Flexible Control of Mutual Inhibition: a Neural Model of Two-interval Discrimination

Supporting Methods

Christian K. Machens, Ranulfo Romo, and Carlos D. Brody

1 Overview

In **Section 2**, we describe in detail the neural model and methods used to design the two-node mutual inhibition network of Figs. 3 and 4 of the main text. As part of the Online Supporting Material, documented MATLAB language code is also provided, with the purpose of allowing readers easy reproduction and modification of the network design process and the resulting simulations.

Section 2.1 describes how we were led to the simple mutual inhibition architecture sketched in Fig. 3b of the main text. Section 2.1 also introduces the general framework in which populations of neurons are represented by their average activity ("mean-field," or "rate" modeling), with dynamics governed by first-order differential equations. This approximation essentially depends on assuming that there are enough asynchronously firing neurons per population, with sufficiently slow synaptic kinetics, that when the postynaptic activity is averaged over the spiking population the result can be treated as a smoothly time-varying parameter [5].

Sections 2.2 through 2.7 then go in detail through the network design process, and through the relationship between the rate model of Fig. 3 and the spiking neuron model of Fig. 4.

In **Section 3**, we describe a more complex neural instantiation of the algorithm of Fig. 2. In this instantiation, we adapt Koulakov et al.'s ideas [8] to make the memory maintenance mode of the model (a line attractor) robust against small parameter changes. A future paper will focus on this more robust model and its properties. Here we merely present the robust model as proof, by construction, that such a robust model can be built, and briefly describe how the dynamics of the robust model can be approximately understood in two-dimensional terms, similar to the diagrams of Fig. 3.

In **Section 4**, we briefly illustrate data showing how frontal lobe neurons change from one sign of stimulus-dependency during the loading and maintenance periods, to the opposite sign of stimulus-dependency during stimulus comparison/decision (Fig. 3h of main text).

Contents

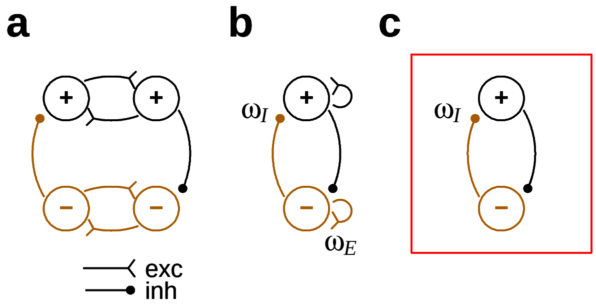

Figure S1: Simplification of connectivity. (a) Connectivity as suggested by analysis of noise covariations (replotted from Fig. 3a). (b) Simplified circuit for the mean-field model: each of the two nodes represents the average activity of "plus" and "minus" neurons, respectively. (c) Simplest circuit, without self-excitation connections. Section 2.3 describes how self-excitation can be incorporated into the framework of the two-node model.

2 Two-node mutual inhibition model (Fig. 3 and 4 of main text)

2.1 General framework: network connectivity and dynamical equations

As described in the main text, the neurophysiological data reveals that during the maintenance period there are *positive* noise covariations between pairs of "plus" PFC neurons and between pairs of "minus" PFC neurons, but *negative* noise correlations between pairs of neurons where one is "plus" and one is "minus." This led us to consider the network architecture sketched in Fig. S1a (replotted from Fig. 3a in the paper) which has excitatory connections between neurons of the same sign and inhibitory connections between neurons of opposite signs. For simplicity, there are only two "plus" neurons and two "minus" neurons shown in the graph; we imagine, however, that there are many more neurons of each type.

Let us suppose that we could qualitatively capture the behavior of this network if we described it, at any instant in time, in terms of the average activity of all "plus" neurons, and the average activity of all "minus" neurons—this is referred to as a "mean-field," or "rate" approximation. In such an approximation, the network can be described by the graph of Fig. S1b, where each node represents the average activity of a population of neurons. (In this averaged description, a connection between two nodes represents connections from all the neurons in the first node to all the neurons in the second node. E.g., an inhibitory connection from the plus to the minus node in Fig. S1b means that every neuron in the plus node sends an inhibitory connection to all the neurons in the minus node.)

If there are many asynchronously firing neurons per node, the average activity of the node will be relatively smooth, and we can describe it in terms of a continuous variable, x. (In Section 2.6 we describe the conditions under which this approximation holds.) In steady state, x will be a function of the excitatory (g_E) and inhibitory (g_I) inputs to the node,

$$x = F(g_I, g_E). \tag{1}$$

When these inputs are changing in time, we may imagine that x will follow them with some characteristic time constant τ. (Following a number of authors (e.g., [5, 11, 3, 10, 6]), Section 2.6 describes how the time constant for our model is in fact the synaptic time constant.) We thus generalize Eq. 1 to

$$\tau\dot{x} = -x + F(g_I, g_E). \tag{2}$$

For the system of Fig. S1b, let the two nodes be x for "plus" and y for "minus." The weights ω_I and ω_E describe the scaling from node outputs to inputs. Suppose each node may receive an additional external excitation signal (E_x to the plus node, E_y to the minus node). The equations describing the dynamics are then

$$\begin{aligned} \tau\dot{x} &= -x + F(\omega_I y \, , \, \omega_E x + E_x) \\ \tau\dot{y} &= -y + F(\omega_I x \, , \, \omega_E y + E_y) \, . \end{aligned} \tag{3}$$

In steady state, the value of each node is implicitly defined as a function of the value of the other node. For example, for the x node,

$$\begin{aligned} \dot{x} &= 0 \Rightarrow \; x = F(\omega_I y \, , \, \omega_E x + E_x) \\ \text{solving for } x, \quad &\Rightarrow \; x = \hat{F}(y) \, . \end{aligned} \tag{4}$$

The implicitly-defined solution curve $x = \hat{F}(y)$ is known as the "nullcline" for x, and describes the set of points where $\dot{x} = 0$. Its shape and position depend on the various parameters of the system, i.e., the parameters $\omega_I, \omega_E, E_x, E_y$ as well as the parameters that define the function $F(\cdot,\cdot)$ itself. Wherever the nullclines of both x and y cross, by definition both $\dot{x} = 0$ and $\dot{y} = 0$. Consequently, these crossings define the fixed points of the system.

• *The design of the network is in essence a search for appropriate fixed points during each of the phases of the two-interval discrimination task: an f1-dependent stable fixed point for loading, a quasi-continuum of fixed points for maintenance, and an f2-dependent unstable fixed point for comparison/decision. For the two-node network of Figs. 3 and 4 in the main text, we perform this search over two parameters that are part of the definition of $F(\cdot,\cdot)$, namely σ, and s_{max}, defined below in Eq. 7 and Eq. 8; as well as over the parameters w_I, E_x, and E_y.*

As we will describe below, searching over ω_E is also possible, and fits easily within the very same framework. However, computing nullclines (Eq. 4) is slower than simply computing input/output functions (Eq. 1). For maximum simplicity and computational efficiency, then, we made the model of Figs. 3 and 4 a purely mutual-inhibition network, without self-excitation: $\omega_E = 0$ (Fig. S1c). We later return to using self-excitation within a "plus" or "minus" population in the more complex, robust model of Section 3.

During the design of the mean-field network of Fig. 3 of the main text, we used one more simplifying approximation. Following [7] (see Section 2.2 and Fig. S3 below), we found that for our single-neuron model, inhibitory and excitatory inputs in Eq. 1 combine approximately linearly. That is, the function $F(\cdot,\cdot)$ can be approximated, using a scaling factor λ, as

$$F(g_I, g_E) \approx f(-g_I + \lambda g_E) \, . \tag{5}$$

Using this approximation, the traces of Fig. 3 therefore follow equations of the form:

$$\boxed{\begin{aligned} \tau\dot{x} &= -x + f(-\omega_I y + E_x) \\ \tau\dot{y} &= -y + f(-\omega_I x + E_y) \qquad (6) \end{aligned}}$$

Once the design of the mean-field network of Fig. 3 was complete, we confirmed (Fig. 4 main text, and Fig. S4 below) that the neuronal and connectivity parameters used in the mean-field approximation also led to the same behavior in a network where each node was not approximated by its average activity or by Eq. 5, but was in fact simulated as a collection of multiple, noisy, leaky integrate-and-fire neurons.

