Context-modular memory networks support high-capacity, flexible, and robust associative memories

William F. Podlaski,* **Everton J. Agnes**,† and **Tim P. Vogels**†

Centre for Neural Circuits and Behaviour, University of Oxford, OX1 3SR Oxford, United Kingdom

Context, such as behavioral state, is known to modulate memory formation and retrieval, but is usually ignored in associative memory models. Here, we propose several types of contextual modulation for associative memory networks that greatly increase their performance. In these networks, context inactivates specific neurons and connections, which modulates the effective connectivity of the network. Memories are stored only by the active components, thereby reducing interference from memories acquired in other contexts. Such networks exhibit several beneficial characteristics, including enhanced memory capacity, high robustness to noise, increased robustness to memory overloading, and better memory retention during continual learning. Furthermore, memories can be biased to have different relative strengths, or even gated on or off, according to contextual cues, providing a candidate model for cognitive control of memory and efficient memory search. An external context-encoding network can dynamically switch the memory network to a desired state, which we liken to experimentally observed contextual signals in prefrontal cortex and hippocampus. Overall, our work illustrates the benefits of organizing memory around context, and provides an important link between behavioral studies of memory and mechanistic details of neural circuits.

SIGNIFICANCE

Memory is context dependent — both encoding and recall vary in effectiveness and speed depending on factors like location and brain state during a task. We apply this idea to a simple computational model of associative memory through contextual gating of neurons and synaptic connections. Intriguingly, this results in several advantages, including vastly enhanced memory capacity, better robustness, and flexible memory gating. Our model helps to explain (i) how gating and inhibition contribute to memory processes, (ii) how memory access dynamically changes over time, and (iii) how context representations, such as those observed in hippocampus and prefrontal cortex, may interact with and control memory processes.

Keywords: associative memory, context-dependent gating, recall, continual learning, cognitive control.

Context may refer to a variety of internal or external variables that an organism has access to[1] — e.g., background sensory information, spatial location, and emotional or behavioral states — that can affect neural processing and cognition[2–4]. Memory is no exception to this phenomenon — behavioral studies have long demonstrated the effects of context on memory acquisition and retrieval for Pavlovian conditioning[5] and free recall[6], and there exists several algorithmic models of context-dependent memory[7–9]. The underlying neural circuits that enable context-dependent processing are just beginning to be explored, with evidence for inhibitory gating[10–13] and contributions from the hippocampus, prefrontal cortex, and amygdala, which dynamically interact[1].

Contextual signals are likely relayed to many brain areas through top-down signals[14–16]. Recent evidence suggests a role for excitation[17], but also for different inhibitory cell types in controlling top-down modulation[18,19]. Such modulation may place the network into different states for storage and retrieval of memory[20,21] — e.g., through modulation[22], or changes in the balance of excitation and inhibition[23,24]. Despite the clear evidence for such context-dependent state changes, and their proposed use in models of other cognitive functions[25,26], to our best knowledge they have not yet been included in models of (auto-)associative memory.

Associative memory is typically modelled using abstract recurrent networks[27], in which each memory is represented by the co-activation of a set of neurons, forming a cell assembly. In these models, memory patterns are stored as attractors of the network dynamics via, e.g., a Hebbian-like learning rule[28,29], in which the connections between neurons with correlated activity are strengthened. Though the substrate and learning mechanisms behind memory formation have yet to be fully uncovered, substantial experimental evidence exists supporting the emergence of cell assemblies during learning and for correlation-based Hebbian plasticity[30].

For many associative network models, the number of stable memories that can be stored scales with the network size[27,31]. A standard Hopfield model holds approximately $0.138N$ memories[32] (where N is the number of neurons in the network), and several extensions and variants have been proposed to account for higher memory capacity or biological realism[27]. Among these variants, introduction of more general learning rules can lead to an increase in the number of stable memories up to a limit of $2N$ for memory patterns that activate half of the neurons[33]. The introduction of sparsity through low-activity patterns can further increase this number to more than $10N$ for a sparsity of 1% or less[34].

However, even these improved models come with caveats, such as unrealistic assumptions (e.g., non-local learning rules) or other undesired properties (e.g., blackout catastrophic interference[31,35] — all memories lose stability if the maximum capacity is surpassed), suggesting that our understanding of associative memory is still incomplete. Considering sequential memory storage (*continual learning*), blackout interference can be made more gradual by imposing weight bounds, causing memories to be slowly forgotten[27,31] (so-called *palimsest*

* Correspondence: william.podlaski@gmail.com;
* Present address: Champalimaud Centre for the Unknown, Lisbon, Portugal
† Co-senior author

9th January 2020

memories). However, memory capacity is limited in this case, as old memories are quickly overwritten with new ones[36]. Various remedies have been proposed to alleviate forgetting in artificial neural networks during continual learning, including context-dependence and architectural modularity[37,38], but it is unclear how these methods might operate in a more biologically-realistic setting.

