

1 Simple Framework for Constructing Functional Spiking Recurrent 2 Neural Networks

3 Robert Kim ^{1,2,3 *}, Yinghao Li ¹, Terrence J. Sejnowski ^{1,4,5 *}

4 ¹ Computational Neurobiology Laboratory, Salk Institute for Biological Studies, La Jolla, CA
5 92037, USA

6 ² Neurosciences Graduate Program, University of California San Diego, La Jolla, CA 92093, USA
7 ³ Medical Scientist Training Program, University of California San Diego, La Jolla, CA 92093,
8 USA

9 ⁴ Institute for Neural Computation, University of California San Diego, La Jolla, CA 92093, USA
10 ⁵ Division of Biological Sciences, University of California San Diego, La Jolla, CA 92093, USA

11 * Correspondence: rkim@salk.edu (R.K.), terry@salk.edu (T.J.S.)

12 Abstract

13 Cortical microcircuits exhibit complex recurrent architectures that possess dynamically rich prop-
14 erties. The neurons that make up these microcircuits communicate mainly via discrete spikes,
15 and it is not clear how spikes give rise to dynamics that can be used to perform computationally
16 challenging tasks. In contrast, continuous models of rate-coding neurons can be trained to perform
17 complex tasks. Here, we present a simple framework to construct biologically realistic spiking re-
18 current neural networks (RNNs) capable of learning a wide range of tasks. Our framework involves
19 training a continuous-variable rate RNN with important biophysical constraints and transferring
20 the learned dynamics and constraints to a spiking RNN in a one-to-one manner. We validate
21 our framework on several cognitive task paradigms to replicate previously observed experimental
22 results. We also demonstrate different ways to exploit the biological features of our models to
23 elucidate neural mechanisms underlying cognitive functions.

24 Introduction

25 Understanding how seemingly irregular and chaotic neural activity facilitates information process-
26 ing and supports complex behavior is a major challenge in neuroscience. Previous studies have
27 employed models based on recurrent neural networks (RNNs) of continuous-variable rate units to
28 characterize network dynamics underlying neural computations [1–6].

29 Methods commonly used to train rate networks to perform cognitive tasks can be largely clas-
30 sified into three categories: recursive least square (RLS)-based, gradient-based, and reward-based
31 algorithms. The First-Order Reduced and Controlled Error (FORCE) algorithm, which utilizes
32 RLS, has been widely used to train RNNs to produce complex output signals [2] and to reproduce
33 experimental results [3, 7, 8]. Gradient descent-based methods, including Hessian-free methods,
34 have been also successfully applied to train rate networks in a supervised manner and to replicate
35 the computational dynamics observed in networks from behaving animals [4, 9, 10]. Unlike the
36 previous two categories (i.e. RLS-based and gradient-based algorithms), reward-based learning
37 methods are more biologically plausible and have been shown to be as effective in training rate
38 RNNs as the supervised learning methods [11, 12]. Even though these models have been vital
39 in uncovering previously unknown computational mechanisms, continuous rate networks do not
40 incorporate basic biophysical constraints such as the spiking nature of biological neurons.

41 Training spiking network models where units communicate with one another via discrete spikes
42 is more difficult than training continuous rate networks. The non-differentiable nature of spike sig-
43 nals prevents the use of gradient descent-based methods to train spiking networks directly, although
44 several differentiable models have been proposed [13, 14]. Due to this challenge, FORCE-based
45 learning algorithms have been most commonly used to train spiking recurrent networks. While
46 recent advances have successfully modified and applied FORCE training to construct functional
47 spike RNNs [5, 15–18], FORCE training is computationally inefficient and unstable when connec-
48 tivity constraints, including separate populations for excitatory and inhibitory populations (Dale’s
49 principle) and sparse connectivity patterns, are imposed [16]. Consistent with these limitations,
50 there are only few examples of biologically realistic spiking RNN models trained via the FORCE
51 algorithm. In these examples, moderately sparse spiking networks that obey Dale’s principle were
52 trained to produce simple oscillatory output signals [16, 18].

53 Here we present a computational framework for constructing functional spiking neural networks
54 that can easily incorporate biophysical constraints. Our method involves training a continuous-
55 variable rate RNN using a gradient descent-based method, and transferring the learned dynamics

56 of the rate network along with the constraints to a spiking network model in a one-to-one manner.
57 The gradient descent learning algorithm allowed us to easily optimize many parameters including
58 the connectivity weights of the network and the synaptic decay time constant for each unit. In
59 addition, Dale’s principle and additional connectivity patterns can be enforced without signifi-
60 cantly affecting computational efficiency and network stability using the recurrent weight matrix
61 parametrization method proposed by Song et al. [10]. We demonstrate the flexibility and the
62 versatility of our framework by constructing spiking networks to perform several tasks ranging
63 from a simple Go-NoGo task to a more complex task that requires input integration and working
64 memory. Furthermore, we demonstrate how biologically realistic spiking RNNs constructed from
65 our framework allow us to utilize both rate- and spike-based measures to better understand the
66 network dynamics underlying cognitive behavior.

67 Results

68 Here we provide a brief overview of the two types of recurrent neural networks (RNNs) that we
69 employed throughout this study (more details in Methods): continuous-variable firing rate RNNs
70 and spiking RNNs. The continuous-variable rate network model consisted of N rate units whose
71 firing rates were estimated via a nonlinear input-output transfer function [1, 2]. The model was
72 governed by the following set of equations:

$$\tau_i \frac{dx_i}{dt} = -x_i + \sum_{j=1}^N w_{ij}^{rate} r_j^{rate} + I_{ext} \quad (1)$$

$$r_i^{rate} = \phi(x_i) \quad (2)$$

73 where τ_i is the synaptic decay time constant for unit i , x_i is the synaptic current variable for unit
74 i , w_{ij}^{rate} is the synaptic strength from unit j to unit i , and I_{ext} is the external current input to unit
75 i . The firing rate of unit i (r_i^{rate}) is given by applying a nonlinear transfer function ($\phi(\cdot)$) to the
76 synaptic current variable. In order to make the network biologically realistic, we chose the transfer
77 function to be a non-negative saturating function (standard sigmoid function) and parametrized
78 the connectivity matrix ($w_{ij}^{rate} \in W^{rate}$) to enforce Dale’s principle and additional connectivity
79 constraints (see Methods).

80 The second RNN model that we considered was a network composed of N spiking units.
81 Throughout this study, we focused on networks of leaky integrate-and-fire (LIF) units whose mem-

82 brane voltage dynamics were given by:

$$\tau_m \frac{dv_i}{dt} = -v_i + \sum_{j=1}^N w_{ij}^{spk} r_j^{spk} + I_{ext} \quad (3)$$

83 where τ_m is the membrane time constant (set to 10 ms throughout this study), v_i is the membrane
84 voltage of unit i , w_{ij}^{spk} is the synaptic strength from unit j to unit i , r_j^{spk} represents the synaptic
85 filtering of the spike train of unit j , and I_{ext} is the external current source. The discrete nature
86 of r_j^{spk} (see Methods) has posed a major challenge for directly training spiking networks using
87 gradient-based supervised learning. Even though the main results presented here are based on LIF
88 networks, our method can be generalized to quadratic integrate-and-fire (QIF) networks with only
89 few minor changes to the model parameters (see Supplementary Table 1, Supplementary Notes,
90 Supplementary Fig. 4).

91 For each example shown in the study, we present the results obtained from a representative
92 trained model, but all the results were robust and reproducible from multiple trained networks
93 with random initialization conditions. Continuous rate network training was implemented using
94 the open-source software library TensorFlow in Python, while LIF/QIF network simulations along
95 with the rest of the analyses were performed in MATLAB.

96 **Transfer Learning from Continuous Rate Networks to Spiking Networks.** In order to
97 construct functional spiking networks that perform cognitive tasks, we developed a simple pro-
98 cedure that directly maps dynamics of a trained continuous rate RNN to a spiking RNN in a
99 one-to-one manner. The first step of our method involves training a continuous rate RNN to per-
100 form a task. Throughout this study, we used a gradient-descent supervised method, known as
101 Backpropagation Through Time (BPTT), to train rate RNNs to produce target signals associated
102 with a specific task [19].

103 The units in a rate RNN are sparsely connected via W^{rate} and receive a task-specific input
104 signal through weights (W_{in}) drawn from a normal distribution with zero mean and unit variance.
105 The network output (o^{rate}) is then computed using a set of linear readout weights:

$$o^{rate}(t) = W_{out}^{rate} \cdot \mathbf{r}^{rate}(t) \quad (4)$$

106 where W_{out}^{rate} is the readout weights and $\mathbf{r}^{rate}(t)$ is the firing rate estimates from all the units in
107 the network at time t . The recurrent weight matrix (W^{rate}), the readout weights (W_{out}^{rate}), and the
108 synaptic decay time constants (τ) are optimized during training, while the input weight matrix
109 (W_{in}) stays fixed (see Methods).

110 Once the rate network model is trained, the three sets of the weight matrices (W_{in} , W^{rate} , and
111 W_{out}^{rate}) along with the tuned synaptic time constants (τ) are transferred to a network of LIF spiking
112 units. The spiking RNN is initialized to have the same topology as the rate RNN. The input weight
113 matrix and the synaptic time constants are simply transferred without any modification, but the
114 recurrent connectivity and the readout weights need to be scaled by a constant factor (λ) in order
115 to account for the difference in the firing rate scales between the rate model and the spiking model
116 (see Methods).

117 In Fig. 1, we trained a small continuous rate network of $N = 200$ units (162 excitatory and 38
118 inhibitory units) on a simple task modeled after a Go-NoGo task to demonstrate our framework.
119 Using BPTT, the network was trained to produce a positive mean population activity approaching
120 +1 after a brief input pulse (Fig. 1A). For a trial without an input pulse, the network was trained
121 to maintain its output close to zero. The trained rate RNN performed the task correctly on all
122 test trials with a mean synaptic decay time constant of 28.2 ± 9.4 ms (Fig. 1B and 1C).

123 Next, we directly transferred the input weight matrix (W_{in}) and the optimized synaptic time
124 constants to a network of LIF units. The connectivity matrix (W^{rate}) and the readout weights
125 (W_{out}^{rate}) were scaled by a factor of $\lambda = 0.02$ (see Methods on how it was computed) and transferred
126 to the spiking network. When the weights were not scaled (i.e. $\lambda = 1$), the spiking network could
127 not perform the task (output signals for both Go and NoGo trials converged) and produced largely
128 fluctuating signals (Fig. 1D top). With an appropriate value for λ , the LIF network performed
129 the task with the same accuracy as the rate network (Fig. 1D bottom), and the LIF units fired at
130 rates similar to the “rates” of the continuous network units (Fig. 1E).

131 Our framework also allows seamless integration of additional functional connectivity constraints.
132 For example, a common cortical microcircuitry motif where somatostatin-expressing interneurons
133 inhibit both pyramidal and parvalbumin-positive neurons can be easily implemented in our frame-
134 work (see Methods and Supplementary Fig. 1). In addition, Dale’s principle is not required for our
135 framework (Supplementary Fig. 2).

136 **Excitatory and inhibitory dynamics during autonomous oscillatory network activities.**

137 Next, we tested our framework on an autonomous oscillation task where a rate RNN was first
138 trained to produce a periodic output signal in the absence of external input signals (Fig. 2A).
139 The target signal used to train the rate network was a simple 1 Hz sine wave. The rate network
140 composed of 98 excitatory units and 102 inhibitory units was successfully trained to produce the
141 target sinusoidal signal autonomously (Fig. 2B top). An LIF model was endowed with the same

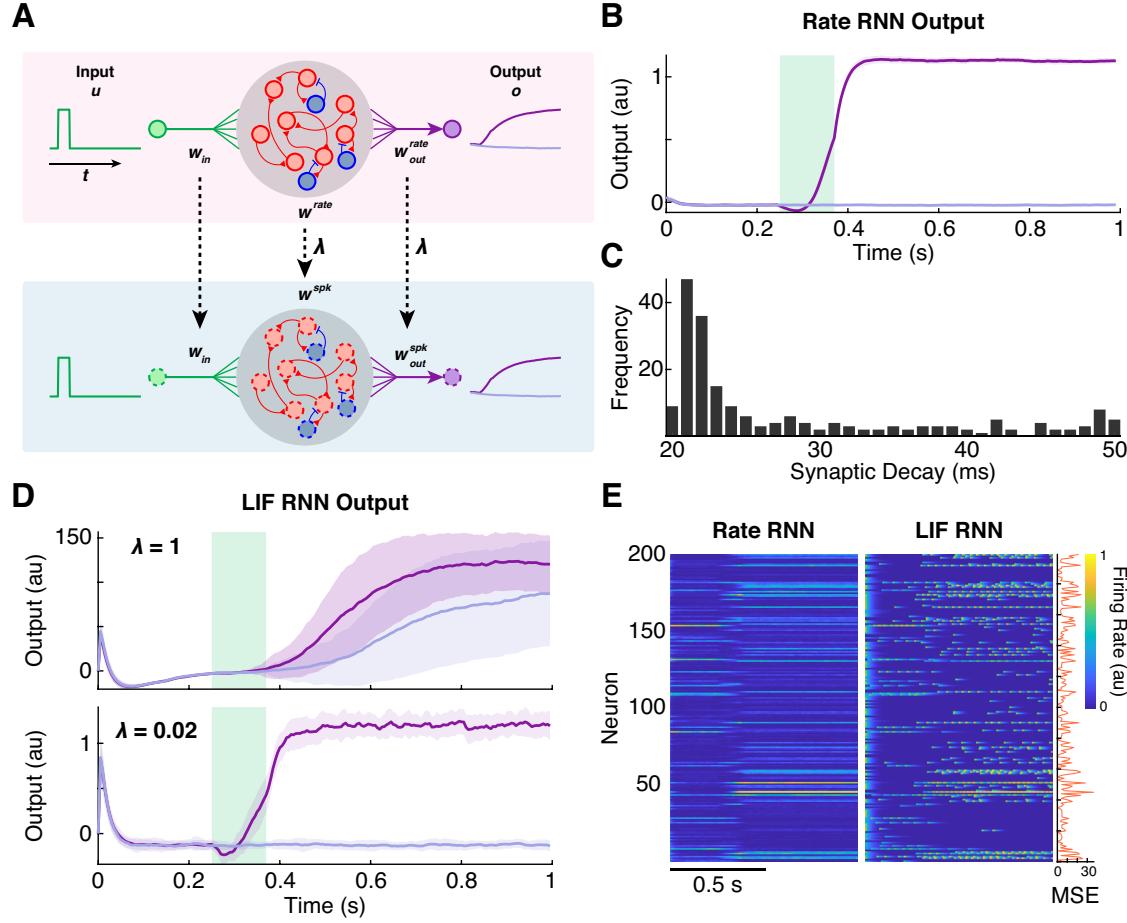


Fig. 1 | Transfer learning from rate RNNs to spiking RNNs. **A.** Schematic diagram illustrating transfer learning from a continuous rate RNN model (top) to a spiking RNN model (bottom). A rate network with excitatory (red circles with a solid outline) and inhibitory (blue circles with a solid outline) units was trained to perform a Go-NoGo task. The optimized synaptic decay time constants (τ) along with the weight parameters (W_{in} , W^{rate} , and W^{rate}_{out}) were transferred to a spiking network with LIF units (red and blue circles with a dashed outline). The connectivity and the readout weights were scaled by a constant factor, λ . **B.** Trained rate RNN performance on the Go-NoGo task. The mean \pm SD output signals from 100 Go trials (dark purple) and from 100 NoGo trials (light purple) are shown. The green box represents the input stimulus given for the Go trials. **C.** Distribution of the tuned synaptic decay time constants (τ). Mean \pm SD, 28.2 ± 9.4 ms. **D.** LIF RNN performance on the Go-NoGo task without scaling ($\lambda = 1$; top) and with appropriate scaling ($\lambda = 0.02$; bottom). Mean \pm SD over 100 Go and 100 NoGo trials. **E.** Comparison of the time-varying rates from the trained rate network (left) and the spiking model (right). A single Go trial was used to extract the rates from the rate RNN and the firing rates scaled by λ from the LIF network. The mean squared error (MSE) was computed to quantify the difference in firing rate between the two network models (orange line).