We now turn to describing how to find the functions $F(\cdot,\cdot)$ and $f(\cdot)$, as well as the steps necessary to find the neuronal and network parameters that result in the behavior of Figs. 3 and 4 of the main text.

2.2 Numerical computation of the i/o function $x = F(g_I, g_E)$

The single-neuron model that will form the basis of the network-of-neurons model is a simple, conductance-based, leaky integrate-and-fire, saturating synapse, single compartment model. During each time-step of the simulations, we inject independent noise into the membrane potential of individual model neurons, to mimic the effect of background synaptic activity on the cells.

For the continuous approximation, we will consider the *inputs* to the single-neuron model to be:

- g_E, the excitatory input conductance.
- g_I, the inhibitory input conductance.

While the *output* is:

- s, a synaptic output variable.

The equations driving the single neuron's membrane voltage V are:

$$\begin{array}{ll} \text{If last spike was} < t_{\text{refrac}} \text{ ago,} & V = V_{\text{reset}} \\ \text{Else if } V \geq V_{\text{thresh}} & V = V_{\text{reset}}; \ \text{emit spike} \\ \text{Else} & C\dot{V} = g_L(E_L - V) \ + g_E(E_E - V) + g_I(E_I - V) + \sigma\eta(t) \end{array} \qquad (7)$$

where C is the membrane capacitance, g_L the leak conductance, and E_L, E_E, and E_I are the reversal potentials of the leak, the excitatory, and inhibitory currents. The term $\eta(t)$ denotes the injected Gaussian white noise. A short simulation of an integrate-and-fire neuron is shown in Fig. S2a for a constant excitatory input. The resulting spike train is indicated by the vertical lines. Fig. S2b shows the firing rate of the neuron as a function of the inhibitory input g_I.

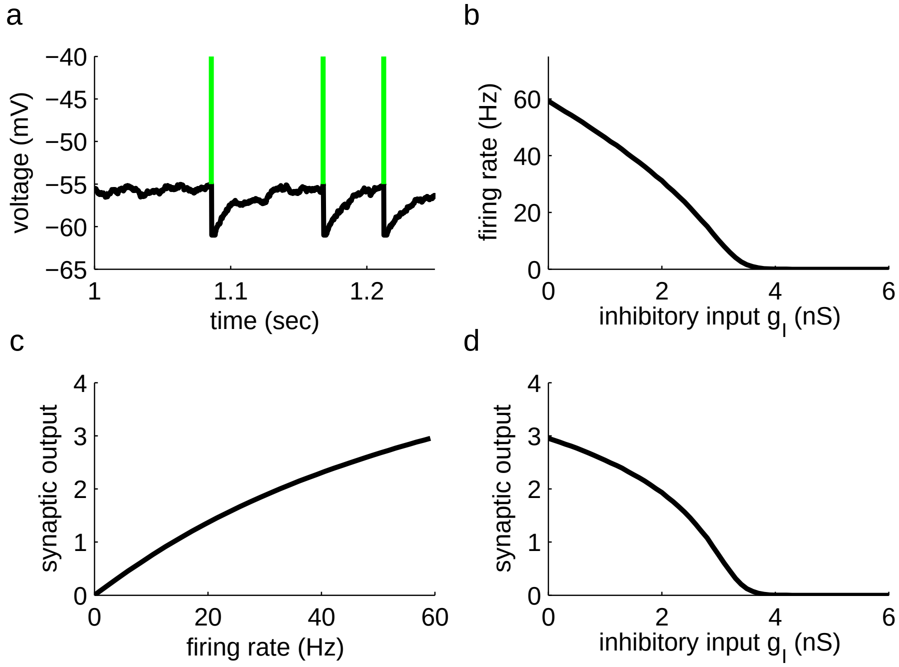

Figure S2: (a) The membrane potential of the integrate-and-fire neuron versus time for a constant excitatory conductance input. The random fluctuations in the voltage trace, as well as the irregularity of the spikes (marked in green), are caused by the Gaussian noise. (b) Firing rate of an integrate-and-fire neuron as a function of its inhibitory conductance input g_I. An excitatory conductance input $g_E = 2$ nS was held constant here. (c) Average synaptic output variable of an integrate-and-fire neuron as a function of its firing rate. (d) Average synaptic output variable of an integrate-and-fire neuron as a function of its inhibitory conductance input g_I. We call this relation the i/o function of the neuron.

Following [12], we model saturating synapses. The neuron's synaptic output variable s is a function of when the neuron emits spikes, and is proportional to the fraction of open receptor channels on the postsynaptic membrane. The fraction of open channels decreases exponentially with time constant τ in the absence of a spike, but if a spike arrives, s increases by $(s_{\max} - s)/s_{\max}$, where the synaptic saturation $s_{\max}$ denotes the maximum value that s can reach. That is, the variable s evolves according to:

$$\begin{array}{ll} \text{If a presynaptic spike was emitted,} & \text{increment s by } (s_{\max} - s)/s_{\max} \\ \text{Else} & \tau\dot{s} = -s \end{array} \tag{8}$$

Fig. S2c shows a neuron's average synaptic output as a function of its firing rate.

For a given set of neuron and synapse parameters, we can therefore compute a neuron's i/o function $s = F(g_I, g_E)$ numerically from these differential equations.

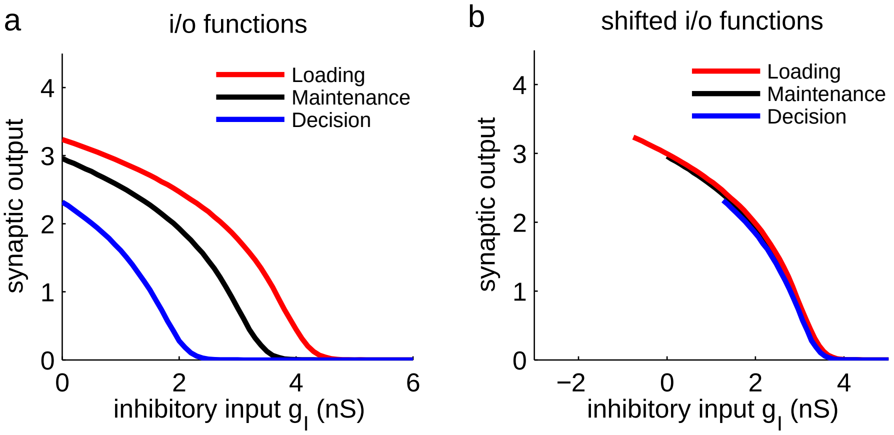

Figure S3: (a) Synaptic output of a neuron as a function of its inhibitory conductance input g_I. Shown are curves for three different values of the excitatory conductance input g_E, here corresponding to the loading, maintenance, and decision modes of the two-node network. (b) The curves have approximately the same shape and are therefore just shifted versions of each other. We therefore describe the i/o function of a neuron as dependent on a single conductance input only, $x = f(-g_I + \lambda g_E)$.

2.2.1 The approximation $F(g_I, g_E) \approx f(-g_I + \lambda g_E)$

In the two-node model, we are mostly interested in the synaptic output of a neuron as a function of its inhibitory input g_I since the excitatory inputs are held constant during each of the three phases. Numerically, we found that changing g_E during loading, maintenance, and decision does not change the i/o function shape much at all, but instead mostly shifts it along the inhibitory input axis (Fig. S3a). In practice, we therefore simply computed the neuron's i/o function for the value of g_E used during memory maintenance (black curve in Fig. S3a) and shifted this function when the external excitation changed. This procedure corresponds to approximating the neuron's i/o function by a function $f(\cdot)$ such that $s = f(-g_I + \lambda g_E) \approx F(g_I, g_E)$. Here, λ is a constant factor that describes by how much a change in excitatory inputs shifts the i/o functions along the inhibitory axis, see Fig. S3a. This factor was determined numerically. The accuracy of this approximation is depicted in Fig. S3b.

Within the supporting MATLAB-package, numerical computation of $f(\cdot)$ and determination of λ are carried out by the function `fcurve.m`; see also Section 2.8.

2.3 Designing the two-node mean-field network: Fig. 3

The main goal of the design procedure is to obtain a good overlap between the i/o functions of the “plus” and “minus” nodes, for the maintenance mode in Fig. 3f of the main text. This is the most delicate step; the loading and the comparison/decision dynamical modes then easily follow.

