An Analytical Framework for Multi-Area Balanced Networks

Josue Casco-Rodriguez JC135@RICE.EDU
Rice University, Houston, TX
Mitra Javadzadeh JAVADZADEH@CSHL.EDU
Cold Spring Harbor Laboratory, Cold Spring Harbor, NY

Editors: List of editors' names

Abstract

A hallmark of neocortical architecture is recurrent connectivity both within and between local sub-networks (cortical areas). Within a cortical area, excitation-inhibition balance (balanced positive and negative connections) shapes neural activity dynamics (1; 2; 3; 4), while reciprocal connections between areas are excitatory. How this multi-area structure shapes neural dynamics remains largely unknown (but see (5; 6)). We present an analytical framework for balanced multi-area networks, revealing key features of cortical computation; we find that local connectivity within an area determines its responses to inputs received locally (extra-cortical), but not to inputs relayed from other cortical areas. Local responses to these relayed inputs are instead primarily driven by long-range inter-area connections. Moreover, we find that the asymmetry of inter-area connections (feedforward vs feedback strength) can modulate the joint dynamics across areas and implement a tradeoff between regimes that promote similarity or divergence of activity across areas.

Keywords: excitation–inhibition balance, recurrent networks, cortical dynamics, multi-area networks, asymmetry, amplification

1. Introduction

The mammalian neocortex is composed of many coupled sub-networks (cortical areas) that are heavily interconnected. While there are clear signatures of hierarchy in this multi-area network—such as feedforward projections from lower to higher areas, which has led to them being traditionally modeled as deep feedforward architectures—there also exist extensive lateral connections that recur between areas at similar hierarchical levels (7). These recurrent pathways challenge the notion of purely feedforward processing, and the question remains: how do computations unfold within the recurrent, multi-area architecture of the cortex?

Within a cortical area, local excitation-inhibition (E/I) balance (balanced positive and negative connections) tightly constrains neural activity dynamics, placing the circuit in a regime of balanced amplification (2). In this regime, activity is described by an effective feedforward motif in which E/I unbalanced modes drive E/I balanced modes. Cortical inhibition, however, is predominantly local, while E neurons project long-range and to other areas. These reciprocal excitatory inter-area connections produce characteristic dynamical motifs, notably, they give rise to slow dynamical modes that promote similar activity patterns across areas (6). Here, we build on these findings and introduce a general analytical framework for understanding computation in multi-area, cortical-like networks. This framework smoothly interpolates between purely feedforward and fully recurrent architectures and reveals key principles governing the dynamics of distributed cortical computation.

2. Background

We examine minimal linear recurrent networks with E/I connectivity. At time t, network activity $\boldsymbol{r}(t)$ evolves according to recurrent weights $\boldsymbol{W}_{ei}$ and external input $\boldsymbol{s}(t)$: $\tau\dot{\boldsymbol{r}}(t) = -\boldsymbol{r}(t) + \boldsymbol{W}_{ei}\boldsymbol{r}(t) + \boldsymbol{s}(t)$, where $\boldsymbol{W}_{ei} = \begin{bmatrix} e & -i \\ e & -i \end{bmatrix}$, and $\boldsymbol{r} = \begin{bmatrix} r_e \\ r_i \end{bmatrix}$. E/I connectivity matrices of the form $\boldsymbol{W}_{ei}$ do not lend themselves well to diagonalization (they have non-orthogonal eigenvectors and incur non-normal dynamics). Our objective is to find an orthonormal basis $\boldsymbol{Q}$ which transforms $\boldsymbol{W}_{ei}$ into an upper triangular matrix (Schur form; $\widehat{\boldsymbol{W}}_{ei} = \boldsymbol{Q}^T\boldsymbol{W}_{ei}\boldsymbol{Q}$), such that the connectivity can be interpreted as an effective feedforward motif, with more interpretable information flow (see appendix A.1).

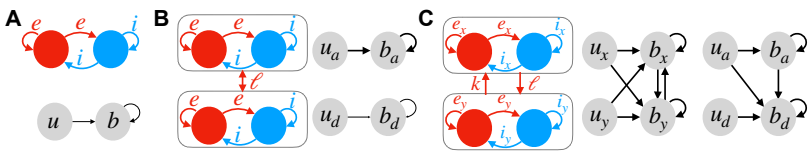

Figure 1: *(A) Recurrently connected E-I network (top) and its feedforward equivalence from unbalanced (u) to balanced (b) modes (bottom). (B) Symmetric two-area E-I networks (left) and its feedforward equivalence (right), showing independent $u \to b$ amplification in agree (a) and disagree (d) subnetworks. (C) The general two-area network (left) and its equivalent circuits in the quasi-Schur (middle) and Schur (right) bases.*

2.1. Balanced amplification

Previous work (2) has described a straightforward interpretation of simple E-I networks in a Schur basis of $\boldsymbol{W}_{ei}$, where an *E/I unbalanced* mode, $\boldsymbol{u}^T = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & -1 \end{bmatrix}$, feeds an *E/I balanced* mode, $\boldsymbol{b}^T = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \end{bmatrix}$ (fig. 1A, appendix A.1): $\widehat{\boldsymbol{W}}_{ei} = \begin{bmatrix} e-i & e+i \\ 0 & 0 \end{bmatrix}$, $\boldsymbol{Q} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{b} & \boldsymbol{u} \end{bmatrix}$

2.2. Consensus building between areas

This decomposition was later extended to the case of two identical E/I networks (X and Y) coupled via excitatory connections (6): $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{ei} & \boldsymbol{L} \\ \boldsymbol{L} & \boldsymbol{W}_{ei} \end{bmatrix}$, $\boldsymbol{r} = \begin{bmatrix} \boldsymbol{r}_x \\ \boldsymbol{r}_y \end{bmatrix}$, where $\boldsymbol{L} = \begin{bmatrix} \ell & 0 \\ \ell & 0 \end{bmatrix}$, and $\boldsymbol{r}_x^T = \begin{bmatrix} \boldsymbol{r}_{x,e}^T & \boldsymbol{r}_{x,i}^T \end{bmatrix}$, $\boldsymbol{r}_y^T = \begin{bmatrix} \boldsymbol{r}_{y,e}^T & \boldsymbol{r}_{y,i}^T \end{bmatrix}$ are local activity within the areas (fig. 1B). They showed that these networks exhibit two separate feedforward motifs: (a) an *agree* subnetwork, characterized by similar patterns of activity across the areas, where, the *unbalanced agree* mode $\boldsymbol{u}_{agree}^T = \begin{bmatrix} \boldsymbol{u}_x^T & \boldsymbol{u}_y^T \end{bmatrix}$ feeds the *balanced agree* mode $\boldsymbol{b}_{agree}^T = \begin{bmatrix} \boldsymbol{b}_x^T & \boldsymbol{b}_y^T \end{bmatrix}$, and (b) a *disagree* subnetwork, characterized by opposite patterns across areas, where *unbalanced disagree* mode $\boldsymbol{u}_{disagree}^T = \begin{bmatrix} \boldsymbol{u}_x^T & -\boldsymbol{u}_y^T \end{bmatrix}$ feeds the *balanced disagree* mode $\boldsymbol{b}_{disagree}^T = \begin{bmatrix} \boldsymbol{b}_x^T & -\boldsymbol{b}_y^T \end{bmatrix}$:

$$\widehat{\boldsymbol{W}} = \begin{bmatrix} \widehat{\boldsymbol{W}}_{ei} + \boldsymbol{L}^T & \boldsymbol{0} \\ \boldsymbol{0} & \widehat{\boldsymbol{W}}_{ei} - \boldsymbol{L}^T \end{bmatrix}, \quad \boldsymbol{Q} = \frac{1}{\sqrt{2}}\begin{bmatrix} \boldsymbol{b}_x & \boldsymbol{u}_x & \boldsymbol{b}_x & \boldsymbol{u}_x \\ \boldsymbol{b}_y & \boldsymbol{u}_y & -\boldsymbol{b}_y & -\boldsymbol{u}_y \end{bmatrix} \tag{1}$$

Long-range connections contribute positively to the connectivity of the agree subnetwork ($\widehat{\boldsymbol{W}}_{ei} + \boldsymbol{L}^T$), increasing both feedforward strengths and time constants, while weakening the

weights of the disagree subnetwork ($\widehat{\boldsymbol{W}}_{ei} - \boldsymbol{L}^T$). As a result, the agree subnetwork will dominate, leading to activity converging dynamically towards similar cross-area patterns.

3. Methods

Here, we generalize this framework to explain activity dynamics of two asymmetrically coupled, non-identical E/I networks. For simplicity, we assume each network is perfectly balanced, $e = i$. We consider two networks X, Y with different weights and asymmetric excitatory connections: X feeds Y with strength ℓ, and vice versa with strength k, $\boldsymbol{W} = \begin{bmatrix} \boldsymbol{W}_{x,ei} & \boldsymbol{K} \\ \boldsymbol{L} & \boldsymbol{W}_{y,ei} \end{bmatrix}$, where $\boldsymbol{K} = \begin{bmatrix} k & 0 \\ k & 0 \end{bmatrix}$ (fig. 1C left). We introduce two orthonormal bases that can provide insights into these networks.

The first is a quasi-Schur *single-area* basis consisting of E-I balanced/unbalanced modes in each area. In this basis, the balanced modes of each area, $\boldsymbol{b}_x, \boldsymbol{b}_y$, are fed by each other and the unbalanced modes of each area, $\boldsymbol{u}_x, \boldsymbol{u}_y$ (fig. 1C middle):

$$\widehat{\boldsymbol{W}}_{single} = \begin{bmatrix} \boldsymbol{W}_e - \boldsymbol{W}_i & \boldsymbol{W}_e + \boldsymbol{W}_i \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{Q}_{single} = \begin{bmatrix} \boldsymbol{b}_x & \boldsymbol{0} & \boldsymbol{u}_x & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{b}_y & \boldsymbol{0} & \boldsymbol{u}_y \end{bmatrix} \tag{2}$$

where $\boldsymbol{W}_e = \begin{bmatrix} e_x & k \\ \ell & e_y \end{bmatrix}$ and $\boldsymbol{W}_i = \begin{bmatrix} i_x & 0 \\ 0 & i_y \end{bmatrix}$. As $\widehat{\boldsymbol{W}}_{single}$ is still recurrent ($\boldsymbol{b}_x$ and $\boldsymbol{b}_y$ feed each other), we also provide a Schur *multi-area* basis $\boldsymbol{Q}_{multi}$ to establish a feedforward equivalence for the whole system, where the basis describes patterns of generalized agreements or disagreements across the two areas (fig. 1C right). In this basis, the resulting $\widehat{\boldsymbol{W}}$ is $\widehat{\boldsymbol{W}}_{multi}$:

$$\widehat{\boldsymbol{W}}_{multi} = \begin{bmatrix} -\sqrt{k\ell} & \frac{\alpha N_1}{N_2} & k-\ell & \frac{\beta}{N_1 N_2} \\ 0 & 0 & \sqrt{k\ell} & \frac{N_2}{N_1} \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \boldsymbol{Q}_{multi} = \begin{bmatrix} \frac{\sqrt{k}}{N_1}\boldsymbol{b}_x & \frac{\sqrt{k}F}{N_2}\boldsymbol{u}_x & \frac{\sqrt{\ell}}{N_1}\boldsymbol{b}_x & \frac{\sqrt{\ell}G}{N_2}\boldsymbol{u}_x \\ -\frac{\sqrt{\ell}}{N_1}\boldsymbol{b}_y & -\frac{\sqrt{\ell}G}{N_2}\boldsymbol{u}_y & \frac{\sqrt{k}}{N_1}\boldsymbol{b}_y & \frac{\sqrt{k}F}{N_2}\boldsymbol{u}_y \end{bmatrix} \tag{3}$$

where $N_1 = \sqrt{k+\ell}$, $N_2 = \sqrt{kF^2 + \ell G^2}$, $F = y + \sqrt{k\ell}$, $G = e_x + \sqrt{k\ell}$, $\alpha = (e_x e_y - k\ell)$, $\beta = \sqrt{k\ell}(e_x^2 + k^2 - e_y^2 - \ell^2) + (k-\ell)(e_x\ell + e_y k)$, and $\otimes$ is the Kronecker product. The columns of $\boldsymbol{Q}_{multi}$ are asymmetrically-weighted (generalized) agree/disagree modes; in order, generalized balanced disagree ($\boldsymbol{b}_{disagree}$), unbalanced disagree ($\boldsymbol{u}_{disagree}$), balanced agree ($\boldsymbol{b}_{agree}$), and unbalanced agree ($\boldsymbol{u}_{agree}$). Decomposing the network in the above bases allow us to analytically characterize network dynamics as a function of model parameters.

4. Results

First, we asked how local computation within a cortical area is modulated by other interconnected areas (fig. 2A top). Our derivations show that, when area X receives input, its response is modulated by $k\ell$, the strength of connections between X and Y, but not by e_y, the connectivity within area Y ($\boldsymbol{b}_x = \frac{1}{1-k\ell}\boldsymbol{s}_{b,x} + \frac{2e_x + k\ell}{1-k\ell}\boldsymbol{s}_{u,x}$ at steady-state, see appendix D and fig. 2 red). This finding explains how, despite dense inter-area connectivity, specialized processing within an area is not corrupted by the recurrent dynamics of connected areas.