142 dynamical properties after the weight matrices and the synaptic time constants were transferred.

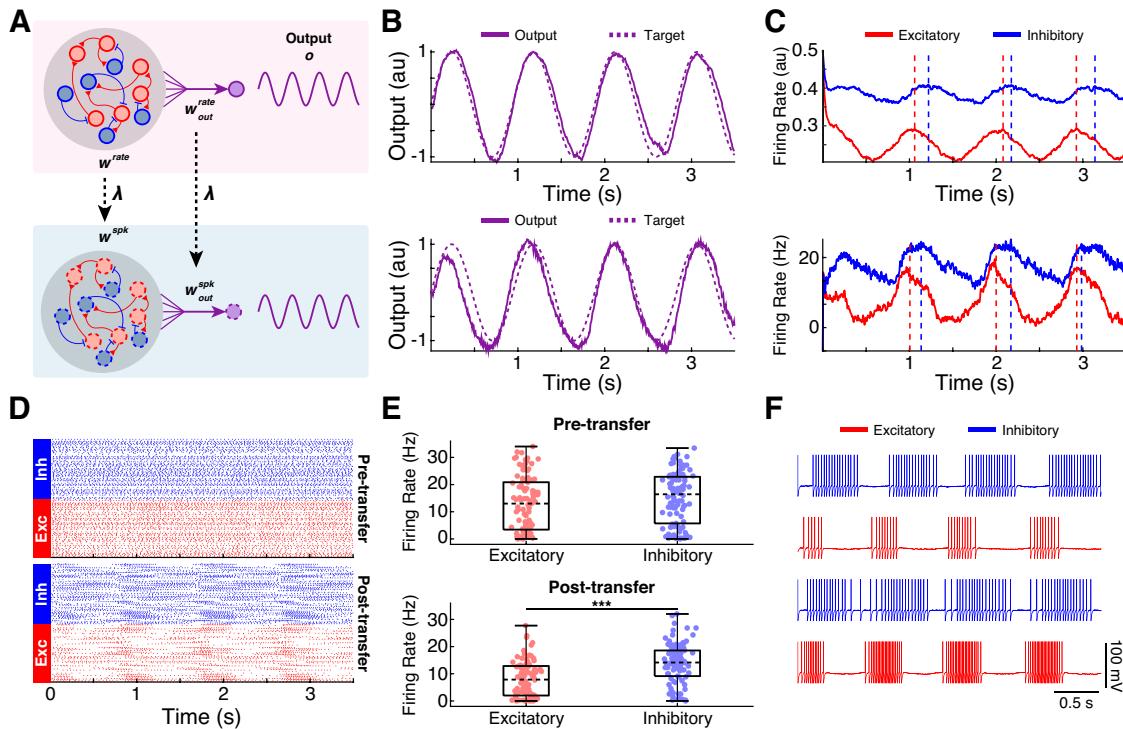


Fig. 2 | Autonomous oscillation task. **A.** Schematic outlining the autonomous oscillation task. A rate RNN was trained to produce a slow sinusoidal output signal without any external input signals (top). A network of LIF units was constructed to perform the same task using the transfer learning method (bottom). **B.** Comparison of the output signals from the trained rate model (top) and the LIF model (bottom). **C.** Comparison of the average excitatory (red) and inhibitory (blue) firing rates from the rate model (top) and the LIF model (bottom). The red and blue vertical dashed lines represent the local maxima for the mean excitatory and inhibitory firing rates, respectively. **D.** Raster plots of spiking activities from the LIF network before transfer learning (top) and after transfer learning (bottom). **E.** Average firing rates from the excitatory and inhibitory units from the LIF network before transfer learning (top) and after transfer learning (bottom). The average firing rates between the two populations did not differ significantly before transfer learning. The asterisks (****) indicate a significant difference at $p < 0.001$ (two-tailed Student's t-test). Box plot central lines, median; bottom and top edges, lower and upper quartiles. **F.** Example membrane voltage tracings from two excitatory and two inhibitory units in the LIF network model (post-transfer learning).

143 The spiking network produced and maintained the same sinusoidal target signal autonomously
 144 (Fig. 2B bottom).

145 In both networks (rate and LIF networks), inhibition closely tracked excitation with a temporal
 146 delay, as revealed in the average rate signals of the two populations in each network (Fig. 2C).
 147 These findings suggest that both networks operate in an excitation-inhibition balanced regime and
 148 are aligned with the previous experimental results in which excitation followed by inhibition was

149 shown to provide a narrow window for sensory integration [20–23].

150 Next, we investigated the effect of transfer learning on the network spiking activity. Prior to
151 transfer of the weights, the spiking units connected via the initial random, sparse (10% sparsity)
152 weights fired at high rates continuously (Fig. 2D top and 2E top). After transfer learning, the
153 units fired in a more structured manner with the excitatory units and the inhibitory units firing on
154 average at 8.2 Hz and 13.7 Hz, respectively (Fig. 2D bottom and 2E bottom). The post-transfer
155 learning LIF units also exhibited diverse patterns of firing activities where many excitatory units
156 fired during the windows provided by delayed inhibition (Fig. 2F).

157 Our framework can be also used to construct LIF networks to produce sinusoidal signals with
158 faster frequencies and more complex signals (Supplementary Fig. 3).

159 **Rate dynamics and mixed selectivity during context-dependent input integration.**
160 The tasks considered so far did not require complex cognitive computations. In this section,
161 we consider a more complex task modeled after the context-dependent sensory integration task
162 employed by Mante et al. [4]. Briefly, Mante et al. [4] trained rhesus monkeys to integrate inputs
163 from one sensory modality (dominant color or dominant motion of randomly moving dots) while
164 ignoring inputs from the other modality. A contextual cue was also given to instruct the monkeys
165 which sensory modality they should attend to. The task required the monkeys to utilize flexible
166 computations as the same modality can be either relevant or irrelevant depending on the contextual
167 cue. Previous works have successfully trained continuous rate RNNs to perform a simplified version
168 of the task and replicated the neural dynamics present in the experimental data [4, 10, 12]. Using
169 our framework, we constructed the first spiking RNN model to our knowledge that can perform
170 the task and capture the dynamics observed in the experimental data.

171 For the task paradigm, we adopted a similar design as the one used by the previous modeling
172 studies [4, 10, 12]. A network of recurrently connected units received two streams of noisy input
173 signals along with a constant-valued signal that encoded the contextual cue (Fig. 3A; see Methods).
174 To simulate a noisy sensory input signal, a random Gaussian time-series signal with zero mean and
175 unit variance was first generated. Each input signal was then shifted by a positive or negative
176 constant (“offset”) to encode evidence toward the (+) or (-) choice, respectively (see Methods).
177 Therefore, the offset value determined how much evidence for the specific choice was represented
178 in the noisy input signal. The network was trained to produce an output signal approaching +1
179 (or -1) if the cued input signal had a positive (or negative) mean (Fig. 3A). For example, if the
180 cued input signal was generated using a positive offset value, then the network should produce an

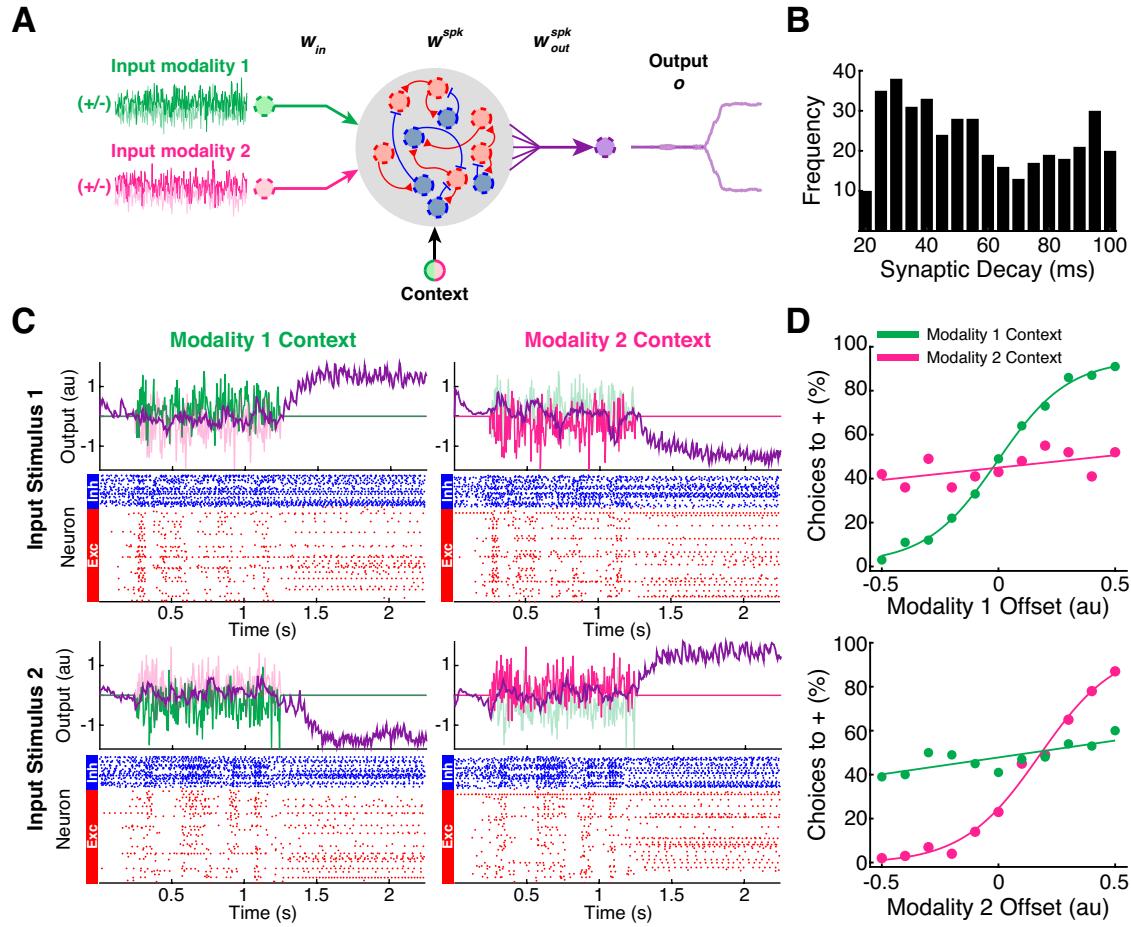


Fig. 3 | Biologically realistic spiking network performing a context-dependent input integration task. **A.** Diagram illustrating the task paradigm modeled after the context-dependent task used by Mante et al. [4]. Two streams of noisy input signals (green and magenta lines) along with a context signal were delivered to the LIF network. The network was trained to integrate and determine if the mean of the cued input signal (i.e. cued offset value) was positive (“+” choice) or negative (“-” choice). **B.** Distribution of the optimized synaptic decay time constants (τ). Mean \pm SD, 51.0 ± 25.0 ms. **C.** Example output responses and spike raster plots from the LIF network model for two different input stimuli (rows) and two contexts (columns). The network successfully integrated and responded to the cued modality (dark green and dark magenta lines). The noisy input signals are scaled by 0.5 vertically for better visualization of the network responses (purple lines). **D.** Psychometric curves showing the percentage of trials where the LIF network indicated “+” choice as a function of the modality 1 offset values (top) and modality 2 offset values (bottom).

181 output that approaches +1 regardless of the mean of the irrelevant input signal.

182 A network of 400 rate units (299 excitatory and 101 inhibitory units) was successfully trained
183 to perform the task. Unlike the simple Go-NoGo task (Fig. 1C), the integration task required

184 more units with slow synaptic decay time constants, as evidenced by the bimodal distribution of
185 the optimized time constants (Fig. 3B). This is consistent with recent experimental results where
186 neurons with long timescales played an important role in integration and processing of accumulated
187 evidence [24].

188 Next, the dynamics of the trained rate RNN were transferred to a network of LIF units. The
189 spiking network performed the same task equally well (Fig. 3C). The psychometric curves of the
190 spiking network further confirmed that the network could indeed integrate the relevant input
191 modality, while successfully ignoring the irrelevant modality (Fig. 3D). In other words, the network
192 behavior was strongly dependent on the cued modality offset values, while the uncued modality
193 offset values did not affect the network behavior. We also transferred the rate network dynamics
194 to a network of QIF units and obtained similar results (Supplementary Fig. 4).

195 After verifying that the spiking RNN could perform the task reliably, we investigated whether
196 the population dynamics underlying the spiking network were similar to the dynamics observed
197 in the group of neurons recorded by Mante et al. [4]. Consistent with the experimental results,
198 individual LIF units displayed mixed representation of the four task variables (modality 1, modality
199 2, network choice, and context; Fig. 4A). To further characterize the mixed representation, we
200 performed the multivariate linear regression and the targeted dimensionality reduction techniques
201 developed by Mante et al. [4] (see Methods). The de-noised regression correlation coefficients
202 computed across all the units in the network revealed that the individual units encoded multiple
203 task variables (Fig. 4B). More importantly, the network did not contain any distinct subgroups
204 that specialized to represent the individual task variables, as indicated by the absence of clusters
205 in the coefficients in Fig. 4B. This was also the case for the network of neurons recorded from the
206 monkeys performing the task [4]. The targeted dimensionality reduction method applied to the
207 binned spike data from the LIF network displayed the characteristic line attractor dynamics in
208 the state space (Fig. 4C). The population responses formed arc-like trajectories along the choice
209 axis, and the amplitude values of the trajectories were correlated with the offset values (compare
210 Fig. 4C to Fig. 2 from Mante et al. [4]).

