random components of the stimuli, we generated a random stimulus that had a very small (< 0.001) Pearson correlation coefficient with the original stimulus. The Pearson correlation coefficient $c(S, T)$ of the two stimuli $S,\ T$ was computed as

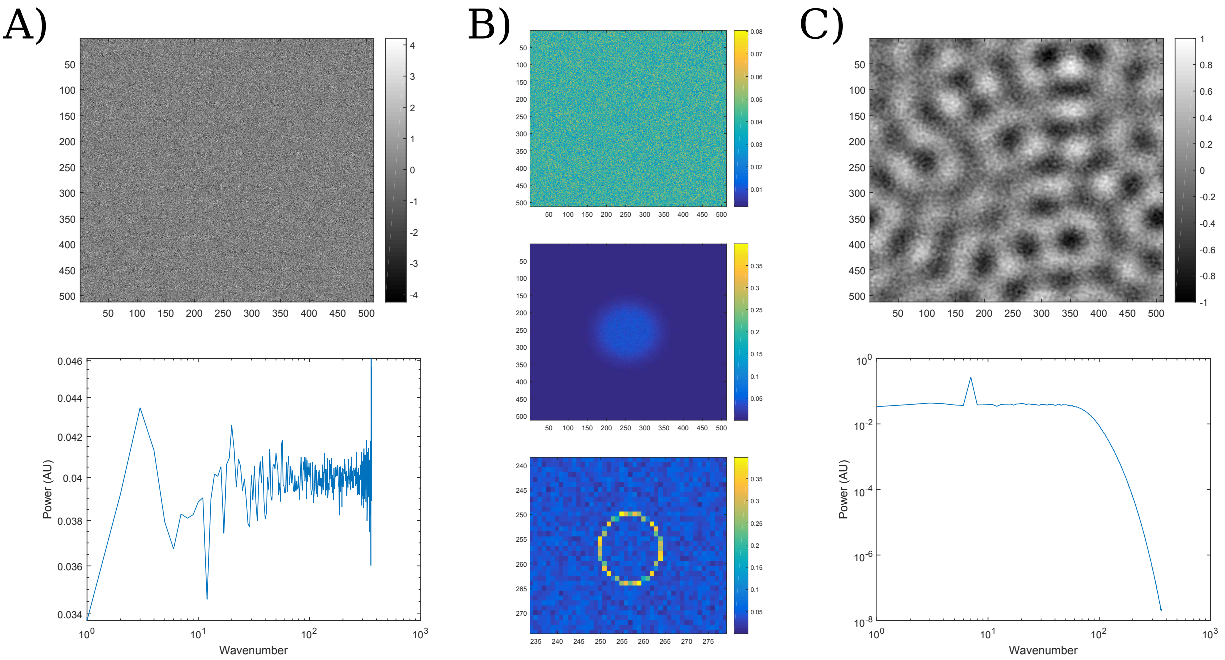

Figure 26: Construction of noisy images with selected dominant modes. To test more complex stimuli, random images were created and filtered to have dominant wavenumbers. (A) Top: A random matrix with 512 × 512 was generated, with values sampled from $\mathcal{N}(0, 1)$. Bottom: The radially-averaged power spectrum shows an even power distribution across the different wavenumbers. (B) The image is transformed into (shifted) Fourier space (top), and high frequencies are reduced by pointwise-multiplying the elements by a cirular logistic function (middle). Finally, elements that are a distance p from the center, which correspond to a wavenumber of p, are multiplied by a chosen amount to increase their relative contributions (bottom). (C) Top: The final image that is fed into the network, after normalizing it to have the same energy as the original image in (A). Bottom: the radially-averaged power spectrum of the transformed image shows the low-pass filter and increase in amplitude for the mode 7.

$$c(S,\ T) = \frac{\mathrm{Cov(S,\ T)}}{\sqrt{\mathrm{Var}(S)\mathrm{Var}(T)}} = \frac{\langle (S - \mathrm{M}(S))(T - \mathrm{M}(T)) \rangle}{\sqrt{\mathrm{Var}(S)\mathrm{Var}(T)}}, \tag{3.13}$$

where $\mathrm{M}(S)$ is the mean of S and the averages indicated by the brackets are taken over all of the elements of S and T. Thus, $0 \leq c(S,\ T) \leq c(S,\ S) = 1$ and $c(S,\ T) = 0$ if S and T are uncorrelated.

3.4.4 *Methods for the network on the ring*

While studying the network on a ring allows us to employ the bifurcation software AUTO, we encounter some limitations. In particular, including too many populations make the bifurcation computations too slow to be useful; thus, we limit N to be less than 100 (N excitatory and N inhibitory populations, corresponding to $2N$ ODEs). Since we must limit N to a lower number, the spatial discretization in simulations and bifurcations is a much larger potential issue that could lead to spurious results. In particular, we need to ensure that the network with stimulus of spatial period T_0 is symmetric under shifts of the populations by T_0 populations. This condition corresponds to stipulating for k to evenly divide N. By choosing N to be 60, we simultaneously allow for a computationally-feasible network size and for a large number of permissible stimulus wavenumbers: $P = \{0, 1, 2, 3, 4, 5, 10, 12, 15\}$. We do not include wavenumbers above 15 due to discretization effects that occur by making k too large. In particular, for $k = 20$, the symmetry of the stimulus is lost: Since every third population must receive the same stimulus when $k = 20$, one of these, say u_0, will receive a stimulus value of q, while u_1 and u_2 will receive a stimulus value of $-q/2$. For $k \geq 30$, the stimulus can no longer be reasonably approximated by a cosine: for $k = 30$, the excitatory populations receive the values of $\pm q$, while $k = 60$ is equivalent to $k = 0$.

3.4.4.1 ***Partial-field stimuli methods*** In order to test whether the network could model the spread of excitatory activity often associated with epileptic seizures, we limited the stimulus to project to only approximately 2/3 of the populations (39 excitatory and 39 inhibitory populations in a contiguous interval — in particular, populations 41–59 and 0–18; thus, the stimulus-free region corresponds to the middle third of the figures as indicated).

We then increased the excitability of the network by letting $a_{ee} = 7 < a_{ee}^*$ and found that spatiotemporal patterns that formed in the populations to which the stimulus projected (the *stimulus region*) could then spread to form spatiotemporal patterns in the populations to which the stimulus did not project (the *stimulus-free region*). Moreover, these patterns would often persist upon removal of the stimulus (i.e., setting $q = 0$ and letting the system continue to evolve), as can occur in PnSE following the removal of the stimulus. Thus, the patterns in which we are interested involve large-amplitude oscillations, so that we neglect small-amplitude oscillations.

To systematically test the network's behaviors for different stimulus modes k, we fixed a q value and integrated the system forward for 2000ms from uniform random initial conditions on the small interval [-0.01, 0.01]. Using a step size of $\Delta\ q = 0.05$, we then found the smallest $q(k)$ value that allowed for the pattern to spread, defined as the smallest q value that exhibited spreading of activity into the stimulus-free region for at least 1 of 5 attempts of integrating 2000ms with different random seeds. To determine if oscillatory activity of the network would persist and to what mode it would evolve, we then removed the stimulus by letting $q = 0$ and integrated forward (beginning from the final values of an integration where the activity had spread to the stimulus-free region — except where noted in the text in Sec *3.2.2.4* for another 2000ms and examined the network over the final few hundred milliseconds.

We found that the network was very sensitive to constant stimuli (i.e., stimuli with wavenumber 0), requiring a smaller amplitude q to induce oscillations than for nonzero wavenumbers. Since such a high sensitivity to constant stimuli does not match most data obtained on pattern-sensitive epilepsy or visual discomfort, this introduces an additional constraint on our parameters. That is, a_{ee} must be sufficiently less than a_{ee}^* so that (1) the network is sensitive to a band of stimuli with nonzero wavenumbers; (2) the network is relatively insensitive to stimuli of wavenumbers outside of this sensitive band, including constant stimuli.

We note that the stimulus amplitude to the excitatory populations is q, and that to the inhibitory populations is $r\,q$. Since we vary q in our investigations while maintaining r at a fixed value, we will hereafter refer to q as the amplitude of the stimulus.

3.4.4.2 ***Multiple stimuli methods*** Stimuli tend to be composed of multiple spatial frequencies. To gain traction on how stimuli of different frequencies affect the network's stability, we examine how the network on the ring responds to just two stimuli of possibly different amplitudes. Even this question is very broad; since P has 10 possible values there are $\binom{10}{2} = 45$ possible combinations. Recall that the network is most sensitive to $k = 2m^*$ (=10 here). Since we are most interested in stimulus modes to which the network is especially sensitive, we here study how the modes k_1, for $k_1 \in P$, and $k_2 = 2m^* = 10$ interact. Thus, the stimulus that we provide is

$$q_1 \cos 2\pi k_1 x/N + q_2 \cos 2\pi 10x/N \tag{3.14}$$

To study how the interaction of stimuli of these two modes affect the network, we track the stability of the steady state by following the Hopf bifurcation at $q_*(k_j)$ as a function of $q_*(k_{3-j})$ for $j = 1,\ 2$. For example, $q_*(10) \approx 0.18$, and $q_*(5) \approx 0.71$; we can then find the curve of Hopfs $q_*(5,\ 10) = 0$ by following either $q_*(5)$ as a function of $q_*(10)$ or vice versa. We can then determine how the addition of a stimulus of mode k_1 affects one of k_2, and vice versa.

4.0 CONCLUSION

Here we provide a brief review of the contents of this dissertation.

In Chap 1, we explored evidence of oscillations of populations of neurons in different contexts: First, we looked at oscillatory activity that arises in scenarios involving working memory and neural binding; second, we saw evidence of spatially-resonant oscillatory responses to static, spatially patterned stimuli that arose in situations spanning normal contrast sensitivity, visual discomfort, and pattern-sensitive epilepsy.

Our overarching thesis is that such large-scale neuronal oscillatory activity can be explained within the framework of mean-field neuronal models, in particular through the mechanisms provided by Hopf and Turing-Hopf bifurcations. In Chap 1, we examined minimalistic examples of dynamical systems in which these bifurcations arise. Moreover, since saddle-node bifurcations of periodic orbits often coincide with subcritical Hopf bifurcations in neural field networks, our simple Hopf example was constructed to keep solutions bounded, leading to this secondary bifurcation that, importantly, gives rise to stable limit cycles.

In Chap 2 and Appendices A, B, C, and D, we introduced and explored the dynamics that can occur in a neural field model that includes the activities of NMDA synapses. These synapses take longer to activate and decay, possibly leading to the persistent activity seen in delayed-response working-memory tasks. We derived a Wilson-Cowan-type model based on simple models of AMPA, GABA, and NMDA synaptic activity and found biologically-plausible parameter regimes that allowed for bistability between low and high states, both steady-state and oscillatory.

When several populations were coupled together, the oscillatory activity allowed for much richer dynamics that may be relevant to working memory. In particular, we found that the extra information available in the phase of the oscillations allowed for a distinction in the dynamics that can't obtain in steady-state models. For example, two populations could oscillate synchronously or asynchronously. With reasonable parameter values, we found that up to three populations could oscillate out-of-phase with respect to each other, or several more than that could oscillate in-phase with each other. The dynamics were otherwise limited only combinatorically, so that several groups of populations oscillated in such a way that populations within a group oscillated synchronously, while populations between groups oscillated asynchronously. Such dynamics could correspond to bound and distinct items in memory, respectively.

By following these oscillatory solutions with respect to important network parameters such as the synaptic timescales and interpopulation coupling strengths, we showed that these biophysically germane activities could be obtained with varying ranges of plausible parameter values. We also looked at how the dynamics changed as these parameter values varied, and found intuitive rationales for why bifurcations arose, indicating, e.g., why oscillations were possible for some parameter regimes but not others. We gained insight by looking at the network in the weak-coupling limit. For example, different dynamical states were more or less accessible from other states in a way that depended directly on the length of the boundary that separated the respective basins of attraction.

In Chap 3 and Appendices E and F, we looked at how we might model neuronal populations that exhibit spatially resonant responses that arise in contrast sensitivity, visual discomfort, and pattern-sensitive epilepsy to static stimuli with wavenumbers near 3–5 cycles per degree (cpd). We proposed that so-called Turing-Hopf bifurcations could provide a mechanism that leads to the loss of stationary activity to spatially and temporally varying responses. By linearizing our spatially extended network, we found parameters that allowed for a pair of purely imaginary nontrivial complex-conjugate eigenvalues to exist at a nonzero wavenumber, m^*. By simulating the network, we recorded the network's spontaneous and stimulus-induced dynamics.

Since cortex can be approximated as a sheet of interacting neurons, we first simulated the network on the torus; i.e., extended in two spatial dimensions with periodic boundary conditions. To obtain enough data for these simulations to be of use required a way to rapidly integrate such networks forward in time. Making use of developments in general-purpose

GPU computing, we created code in CUDA C that could simulate networks composed of 512×512 excitatory and 512×512 inhibitory populations quickly enough to see dynamics emerge in real time. The performance achieved was sufficient to compute bifurcation diagrams by sweeping over sets of parameter values, simulating the network for fixed values and measuring the variance after transient dynamics had passed to determine the oscillatory state of the network.

We found that spontaneous oscillations with wavenumbers near m^* arose once the recurrent excitation (a_{ee}) crossed a threshold value (a_{ee}^*). Striped stimuli consisting of simple cosine functions resulted in spatial resonances that matched the psychophysically determined sensitive band of 3–5cpd at a_{ee} values below this threshold. Interestingly, the most sensitive stimulus wavenumbers were near $2m^*$. This resonance carried over to more complex stimuli that were created by filtering noisy two-dimensional stimuli to exhibit a mix of lower-frequency modes and one dominant mode whose contribution was amplified. These stimuli looked like somewhat unstructured, noisy lattices. Such noisy images whose spatial components in the 3–5cpd range were amplified had been found to be more aversive to human subjects than corresponding images without such amplification. Thus, we found a set of behaviors in the network on the torus that matched motivating empirical results.

To better understand the behaviors we observed, we modeled the network on the ring (i.e., extended in one spatial dimension with periodic boundary conditions), allowing us to gain more mathematical traction on the system. In particular, this setting allowed us to characterize the network behaviors based on the bifurcations that arose, using numerical and analytical methods. XPP-AUTO allowed us to numerically compute bifurcation values, showing that the network's instabilities indeed arose as a result of Hopf and Turing-Hopf bifurcations.

Following these bifurcations as functions of the recurrent excitation, a_{ee}, and the stimulus amplitude, q, provided local stability curves and showed that near the onset of instability, at $(q,\, a_{ee}) = (0,\, a_{ee}^*)$, the behavior of the network varied widely across stimulus modes. Stimuli with modes 0 and $2m^*$ were found to show the highest local sensitivities, exhibiting linear stability curves near onset. This matched our findings in one and two spatial dimensions, explaining in part why we had to decrease a_{ee} to sufficiently low values to better match empirical results that indicate that the visual cortex is less sensitive to constant inputs. Stimuli with other spatial frequencies produced concave up or down stability curves. A perturbation calculation near the onset of instability mathematically revealed why the 0 and $2m^*$ stimulus modes produced linear stability curves while the other modes produced quadratic stability curves. Moreover, the calculation provided plots that very closely matched the numerically-computed graphs near the instability.

Since seizures are thought to occur by the spreading of local instabilities across neuronal networks, we tested the responses of the network on the ring to partial-field stimuli. We found that at higher levels of recurrent excitation that were still below threshold (i.e., $a_{ee} < a_{ee}^*$), large-amplitude oscillatory activity that was triggered in the stimulus region spread across the entire medium in a way that showed the same resonance pattern as for the full-field stimuli. Once this activity had spread, the network oscillations generally persisted following the removal of the stimulus. This lends support to proposed connective threading between visual discomfort and pattern-sensitive epilepsy: If the neuronal network is not very excitable, stimuli only cause local spatiotemporal activity, while if the network is more excitable, this local activity around the neurons to which the stimulus projects can spread across the network to regions that receive no such stimulus inputs. Since the network continues to oscillate upon removal of the stimulus, this also demonstrates the network is bistable or multistable, as has been proposed. Indeed, following spatiotemporal patterns with wavenumbers near m^* in AUTO that tended to match the persistent activity we observed confirmed the existence of regions of multistability between a stable resting state and large-amplitude spatiotemporally patterned activities that arose as subcritical Turing-Hopf bifurcations.

Finally, since most real images are composed of many wavenumbers, we studied how the network responded to such mixed-mode stimuli. As this is a multifaceted question that can lead to a large variety of explorations, we narrowed our study to how different wavenumbers impacted the stability of the network to the most sensitive stimulus wavenumber, $2m^*$. Most of the responses of the network to such mixed stimuli were as expected; e.g., increasing the contribution of the $2m^*$ mode to other modes generally made the network more sensitive, while increasing the contribution of other modes to the $2m^*$ mode generally made the network less sensitive. However, unexpected behaviors arose as well; for example, increasing the contribution of the $2m^*$ mode to stimuli with modes 2 and 12 could in fact *decrease* the sensitivity of the network. Further study of such mixed-mode stimuli could reveal interesting network dynamics that might be applicable to real-world contexts. For example, while the sensitivities to stimuli with one dominant mode has been well documented, to our knowledge sensitivities to stimuli with multiple dominant modes has not been explored. Guidelines that preclude the appearance of images with dominant wavenumbers within the 3–5cpd range in video and still images, for example, might be revisited to factor in other mixed-mode stimuli as well.

APPENDIX A

MEAN FIELD MODEL MOTIVATION AND SPIKING NETWORK COMPARISON

Consider a population of N all-to-all coupled excitatory and inhibitory noisy quadratic integrate-and-fire (QIF) neurons. For simplicity, we use the same number of excitatory and inhibitory neurons. The network has the form:

$$\begin{aligned}\frac{dV_{e,j}}{dt} &= I_{e,j} + \sigma_e \xi_{e,j}(t) + V_{e,j}^2 \\ \frac{dV_{i,j}}{dt} &= I_{i,j} + \sigma_i \xi_{i,j}(t) + V_{i,j}^2,\end{aligned} \tag{A.1}$$

where ξ are independent white noise processes and the drives include the synaptic interactions and inputs, $I_{e,i}^0(t)$. These inputs are given as:

$$\begin{aligned}I_{e,j} &= I_e^0(t) - \theta_e + a_{ee}\frac{1}{N}\sum_{k=1}^{N} u_k - a_{ei}\frac{1}{N}\sum_{k=1}^{N} v_k + a_{en}h(V_{e,j})\frac{1}{N}\sum_{k=1}^{N} n_k \\ I_{i,j} &= I_i^0(t) - \theta_i + a_{ie}\frac{1}{N}\sum_{k=1}^{N} u_k - a_{ii}\frac{1}{N}\sum_{k=1}^{N} v_k + a_{in}h(V_{i,j})\frac{1}{N}\sum_{k=1}^{N} n_k,\end{aligned}$$

where the variables and parameters are as described in Sec 2.2. For the QIF model, when $V(t^-) = +\infty$, it is reset to $V(t^+) = -\infty$. (In simulations, we replace $\pm\infty$ by ± 100.) We use current-based interactions rather than conductance-based as they are easier to reduce to a mean-field [118]. The NMDA current (subscripted n) has an additional voltage dependence [120]:

$$h(V) = 1/(1 + \exp(-0.062(V - 60))[Mg]/3.57)$$

where $[Mg]$ is the extracellular magnesium concentration (set to either 1 or 0 in this paper). (We note that the voltage is shifted by +60, since our simple QIF model has a resting potential at around 0 as opposed to -60.) The synaptic gating variables for AMPA (u_j) and GABA (v_j) obey simple first-order dynamics:

$$\begin{aligned} \tau_e u_j' &= -u_j + \zeta \sum_k \delta(t - t_{jk}^e) \\ \tau_i v_j' &= -v_j + \zeta \sum_k \delta(t - t_{jk}^i) \end{aligned}$$

where ζ is a constant that we will choose to match the simulations and $t_{jk}^{e,i}$ are the k^{th} spikes of the j^{th} neuron. We use a simple first-order voltage-gated model of Golomb et al. [119] for the NMDA synapses:

$$\frac{dn_j}{dt} = -n_j/\tau_n + a_n/(1 + \exp(-(V_{e,j} - V_{th})/V_{shp}))(1 - n_j) \qquad \text{(A.2)}$$

with $\tau_n = 144, V_{th} = 60, a_n = 7, V_{shp} = 5$.