Finally, how accessible each memory is (during recall) may limit the theoretically achievable storage capacity[39,40], which is often ignored in mechanistic memory models. For example, it has been posited that some cases of memory forgetting, such as amnesia, may be partially due to deficits in memory accessibility rather than decay of the memories themselves[41–43]. Here too, context dependence may control memory expression and search[8,10], for example, by directing retrieval towards particular memories, through gating or biasing of memory strength[44].

In this work, we propose a new class of context-dependent associative memory models, which we call *context-modular memory networks*, inspired by previous theoretical studies[37,45] and experimental findings[10,12,13,21,39,46]. In our model, memories are assigned to different contexts, defined by a set of active neurons and connections. We show that this modular architecture results in enhanced memory capacity, as well as robustness to overloading and continual learning, thereby providing a direct benefit for organizing memories in contextual categories. Furthermore, we propose that a separate "context-encoding" network interacts with the associative memory network, leading to a model which dynamically gates memory expression. Our model provides strong evidence for the benefits of context-dependent memory and draws links between mechanistic circuit details and memory function which can be tested experimentally.

Results

Inspired by classic models of associative memory[27,29,34] we introduce a context-dependent, i.e., context-modular, memory network (Fig. 1). It consists of a recurrent network of N neurons which exhibit elevated or suppressed activity levels (taking values of 1 or 0, respectively; see Methods for details). We define a set of s contextual states that control the network in the following two ways: first, each context may define a corresponding subset of available neurons (N_{cxt} of the total N neurons, with activity level $a = N_{\text{cxt}}/N$). All other neurons are kept inactive. We will refer to this type of contextual control as *neuron-specific gating*. Second, each context can also define a subset of available synaptic inputs per neuron (K of the total N_{cxt} inputs on average, with connectivity sparseness $b = K/N_{\text{cxt}}$). All other inputs are transiently gated off. We call this type of contextual control *synapse-specific gating*. A contextual *subnetwork* may be defined by neuron-specific gating, synapse-specific gating, or both. In a given network realization, each context can host a set of p distinct memories, i.e., patterns in which half of the neurons in the corresponding subnetwork exhibit elevated activity, chosen randomly. The total number of patterns in the network is thus $P = sp$ memories. The synaptic connectivity between neurons is defined using a Hebbian learning rule[29] (Methods). For the majority of this study, we consider networks in which only one context is active at any given time. The contextual state is imposed on the memory network by a separate context-encoding network that dynamically interacts with the memory network (Fig. 1B).

The gating schemes defined above can be interpreted as temporarily modifying the network such that there is a different *effective* connectivity matrix (and energy landscape) for each

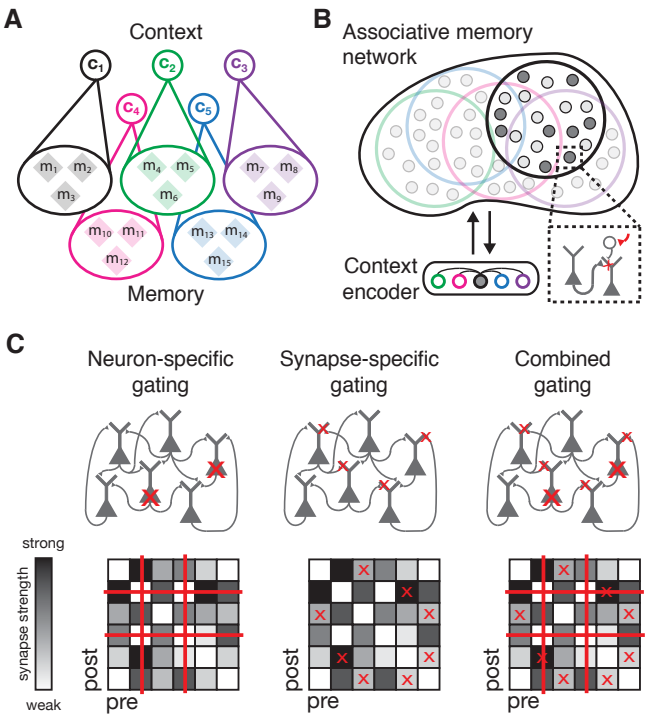

FIG. 1. Schematic of the context-modular memory network. A, Associative memory is defined hierarchically through a set of contexts (c_1 to c_5) and memory patterns (m_1 to m_{15}) assigned to each one. **B**, Network implementation: neurons are arranged into contextual configurations (subnetworks) in two ways: *neuron-specific* gating, where context is defined as a proportion of available neurons (colored rings; defined randomly, spatial localization is illustrative), and *synapse-specific* gating, where context is defined as a proportion of gated synapses (red cross, bottom right inset). Context is controlled by an external context-encoding network, such that one context is active at a time (black ring), and memories outside of the active context remain dormant. **C**, Contextual configurations change the effective connectivity matrix of the associative memory network: neuron-specific gating removes particular columns and rows (left), synapse-specific gating removes individual elements (center), and together, they will implement both effects (right).