211 **Working memory and neuronal synchronization modulated by inhibitory units.** While
212 the spiking network that we constructed in the previous section reproduced the rate dynamics
213 manifested by recorded neurons, our spiking models provide additional information that can be
214 explored. For instance, the spiking nature and the separate excitatory and inhibitory populations
215 of our RNNs allow us to investigate the functional role of inhibitory units in governing local neu-

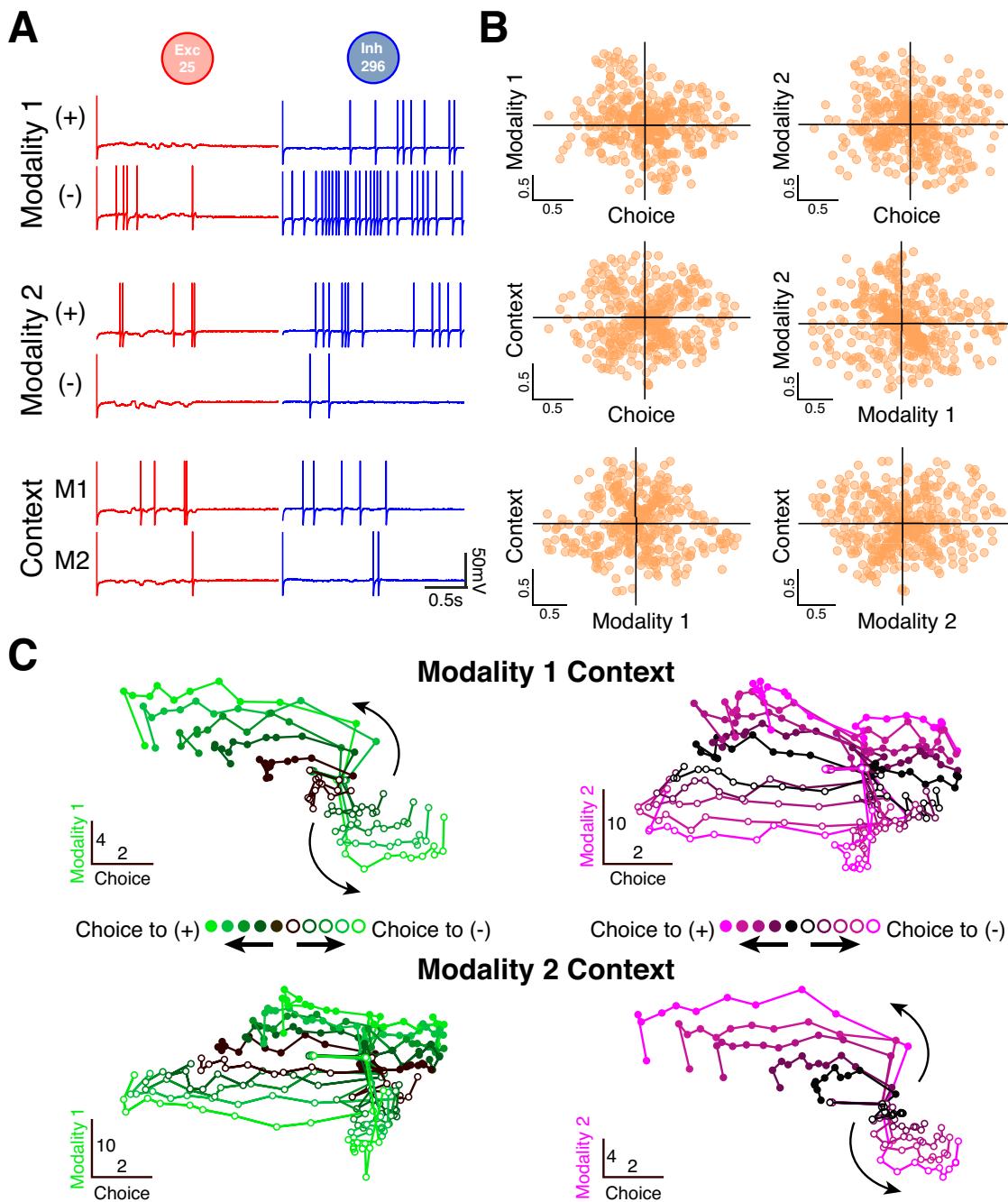


Fig. 4 | Caption on next page.

216 ronal synchrony and network behavior. Previous studies have shown that inhibitory interneurons,
217 especially parvalbumin (PV)-positive interneurons, are critical for regulating neuronal synchrony
218 [20, 23, 25]. Dysfunction and disruption of inhibitory signaling mediated by PV interneurons have
219 been strongly associated with network dysfunctions along with behavioral impairment relevant to
220 various neuropsychiatric disorders [26, 27]. Consistent with these findings, a recent study using a

Fig. 4 | (Previous page) The LIF network model employs mixed representations of the task variables. **A.** Mixed representation of the task variables at the level of single units. An excitatory unit (red) and an inhibitory unit (blue) with mixed representation of three task variables (modality 1, modality 2, and context) are shown as examples. The excitatory neuron preferred modality 1 input signals with negative offset values, modality 2 signals with positive offset values, and modality 1 context (left column). The inhibitory neuron also exhibited similar biases (right column). **B.** De-noised regression coefficients from all the units in the network. The coefficients were obtained from the multivariate linear regression analysis used by Mante et al. [4]. **C.** Average population responses projected to a low dimensional state space. The targeted dimensionality reduction technique (developed by Mante et al. [4]) was used to project the population activities to the state space spanned by the task-related axes (which were obtained using the regression analysis mentioned in **B**). For the modality 1 context (top row), the population responses from the trials with various modality 1 offset values were projected to the choice and modality 1 axes (left). The same trials were sorted by the irrelevant modality (modality 2) and shown on the right. Similar conventions used for the modality 2 context (bottom row). The offset magnitude (i.e. amount of evidence toward “+” or “-” choice) increases from dark to light. Filled and empty circles correspond to “+” choice and “-” choice trials, respectively.

221 mouse model of schizophrenia showed that decreased activity of PV neurons led to desynchronization
222 of pyramidal neurons and working memory deficits often seen in schizophrenia [28].

223 Motivated by this recent study, we constructed an excitatory-inhibitory spiking RNN to perform
224 a task that required working memory and employed a spike-based synchrony measure to charac-
225 terize how inhibitory signaling contributes to precise neuronal synchrony and working memory
226 maintenance. We used a temporal exclusive-OR (XOR) task paradigm, where each trial began
227 with two sequential stimuli separated by a brief delay period (Fig. 5A). A network of 200 LIF
228 units (158 excitatory and 42 inhibitory units) with an average synaptic decay of 44.4 ± 28.0 ms
229 could successfully perform the task (Fig. 5B). During a stimulus period, the input signal was held
230 at either -1 or +1. If the two sequential stimuli had the same sign (+1/+1 or -1/-1), then the
231 network was trained to produce an output signal approaching +1 (Fig. 5C top). If the stimuli had
232 different signs (+1/-1 or -1/+1), the output of the network approached -1 (Fig. 5C bottom). This
233 task is a classical example of working memory tasks as it requires the network to briefly retain and
234 recall the first stimulus identity in order to make a correct decision during the response window.

235 The neural population trajectories projected to a low-dimensional space discovered by principal
236 component analysis (PCA) revealed how the spiking network performed the working memory task
237 (Fig. 5D; Supplementary Fig. 5; see Methods). During the fixation period before the presentation

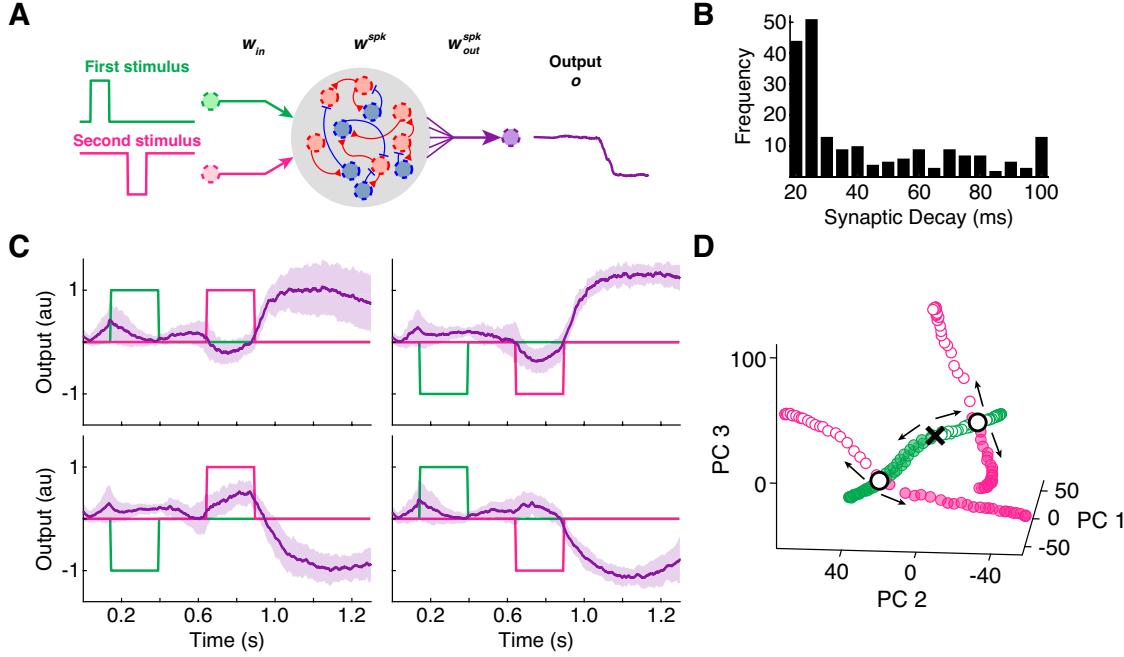


Fig. 5 | LIF network model performing a temporal XOR task. **A.** Schematic diagram showing the temporal XOR task paradigm. Two sequential stimuli (250 ms in duration each) separated by a 250 ms delay were given to the LIF network model. If the two stimuli did not match (shown here), the network output approached -1. For a matching case (+/+ or -/-), the output approached +1. **B.** Distribution of the optimized synaptic decay time constants (τ). Mean \pm SD, 44.4 ± 28.0 ms. **C.** Network performance for each trial condition. Mean \pm SD across 50 trials for each condition (dark purple). **D.** Low dimensional neural trajectory evolution during the XOR task. PCA was applied to the neural responses from the onset of the first stimulus period to the end of the second stimulus window. Black cross, first stimulus onset; black circles, second stimulus onset; green filled circles, “+1” first stimulus; green empty circles, “-1” first stimulus; magenta filled circles, “+1” second stimulus; magenta empty circles, “-1” second stimulus. See Supplementary Fig. 5 for different views.

238 of the first stimulus, all four trajectories corresponding to the four trial types stayed together as
 239 expected (data not shown). Then the trajectories diverged based on the identity of the first stimulus
 240 forming two stable “tunnels” traveling in the opposite directions: one for the “+1” first stimulus
 241 and the other for the “-1” first stimulus (green filled and empty circles in Fig. 5D). During the
 242 delay period, the dynamical landscape was maintained, and the two tunnels stayed well-separated.
 243 During the second stimulus period, these two tunnels bifurcated again to form four trajectories
 244 in a manner that allowed the network to preserve all three task variables (first stimulus, second
 245 stimulus, and response). The first principal component (PC) encoded the information related to
 246 the first stimulus: the trajectories with the “+1” first stimulus resided in the negative PC 1 region,

247 while the trajectories corresponding to the “-1” first stimulus stayed in the positive PC 1 area. The
248 identity of the second stimulus was represented by the third PC, and the second PC encoded the
249 network response variable. Therefore, the low-dimensional neural response trajectories revealed
250 how short-term memories were represented in the spiking network performing the temporal XOR
251 task.

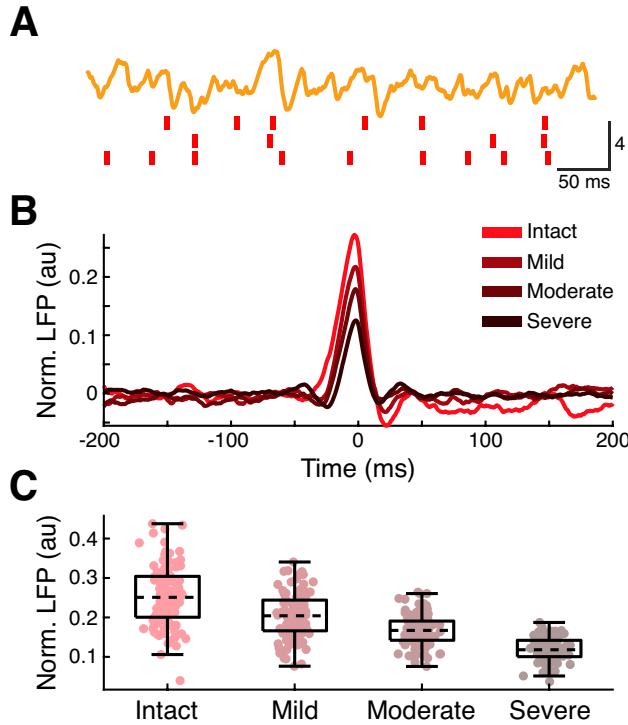


Fig. 6 | Inhibitory units tightly regulate network synchrony. **A.** Example LFP proxy signal (orange) with spikes from three randomly selected excitatory units (red). Spontaneous activities were extracted from the LIF network model constructed to perform the XOR task. The LFP signal was normalized using z-score transformation. **B.** STAs computed from the LIF network with different degrees of inhibitory signaling impairment. For each impairment condition, the average STA time-series over 100 trials is shown. Fraction of suppressed inhibitory units increases from light to dark. **C.** The STA amplitude values at spike times ($t = 0$) from all 100 trials are shown for each condition. Intact, no suppression of inhibitory units; mild, 24% inhibitory units suppressed; moderate, 48% inhibitory units suppressed; severe, 71% inhibitory units suppressed. Box plot central lines, median; bottom and top edges, lower and upper quartiles. All pairwise Student’s t-tests were significant at $p < 0.001$.