A signature of cortical architecture is the existence of parallel pathways: External inputs reach a cortical area both indirectly, via other cortical areas, and directly through extra-cortical routes (e.g. higher visual areas recieve input from both primary visual

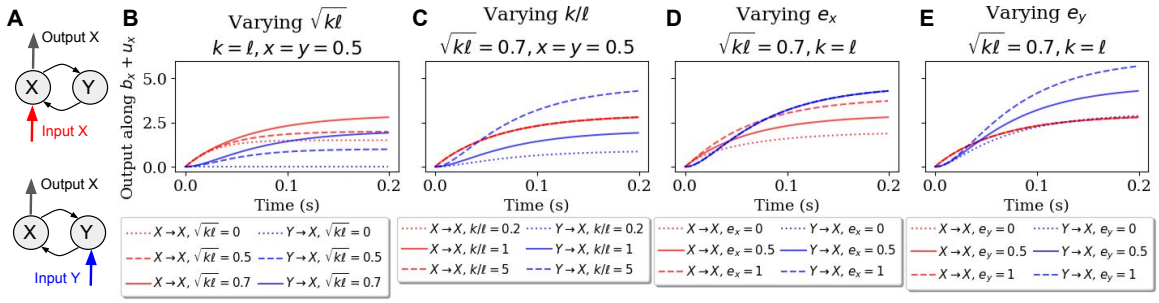

Figure 2: *Response of area X to inputs received either by itself (red; A top) or through area Y (blue; A bottom), for varying the product of long-range weights $\sqrt{k\ell}$ (B), asymmetry of long-range weights k/ℓ (C), X's local connectivity e_x (D), and Y's local connectivity e_y (E).*

cortex and directly from higher-order thalamus). This raises the question of whether cortically relayed and direct inputs are processed differently. To probe this, we analyzed how area X responds to input arriving through a connected area Y (fig. 2 A bottom). Our analytical framework shows that area X's response to Y's input depends on all connectivity strengths (e_y, k, l), *except*, surprisingly, X's own connectivity e_x ($\boldsymbol{b}_x = \frac{k}{1-k\ell}\boldsymbol{s}_{b,y} + \frac{k(1+2e_y)}{1-k\ell}\boldsymbol{s}_{u,y}$ at steady-state; appendix D; fig. 2 blue). Thus, recurrent dynamics in X are effectively invisible to cortically relayed inputs. Together, these results suggest that local, single-area recurrent computations can only be recruited by direct inputs (fig. 2, red), while cortically relayed inputs primarily engage long-range connections, bypassing local processing in intermediate areas (fig. 2, blue).

Finally, our analytical framework reveals how multi-area dynamics act on shared inputs received by both areas X and Y. Prior work (6) showed that in symmetric networks, the balanced agree and disagree modes form orthogonal eigenvectors of $\boldsymbol{W}$, with agree modes having the longest time constant, such that inputs along the agree mode (similar to both areas) remain aligned to it (fig. 3 left). Our general results extend this finding: Asymmetry in long-range connections rotates the slowest eigenvector, thereby rotating the flow field. As a result, inputs along the agree mode can evolve into disagree activity (opposite across areas; fig. 3 right). Our work therefore provides a general framework for understanding computation in multi-area networks, where inter-areal asymmetries control the trade off between regimes that promote similarity or divergence of activity across areas.

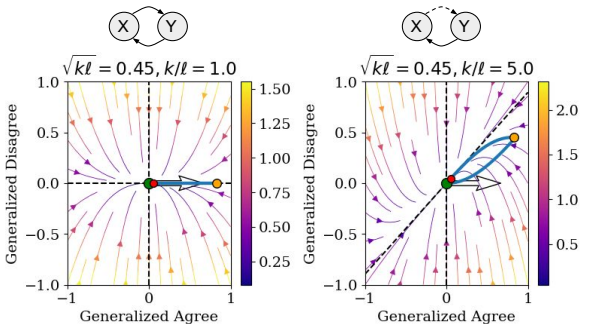

Figure 3: *Flow field of network dynamics projected on the generalized agree ($\boldsymbol{b}_{agree}$) and disagree ($\boldsymbol{b}_{disagree}$) modes for a network with symmetric long-range weights ($k = \ell$; left) and asymmetric weights ($k/\ell = 5$; right). Dashed lines are the eigenvectors of the dynamics. The color indicates the magnitude of velocity. Trajectories (blue) show the network response to an input to the generalized agree mode (white arrows). The red and yellow points mark the stimulus onset and offset accordingly.*

Future directions. The above results were derived in tightly balanced ($e = i$) networks with one E and one I unit per area. We are now extending this work to (1) examine how local E/I imbalance alters our findings, (2) generalize to networks of N neurons, where scalar connections strengths become (circulant) $N \times N$ matrices (see appendices B and C).

References

[1] Yashar Ahmadian, Daniel B Rubin, and Kenneth D Miller. Analysis of the stabilized supralinear network. *Neural computation*, 25(8):1994–2037, 2013.

[2] Brendan K Murphy and Kenneth D Miller. Balanced amplification: a new mechanism of selective amplification of neural activity patterns. *Neuron*, 61(4), 2009.

[3] Nicolas Brunel. Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of computational neuroscience*, 8(3):183–208, 2000.

[4] Carl Van Vreeswijk and Haim Sompolinsky. Chaos in neuronal networks with balanced excitatory and inhibitory activity. *Science*, 274(5293):1724–1726, 1996.

[5] Madhura R Joglekar, Jorge F Mejias, Guangyu Robert Yang, and Xiao-Jing Wang. Inter-areal balanced amplification enhances signal propagation in a large-scale circuit model of the primate cortex. *Neuron*, 98(1):222–234, 2018.

[6] Mitra Javadzadeh, Marine Schimel, Sonja B Hofer, Yashar Ahmadian, and Guillaume Hennequin. Dynamic consensus-building between neocortical areas via long-range connections. *bioRxiv*, 2024.

[7] Daniel J Felleman and David C Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral cortex (New York, NY: 1991)*, 1(1):1–47, 1991.

[8] Pat Vatiwutipong and Nattakorn Phewchean. Alternative way to derive the distribution of the multivariate Ornstein–Uhlenbeck process. *Advances in Difference Equations*, 2019(1), 2019.

Appendix A. Math Preliminaries

A.1. Schur decomposition

Throughout the text, we make use of *Schur decompositions* of matrices. These entail finding an orthonormal basis Q such that a matrix W becomes upper triangular: $W = Q\widehat{W}Q^*$, where $\widehat{W}$ is upper triangular. Unlike some other matrix decompositions, Schur decompositions are not unique. Previous work (2) has proposed Schur decomposition to analyze recurrent E/I networks, expressing them as feedforward via a basis Q. For example, an excitatory neuron excites itself and an inhibitory neuron with strength e, and the inhibitory neuron inhibits itself and the excitatory neuron with strength $-i$ ($W = \begin{bmatrix} e & -i \\ e & -i \end{bmatrix}$). Using a Schur basis, it becomes visible that there is an *unbalanced* mode of activity $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$ that feeds, with strength $e+i$, a *balanced* mode $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ that feeds itself with strength $e-i$ (fig. 1):

$$Q^*WQ = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} e & -i \\ e & -i \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} e-i & e+i \\ 0 & 0 \end{bmatrix} = \widehat{W}$$

Notably, the Schur decomposition of this example weight matrix is able to handle the case $e = i$, whereas eigendecomposition cannot (when $e = i$, W is not diagonalizable).