Fine-tuning the parameters for optimal overlap

For the design of the i/o functions used in our model, we first chose a constant excitatory input, g_E, that results in firing rates of ≈ 60 Hz in the absence of inhibitory inputs; this value was chosen to roughly match the maximum firing rates found in the data during the memory maintenance mode. We then searched for a combination of neuron and synapse parameters that guarantees a close overlap of the i/o functions for this value of the excitatory input, as shown in Fig. 3f.

To achieve this overlap, we found that three parameters were of specific importance: ω_I, the mutual inhibitory synaptic weight; $s_{\max}$, the synaptic saturation parameter, and σ, the membrane voltage noise parameter. To obtain the simplest possible model, we restricted ourselves to the no-self-excitation condition ($\omega_E = 0$; Fig. S1). Below, we include in square brackets the steps in the design process that would need to be taken if this constraint were relaxed.

Changing ω_I simply scales the output axis of the i/o function of a node. Hence, the two main parameters determining i/o function *shape* are $s_{\max}$, the synaptic saturation parameter (which mostly controls the flattening of the top part of the curve); and σ, the membrane voltage noise parameter (which controls how sharp the bottom "knee" of the curve is). We numerically explored the 2-dimensional space of these two parameters along the following steps:

1. Take a single spiking neuron model, with synaptic saturation parameter $s_{\max}$ and membrane voltage noise parameter σ.

2. As a function of a constant inhibitory input g_I, numerically compute the i/o function of a single neuron by averaging its synaptic output over long periods of time ($T = 1000$ seconds); this yields the continuous i/o function $f(\cdot)$ of a node. (In practice we approximate it as a look-up table.)

   [If allowing self-excitation, $\omega_E \neq 0$, we may choose to use the λ-approximation of Section 2.2.1, in which case it is sufficient to compute $f(\cdot)$. For greatest precision, the full two-dimensional $F(\cdot,\cdot)$ function should be computed over a fine grid of inputs g_I and g_E.]

3. For the current values of $s_{\max}$ and σ, find the optimal ω_I, as follows: Lay down the two symmetrical i/o functions, as in the mutually inhibitory circuit phase plane plot of Fig. 3f. Changing the value of ω_I, the mutual inhibitory synaptic weight, corresponds to simply scaling these i/o functions. We find the optimal ω_I by numerically finding the scaling that minimizes the distance between the central 65% of the two i/o curves.

   [If allowing self-excitation, $\omega_E \neq 0$, we would loop over possible values of w_E, and for each of these numerically find the nullcline $\hat{F}(\cdot)$, as defined in Eq. 4. The parameter ω_I will still act as a scaling of the nullcline, allowing swift looping over ω_I. For each ω_E, then, we find the optimal ω_I, and we loop over ω_E to find the optimal ω_I and ω_E]

4. Iterate steps 1–3 over a range of values of $s_{\max}$ and σ to find the optimal $s_{\max}, \sigma, \omega_I$.

   [If allowing self-excitation, $\omega_E \neq 0$, iteration over steps 1–3 would lead to the optimal $s_{\max}, \sigma, \omega_I$, and ω_E.]

Within the supporting MATLAB-package, this tuning procedure (with ω_E fixed at 0) is carried out by the function `autotuner.m`; see also Section 2.8. In addition, the resulting fine-tuned parameter values are provided in Section 2.7.

Parameters for loading and decision

Once the memory maintenance mode has been designed, the loading and decision modes can be set up quite easily. In practice, we chose the values for the excitatory inputs during loading and decision by hand, using the i/o function (or nullcline) plots of Fig. 3e-g as a guide. The excitatory inputs during loading and decision that were used in our simulations are provided in Section 2.7.

2.4 Implementing the two-node network with spiking neurons: Fig. 4

For Fig. 4a and c of the main text, we simulated a network of $N = 250$ spiking neurons per node. All neurons and synapses in a node had the same parameters. Every neuron received an excitatory conductance input g_E whose value depended on the phase of the task. Moreover, the plus and minus neurons received additional, oppositely tuned, excitatory inputs during loading and decision, corresponding to the inputs during the presence of a stimulus (Fig. 3e,g). Every neuron also received an inhibitory conductance input $g_I(t)$ from all the neurons of the other node.

Henceforth, we number all neurons from 1 to $2N$; neurons 1 through N are "plus" node neurons, while neurons $N+1$ through $2N$ are "minus" node neurons.

We define a connectivity matrix W_{ij}, where W_{ij} is the weight of the inhibitory synapse from the j-th neuron to the i-th neuron. $W_{ij} = 0$ if i and j are from the same node and $W_{ij} = \omega_I/N$ if they are from opposite nodes. (Dividing the individual synaptic inputs by N ensures that the total inhibitory conductance input to a neuron is ω_I times the average synaptic output of the other node, independently of the number of neurons per node.)

The inhibitory conductance input to neuron i is given by summing over all of its synaptic inputs,

$$g_{I,i}(t) = \sum_{j=1}^{2N} W_{ij} s_j(t) \tag{9}$$

where $s_j(t)$ is the synaptic output variable of the j-th neuron.

2.5 Direct comparison of the spiking network and the mean-field approximation

Fig. S4 shows the comparison of the synaptic outputs in the mean-field and the spiking two-node model. In general, the more neurons there are per node, and the longer the time constants of the spiking model (i.e., the synaptic or membrane time constants), when compared to the typical interspike interval

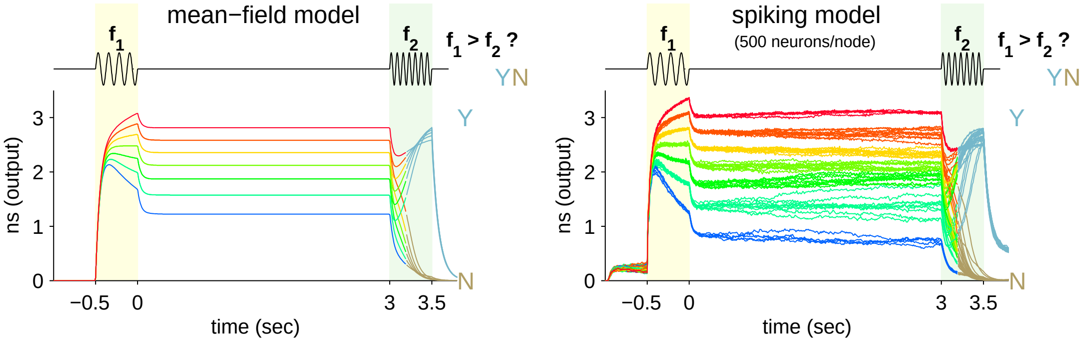

Figure S4: Comparison of synaptic outputs in the mean-field (left) and spiking (right) two-node model. As in the main text (Fig.1 and 4), the first stimulus frequency f1 is shown in the colors of the rainbow, the decision outcome is coded in light blue and brown. For clarity, the simulations in the mean-field model (left) were done without additive noise. In the spiking model, noise is present due to the asynchronous spiking of the neurons; to give an estimate of this noise, we present five different runs using $N = 500$ neurons per node. Unlike the main text, in which differences of 8 Hz between f1 and f2 were used (Fig. 1, Fig. 4), in the present figure we run through all combinations of (f1,f2) in which |f2-f1| = 4 Hz—a slightly more difficult comparison/decision than in the main text. The set of trial types is then (f1,f2) = { (10,14); (14,18); (18,22); (22,26); (26,30); (30,34); (14,10); (18,14); (22,18); (26,22); (30,26); (34,30) } Hz.

of the summed spike trains, the better the approximation between the spiking network model and its mean-field description, Eq. 6. The reasons for the slight differences between the two simulations (apart from the noise in the spiking network), are explained below.

Within the supporting MATLAB-package, Fig. S4 can be recreated using the functions `mfmaster.m` and `spikemaster.m`, cf. also Section 2.8.

2.6 Technical derivation of the mean-field dynamics: when does the mean-field approximation hold?

The mathematical rationale of the mean-field approach has been discussed at length in the literature [5, 11, 3, 10, 6]. Here, we provide a very brief outline of the basic ideas.

The integrate-and-fire neuron translates its conductance inputs into a spike train. For the *i*-th neuron, we write formally,

$$r_i(t) = R_i\big[g_{E,i}(t), g_{I,i}(t)\big] \qquad (10)$$

where $g_{E,i}(t)$ and $g_{I,i}(t)$ denote the synaptic conductance inputs and $r_i(t)$ is the generated spike train. The (stochastic) map $R_i[\cdot,\cdot]$ denotes the action of a single integrate-and-fire neuron: it maps the time series $g_{E,i}(t)$ and $g_{I,i}(t)$ into a spike train $r_i(t)$.