252 Finally, we studied how attenuated inhibitory signaling altered the neuronal synchrony and the
253 network dynamics. In order to model diminished PV interneuron-mediated signaling transmission,
254 we suppressed random subpopulations of the inhibitory units in the trained model by delivering
255 strong hyperpolarizing currents throughout the trial. The size of the subpopulations was varied

256 to simulate different degrees of inhibitory signaling attenuation. We considered three levels of
257 inhibitory unit suppression: weak (24% of inhibitory units suppressed), moderate (48% inhibitory
258 units suppressed), and severe (71% inhibitory units suppressed). We first characterized the sponta-
259 neous (no input stimuli) excitatory population synchrony for each of the three levels by computing
260 spike-triggered average (STA) of local field potential (LFP) proxy signals (see Methods) [29]. If
261 neurons fire in a synchronized manner with respect to the local population activity (as estimated
262 by the LFP), the STA signal shows a prominent peak around each spike time [29, 30]. Here, the
263 LFP signals were modeled as the average synaptic inputs into the excitatory units and normalized
264 to the z-scores for each trial (Fig. 6A; see Methods). For the intact network, the excitatory units
265 fired more often during synchronous excitatory synaptic input activities leading to a large positive
266 peak in the average STA signal (Fig. 6B and 6C). As the fraction of the suppressed inhibitory
267 units increased, the STA peak amplitude decreased indicating desynchronization of the excitatory
268 units (Fig. 6B and 6C). In these impaired network models, the excitatory units fired more spon-
269 taneously (Supplementary Fig. 6). The increased spontaneous excitatory activities and disrupted
270 network synchrony are in line with the recent findings where hypofunctioning PV interneurons led
271 to desynchronized assemblies of pyramidal cells with increased spontaneous activities [28].

272 To assess the severity of working memory impairment in each of the models, we focused on
273 encoding and maintenance of the first stimulus identity by the excitatory population using a cross-
274 temporal pattern analysis method, which previous studies have successfully employed to probe
275 dynamic working memory coding [12, 24, 31–33]. For each inhibitory suppression level, we obtained
276 excitatory population responses for each trial type. Then these responses were grouped by the first
277 stimulus identity only and were split into a training and a test dataset. A linear, maximum-
278 correlation classifier was then trained to decode the identity of the first stimulus at each time
279 point of the trial (see Methods). For the intact model, the excitatory units encoded the first
280 stimulus robustly across the entire trial duration (Fig. 7 top left). This is consistent with the low
281 dimensional trajectories shown in Fig. 5D along with the previous experimental findings where
282 stable representations of stimuli persisted long after the presentation of the stimuli [24, 33]. The
283 population coding of the first stimulus was disrupted as the inhibitory units were suppressed
284 (Fig. 7). In the most severe case (71% of the inhibitory units suppressed), the identity of the first
285 stimulus could only be decoded during the first stimulus period (Fig. 7 bottom right), suggesting
286 that the loss of inhibitory signaling disrupted the stable working memory representation of the
287 first stimulus identity. The neural responses projected to the first two PCs confirmed that the

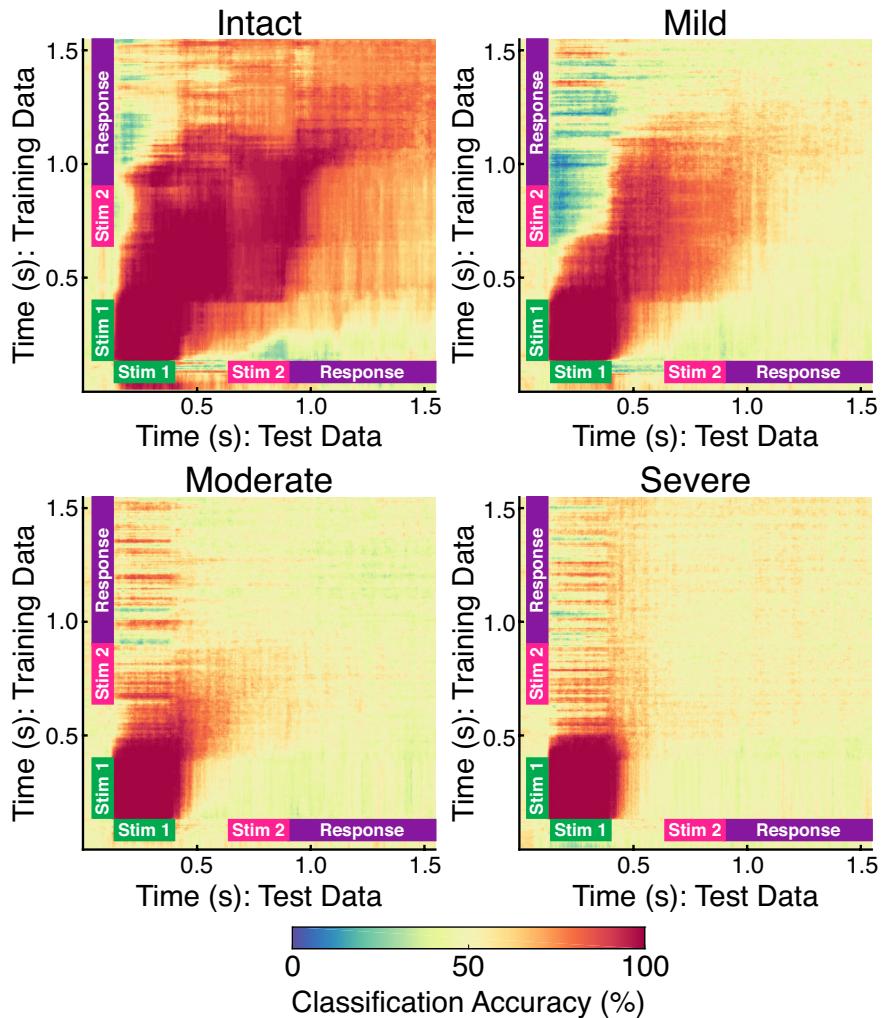


Fig. 7 | Suppression of the inhibitory units impairs working memory. Cross-temporal discriminability analysis of the first stimulus identity revealed reliable encoding that generalized across the trial (first stimulus, delay, second stimulus, and response epochs) for the intact network (top left). For a weak impairment condition (24% inhibitory units suppressed), the first stimulus could be decoded robustly until the beginning of the response epoch (top right). As the fraction of the suppressed inhibitory units increased further, the reliability of the stimulus encoding dropped markedly (bottom row).

memory of the first stimulus identity was indeed abolished in the moderate and severe models (Supplementary Fig. 7). On the other hand, suppressing a significant portion of the excitatory units (50% of the excitatory units suppressed) did not produce network desynchronization and working memory deficits (Supplementary Fig. 8). These findings indicate that the inhibitory units in our spiking model are critical for controlling network dynamics and carrying out important computations.

294 Discussion

295 In the current study, we presented a simple framework that harnesses the dynamics of trained
296 continuous rate network models to produce functional spiking RNN models. This framework can
297 flexibly incorporate functional connectivity constraints and heterogeneous synaptic time constants.
298 The spiking RNNs were constructed to perform various cognitive task paradigms, including context-
299 dependent input integration and working memory tasks; rate- and spike-based measures illuminated
300 the neural dynamics underlying cognitive processes.

301 The type of approach used in this study (i.e. conversion of a rate network to a spiking net-
302 work) has been previously employed in neuromorphic engineering to construct power-efficient deep
303 spiking networks [34–36]. These studies mainly employed feedforward multi-layer networks or con-
304 volutional neural networks aimed to accurately classify input signals or images without placing too
305 much emphasis on biophysical limitations. The overarching goal in these studies was to maximize
306 task performance while minimizing power consumption and computational cost. On the other
307 hand, the main aim of the present study was to construct spiking recurrent network models that
308 abide by important biological constraints in order to relate emerging mechanisms and dynamics
309 to experimentally observed findings. To this end, we have carefully designed our continuous rate
310 RNNs to include several biological features. These include (1) non-negative firing rates (imposed
311 by the sigmoid transfer function), (2) sparse connectivity that respects Dale’s principle, and (3)
312 heterogeneous synaptic decay time constants. Incorporating these biologically motivated details
313 into our rate network model enabled us to utilize transfer learning to create a functional spiking
314 model.

315 Recent studies have proposed methods that built on the FORCE method to train spiking RNNs
316 [5, 15–17]. Conceptually, our work is most similar to the work by DePasquale et al. [16]. The
317 method developed by DePasquale et al. [16] also relies on mapping a trained continuous-variable
318 rate RNN to a spiking RNN model. However, the rate RNN model used in their study was designed
319 to provide dynamically rich auxiliary basis functions meant to be distributed to overlapping popu-
320 lations of spiking units. Due to this reason, the relationship between their rate and spiking models
321 is rather complex, and it is not straightforward to impose functional connectivity constraints on
322 their spiking RNN model. An additional procedure was introduced to implement Dale’s principle,
323 but this led to more fragile spiking networks with considerably increased training time [16]. The
324 one-to-one mapping between rate and spiking networks employed in our method solved these prob-
325 lems without sacrificing network stability and computational cost: biophysical constraints that we

326 wanted to incorporate into our spiking model were implemented in our rate network model first
327 and then transferred to the spiking model.

328 The recurrent weight parametrization method proposed by Song et al. [10] to train continuous
329 rate RNNs that satisfy Dale’s principle was also employed in our study to constrain our rate mod-
330 els. Surprisingly, this constraint was transferable to spiking RNNs to produce separate excitatory
331 and inhibitory populations. This biological feature allowed us to characterize the functional role of
332 inhibitory units in governing neuronal synchrony and network dynamics (Fig. 6 and Fig. 7). Fur-
333 thermore, other connectivity motifs motivated by biology can be enforced and transferred using
334 our framework (Supplementary Fig. 1).

335 In addition to imposing specific connectivity patterns, we have also optimized synaptic decay
336 time constants. Previous studies have investigated homogeneous models where all the units in a
337 network shared the same time constant [16–18, 37]. However, Kim and Chow [5] underscored the
338 importance of synaptic time scales in training spiking recurrent networks. If all the units in a
339 network have slow synaptic time constants, they cannot track fast changes present in the target
340 dynamics. On the other hand, if the synaptic time scale is too fast, the ability for spikes to encode
341 continuous signals deteriorates resulting in a large “sampling” error. Instead of having all the
342 units operate in the same time scale, we have included the synaptic time constants as another set
343 of model parameters to be optimized via backpropagation. This modification allowed our spiking
344 networks to exploit units with a diverse range of synaptic time scales. Diversity of neuronal and
345 synaptic properties is found throughout the brain and may be a general principle.

346 Since our framework involves rate RNNs that operate in a rate coding scheme, the spiking RNNs
347 that our framework produces also employ rate coding by nature. Previous studies have shown that
348 spike-coding can improve spiking efficiency and enhance network stability [15, 21, 38], and it will
349 be important to build on our current method to include spike-coding schemes. In addition, our
350 framework does not model nonlinear dendritic processes which have been shown to play a significant
351 role in efficient input integration and flexible information processing [17, 39, 40]. Incorporating
352 nonlinear dendritic processes into our platform using the method proposed by Thalmeier et al.
353 [17] will be an interesting next step to further investigate the role of dendritic computation in
354 information processing. Lastly, the backpropagation method utilized in our framework to train
355 rate RNNs in a supervised manner is not biologically plausible. However, previous studies have
356 validated and uncovered neural mechanisms observed in experimental settings using RNN models
357 trained with backpropagation [4, 10, 37]. Thus, a network model may be biologically plausible,

358 and improve our understanding of neural systems, even if it was constructed using non-biological
359 means. Testing if our framework can be generalized to support more biologically realistic training
360 methods, such as reinforcement learning methods, is also an important future direction.

361 In summary, we provide an easy-to-use platform that converts a continuous recurrent network
362 model to a more biologically realistic, spiking model. The framework along with the findings
363 presented in this study will be valuable for future experimental and theoretical studies aimed at
364 uncovering neural computations underlying cognition.

365 References

- 366 [1] Sompolinsky, H., Crisanti, A. & Sommers, H. J. Chaos in random neural networks. *Phys. Rev. Lett.*
367 **61**, 259–262 (1988).
- 368 [2] Sussillo, D. & Abbott, L. Generating coherent patterns of activity from chaotic neural networks.
369 *Neuron*. **63**, 544 – 557 (2009).
- 370 [3] Laje, R. & Buonomano, D. V. Robust timing and motor patterns by taming chaos in recurrent neural
371 networks. *Nature Neuroscience*. **16**, 925–933 (2013).
- 372 [4] Mante, V., Sussillo, D., Shenoy, K. V. & Newsome, W. T. Context-dependent computation by recurrent
373 dynamics in prefrontal cortex. *Nature*. **503**, 78–84 (2013).
- 374 [5] Kim, C. M. & Chow, C. C. Learning recurrent dynamics in spiking networks. *eLife*. **7**, e37124 (2018).
- 375 [6] Mastrogiuseppe, F. & Ostojic, S. Linking connectivity, dynamics, and computations in low-rank recur-
376 rent neural networks. *Neuron*. **99**, 609–623 (2018).
- 377 [7] Enel, P., Procyk, E., Quilodran, R. & Dominey, P. F. Reservoir computing properties of neural dynamics
378 in prefrontal cortex. *PLOS Computational Biology*. **12**, e1004967 (2016).
- 379 [8] Rajan, K., Harvey, C. D. & Tank, D. W. Recurrent network models of sequence generation and memory.
380 *Neuron*. **90**, 128 – 142 (2016).
- 381 [9] Barak, O., Sussillo, D., Romo, R., Tsodyks, M. & Abbott, L. From fixed points to chaos: Three models
382 of delayed discrimination. *Progress in Neurobiology*. **103**, 214 – 222 (2013).
- 383 [10] Song, H. F., Yang, G. R. & Wang, X.-J. Training excitatory-inhibitory recurrent neural networks for
384 cognitive tasks: A simple and flexible framework. *PLOS Computational Biology*. **12**, e1004792 (2016).
- 385 [11] Song, H. F., Yang, G. R. & Wang, X.-J. Reward-based training of recurrent neural networks for
386 cognitive and value-based tasks. *eLife*. **6**, e21492 (2017).
- 387 [12] Miconi, T. Biologically plausible learning in recurrent neural networks reproduces neural dynamics
388 observed during cognitive tasks. *eLife*. **6**, e20899 (2017).
- 389 [13] Huh, D. & Sejnowski, T. J. Gradient descent for spiking neural networks. In Bengio, S., Wallach, H.,
390 Larochelle, H., Grauman, K., Cesa-Bianchi, N. & Garnett, R., editors, *Advances in Neural Information
391 Processing Systems 31*. pages 1433–1443 (2018).
- 392 [14] Lee, J. H., Delbruck, T. & Pfeiffer, M. Training deep spiking neural networks using backpropagation.
393 *Frontiers in Neuroscience*. **10**, 508 (2016).
- 394 [15] Abbott, L. F., DePasquale, B. & Memmesheimer, R.-M. Building functional networks of spiking model
395 neurons. *Nature Neuroscience*. **19**, 350–355 (2016).
- 396 [16] DePasquale, B., Churchland, M. M. & Abbott, L. F. Using firing-rate dynamics to train recurrent
397 networks of spiking model neurons. Preprint at arXiv <https://arxiv.org/abs/1601.07620> (2016).
- 398 [17] Thalmeier, D., Uhlmann, M., Kappen, H. J. & Memmesheimer, R.-M. Learning universal computations
399 with spikes. *PLOS Computational Biology*. **12**, e1004895 (2016).
- 400 [18] Nicola, W. & Clopath, C. Supervised learning in spiking neural networks with force training. *Nature
401 Communications*. **8**, 2208 (2017).