Since coupling is all-to-all, we let $\bar{u} = (1/N)\sum_j u_j$, $\bar{v} = (1/N)\sum_j v_j$, and $\bar{n} = (1/N)\sum_j n_j$. Thus, for the AMPA and GABA we get:

$$\begin{aligned} \tau_e \bar{u}' &= -\bar{u} + \zeta(1/N) \sum_{j,k} \delta(t - t_{jk}^e) \\ \tau_i \bar{v}' &= -\bar{v} + \zeta(1/N) \sum_{j,k} \delta(t - t_{jk}^i). \end{aligned}$$

We observe that in the limit as N gets large, the sums approach the mean firing rate of the neurons in each population given the total inputs, I_e, I_i. For the NMDA, we obtain:

$$\frac{d\bar{n}}{dt} = -\bar{n}/\tau_n + \frac{1}{N} \sum_j a_n(1 - n_j)/(1 + \exp(-(V_{e,j} - V_{th})/V_{shp}))$$

At this point, we make one of our main approximations. Since function $R(V) = 1/(1 + \exp(-(V - V_{th})/V_{shp}))$ is close to 0 when the neuron is not firing, but grows rapidly when it is firing, we will approximate its average by the average excitatory activity, $\bar{u}$ raised to a power $p > 1$ to account for the threshold-like behavior and, thus, obtain the following:

$$\frac{d\bar{n}}{dt} = -\bar{n}/\tau_n + a_n C_2 \bar{u}^p (1 - \bar{n})$$

with C_2 chosen to match the amplitude of $\bar{n}$ for the spiking model.

To obtain the firing rate, consider the mean first passage time for a noise driven QIF model,

$$dV = (V^2 + I)dt + \sigma dW, \qquad V(0) = -\infty,$$

where σ is the noise and I is the drive. The expected time $T(I, \sigma)$ for $V(t)$ to reach $+\infty$ leads to the expected firing rate, $\nu(I, \sigma) = 1/T$. For the zero noise case, $\nu = \sqrt{[I]_+}/\pi$, where $[I]_+$ is the positive part of I. With noise, this rate can be closely approximated by the nonlinearity:

$$f(I) = \frac{1}{\pi}\sqrt{I/(1 - \exp(-\beta I))},$$

where β is chosen to best fit for a given σ [72].

With this approximation of the noisy firing rate, we get the approximate equations for $\bar{u}, \bar{v}$:

$$\begin{aligned} \tau_e \bar{u}' &= -\bar{u} + \zeta f(I_e) \\ \tau_i \bar{v}' &= -\bar{v} + \zeta f(I_i), \end{aligned}$$

where

$$\begin{aligned} I_e &= I_e^0(t) - \theta_e + a_{ee}\bar{u} - a_{ei}\bar{v} + a_{en}\bar{n}\bar{h}(V_e) \\ I_i &= I_i^0(t) - \theta_i + a_{ie}\bar{u} - a_{ii}\bar{v} + a_{in}\bar{n}\bar{h}(V_i), \end{aligned}$$

where $\bar{h}(V_e) = (1/N)\sum_j h(V_{e,j})$, and similarly for $\bar{h}(V_i)$. We can write down the mean-field formulation for the dynamics of $\bar{u}$, $\bar{v}$, and $\bar{n}$ much more easily if their dynamics only depend on these variables, and not the voltages. We would therefore prefer to approximate the functions $\bar{h}(V_{e,i})$ by writing them in terms of $\bar{u}$, $\bar{v}$, or $\bar{n}$, if possible. If we plot $\bar{h}(V_e)$ against $\bar{u}$, we see in Fig 27A that (1) $\bar{h}(V_e)$ does not vary much, taking on values from ≈ 0.05 to ≈ 0.2 (whereas $h(V_{e,j})$ ranges from nearly 0 when $V_{e,j}$ resets to -100 all the way to nearly

1 just before $V_{e,j}$ reaches the spiking threshold value of +100) and (2) $\bar{h}(V_e)$ might be well approximated to first order as an affine function of $\bar{u}$. In fact, due to (1), we might in fact approximate $\bar{h}(V_e)$ to zeroth order; i.e., let $\bar{h}(V_e)$ be approximated by a constant, C_1. In Fig 27B, we see that the average value of $\bar{h}(V_e)$ is approximately 0.1, so we set $C_1 = 0.1$ (red dashed line in Fig 27A and B). We note that since the timescales of the excitatory and inhibitory membrane constants are both 1 in Eq A.1, $\bar{h}(V_i)$ exhibits the same dynamics that $\bar{h}(V_e)$ does, so that we can use the same constant, C_1, to approximate this function as well.

If we choose $\zeta = \pi$, then, the equations for $\bar{u}, \bar{v}$ are exactly the same as those that we have analyzed throughout the paper. Putting all these parts together, we obtain the mean-field model of the paper:

$$\begin{aligned}
\tau_e \bar{u}' &= -\bar{u} + \zeta f(I_e^0(t) + a_{ee}\bar{u} - a_{ei}\bar{v} + C_1 a_{en}\bar{n} - \theta_e) \\
\tau_i \bar{v}' &= -\bar{v} + \zeta f(I_i^0(t) + a_{ie}\bar{u} - a_{ii}\bar{v} + C_1 a_{in}\bar{n} - \theta_i) \\
\bar{n}' &= -\bar{n} + C_2 a_n \tau_n \bar{u}^p (1 - \bar{n})
\end{aligned}$$

where the two constants, C_1, C_2 come from our approximations of the voltage dependences in the NMDA synapses and currents.

Fig 28 shows a simulation of the spiking model with 200 excitatory and 200 inhibitory cells and all parameters as in Chap 2 except as indicated in the text and figure caption in this appendix. The shapes and amplitudes are quite close, although the frequency is faster in the mean field than in the spiking model. We have shown in this appendix that we can find a spiking model that will generate the same dynamics as our Wilson-Cowan type system. Furthermore, we have given a heuristic (via two approximations) way to incorporate NMDA into the mean field model that closely matches the dynamics of the spiking model on which it is based.

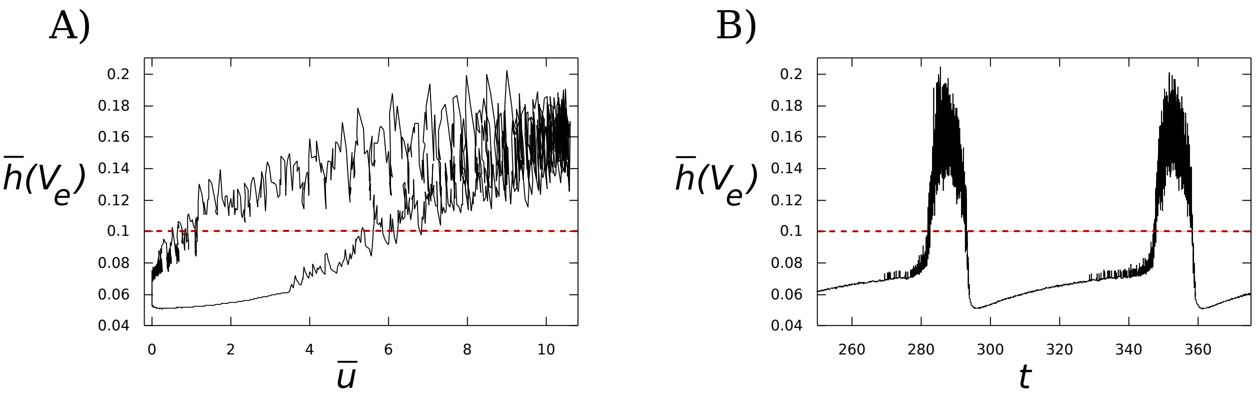

Figure 27: Average voltage activation function for NMDA receptors. During one oscillation of the network, $\bar{h}(V_e)$ traces out a path with respect to $\bar{u}$ in (A) that, to first order, can be approximated as an affine function of $\bar{u}$. However, since the range of $\bar{h}(V_e)$ is only $\approx (0.05,\, 0.2)$, we choose here to approximate it to zeroth order as a constant, C_1. In (B), we see the average value of $\bar{h}(V_e)$ is ≈ 0.1, and so we set $C_1 = 0.1$, indicated by the red dashed line in both figures. Since the inhibitory voltages have the same membrane time constant and show the same dynamics as the excitatory voltages, we can approximate $\bar{h}(V_i)$ with the same value.

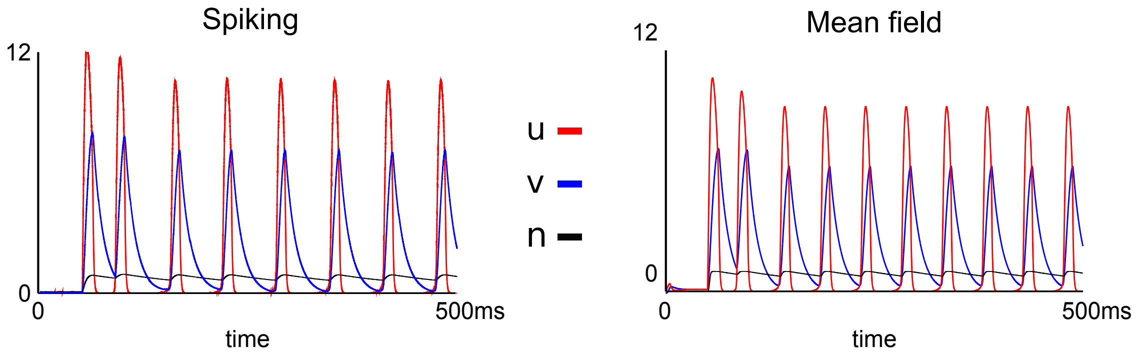

Figure 28: Comparison of the spiking model with the mean field model. Left: The average of u_j, v_j, n_j for the full spiking model with $[Mg] = 1mM$, $a_{en} = 40$, $a_{in} = 1$, $\zeta = \pi$, $a_n = 7$. Stimulus is a 50ms pulse given at $t = 50$. Right: The mean field model with $C_1 a_{en} = 4$, $C_1 a_{in} = 0.1$, $C_2 a_n \tau_n = 2$.

APPENDIX B

THE U, V, N SYSTEM

Here we examine the mechanisms by which persistent steady state and oscillatory behaviors arise in our model when the excitatory neurons are excited by a brief input stimulus. The NMDA dynamics ensure NMDA stays bounded within $[0, 1]$ (given initial conditions in the same interval). The null surface for NMDA is sigmoidal ($n = \frac{a_n \cdot u^p}{1+a_n \cdot u^p}$), suggesting the possibility of bistability between a low state and a high state. Indeed, the NMDA allows for latch-like behavior of the system [29], so that low NMDA activity coincides with low u and v values (or low average values in the oscillatory case), and high (average for the oscillatory case) NMDA coincides with higher u and v values. Although Eq (2.1) in Chap 2 describes a three-dimensional system when $N = 1$, we may gain some intuition for the observed dynamics by fixing n at a constant level (since it evolves slowly and changes little compared to u and v) and examining the reduced u-v system. Thus, changing n in this planar system is equivalent to changing $\theta_{e,i}$ in tandem in the original system. Foregoing the stimulus, the reduced system is:

$$\begin{aligned} u' &= -u + f(a_{ee} \cdot u - a_{ei} \cdot v + a_{en} \cdot n - \theta_e) \\ \tau_i \cdot v' &= -v + f(a_{ie} \cdot u - a_{ii} \cdot v + a_{in} \cdot n - \theta_i) \end{aligned} \tag{B.1}$$

We may now examine the nullclines in Eq (B.1). As we see in Fig 29, for low n (Fig 29A) the v-nullcline intersects the left branch of the u nullcline, while for high n (Fig 29B), the

v-nullcline intersects the middle branch of the u-nullcline. Thus, the low n case results in a stable fixed point, and the high n case results in a fixed point at somewhat higher u and v that may be stable or unstable, depending on τ_i. For low τ_i (e.g., $\tau_i = 1$), this high fixed point is stable; for higher τ_i, a stable oscillation emerges, as suggested in the example trajectories in Fig 29B. We note that the "high" point in this case still corresponds to low u and v values, though in the oscillatory case this can still lead to large maxima, as we see in Fig 29B, e.g. The nullcline intersection can change based on parameter values, however. Lowering a_{ei} to 8, e.g., results in a middle-branch intersection at much higher u and v values, as suggested by Fig 30A. Therefore, depending on the values of the other parameters, varying n (still as a parameter) can allow such a two-dimensional system to switch between low- and high-activity states that can be either steady-state or oscillatory.

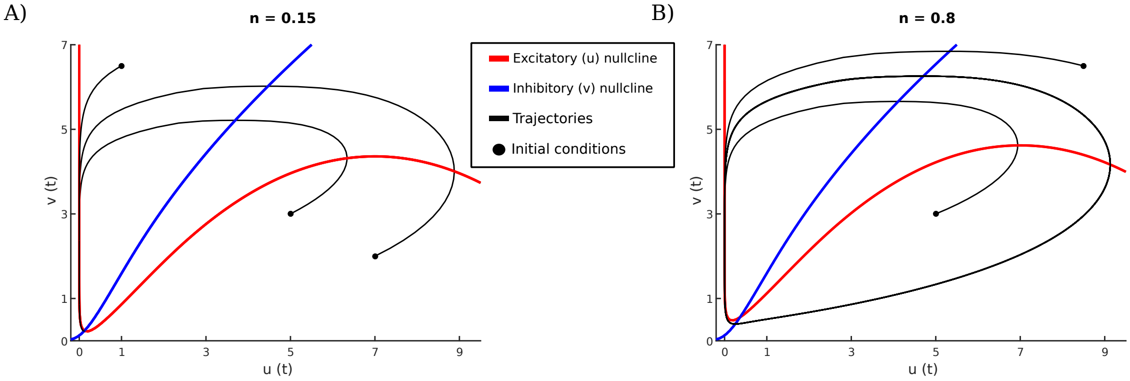

Figure 29: Nullclines and example trajectories for Eq (B.1). Parameter values are as given in Sec *2.2*. (A) For lower NMDA (n) values, the v-nullcline intercepts the left branch of the u-nullcline, resulting in a stable fixed point. (B) For higher NMDA values, the v-nullcline intercepts the middle branch of the u-nullcline; the fixed point is stable for low τ_i values and unstable for larger τ_i values. Here, $\tau_i = 12$, so that the fixed point is unstable and the system has a stable limit cycle, as suggested by the example trajectories.

In the full three-dimensional system, we set up our u and v nullsurfaces so that their low-n and high-n cross-sections intersect as the corresponding 2-dimensional nullclines do, using the parameters given in Sec *2.2*. As the NMDA nullsurface is sigmoidal, it can act as a dynamic latch, so that once n is excited enough (via the $s(t) \to u \to n$ path), the system

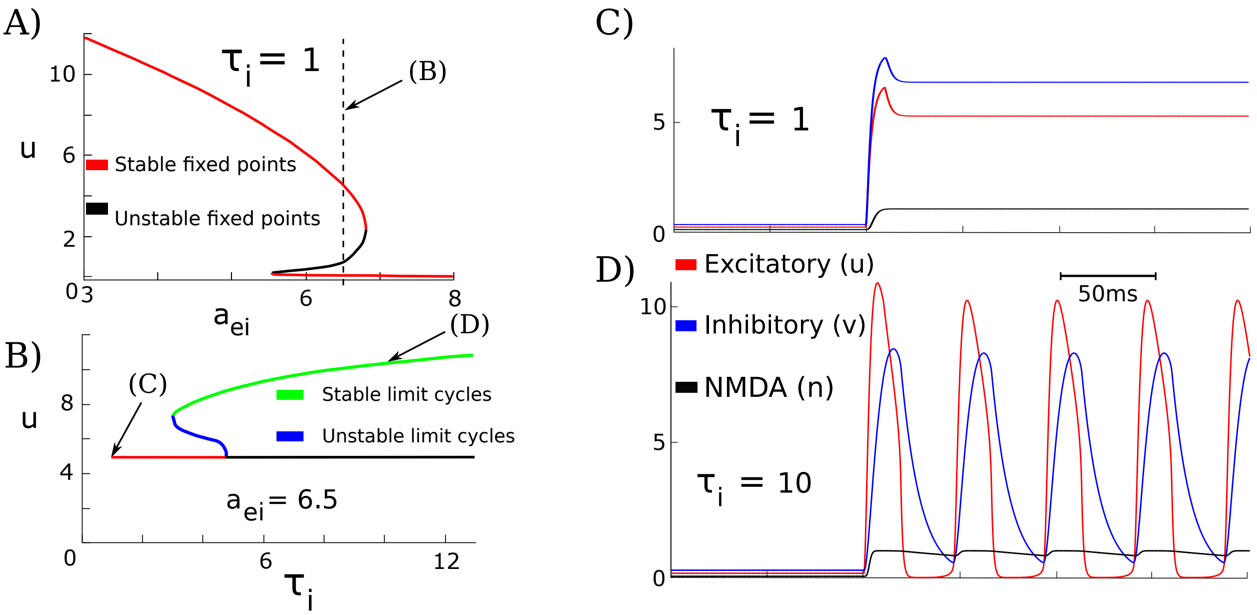

Figure 30: Bifurcations and dynamics for one population for Eq (2.1) in Chap 2. (A) There are open sets of parameters, such as a_{ei}, that allow for bistability between low states and high states. (B) Increasing τ_i leads to a subcritical Hopf bifurcation of the high fixed point. In (C) and (D), a stimulus of width 10ms and amplitude of 5 (au) is applied to the u populations at $t = 100$ms, leading the circuit to evolve to high active states: a high fixed point in the case of $\tau_i = 1$ in (C) and large-amplitude oscillations in the case of $\tau_i = 10$ in (D). Since in both cases the circuit returns to the low steady state if the stimulus amplitude or width are below threshold values, both cases display bistability between low rest states and high active states.

is attracted to the high state. As we just mentioned above, the existence of bistability is, of course, dependent on system parameters; for example, we see that two stable fixed points exist for a range of a_{ei} values in Fig 30A. As τ_i increases, the high fixed point destabilizes and a limit cycle is born via a subcritical Hopf bifurcation (Fig 30B). While this allowed us to easily find parameters with which both a stable low fixed point and a stable high, oscillatory, state coexist, we ended up using $a_{ei} = 10$, not 8 as in Fig 30B. For $a_{ei} = 10$, the oscillations are lost as a fold of limit cycles as in Fig 30B as τ_i decreases, but there is no Hopf; rather, we end up with an isola of limit cycles as in Fig 5A in Chap 2. If τ_i decreases beyond the left fold, only the low steady state stably exists.