In the mean-field approach, we will reduce the full dynamics of the system to the dynamics of the average synaptic outputs of each node. Recalling that neurons 1 through N are “plus” node neurons, and neurons $N+1$ through $2N$ are “minus” node neurons, we introduce population variables for the total synaptic output of the two nodes:

$$x(t) = \frac{1}{N}\sum_{j=1}^{N} s_j(t)$$
$$y(t) = \frac{1}{N}\sum_{j=N+1}^{2N} s_j(t) \quad . \tag{11}$$

Note that by plugging the weight matrix into Eq. 9, and using the population variables of Eq. 11, we obtain $g_{I,i}(t) = \omega_I x(t)$ for $i = 1 \ldots N$ and $g_{I,i}(t) = \omega_I y(t)$ for $i = N+1 \ldots 2N$.

We can now write down differential equations for the population variables $x(t)$ and $y(t)$. Inserting the spike train definition Eq. 10 into the synaptic variable Eq. 8, and summing over neurons:

$$\tau\dot{x} = -x + \frac{1}{N}\sum_{j=1}^{N} \tau \frac{s_{\max} - s_j}{s_{\max}} R_j\left[g_{E,x}(t), \omega_I y\right]$$
$$\tau\dot{y} = -y + \frac{1}{N}\sum_{j=N+1}^{2N} \tau \frac{s_{\max} - s_j}{s_{\max}} R_j\left[g_{E,y}(t), \omega_I x\right]$$

If the neurons are sufficiently noisy (i.e., the noise terms σ are sufficiently large), the timing of individual spikes will become roughly uniform and we can average over the conductance changes induced by single spikes. For constant conductivity inputs g_I and g_E, let us define the activity or i/o function of a neuron as

$$F(g_I, g_E) = \left\langle \tau \frac{s_{\max} - s(t)}{s_{\max}} R[g_I, g_E] \right\rangle \tag{12}$$

where the angular brackets denote a time average. The i/o function $F(\cdot,\cdot)$ therefore maps constant conductivity inputs into a single number—the synaptic output; this is the same function encountered in Eq. 1.

If there are enough neurons in each node, and if the neurons are sufficiently noisy, then the time average in Eq. 12 can be replaced by an ensemble average, i.e., by the average synaptic output of all the neurons in a node. When many neurons are used, the jumps in the average synaptic output caused by the postsynaptic potentials of individual synapses will be very small and the approximation will be good. (This is closely analogous to being able to define a smooth average firing rate over many neurons, even though each individual spike train is composed of delta-functions.) Moreover, having a long synaptic decay means that, within a single synapse, there is averaging over many spikes. This reduces the number of neurons necessary to average over.

For the average in Eq. 12 to be only a function of the *present* g_E and g_I inputs, we need to assume that the synapses are always in steady-state. Accordingly, in our model the mean-field approximation is not strictly valid after each of the rapid shifts in the inputs (between stimulus f1, delay-period, and stimulus f2). Indeed, we find that on upwards changes of the excitatory inputs, the spiking neural network relaxes faster to its stationary state than the mean-field approximation, resulting in the slight differences during loading/maintenance between the two panels of Fig. S4. Overall, the mean-field approximation is nevertheless quite good, as can be seen by comparing the two panels in Fig. S4, Using Eq. 12, we obtain

$$\tau\dot{x} = -x + F(g_{E,x}(t), \omega_I y)$$
$$\tau\dot{y} = -y + F(g_{E,y}(t), \omega_I x)$$

For the parameters of our model, these equations can be further simplified by noting that the inhibitory inputs act subtractive on the i/o function (see Fig. S3 and also [7, 10]). We can therefore define a function $f(\cdot)$ such that $F(a,b) \approx f(-b+\lambda a)$ holds for a constant λ. Using this simplification, we obtain

$$\tau\dot{x} = -x + f(-\omega_I y + \lambda g_{E,x}(t))$$
$$\tau\dot{y} = -y + f(-\omega_I x + \lambda g_{E,y}(t))$$

where λ is a constant factor. The function $f(\cdot)$ now corresponds to the fine-tuned synaptic i/o function of the neurons, as shown in Fig. S2d. If we define the external inputs as $E_x = \lambda g_{E,x}(t)$ and $E_y = \lambda g_{E,y}(t)$, then we have recovered Eqs. 6. This derivation also shows that the time constant τ in Eqs. 6 correspond to the synaptic time constant.

2.7 Table of parameters to reproduce Fig.3 and 4 in main text

(1) **Simulation Method:**
ODEs were simulated using the Euler method $\Delta t = 0.1$ msec. Pilot experiments with more sophisticated and more computationally intensive methods (e.g., Reverse Euler, Runge-Kutta) led to identical overall results.

(2) **Number of neurons per population:**
$N = 250$

(3) **Integrate-and-fire neuron:**[1]
$C = 0.2$nF, $g_L = 10$nS, $E_L = -60$mV, $\sigma = 0.6$, $V_r = -61$mV, $V_\theta = -55$mV, $\tau_{\text{ref}} = 2$ms, $E_I = -75$mV, $E_E = -5$mV

(4) **Inhibitory synapses:**
$\tau_I = 80$ms, $s_{\max} = 7$.
GABA$_B$ synapses can provide inhibition with this slow timescale [4]. We have neglected the comparatively slow rise time of GABA$_B$ synapses; in this context, it is irrelevant, merely making the mean-field approximation better.

(5) **Inhibitory synaptic weights:**
$W_{ij} = 0$ if i and j are from the same node
$W_{ij} = \omega_I = 0.0011575$ if i and j are from different nodes.

(6) **External, excitatory inputs:**[2]

[1] A consistent set of units in the integrate-and-fire neurons is: Voltage in mV, time in msec, capacity in nF, and conductances in μS. Accordingly, the conductances provided here need to be converted into μS, i.e., divided by a factor 1000.

- Loading: $f_1 = (10, 14, 18, 22, 26, 30, 34)$ Hz, $t = -0.5 \dots 0$ sec
  $g_{E,x} = 2.3 + (-0.105, -0.07, -0.035, 0, 0.035, 0.07, 0.105)$ nS,
  $g_{E,y} = 2.3 + (0.105, 0.07, 0.035, 0, -0.035, -0.07, -0.105)$ nS.
- Memory: $t = 0 \dots 3$ sec
  $g_{E,x} = g_{E,y} = 2$ nS.
- Decision: $f_2 = (10, 14, 18, 22, 26, 30, 34)$ Hz, $t = 3 \dots 3.5$ sec
  $g_{E,x} = 1.5 + (0.105, 0.07, 0.035, 0, -0.035, -0.07, -0.105)$ nS
  $g_{E,y} = 1.5 + (-0.105, -0.07, -0.035, 0, 0.035, 0.07, 0.105)$ nS.

2.8 MATLAB software package to fully reproduce two-node model and its design

As part of the supporting online material, we also provide the software that we used to design and simulate the two-node model. This is intended to spare the reader the tedious job of implementing the model him-/herself.

The main files provided by the software package are:

`Contents.m`	Contents.
`Tutorial.m`	Brief tutorial on how to use the programs.
`initpar.m`	All the parameters of the two-node model as used in the main text, including the look-up table for the fine-tuned i/o function.
`fcurve.m`	This program computes a neuron's i/o function, using the parameters specified in `initpar.m`.
`autotuner.m`	This program finds the optimal values for σ, $s_{\max}$, and ω_I, otherwise using the parameters specified in `initpar.m`.
`mfsim.m`	This program animates the two-node model in the spirit of Fig. 3. By changing parameters in `initpar.m`, the influence of noise and stability of the memory period can be investigated.
`spikesim.m`	Simulation of the spiking two-node network. Returns the synaptic variables over time as well as the spike trains of all the neurons.
`mfmaster.m`	This program uses `mfsim.m` to create the mean-field simulations of Fig. 3, also shown in Fig. S4.
`spikemaster.m`	This program uses `spikesim.m` to create the spiking simulations of Fig. 4, also shown in Fig. S4.