- 402 [19] Werbos, P. J. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*.
403 **78**, 1550–1560 (1990).
- 404 [20] Isaacson, J. S. & Scanziani, M. How inhibition shapes cortical activity. *Neuron*. **72**, 231–243 (2011).
- 405 [21] Denéve, S. & Machens, C. K. Efficient codes and balanced networks. *Nature Neuroscience*. **19**, 375–382
406 (2016).
- 407 [22] Gupta, N., Singh, S. S. & Stopfer, M. Oscillatory integration windows in neurons. *Nature Communica-
408 tions*. **7**, 13808 (2016).
- 409 [23] Cardin, J. A. Inhibitory interneurons regulate temporal precision and correlations in cortical circuits.
410 *Trends in Neurosciences*. **41**, 689–700. (2018).
- 411 [24] Wasmuht, D. F., Spaak, E., Buschman, T. J., Miller, E. K. & Stokes, M. G. Intrinsic neuronal dynamics
412 predict distinct functional roles during working memory. *Nature Communications*. **9**, 3499 (2018).
- 413 [25] Ferguson, B. R. & Gao, W.-J. PV interneurons: Critical regulators of E/I balance for prefrontal
414 cortex-dependent behavior and psychiatric disorders. *Frontiers in Neural Circuits*. **12**, 37 (2018).
- 415 [26] Murray, A. J., Woloszynowska-Fraser, M. U., Ansel-Bollepalli, L., Cole, K. L. H., Foggetti, A., Crouch,
416 B., Riedel, G. & Wulff, P. Parvalbumin-positive interneurons of the prefrontal cortex support working
417 memory and cognitive flexibility. *Scientific Reports*. **5**, 16778 (2015).
- 418 [27] Zick, J. L., Blackman, R. K., Crowe, D. A., Amirikian, B., DeNicola, A. L., Netoff, T. I. & Chafee,
419 M. V. Blocking NMDAR disrupts spike timing and decouples monkey prefrontal circuits: Implications
420 for activity-dependent disconnection in schizophrenia. *Neuron*. **98**, 1243–1255 (2018).
- 421 [28] Marissal, T., Salazar, R. F., Bertollini, C., Mutel, S., De Roo, M., Rodriguez, I., Müller, D. & Carleton,
422 A. Restoring wild-type-like CA1 network dynamics and behavior during adulthood in a mouse model
423 of schizophrenia. *Nature Neuroscience*. **21**, 1412–1420 (2018).
- 424 [29] Fries, P., Reynolds, J. H., Rorie, A. E. & Desimone, R. Modulation of oscillatory neuronal synchro-
425 nization by selective visual attention. *Science*. **291**, 1560–1563 (2001).
- 426 [30] Denker, M., Roux, S., Lindén, H., Diesmann, M., Riehle, A. & Grün, S. The Local Field Potential
427 Reflects Surplus Spike Synchrony. *Cerebral Cortex*. **21**, 2681–2695 (2011).
- 428 [31] Meyers, E. M., Freedman, D. J., Kreiman, G., Miller, E. K. & Poggio, T. Dynamic population coding
429 of category information in inferior temporal and prefrontal cortex. *Journal of Neurophysiology*. **100**,
430 1407–1419 (2008).
- 431 [32] Stokes, M. G., Kusunoki, M., Sigala, N., Nili, H., Gaffan, D. & Duncan, J. Dynamic coding for cognitive
432 control in prefrontal cortex. *Neuron*. **78**, 364–375 (2013).
- 433 [33] Spaak, E., Watanabe, K., Funahashi, S. & Stokes, M. G. Stable and dynamic coding for working
434 memory in primate prefrontal cortex. *Journal of Neuroscience*. **37**, 6503–6516 (2017).
- 435 [34] Diehl, P. U., Neil, D., Binas, J., Cook, M., Liu, S. & Pfeiffer, M. Fast-classifying, high-accuracy spiking
436 deep networks through weight and threshold balancing. In *2015 International Joint Conference on
437 Neural Networks (IJCNN)*. pages 1–8 (2015).
- 438 [35] Diehl, P. U., Zarrella, G., Cassidy, A., Pedroni, B. U. & Neftci, E. Conversion of artificial recurrent

- 439 neural networks to spiking neural networks for low-power neuromorphic hardware. In *2016 IEEE*
440 *International Conference on Rebooting Computing (ICRC)*. pages 1–8 (2016).
- 441 [36] Hunsberger, E. & Eliasmith, C. Training spiking deep networks for neuromorphic hardware. Preprint
442 at arXiv <https://arxiv.org/abs/1611.05141> (2016).
- 443 [37] Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J. & Wang, X.-J. Computing by robust
444 transience: How the fronto-parietal network performs sequential, category-based decisions. *Neuron*.
445 **93**, 1504–1517 (2017).
- 446 [38] Alemi, A., Machens, C. K., Denéve, S. & Slotine, J.-J. E. Learning nonlinear dynamics in efficient,
447 balanced spiking networks using local plasticity rules. In *AAAI*. (2018).
- 448 [39] Ujfalussy, B. B., Makara, J. K., Branco, T. & Lengyel, M. Dendritic nonlinearities are tuned for efficient
449 spike-based computations in cortical circuits. *eLife*. **4**, e10056 (2015).
- 450 [40] Yang, G. R., Murray, J. D. & Wang, X.-J. A dendritic disinhibitory circuit mechanism for pathway-
451 specific gating. *Nature Communications*. **7**, 12815 (2016).

452 Acknowledgements

453 We are grateful to Ben Huh, Gerald Pao, Jason Fleischer, Debha Amatya, Yusi Chen, and Ben
454 Tsuda for helpful discussions and feedback on the manuscript. We also thank Jorge Aldana for
455 assistance with computing resources. This work was funded by the National Institute of Mental
456 Health (F30MH115605-01A1 to R.K.), Harold R. Schwalenberg Medical Scholarship (R.K.), and
457 Burnand-Partridge Foundation Scholarship (R.K.). We also gratefully acknowledge the support of
458 NVIDIA Corporation with the donation of the Quadro P6000 GPU used for this research. The
459 funders had no role in study design, data collection and analysis, decision to publish, or preparation
460 of the manuscript.

461 Author contributions

462 R.K. and T.J.S. designed the study and wrote the manuscript. R.K. and Y.L. performed model
463 analyses and simulations.

464 Declaration of interests

465 The authors declare no competing interests.

466 Methods

467 **Continuous rate network structure.** The continuous rate RNN model contains N units recur-
 468 rently connected to one another. The dynamics of the model is governed by

$$\tau \frac{dx}{dt} = -x + W^{rate} r^{rate} + I_{ext} \quad (5)$$

469 where $\tau \in \mathbb{R}^{1 \times N}$ corresponds to the synaptic decay time constants for the N units in the network
 470 (see **Training details** on how these are initialized and optimized), $x \in \mathbb{R}^{1 \times N}$ is the synaptic current
 471 variable, $W^{rate} \in \mathbb{R}^{N \times N}$ is the synaptic connectivity matrix, and $r^{rate} \in \mathbb{R}^{1 \times N}$ is the output of the
 472 units. The output of each unit, which can be interpreted as the firing rate estimate, is obtained
 473 by applying a nonlinear transfer function to the synaptic current variable (x) elementwise:

$$r^{rate} = \phi(x)$$

474 We use a standard logistic sigmoid function for the transfer function to constrain the firing rates
 475 to be non-negative:

$$\phi(x) = \frac{1}{1 + \exp(-x)} \quad (6)$$

476 The connectivity weight matrix (W^{rate}) is initialized as a random, sparse matrix drawn from a
 477 normal distribution with zero mean and a standard deviation of $1.5/\sqrt{N \cdot P_c}$ where $P_c = 0.10$ is
 478 the initial connectivity probability.

479 The external currents (I_{ext}) include task-specific input stimulus signals (see **Implementation**
 480 **of computational tasks and figure details**) along with a Gaussian white noise variable:

$$I_{ext} = W_{in} u + \mathcal{N}(0, 0.01)$$

481 where the time-varying stimulus signals ($u \in \mathbb{R}^{N_{in} \times 1}$) are fed to the network via $W_{in} \in \mathbb{R}^{N \times N_{in}}$,
 482 a Gaussian random matrix with zero mean and unit variance. N_{in} corresponds to the number of
 483 input signals associated with a specific task, and $\mathcal{N}(0, 0.01) \in \mathbb{R}^{N \times 1}$ represents a Gaussian random
 484 noise with zero mean and variance of 0.01.

485 The output of the rate RNN at time t is computed as a linear readout of the population activity:

$$o^{rate}(t) = W_{out}^{rate} r^{rate}(t)$$

486 where $W_{out}^{rate} \in \mathbb{R}^{1 \times N}$ refers to the readout weights.

487 Eq. (5) is discretized using the first-order Euler approximation method:

$$\begin{aligned} \mathbf{x}_t = & \left(1 - \frac{\Delta t}{\tau}\right) \mathbf{x}_{t-1} + \frac{\Delta t}{\tau} (W^{rate} \mathbf{r}_{t-1}^{rate} + W_{in} \mathbf{u}_{t-1}) \\ & + \mathcal{N}(0, 0.01) \end{aligned}$$

488 where $\Delta t = 5$ ms is the discretization time step size used throughout this study.

489 **Spiking network structure.** For our spiking RNN model, we considered a network of leaky
490 integrate-and-fire (LIF) units governed by

$$\tau_m \frac{d\mathbf{v}}{dt} = -\mathbf{v} + W^{spk} \mathbf{r}^{spk} + \mathbf{I}_{ext} \quad (7)$$

491 In the above equation, $\tau_m = 10$ ms is the membrane time constant shared by all the LIF units,
492 $\mathbf{v} \in \mathbb{R}^{1 \times N}$ is the membrane voltage variable, $W^{spk} \in \mathbb{R}^{N \times N}$ is the recurrent connectivity matrix,
493 and $\mathbf{r}^{spk} \in \mathbb{R}^{1 \times N}$ represents the spike trains filtered by a synaptic filter. Throughout the study,
494 the double exponential synaptic filter was used to filter the presynaptic spike trains:

$$\begin{aligned} \frac{dr_i^{spk}}{dt} &= -\frac{r_i^{spk}}{\tau_i} + s_i \\ \frac{ds_i}{dt} &= -\frac{s_i}{\tau_r} + \frac{1}{\tau_r \tau_i} \sum_{t_i^k < t} \delta(t - t_i^k) \end{aligned}$$

495 where $\tau_r = 2$ ms and τ_i refer to the synaptic rise time and the synaptic decay time for unit i ,
496 respectively. The synaptic decay time constant values ($\tau_i \in \boldsymbol{\tau}$) are trained and transferred to our
497 LIF RNN model (see **Training details**). The spike train produced by unit i is represented as a
498 sum of Dirac δ functions, and t_i^k refers to the k -th spike emitted by unit i .

499 The external current input (\mathbf{I}_{ext}) is similar to the one used in our continuous model (see **Con-**
500 **tinuous rate network structure**). The only difference is the addition of a constant background
501 current set near the action potential threshold (see below).

502 The output of our spiking model at time t is given by

$$o^{spk}(t) = W_{out}^{spk} \mathbf{r}^{spk}(t)$$

503 Other LIF model parameters were set to the values used by Nicola and Clopath [18]. These
504 include the action potential threshold (-40 mV), the reset potential (-65 mV), the absolute refrac-
505 tory period (2 ms), and the constant bias current (-40 pA). The parameter values for the LIF and
506 the quadratic integrate-and-fire (QIF) models are listed in Supplementary Table 1.

507 **Training details.** In this study, we only considered supervised learning tasks. A task-specific
 508 target signal (\mathbf{z}) is used along with the rate RNN output (\mathbf{o}^{rate}) to define the loss function (\mathcal{L}),
 509 which our rate RNN model is trained to minimize. Throughout the study, we used the root mean
 510 squared error (RMSE) defined as

$$\mathcal{L} = \sqrt{\left(\sum_{t=1}^T (z(t) - o^{rate}(t))^2 \right)} \quad (8)$$

511 where T is the total number of time points in a single trial.

512 In order to train the rate model to minimize the above loss function (Eq. 8), we employed
 513 ADaptive Moment Estimation (ADAM) stochastic gradient descent algorithm. The learning rate
 514 was set to 0.01, and the TensorFlow default values were used for the first and second moment
 515 decay rates. The gradient descent method was used to optimize the following parameters in the
 516 rate model: synaptic decay time constants (τ), recurrent connectivity matrix (W^{rate}), and readout
 517 weights (W_{out}^{rate}).

518 Here we describe the method to train synaptic decay time constants (τ) using backpropagation.
 519 First, the time constants are initialized with random values within the specified range:

$$\tau = \sigma(\mathcal{N}(0, 1)) \cdot \tau_{step} + \tau_{min}$$

520 where $\sigma(\cdot)$ is the sigmoid function (identical to Eq. 6) used to constrain the time constants to
 521 be non-negative. The time constant values are also bounded by the minimum (τ_{min}) and the
 522 maximum ($\tau_{max} = \tau_{min} + \tau_{step}$) values. The error computed from the loss function (Eq. 8) is then
 523 backpropagated to update the time constants at each iteration:

$$\frac{\partial \mathcal{L}}{\partial \tau} = \frac{\partial \mathcal{L}}{\partial \mathbf{r}} \cdot \frac{\partial \mathbf{r}}{\partial \mathbf{x}} \cdot \frac{\partial \mathbf{x}}{\partial \tau}$$

524 The method proposed by Song et al. [10] was used to impose Dale’s principle and create separate
 525 excitatory and inhibitory populations. Briefly, the recurrent connectivity matrix (W^{rate}) in the
 526 rate model is parametrized by

$$W^{rate} = [W^{rate}]_+ \cdot D \quad (9)$$

527 where the rectified linear operation ($[\cdot]_+$) is applied to the connectivity matrix at each update step.
 528 The diagonal matrix ($D \in \mathbb{R}^{N \times N}$) contains +1’s for excitatory units and -1’s for inhibitory units in
 529 the network. Each unit in the network is randomly assigned to one group (excitatory or inhibitory)
 530 before training, and the assignment does not change during training (i.e. D stays fixed).