We may gain further intuition for how the separation of timescales allows for oscillations. Once the excitatory stimulus kicks the AMPA population sufficiently, the population begins an excursion around phase space. In turn, it excites the inhibitory population, which then chases the AMPA population, curtailing its growth. As the inhibitory timescale is slower than that of the excitatory population ($\tau_i > \tau_e = 1$), the inhibition eventually wins out, quenching the activity of the AMPA population, as we see in Fig 30D. This is why the NMDA population is important: The NMDA is excited by the AMPA, but decays much slower than the inhibition ($\tau_n > \tau_i$), allowing the NMDA to outlast the upstroke of the inhibition. By doing so, the NMDA can then re-excite the AMPA population once the inhibition is sufficiently low, producing the observed wavetrain in Fig 30D.

Thus, if τ_n is too small, the NMDA population will increase rapidly with AMPA, but will then decay too quickly to maintain the activity of the AMPA population after the inhibitory population has quenched it. Clearly the same problem will occur if τ_i is too large. Therefore, the ratio $\frac{\tau_i}{\tau_n}$ must be small enough. However, if τ_i is too small, two possibilities emerge: (1) the inhibition (v) decays too quickly to quench the excitatory population, in which case activation will lead to a stable up state; (2) the inhibition activates very rapidly, quenching the excitatory activity before it increases enough from its baseline levels, in which case no activation occurs: only the low steady state is stable. For our parameter choices, decreasing τ_i leads to (2), as we mentioned above. In either case, the initial phase space excursion that occurs in the oscillatory case cannot take place at all. Instead, after receiving a stimulus the excitation will either settle at the high fixed point in the case of (1) with a strong enough stimulus (as we can see in Fig 30C), or else it will simply decay back to the low steady state. Hence, we now have some intuition to understand the general shapes of the bifurcation curves in Fig 5B, 5C, and 5D in Chap 2.

APPENDIX C

WEAK COUPLING ANALYSIS

Using the parameters shown in Sec *2.2* in Chap 2 for a single local $u-v-n$ circuit, we can perform a weak coupling analysis (WCA) around the limit cycle solution that appears in the up state. That is, since the coefficients that we use tend to be small compared to the self coupling, the WCA can provide some insights into the possible patterns when M groups are oscillating. The results should hold for any size network, since the "down" circuits contribute very little to the interactions in the groups that are oscillating. To perform the WCA, we first see that

$$f(a_{ee}u + a_{en}n - a_{ei}v - \theta_e - \epsilon\hat{v}) \approx f(I_e) - \epsilon\, f'(I_e)\hat{v},$$

where $I_e = a_{ee}u + a_{en}n - a_{ei}v - \theta_e$, and with similar terms for other types of coupling. Recall that in WCA, we reduce equations of the form

$$X_i' = F(X_i) + \epsilon G_i(X_1, \ldots, X_N), \qquad i = 1, \ldots, N \tag{C.1}$$

to a set of equations of the form:

$$\theta_i' = 1 + H_i(\theta_1 - \theta_i, \ldots, \theta_N - \theta_i).$$

To achieve this reduction, we assume there is a $T-$periodic limit cycle solution to the uncoupled system (i.e., Eq (C.1) with ϵ set to 0), $U' = F(U)$, and let $Z(t)$ be the unique solution (adjoint solution) to $Z' = -(D_X F(U))^T$, with $U'(t) \cdot Z(t) = 1$. The phase interaction functions H_i are defined as

$$H_i(\phi_1, \ldots, \phi_N) = \frac{1}{T} \int_0^T Z(t) \cdot G_i(U(t + \phi_1), \ldots, U(t + \phi_N))\, dt.$$

Since all coupling is summed up, we only need to compute the interaction with one other circuit. Thus, we need to compute the adjoint solution and the basic limit cycle. Letting $(u(t), v(t), n(t))$ be the basic limit cycle for the isolated population and $(u^*(t), v^*(t), n^*(t))$ be the corresponding adjoint solution, we compute the following four interaction functions:

$$\begin{aligned}
H_{ee}(\phi) &= \frac{1}{T} \int_0^T u^*(t) f'(I_e(t))[a_{ee} u(t+\phi) + a_{en} n(t+\phi)]\ dt \\
H_{ei}(\phi) &= -\frac{1}{T} \int_0^T u^*(t) f'(I_e(t)) a_{ei} v(t+\phi)\ dt \\
H_{ie}(\phi) &= \frac{1}{T} \int_0^T v^*(t) f'(I_i(t))[a_{ie} u(t+\phi) + a_{in} n(t+\phi)]\ dt \\
H_{ii}(\phi) &= -\frac{1}{T} \int_0^T v^*(t) f'(I_i(t)) a_{ii} v(t+\phi)\ dt.
\end{aligned}$$

Once we have the interaction functions, we can study the dynamics of the oscillators when weakly coupled. We form the composite function:

$$H(\phi) = c_{ee} H_{ee}(\phi) + c_{ei} H_{ei}(\phi) + c_{ie} H_{ie}(\phi) + c_{ii} H_{ii}(\phi).$$

The coupled phase equations satisfy

$$\theta_i' = \sum_{j \neq i} H(\theta_j - \theta_i),$$

where i varies from 1 to N active groups. To determine the locked patterns, we reduce the dimension to $N-1$ by setting $\theta_1 = 0$ and subtracting θ_1' from the remaining $N-1$ equations:

$$\psi_i' = \sum_{j \neq i} H(\psi_j - \psi_i) - \sum_{j > 1} H(\psi_j).$$

Here, ψ_i represents the phase relative to θ_1. Stable fixed points of this $(N-1)$ – dimensional system correspond to the attracting dynamics of the weakly coupled system. Table 1 summarizes the attractors for up to 4 active groups when each of the individual coupling terms are set to 1 and the rest are set to zero. We note that the behavior of EE coupling serves mainly to synchronize and that the EI and IE coupling have similar behavior. Small amounts of EE coupling in addition to EI coupling (what we use in the full model) behave like the EI coupling alone as long as the EE coupling is not too big. For example, with EI coupling and three active groups, WCA predicts that there will be synchrony (S), a splay state (L), and a clustered state (C) with two oscillators synchronized and the third out of phase (see Fig 9 in Chap 2, where L is indicated as OP and C as MP). Since the coupling is all-to-all and symmetric, all possible permutations of the attractors occur.

This analysis explains many of the interactions we see with active populations. We remark that the analysis is only valid when the coupling is weak enough, but it still manages to include many of the attractors that are seen when there are two or more active populations. We finally note that for 5 or more active populations we only see various clustered states and synchrony with weak coupling; we do not see any splay states. Thus, at least when the coupling is weak, there is limited capacity in the network.

Active	**EE**	**EI**	**IE**	**II**
2	S, A	S, A	S, A	N
3	S	S, L, C	S, L, C	C
4	S	S, CS, C, L3	S, CS, C, L	C

Table 1: Weak coupling summary of the dynamics for up to 4 active groups. The states are synchronous (S), anti-phase (A), nonsynchronous (N), clustered (C), symmetric cluster (CS), splay (L), and semi-splay (L3). By synchronous, we mean that all the oscillators fire together in-phase; anti-phase means the two oscillators fire a half cycle apart; clustered means that two groups of oscillators form that are synchronous within the group and out-of-phase between groups; symmetric clusters mean that there are equal numbers in each group; nonsynchronous means neither synchronous nor anti-phase; splay is the state: $0,\ 1/N,\ 2/N,\ \ldots,\ (N-1)/N$; semi-splay is a state where 2 oscillators are synchronized and the other two are out of phase but not synchronized themselves. Thus, A, L, and N correspond to the out-of-phase (OP) oscillations described in Chap 2 while C, CS, and L3 correspond to the mixed-phase (MP) oscillations.

APPENDIX D

CHANGE IN OP AND S DYNAMICS WITH VARYING COUPLING STRENGTHS AND SYNAPTIC TIMESCALES

OP solutions (N=2)

The OP state arises through competition that is facilitated by the mutual inhibition, c_{ei}. For two populations P1 and P2, as we have here, each excitatory component, u_1 and u_2, peaks twice during each oscillation; once at a large value (the large peak) and once at a small value (the small peak) (see Fig 31). Henceforth we will refer to the population that peaks at a large value as the primary population, and the one that peaks at a small value as the secondary population. For example, suppose P1 is the primary population and P2 is the secondary population as in the first half of the oscillation in, e.g., Fig 31B. The downstrokes for both u_1 from its large peak and u_2 from its small peak are caused by the upstroke of v_1. Thus, u_2 (and therefore P2, since our readout is the excitatory component) is kept inactive by the competitive activity from v_1.

What happens as the coupling strengths c_e and c_{ei} change? Several characteristics may become altered, including the maxima and minima that the various components achieve, the period of the oscillation, and the relative phase timings of the components. However, since the populations are coupled together only through the excitatory components, perhaps the most important change is the increase in the amplitude of the small peak. Once the small peak increases too much, the secondary population may no longer be suppressed, becoming

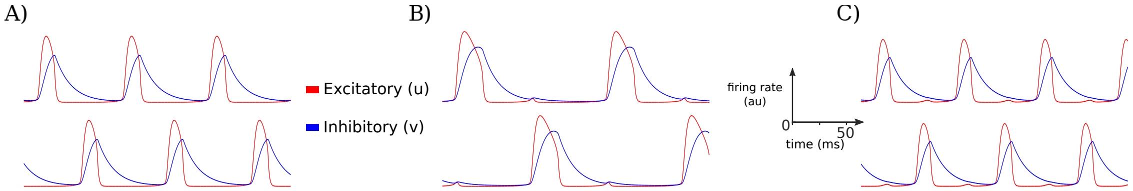

Figure 31: Example traces of OP states for $N = 2$ and varying c_e and c_{ei}. As c_e and c_{ei} increase, the small peak of the excitatory population becomes more pronounced. (A) $c_e = 0,\ c_{ei} = 0.01$. Here no small peak occurs. (B) $c_e = 0,\ c_{ei} = 0.54$. The primary excitatory population (top plot in blue at the beginning of the timecourse) begins to rise again after its first large excursion from its baseline value, but the secondary inhibitory population (bottom red plot at the start of the trace) comes on and is seen to stop the first population in its tracks, so that only a small, secondary excursion occurs. (C) $c_e = 0.055,\ c_{ei} = 0.0855$. The presence of the excitatory coupling from nonzero c_e pushes the small peak in such a way as to be more coincident with the large peak of the alternate population; however, the inhibition due to c_{ei} still suppresses it, again only allowing for a small excursion from baseline.

c_e	Period	Large u peak	Small u peak	v_{max}	v_{min}	n_{min}
0.00	92	10.0	0.05	7.5	0.15	0.59
0.05	89	9.3	0.64	6.8	0.17	0.60

Table 2: OP state changes with c_e. An example of the changes to the OP state for $N = 2$ that occur as c_e increases while c_{ei} remains fixed at 0.3.

active and perhaps synchronizing with the primary population. This is manifested as a bifurcation (see Fig 8C in Chap 2). Indeed, in all of the cases we examined, if either c_e or c_{ei} increases beyond the bifurcation values, the network transitions to synchrony in numerical simulations. We first look at varying c_e and then the more complicated and somewhat paradoxical picture of varying c_{ei}.

Fixed c_{ei}, varying c_e

Fixing c_{ei} and increasing c_e results in (1) slightly decreased period; (2) decreased large peak; (3) increased small peak (Table 2). Suppose again that P1 is the primary population and P2 is the secondary population, so that $u_1 > u_2$ in the interval of interest. We first observe that as c_e increases, u_2 receives significantly more excitation, while u_1 receives *less* excitation. This is a direct result of the normalization we have used. In particular, from Eq (2.2) in Chap 2 we see that for two populations,

$$\widetilde{u}_1 = \frac{u_1 + c_e\, u_2}{1 + c_e}.$$

Differentiating $\widetilde{u}_1$ with respect to c_e shows that it monotonically decreases as c_e increases if $u_1 > u_2$, as we have assumed. As a result, u_1's maximum decreases, so that the width of the pulse of u_1 decreases, whereas the length of its quiescent phase experiences almost no change. Thus, the small decrease in period is mostly due to the decrease in the amplitude

c_{ei}	Period	Large u peak	Small u peak	v_{max}	v_{min}	n_{min}
0.00	49	9.2	N/A	6.3	0.39	0.77
0.54	114	10.4	0.64	8.2	0.14	0.53

Table 3: OP state changes with c_{ei}. An example of the main changes to the OP state for $N = 2$ that occur as c_{ei} increases while c_e remains fixed 0. Note that the value for the small u peak indicates "not applicable" since the peak in the secondary excitatory component only occurs for large enough c_{ei} and c_e.

of the large peak. For nonzero c_{ei}, the increase in the maximum of u_2 in turn further excites v_2, so that u_1 experiences greater inhibition. Thus, in addition to the effect just now described, increasing c_e results in a lower maximum for u_1 since v_2 inhibits u_1 more when c_{ei} is nonzero. In either case, once the maximum of the secondary u becomes too large relative to that of the primary u, the splay state is lost to synchrony.

Fixed c_e, varying c_{ei}

As c_{ei} increases, we observe the following changes: (1) the peak of the large amplitude changes, increasing monotonically until just before bifurcation; (2) the small peak changes, generally increasing monotonically; (3) the period of the oscillation changes, increasing monotonically until just before bifurcation; (4) the inhibition generally peaks at larger values and decays to smaller values; (5) the NMDA generally decays to lower levels (Table 3). There are some subtle differences when c_e is low or high, and we discuss each in turn.

When c_e is low or zero, the effects of increasing c_{ei} are straightforward. We again suppose P1 is the primary population and P2 is the secondary population. For simplicity we will focus on $c_e = 0$. The excitatory component of the primary population, u_1, cannot begin its larger upstroke until $\widetilde{v}_1$ is sufficiently low. Since v_1 is very low before u_1 begins its large upstroke, v_2 provides most of the inhibition that keeps u_1 low before its large upstroke.

However, we again look at the coupling term:

$$\widetilde{v}_1 = \frac{v_1 + c_{ei}\, v_2}{1 + c_{ei}}, \tag{D.1}$$

Differentiating with respect to c_{ei}, we see that $\widetilde{v}_1$ increases (for fixed v_1 and v_2) with increasing c_{ei} when $v_2 > v_1$. Thus, as c_{ei} increases, both v_1 and v_2 must decay to lower values before releasing u_1.

For identical reasons, u_2 receives less inhibition at the beginning of the large upstroke of u_1 as c_{ei} increases. That is, $\widetilde{v}_2$ decreases (for fixed v_1 and v_2) as c_{ei} increases when $v_2 > v_1$. This leads to lower inhibition for u_2 as it begins its small upstroke, allowing it to peak at a higher value (we note that n_2 has also decayed to a lower value, leading to less excitation for u_2; however, since a_{en} is much smaller than a_{ei}, this effect is much smaller).

In summary, larger c_{ei} requires lower inhibitory component values in order for the excitatory components to activate, leading to larger amplitudes of the small peak, a result that eventually destabilizes the antiphase solution. In this scenario, it seems that the increases in the period and the amplitude of the large peak are secondary effects.

We note that there is some competition between the effects of c_{ei}. On the one hand, larger c_{ei} may allow a smaller amount of inhibition of one population to quench the activity of the other; on the other hand, for reasons outlined in the above paragraph, larger c_{ei} can increase the amplitudes of the smaller peaks. Thus, depending on, for example, the precise phase timings of the various components of the two populations, larger c_{ei} could either quench or enhance the activity of the secondary population.

When c_e is larger, we observe exactly these competitive effects as c_{ei} increases. In particular, while the large peak increases monotonically until just before bifurcation, the small peak first increases, then decreases, then increases again. This pattern leads directly to the nonmonotonic behavior of curve (iv) in Fig 8C in Chap 2. That is, for some c_e values, the OP state is lost, regained, and lost again as folds of limit cycles with increasing c_{ei}. This appears to be due directly to the nonmonotonic behavior of the small peak for the reasons we described above. In particular, the c_{ei} values at which the amplitude of the small peak begins to decrease converge to the minimum of curve (iv) in Fig 8C in Chap 2. We note that this competition that leads to the nonmonotonic curve depends on the particular parameters. We have explored other parameter sets, e.g., with which this curve simply decreases monotonically as c_{ei} increases.

Synchronous solutions, varying c_{ei}

In Fig 6D in Chap 2, we see a general tendency for increasing c_{ei} to increase the interval of τ_i for which we obtain stable M-S solutions, where $M \in \{1, ..., N\}$. Here we provide some heuristic reasoning for why this may be the case.

We first note that when $M = N$, as we described at the beginning of Sec *2.3.2.2*, the network oscillates as if there were only one population, although the c_{ei} value somewhat changes the range of parameters that allows for this oscillation to exist stably. This is expected, as any perturbation of one of the populations will mean it will feel inhibition from the remaining $M - 1$ populations, and will provide inhibition to them as well. In particular though, we note that as c_{ei} changes for this case, the behavior of the solutions do not change at all; neither the period nor any of the amplitudes change with varying c_{ei}. This is not the case for $M \neq N$.

For $M < N$, we begin to observe changes in the period, the amplitudes of the various components, and the waveform of the solutions. Generally speaking, increasing c_{ei} both increases the largest and decreases the smallest τ_i values that admit stable oscillations, as we see in Fig 6D in Chap 2. While this is not strictly the case for $M = 4$ as the smaller τ_i limit increases very slightly, it indicates the trend. We note that while the cases of $M = 2, ..., N-1$ may be argued similarly to the case of $M = 1$, there are certain subtleties that somewhat complicate the picture. Since, as we see in Fig 6D in Chap 2, the range of τ_i increases for the 1-S oscillation in the same way as for $M = 2$ or 3 and nearly in the same way as for $M = 4$, we will focus on the simplest case of $M = 1$.