contextual state. Neuron-specific gating effectively removes specific rows and columns from the connectivity (Fig. 1C, left), whereas synapse-specific gating removes individual entries in the connectivity matrix, thereby making it more diluted, or sparse (Fig. 1C, middle). The combination of the two produces a smaller and sparser connectivity matrix (Fig. 1C, right), with a potentially large number of synaptic connections remaining hidden, to be used in other contexts. In the following, we study the properties of these context-modular memory networks, and make comparisons with classic associative memory models[29,34].

Neuron-specific gating vastly improves memory capacity. Starting with the model of neuron-specific gating (Fig. 1C, left; $a \leq 1$, $b = 1$), we studied memory capacity using established signal-to-noise and heuristic mean-field methods[27,32]. Memory capacity denotes the maximum number of stable patterns stored in the network, divided by the network size (Methods). Here, we have two notions of capacity. The subnetwork capacity is defined as

$$\alpha_{\text{cxt}} = p/N_{\text{cxt}}. \tag{1}$$

Thus, α_{cxt} is the number of patterns stored in each subnetwork (context), p, divided by the number of active neurons in the subnetwork, N_{cxt}. For simplicity, we impose that all subnetworks have the same size and store the same number of patterns. The total network capacity is thus

$$\alpha = P/N = sp/N = \alpha_{\text{cxt}} sa, \tag{2}$$

which is the total number of patterns, $P = sp$, divided by the full network size, N, or equivalently, the subnetwork capacity, α_{cxt}, times the number of contexts, s, times the subnet ratio, $a = N_{\text{cxt}}/N$.

The stability of a memory pattern ν (of context k) can be evaluated by estimating the total input that a particular neuron i receives when this memory is being recalled, denoted as $h_i^{k\nu}$. For large networks storing a random set of patterns, the input to a neuron in an elevated state is well approximated by

$$\begin{aligned} h_i^{k\nu} &\approx 1 + \mathcal{N}\left(0, p/N_{\text{cxt}}\right) + \mathcal{N}\left(0, (s-1)pa/N\right) \\ &\approx 1 + \mathcal{N}(0, \alpha_{\text{cxt}}) + \mathcal{N}(0, \alpha a), \end{aligned} \tag{3}$$

where $\mathcal{N}(\mu, \sigma^2)$ is a normally-distributed random variable with mean μ and variance σ^2.

The two normally-distributed terms of Eq. 3 represent two different sources of noise (*crosstalk*) in the network connectivity that interfere with the stability of pattern ν of subnetwork k. The first term is the noise from the other $p-1$ patterns of context k, which scales with the number of patterns per context, p. The second term represents the noise coming from all of the $(s-1)p$ other patterns in the other contexts. This term scales with the total number of other patterns, but also with the relative amount of shared active neurons, i.e., the subnet ratio, a. If the memory is stable, then the probability that this neuron will inactivate should be low (i.e., $h_i^{k\nu} > 0$ with high probability). From this assumption, we arrive at an estimate of the maximum subnet capacity of

$$\alpha_{\text{cxt}} = \frac{\alpha_{\text{H}}}{1 + (s-1)a^2}, \tag{4}$$

with $\alpha_{\text{H}} \approx 0.138$ being the standard Hopfield network capacity[32] (*Hopfield limit*).

The intuition gained from the signal-to-noise analysis was confirmed by a more accurate mean field approach[27,32] and simulations (Methods), shown in Fig. 2A-C. The denominator in Eq. 4 is always greater than or equal to 1, and so $\alpha_{\text{cxt}} \leq \alpha_{\text{H}}$. Thus, the subnetwork capacity is upper bounded by the Hopfield limit (Fig. 2A-C, green line & arrow), and decreases as the number of subnetworks or the subnet ratio increases. This is because each subnetwork intuitively acts as a standalone Hopfield network of size N_{cxt} with p patterns, but with more noise in the weights due to the influence of the other $(s-1)p$ memories. The second term in the denominator of Eq. 4, $(s-1)a^2$, functions as a measure of the amount of overlap between subnetworks (explicitly, it is the expected number of additional contexts that each synapse will participate in).