3 Multi-node robust model : solving the fine-tuning problem

The maintenance phase of the simple two-node model of Section 2 requires fine-tuning of the model's parameters to a precision that is unlikely to be easily achieved in biology ($< 0.5\%$). This is a generic difficulty that all line attractors face [13, 1]. To address this problem, we adapted the ideas of Koulakov et al. (2002) to our model. The essential concept behind creating a more robust line attractor is to discretize: that is, replace the strictly flat L-function of the line attractor (see Fig. 2 of main text) with an L-function that has a densely-spaced set of individual, discrete, wells (stable fixed points of the dynamics). The existence of each of these can be relatively robust against parameter change. If spaced densely enough, the set of wells can approximate a continuum of stable points [8, 1].

Hysteresis in the input-output function of a node can create such a well. By building line attractors out of a chain of nodes, a chain of wells can therefore be created [8]. Here we build two such chains, connected to each other by mutual inhibition (Fig. S5). Each chain represents one of the "plus" or "minus" neural populations. Although the mean-field description of this system now has many more than two dimensions, its dynamical behavior can still be *approximately* described by two variables, which influence each other in a manner very similar to that described in Fig. 3 of the main text (Fig. S6). This enables us to achieve loading and comparison/decision-making, in addition to a robust maintenance mode.

Our principal goal in this section is to demonstrate, by construction, that such a robust model can exist. The essence of this section is therefore contained in Fig. S7, which illustrates the robustness of the model, and Section 3.7, which provides the full network parameters necessary to reconstruct the results of Fig. S7. A forthcoming paper (Machens and Brody, in preparation) will focus on the robust model and will describe in detail the motivation and methods behind its design, the two-dimensional approximation of the robust model, and the robust model's properties.

3.1 Network connectivity

The basic architecture of the robust model is shown in Fig. S5. The network consists of a one-dimensional chain of n units (seven of which are shown in the graph). Each unit is composed of four nodes, two excitatory and two inhibitory nodes. As in the two-node model, each node represents the average activity of a population of neurons; a connection between two nodes therefore means that each neuron in the first node connects to all the neurons in the second node. The robust model requires both excitatory and inhibitory connections; biological plausibility (Dale's law) requires that each node be composed of either inhibitory or excitatory neurons. To simplify the design of the network, inhibitory nodes in our robust model get their only input from a single presynaptic excitatory node in the same unit. Consequently, these inhibitory nodes just transform the output of the excitatory nodes into an inhibitory output and each four-node unit can therefore be collapsed into a two-node unit that is equivalent to the two-node model. We assume that all the (top) plus-nodes are coupled by identical excitatory connections and so are all the (bottom) minus-nodes. We will also refer to the plus- and minus-nodes as the plus- and minus-layer of the robust network.

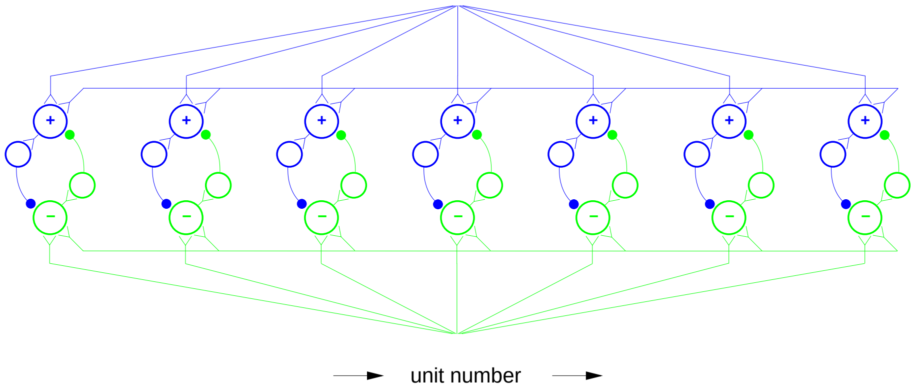

Figure S5: Connectivity of the robust model. The robust model consists of spatially arranged units (seven of which are shown); each unit consists of four nodes. Since the inhibitory nodes only transform an excitatory input into an inhibitory output of equal size, each unit can be thought of as a two-node model in which the nodes can simultaneously have excitatory and inhibitory outputs.

The motivation for this connectivity is two-fold: First, the connectivity is in agreement with the noise correlations: there are inhibitory connections between neurons of opposite signs and excitatory connections between neurons of the same sign. Second, both the upper (plus) layer and the lower (minus) layer in isolation correspond to the Koulakov model for a robust line attractor [8]. This feature of the layers will allow us to make the model robust against changes in the synaptic parameters.

3.2 Approximate 2-d description of mean-field network dynamics

Although the robust model is more involved than the two node model, its dynamics can be approximately projected down onto two dimensions. If we plot the summed activity of the plus neurons in the upper layer versus the summed activity of the minus neurons in the lower layer, then we obtain phase space plots as shown in Fig. S6a,c,e. The nullclines in these plots are just approximations: each nullcline describes the set of points at which no driving forces act on the summed activity of the respective layer.[2] Unfortunately, it is not possible to reduce the dynamics of this network into two dimensions and make the robustness "visible" at the same time. A detailed mathematical explanation of these issues will appear in a forthcoming paper (Machens and Brody, in preparation).

The plots in Fig. S6a,c,e are equivalent to the plots in Fig. 3e–g in the main paper. The activity of the top (plus) and bottom (minus) layer is displayed in the graphs on the top and bottom panels of Fig. S6b,d,f. Note that the different memory states of the continuous attractor correspond to shifted activity profiles (Fig. S6b). These states are made robust against changes in the network parameters by the hysteretic activity function of the excitatory neurons (see Fig. S8b and below); briefly, the

[2] Technically, the nullclines are obtained by expanding the differential equations for the summed activities of the two layers around the line attractor regime. Accordingly, the nullclines are exact on the line attractor (Fig. S6c), but approximations around it. The qualitative deviations of the nullclines in Fig. S6a, when compared with the two-node model (Fig. 3e) are therefore not significant and could be artefacts of the approximations (Machens and Brody, in prep.).

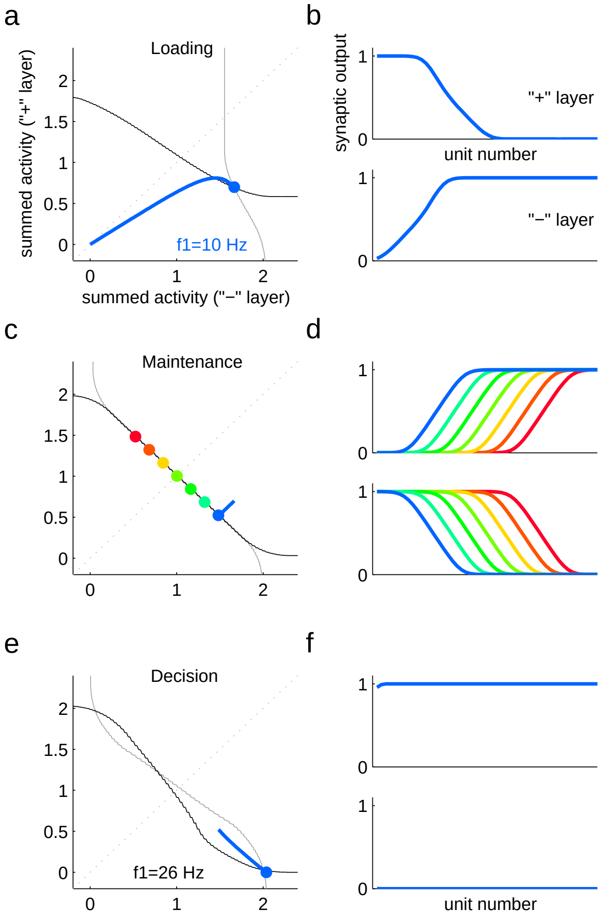

Figure S6: Loading, memory, and decision dynamics of the robust model; compare this figure with Fig.3e-g of the main text. (a) Phase space plot of the loading phase of the robust model. The grey and black curves show the nullclines of the dynamics; the blue curve a trajectory when the stimulus f1=10 Hz is loaded. (b) Synaptic output of the excitatory neurons in the (top) plus and (bottom) minus layers after the loading dynamics has reached the single stable fixed point. The x-axis (“unit number”) refers to the spatial ordering of the units as shown in Fig. S5. (c) Memory maintenance mode of the robust model. The nullclines (grey and black curves) now overlap completely. The colored dots show different stable memories. Same color code as in Fig. 1 and Fig. 4 of the main text. (d) Activities in the upper and lower layer for the different memory states. Note that the excitatory synaptic outputs actually saturate (upper part of the sigmoidal functions). (e) Decision mode of the robust model. The blue trajectory shows the decision for a memory state corresponding to f1=10 Hz and a second stimulus f2=26 Hz. (f) Activity in the upper and lower layers after the decision dynamics has taken place.

hysteresis in the i/o functions acts like a brake against any shifts of the spatial activity profile.[3] Details on this concept can be found in Koulakov et al. (2002).