As τ_i increases, the period lengthens and the maxima of the excitatory and inhibitory solutions increase. As we explain in Appendix B, if τ_i is too large relative to τ_n, the NMDA will not outlast the inhibition and the oscillations will cease. If τ_i is too small (and we are not in a parameter regime that allows for a stable high steady state; see Appendix B) the inhibition activates very rapidly, quenching the excitatory activity before it increases enough from its baseline levels, in which case no activation occurs: only the low steady state is stable. The periods and amplitudes of the populations also change with c_{ei} as shown in Fig 32. The amplitudes increase monotonically with c_{ei} (although perhaps by very little, as we see in Fig 32 for $\tau_i = 39.8$), while the periods may increase or decrease.

The stronger trend with increasing c_{ei} that we observe in Fig 6D in Chap 2 (where c_{ei} increases from 0.03 to 0.07) is the increase in the upper τ_i limit. To see why this might occur, consider $M = 1$, suppose that P1 is the active population, and note that the populations that are inactive are not only at similarly low levels, but in fact are themselves oscillating synchronously at low values. Therefore, the inhibition that u_1 receives is given by Eq (2.2) in Chap 2, which we specify for the case of $\widetilde{v}_1$ here:

$$\widetilde{v}_1 = \left(v_1 + c_{ei} \sum_{k=2}^{N} v_k \right) \Big(1 + c_{ei} \; (N-1) \Big)^{-1},$$

which simplifies to

$$\widetilde{v}_1 = \frac{v_1 + c_{ei} v_j}{1 + c_{ei}},$$

where j can be anything in $\{2, ..., N\}$ since, as we mentioned, $\{$P2, ..., P$N\}$ are synchronous. This, of course, is identical to Eq (D.1), and so the same analysis can be applied as was done in that case. In particular, we note that since P1 is the only active population, u_1 is always (or nearly so) larger than u_j, for $j \in \{2, ..., N\}$. Thus, following the above analysis (where P1 is essentially always the primary population), $\widetilde{v}_1$ *decreases* with increasing c_{ei}, so that u_1 receives *less* inhibition. This exactly explains the increase in the maxima of u_1 with increasing c_{ei} (and, since v_j is excited by u_j, the increase in the maximum of v_1 as well). More relevantly, u_1 needs less excitation from n_1 to maintain the large oscillations for larger c_{ei} values, allowing the oscillations to remain stable for larger τ_i values, as we see in Fig 6D in Chap 2.

The same explanation holds for low τ_i values. As c_{ei} increases, u_1 receives less inhibition, so that u_1 can maintain its activity for smaller τ_i. For example, suppose $c_{ei} = 0.03$ and

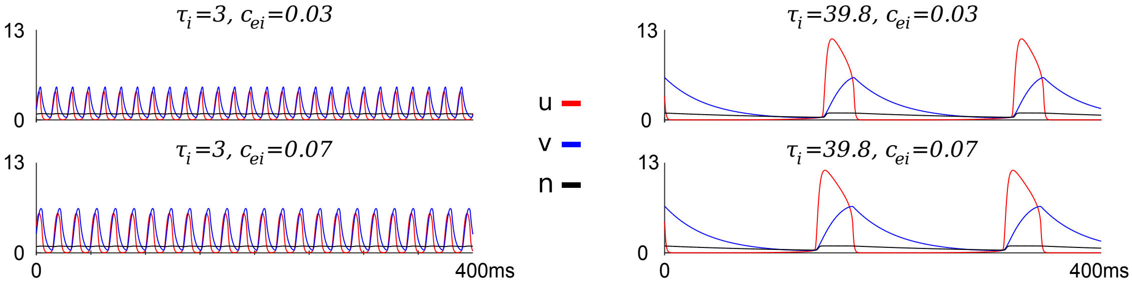

Figure 32: Example traces of one population oscillating with $N = 5$. In each row, c_{ei} is fixed (and increases as we go from the top row to the bottom row), while τ_i increases from left to right. As τ_i increases with c_{ei} fixed, the period and amplitude increase monotonically and substantially, while as c_{ei} increases with τ_i fixed, the amplitudes increase monotonically, but by much smaller amounts for the given range of c_{ei} values, while the periods show more complicated behaviors. For $\tau_i = 3$ (left 2 plots), the period increases monotonically with c_{ei}, while for $\tau_i = 39.8$ (right 2 plots), the period in fact decreases and then increases as c_{ei} increases from 0.03 to 0.07. For both τ_i values, decreasing c_{ei} just a little bit below 0.03 results in the loss of the oscillations to the low steady state.

$\tau_i = 2.9$, the lower point for $M = 1$ in Fig 6D in Chap 2. As c_{ei} increases, u_1 receives less inhibition, so that v_1 needs to activate even faster to prevent u_1 from its large excursion from baseline. Thus, τ_i must be lowered still further for $c_{ei} = 0.07$ before P1 will be unable to remain active (down to $\tau_i = 2$ in this example).

APPENDIX E

LINEAR STABILITY ANALYSIS OF THE WILSON-COWAN NETWORK EXTENDED IN ONE SPATIAL DIMENSION

We will show explicitly how we determined the analytical stability curves shown in Fig 24 in Chap 3. The calculations shown in this appendix were input into Mathematica to determine the numerical values of the relevant variables, allowing us to make the plots in Fig 24.

E.1 DETERMINATION OF THE EIGENVALUES

E.1.1 *The nonhomogeneous steady state*

Letting m^* be our critical wavenumber and N the number of neuronal populations, we set $\omega^* = \frac{2\pi m^*}{N}$ and $\mu = \frac{2\pi k}{N}$. We'll designate the eigenvalue of the linearization (Eq (3.10)) as $\lambda_0 = i\zeta_0$. Since we'll be analyzing the behavior of our Wilson-Cowan network in one spatial dimension when we turn on the stimulus, for this analysis we let the amplitude of the stimulus, q, be a small parameter. Thus, we wish to perform a linear stability analysis on the system given by

$$\begin{aligned} \tau_e \ u_t(x,t) &= -u + f_e(J_{ee} * u - J_{ei} * v + q\cos\mu x) \\ \tau_i \ v_t(x,t) &= -v + f_i(J_{ie} * u - J_{ii} * v + qr\cos\mu x), \end{aligned} \tag{E.1}$$

where our domain is [0, 1] and $f_{e,i} = f(x - \theta_{e,i}) - f(-\theta_{e,i})$, so that our homogeneous $(q = 0)$ steady state is $(0, 0)$.

We use $\alpha_1 = f'(\theta_e),\ \alpha_2 = f''(\theta_e)/2,\ \alpha_3 = f'''(\theta_e)/6$ and $\beta_1 = f'(\theta_i),\ \beta_2 = f''(\theta_i)/2,\ \beta_3 = f'''(\theta_i)/6$. Expanding $f_{e,i}$ therefore gives

$$\begin{aligned} f_e(x) &= \alpha_1 x + \alpha_2 x^2 + \alpha_3 x^3 + ... \\ f_i(x) &= \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + ... \end{aligned} \tag{E.2}$$

since $f_{e,i}(0) = 0$.

Expanding the nonhomogeneous steady state provides

$$\begin{aligned} u_{ss} &= \overline{u} + qu_0 + q^2 u_1 = qu_0 + q^2 u_1 \\ v_{ss} &= \overline{v} + qv_0 + q^2 v_1 = qv_0 + q^2 v_1, \end{aligned} \tag{E.3}$$

since $(\overline{u},\ \overline{v}) = (0, 0)$ is the homogeneous steady state. Since $u_{ss_t} = v_{ss_t} = 0$, we expand (E.1) to obtain

$$\begin{aligned} qu_0 + q^2 u_1 =& q\alpha_1 \left[J_{ee} * u_0 - J_{ei} * v_0 + \cos\mu x\right] + \\ & q^2 \left[\alpha_1 \left(J_{ee} * u_1 - J_{ei} * v_1\right) + \alpha_2 \left(J_{ee} * u_0 - J_{ei} * v_0 + \cos\mu x\right)^2\right], \end{aligned} \tag{E.4}$$

and similarly for v. Therefore, our first terms are

$$\begin{aligned} u_0 &= \alpha_1 \left[J_{ee} * u_0 - J_{ei} * v_0 + \cos\mu x\right] \\ v_0 &= \beta_1 \left[J_{ie} * u_0 - J_{ii} * v_0 + r\cos\mu x\right]. \end{aligned} \tag{E.5}$$

As we are on a periodic domain and apply a cosine stimulus, we look for solutions

$$u_0 = \gamma_0 \cos\mu x, \quad v_0 = \eta_0 \cos\mu x. \tag{E.6}$$

While Eq (E.5) is not an algebraic system, we are in fact able to transform it into one by observing how convolutions with sinusoidal functions can be simplified. Consider any kernel function $K(x)$ and let $\xi = \frac{2\pi l}{L}$. Then the convolution

$$\int_{-\infty}^{\infty} K(y)\left(e^{i\xi(x-y)}\right) dy = e^{i\xi x} \int_{-\infty}^{\infty} K(y) e^{-i\xi y} dy = \widehat{K}(l) \cdot e^{i\xi x},$$

where $\widehat{K}(\xi)$ is the Fourier transform of K. If $K(x)$ is even, then the same is true when we convolve K against $e^{-i\xi x}$:

$$\int_{-\infty}^{\infty} K(y)\left(e^{-i\xi(x-y)}\right) dy = \int_{-e^{-i\xi x}\infty}^{\infty} K(y) e^{i\xi y} dy$$

$$= \int_{-e^{-i\xi x}\infty}^{\infty} K(y) e^{-i\xi y} dy = \widehat{K}(l) \cdot e^{-i\xi x},$$

where we've used the change of variables $y \leftarrow -y$ in the second step. Finally, putting these two identities together further provides the same type of identity when an even $K(x)$ is convolved against a cosine function:

$$\int_{-\infty}^{\infty} K(y) \cos \xi x = \frac{1}{2} \int_{-\infty}^{\infty} K(y) \left(e^{i\xi(y-x)} + e^{-i\xi(y-x)} \right)$$

$$= \int_{-\infty}^{\infty} K(y) e^{-i\xi y} dy = \widehat{K}(l) \cdot \cos \xi x \quad dy.$$

Clearly, the same holds when convolving against a sine function. Summarizing, if we have an even function $K(x)$ and a periodic function $P(x;\, \xi) = Ae^{i\xi x} + Be^{-i\xi x}$ with $A,\, B \in \mathbb{C}$, then

$$(K * P)(x;\, \xi) = \int_{-\infty}^{\infty} K(y) P(x - y;\, \xi) dy = \widehat{K}(l) \cdot P(x;\, \xi). \tag{E.7}$$

We'll use this result extensively. For example, $J_{ee} * \gamma_0 \cos \mu x = \widehat{J}_{ee}(k) \cdot \gamma_0 \cos \mu x$, where $\widehat{J}_{ee}(k) = \int_{-\infty}^{\infty} J_{ee}(y) \cdot \exp\left(\frac{-2\pi i k}{N} y\right)\, dy$. We also note that our Gaussian kernel remains a Gaussian under a Fourier transformation:

$$\frac{a_{ee}}{\sigma_e \sqrt{\pi}} \int_{-\infty}^{\infty} \exp\left(\frac{-y^2}{\sigma_e^2}\right) \cdot \exp\left(-2\pi i k y\right)\, dy = a_{ee}\ \exp\left(-\pi k \sigma_e\right)^2.$$

Thus, plugging (E.6) into (E.5) and using (E.7) provides our first perturbed terms

$$\gamma_0 = \alpha_1 \left[\widehat{J}_{ee}(k)\gamma_0 - \widehat{J}_{ei}(k)\eta_0 + 1 \right]$$
$$\eta_0 = \beta_1 \left[\widehat{J}_{ie}(k)\gamma_0 - \widehat{J}_{ii}(k)\eta_0 + r \right].$$

This gives

$$\begin{bmatrix} 1 - \alpha_1 \widehat{J}_{ee}(k) & \alpha_1 \widehat{J}_{ei}(k) \\ -\beta_1 \widehat{J}_{ie}(k) & 1 + \beta_1 \widehat{J}_{ii}(k) \end{bmatrix} \begin{bmatrix} \gamma_0 \\ \eta_0 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \beta_1 r \end{bmatrix},$$

which we invert to obtain γ_0 and η_0.

We now find the $\mathcal{O}(q^2)$ terms. From the $\mathcal{O}(q^2)$ terms of (E.4) we obtain

$$\begin{aligned} u_1 - \alpha_1 J_{ee} * u_1 + \alpha_1 J_{ei} * v_1 &= \alpha_2 \left(J_{ee} * u_0 - J_{ei} * v_0 + \cos\mu x\right)^2 = \frac{\alpha_2 u_0^2}{\alpha_1^2} \\ \psi_1 - \beta_1 J_{ie} * u_1 + \beta_1 J_{ii} * v_1 &= \beta_2 \left(J_{ie} * u_0 - J_{ii} * v_0 + r\cos\mu x\right)^2 = \frac{\beta_2 v_0^2}{\beta_1^2}. \end{aligned}$$

Applying (E.6) (and using $\cos^2 x = \frac{1}{2}(1+\cos 2x)$) to the RHS of the above system implies that

$$u_1 = \gamma_1 + \gamma_2 \cos 2\mu x, \quad v_1 = \eta_1 + \eta_2 \cos 2\mu x.$$

This provides

$$\begin{aligned} \gamma_1 - \alpha_1 \widehat{J}_{ee}(0)\gamma_1 + \alpha_1 \widehat{J}_{ei}(0)\eta_1 &= \frac{\alpha_2}{2\alpha_1^2}\gamma_0^2 \\ \eta_1 - \beta_1 \widehat{J}_{ie}(0)\gamma_1 + \beta_1 \widehat{J}_{ii}(0)\eta_1 &= \frac{\beta_2}{2\beta_1^2}\eta_0^2 \\ \gamma_2 - \alpha_1 \widehat{J}_{ee}(2k)\gamma_2 + \alpha_1 \widehat{J}_{ei}(2k)\eta_2 &= \frac{\alpha_2}{2\alpha_1^2}\gamma_0^2 \\ \eta_2 - \beta_1 \widehat{J}_{ie}(2k)\gamma_2 + \beta_1 \widehat{J}_{ii}(2k)\eta_2 &= \frac{\beta_2}{2\beta_1^2}\eta_0^2. \end{aligned}$$

Our $\cos 0x$ terms are given by

$$\begin{bmatrix} 1 - \alpha_1 \widehat{J}_{ee}(0) & \alpha_1 \widehat{J}_{ei}(0) \\ -\beta_1 \widehat{J}_{ie}(0) & 1 + \beta_1 \widehat{J}_{ii}(0) \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \eta_1 \end{bmatrix} = \begin{bmatrix} \frac{\alpha_2}{2\alpha_1^2}\gamma_0^2 \\ \frac{\beta_2}{2\beta_1^2}\eta_0^2 \end{bmatrix},$$

and our $\cos 2\mu x$ terms are given by

$$\begin{bmatrix} 1 - \alpha_1 \widehat{J}_{ee}(2k) & \alpha_1 \widehat{J}_{ei}(2k) \\ -\beta_1 \widehat{J}_{ie}(2k) & 1 + \beta_1 \widehat{J}_{ii}(2k) \end{bmatrix} \begin{bmatrix} \gamma_2 \\ \eta_2 \end{bmatrix} = \begin{bmatrix} \frac{\alpha_2}{2\alpha_1^2}\gamma_0^2 \\ \frac{\beta_2}{2\beta_1^2}\eta_0^2 \end{bmatrix}.$$

Solving these two linear systems provides our coefficients.

E.1.2 *Linearization about the nonhomogenous steady state*

We now prepare to linearize about our nonhomogenous steady state. If we let

$$L_e(u,v) := J_{ee} * u - J_{ei} * v$$

$$L_i(u,v) := J_{ie} * u - J_{ii} * v,$$

we have, from (E.1) and (E.2),

$$\begin{aligned} \tau_e\ u_t =& -u + \alpha_1\left(L_e(u,v) + q\cos\mu x\right) + \alpha_2\left(L_e(u,v) + q\cos\mu x\right)^2 \\ & + \alpha_3\left(L_e(u,v) + q\cos\mu x\right)^3 \\ \tau_i\ v_t =& -v + \beta_1\left(L_e(u,v) + qr\cos\mu x\right) + \beta_2\left(L_e(u,v) + qr\cos\mu x\right)^2 \\ & + \beta_3\left(L_e(u,v) + qr\cos\mu x\right)^3 . \end{aligned} \tag{E.8}$$

We perturb u and v so that $u = u_{ss} + w$, $v = v_{ss} + y$, where w and y are small, and rearrange (E.8):

$$\begin{aligned} \tau_e\ w_t = -(u_{ss} + w) &+ \alpha_1\left[L_e(u_{ss} + w, v_{ss} + y) + q\cos\mu x\right] \\ &+ \alpha_2\left[L_e(u_{ss} + w, v_{ss} + y) + q\cos\mu x\right]^2 \\ &+ \alpha_3\left[L_e(u_{ss} + w, v_{ss} + y) + q\cos\mu x\right]^3 \end{aligned}$$

$$\begin{aligned} \tau_i\ y_t = -(v_{ss} + y) &+ \beta_1\left[L_i(u_{ss} + w, v_{ss} + y) + qr\cos\mu x\right] \\ &+ \beta_2\left[L_i(u_{ss} + w, v_{ss} + y) + qr\cos\mu x\right]^2 \\ &+ \beta_3\left[L_i(u_{ss} + w, v_{ss} + y) + qr\cos\mu x\right]^3 . \end{aligned}$$

Each term in brackets is $(L_e(w,y) + [L_e(u_{ss}, v_{ss}) + q\cos\mu x])$, which we can expand using the binomial theorem to get

$$\begin{aligned}
w_t = & \cancel{-u_{ss}} - w + \cancel{\alpha_1 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right]} \\
& +\alpha_1 L_e(w,y) + \cancel{\alpha_2 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right]^2} \\
& \qquad \xcancel{+\alpha_2 \left[L_e(w,y)\right]^2} \\
& \qquad + 2\alpha_2 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right] L_e(w,y) \\
& \qquad + \cancel{\alpha_3 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right]^3} \\
& \qquad + 3\alpha_3 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right]^2 L_e(w,y) \\
& \qquad \xcancel{+3\alpha_3 \left[L_e(u_{ss}, v_{ss}) + q\cos\mu x\right] L_e(w,y)^2} \\
& \qquad \xcancel{+\alpha_3 \left[L_e(w,y)\right]^3}
\end{aligned}$$

where the $\cancel{}$ is due to u_{ss}'s construction and $\xcancel{}$ is due to our only looking up to leading order in (w,y) since we are linearizing about the steady state.