From Eq. 2 and Eq. 4, we arrive at an expression for the maximum full network capacity,

$$\alpha = \frac{\alpha_{\text{H}} sa}{1 + (s-1)a^2}. \tag{5}$$

Taking the limit of large s (such that $s - 1 \approx s$), the full network capacity approaches $\alpha = \alpha_{\text{H}} \sqrt{s} \approx \alpha_{\text{H}}/a$. It follows that the high capacity emerges due to the sparsity in the subnetwork representation, growing sublinearly as a function of the number of contexts. Furthermore, the optimal subnetwork size for a fixed number of contexts is $a^* = 1/\sqrt{s-1}$.

Mean field results support this analysis: although subnetwork capacity remains below the Hopfield limit, the total network capacity grows well above α_{H} for many parameter values (Fig. 2D-F). For example, the capacity of a network with 200 distinct contexts is $\alpha \approx 1.2$, almost an order of magnitude higher than the Hopfield limit. We thus see that the network as a whole has substantially increased memory capacity, due to reduced interference between memories found in different contexts. Importantly, this reduced interference depends upon the fact that the majority of memories are not retrievable in each context.

Neuron-specific gating takes advantage of reduced interference by having *low-activity* memory patterns, known to increase memory capacity dramatically[34]. The capacity of the low-activity Hopfield network is comparable to our network (Fig. 2E, dashed green line), but only for very low activity levels with low information content per pattern[47]. In contrast, neuron-specific gating enables both high memory capacity and high information content (Fig. S1).

Synapse-specific gating can further improve capacity. We next studied networks with synapse-specific gating (Fig. 1C, middle; $a = 1$, $b \leq 1$), in which each subnetwork contains the full set of N neurons, but shares only a proportion of weights with the other contexts. Initially, we chose a random subset of synaptic weights to be removed for each context. Thus, each subnetwork can be seen as a diluted Hopfield network[27]. Repeating the same analysis as before, the total input coming into a particular neuron i (of pattern ν in context k) is

$$\begin{aligned} h_i^{k\nu} &\approx 1 + \mathcal{N}\left(0, \tfrac{1}{b} p/N_{\text{cxt}}\right) + \mathcal{N}\left(0, (s-1)p/N\right) \\ &= 1 + \mathcal{N}(0, \alpha_{\text{cxt}}/b) + \mathcal{N}(0, \alpha), \end{aligned} \tag{6}$$

which again contains two crosstalk (noise) terms. The fact that the parameter b appears in the first noise term and not in the second reflects the fact that each individual pattern becomes less stable with increasing sparsity[48], with no benefit across contexts either. From Eq. 6, we arrive at a capacity per context of

$$\alpha_{\text{cxt}} = \frac{\alpha_{\text{H}} b}{1 + (s-1)b}. \tag{7}$$

The factor b in the numerator reflects the fact that, even without multiple contexts, memory capacity degrades roughly linearly with network dilution[27,48]. The total network capacity is

$$\alpha = \frac{\alpha_{\text{H}} bs}{1 + (s-1)b}. \tag{8}$$

In this case, the optimal ratio of inputs per neuron is $b^* = 1$, which is independent of the number of contexts s, and thus the overall network capacity is bounded from above by α_{H}. In other words, while synapse-specific gating does enable contextual grouping, the number of stable memories is restricted by the Hopfield limit. Mean-field results confirmed this intuition (Fig. S2).

While random synapse-specific gating proves ineffective, we instead devise a more selective way of imposing synaptic control. We note here that the Hebbian learning rule (Methods) sets the synaptic weight between each pair of neurons according to the correlation in their activity across all patterns, which