3.3 Simulation results of the spiking robust model

While the tuning of the robust model was performed in the mean-field framework, the final simulation were done in a spiking network. For a correctly tuned set of parameters, the activity of a spiking neuron during the two-interval discrimination task is shown in Fig. S7a, in direct correspondence to Fig. 4a from the paper. Unlike the two-node model in the paper, even moderate *global* changes of the parameters do not destroy the ability of the robust model to perform the two-interval discrimination task. Fig. S7b and c shows that the robust model works even if *all* the recurrent excitatory weights ω_E are changed by -5% or $+5\%$, respectively. Larger global changes, however, destroy the dynamics (Fig. S7d and S7e with -10% and $+10\%$, respectively).

3.4 Mean-field equations for robust model

In the mean-field approximation, the effect of an excitatory and inhibitory node in the same layer are combined into two-dimensional i/o functions $x_E = F_E(g_I, g_E)$ and $x_I = F_I(g_I, g_E)$, see Fig. S8a, where g_I and g_E are the inhibitory and excitatory conductance inputs to the excitatory neurons, x_E is the synaptic output of the excitatory neurons, and x_I is the synaptic output of the inhibitory neurons. The computation of these i/o functions from the detailed neuron and synapse model is explained below. The synaptic output variables are described by the following dynamical equations for the plus (x) and minus (y) layers ($k = 1 \dots n$ where n is the number of units):[4]

$$
\begin{aligned}
\tau \dot{x}_{E,k} &= -x_{E,k} + F_E(-\omega_I y_{I,k}\ ,\ \sum_{j=1}^{n} \omega_E x_{E,j} + E_{x,k}) \\
\tau \dot{x}_{I,k} &= -x_{I,k} + F_I(-\omega_I y_{I,k}\ ,\ \sum_{j=1}^{n} \omega_E x_{E,j} + E_{x,k}) \\
\tau \dot{y}_{E,k} &= -y_{E,k} + F_E(-\omega_I x_{I,k}\ ,\ \sum_{j=1}^{n} \omega_E y_{E,j} + E_{y,k}) \\
\tau \dot{y}_{I,k} &= -y_{I,k} + F_I(-\omega_I x_{I,k}\ ,\ \sum_{j=1}^{n} \omega_E y_{E,j} + E_{y,k})
\end{aligned}
$$

The parameter ω_I denotes the weight of the inhibitory synapses within each unit and ω_E the weight of the recurrent excitatory synapses that connect nodes within a layer. As in the two-node model, the plus- and minus-nodes receive excitatory inputs from S2 during presentation of stimulus f1, with plus neurons in S2 projecting onto the plus nodes; and minus neurons in S2 projecting onto the minus nodes.

[3] In the dynamical algorithm, Fig. 2 in the main text, this corresponds to an undulating L-function in the maintenance state.

[4] For simplicity, we neglect the hysteretic nature of the i/o functions in the mean field description here; further below, we show how to include it.

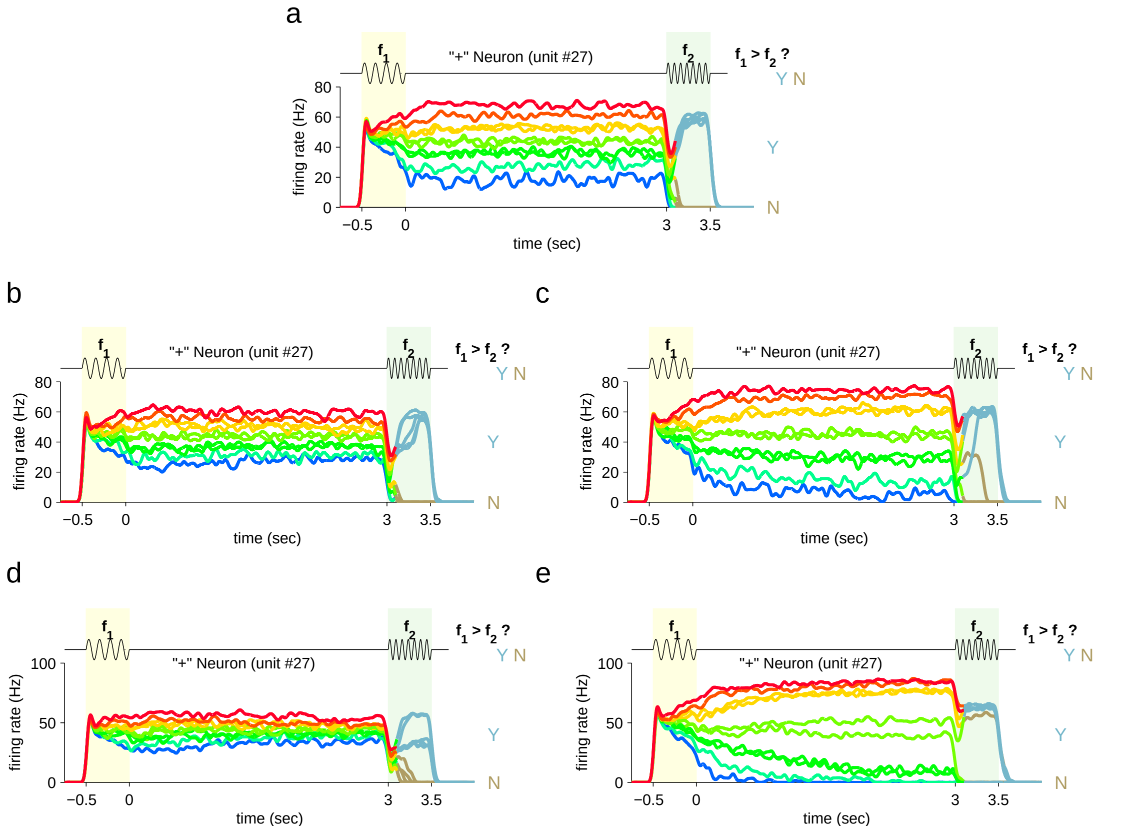

Figure S7: (a) Firing rates of simulated data for correct, fine-tuned set of parameters. (b) Firing rates with ω_E decreased by 5%. (c) Firing rates with ω_E increased by 5%. (d) Firing rates with ω_E decreased by 10%. (e) Firing rates with ω_E increased by 10%. The color code corresponds to the one chosen in Fig. 1 and 4 (main text) and Fig. S4.

During stimulus f2, the functional connectivity is switched again as suggested in Fig. 3h. Furthermore, just as in the two-node model, an external signal E controls the phases of the task. In contrast to the two-node model, however, the external control signal is now unit-dependent during the maintenance period, $E = E_c + \alpha k$; this unit-dependent input is a requirement of Koulakov et al.'s robust model. Further details on the inputs are given in Section 3.7.

3.5 Computing the hysteretic i/o function

The i/o function in the robust model combines the effect of an excitatory and inhibitory node in each unit, see Fig. S8a. The *inputs* to this combined system are

- g_E, the excitatory input conductance to the excitatory neuron.
- g_I, the inhibitory input conductance to the excitatory neuron.

While the *outputs* are:

- s_I, the synaptic output variable of the inhibitory interneuron.
- s_E, the synaptic output variable of the excitatory neuron.