The treatment of the y_t equation is idential. For the w equation, we are left with

$$\begin{aligned}
\tau_e\ w_t = & -w + \alpha_1 L_e(w,y) \\
& + 2q\alpha_2 \left[L_e(\gamma_0 \cos\mu x, \eta_0 \cos\mu x) + \cos\mu x)\right] L_e(w,y) \\
& + 2q^2\alpha_2 \left[L_e(\gamma_1 + \gamma_2 \cos 2\mu x, \eta_1 + \eta_2 \cos 2\mu x)\right] L_e(w,y) \\
& + 3q^2\alpha_3 \left[L_e(\gamma_0 \cos\mu x, \eta_0 \cos\mu x) + \cos\mu x)\right]^2 L_e(w,y),
\end{aligned}$$

or

$$\begin{aligned}
\tau_e\ w_t = & -w \quad + \quad \alpha_1 L_e(w,y)) \\
& + \quad q \quad \Big[\quad 2\alpha_2 \cos\mu x \left(\gamma_0\, \widehat{J}_{ee}(k) - \eta_0\, \widehat{J}_{ei}(k) + 1\right)\Big] L_e(w,y) \\
& + \quad q^2 \quad \Big[\quad 2\alpha_2 \left\{\gamma_1\, \widehat{J}_{ee}(0) - \eta_1\, \widehat{J}_{ei}(0) + \cos 2\mu x \left(\gamma_2\, \widehat{J}_{ee}(2k) - \eta_2\, \widehat{J}_{ei}(2k)\right)\right\} \\
& \qquad + \frac{3}{2}\alpha_3 \left(1 + \cos 2\mu x\right) \left(\gamma_0\, \widehat{J}_{ee}(k) - \eta_0\, \widehat{J}_{ei}(k) + 1\right)^2 \Big] L_e(w,y).
\end{aligned}$$

Similarly, our y_t equation is

$$\begin{aligned}\tau_i\ y_t = & -y \quad + \ \beta_1 L_i(w,y)) \\ & + \quad q \quad \Big[\quad 2\beta_2 \cos\mu x \left(\gamma_0\,\widehat{J}_{ie}(k) - \eta_0\,\widehat{J}_{ii}(k) + r\right)\Big]\, L_i(w,y) \\ & + \quad q^2 \quad \Big[\quad 2\beta_2 \left\{\gamma_1\,\widehat{J}_{ie}(0) - \eta_1\,\widehat{J}_{ii}(0) + \cos 2\mu x \left(\gamma_2\,\widehat{J}_{ie}(2k) - \eta_2\,\widehat{J}_{ii}(2k)\right)\right\} \\ & \qquad\qquad + \frac{3}{2}\beta_3\,(1+\cos 2\mu x)\left(\gamma_0\,\widehat{J}_{ie}(k) - \eta_0\,\widehat{J}_{ii}(k) + r\right)^2\Big]\, L_i(w,y).\end{aligned}$$

Letting

$$\begin{aligned} A &:= 2\alpha_2\left(\gamma_0\,\widehat{J}_{ee}(k) - \eta_0\,\widehat{J}_{ei}(k) + 1\right) \\ B &:= 2\beta_2\left(\gamma_0\,\widehat{J}_{ie}(k) - \eta_0\,\widehat{J}_{ii}(k) + r\right)\end{aligned}$$

$$\begin{aligned} C_0 :=\, & 2\alpha_2\left(\gamma_1\,\widehat{J}_{ee}(0) - \eta_1\,\widehat{J}_{ei}(0)\right) \\ & + \ \frac{3}{2}\alpha_3\left(\gamma_0\,\widehat{J}_{ee}(k) - \eta_0\,\widehat{J}_{ei}(k) + 1\right)^2\end{aligned}$$

$$\begin{aligned} C_2 :=\, & 2\alpha_2\left(\gamma_2\,\widehat{J}_{ee}(2k) - \eta_2\,\widehat{J}_{ei}(2k)\right) \\ & + \ \frac{3}{2}\alpha_3\left(\gamma_0\,\widehat{J}_{ee}(k) - \eta_0\,\widehat{J}_{ei}(k) + 1\right)^2\end{aligned}$$

$$\begin{aligned} D_0 :=\, & 2\beta_2\left(\gamma_1\,\widehat{J}_{ie}(0) - \eta_1\,\widehat{J}_{ii}(0)\right) \\ & + \ \frac{3}{2}\beta_3\left(\gamma_0\,\widehat{J}_{ie}(k) - \eta_0\,\widehat{J}_{ii}(k) + r\right)^2,\end{aligned}$$

$$\begin{aligned} D_2 :=\, & 2\beta_2\left(\gamma_2\,\widehat{J}_{ie}(2k) - \eta_2\,\widehat{J}_{ii}(2k)\right) \\ & + \ \frac{3}{2}\beta_3\left(\gamma_0\,\widehat{J}_{ie}(k) - \eta_0\,\widehat{J}_{ii}(k) + r\right)^2,\end{aligned}$$

we have

$$\begin{aligned}\tau_e\ w_t = & [-w + \alpha_1 L_e(w,y))] \\ & + qA\cos\mu x\, L_e(w,y) \\ & + q^2\,(C_0 + C_2\cos 2\mu x)\, L_e(w,y)\end{aligned}$$

$$\begin{aligned}\tau_i \; y_t &= [-y + \beta_1 L_i(w,y))] \\ &\quad + qB \cos \mu x \, L_i(w,y) \\ &\quad + q^2 \left(D_0 + D_2 \cos 2\mu x\right) L_i(w,y).\end{aligned}$$

Letting $\Omega(x,t) = \left[\begin{array}{c} w(x,t) \\ y(x,t) \end{array}\right]$, $\text{Ł}_e \leftarrow \frac{L_e}{\tau_e}$, and $L_i \leftarrow \frac{L_i}{\tau_e}$, we thereby obtain

$$\begin{aligned}\Omega(x,t)_t = \; & \left\{ \left[\begin{array}{cc} \frac{-1+\alpha_1 J_{ee}*}{\tau_e} & -\frac{\alpha_1}{\tau_e} J_{ei}* \\ \frac{\beta_1}{\tau_i} J_{ie}* & \frac{-1-\beta_1 J_{ii}*}{\tau_i} \end{array}\right] + q \cos \mu x \left[\begin{array}{cc} \frac{A}{\tau_e} \cdot J_{ee}* & -\frac{A}{\tau_e} \cdot J_{ei}* \\ \frac{B}{\tau_i} J_{ie}* & \frac{-B}{\tau_i} J_{ii}* \end{array}\right] \right. \\ & \left. + \; q^2 \left(\left[\begin{array}{cc} \frac{C_0}{\tau_e} \cdot J_{ee}* & -\frac{C_0}{\tau_e} \cdot J_{ei}* \\ \frac{D_0}{\tau_i} J_{ie}* & \frac{-D_0}{\tau_i} J_{ii}* \end{array}\right] + \cos 2\omega x \left[\begin{array}{cc} \frac{C_2}{\tau_e} \cdot J_{ee}* & -\frac{C_2}{\tau_e} \cdot J_{ei}* \\ \frac{D_2}{\tau_i} J_{ie}* & \frac{-D_2}{\tau_i} J_{ii}* \end{array}\right] \right) \right\} \Omega,\end{aligned}$$

which we denote as

$$\Omega_t = L_0 \, \Omega + q \cos \mu x \; L_1 \, \Omega + q^2 \left(L_2^0 + \cos 2\mu x \; L_2^2\right) \Omega.$$

E.1.3 *Expansion of terms in the stimulus amplitude*

We make the ansatz $\Omega(x,t) = Z(x) e^{\lambda t}$ and expand the RHS terms, Z and λ, as power series of q, giving us, up to $\mathcal{O}(q^2)$

$$\begin{aligned}&\left(\lambda_0 + q\lambda_1 + q^2\lambda_2\right)\left(Z_0 + qZ_1 + q^2 Z_2\right) e^{\lambda t} \\ &\qquad = \left[L_0 \left(Z_0 + qZ_1 + q^2 Z_2\right) + q \cos \mu x \, L_1 \left(Z_0 + qZ_1 + q^2 Z_2\right)\right. \\ &\qquad\quad \left. + \; q^2 \left(L_2^0 + \cos 2\mu x \, L_2^2\right)\left(Z_0 + qZ_1 + q^2 Z_2\right)\right] e^{\lambda t}.\end{aligned} \tag{E.9}$$

E.1.3.1 ***Zeroth-order terms***

Noting that $\Omega(x,t)\ e^{-\lambda t} = Z_0(x) + qZ_1(x) + q^2 Z_2(x)$, we see that the 0th-order equation is our original linearization:

$$\mathcal{O}(q^0) : \lambda_0 Z_0 = L_0\, Z_0. \tag{E.10}$$

Therefore, the eigenfunctions Z_0 are spanned by $\{e^{i\omega^* x},\, e^{-i\omega^* x}\}$, so that Eq (E.10) becomes

$$\Phi_0 = \widehat{L}_0(m^*)\Phi_0, \tag{E.11}$$

where $\widehat{L}_0(m^*)$ is our original linearization, λ_0 is the eigenvalue we found earlier ($\lambda_0 = i\zeta_0 \neq 0$), and $\Phi_0 = \begin{bmatrix} \phi_{0,1} \\ \phi_{0,2} \end{bmatrix} \in \mathbb{C}^2$ is the associated eigenvector.

E.1.3.2 ***First-order terms***

Our $\mathcal{O}(q)$ equation is

$$\mathcal{O}(q) : \lambda_1 Z_0 + \lambda_0 Z_1 = L_0\, Z_1 + \cos\mu x\ L_1\, Z_0$$

$$\iff\ (\Lambda_0 - L_0)\, Z_1 = (-\Lambda_1 + \cos\mu x\ L_1)\, Z_0, \tag{E.12}$$

where $\Lambda_j = \lambda_j \cdot \mathrm{Id}_{2\times 2}$.

By the Fredholm alternative this has a solution iff

$$\left\langle (-\Lambda_1 + \cos\mu x\ L_1) Z_0, \widetilde{Z}_0 \right\rangle = 0 \quad \forall \widetilde{Z}_0 \in \mathrm{NS}(\Lambda_0 - L_0)^*. \tag{E.13}$$

Since $(\Lambda_0 - L_0)^* = \bar{\lambda}_0 - L_0^T$, the null space is spanned by $\Psi_0 e^{\pm i\omega^* x}$, where $\Psi_0 = \begin{bmatrix} \psi_{0,1} \\ \psi_{0,2} \end{bmatrix} \in \mathbb{C}^2$.

Hence, Ψ_0 is the eigenvector of $\widehat{L}_0^T(m^*)$ associated with $\overline{\lambda_0} = -\, i\zeta_0$. We use the usual inner product for complex vector-valued functions:

$$\begin{aligned}\left\langle \vec{a}\ f(x),\ \vec{b}\ g(x)\right\rangle &= \int \vec{b}^*\ \vec{a}\ f(x)g^*(x)\ dx\\ &= \int \left(\overline{b_1}a_1 + \overline{b_2}a_2\right) f(x)g^*(x)\ dx,\end{aligned}$$

where $\vec{a},\ \vec{b} \in \mathbb{C}^2$ and $f,\ g \in C^0(\mathbb{C}^1, \mathbb{C}^1)$.

We rewrite the Fredholm condition, Eq (E.13), as

$$\lambda_1 \left\langle Z_0,\ \widetilde{Z}_0\right\rangle = \left\langle \cos\mu x\ L_1 Z_0,\ \widetilde{Z}_0\right\rangle, \tag{E.14}$$

and recall that $\widetilde{Z}_0$ is spanned by $e^{\pm i\omega^* x}$. Taking the first eigenfunction, $e^{i\omega^* x}$ for $\widetilde{Z}_0$ implies that the RHS of Eq (E.14) is only nonzero if $\cos\mu x\ e^{i\omega^* x} = e^{-i\omega^* x}$ or $\cos\mu x\ e^{-i\omega^* x} = e^{-i\omega^* x}$. These reduce to corresponding algebraic conditions:

$$\pm\mu + \omega^* = -\omega^* \tag{E.15}$$

$$\pm\mu - \omega^* = -\omega^*. \tag{E.16}$$

Clearly, condition (E.15) is only satisfied when $\mu = \mp 2\omega^*$, while condition (E.16) is only satisfied when $\mu = 0$. Taking the second eigenfunction, $e^{-i\omega^* x}$, for $\widetilde{Z}_0$ provides similar results. Thus, we can only get nonzero λ_1 if $\mu = 0$ or $\pm 2\omega^*$. To calculate λ_1 for such μ, we rewrite $\cos\mu x$ in terms of complex exponentials and expand Z_0 in terms of its eigenfunctions:

$$Z_0 = a\Phi_0 e^{i\omega^* x} + b\Phi_0 e^{-i\omega^* x}.$$

For each μ value, we will consider each eigenfunction of L_0^T in turn, first letting $\widetilde{Z}_0 = \Psi_0 e^{i\omega^* x}$, and then $\widetilde{Z}_0 = \Psi_0 e^{-i\omega^* x}$, noting that this is equivalent to expanding $\widetilde{Z}_0$ in terms of its eigenfunctions. Thus, we consider Eq (E.14), expanded as

$$\begin{aligned}&\lambda_1 \left(\left\langle a\Phi_0 e^{i\omega^* x},\ \Psi_0 e^{\pm i\omega^* x}\right\rangle + \left\langle b\Phi_0 e^{-i\omega^* x},\ \Psi_0 e^{\pm i\omega^* x}\right\rangle\right) =\\ &\frac{1}{2}\left(\left\langle a\widehat{L}_1(m^*)\Phi_0\left[e^{i(\omega^*+\mu)x} + e^{-i(\omega^*-\mu)x}\right],\ \Psi_0 e^{\pm i\omega^* x}\right\rangle\right.\\ + &\left.\left\langle b\widehat{L}_1(m^*)\Phi_0\left[e^{i(\mu-\omega^*)x} + e^{-i(\mu+\omega^*)x}\right],\ \Psi_0 e^{\pm i\omega^* x}\right\rangle\right).\end{aligned}$$

For $\mu = \pm 2\omega^*$, $\widetilde{Z}_0 = \Psi_0 e^{im^*x}$ gives us

$$\lambda_1 b \left\langle \Phi_0,\, \Psi_0 \right\rangle = \frac{1}{2} a \left\langle \widehat{L}_1(m^*)\Phi_0,\, \Psi_0 \right\rangle,$$

and $\widetilde{Z}_0 = \Psi_0 e^{-im^*x}$ gives

$$\lambda_1 a \left\langle \Phi_0,\, \Psi_0 \right\rangle = \frac{1}{2} b \left\langle \widehat{L}_1(m^*)\Phi_0,\, \Psi_0 \right\rangle.$$

For $\mu = 0$ and $\widetilde{Z}_0 = \Psi_0 e^{im^*x}$, we have

$$\lambda_1 b \left\langle \Phi_0,\, \Psi_0 \right\rangle = b \left\langle \widehat{L}_1(m^*)\Phi_0,\, \Psi_0 \right\rangle,$$

while for $\widetilde{Z}_0 = \Psi_0 e^{-im^*x}$, we instead obtain

$$\lambda_1 a \left\langle \Phi_0,\, \Psi_0 \right\rangle = a \left\langle \widehat{L}_1(m^*)\Phi_0,\, \Psi_0 \right\rangle.$$

Denoting the left inner product (which we note is now simply the scalar product in $\mathbb{C}^2$) as I_1 and the right inner product as I_2, we have $\lambda_1 = \frac{I_2}{I_1}$ for $\mu = 0$, while for $\mu = 2\omega^*$ we have 2 equations that can be written as a 2×2 eigenvalue problem:

$$\lambda_1 \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 & \frac{I_2}{2I_1} \\ \frac{I_2}{2I_1} & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$$

Since this is a 2×2 circulant matrix, the eigenvalues are given as the sum and difference of the two distinct entries. In total, we have

$$\lambda_1 = \frac{I_2}{I_1} \qquad \text{for } \mu = 0 \tag{E.17}$$

$$\lambda_1 = \pm\frac{I_2}{2I_1} \qquad \text{for } \mu = 2\omega^*. \tag{E.18}$$

Thus, it seems like there is some ambiguity in the stability for $\mu = 2\omega^*$. However, we can make some sense of these results by recalling the stability curves in the $q - a_{ee}$ plane in Fig 19 of Chap 3, curves that we will derive near the onset of instability below in Sec *E.2*. We observed in Sec *3.2.2.3* how the even and odd symmetries that arose are due to the odd and even symmetries of the stimulus with respect to q. Additionally, in Sec *E.2* below, we will see that the stability curves for $\mu = 0$ and $2\omega^*$ are locally linear due to the presence of the $\mathcal{O}(q)$ eigenvalue terms that we have just found. Therefore, while the stability curve for

$\mu = 0$ will be a straight line that goes through $(q,\, a_{ee}) = (0,\, a^*_{ee})$, that for $\mu = 2\omega^*$ must be a nonsmooth V- or upside-down V-shaped curve with a local extremum at $(0,\, a^*_{ee})$, as we see in Fig 19B.

In fact, for $\mu = 2\omega^*$, we found curves corresponding to both signs for λ_1. The upper V-shaped branch is not shown in Fig 19B because it did not correspond to a loss of stability of the steady state. It also corresponded to another Hopf curve, wherein the real parts of a pair of complex-conjugate eigenvalues become positive for $a_{ee} > a^*_{ee}$ as q increased from 0. However, when $q = 0$, the real parts of *two* pairs of complex-conjugate eigenvalues become positive as a_{ee} increases from below to above a^*_{ee}, resulting from the two modes that we considered in the above calculation, $\pm\omega^*$. When the stimulus is turned on, the symmetry is broken, so that the two pairs of eigenvalues cross the imaginary axis *sequentially*. That is, the stimulus has selected which of the two modes, $\pm\omega^*$, will destabilize; positive q leads to one mode, negative q leads to the opposite mode. Hence, the curves may be better understood as two stability curves (lines) for the two different modes, $\pm\omega^*$, of opposite-signed slope. For nonzero q values, as a_{ee} increases from below to above the upside-down V, the steady state is lost as the real parts of one pair of eigenvalues become positve; as a_{ee} further increases from below to above the upper V curve, the other mode destabilizes as the real parts of the other pair of eigenvalues become positive. However, since the steady state has *already* destabilized, the upper V curve does not correspond to the stability boundary of the network. Hence, the stability boundary is always the upside-down V-shaped curve. We will observe similar behaviors for the quadratic stability curve of $\mu = m^*$ below.