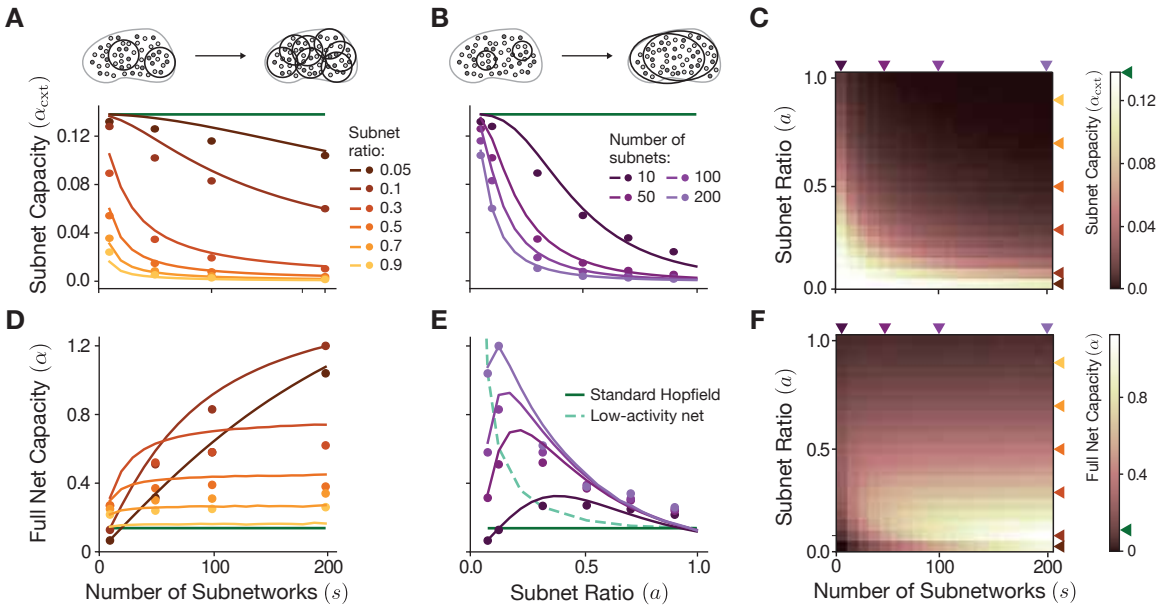

FIG. 2. Memory capacity of the context-modular memory network with neuron-specific gating. **A-C**, Mean-field capacity estimation (solid lines) and numerical simulations (dots) for subnet capacity ($\alpha_{cxt} = p/N_{cxt}$), as a function of the number of subnetworks, s (**A**) for fixed subnet ratio, $a = N_{cxt}/N$, and vice-versa (**B**), and over the full range of parameters (**C**; mean field only). Lines plotted in **A,B** are slices through **C** indicated by the colored triangles. Comparison is made with the standard Hopfield network (green). **D-F**, Same as **A-C** but for overall network capacity (full net capacity; $\alpha = sp/N = \alpha_{cxt}sa$). Memory capacity of the low-activity Hopfield network[34] is plotted in **E** with same activity level $a_{LA} = a/2$ (turquoise, dashed). $N = 10000$ for all simulations.

acts to stabilize the majority of patterns[27]. However, the set of memories in the *active* context may produce substantially different correlations, rendering some synaptic weights ineffective at stabilizing the majority of memories in this *active set*. Such synaptic weights have a net *harmful* effect on memory recall, and performance would improve for some contexts if these weights were set to zero. We thus propose and test the following gating scheme: if there is a mismatch between the sign of the synaptic weight serving to stabilize all memories versus serving to stabilize memories belonging to a specific context, then this synapse is turned off for this context (Fig. 3A, middle). We refer to this scheme as *targeted* synapse-specific (TaSS) gating.

Interestingly, TaSS gating bears a resemblance to networks with binary synaptic weights[49], in which the standard Hebbian learning rule is passed through a sign-function, and weights are set to +1 or −1. Previous work has shown that such binary synapse networks maintain a memory capacity of $\alpha_B \approx 0.1$[49], close to the standard Hopfield model ($\alpha_H \approx 0.138$). We devised a rough estimate of the memory capacity of networks with TaSS gating (combined with neuron-specific gating) using this binary synapse capacity combined with an estimate of the proportion of gated connections (Fig. 3A, right; Methods).

For parameter ranges with high overlap between contexts (large a and s), we observe that the network connectivity decreases to 50% (Fig. 3B-D) – this is because, when enough "noise" has been added to the weight matrix, each element will have the desired sign approximately half of the time. In these parameter ranges, the subnetwork capacity approaches $\alpha_{cxt} \approx \frac{1}{2}\alpha_B \approx 0.05$ (Fig. 3E-G), with the factor 1/2 reflecting the fact that sparse connectivity degrades capacity roughly linearly[48]. The full network memory capacity therefore scales linearly with the number of contexts ($\alpha \approx \frac{1}{2}\alpha_B sa$), growing to a very large value ($\alpha \approx 8$ — up to 60 times that of the standard Hopfield network for $s = 200$ contexts, and much higher than the low-activity variant over similar activity levels, Fig. 3H-J). Furthermore, we also see that the optimal subnetwork size is $a^* = 1$, and so the addition of neuron-specific gating to TaSS gating does not further increase memory capacity. This does not, however, render neuron-specific gating useless – for example, the two schemes have substantial differences in implementational complexity, as discussed next.