In the robust model, we used a two-compartmental integrate-and-fire neuron model for the excitatory neurons with dendritic NMDA synapses. These synapses give rise to a hysteretic activity function as shown in Fig. S8b. The somatic and dendritic compartments are linked with conductance g_i. The membrane potential of the dendritic and somatic compartments, V_d and V_s, obey the following differential equations

$$\begin{aligned} C\dot{V}_d &= g_{L,d}(E_L - V_d) + g_i(V_s - V_d) \\ &\quad + a(V_d)\lambda_{E,d}g_E(t)(E_E - V_d) + \lambda_{I,d}g_I(t)(E_I - V_d) + \sigma_1\eta_1(t) \\ C\dot{V}_s &= g_{L,s}(E_L - V_s) + g_i(V_d - V_s) \\ &\quad + \lambda_{E,s}g_E(t)(E_E - V_s) + \lambda_{I,s}g_I(t)(E_I - V_s) + \sigma_2\eta_2(t) \end{aligned}$$

where synapses can now connect independently to both compartments. There are three types of synapses: GABA synapses that connect to both the dendrite and the soma; an AMPA synapse that connects to the soma only; and an NMDA synapse that connects to the dendrite only. NMDA synapses are voltage-dependent due to the Magnesium block. To account for this dependence, we follow [9] and define

$$a(V) = \frac{1}{1 + 0.3[\mathrm{Mg}]e^{-0.08V}}$$

which adds hysteresis, see Fig. S8b [9, 8]. The inhibitory and excitatory conductance inputs, $g_I(t)$ and $g_E(t)$ act on both the dendritic and somatic compartments; the constants λ are scaling factors that control the relative strength of the synapses on the dendrite and soma. A spike is emitted if the voltage in the somatic compartment exceeds the threshold V_{thresh}, after which the somatic membrane potential is reset to $V_s(t) = V_{\text{reset}}$ for a refractory time interval of length τ_{ref}.

The excitatory neuron connects with an AMPA synapse onto the inhibitory interneuron. The synaptic variable of this AMPA synapse evolves according to Eq. 8; its parameters are provided in Section 3.7. The interneuron itself is a simple integrate-and-fire neuron,

$$C\dot{V} = g_L(E_L - V) + \tilde{g}_E(t)(E_E - V) + \sigma\eta(t)$$

where $\tilde{g}_E(t)$ is the AMPA conductance input from the excitatory neuron. For simplicity, the parameters of this neuron were chosen such that it acts like an identity transform, i.e., it converts the (excitatory) synaptic output of the excitatory neuron into an inhibitory synaptic output of equal size, the parameters of this neuron are again provided in Section 3.7.

The outputs of this combined system are the synaptic outputs of the excitatory and inhibitory neurons (Fig. S8a), i.e., the spike trains of these neurons put through Eq. 8. Since the network contains three different synapse types (AMPA, NMDA, and GABA) with different sets of parameters for the synaptic variables, there are technically three distinct output variables of the combined system. However, by setting the ratio of saturation parameters of the NMDA and AMPA synapses equal to the ratio of their time constants, these synapses assume the same average values once the model has settled into a fixed point (stationary state). Since, just as in the two-node model, the fixed points of the network really matter while details of the transient trajectories between fixed points do not, we can describe both AMPA and NMDA synapses by the same synaptic output variable, s_E.

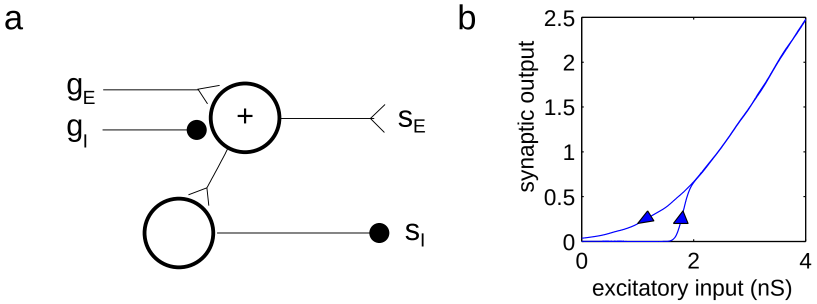

Figure S8: (a) We combine the excitatory and inhibitory neuron into an i/o function with conductance inputs g_E and g_I and synaptic outputs s_E and s_I. (b) Due to the NMDA synapses in the excitatory neurons, the i/o function of the combined system (a) is hysteretic. Shown is a synaptic output variable as a function of the excitatory conductance input g_E (and in the absence of an inhibitory conductance input) The parameters for inhibitory interneurons (see section 3.7) were chosen such that their net input-ouput function is very close to the identity function. Thus, the ouput axis in panel b represents both the magnitude of s_E and s_I.

For a given set of neuron and synapse parameters, we can now compute the neuron's i/o functions $s_E = F_E(g_I, g_E)$ and $s_I = F_I(g_I, g_E)$ numerically from the differential equations of the underlying neurons and synapses. Due to the hysteretic nature of the NMDA synapses, these functions are not unique; rather, there are two separate functions $F_E(\cdot,\cdot)$, and two separate functions $F_I(\cdot,\cdot)$ (four functions altogether), with the choice within each type of function depending on whether the NMDA synapse is in the up- or down-state. In principle, these up- and down-state can be modeled in the mean-field model as well: one needs to introduce two thresholds to describe the on- and offset of the Mg-block (corresponding to the points in Fig. S8 at which the curves split) and then dynamically keep track of whether the combined system is in the up- or down-state.

A detailed description of the motivation and choices behind the model's structure and parameter values will be provided elsewhere (Machens and Brody, in preparation). Briefly, the design of the excitatory neuron was guided by the goal of having hysteretic i/o functions, the inhibitory interneurons were designed as "dummy" neurons that simply transform an excitatory synaptic output into an inhibitory synaptic output. In Section 3.7, we provide all the parameters necessary to compute the i/o function.

3.6 Implementing the robust model in a network of spiking neurons

For Fig. S7, we simulated a network with $N = 250$ spiking neurons per node. In this network, all excitatory neurons receive an excitatory conductance input g_E whose value depended on the phase of the task, the position (unit number k) of the neuron, and whether the neuron is part of the plus or minus layer. These values are provided in Section 3.7. Together with the recurrent inputs from within the same layer, the total excitatory conductance input for an excitatory neuron from the i-th unit is

$$g_{E,i}(t) = \frac{1}{N}\sum_{j=1}^{n} \omega_E s_{E,j}(t) + g_{E,i}^{\text{ext}}(t) \qquad (13)$$

All the excitatory neurons also receives an inhibitory conductance input $g_I(t)$ from all the neurons of the opposite node; for the *i*-th neuron, this conductance input is given by summing over all of its synaptic inputs,

$$g_{I,i}(t) = \frac{1}{N}\sum_{j=1}^{2N} \omega_I s_{I,j}(t) + g_{I,i}^{\text{ext}}(t) \tag{14}$$

where $g_{I,i}^{\text{ext}}(t)$ is a constant inhibitory conductance input that is only applied during loading.

The fraction $1/N$ in front of these equations ensures that the model scales correctly with the number of neurons per node.

3.7 Full parameters to reproduce the robust model

(1) **Simulation Method:**
ODEs were simulated using the Euler method with $\Delta t = 0.1$ msec.

(2) **Number of units:**
$N = 49$

(3) **Number of neurons per population:**
$M = 250$

(4) **Excitatory neurons:**
$C_d = 0.01$nF, $C_s = 0.4$nF, $g_{L,d} = 1$nS, $g_{L,s} = 20$nS, $E_L = -60$mV, $\sigma_d = 0.00003$, $\sigma_s = 0.3$, $V_r = -60$mV, $V_\theta = -55$mV, $\tau_{\text{ref}} = 2$ms, $g_i = 0.6$nS, $E_I = -75$mV, $E_E = 10$mV, Mg $= 0.5$.

(5) **Feedforward synapse from excitatory to inhibitory neurons:**

- AMPA: $\tau_E = 25$ms, $s_{E,\text{max}} = 500$.

(6) **Inhibitory interneurons:**
$C = 0.5$nF, $g_L = 10$nS, $E_L = -59$mV, $\sigma = 0.3$, $V_r = -59$mV, $V_\theta = -55$mV, $\tau_{\text{ref}} = 2$ms, $E_I = -75$mV, $E_E = 10$mV. Conductance inputs: $\tilde{g}_E = 0.7\text{nS} + 0.0014\langle s_F \rangle$ where $\langle s_F \rangle$ is the average synaptic output variable (using the synaptic parameters from above [5]) from the excitatory node.

(7) **Recurrent synapses:**

- GABA: $\tau_I = 25$ms, $s_{I,\text{max}} = 5$.
- AMPA: $\tau_E = 25$ms, $s_{E,\text{max}} = 1$.
- NMDA: $\tau_E = 100$ms, $s_{E,\text{max}} = 4$.