E.1.3.3 *Second-order terms*

We are now in position to move on to $\mathcal{O}(q^2)$ terms for $\mu \neq 0,\, \pm 2\omega^*$ (since the stabilities of these are obviously dominated by the $\mathcal{O}(q)$ eigenvalues). We will find that we need the solution to Eq (E.12) after all. We may now take $\lambda_1 = 0$, so that the RHS, written in terms of the eigenfunctions of L_0, is

$$\frac{1}{2}\widehat{L}_1(m^*)\Phi_0\left(a\, e^{i(\omega^*+\mu)} + a\, e^{i(\omega^*-\mu)} + b\, e^{-i(\omega^*+\mu)} + b\, e^{-i(\omega^*-\mu)}\right). \tag{E.19}$$

This defines the form of Z_1, so that

$$\begin{aligned} Z_1 \; &= \Phi_1 \left(a\, e^{i(\omega^* - \mu)} + b\, e^{-i(\omega^* - \mu)}\right) \\ &+ \Phi_2 \left(a\, e^{i(\omega^* + \mu)} + b\, e^{-i(\omega^* + \mu)}\right), \end{aligned} \tag{E.20}$$

where $\Phi_1,\, \Phi_2 \in \mathbb{C}^2$. Plugging this into the LHS of Eq (E.12) and equating it with Eq (E.19) provides two linear systems that define Z_1:

$$\begin{aligned} \left(\Lambda_0 - \widehat{L}_0(m^* + k)\right) \Phi_1 \; &= \tfrac{1}{2} \widehat{L}_1(m^*)\Phi_0 \\ \left(\Lambda_0 - \widehat{L}_0(m^* - k)\right) \Phi_2 \; &= \tfrac{1}{2} \widehat{L}_1(m^*)\Phi_0 \end{aligned} \tag{E.21}$$

We can now proceed to our $\mathcal{O}(q^2)$ terms.

$$\mathcal{O}(q^2) : \lambda_2 Z_0 + \lambda_1 Z_1 + \lambda_0 Z_2 = L_0\, Z_2 + \cos \mu x\; L_1\, Z_1 + \left(L_2^0 + \cos 2\mu x\; L_2^2\right) Z_0$$

$$\iff \; (\Lambda_0 - L_0)\, Z_2 = E,$$

$$\text{where } E \mathrel{\mathop:}= \left(-\Lambda_2 + \left(L_2^0 + \cos 2\mu x\; L_2^2\right)\right) Z_0 + \cos \mu x\, L_1\; Z_1,$$

since we can assume $\lambda_1 = 0$. Again using Fredholm, this has a solution iff $\left\langle E,\, \widetilde{Z}_0 \right\rangle = 0$. We proceed exactly as we did for the $\mathcal{O}(q)$ case by expanding Z_0 as $ae^{i\omega * x} + be^{-i\omega^* x}$, rewriting the cosine terms as exponentials, writing out Z_1 using Eq (E.20) and (E.21), and treating $\widetilde{Z}_0$ sequentially as $e^{\pm i\omega^* x}$. After doing so, we obtain different sets of equations depending on μ. We begin with the simpler case of general μ; i.e., $\mu \neq 0$, ω^*, or $2\omega^*$. In this case, we obtain

$$\lambda_2 \left\langle \Phi_0,\, \Psi_0 \right\rangle = \left\langle \widehat{L}_2^0(m^*)\Phi_0,\, \Psi_0 \right\rangle + \frac{1}{2} \left\langle \widehat{L}_1(m^* - k)\Phi_1,\, \Psi_0 \right\rangle + \frac{1}{2} \left\langle \widehat{L}_1(m^* + k)\Phi_2,\, \Psi_0 \right\rangle.$$

Labeling the two new, rightmost inner products in order as I_3 and I_4 (recall, $I_2 = \left\langle \widehat{L}_1(m^*)\Phi_0, \Psi_0 \right\rangle$ and does not appear here) gives us

$$\lambda_2 = \frac{1}{I_1} \left(I_2 + \frac{I_3 + I_4}{2} \right) \qquad \text{for } \mu \neq 0,\ \omega^*,\ 2\omega^*. \tag{E.22}$$

For $\mu = m^*$, we obtain the following two equations, where the first corresponds to $\widetilde{Z}_0 = e^{i\omega^* x}$ and the second to $\widetilde{Z}_0 = e^{-i\omega^* x}$:

$$\begin{aligned}
\lambda_2 b \left\langle \Phi_0,\, \Psi_0 \right\rangle &= b \left\langle \widehat{L}_2^0(m^*)\Phi_0,\, \Psi_0 \right\rangle + \frac{a}{2} \left\langle \widehat{L}_2^2(m^*)\Phi_0,\, \Psi_0 \right\rangle \\
&\quad + \frac{a+b}{2} \left\langle \widehat{L}_1(0)\Phi_1,\, \Psi_0 \right\rangle + \frac{b}{2} \left\langle \widehat{L}_1(2m^*)\Phi_2,\, \Psi_0 \right\rangle \\
\lambda_2 a \left\langle \Phi_0,\, \Psi_0 \right\rangle &= a \left\langle \widehat{L}_2^0(m^*)\Phi_0,\, \Psi_0 \right\rangle + \frac{b}{2} \left\langle \widehat{L}_2^2(m^*)\Phi_0,\, \Psi_0 \right\rangle \\
&\quad + \frac{a+b}{2} \left\langle \widehat{L}_1(0)\Phi_1,\, \Psi_0 \right\rangle + \frac{a}{2} \left\langle \widehat{L}_1(2m^*)\Phi_2,\, \Psi_0 \right\rangle .
\end{aligned}$$

We've encountered all but the third inner products already. Labeling the third one as I_5, we get the following 2×2 eigenvalue problem:

$$I_1 \lambda_2 \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} I_2 + \frac{I_3+I_4}{2} & \frac{I_3+I_5}{2} \\ \frac{I_3+I_5}{2} & I_2 + \frac{I_3+I_4}{2} \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} .$$

Once again, we have a circulant matrix whose eigenvalues λ_2 are given by the addition and subtraction of the entries. We designate these by whether we add (λ_2^+) or subtract (λ_2^-) the entries:

$$\lambda_2^+ = \frac{1}{I_1}\left(I_2 + I_3 + \frac{I_4 + I_5}{2} \right) \tag{E.23}$$

$$\lambda_2^- = \frac{1}{I_1}\left(I_2 + \frac{I_4 - I_5}{2} \right) . \tag{E.24}$$

These eigenvalues, again arising from the two modes $\pm\omega^*$, will lead to quadratic curves in the $q - a_{ee}$, as we will see below in Sec *E.2*. While they lack the symmetry of the two V-shaped linear curves found for $\mu = 2\omega^*$, they are analogous, producing the same behaviors: The lower curve defines the stability boundary so that, for nonzero q, as a_{ee} increases from below to above the lower curve, the real parts of one pair of complex-conjugate eigenvalues become positive, and as a_{ee} further increases to be above the upper quadratic curve, the real parts of the second pair of complex-conjugate eigenvalues become positive. We now turn to the stability curves discussed here and shown in Fig 19B, and see how we can use the eigenvalues found in this section to analytically determine these curves near $(q,\, a_{ee}) = (0,\, a_{ee}^*)$.

E.2 DETERMINATION OF THE STABILITY CURVES

In order to determine the Hopf curves (in particular the curves that determine the stability of the steady state) in the $q - a_{ee}$ plane, we first recall that we expanded λ as a function of q:

$$\lambda = \lambda_0 + \lambda_1\, q + \lambda_2\, q^2 \quad \text{for small } q.$$

We know that λ_0 is dependent on a_{ee}, and that $\mathbb{R}(\lambda) = 0$ when $a_{ee} = a_{ee}^*$, the critical a_{ee} value when $q = 0$, so that

$$\mathbb{R}(\lambda)(a_{ee}, q) = \alpha(a_{ee} - a_{ee}^*) + \beta\, q + \delta\, q^2, \text{ where } \beta = \mathbb{R}(\lambda_1) \text{ and } \delta = \mathbb{R}(\lambda_2).$$

Letting $\mathbb{R}(\lambda) = 0$ provides the critical curve in $a_{ee} - q$ space:

$$(a_{ee} - a_{ee}^*) + \beta\, q + \delta\, q^2 = 0. \tag{E.25}$$

We note that we are only concerned with finding δ if β is 0, since otherwise the local behavior will be dominated by the linear term. Letting $\rho = a_{ee} - a_{ee}^*$, we have $\alpha = \frac{d}{d\rho}\mathbb{R}(\lambda)|_0$. Letting $A(\rho) = \widehat{L}_0(\rho; m^*)$ be the linearization about the critical wavenumber m^* (when $a_{ee} = a_{ee}^*$), we have

$$A(\rho) = \begin{bmatrix} \frac{-1+\alpha_1(\rho + a_{ee}^*)\widehat{K}_e(m^*)}{\tau_e} & -\frac{\alpha_1}{\tau_e}\widehat{J}_{ie}(m^*) \\ \frac{\beta_1}{\tau_i}\widehat{J}_{ei}(m^*) & \frac{-1-\beta_1\widehat{J}_{ii}(m^*)}{\tau_i} \end{bmatrix},$$

where $(\rho + a_{ee}^*)\widehat{K}_e(m^*) = a_{ee}\widehat{K}_e(m^*) = \widehat{J}_{ee}(m^*)$. Recall from our linearization, Eq (E.11), that we have the following equations

$$A(0)\,\Phi_0 = i\zeta_0\,\Phi_0 = \lambda_0\,\Phi_0$$
$$A^*(0)\,\Psi = -i\zeta_0\,\Psi = \overline{\lambda_0}\,\Psi.$$

More generally,

$$A(\rho)\,\Phi(\rho) = \lambda(\rho)\,\Phi(\rho).$$

Differentiating with respect to ρ, evaluating at 0, and keeping the real parts provides

$$A'(0)\,\Phi_0 + A(0)\,\Phi'(0) = \alpha\,\Phi_0 + \lambda_0\,\Phi'(0)$$

$$\implies (\Lambda_0 - A(0))\,\Phi'(0) = (A'(0) - \alpha\,\mathrm{Id})\,\Phi_0,$$

where some of the previous variables we have used correspond to quantities here; i.e., $\lambda_0 = \lambda(0)$, $\Phi_0 = \Phi(0)$. By the Fredholm alternative this has a solution $\Phi'(0)$ iff

$$\begin{aligned} \langle A'(0) - \alpha\,\mathrm{Id})\,\Phi_0, \Psi\rangle = 0 \\ \implies \alpha\,\langle \Phi_0, \Psi\rangle = \langle A'(o)\,\Phi_0, \Psi\rangle \\ \implies \alpha = \frac{\langle A'(0)\,\Phi_0, \Psi\rangle}{\langle \Phi_0, \Psi\rangle}, \end{aligned}$$

where

$$A'(0) = \begin{bmatrix} \frac{\alpha_1 \widehat{J}_{ee}(m^*)}{a_{ee}} & 0 \\ 0 & 0 \end{bmatrix}.$$

Therefore, we have all of the values necessary to determine the stability curve, Eq (E.25). Plotting these for different μ on the graphs of those determined numerically with AUTO shows very good agreement between numerical and theoretical curves (Fig 24).

APPENDIX F

SPECIFIC GENERAL-PURPOSE GPU PROGRAMMING: OUTLINE FOR SIMULATING SPATIALLY EXTENDED WILSON-COWAN NETWORKS

F.1 BACKGROUND

Since the cortex is better approximated as a sheet of neurons rather than a line, we wanted to explore how the network behaved when extended in two spatial dimensions. This necessitated using custom code, and we first turned to MATLAB since it allows for rapid prototyping. However, the simulations were excruciatingly slow for networks of around 100×100 populations. Indeed, this led to misleading results: cases in which we thought we saw instances of spatiotemporal pattern formation turned out to be transients that evolved to steady states. Clearly, a framework allowing for more rapid simulations was necessary to implement anything more than a few simple cherry-picked examples.

Throughout the years, high-performance computing has often been implemented in the programming languages C and C++. These are compiled languages with vast libraries for scientific computing, and are known to produce rapidly executed code. For speed in simulations, they are hard to beat. Indeed, XPP-AUTO is written in C, as is the operating system Linux, and the high-performance code of the open source finite element solver FEniCS, while being accessible in both C++ and Python implementations, is written in C++ [128].

However, the landscape in high-performance computing has shifted over the past decade, as GPUs (Graphical Processing Units) with many more processors than CPUs (Central

Processing Units) have become available. While these processors are far simpler than those found in modern CPUs, they often more than make up for this by allowing for massively parallel computations. With the development of a hardware architecture — called CUDA architecture — that facilitated access to all of the processing and memory elements of the device, along with a software interface in which to access these resources, came the advent of true GPGPUs: General-purpose GPUs. The language NVIDIA developed to go along with their CUDA architecture is CUDA C; in fact there are now two languages, often referred to as CUDA C/C++. (Note: NVIDIA now simply treats "CUDA" as a term to refer to its GPU architecture and programming languages. Originally they used it as as an acronym for Compute Unified Device Architecture.)

Since CUDA C seemed to require very little knowledge over and above C, we concluded that if we went to the trouble to program our network in C, we might as well try out CUDA C. We found (expectedly) massive gains in performance compared to our MATLAB code. Instead of taking minutes to run 100×100 networks for hundreds of time steps, our CUDA C code takes seconds to run 512×512 networks, or indeed even 1024×1024 networks. In the end, to run the Wilson-Cowan network for a time period of thousands of milliseconds takes minutes, allowing us to sweep through sets of parameters to characterize the network dynamics. Of course, how well such code performs compared to an implementation done purely in C will depend on the platform, including the particular CPU and GPU used. For our setup, which included an NVIDIA GeForce GTX 970 GPU, we found that the CUDA C implementation was ≈ 3.5 times faster than the C implementation, more than justifying the small amount of additional education required. To that end, we turned to a clear and helpful book, *CUDA by Example* [129]. This text clearly and methodically progresses from very simple to fairly sophisticated examples, only assuming a knowledge of C. Additionally, the authors provide freely available code on GitHub (https://github.com/CodedK/CUDA-by-Example-source-code-for-the-book-s-examples-). Importantly, this includes graphics helper classes with simple programmer-facing implementations. Thus, with only a few more lines of code we can visualize simulations as they run in real time.

In this appendix, we will outline the procedures we used to produce these simulations. As the code will be made available shortly for anyone to use, we will generally keep the discussion herein at a higher level, only including a simple example and some code snippets when helpful. Our full code should be included as supplementary files to the forthcoming paper that will be based on Chap 3.

As almost our entire CUDA and CUDA C knowledge base stems from the information in *CUDA by Example*[129], we will generally avoid citing the text, with only some exceptions to point the reader to specific useful infomation. We also note that the reader should refer to [129] and NVIDIA's documentation for instructions involving the installation of CUDA C, how to compile "*.cu" files, and what is necessary for your system so that your code cooperates well with OpenGL. Finally, I am not a programmer; in fact, I have only taken two formal computer science courses, the last of which was early on in my undergraduate career. The errors that are likely to exist in this brief exposition are my own. Please have some grains of salt and your favorite search engine at the ready.

We first introduce the GPU hardware and programming distinctions that arise as a result of this hardware as compared with traditional serial programming for a CPU. These software distinctions are fundamental to programming for GPUs. To illustrate these differences and the power of GPUs, we then explore the simple canonical example of parallel programming: adding two vectors. In GPUs, no loops are required to do so, and we will see how this is implemented. In doing so, we'll gain the understanding necessary to implement many basic routines that may be of interest to the scientific reader. We then look more at our specific implementation: simulating neural field equations. An important aspect of speeding up performance for such systems is using fast Fourier transforms to speed up the computation of the convolutions that provide nonlocal coupling. We see how to do so, and then outline the remainder of our implementation. We hope this appendix might help others who are interested in simulating large networks do so using the massively parallel capabilities represented within the modern-day GPU.

F.2 GPU HARDWARE AND CONSEQUENCES FOR SOFTWARE

GPUs are constructed differently from CPUs, and it is difficult to make meaningful, simple, quantitative performance comparisons. Generally speaking, (GP)GPUs are designed for (general) parallel computations, with an architecture consisting of hundreds to thousands of simple processing units known as ALUs, arithmetic logic units (known as "CUDA cores" for NVIDIA GPUs or "stremaing processors" AMD GPUs). CUDA C was the first widely available language written to easily and efficiently run code on both GPUs and CPUs. Since (1) these represent two separate pieces of hardware with distinct processing units and memory, only connected through communication pipelines, and (2) a programming interface that simplifies and automates the allocation of the hundreds of equivalent computing units would greatly facilitate GPGPU programming, two main distinctions arise when programming in CUDA C. These distinctions rely on whether we are talking about item in the purview of the GPU or of the CPU. Since the GPU is secondary to the CPU and its associated memory, the former are *device* items, whereas those for the CPU are *host* items. The first distinction involves functions, and the second distinction involves memory.

F.2.1 *A distinction of functions*

Code that is intended to run on the GPU must somehow be distinguished from code intended to run on the CPU. Thus, device functions are unambiguously marked as such. Device functions that can be called from host functions are marked with the precursor `__global__`, whereas those that can only be called from other device functions are marked with the precursor `__device__`. Additionally, although CUDA C does a remarkable job at keeping the specific allocation of resources automated, we do have some choices in the resource usage. That usage comes down to how we organize so-called *threads*. These are abstract structures associated with processes that we consider to be run parallel to each other. So, for example, we can determine how many threads we would like based on the number of parallel computations we need. If we're adding two vectors of length N, we'll want N threads. These are abstract in the sense that the programmer does not know, and has no control over, how these threads are distributed over the GPU's ALUs.

Threads are further grouped into *warps* and *blocks*. Warps have more to do with how CUDA executes at a low level, and with ways to address how variables are handled efficiently in device memory. They are fixed groupings of threads: there is no programmer control over how to organize threads with warps. See Sec 6.2.3 in [129] for more information. In contrast, blocks form a programmer-adjustable way to organize threads. Blocks can be broken up into 2-D grids, and threads into 3-D grids, so that each block is composed of `(blockDim.x)`·`(blockDim.y)`·`(blockDim.z)` threads. The particular block and thread structure used is in part preference, and in part, again, based on memory considerations. In particular, all threads within a block can share memory. As a result, there are a limited number of threads per block allowed, usually 512. Currently, our implementation of the simulations of the 2-D Wilson-Cowan network does not make use of such shared memory (see Sec 5.3 in [129] for more information). Instead, we simply use a two-dimensional block-and-thread structure, wherein we have M blocks that each have N threads per block, forming an $M \times N$ reference grid. Note that, since we do not make use of shared memory, we could just as easily have used $M \times N$ blocks with 1 thread per block. That is to say, the structure of the grid of threads we choose need not reflect the underlying spatial structure of the system being operated on.

Whatever structure we choose, the CUDA runtime needs to know what that structure is when running a device function. Thus, when a device function is called, the structure must be made explicit. For example, if we run the function `__device__ void kernel( float* data )`, we need to tell it how many blocks and threads to launch, and in what configuration. This is done by specifying these in the function call using triple angular-bracket notation. For example, to call the kernel with 5 blocks and 5 threads per block, we would write `kernel<<<5,5>>>(data)`. This is where the vastly different behavior of parallel code comes into play. Instead of simply running the function `kernel`, *25 instantiations of the function are run in parallel* (or, should be thought to run in parallel; again, the exact execution is left up to the CUDA runtime). Each copy of the function is indexed according to its position within the 5×5 block-thread structure we chose. If `data` is an array with 25 elements, we can map these indices to array indices so that each instantiation will perform an operation on a different element of `data` in parallel. We see a specific implementation below in Sec *F.3*. Of course, in classical serial programming, we would instead run a loop through the array elements to operate on all of them. This is the main conceptual difference between GPGPU programming and classical serial programming. In essence, all of the rest are simply details.