531 To impose specific connectivity patterns, we apply a binary mask ($M \in \mathbb{R}^{N \times N}$) to Eq. 9:

$$W^{rate} = ([W^{rate}]_+ \cdot D) \odot M$$

532 where \odot refers to the Hadamard operation (elementwise multiplication). Similar to the diagonal
 533 matrix (D), the mask matrix stays fixed throughout training. For example, the following mask
 534 matrix can be used to create a subgroup of inhibitory units (Group A) that do not receive synaptic
 535 inputs from the rest of the inhibitory units (Group B) in the network (see Supplementary Fig. 1):

$$m_{ij} = \begin{cases} 0 & i \in \text{Group A}, j \in \text{Group B} \\ 1 & \text{otherwise} \end{cases}$$

536 where $m_{ij} \in M$ establishes (if $m_{ij} = 1$) or removes (if $m_{ij} = 0$) the connection from unit j to unit
 537 i .

538 **Transfer learning from a rate model to a spiking model.** In this section, we describe the
 539 method that we developed to perform transfer learning from a trained rate model to a LIF model.
 540 Once the rate RNN model is trained using the gradient descent method outlined in **Training**
 541 **details**, the rate model parameters are transferred to a LIF network in a one-to-one manner.
 542 First, the LIF network is initialized to have the same topology as the trained rate RNN. Next, the
 543 input weight matrix (W_{in}) and the synaptic decay time constants (τ) are transferred to the spiking
 544 RNN without any modification. Lastly, the recurrent connectivity matrix (W^{rate}) and the readout
 545 weights (W_{out}^{rate}) are scaled by a constant number, λ , and transferred to the spiking network.

546 If the recurrent connectivity weights from the trained rate model are transferred to a spiking
 547 network without any changes, the spiking model produces largely fluctuating signals (as illustrated
 548 in Fig. 1D), because the LIF firing rates are significantly larger than 1 (whereas the firing rates of
 549 the rate model are constrained to range between zero and one by the sigmoid transfer function).

550 To place the spiking RNN in the similar dynamic regime as the rate network, we first assume
 551 a linear relationship between the rate model connectivity weights and the spike model weights:

$$W^{spk} = \lambda \cdot W^{rate}$$

552 Using the above assumption, the synaptic drive (d) that unit i in the LIF RNN receives can be

553 expressed as

$$\begin{aligned}
 d_i^{spk}(t) &= \sum_{j=1}^N w_{ij}^{spk} \cdot r_j^{spk}(t) \\
 &\approx \sum_{j=1}^N (\lambda \cdot w_{ij}^{rate}) \cdot r_j^{spk}(t) \\
 &= \sum_{j=1}^N w_{ij}^{rate} \cdot (\lambda \cdot r_j^{spk}(t))
 \end{aligned} \tag{10}$$

554 where $w_{ij}^{spk} \in W^{spk}$ is the synaptic weight from unit j to unit i .

555 Similarly, unit i in the rate RNN model receives the following synaptic drive at time t :

$$d_i^{rate}(t) = \sum_{j=1}^N w_{ij}^{rate} \cdot r_j^{rate}(t) \tag{11}$$

556 If we set the above two synaptic drives (Eq. 10 and Eq. 11) equal to each other, we have:

$$\begin{aligned}
 d_i^{spk}(t) &= d_i^{rate}(t) \\
 \sum_{j=1}^N w_{ij}^{rate} \cdot (\lambda \cdot r_j^{spk}(t)) &= \sum_{j=1}^N w_{ij}^{rate} \cdot r_j^{rate}(t)
 \end{aligned} \tag{12}$$

557 Generalizing Eq. 12 to all the units in the network, we have

$$\mathbf{r}^{rate}(t) = \lambda \cdot \mathbf{r}^{spk}(t)$$

558 Therefore, if there exists a constant factor (λ) that can account for the firing rate scale difference
 559 between the rate and the spiking models, the connectivity weights from the rate model (W^{rate})
 560 can be scaled by the factor and transferred to the spiking model.

561 The readout weights from the rate model (W_{out}^{rate}) are also scaled by the same constant factor
 562 (λ) to have the spiking network produce output signals similar to the ones from the trained rate
 563 model:

$$\begin{aligned}
 o^{rate}(t) &= W_{out}^{rate} \cdot \mathbf{r}^{rate}(t) \\
 &\approx W_{out}^{rate} \cdot (\lambda \cdot \mathbf{r}^{spk}(t)) \\
 &= (\lambda \cdot W_{out}^{rate}) \cdot \mathbf{r}^{spk}(t) = o^{spk}(t)
 \end{aligned}$$

564 In order to find the optimal scaling factor, we developed a simple grid search algorithm. For a
 565 given range of values for λ (ranged from 0.0125 to 0.10 with a step size of 0.0001), the algorithm
 566 finds the optimal value that minimizes the RMSE between the rate network output and the spiking
 567 model output signals.

568 **Implementation of computational tasks and figure details.** In this section, we describe the
569 details of the parameters and methods used to generate all the main figures in the present study.

570 **Fig. 1.** A rate RNN of $N = 200$ units (162 excitatory and 38 inhibitory units) was trained to
571 perform a Go-NoGo task. Each trial lasted for 1000 ms (200 time steps with 5 ms step size).
572 The minimum and the maximum synaptic decay time constants were set to 20 ms and 50 ms,
573 respectively. An input stimulus with a pulse 125 ms in duration was given for a Go trial, while
574 no input stimulus was given for a NoGo trial. The network was trained to produce an output
575 signal approaching +1 after the stimulus offset for a Go trial. For a NoGo trial, the network was
576 trained to maintain its output at zero. A trial was considered correct if the maximum output signal
577 during the response window was above 0.7 for the Go trial type. For a NoGo trial, if the maximum
578 response value was less than 0.3, the trial was considered correct. For training, 6000 trials were
579 randomly generated, and the model performance was evaluated after every 100 trials. Training
580 was terminated when the loss function fell below 7 and the task performance reached at least 95%.
581 The termination criteria were usually met at or before 2000 trials for this task. A scaling factor of
582 0.02 ($\lambda = 0.02$) was used to construct a LIF network model for this task.

583 **Fig. 2.** A rate RNN model with $N = 200$ units (98 excitatory and 102 inhibitory units) was
584 trained to produce a sinusoidal signal (1 Hz) autonomously. The synaptic decay time constants
585 were set to range from 20 ms to 50 ms. Each trial lasted for 3500 ms or 700 time steps with 5 ms
586 step size, and the training was terminated when the loss function fell below 5. A scaling factor of
587 0.0286 ($\lambda = 0.0286$) was used to construct a LIF network model for this task.

588 **Fig. 3.** A network of $N = 400$ continuous-variable units (299 excitatory and 101 inhibitory units)
589 were trained to perform the context-dependent input integration task. The input matrix ($\mathbf{u} \in$
590 $\mathbb{R}^{4 \times 750}$) contained four stimuli channels across time (750 time steps with 5 ms step size). The
591 first two channels corresponded to the modality 1 and modality 2 noisy input signals. These
592 signals were modeled as white-noise signals (sampled from the standard normal distribution) with
593 constant offset terms. The sign of the offset term modeled the evidence toward (+) or (-) choices,
594 while the magnitude of the offset determined the strength of the evidence. The noisy signals were
595 only present during the stimulus window (250 ms – 2500 ms). Once the network was trained,
596 the stimulus duration was shortened to 1000 ms (250 ms – 1250 ms). The last two channels of \mathbf{u}
597 represented the modality 1 and the modality 2 context signals. For instance, the third channel of
598 \mathbf{u} is set to one and the fourth channel is set to zero to model Modality 1 context.

599 For each trial used to train the rate model, the offset values for the two modality input signals

600 were randomly set to -0.5 or +0.5. The context signals were randomly set such that either modality
601 1 (third input channel is set to 1) or modality 2 (fourth input channel is set to 1) was cued for
602 each trial. If the offset term of the cued modality was +0.5 (or -0.5) for a given trial, the network
603 was instructed to produce an output signal approaching +1 (or -1) after the stimulus window.
604 The model performance was assessed after every 100 training trials, and the training termination
605 conditions were same as the ones used for *Figure 1*. A scaling factor of 0.0182 was used to construct
606 a LIF network model for this task.

607 For the psychometric curves (Fig. 3D), the offset value was varied from -0.5 to +0.5 with a step
608 size of 0.1.

609 **Fig. 4.** To investigate how task variables and offset values affected the network dynamics, we used
610 the spiking network constructed in Fig. 3 to generate neural responses for different trial conditions.
611 We considered 11 levels of offset ranging from -0.5 to +0.5 with a step size of 0.1. Therefore, there
612 were a total of 242 trial conditions: (11 offsets for modality 1) \times (11 offsets for modality 2) \times (2
613 contexts). For each condition, we generated 50 trials and extracted spike trains from all the units.
614 The spike data was preprocessed in a similar manner as done by Mante et al. [4]. Briefly, time-
615 varying firing rates were first estimated by counting spikes in a non-overlapping, sliding window
616 (50 ms in duration). Next, the firing rates from all the trials ($242 \times 50 = 12100$ trials) were
617 concatenated, resulting in a large matrix with 400 rows (one for each unit) and $12100 \times T$ columns
618 (where $T = 44$ is the number of time points in each trial). The firing rates of each unit (i.e.
619 each row of the large matrix) were normalized by z-score transformation using the mean and the
620 standard deviation across all the trials (i.e. across the columns of the matrix). The z-scored neural
621 responses were then used for the multi-variable linear regression and the targeted dimensionality
622 reduction analyses (implemented using the details outlined in Mante et al. [4]).

623 **Fig. 5.** A rate RNN network composed of $N = 200$ units (158 excitatory and 42 inhibitory units)
624 was trained to perform a temporal exclusive OR (XOR) task. The input matrix ($\mathbf{u} \in \mathbb{R}^{2 \times 300}$)
625 contained two input channels for two sequential stimuli (over 300 time steps with 5 ms step size).
626 The first channel delivered the first stimulus (250 ms in duration), while the second channel modeled
627 the second stimulus (250 ms in duration) which began 50 ms after the offset of the first stimulus.
628 The short delay (50 ms) allowed the model to learn the task efficiently, and the delay was increased
629 to 250 ms after training without affecting the model performance. During each stimulus window,
630 the corresponding input channel was set to either -1 or +1. If the two sequential stimuli had the
631 same sign (-1/-1 or +1/+1), the network was trained to produce an output signal approaching +1

632 after the offset of the second stimulus. If the stimuli had opposite signs (-1/+1 or +1/-1), then the
633 network produced an output signal approaching -1. Training was stopped when the loss function
634 fell below 7, and the task performance was greater than 95%. A scaling factor of 0.0167 was used
635 to construct a LIF network model for this task.

636 Principal component analysis (PCA) was performed on the instantaneous firing rates obtained
637 from the LIF network. The firing rates were estimated by applying the double synaptic filter shown
638 in **Spiking network structure**. For each trial condition, neural responses were extracted from
639 50 trials. There were a total of 200 trials (4 trial conditions and 50 trials per condition). PCA
640 was then applied to the neural responses (concatenated across all the trials), and the top three
641 principal components were used to represent the low dimensional network activities (Fig. 5D).

642 **Fig. 6.** To compute spike-triggered averages (STAs) of local field potential (LFP) proxy signals,
643 spontaneous spike trains were first extracted from the LIF model: for each “spontaneous” trial
644 (i.e. no stimulus input given to the network), spike trains from the excitatory units were extracted.
645 For each trial, the LFP proxy signal was modeled as z-scored average synaptic inputs into the
646 excitatory units over time. A 400-ms window, centered at each spike time in the extracted spike
647 trains, was used to extract spike-triggered LFP segments. These segments were then averaged to
648 obtain the STA for the trial. For each inhibitory suppression condition (intact, mild, moderate,
649 and severe), STAs were computed from 100 trials and averaged across the trials.

650 **Fig. 7.** A cross-temporal decoding method similar to the ones used by Miconi [12], Meyers et al.
651 [31] was employed to assess the encoding stability of the LIF network. More specifically, we studied
652 the stability of the first stimulus encoding by the excitatory units for each inhibitory suppression
653 condition. For each trial condition (-1/-1, -1/+1, +1/-1, +1/+1), population activities (time-
654 varying firing rates) from 50 trials were extracted. These trials were then separated by the identity
655 of the first stimulus leading to two groups of neural responses from the “-1” condition (100 trials
656 with the “-1” first stimulus) and the “+1” condition (100 trials with the “+1” first stimulus). The
657 first half of the neural activities from each condition was chosen as a training dataset, while the
658 second half was used for testing. A maximal-correlation classifier (identical to the one used by
659 Miconi [12]) was then trained on the training dataset and tested on the test data.

660 **Code availability**

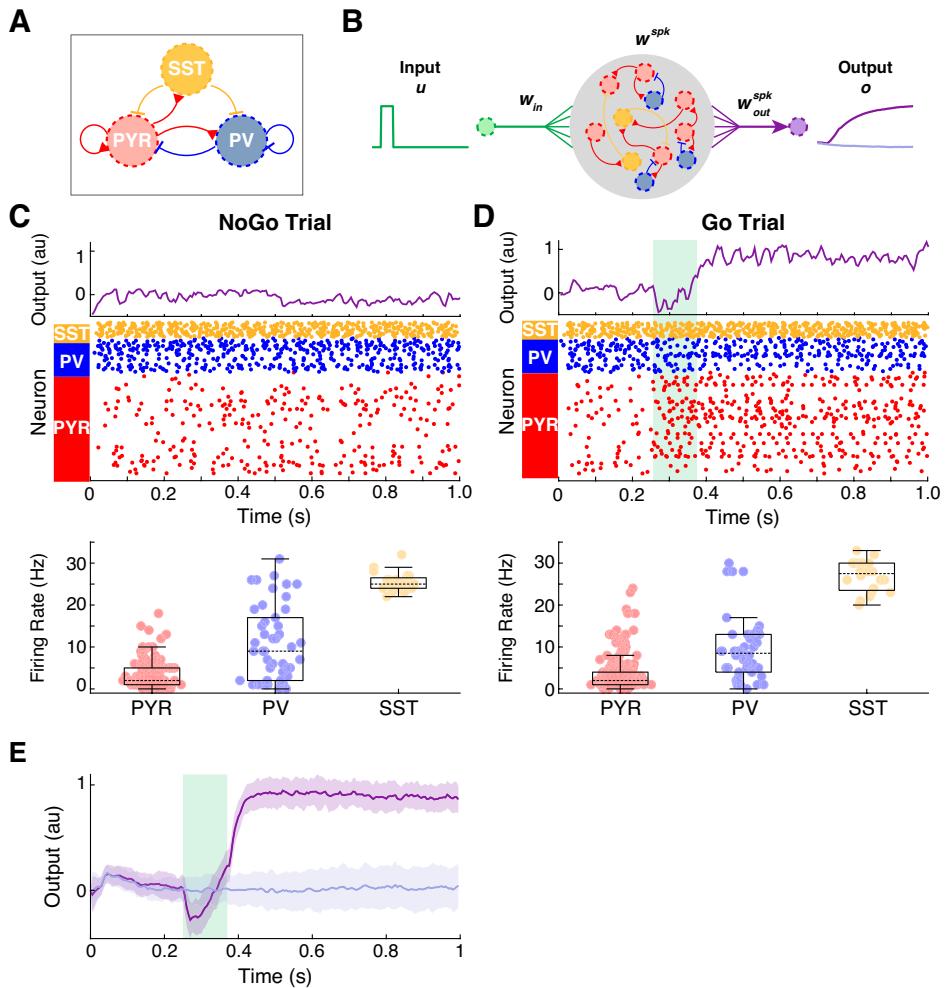
661 The implementation of our framework and the codes to generate all the figures in this work are
662 available at <https://github.com/rkim35/spikeRNN>.