A.2. Upper triangularization via block-Hadamard matrices

We are interested in upper-triangularizing matrices with particular structures. To this end, the following properties of block-Hadamard matrices $\boldsymbol{H} = \begin{bmatrix} I & I \\ I & -I \end{bmatrix}$ are useful:

$$\begin{bmatrix} I & I \\ I & -I \end{bmatrix} \begin{bmatrix} A & B \\ A & B \end{bmatrix} \begin{bmatrix} I & I \\ I & -I \end{bmatrix} = \begin{bmatrix} 2(A+B) & 2(A-B) \\ 0 & 0 \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} I & I \\ I & -I \end{bmatrix} \begin{bmatrix} A & B \\ B & A \end{bmatrix} \begin{bmatrix} I & I \\ I & -I \end{bmatrix} = \begin{bmatrix} 2(A+B) & 0 \\ 0 & 2(A-B) \end{bmatrix} \tag{5}$$

A.3. Orthogonality of a 2×2 matrix of diagonal blocks

Later in the text, we will be decomposing matrices using orthogonal bases composed of diagonal blocks. Specifically, we are interested in matrices with four square diagonal blocks of the same shape, A, B, C, D, arranged into a 2×2 real block matrix. We seek the conditions on A, B, C, D that ensure the block matrix is orthogonal:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} A & C \\ B & D \end{bmatrix} = \begin{bmatrix} A & C \\ B & D \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$
$$\begin{bmatrix} A^2+B^2 & AC+BD \\ AC+BD & C^2+D^2 \end{bmatrix} = \begin{bmatrix} A^2+C^2 & AB+CD \\ AB+CD & B^2+D^2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \tag{6}$$

eq. (6) must be satisfied to ensure orthogonality. Inspecting its diagonal reveals the conditions $A^2 = D^2$ and $B^2 = C^2 = I - A^2$. To satisfy the assumption that A, B, C, D

are real, $I - A^2$ must be nonnegative: A must have a spectral radius no greater than 1. Assuming A is given and satisfies this constraint, our matrix of diagonal blocks must take this form:

$$\begin{bmatrix} A & b(I-A^2)^{1/2} \\ c(I-A^2)^{1/2} & dA \end{bmatrix},$$

where $b, c, d \in \{-1, 1\}$. From inspection of the anti-diagonal of Equation (6), we can deduce the constraints on b, c, d assuming that A and $I - A^2$ are invertible (if either A or $I - A^2$ are zero, then b, c, d are unconstrained):

$$cA(I-A^2)^{1/2} + bdA(I-A^2)^{1/2} = bA(I-A^2)^{1/2} + cdA(I-A^2)^{1/2} = 0$$
$$c + bd = b + cd = 0$$

The second equation simplifies to $c+bd = 0$, since multiplying the second set of terms $b+cd$ by d produces $c + bd$ because $d^2 = 1$. Therefore, $c = -bd$: either $c = 1$ and $b = -d$, or $c = -1$ and $b = d$. We rewrite this constraint as $d = -bc$ to produce our final expression with unconstrained $b, c \in \{-1, 1\}$.

Lemma 1 *Any block 2×2 matrix composed of equally sized real diagonal blocks must assume the following form (up to multiplication by -1), where $b, c \in \{-1, 1\}$ and the spectral radius of A can be no larger than 1:*

$$\begin{bmatrix} A & b(I-A^2)^{1/2} \\ c(I-A^2)^{1/2} & -bcA \end{bmatrix}$$

Appendix B. Single-area basis Q_{single}

Now we will consider a vector of activity from two groups of neurons, x and y; each group has separate excitatory and inhibitory populations. We aggregate all neurons' activities as a vector r, which evolves over time via $\tau\dot{r}(t) = -r(t) + Wr(t) + s(t)$, for some stimulus $s(t)$ and weight matrix $W = \begin{bmatrix} E_x & -I_x & L_k & 0 \\ E_x & -I_x & L_k & 0 \\ L_l & 0 & E_y & -I_y \\ L_l & 0 & E_y & -I_y \end{bmatrix}$, where $r = \begin{bmatrix} e_x \\ i_x \\ e_y \\ i_y \end{bmatrix}$. Every weight block $E_x, E_y, I_x, I_y, L_k, L_l$ consists of positive entries, and all four blocks of r are vectors in $\mathbb{R}^N$.

Without any additional assumptions, we can already make an initial decomposition $W = Q_{init}\widehat{W}Q_{init}$ of W via basis Q_{init} (see Section A.2):

$$\widehat{W}_{init} = \begin{bmatrix} E_x - I_x & E_x + I_x & L_k & L_k \\ 0 & 0 & 0 & 0 \\ L_l & L_l & E_y - I_y & E_y + I_y \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad Q_{init} = \frac{1}{\sqrt{2}} \begin{bmatrix} I & I & 0 & 0 \\ I & -I & 0 & 0 \\ 0 & 0 & I & I \\ 0 & 0 & I & -I \end{bmatrix} \tag{7}$$

Next, we will assume: (1) that all aforementioned weight blocks are mutually diagonalizable by some real orthogonal P (for example, PE_xP^T is diagonal), and (2) $E_x = I_x$ and

$E_y = I_y$. Then, we can obtain a further decomposed decomposed matrix $\widehat{W}_{single}$, where the basis Q_{single} is composed of single-area activations:

$$\widehat{W}_{single} = \begin{bmatrix} 0 & 2PE_xP^T & PL_kP^T & PL_kP^T \\ 0 & 0 & 0 & 0 \\ PL_lP^T & PL_lP^T & 0 & 2PE_xP^T \\ 0 & 0 & 0 & 0 \end{bmatrix} \triangleq \begin{bmatrix} 0 & \Lambda_x & \Lambda_k & \Lambda_k \\ 0 & 0 & 0 & 0 \\ \Lambda_l & \Lambda_l & 0 & \Lambda_y \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{8}$$

$$Q_{single} = \frac{1}{\sqrt{2}} \begin{bmatrix} P & P & 0 & 0 \\ P & -P & 0 & 0 \\ 0 & 0 & P & P \\ 0 & 0 & P & -P \end{bmatrix} \tag{9}$$

One straightforward way to ensure mutual diagonalizability of all weight blocks is for them to all be circulant. For our results, we will generally assume that all elements of $\widehat{W}_{single}$ are non-negative and real; the latter entails that all weight blocks are circulant, while the former, in the circulant case, places additional constraints on the values of each Λ block. The $\widehat{\boldsymbol{W}}_{single}, \boldsymbol{Q}_{single}$ used in the main text can be obtained through a simple re-ordering of Q_{single}.