F.2.2 ***A distinction of memory and variables***

Recall, the CPU and GPU each have access to their own independent stores of memory. Yet, variable values obtained through declarations and computations done in one environment need to somehow pass to the other environment. For example, above, the device function `kernel` operated on the array `data`. We see that, in some sense that we will explore soon, `data` was declared within host code, since it is passed as an argument to `kernel`. Computations will be done on `data` within `kernel`, the results to which we will presumably want access from the host side. For example, you may want to plot the results or save them to the hard drive. However, *variables that live on the device side cannot* ***directly*** *be accessed to read or write on the host side.* We will see how we can access them below. But the lesson is that, in general, we simply have to keep track of host variables and device variables. We note that the read restriction is one-way: device functions *can* read variables allocated in host memory that are passed to them as arguments. They cannot, however, write to them. We are now at a point where a simple example will help clarify these ideas and make explicit how to handle device and host functions and variables.

F.3 A SIMPLE EXAMPLE

We now consider the prototypical, embarrassingly parallel problem of adding two vectors. We follow Sec 4.2.1 in [129] closely.

```
#include<stdio.h>

// Constant, accessible by both device and host functions
#define N 5
```

```
__global__ void add( int *v, int *u, int *w )
{
   //thread index
   i = blockIdx.x;

   // This ensures we do not try to access memory locations outside of the vectors
   if (i < N)
      w[i] = u[i] + v[i];
}

int main(void)
{

   // Declare vectors (pointers, equivalent to arrays)
   int *host_u, *host_v, *host_w;
   int *dev_u, *dev_v, *dev_w;

   // Allocate host memory for the host vectors
   host_u = (int *)malloc( N * sizeof(int) );
   host_v = (int *)malloc( N * sizeof(int) );
   host_w = (int *)malloc( N * sizeof(int) );

   // Allocate device memory for the device vectors
   cudaMalloc( (void **)&dev_u, N * sizeof(int) );
   cudaMalloc( (void **)&dev_v, N * sizeof(int) );
   cudaMalloc( (void **)&dev_w, N * sizeof(int) );

   // Fill in host vectors, setting w to values different from the sum
   for( i= 0; i < N; i++ )
   {
      host_u[i] = i;
      host_v[i] = -i;
      host_w[i] = i;
   }

   // Print w for later comparison:
   printf( "Vector w is initialized to be [ " );
   for( i = 0; i < N; i++ )
   {
     printf( "%d, ", host_w[i] );
   }
   printf( "]\n\n" );

   // Copy these to device vectors u and v
   cudaMemcpy( dev_u, host_u, N * sizeof(int), cudaMemCpyHostToDevice );
   cudaMemcpy( dev_v, host_v, N * sizeof(int), cudaMemCpyHostToDevice );
```

```
  // Now we can add them
  add<<<N,1>>>( dev_u, dev_v, dev_w );

  // We want to read out the result, so copy device w to host
  cudaMemcpy( host_w, dev_w, N * sizeof(int), cudaMemcpyDeviceToHost );

  // We should get all 0s:
  printf( "The resulting vector is w = [ " );
  for( i = 0; i < N; i++ )
  {
    printf( "%d, ", host_w[i] );
  }
  printf( "]" );

  // Free memory
  free( host_u );
  free( host_v );
  free( host_w );

  cudaFree( dev_u );
  cudaFree( dev_v );
  cudaFree( dev_w );

  return 0;

}
```

This program is in some sense deceptively long. In particular, we note that 17 of the 31 lines of code are simply memory management. However, we're using a variant of C, so we would expect nothing different! Let's tackle these memory lines by first walking through the host function `main`.

F.3.1 *Host side: Memory management and considerations*

After we instantiate the variables, we allocate memory for them. `malloc` and `cudaMalloc` both allow for dynamic arrays, so that we can input size in the command line, for instance. The `cudaMalloc` lines in particular are somewhat intimidating. Note that for neither memory allocation routine does the type matter; all that matters is that the right size is allocated, ensured by the (last) argument. For `malloc`, the type is "fixed" ex post facto by typecasting the returned value to the variable types `int *`. The `cudaMalloc` function allocates memory by accepting a pointer to the vector (e.g., `&dev_u`). Since the type does not matter, only the memory amount, the function only accepts `void` pointers. Of course, `dev_u` itself is a pointer, so we typecast it as a pointer to a pointer of type `void`: `(void **)&dev_u`. Thus, we see why above, in Sec *F.2.2*, the host side knew about the device variable `data`: it was declared and allocated memory on the host side beforehand.

Additionally, although the device functions may manipulate the data passed to it, we usually want them to have a chosen set of initial values. This can be done either on the host side or device side. Here, we see the former method. We fill in the host variables, and then copy them to the device variables. We could instead have initialized the values in the device function before adding them, saving ourselves this trouble. However, the values may come from external values that must be read in via host code, forcing us to implement a solution like we do here. The worst this does, of course is add an extremely slight time and code overhead.

Finally, we are in a position to take advantage of the GPU. A simple call to the device function `__global__ void add` is all that we need on the host side. The `<<<5,1>>>` notation indicates we'd like 5 blocks with one thread each. We'll go over the implementation of this function in just a bit. The result of the computation is stored in `dev_w`. Of course, we can't access device variables on the host side, so we must use `cudaMemcpy` in the reverse direction to copy the values to the host vector `host_w`. Now we can print the values (perhaps to a file if desired) and finally free up the memory that was allocated to the host and device variables. Now, let's take a look at the device function `add`.

F.3.2 *Device side: N instantiations*

We can immediately identify the device function by the `__global__` precursor, indicating a device function that can be called from host code. Of course, since it is called from the host side, it can't return any values, and so it is given the type of `void`. Its arguments are the 3 pointers (i.e., arrays; i.e., vectors). When this function is called, the block-thread structure was specified as N blocks, 1 thread per block. Thus, *N instantiations of* `add` *are created*, where each is indexed by its block and thread coordinates. Here we see that the block indices are given as `blockIdx.x` and `blockIdx.y`, while the thread indices are similarly given as `threadIdx.x`, `threadIdx.y`, and `threadIdx.z`. Thus, the local variable `i` is set to 0 in the first instantiation, and is set to N-1 (=4) in the last instantiation. Each instantiation has the very simple task of assigning the sum of $u[i]$ and $v[i]$ to a third variable, $w[i]$. Of course, if we had accidentally called for more threads than there were elements of the array, memory values that don't belong to `dev_w` would be written to, with different potential unpleasant results ensuing. To ensure we only write to memory locations that are indeed contained with `dev_w`, we add a simple check condition: proceed with the operation only if the index is less than the size of the array. Somewhat embarrassingly, this simple example very nearly covers the extent to which our simulation code employs the GPU, as the integration step is just vector addition. The coupling of the populations is the only other aspect of the network integration that will benefit from some GPU treatment, and to visualize we use the GPU as well. Of course, there are many bells and whistles that we've added to make it more functional, but nearly all of these are implemented on the host side.

F.4 NEURAL FIELD IMPLEMENTATION

To simulate neural field equations, we need to integrate excitatory and inhibitory populations forward in time. At each time step, the amount of excitation and inhibition that each population receives changes based on the values at the previous time step of the populations to which it is coupled. Once these couplings are determined, at each time step we can perform a standard integration routine, such as the Euler method, in a highly parallel routine involving simple vector addition, like the example above. For neural fields, the coupling is computed as convolutions of a kernel with the local populations. We will first examine how we can perform this operation efficiently using the GPU, and then outline the steps that we use to simulate the network.

F.4.1 *Convolutions*

Each population receives excitation and inhibition that is computed by convolving the chosen kernel (e.g., Gaussian) with the respective populations. This involves N multiplications and additions. Since there are N populations, this leads to an algorithm of $\mathcal{O}(N^2)$ computational complexity; i.e., $\sim N^2$ computational steps are required. Far fewer steps can be taken if we instead use Fourier transforms (FTs) to compute this integral. The ability to do so is encapsulated within the Convolution Theorem, which simply states that the convolution of two functions is the inverse FT of the product of the FTs of the two functions. We show this for the continuous case of two scalar-valued functions on $\mathbb{R}$, $f(x)$ and $g(x)$. That is, we want to show

$$f * g(x) = \mathcal{F}^{-1}(\mathcal{F}(f(x)) \cdot \mathcal{F}(g(x)).)$$

Since

$$f * g(x) = \int\limits_{-\infty}^{\infty} f(y)g(x-y)\, dy,$$

applying the FT gives us

$$\mathcal{F}(f * g(x)) = \int\limits_{-\infty}^{\infty} e^{-i\omega x} \int\limits_{-\infty}^{\infty} f(y)g(x-y)\, dy\, dx.$$

Letting $z = x - y$, we have

$$\int\limits_{-\infty}^{\infty} e^{-i\omega(z+y)} \int\limits_{-\infty}^{\infty} f(y)g(z)\, dy\, dz$$

,

so that we can separate the integrals, giving

$$\int\limits_{-\infty}^{\infty} f(y)e^{-i\omega y}\, dy \cdot \int\limits_{-\infty}^{\infty} g(y)e^{-i\omega y}\, dy = \mathcal{F}(f) \cdot (F)(g).$$

Applying the inverse FT to both sides provides the result. The theorem also holds for discrete functions and the discrete FT ([130], Sec 8.5-2). Why go through the trouble? Because of the discrete fast Fourier transform (DFFT), which turns the computation from an $\mathcal{O}(N^2)$

operation to an $\mathcal{O}(N\log(N))$ one when we use the DFFT. In CUDA, this is implemented using the cuFFT API, and is modeled after the well-known FFTW implementation that is used in languages such as C/C++. We note that, unless the data is padded with zeros, e.g., periodic boundary conditions are implicitly used.

When cuFFT is employed, planning phases are first developed to allow the algorithm to perform efficiently on the given hardware:

```
  cufftPlan2d( &FFTplanForward, N, N, CUFFT_R2C );
  cufftPlan2d( &FFTplanReverse, N, N, CUFFT_C2R );
```

These plans, the first arguments, are computed once per simulation and then stored for future steps. Since each convolution involves the FT of the kernel, we also need only compute it once per simulation for each of the excitatory and the inhibitory kernels. For the this and the remainder, we only show the steps for the data involving the excitatory populations:

```
cufftExecR2C( FFTplanForward, (cufftReal *)devKernE, (cufftComplex
    *)devKernE_FFT );
```

At each time step, we then compute the FTs of the neuronal populations:

```
cufftExecR2C( FFTplanForward, (cufftReal *)devDataE, (cufftComplex
    *)devDataE_FFT );
```

and compute the pointwise (complex) product of the FFTs in a simple device function, storing the product back into the first argument:

```
fftproduct<<<BLOCKS,BLOCKSIZE>>>( devDataE_FFT, devKernE_FFT );
```

Finally, we inverse the transform, storing the convolved data in the last argument:

```
cufftExecC2R( FFTplanReverse, (cufftComplex *)devDataE_FFT, (cufftReal
    *)devDataE_convolved );
```

The convolved data is then input into the Euler integration step. We show the relevant part:

```
u  = devDataE[i];
KE = devDataE_convolved[i];
KI = devDataI_convolved[i];
s  = stimulus[i];
```

```
devDataE[i] += dt * ( -U + f( beta * (aee*KE - aie*KI - the + q*s) / taue;
```

f is the firing rate function, defined as a device function.

F.4.2 *Implementation outline*

Having seen how the integration is performed, the remainder of the implementation is straightforward. Since the time overhead is minimal, and the code overhead makes for less streamlined code, steps that are implemented once are done on the host side. We outline the steps here.

- Allocate memory for variables

  This includes host and device arrays, such as the excitatory and inhibitory population values, the stimulus pattern, the kernel arrays, and the FFTs and convolved data

- Initialize variables, including:
  - Initial conditions for the excitatory and inhibitory populations (which might be random or pulled from the final state of a previous run)
  - The parameter values (the coupling strengths, timescales, etc.)
  - The stimulus and kernels. For example, we might want a striped cosine stimulus and gaussian kernels.

- Integrate (and animate)

  At each time step, compute the next values of the excitatory and inhibitory populations by (1) computing the amount of excitation and inhibition received for each population through the convolution steps outlined above, and (2) integrating forward with, e.g., an Euler time step, also shown above. We wish to view the results, so adjust the bitmap values based on the u and v values. These are output via the graphics API provided in [129].

- Free memory

  Take out the garbage. Memory leaks are annoying.

F.4.3 *Embellishments*

That's all there is to the basic implementation. However, to make a more useful version, there are additional features we'd like to have. These can all be classified under interactivity, either at the command line or during the simulation. The point is, we'd like to change parameters without having to recompile the code every time, and we'd like to be able to get outputs that we can port into other programs for further analysis and visualization. While there are some exceptions, we implement these in ways that usually don't involve addtional device code. Here we provide an outline of the additional features our code provides.

- Parameter input

  By declaring variables with preprocessor macros, we can define the parameters based on inputs that come at runtime. By taking input at the command line, files can be read in and parsed for these parameter values. Hence, we can easily allow the network parameters to only be defined at runtime by reading in a separate parameters file.

- Initial conditions and stimulus inputs

  Other files can be read in and parsed for values of interest. For example, instead of beginning with random initial conditions, we may wish to start where a previous simulation left off. Or we may wish to create stimuli in a different program (e.g., perhaps we already have images that we wish to test) and use them. These possibilities are simply done with slight modifications to the method used to take in parameter values at the command line.

- Interactivity

  The animation API provided by [129] was designed to take keyboard input so that it could quit when, e.g., the `Esc` button was pressed. However, this effectively allowed for keyboard interactivity. Though cumbersome, it is straightforward to tie key values to different actions. Thus, e.g., the space bar can be programmed to cause the simulation to pause, "l" can be programmed to list the current parameters, and individual letters can be built into a dynamically allocated array to allow for various functionalities, such as for the parameter values to be changed on the fly.

- Output

> In order to start a simulation from the final state of a previous simulation, we obviously need to have first output that state. Additionally, we may need data for later analysis and visualization. For example, we may wish to save images or movies for later presentations or publication. By allowing for keyboard input and incorporating command-line programs such as `ffmpeg`, we are able to save such data at a keystroke. Full data for 2 arrays of 512×512 elements over any reasonable period of time takes a significant amount of storage space. Thus, we have also allowed for "virtual electrodes" to be positioned from an input file read at the command line, defining locations at which we can take data over longer time periods while keeping the file sizes low. In the near future, this feature will be more interactive, allowing for dynamic adjustment of these positions. Finally, we can also dynamically save the state of the system. The parameters and final state of the network are saved in a user-named folder. Two shell scripts are added to the folder: one to start the simulation with the given set of parameters and random initial conditions, and one that instead uses the final state of the saved simulation to begin. The user also has the option to write notes to a NOTES.txt file that opens in `EMACS`.

By providing the input and output features we've made available in our code, the exploration of network dynamics for different sets of parameters is greatly facilitated. In particular, we've written Python code that takes in the results of a two-parameter sweep and creates an interactive bifurcation diagram. The variance of the network, averaged over spatial elements, is shown, and clicking on any point on the plot runs the simulation, beginning from the last state. Thus, by only saving the final state of the network and the computed variance, we can rapidly "point-and-click" to examine the dynamics allowed by the network after transient time periods. We will also provide a version of this code online.

We expect to implement more features in our code in the future, including additional interactivity and a more user-friendly GUI. We also note that these methods can be easily adapted for other, similar purposes. For example, spiking networks can be accommodated in a stratightforward manner by changing the particular functional forms used and the connectivity employed. That is, instead of convolutions, the couplings would be defined by weight matrices; however, just as for convolutions, these can be computed in a separate function that is then fed into the integration step as we do here. CUDA has linear algebra libraries to facilitate such computations; indeed, these are employed in the cuFFT library. In the future, we may work to extend our code to such areas of application.

BIBLIOGRAPHY

1. Lapicque L. Quantitative investigations of electrical nerve excitation treated as polarization. 1907. Biological Cybernetics. 2007;97(5-6):341.

2. Brunel N, Van Rossum MC. Quantitative investigations of electrical nerve excitation treated as polarization. Biological Cybernetics. 2007;97(5-6):341–349.

3. Brunel N, Van Rossum MC. Lapicque's 1907 paper: from frogs to integrate-and-fire. Biological Cybernetics. 2007;97(5-6):337–339.

4. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology. 1952;117(4):500–544.

5. Beurle RL. Properties of a mass of cells capable of regenerating pulses. Phil Trans R Soc Lond B. 1956;240(669):55–94.

6. Wilson HR, Cowan JD. Excitatory and inhibitory interactions in localized populations of model neurons. Biophysical journal. 1972;12(1):1–24.

7. Wilson HR, Cowan JD. A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue. Kybernetik. 1973;13(2):55–80.

8. Jirsa VK, Haken H. Field theory of electromagnetic brain activity. Physical Review Letters. 1996;77(5):960.

9. Jirsa VK, Haken H. A derivation of a macroscopic field theory of the brain from the quasi-microscopic neural dynamics. Physica D: Nonlinear Phenomena. 1997;99(4):503–526.

10. Coombes S, Byrne Á. Next generation neural mass models. In: Nonlinear Dynamics in Computational Neuroscience. Springer; 2019. p. 1–16.

11. Pinto DJ, Patrick SL, Huang WC, Connors BW. Initiation, propagation, and termination of epileptiform activity in rodent neocortex in vitro involve distinct mechanisms. Journal of Neuroscience. 2005;25(36):8131–8140.

12. Harris JD, Ermentrout B. Traveling waves in a spatially-distributed Wilson–Cowan model of cortex: From fronts to pulses. Physica D: Nonlinear Phenomena. 2018;369:30–46.

13. Wang Y, Goodfellow M, Taylor PN, Baier G. Phase space approach for modeling of epileptic dynamics. Physical Review E. 2012;85(6):061918.

14. Kilpatrick ZP, Ermentrout B. Wandering bumps in stochastic neural fields. SIAM Journal on Applied Dynamical Systems. 2013;12(1):61–94.

15. Horn D, Sagi D, Usher M. Segmentation, binding, and illusory conjunctions. Neural Computation. 1991;3(4):510–525.

16. Ermentrout GB, Cowan JD. A mathematical theory of visual hallucination patterns. Biological Cybernetics. 1979;34(3):137–150.

17. Rule M, Stoffregen M, Ermentrout B. A model for the origin and properties of flicker-induced geometric phosphenes. PLoS Computational Biology. 2011;7(9):e1002158.

18. Bressloff PC, Cowan JD, Golubitsky M, Thomas PJ, Wiener MC. Geometric visual hallucinations, Euclidean symmetry and the functional architecture of striate cortex. Philosophical Transactions of the Royal Society of London B: Biological Sciences. 2001;356(1407):299–330.