(8) **Synaptic weights:**
Connectivity within each unit:

- GABA synapses: $\omega_I = 0.034$
- AMPA/NMDA synapses: $\omega_E = 0.0001$

(9) **External inputs onto excitatory neurons:**
Inputs during loading/maintenance/decision:

- Loading: $f_1 = (10, 14, 18, 22, 26, 30, 34)$ Hz, $t = -0.5 \ldots 0$ sec
$g^{\text{ext}}_{E,x} = 4.2 + (-0.066, -0.044, -0.022, 0, 0.022, 0.044, 0.066)$ nS
$g^{\text{ext}}_{E,y} = 4.2 + (0.066, 0.044, 0.022, 0, -0.022, -0.044, -0.066)$ nS
$g^{\text{ext}}_{I,d} = 0.1$ nS.
- Maintenance: for unit $k = 1 \ldots n$
$g^{\text{ext}}_{E,x,k} g_{E,y,k} = 2.2 + (k - 0.5(n+1)) * 1.885$ nS;
- Decision: $f_2 = (10, 14, 18, 22, 26, 30, 34)$ Hz, $t = 3 \ldots 3.5$ sec
$g^{\text{ext}}_{E,x} = 0.3 + (-0.3, -0.2, -0.05, 0, 0.05, 0.2, 0.3)$ nS
$g^{\text{ext}}_{E,y} = 0.3 + (0.3, 0.2, 0.05, 0, -0.05, -0.2, -0.3)$ nS

(10) Scaling factors:

- Inhibition/Soma: $\lambda_{I,s} = 0.112$
- Inhibition/Dendrite: $\lambda_{I,d} = 1$
- Excitation/Soma: $\lambda_{E,s} = 0.4$
- Excitation/Dendrite: $\lambda_{E,d} = 12.5$

4 Change in connection sign between area S2 and frontal lobe neurons

In this modeling paper, we have focused on neuronal responses that represent characteristics found over populations of recorded neurons. We do note, however, that taken individually, different experimentally recorded neurons can have response properties that are quite different from each other [2]. The robust model of Section 3 displays only part of the heterogeneity found experimentally; whether the remaining response heterogeneity plays a functional role in two-interval discrimination remains unknown.

Despite the overall heterogeneity, one of the characteristics that is robustly found across individual neurons of the population, and used in the models presented here, is that frontal neurons with "plus" sign activity during the memory maintenance period tend to fire more for "Yes" decisions during the comparison/decision period; while neurons defined as "minus" within the memory period fire more for "No" decisions (Fig. 1c,d; Brody, Romo et al., in preparation). Because high f2 stimuli are more likely to lead to "No" decisions, this means that the stimuli presented during the loading period that drive frontal neurons to higher firing rates during the memory period nevertheless drive the same neurons to *lower* firing rates when presented during the comparison/decision period (Figs. S9c, S9d). Conversely, stimuli presented during the loading period that drive frontal neurons to lower firing rates during the memory period nevertheless drive frontal neurons to *higher* firing rates when presented during comparison/decision. This inversion of the sign of the stimulus→response mapping does not occur for neurons in area S2 (Figs. S9e, S9f), and therefore must occur at some point in the transmission of signals between S2 and frontal lobes (Fig. 3h).

(Note that the inversion is not required by the structure of the task: for example, with the same task, the empirical finding could have been that "plus" neurons fired the most for "No" decisions.)

The functional inversion of connection sign between S2 and frontal lobes, implied by the empirical data, is one of two major elements that are required by our model yet are not addressed by it. Instead, the two elements are assumed to exist and to be external to the model. First, we have not specified precisely how the sign switch between S2 and frontal lobes is carried out. The circuit of Fig. 3h serves to illustrate that integrate-and-fire neurons could in fact carry out such a switch. But the model does not distinguish between different approaches that would lead to the same functional sign inversion result. Second, we have not specified the source of the external excitation signal, E, which in the model controls the dynamical mode of the system (Fig. 3).

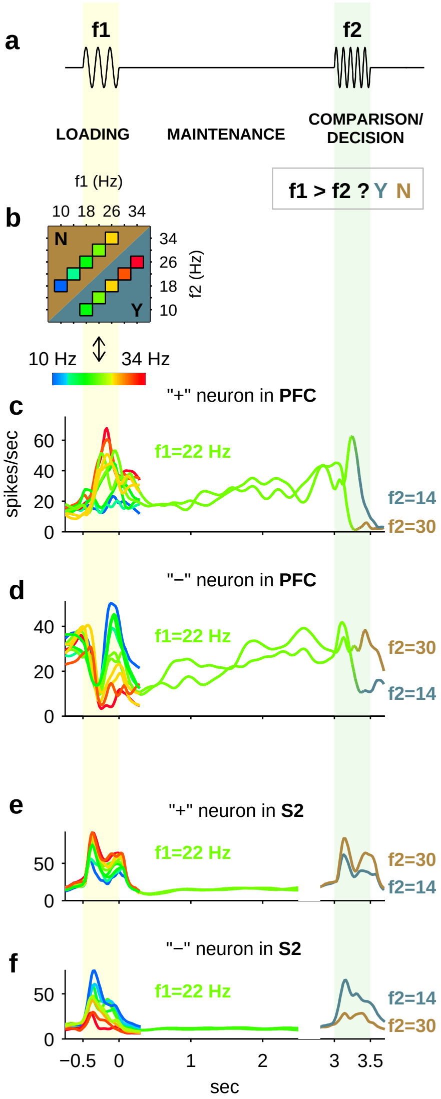

Figure S9: Inversion of the sign of stimulus-dependency of frontal neurons. This figure reproduces Fig. 1 of the main text; but here, after t=0.3 sec we show for clarity only those trials in which stimulus f1 was 22 Hz, and stimulus f2 was either 14 or 30 Hz. **b:** The "plus" neuron is excited by high stimuli during loading, but *inhibited* by high stimuli during comparison/decision. **c:** Conversely, the "minus" neuron is excited by low stimuli during loading, but inhibited by the low stimulus during comparison/decision. **d,e:** In contrast, neurons in area S2 have equal response signs during loading and during the first part of comparison/decision.

References

[1] C. D. Brody, R. Romo, A. Kepecs. Basic mechanisms for graded persistent activity: discrete attractors, continuous attractors, and dynamic representations. *Curr. Opin. Neurobiol.* 13:204–211, 2003.

[2] C. D. Brody, A. Hernandez, A. Zainos, R. Romo. Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. *Cereb. Cortex* 13:1196-1207, 2003.

[3] N. Brunel and X.-J. Wang. Effects of neuromodulation in a cortical network model of working memory dominated by recurrent inhibition. *J. Comput. Neurosci.* 11:63–85, 2001.

[4] A. Destexhe, T. Bal, D. A. McCormick, T. J. Sejnowski. Ionic mechanisms underlying synchronized oscillations and propagating waves in a model of ferret thalamic slices. *J. Neurophysiol.* 76:2049–2070, 1996.

[5] B. Ermentrout. Reduction of conductance based models with slow synapses to neural nets. *Neural Comp.* 6:679–95, 1994.

[6] J. Hertz, A. Lechner, M. Ahmadi. Mean-field methods for cortical network dynamics. *arXiv:q-bio.NC* 0402023.

[7] G.R. Holt and C. Koch. Shunting inhibition does not act divisively on firing rates. *Neural Comp.* 9:1001-1013, 1997.

[8] A.A. Koulakov, S. Raghavachari, A. Kepecs, and J.E. Lisman. Model for a robust neural integrator. *Nat. Neurosci.* 5:775–82, 2002.

[9] J.E. Lisman, J.-M. Fellous, and X.-J. Wang. A role for NMDA channels in working memory. *Nat. Neurosci.* 1:273–275, 1998.

[10] P. Miller, C.D. Brody, R. Romo, and X.-J. Wang. Recurrent network model of somatosensory parametric working memory in the prefrontal cortex. *Curr. Opin. Neurobiol.* 13:204–211, 2003.

[11] D.J. Pinto, J.C. Brumberg, D.J. Simons, G.B. Ermentrout. A quantitative population model of whisker barrels: re-examining the Wilson-Cowan equations. *J. Comput. Neurosci.* 3:247-264, 1996.

[12] H.S. Seung, D.D. Lee, B.Y. Reis, and D.W. Tank. Stability of the memory of eye position in a recurrent network of conductance-based model neurons. *Neuron* 26:259–71, 2000.

[13] H. S. Seung, D. D. Lee, B. Y. Reis, D. W. Tank. The autapse: a simple illustration of short-term analog memory storage by tuned synaptic feedback. *J. Comput. Neurosci.* 9:171–185, 2000.