Note that in the main text, networks consist of one E and one I unit ($N = 1$), so, connectivity matrices reduce to scalars (e.g. $E_x = e_x$)

Appendix C. Multi-area basis Q_{multi}

C.1. Problem statement

From the previous section, we have a partially decomposed weight matrix of diagonal blocks. For convenience, we name each unique diagonal block and also rewrite the partially decomposed weights as a 2×2 block matrix of 2×2 upper-triangular block matrices (the names of block matrices here are not always consistent with the main text):

$$\widehat{W}_{single} = \begin{bmatrix} 0 & \Lambda_x & \Lambda_k & \Lambda_k \\ 0 & 0 & 0 & 0 \\ \Lambda_l & \Lambda_l & 0 & \Lambda_y \\ 0 & 0 & 0 & 0 \end{bmatrix} \triangleq \begin{bmatrix} X & K \\ L & Y \end{bmatrix}$$

We seek a Schur decomposition of the partially decomposed matrix $\widehat{W}_{single} = Q_{single}^T W Q_{single}$: a real orthogonal matrix Q_{diag} such that $\widehat{W}_{multi} = Q_{diag}^T (Q_{single}^T W Q_{single}) Q_{diag}$ is upper triangular. Since the Schur decomposition basis Q_{diag} is not unique, we impose constraints on Q_{diag} that satisfy a subjective notion of simplicity:

1. $Q_{diag} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ is a 2×2 matrix of real diagonal blocks.
2. A, B, C, D should be non-negative.
3. When $\Lambda_x = \Lambda_y$ and $\Lambda_k = \Lambda_l$, A, B, C, D should be proportional to $\pm I$.

C.2. Conditions for upper triangularity

Now we write $\widehat{W}_{multi} = Q_{diag}^T \widehat{W}_{single} Q_{diag}$ to find constraints on A, B, C, D that ensure its upper triangularity:

$$\begin{aligned} \widehat{W}_{multi} = Q_{diag}^T \widehat{W}_{single} Q_{diag} &= \begin{bmatrix} A & C \\ B & D \end{bmatrix} \begin{bmatrix} X & K \\ L & Y \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \\ &= \begin{bmatrix} AX + CL & AK + CY \\ BX + DL & BK + DY \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \\ &= \begin{bmatrix} AXA + CLA + AKC + CYC & AXB + CLB + AKD + CYD \\ BXA + DLA + BKC + DYC & BXB + DLB + BKD + DYD \end{bmatrix} \end{aligned} \tag{10}$$

Since A, B, C, D are diagonal and X, K, L, Y are upper triangular, each of the four blocks of eq. (10) is already upper-triangular. Therefore, we need only set the bottom-left block to zero to ensure upper triangularity.

However, Lemma 1 proves that our diagonal matrices A, B, C, D are themselves constrained if Q_{diag} is orthogonal (theorem 1), so we update Q_{diag} and eq. (10) accordingly:

$$Q_{diag} = \begin{bmatrix} A & bB \\ cB & -bcA \end{bmatrix}, \quad \text{where } b, c \in \{-1, 1\} \text{ and } B = (I - A^2)^{1/2} \tag{11}$$

$$\widehat{W}_{multi} = \begin{bmatrix} AXA + cBLA + cAKB + BYB & bAXB + bcBLB - bcAKA - bBYA \\ bBXA - bcALA + bcBKB - bAYB & BXB - cALB - cBKA + AYA \end{bmatrix} \tag{12}$$

Now we set the bottom-left block of $\widehat{W}_{multi}$ (eq. (12)) to zero, and assume B is invertible so that $A = MB = BM$, where $M = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix}$ is also diagonal:

$$\begin{gathered} bBXA - bcALA + bcBKB - bAYB = 0 \\ BXA - cALA + cBKB - AYB = 0 \\ BXMB - cBMLMB + cBKB - BMYB = 0 \\ XM - cMLM + cK - MY = 0 \end{gathered}$$

Next we write out each term and add or subtract them appropriately:

$$\begin{aligned} XM &= \begin{bmatrix} 0 & \Lambda_x \\ 0 & 0 \end{bmatrix} \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} = \begin{bmatrix} 0 & M_2\Lambda_x \\ 0 & 0 \end{bmatrix} \\ cMLM &= \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} c\Lambda_l & c\Lambda_l \\ 0 & 0 \end{bmatrix} \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} = \begin{bmatrix} cM_1^2\Lambda_l & cM_1M_2\Lambda_l \\ 0 & 0 \end{bmatrix} \\ cK &= \begin{bmatrix} c\Lambda_k & c\Lambda_k \\ 0 & 0 \end{bmatrix} \\ MY &= \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} 0 & \Lambda_y \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & M_1\Lambda_y \\ 0 & 0 \end{bmatrix} \end{aligned}$$

$$\begin{bmatrix} -cM_1^2\Lambda_l + c\Lambda_k & M_2\Lambda_x - cM_1M_2\Lambda_l + c\Lambda_k - M_1\Lambda_y \\ 0 & 0 \end{bmatrix} = 0 \tag{13}$$

C.3. Solving for M_1, M_2, A, and B

First, we solve the top-left block of eq. (13):

$$M_1 = s\sqrt{\Lambda_k\Lambda_l^{-1}}, \quad s \in \{-1, 1\}, \tag{14}$$

and then we solve the top-right block for M_2:

$$\begin{aligned} M_2\Lambda_x - cM_1M_2\Lambda_l &= -c\Lambda_k + M_1\Lambda_y \\ M_2 &= (M_1\Lambda_y - c\Lambda_k)(\Lambda_x - cM_1\Lambda_l)^{-1} \\ &= \frac{s\sqrt{\Lambda_k\Lambda_l^{-1}}\Lambda_y - c\Lambda_k}{\Lambda_x - cs\sqrt{\Lambda_k\Lambda_l}} \\ &= \frac{s\sqrt{\Lambda_k}\Lambda_y - c\Lambda_k\sqrt{\Lambda_l}}{\Lambda_x\sqrt{\Lambda_l} - cs\sqrt{\Lambda_k}\Lambda_l} \\ M_2 &= \frac{\sqrt{\Lambda_k}(s\Lambda_y - c\sqrt{\Lambda_k\Lambda_l})}{\sqrt{\Lambda_l}(\Lambda_x - cs\sqrt{\Lambda_k\Lambda_l})} \end{aligned} \tag{15}$$

Having solved for $M = \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix}$, we solve for A and B. They must satisfy $A = BM$ (from the definition of M) and $B = (I - A^2)^{1/2}$ (from the orthogonality of our basis Q_{diag}):

$$\begin{aligned} A &= (I - A^2)^{1/2} M \\ A^2 &= (I - A^2) M^2 \\ A^2(I + M^2) &= M^2 \\ A &= aM(I + M^2)^{-1/2}, \quad a \in \{-1, 1\} \\ B &= a(I + M^2)^{-1/2} \end{aligned}$$