19. Kandel ER, Schwartz JH, Jessell TM, of Biochemistry D, Jessell MBT, Siegelbaum S, et al. Principles of neural science. McGraw-hill New York; 2000.

20. Baddeley A. Working memory: looking back and looking forward. Nature Reviews Neuroscience. 2003;4(10):829.

21. Miller G, Galanter E, Pribram K. Plans and the structure of behavior. New York, NY, US: Henry Holt and Co; 1960.

22. Baddeley A, Hitch G. Working memory, GH. Bower (Ed.), The Psychology of Learning and Motivation, vol. 8; 1974.

23. Fuster JM, Alexander GE. Neuron activity related to short-term memory. Science. 1971;173(3997):652–654.

24. Funahashi S, Bruce CJ, Goldman-Rakic PS. Mnemonic coding of visual space in the monkey's dorsolateral prefrontal cortex. Journal of Neurophysiology. 1989;61(2):331–349.

25. Shafi M, Zhou Y, Quintana J, Chow C, Fuster J, Bodner M. Variability in neuronal activity in primate cortex during working memory tasks. Neuroscience. 2007;146(3):1082–1108.

26. Pesaran B, Pezaris JS, Sahani M, Mitra PP, Andersen RA. Temporal structure in neuronal activity during working memory in macaque parietal cortex. Nature Neuroscience. 2002;5(8):805.

27. Kornblith S, Quiroga RQ, Koch C, Fried I, Mormann F. Persistent single-neuron activity during working memory in the human medial temporal lobe. Current Biology. 2017;27(7):1026–1032.

28. Wang XJ. Synaptic basis of cortical persistent activity: the importance of NMDA receptors to working memory. Journal of Neuroscience. 1999;19(21):9587–9603.

29. Lisman JE, Fellous JM, Wang XJ. A role for NMDA-receptor channels in working memory. Nature Neuroscience. 1998;1(4):273.

30. Adler C, Goldberg T, Malhotra A, Pickar D, Breier A. Effects of ketamine on thought disorder, working memory, and semantic memory in healthy volunteers. Biological Psychiatry. 1998;43(11):811–816.

31. Compte A, Brunel N, Goldman-Rakic PS, Wang XJ. Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. Cerebral Cortex. 2000;10(9):910–923.

32. Brody CD, Hernández A, Zainos A, Romo R. Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. Cerebral Cortex. 2003;13(11):1196–1207.

33. Howard MW, Rizzuto DS, Caplan JB, Madsen JR, Lisman J, Aschenbrenner-Scheibe R, Schulze-Bonhave A, Kahana MJ. Gamma oscillations correlate with working memory load in humans. Cerebral Cortex. 2003;13(12):1369–1374.

34. Sokolov A, Lutzenberger W, Pavlova M, Preissl H, Braun C, Birbaumer N. Gamma-band MEG activity to coherent motion depends on task-driven attention. Neuroreport. 1999;10(10):1997–2000.

35. Jensen O, Gelfand J, Kounios J, Lisman JE. Oscillations in the alpha band (9–12 Hz) increase with memory load during retention in a short-term memory task. Cerebral Cortex. 2002;12(8):877–882.

36. Lisman JE, Idiart MA. Storage of 7+/-2 short-term memories in oscillatory subcycles. Science. 1995;267(5203):1512–1515.

37. Feldman J. The neural binding problem (s). Cognitive Neurodynamics. 2013;7(1):1–11.

38. Malsburg Cvd. The correlation theory of brain function. In: Models of neural networks. Springer; 1994. p. 95–119.

39. Gray CM, König P, Engel AK, Singer W. Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. Nature. 1989;338(6213):334.

40. Gray CM, Singer W. Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex. Proceedings of the National Academy of Sciences. 1989;86(5):1698–1702.

41. Bickford RG, Daly D, Keith HM. Convulsive effects of light stimulation in children. AMA American Journal of Diseases of Children. 1953;86(2):170–183.

42. Wilkins AJ. Visual stress. Oxford University Press; 1995.

43. Stefansson SB, Darby CE, Wilkins AJ, Binnie CD, Marlton AP, Smith AT, Stockley, AV. Television epilepsy and pattern sensitivity. British Medical Journal. 1977;2(6079):88–90.

44. Wilkins A, Darby C, Binnie C. Neurophysiological aspects of pattern-sensitive epilepsy. Brain. 1979;102(1):1–25.

45. Schade OH. Optical and photoelectric analog of the eye. Journal of the Optical Society of America. 1956;46(9):721–739.

46. Campbell FW, Robson J. Application of Fourier analysis to the visibility of gratings. The Journal of Physiology. 1968;197(3):551–566.

47. Fernandez D, Wilkins AJ. Uncomfortable images in art and nature. Perception. 2008;37(7):1098–1113.

48. Adjamian P, Holliday IE, Barnes GR, Hillebrand A, Hadjipapas A, Singh KD. Induced visual illusions and gamma oscillations in human primary visual cortex. European Journal of Neuroscience. 2004;20(2):587–592.

49. Moehlis J, Josic K, Shea-Brown ET. Periodic orbit. Scholarpedia. 2006;1(7):1358.

50. Guckenheimer J, Holmes P. Nonlinear oscillations, dynamical systems, and bifurcations of vector fields. vol. 42. Springer Science & Business Media; 2013.

51. Kuznetsov YA. Elements of applied bifurcation theory. vol. 112. Springer Science & Business Media; 2013.

52. Pina JE, Bodner M, Ermentrout B. Oscillations in working memory and neural binding: A mechanism for multiple memories and their interactions. PLoS Comput Biol. 2018; 14(11):e1006517.

53. Fuster JM, Bodner M, Kroger JK. Cross-modal and cross-temporal association in neurons of frontal cortex. Nature. 2000; 405(6784):347.

54. Ku Y, Bodner M, Zhou YD. Prefrontal cortex and sensory cortices during working memory: Quantity and quality. Neuroscience Bulletin. 2015; 31(2):175–182.

55. Cowan N. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. Behavioral and Brain Sciences. 2001; 24(1):87–114.

56. Luck SJ, Vogel EK. The capacity of visual working memory for features and conjunctions. Nature. 1997; 390(6657):279.

57. Vogel EK, Woodman GF, Luck SJ. Storage of features, conjunctions, and objects in visual working memory. Journal of Experimental Psychology: Human Perception and Performance. 2001; 27(1):92.

58. Todd JJ, Marois R. Capacity limit of visual short-term memory in human posterior parietal cortex. Nature. 2004; 428(6984):751.

59. Singer W, Gray CM. Visual feature integration and the temporal correlation hypothesis. Annual Review of Neuroscience. 1995; 18(1):555–586.

60. Singer W. Neuronal synchrony: A versatile code for the definition of relations? Neuron. 1999; 24(1):49–65.

61. Engel AK, König P, Singer W. Direct physiological evidence for scene segmentation by temporal coding. Proceedings of the National Academy of Sciences. 1991; 88(20):9136–9140.

62. Kessler Y, Meiran N. All updateable objects in working memory are updated whenever any of them are modified: Evidence from the memory updating paradigm. Journal of Experimental Psychology: Learning, Memory, and Cognition. 2006; 32(3):570.

63. Fries P, Reynolds JH, Rorie AE, Desimone R. Modulation of oscillatory neuronal synchronization by selective visual attention. Science. 2001; 291(5508):1560–1563.

64. Cowan N, Blume CL, Saults JS. Attention to attributes and objects in working memory. Journal of Experimental Psychology: Learning, Memory, and Cognition. 2013; 39(3):731.

65. Hardman KO, Cowan N. Remembering complex objects in visual working memory: Do capacity limits restrict objects or features? Journal of Experimental Psychology: Learning, Memory, and Cognition. 2015; 41(2):325.

66. Klimesch W. EEG alpha and theta oscillations reflect cognitive and memory performance: A review and analysis. Brain Research Reviews. 1999; 29(2-3):169–195.

67. Başar E, Başar-Eroglu C, Karakaş S, Schürmann M. Gamma, alpha, delta, and theta oscillations govern cognitive processes. International Journal of Psychophysiology. 2001; 39(2-3):241–248.

68. Roux F, Uhlhaas PJ. Working memory and neural oscillations: Alpha–gamma versus theta–gamma codes for distinct WM information? Trends in Cognitive Sciences. 2014; 18(1):16–25.

69. Verduzco-Flores S, Ermentrout B, Bodner M. From working memory to epilepsy: Dynamics of facilitation and inhibition in a cortical network. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2009; 19(1):015115.

70. Lisman J. Working memory: The importance of theta and gamma oscillations. Current Biology. 2010; 20(11):R490–R492.

71. Krystal JH, Abi-Saab W, Perry E, D'Souza DC, Liu N, Gueorguieva R, McDougall L, Hunsberger T, Belger A, Levine L, Breier A Preliminary evidence of attenuation of the disruptive effects of the NMDA glutamate receptor antagonist, ketamine, on working memory by pretreatment with the group II metabotropic glutamate receptor agonist, LY354740, in healthy human subjects. Psychopharmacology. 2005; 179(1):303–309.

72. Ermentrout GB, Terman DH. Mathematical Foundations of Neuroscience. Vol. 35. Springer Science & Business Media; 2010.

73. Pinto DJ, Brumberg JC, Simons DJ, Ermentrout GB, Traub R. A quantitative population model of whisker barrels: re-examining the Wilson-Cowan equations. Journal of Computational Neuroscience. 1996; 3(3):247–264.

74. Curtu R, Ermentrout GB. Oscillations in a refractory neural net. Journal of Mathematical Biology. 2000; 43(1):81–100.

75. Ermentrout B. Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students. Vol. 14. SIAM; 2002.

76. Senior TJ, Huxter JR, Allen K, O'Neill J, Csicsvari J. Gamma oscillatory firing reveals distinct populations of pyramidal cells in the CA1 region of the hippocampus. Journal of Neuroscience. 2008; 28(9):2274–2286.

77. Engel AK, Singer W. Temporal binding and the neural correlates of sensory awareness. Trends in Cognitive Sciences. 2001; 5(1):16–25.

78. Lisman JE, Jensen O. The theta-gamma neural code. Neuron. 2013; 77(6):1002–1016.

79. Raffone A, Wolters G. A cortical mechanism for binding in visual working memory. Journal of Cognitive Neuroscience. 2001; 13(6):766–785.

80. Horn D, Opher I. Temporal segmentation in a neural dynamic system. Neural Computation. 1996; 8(2):373–389.

81. Oberauer K. Interference between storage and processing in working memory: Feature overwriting, not similarity-based competition. Memory & Cognition. 2009; 37(3):346–357.

82. Shastri L, Ajjanagadde V. From simple associations to systematic reasoning: A connectionist representation of rules, variables and dynamic bindings using temporal synchrony. Behavioral and Brain Sciences. 1993; 16(3):417–451.

83. Wendelken C, Shastri L. Multiple instantiation and rule mediation in SHRUTI. Connection Science. 2004; 16(3):211–217.

84. Tallon-Baudry C, Bertrand O, Peronnet F, Pernier J. Induced γ-band activity during the delay of a visual short-term memory task in humans. Journal of Neuroscience. 1998; 18(11):4244–4254.

85. Tallon C, Bertrand O, Bouchet P, Pernier J. Gamma-range activity evoked by coherent visual stimuli in humans. European Journal of Neuroscience. 1995; 7(6):1285–1291.

86. Roux F, Wibral M, Mohr HM, Singer W, Uhlhaas PJ. Gamma-band activity in human prefrontal cortex codes for the number of relevant items maintained in working memory. Journal of Neuroscience. 2012; 32(36):12411–12420.

87. Kaiser J, Rahm B, Lutzenberger W. Temporal dynamics of stimulus-specific gamma-band activity components during auditory short-term memory. Neuroimage. 2009; 44(1):257–264.

88. Lutzenberger W, Ripper B, Busse L, Birbaumer N, Kaiser J. Dynamics of gamma-band activity during an audiospatial working memory task in humans. Journal of Neuroscience. 2002; 22(13):5630–5638.

89. Jensen O, Bonnefond M, VanRullen R. An oscillatory mechanism for prioritizing salient unattended stimuli. Trends in Cognitive Sciences. 2012; 16(4):200–206.

90. Voytek B, Canolty RT, Shestyuk A, Crone NE, Parvizi J, Knight RT. Shifts in gamma phase–amplitude coupling frequency from theta to alpha over posterior cortex during visual tasks. Frontiers in Human Neuroscience. 2010; 4.

91. Medendorp WP, Kramer GF, Jensen O, Oostenveld R, Schoffelen JM, Fries P. Oscillatory activity in human parietal and occipital cortex shows hemispheric lateralization and memory effects in a delayed double-step saccade task. Cerebral Cortex. 2006; 17(10):2364–2374.

92. Worden MS, Foxe JJ, Wang N, Simpson GV. Anticipatory biasing of visuospatial attention indexed by retinotopically specific-band electroencephalography increases over occipital cortex. Journal of Neuroscience. 2000; 20(RC63):1–6.

93. Klimesch W, Sauseng P, Hanslmayr S. EEG alpha oscillations: The inhibition–timing hypothesis. Brain Research Reviews. 2007; 53(1):63–88.

94. Klimesch W. Alpha-band oscillations, attention, and controlled access to stored information. Trends in Cognitive Sciences. 2012; 16(12):606–617.

95. Wang D, Buhmann J, von der Malsburg C. Pattern segmentation in associative memory. Neural Computation. 1990; 2(1):94–106.

96. Horn D, Usher M. Parallel activation of memories in an oscillatory neural network. Neural Computation. 1991; 3(1):31–43.

97. Winder RK, Reggia JA, Weems SA, Bunting MF. An oscillatory Hebbian network model of short-term memory. Neural Computation. 2009; 21(3):741–761.

98. König P, Schillen TB. Stimulus-dependent assembly formation of oscillatory responses: I. Synchronization. Neural Computation. 1991; 3(2):155–166.

99. Sompolinsky H, Golomb D, Kleinfeld D. Global processing of visual stimuli in a neural network of coupled oscillators. Proceedings of the National Academy of Sciences. 1990; 87(18):7200–7204.

100. Raffone A, van Leeuwen C. Dynamic synchronization and chaos in an associative neural network with multiple active memories. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2003; 13(3):1090–1104.

101. Terman D, Wang D. Global competition and local cooperation in a network of neural oscillators. Physica D: Nonlinear Phenomena. 1995; 81(1-2):148–176.

102. Meier M, Haschke R, Ritter HJ. Perceptual grouping through competition in coupled oscillator networks. Neurocomputing. 2014; 141:76–83.

103. Yu G, Slotine JJ. Visual grouping by neural oscillator networks. IEEE Transactions on Neural Networks. 2009; 20(12):1871–1884.

104. Horn D, Usher M. Excitatory–inhibitory networks with dynamical thresholds. International Journal of Neural Systems. 1990; 1(03):249–257.

105. Horn D, Opher I. The importance of noise for segmentation and binding in dynamical neural systems. International Journal of Neural Systems. 1996; 7(04):529–535.

106. Breve FA, Zhao L, Quiles MG, Macau EE. Chaotic phase synchronization and desynchronization in an oscillator network for object selection. Neural Networks. 2009; 22(5-6):728–737.

107. Sakai K, Rowe JB, Passingham RE. Active maintenance in prefrontal area 46 creates distractor-resistant memory. Nature Neuroscience. 2002; 5(5):479.

108. Oberauer K. Declarative and procedural working memory: Common principles, common capacity limits? Psychologica Belgica. 2010; 50(3-4):3–4.

109. Garavan H. Serial attention within working memory. Memory & Cognition. 1998; 26(2):263–276.

110. Oberauer K. Selective attention to elements in working memory. Experimental Psychology. 2003; 50(4):257.

111. Kessler Y, Meiran N. Two dissociable updating processes in working memory. Journal of Experimental Psychology: Learning, Memory, and Cognition. 2008; 34(6):1339.

112. Pertzov Y, Husain M. The privileged role of location in visual working memory. Attention, Perception, & Psychophysics. 2014; 76(7):1914–1924.

113. Harrison WJ, Bays PM. Visual working memory is independent of the cortical spacing between memoranda. Journal of Neuroscience. 2018; p. 2645–17.

114. Doughty J, Garner W. Pitch characteristics of short tones. II. Pitch as a function of tonal duration. Journal of Experimental Psychology. 1948; 38(4):478.

115. Ekman G, Berglund B, Berglund U. Loudness as a function of the duration of auditory stimulation. Scandinavian Journal of Psychology. 1966; 7(1):201–208.

116. Börgers C, Kopell NJ. Gamma oscillations and stimulus selection. Neural Computation. 2008; 20(2):383–414.

117. Borisyuk GN, Borisyuk RM, Khibnik AI, Roose D. Dynamics and bifurcations of two coupled neural oscillators with different connection types. Bulletin of Mathematical Biology. 1995; 57(6):809–840.

118. Ermentrout B. Reduction of conductance-based models with slow synapses to neural nets. Neural Computation. 1994; 6(4):679–695.

119. Golomb D, Amitai Y. Propagating neuronal discharges in neocortical slices: Computational and experimental study. Journal of Neurophysiology. 1997; 78(3):1199–1211.

120. Dayan P, Abbott LF. Theoretical Neuroscience. Vol. 806. Cambridge, MA: MIT Press; 2001.

121. Kim SS, Rouault H, Druckmann S, Jayaraman V. Ring attractor dynamics in the Drosophila central brain. Science. 2017;356(6340):849–853.

122. Billock VA, Tsou BH. Neural interactions between flicker-induced self-organized visual hallucinations and physical stimuli. Proceedings of the National Academy of Sciences. 2007;104(20):8490–8495.

123. Radhakrishnan K, Klass DW. Half a century of visual pattern-sensitive epilepsy. In: Mayo Clinic Proceedings. vol. 79. Elsevier; 2004. p. 269–270.

124. Wilkins A, Nimmo-Smith I, Tait A, McManus C, Sala SD, Tilley A, Arnold KM, Barrie M, Scott S. A neurological basis for visual discomfort. Brain. 1984;107(4):989–1017.

125. Dayan P, Abbott LF, Abbott L. Theoretical neuroscience: computational and mathematical modeling of neural systems. MIT press Cambridge, MA; 2001.

126. Lund JS, Angelucci A, Bressloff PC. Anatomical substrates for functional columns in macaque monkey primary visual cortex. Cerebral Cortex. 2003;13(1):15–24.

127. Morimoto T, Hayakawa T, Sugie H, Awaya Y, Fukuyama Y. Epileptic seizures precipitated by constant light, movement in daily life, and hot water immersion. Epilepsia. 1985;26(3):237–242.

128. Logg A, Mardal KA, Wells G. Automated solution of differential equations by the finite element method: The FEniCS book. vol. 84. Springer Science & Business Media; 2012.

129. Sanders J, Kandrot E. CUDA by example: an introduction to general-purpose GPU programming. Addison-Wesley Professional; 2010.

130. Lathi BP. Linear systems and signals. vol. 2. Oxford University Press New York; 2005.