Implementational complexity of neuron-specific vs. synapse-specific gating. Up to this point, we have considered the capacity of context-modular memory networks assuming that the desired active context has already been imposed externally. However, setting a particular contextual state requires additional control neurons which synapse onto the memory network in order to gate neurons and synapses. We now investigate how many additional neurons and connections are needed to implement each contextual gating scheme to determine if they are feasible. We consider the capacity of a memory network of $N = 10{,}000$ neurons with additional "control" neurons, denoted M, each with on the order of N postsynaptic connections (Fig. 4).

Neuron-specific control could in principle be implemented with a strategically-placed inhibitory synapse onto each neuron of the memory network (Fig. 4A, left; see Discussion for biological implications). Typically, to realistically code for s random configurations, the number of control neurons scales with s[27] — e.g., a winner-take-all architecture with a group of m control neurons per context would require a total of $M = ms$ control neurons (Fig. 4, light blue), and in the perceptron case $M = s/2$ (not shown). However, given the complex and nonlinear nature of real, biological neurons (e.g., single neurons may behave more like multi-layer neural networks[50]), there may be more efficient algorithms for single neuron control. If we assume the most compressed code for context (requiring that each context has a unique representation, and 2^M activity

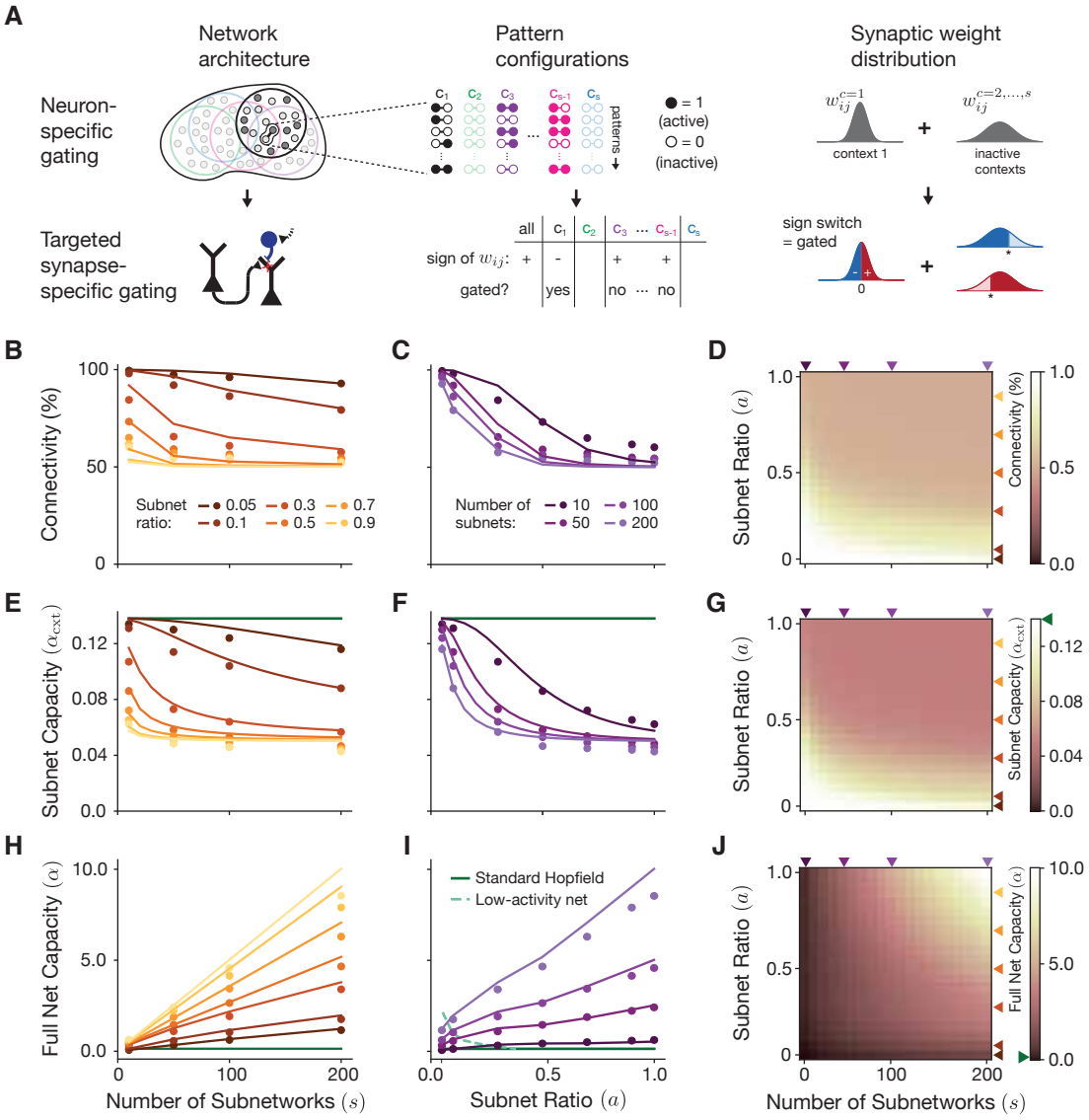

FIG. 3. Targeted synapse-specific (TaSS) gating further enhances memory capacity. A, Schematic of TaSS gating (left). The sign of the synaptic weight for a given pair of neurons (i, j) is compared with the sign of the hypothetical weight considering pattern configurations of each individual context (middle). Connections are gated if the sign changes. The proportion of gated weights (network connectivity) is estimated analytically (right, Methods). **B-D**, Resulting network connectivity following TaSS gating, as a function of number of contexts, s, over fixed subnetwork size, a (**B**), and vice versa (**C**), and over all parameters (**D**). Lines plotted in **B,C** are