663 **Data availability**

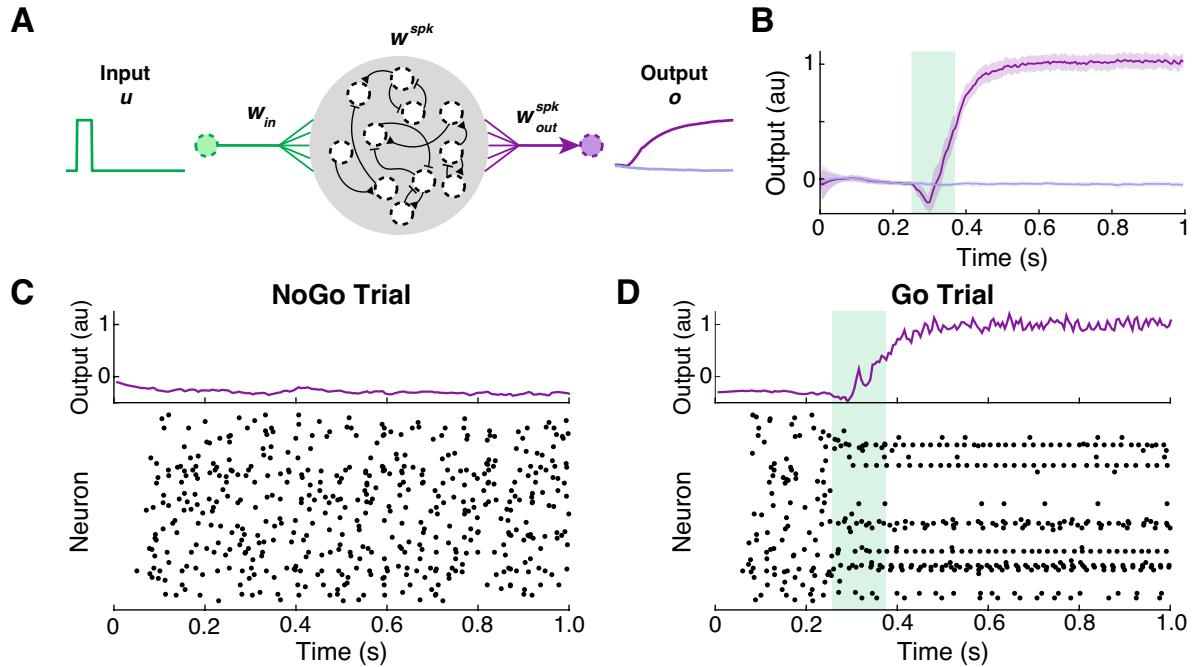
664 The trained models used in the present study are available as MATLAB-formatted data at <https://github.com/rkim35/spikeRNN>:

665 //github.com/rkim35/spikeRNN.

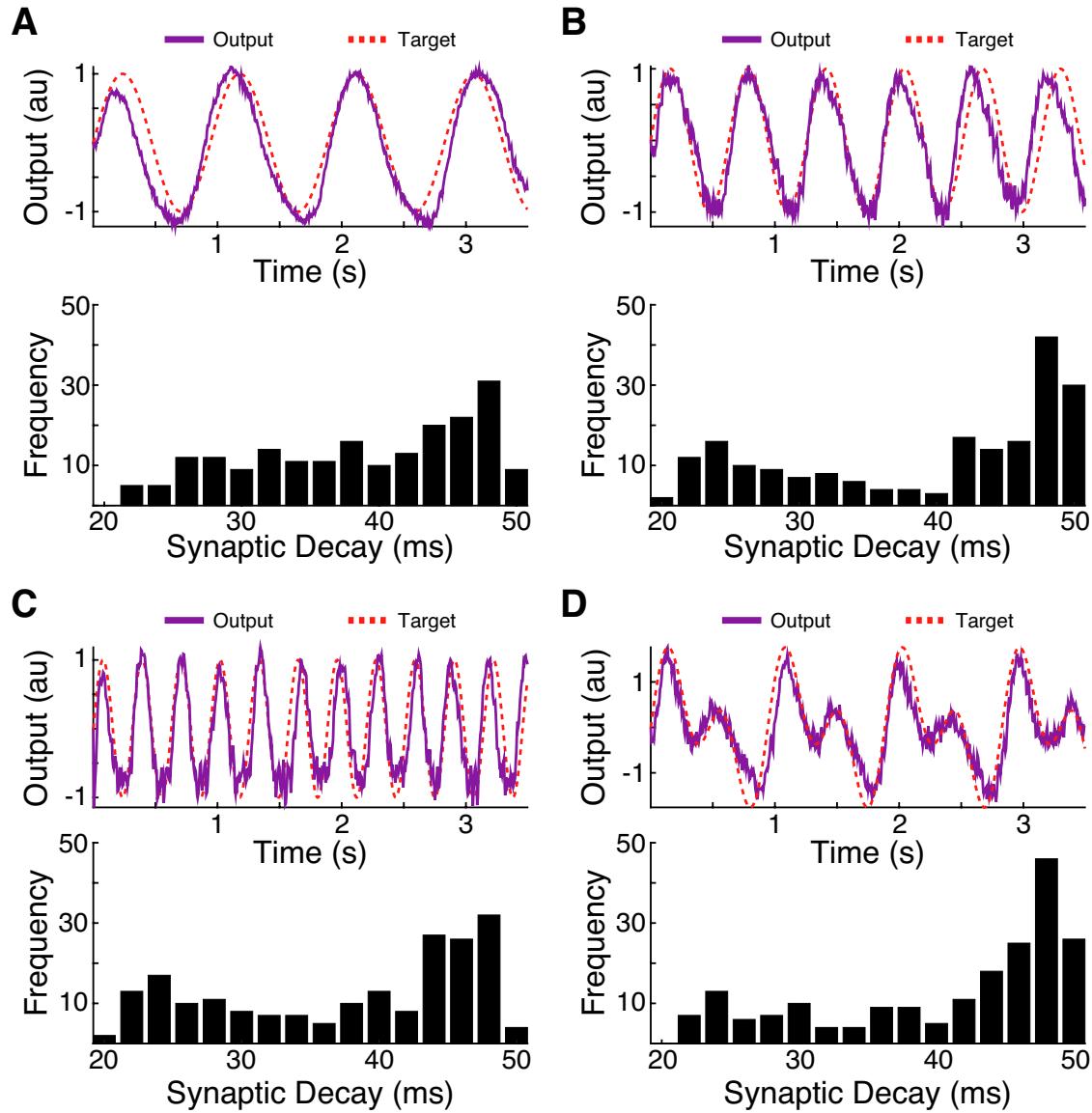
Supplementary Figures



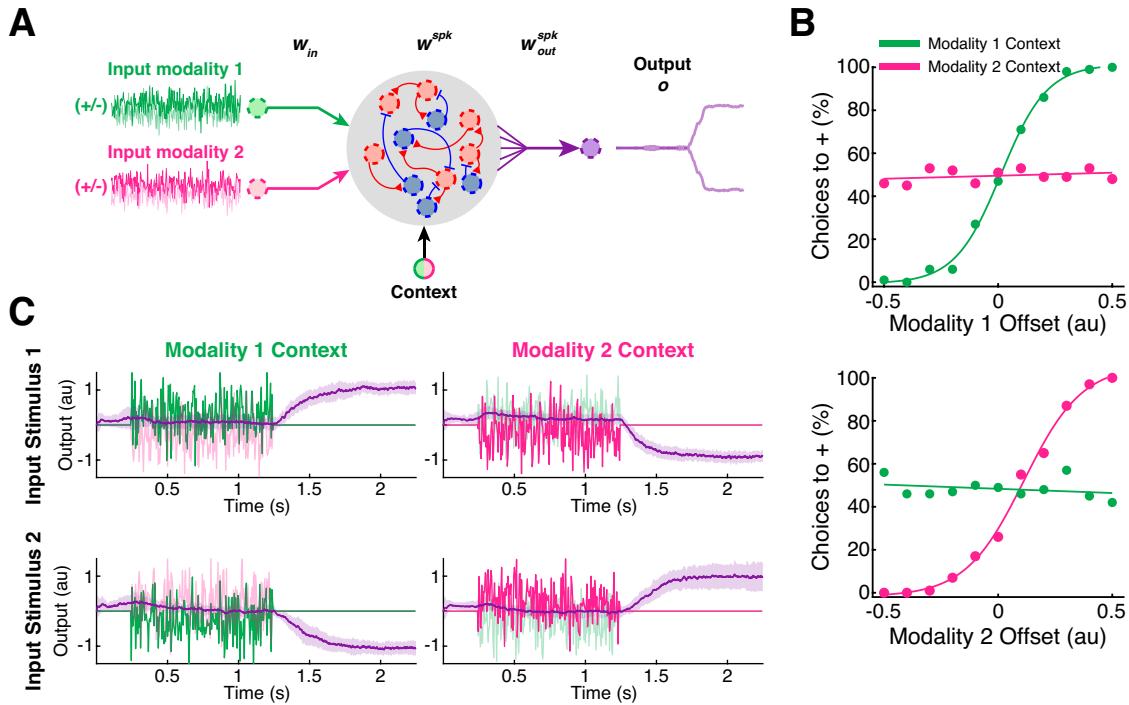
Supplementary Fig. 1 | Incorporation of additional functional connectivity constraints. **A.** Common cortical microcircuit motif where somatostatin-expressing interneurons (SST; yellow circle) inhibit both pyramidal (PYR; red circle) and parvalbumin-expressing (PV; blue circle) neurons. **B.** Schematic illustrating the incorporation of the connectivity motif shown in **A** into a LIF network model. The connectivity pattern was imposed during training of a rate network model ($N = 200$) to perform the Go-NoGo task. There were 134 PYR, 46 PV, and 20 SST units. A spiking model was constructed using the trained rate model with $\lambda = 0.02$. **C.** Example output response and spikes from the LIF network model for a single NoGo trial. Mean \pm SD firing rate for each population is also shown (PYR, 3.08 ± 3.29 Hz; PV, 10.80 ± 8.94 Hz; SST, 25.50 ± 2.33 Hz). **D.** Example output response and spikes from the LIF network model for a single Go trial. Mean \pm SD firing rate for each population is also shown (PYR, 4.72 ± 5.89 Hz; PV, 9.30 ± 8.16 Hz; SST, 27.05 ± 3.98 Hz). Box plot central lines, median; bottom and top edges, lower and upper quartiles. **E.** LIF network model performance on 50 NoGo trials (light purple) and 50 Go trials (dark purple). Mean \pm SD shown.



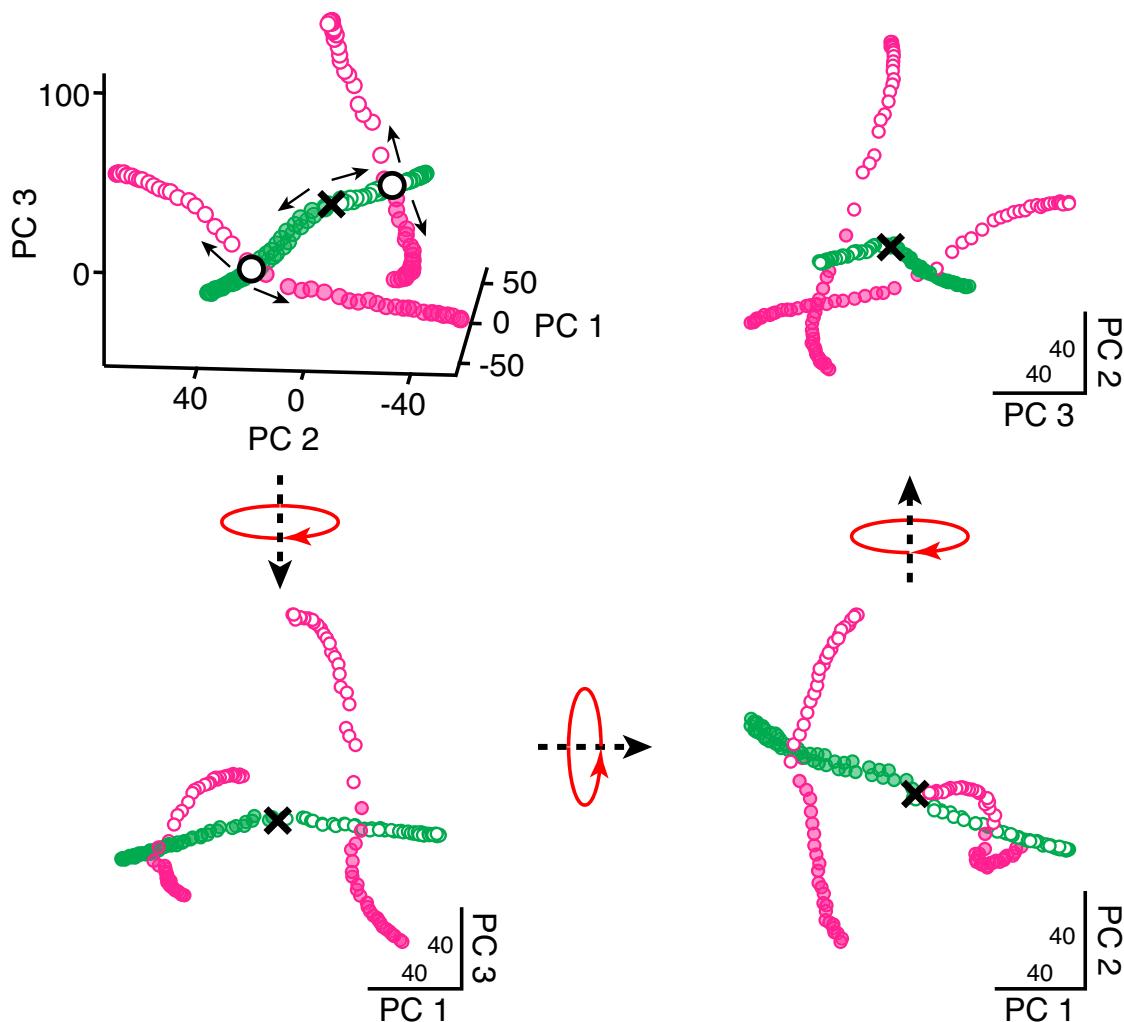
Supplementary Fig. 2 | Dale’s principle constraint can be relaxed. **A.** Schematic diagram showing a LIF network model without Dale’s principle. A rate RNN model ($N = 200$) without Dale’s principle was first trained to perform the Go-NoGo task. The scaling factor (λ) was set to 0.02. Note that each unit (black dotted circles) can exert both excitatory and inhibitory effects. **B.** LIF network model performance on 50 NoGo trials (light purple) and 50 Go trials (dark purple). Mean \pm SD shown. **C.** Example output response (top) and spikes (bottom) from the LIF network model for a single NoGo trial. **D.** Example output response (top) and spikes (bottom) from the LIF network model for a single Go trial.