Since a affects both A and B, it is simply a scalar in front of Q_{diag}, so we set $a = 1$ without loss of generality. Substituting A, B into eq. (11) produces an initial expression of Q_{diag} (eq. (16)), where the elements of M are given by eq. (14) and eq. (15):

$$Q_{diag} = \begin{bmatrix} (I + M^2)^{-1/2} & 0 \\ 0 & (I + M^2)^{-1/2} \end{bmatrix} \begin{bmatrix} M & bI \\ cI & -bcM \end{bmatrix} \tag{16}$$

C.4. Final expression for Q_{diag} and Q_{multi}

Since each nonzero element of M is a fraction, each non-zero element of Q_{diag} in eq. (11) takes the form $\frac{o/u}{\sqrt{1+(o/u)^2}}$ or $\frac{1}{\sqrt{1+(o/u)^2}}$, for some o, u. Here, we re-write Q_{diag} to make it more simple and minimize the number fractional terms. First, we define O, U as the numerators and denominators of every non-zero term in M:

$$M_1 = \frac{\sqrt{\Lambda_k}}{\sqrt{\Lambda_l}} \triangleq \frac{O_1}{U_1}, \quad M_2 = \frac{\sqrt{\Lambda_k}(\Lambda_y - c\sqrt{\Lambda_k \Lambda_l})}{\sqrt{\Lambda_l}(\Lambda_x - c\sqrt{\Lambda_k \Lambda_l})} \triangleq M_1 \frac{F}{G} \triangleq \frac{O_2}{U_2}$$

$$M = \begin{bmatrix} O_1 & 0 \\ 0 & O_2 \end{bmatrix} \begin{bmatrix} U_1 & 0 \\ 0 & U_2 \end{bmatrix}^{-1} \triangleq OU^{-1}$$

Next, we define a matrix of normalization constants N, which we use to simplify A, B:

$$N = \sqrt{O^2 + U^2} = \begin{bmatrix} \Lambda_k + \Lambda_l & 0 \\ 0 & \Lambda_k F^2 + \Lambda_l G^2 \end{bmatrix}^{1/2}$$

$$B = \frac{I}{\sqrt{I + M^2}} = \frac{U}{U\sqrt{I + O^2 U^{-2}}} = \frac{U}{\sqrt{O^2 + U^2}} = \frac{U}{N} = \begin{bmatrix} \sqrt{\Lambda_l} & 0 \\ 0 & \sqrt{\Lambda_l} G \end{bmatrix} N^{-1}$$

$$A = MB = BM = \frac{OU^{-1}U}{\sqrt{O^2 + U^2}} = \frac{O}{\sqrt{O^2 + U^2}} = \frac{O}{N} = \begin{bmatrix} \sqrt{\Lambda_k} & 0 \\ 0 & \sqrt{\Lambda_k} F \end{bmatrix} N^{-1}$$

Finally, we substitute our new expressions of A, B into Q_{diag} (eq. (11)):

$$Q_{diag} = \begin{bmatrix} ON^{-1} & bUN^{-1} \\ cUN^{-1} & -bcON^{-1} \end{bmatrix} = \begin{bmatrix} O & bU \\ cU & -bcO \end{bmatrix} \begin{bmatrix} N^{-1} & 0 \\ 0 & N^{-1} \end{bmatrix}$$

$$Q_{diag} = \begin{bmatrix} \sqrt{\Lambda_k} & 0 & b\sqrt{\Lambda_l} & 0 \\ 0 & \sqrt{\Lambda_k} F & 0 & b\sqrt{\Lambda_l} G \\ c\sqrt{\Lambda_l} & 0 & -bc\sqrt{\Lambda_k} & 0 \\ 0 & c\sqrt{\Lambda_l} G & 0 & -bc\sqrt{\Lambda_k} F \end{bmatrix} \begin{bmatrix} N_1^{-1} & 0 & 0 & 0 \\ 0 & N_2^{-1} & 0 & 0 \\ 0 & 0 & N_1^{-1} & 0 \\ 0 & 0 & 0 & N_2^{-1} \end{bmatrix} \tag{17}$$

To arrive at our final basis Q_{multi}, we multiply Q_{single} (eq. (9)) by Q_{diag} (eq. (17)):

$$Q_{multi} = Q_{single}Q_{diag} = Q_{single}\begin{bmatrix} N_1^{-1}\sqrt{\Lambda_k} & 0 & bN_1^{-1}\sqrt{\Lambda_l} & 0 \\ 0 & N_2^{-1}\sqrt{\Lambda_k}F & 0 & bN_2^{-1}\sqrt{\Lambda_l}G \\ cN_1^{-1}\sqrt{\Lambda_l} & 0 & -bcN_1^{-1}\sqrt{\Lambda_k} & 0 \\ 0 & cN_2^{-1}\sqrt{\Lambda_l}G & 0 & -bcN_2^{-1}\sqrt{\Lambda_k}F \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{bmatrix} PN_1^{-1}\sqrt{\Lambda_k} & PN_2^{-1}\sqrt{\Lambda_k}F & bPN_1^{-1}\sqrt{\Lambda_l} & bPN_2^{-1}\sqrt{\Lambda_l}G \\ PN_1^{-1}\sqrt{\Lambda_k} & -PN_2^{-1}\sqrt{\Lambda_k}F & bPN_1^{-1}\sqrt{\Lambda_l} & -bPN_2^{-1}\sqrt{\Lambda_l}G \\ cPN_1^{-1}\sqrt{\Lambda_l} & cPN_2^{-1}\sqrt{\Lambda_l}G & -bcPN_1^{-1}\sqrt{\Lambda_k} & -bcPN_2^{-1}\sqrt{\Lambda_k}F \\ cPN_1^{-1}\sqrt{\Lambda_l} & -cPN_2^{-1}\sqrt{\Lambda_l}G & -bcPN_1^{-1}\sqrt{\Lambda_k} & bcPN_2^{-1}\sqrt{\Lambda_k}F \end{bmatrix}$$

To obtain the $\boldsymbol{Q}_{multi}$ used in the main text, we set $b = 1, c = -1$ (so that F and G are always positive), and condense $\frac{1}{\sqrt{2}}\begin{bmatrix} P \\ \pm P \end{bmatrix}$ terms into "balanced" and "unbalanced" matrices.