slices through **D** indicated by the colored triangles. **E-J**, Numerical and theoretical capacity estimation for combined subnetwork and targeted synapse-specific gating for a single contextual configuration ($\alpha_{\mathrm{cxt}} = p/N_{\mathrm{cxt}}$; **E-G**) and for the full network ($\alpha = sp/N = \alpha_{\mathrm{cxt}} sa$; **H-J**), plotted as in **B-D**. Comparison is made with standard Hopfield network (green), and low-activity variant (dashed turquoise, **I**).

patterns can be generated with M neurons), a total of s different contextual subnetworks would require input from a network of $M = \log_2(s)$ neurons at minimum (lower bound, Fig. 4, dark blue; see Methods). For finite-sized networks, neuron-specific gating requires a non-negligible amount of extra control neurons — a network with $N = 10{,}000$ neurons and $s = 200$ contexts requires 8, 100, or 4000 control neurons, considering the lower bound, perceptron-case and winner-take-all ($m = 20$) representations, respectively (Fig. 4B-G). Therefore, even assuming the worst case of our assumptions (4000 extra control neurons), the overall memory capacity (now at $\alpha \approx 0.86$ instead of $\alpha \approx 1.2$) is still over six times larger than the Hopfield limit, and can feasibly be implemented given the known structure of cortical circuits.

Synapse-specific gating operates on the level of individual synapses (Fig. 4A, middle), and requires that each of the approximately N^2 synapses in the memory network has a corresponding contextual gating synapse for each of the s contexts, adding $N^2 s$ synapses to the network. Assuming that each neuron can synapse onto at most N other neurons, this would require $M = Ns$ additional control neurons if implemented naively (Fig. 4, dark orange). As a slightly more efficient solution modelling each synaptic gate as a perceptron, we arrive at a limit of $Ns/2$ (Fig. 4, light orange), meaning that 2,000,000 control neurons are needed for a network of size $N = 10{,}000$ with 200 contexts. These two scenarios thus require more context-encoding neurons than memory neurons (Fig. 4B-G), and diminish the capacity to levels below the Hopfield limit. Therefore, implementing synapse-specific gating with full and independent control of each synapse is likely both ineffective