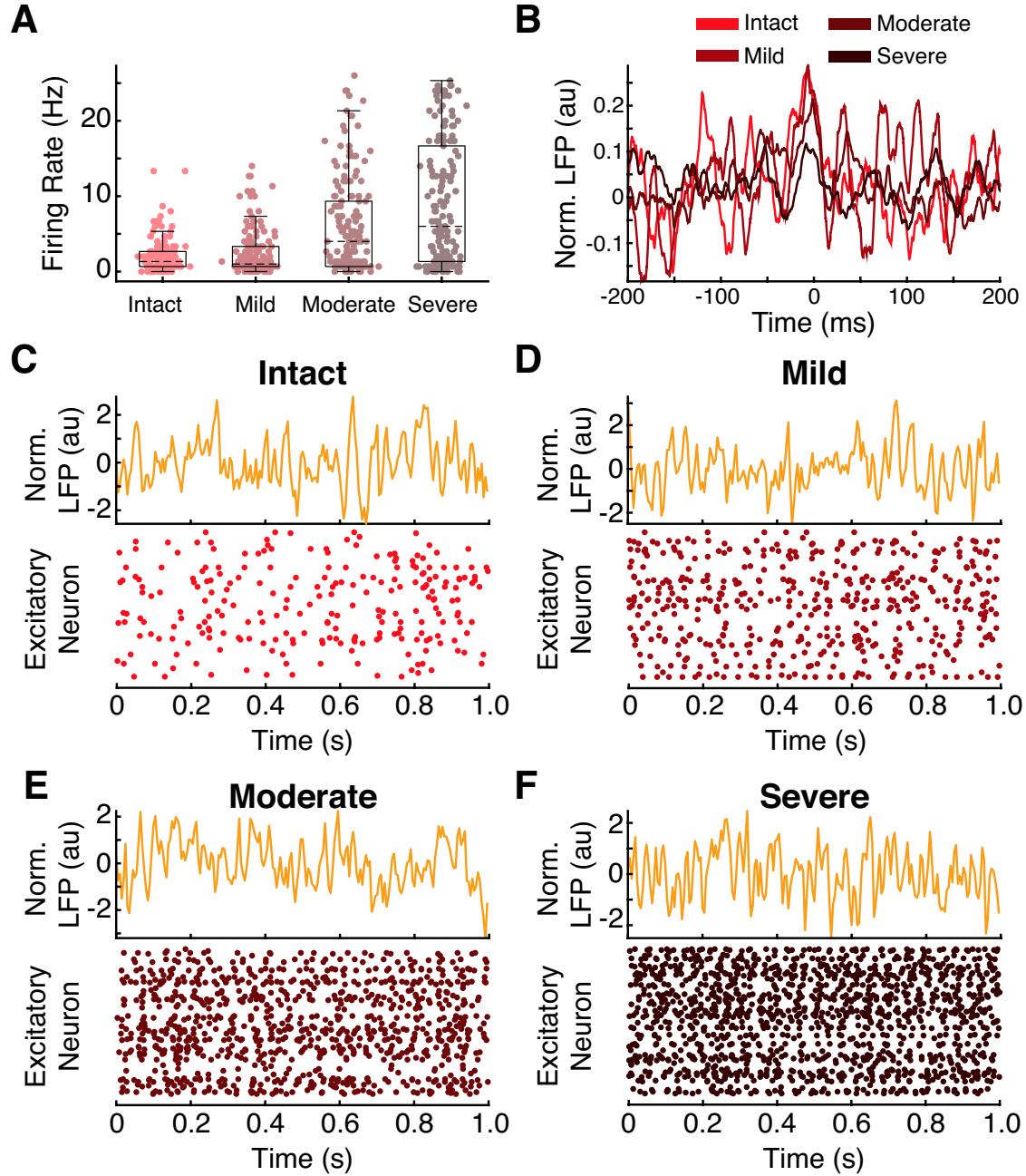
Supplementary Fig. 3 | Autonomous oscillation tasks with different target signals. **A.** Output signal (solid purple line) from a spiking model constructed to produce a 1 Hz sine signal (dotted magenta line) along with the distribution of the trained synaptic time constants (bottom). The spiking model used here is identical to the one used for Fig. 2. **B.** Output signal (solid purple line) from a spiking model constructed to produce a 1.5 Hz sine signal (dotted magenta line). The optimized synaptic decay time constants are also shown (bottom). **C.** Output signal (solid purple line) from a spiking model constructed to produce a 3 Hz sine signal (dotted magenta line) along with the distribution of the trained synaptic time constants (bottom). **D.** Output signal (solid purple line) from a spiking model constructed to produce a target signal obtained by combining the two target signals from **A** and **B**, and the tuned synaptic decay time constants (bottom).



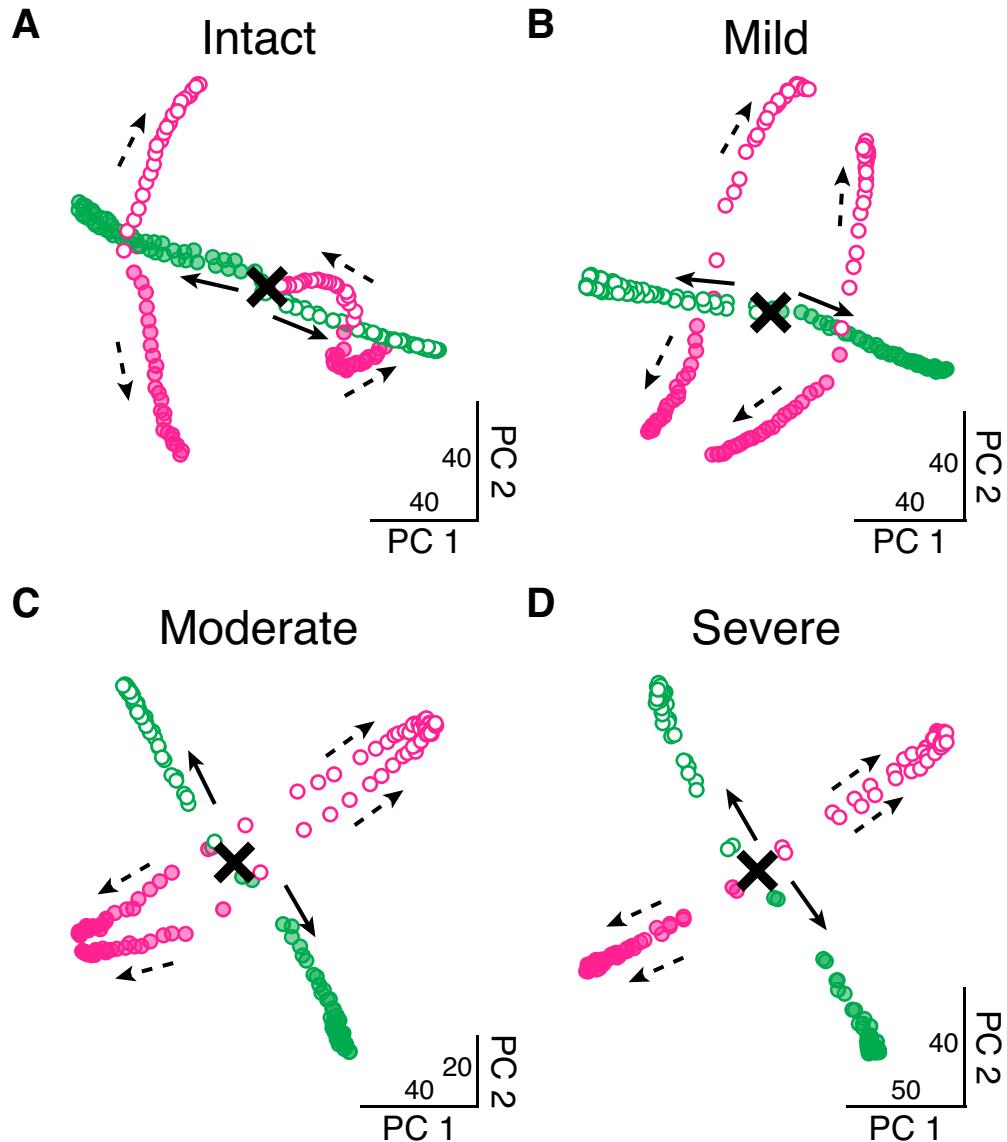
Supplementary Fig. 4 | Quadratic integrate-and-fire (QIF) model constructed to perform the context-dependent input integration task. **A.** The task paradigm and the trained rate network model used for Fig. 3 were employed to build a QIF model. The QIF model parameter values are listed in Supplementary Table 1. **B.** Psychometric curves from the QIF network model. The percentage of trials where the QIF network indicated “+” choice as a function of the modality 1 offset values (top) and modality 2 offsets (bottom). **C.** The QIF model successfully performed the task by integrating cued modality input signals. Example noisy input signals (scaled by 0.5 vertically for visualization; green and magenta lines) from a single trial are shown. Mean \pm SD response signals (purple lines) across 50 trials for each trial type.



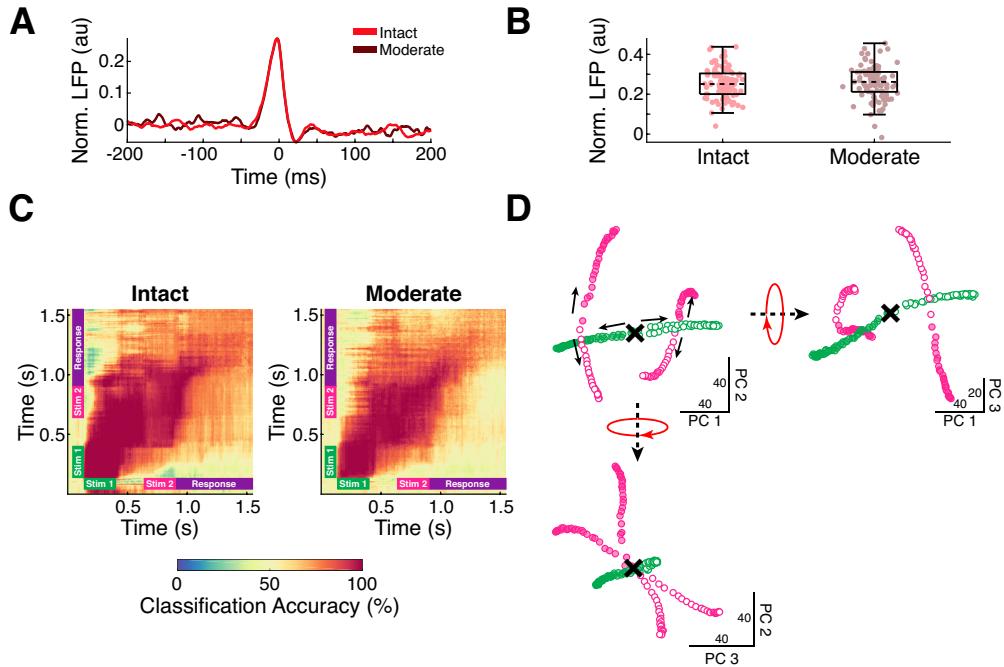
Supplementary Fig. 5 | Low dimensional neural response trajectories of the intact network model during the sequential XOR task. Different views of the three-dimensional PCA plot shown in Fig. 5D (reproduced here for reference).



Supplementary Fig. 6 | Increased spontaneous excitatory unit activities from network models with impaired inhibitory units. **A.** As the fraction of the suppressed inhibitory units increased, the spontaneous firing rates of the excitatory units also increased. For each condition (intact, mild, moderate, and severe), the excitatory firing rates from a single trial are shown. Box plot central lines, median; bottom and top edges, lower and upper quartiles. Same color scheme as Fig. 6. **B.** Single-trial STA signals from the four conditions. The LFP signal and the excitatory spike raster plot for the intact model (**C**), the mild model (**D**), the moderate model (**E**), and the severe model (**F**) are also shown.



Supplementary Fig. 7 | Low dimensional trajectories reveal impaired working memory computations in networks with compromised inhibitory units. Neural population trajectories projected to the first two principal components (PCs) for the intact model (**A**), the mild model (**B**), the moderate model (**C**), and the severe model (**D**). As the fraction of the suppressed inhibitory units increased, the network began to lose the memory of the the first stimulus identity and retained only the second stimulus identity as shown by the two opposite “tunnels” formed during the second stimulus epoch (magenta empty and filled circles) in **D**. The trajectories evolve temporally in the following order: black cross (first stimulus onset), solid arrows (first stimulus epoch), dashed arrows (second stimulus epoch).



Supplementary Fig. 8 | Suppression of excitatory units does not lead to significant neuronal desynchrony and working memory impairment. **A.** STAs computed from the intact LIF network (red line) and the network with 50% of the excitatory units suppressed (brown line). For each condition, the average STA time-series over 100 trials is shown. **B.** The STA amplitude values at spike times ($t = 0$) from the two conditions (intact and moderate) were not significantly different. Box plot central lines, median; bottom and top edges, lower and upper quartiles. **C.** The network model with impaired excitatory units was able to encode the first stimulus identity reliably across the trial epochs. The cross-temporal decoding analysis (same as the one used for Fig. 7) was performed. **D.** Neural response trajectories of the “moderate” network model projected to the first three principal components (PCs). Even with the suppressed excitatory units, the network was able to preserve the three task-related variables: first stimulus identity, second stimulus identity, and response.

Supplementary Notes

For the quadratic integrate-and-fire (QIF) model (Supplementary Fig. 4), we considered a network of units governed by

$$\tau_m \frac{d\mathbf{v}}{dt} = \mathbf{v}^2 + W^{spk} \mathbf{r}^{spk} + \mathbf{I}_{ext}$$

The definitions of the variables are identical to the ones used for the LIF network model.

Supplementary Table

	LIF	QIF
Membrane time constant (τ_m)	10 ms	10 ms
Absolute refractory period	2 ms	2 ms
Synaptic rise time (τ_r)	2 ms	2 ms
Constant bias current	-40 pA	0 pA
Spike threshold	-40 mV	30 mV
Spike reset voltage	-65 mV	-65 mV

Supplementary Table 1 | Parameter values used to construct LIF and QIF networks.