C.5. Triangularized dynamics

Now we express the triangularized dynamics $\widehat{W}_{multi} = Q_{diag}^T \widehat{W}_{single} Q_{diag}$ from eq. (12):

$$\widehat{W}_{multi} = \begin{bmatrix} AXA + cBLA + cAKB + BYB & bAXB + bcBLB - bcAKA - bBYA \\ 0 \text{ (by definition of } M) & BXB - cALB - cBKA + AYA \end{bmatrix},$$

and write out several terms that appear therein. As a shorthand, $J = \begin{bmatrix} \lambda\Lambda_j & \Lambda_j \\ 0 & 0 \end{bmatrix}$ for $j \in \{x, y, l, k\}$.

$$AJ = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}\begin{bmatrix} \lambda\Lambda_j & \Lambda_j \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \lambda A_1\Lambda_j & A_1\Lambda_j \\ 0 & 0 \end{bmatrix}$$

$$AJA = \begin{bmatrix} \lambda A_1^2\Lambda_j & A_1A_2\Lambda_j \\ 0 & 0 \end{bmatrix}, \quad BJB = \begin{bmatrix} \lambda B_1^2\Lambda_j & B_1B_2\Lambda_j \\ 0 & 0 \end{bmatrix}$$

$$AJB = \begin{bmatrix} \lambda A_1B_1\Lambda_j & A_1B_2\Lambda_j \\ 0 & 0 \end{bmatrix}, \quad BJA = \begin{bmatrix} \lambda A_1B_1\Lambda_j & A_2B_1\Lambda_j \\ 0 & 0 \end{bmatrix}$$

$$A_1^2 = N_1^{-2}\Lambda_k = \frac{\Lambda_k}{\Lambda_k + \Lambda_l}$$

$$B_1^2 = N_1^{-2}\Lambda_l = \frac{\Lambda_l}{\Lambda_k + \Lambda_l}$$

$$A_1B_1 = N_1^{-2}\sqrt{\Lambda_k\Lambda_l} = \frac{\sqrt{\Lambda_k\Lambda_l}}{\Lambda_k + \Lambda_l}$$

$$A_1A_2 = (N_1N_2)^{-1}\Lambda_kF$$

$$A_2B_1 = (N_1N_2)^{-1}\sqrt{\Lambda_k\Lambda_l}F$$

$$A_1B_2 = (N_1N_2)^{-1}\sqrt{\Lambda_k\Lambda_l}G$$

$$B_1B_2 = (N_1N_2)^{-1}\Lambda_lG$$

$$N_1N_2 = \left((\Lambda_k + \Lambda_l)\left(\Lambda_kF^2 + \Lambda_lG^2\right)\right)^{1/2}$$

Next, we express each nonzero block of $\widehat{W}_{multi}$.

C.5.1. Top-left block: $AXA + cBLA + cAKB + BYB$

$$
\begin{aligned}
&AXA + cBLA + cAKB + BYB \\
&= \begin{bmatrix} 0 & A_1A_2\Lambda_x \\ 0 & 0 \end{bmatrix} + c\begin{bmatrix} A_1B_1\Lambda_l & A_2B_1\Lambda_l \\ 0 & 0 \end{bmatrix} + c\begin{bmatrix} A_1B_1\Lambda_k & A_1B_2\Lambda_k \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & B_1B_2\Lambda_y \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} cA_1B_1(\Lambda_l+\Lambda_k) & A_1A_2\Lambda_x + cA_2B_1\Lambda_l + cA_1B_2\Lambda_k + B_1B_2\Lambda_y \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\left((\Lambda_x\Lambda_k + c\Lambda_l\sqrt{\Lambda_k\Lambda_l})F + (c\Lambda_k\sqrt{\Lambda_k\Lambda_l} + \Lambda_y\Lambda_l)G\right) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\Big((\Lambda_x\Lambda_k + c\Lambda_l\sqrt{\Lambda_k\Lambda_l})(\Lambda_y - c\sqrt{\Lambda_k\Lambda_l}) \\ & \quad +(\Lambda_y\Lambda_l + c\Lambda_k\sqrt{\Lambda_k\Lambda_l})(\Lambda_x - c\sqrt{\Lambda_k\Lambda_l})\Big) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\Big(\Lambda_x\Lambda_y\Lambda_k + (c\Lambda_y\Lambda_l - c\Lambda_x\Lambda_k)\sqrt{\Lambda_k\Lambda_l} - \Lambda_l^2\Lambda_k \\ & \quad +\Lambda_x\Lambda_y\Lambda_l + (c\Lambda_x\Lambda_k - c\Lambda_y\Lambda_l)\sqrt{\Lambda_k\Lambda_l} - \Lambda_k^2\Lambda_l\Big) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\left(\Lambda_x\Lambda_y\Lambda_k - \Lambda_l^2\Lambda_k + \Lambda_x\Lambda_y\Lambda_l - \Lambda_k^2\Lambda_l\right) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}(\Lambda_k+\Lambda_l)(\Lambda_x\Lambda_y - \Lambda_k\Lambda_l) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & N_2^{-1}\sqrt{\Lambda_k+\Lambda_l}(\Lambda_x\Lambda_y - \Lambda_k\Lambda_l) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} c\sqrt{\Lambda_k\Lambda_l} & \frac{N_1}{N_2}(\Lambda_x\Lambda_y - \Lambda_k\Lambda_l) \\ 0 & 0 \end{bmatrix}
\end{aligned}
$$

C.5.2. Bottom-right block: $BXB - cALB - cBKA + AYA$

$$
\begin{aligned}
&AYA - cBKA - cALB + BXB \\
&= \begin{bmatrix} -c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\left((\Lambda_y\Lambda_k - c\Lambda_k\sqrt{\Lambda_k\Lambda_l})F + (-c\Lambda_l\sqrt{\Lambda_k\Lambda_l} + \Lambda_x\Lambda_l)G\right) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} -c\sqrt{\Lambda_k\Lambda_l} & (N_1N_2)^{-1}\left(\Lambda_kF^2 + \Lambda_lG^2\right) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} -c\sqrt{\Lambda_k\Lambda_l} & N_1^{-1}\sqrt{\Lambda_kF^2 + \Lambda_lG^2} \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} -c\sqrt{\Lambda_k\Lambda_l} & \frac{N_2}{N_1} \\ 0 & 0 \end{bmatrix}
\end{aligned}
$$

C.5.3. Top-right block: $bAXB + bcBLB - bcAKA - bBYA$

$$
\begin{aligned}
&= -bcAKA - bBYA + bAXB + bcBLB \\
&= -bc\begin{bmatrix} A_1^2\Lambda_k & A_1A_2\Lambda_k \\ 0 & 0 \end{bmatrix} - b\begin{bmatrix} 0 & A_2B_1\Lambda_y \\ 0 & 0 \end{bmatrix} + b\begin{bmatrix} 0 & A_1B_2\Lambda_x \\ 0 & 0 \end{bmatrix} + bc\begin{bmatrix} B_1^2\Lambda_l & B_1B_2\Lambda_l \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} bc(B_1^2\Lambda_l - A_1^2\Lambda_k) & -bcA_1A_2\Lambda_k - bA_2B_1\Lambda_y + bA_1B_2\Lambda_x + bcB_1B_2\Lambda_l \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} bc\frac{\Lambda_l^2-\Lambda_k^2}{\Lambda_k+\Lambda_l} & (N_1N_2)^{-1}\left(-b(c\Lambda_k^2 + \Lambda_y\sqrt{\Lambda_k\Lambda_l})F + b(\Lambda_x\sqrt{\Lambda_k\Lambda_l} + c\Lambda_l^2)\right) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} bc(\Lambda_l - \Lambda_k) & b(N_1N_2)^{-1}\Big((c\Lambda_k^2 + \Lambda_y\sqrt{\Lambda_k\Lambda_l})(-\Lambda_y + c\sqrt{\Lambda_k\Lambda_l}) \\ & \quad +(\Lambda_x\sqrt{\Lambda_k\Lambda_l} + c\Lambda_l^2)(\Lambda_x - c\sqrt{\Lambda_k\Lambda_l})\Big) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} bc(\Lambda_l - \Lambda_k) & b(N_1N_2)^{-1}\Big(- c\Lambda_k^2\Lambda_y + (-\Lambda_y^2 + \Lambda_k^2)\sqrt{\Lambda_k\Lambda_l} + c\Lambda_y\Lambda_k\Lambda_l \\ & \quad +c\Lambda_l^2\Lambda_x + (\Lambda_x^2 - \Lambda_l^2)\sqrt{\Lambda_k\Lambda_l} - c\Lambda_x\Lambda_k\Lambda_l\Big) \\ 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} bc(\Lambda_l - \Lambda_k) & b(N_1N_2)^{-1}\left(\sqrt{\Lambda_k\Lambda_l}(\Lambda_x^2 + \Lambda_k^2 - \Lambda_y^2 - \Lambda_l^2) + c(\Lambda_l - \Lambda_k)(\Lambda_x\Lambda_l + \Lambda_y\Lambda_k)\right) \\ 0 & 0 \end{bmatrix}
\end{aligned}
$$

C.5.4. Final expression

The final expression for $\widehat{W}_{multi}$, again setting $b = 1, c = -1$, is:

$$
\widehat{W}_{multi} = \begin{bmatrix} -\sqrt{\Lambda_k\Lambda_l} & \frac{N_1}{N_2}(\Lambda_x\Lambda_y - \Lambda_k\Lambda_l) & \Lambda_k - \Lambda_l & \frac{\beta}{N_1N_2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{\Lambda_k\Lambda_l} & \frac{N_2}{N_1} \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{18}
$$

where $\beta = \sqrt{\Lambda_k\Lambda_l}(\Lambda_x^2 + \Lambda_k^2 - \Lambda_y^2 - \Lambda_l^2) + (\Lambda_k - \Lambda_l)(\Lambda_x\Lambda_l + \Lambda_y\Lambda_k)$. To obtain the $\widehat{\boldsymbol{W}}_{multi}$ used in the main text, we simply condense $\widehat{W}_{multi}$ by writing it as a Kronecker product.

Appendix D. Steady-State Activity

We seek to characterize dynamical systems of the form $\tau\dot{r}(t) = -r(t) + Wr(t) + s(t)$. If we assume $s(t) = s$ is constant, define $\theta = I - W$, define $\mu = \theta^{-1}s$, and set $\tau = 1$ (without loss of generality), such that $\dot{r} = \theta(\mu - r(t))$, then we can easily examine the mean of $r(t)$ according to (8):

$$\begin{aligned}
r(t) &= e^{-\theta t}r(0) + (I - e^{-\theta t})\mu \\
\lim_{t\to\infty} r(t) &= \theta^{-1}s \quad \text{since } \theta \text{ is positive definite} \\
\lim_{t\to\infty} Q^T r(t) &= Q^T(I - W)^{-1}s \\
&= Q^T(Q(I - \widehat{W})Q^T)^{-1}s \\
&= Q^T Q(I - \widehat{W})^{-1}Q^T s \\
&= (I - \widehat{W})^{-1}Q^T s
\end{aligned}$$

In the main text, we are interested in the steady-state values of $r(t)$ when projected via Q_{single}. This requires calculating $(I - \widehat{W}_{single})^{-1}$, which we do by leveraging the property $\begin{bmatrix} A & B \\ 0 & C \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & -A^{-1}BC^{-1} \\ 0 & C^{-1} \end{bmatrix}$:

$$\begin{aligned}
(I - \widehat{W}_{single})^{-1} &= \begin{bmatrix} 1 & -k & -2e_x & -k \\ -\ell & 1 & -\ell & -2e_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^{-1} \\
&= \begin{bmatrix} \begin{bmatrix} \frac{1}{1-k\ell} & \frac{k}{1-k\ell} \\ \frac{\ell}{1-k\ell} & \frac{1}{1-k\ell} \end{bmatrix} & \begin{bmatrix} \frac{1}{1-k\ell} & \frac{k}{1-k\ell} \\ \frac{\ell}{1-k\ell} & \frac{1}{1-k\ell} \end{bmatrix} \begin{bmatrix} 2e_x & k \\ \ell & 2e_y \end{bmatrix} \\ 0 & I \end{bmatrix} \\
&= \begin{bmatrix} \frac{1}{1-k\ell} & \frac{k}{1-k\ell} & \frac{2e_x+k\ell}{1-k\ell} & \frac{k(1+2e_y)}{1-k\ell} \\ \frac{\ell}{1-k\ell} & \frac{1}{1-k\ell} & \frac{\ell(1+2e_x)}{1-k\ell} & \frac{2e_y+k\ell}{1-k\ell} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}$$

We can use the above result to write the steady-state values of $\boldsymbol{Q}_{single}^T\boldsymbol{r}(t)$ as a function of a constant input $\boldsymbol{Q}_{single}^T\boldsymbol{s}$:

$$\lim_{t\to\infty} \begin{bmatrix} \boldsymbol{b}_x \\ \boldsymbol{b}_y \\ \boldsymbol{u}_x \\ \boldsymbol{u}_y \end{bmatrix} = \begin{bmatrix} \frac{1}{1-k\ell} & \frac{k}{1-k\ell} & \frac{2e_x+k\ell}{1-k\ell} & \frac{k(1+2e_y)}{1-k\ell} \\ \frac{\ell}{1-k\ell} & \frac{1}{1-k\ell} & \frac{\ell(1+2e_x)}{1-k\ell} & \frac{2e_y+k\ell}{1-k\ell} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{s}_{b,x} \\ \boldsymbol{s}_{b,y} \\ \boldsymbol{s}_{u,x} \\ \boldsymbol{s}_{u,y} \end{bmatrix}$$

We can immediately see that the output $\boldsymbol{b}_x$ of area x amplifies the inputs $\boldsymbol{s}_{b,x}$ of area x depending only on k, l, e_x, and amplifies the inputs $\boldsymbol{s}_{b,y}, \boldsymbol{s}_{u,y}$ depending only on k, l, e_y.

$$\boldsymbol{b}_x = \left(\frac{1}{1-FB}\boldsymbol{s}_{b,x} + \frac{2e_x + FB}{1-FB}\boldsymbol{s}_{u,x}\right) + \left(\frac{B}{1-FB}\boldsymbol{s}_{b,y} + \frac{B(1+2e_y)}{1-FB}\boldsymbol{s}_{u,y}\right)$$