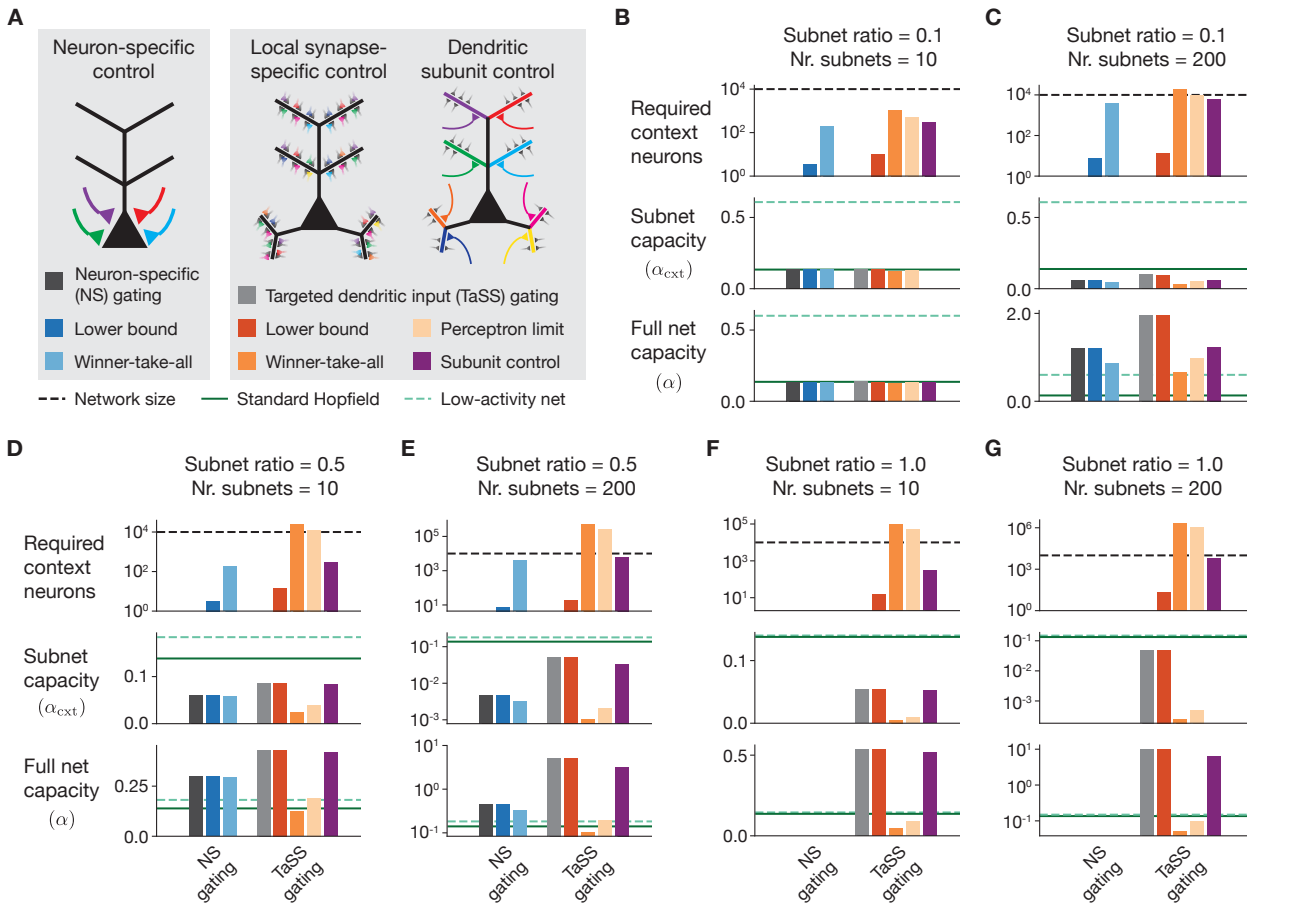

FIG. 4. Complexity of contextual gating schemes. **A**, Schematic of contextual control schemes for neuron-specific (NS) gating (left) and for targeted synapse-specific (TaSS) gating, either with local control (middle) or subunit control (right). **B-G**, Histograms of the number of required context neurons (top), the corrected subnet capacity (middle), and the corrected full net capacity (bottom) for six example parameter sets (assuming $N = 10{,}000$, black dashed line). Legend is given in panel **A**. Note that panels **F,G** are for $a = 1$ and do not include NS gating. The numbers of context neurons are calculated as follows: NS gating baseline (0 context neurons; dark grey); lower bound for NS gating ($\log_2(s)$; dark blue), winner-take-all for NS gating ($20s$; light blue); TaSS gating baseline (0; light grey); lower bound for TaSS gating ($\log_2(Nsa^2)$; red); winner-take-all for TaSS gating (Nsa^2; orange); perceptron limit for TaSS gating ($\frac{1}{2}Nsa^2$; light orange); dendritic subunit control ($30s$; purple). Comparisons made with standard Hopfield network (green solid lines) and low-activity network with same activity level ($a_{LA} = a/2$; turquoise dashed lines).

and infeasible. For completeness, we also posit that each synapse is capable of implementing an arbitrary nonlinear gating procedure, thereby reducing the number of control neurons to a theoretical limit of $\log_2(Ns)$ (Fig. 4, lower bound, red). Such a scheme would retain high memory capacity with relatively few control neurons — e.g., a network of 10,000 memory neurons and 200 contexts requires approximately 20 control neurons — but requires millions of post-synaptic connections per control neuron.

The complexity of synapse-specific gating can be simplified considering the extended morphology of the typical pyramidal cell, enabling control on a dendritic subunit level (Fig. 4A, right). Recent theoretical work has estimated that cortical pyramidal cells may have up to 30 dendritic subunits which can function as quasi-independent electrical compartments[51]. Therefore, if synapses are distributed across these different subunits, it is plausible that synapse-specific gating can be controlled with on the order of 30 contextual synapses, with one on each branch. Considering s contextual states, the network would then require approximately $30s$ control neurons. Such a gating implementation keeps the required neurons consistently below the network size, and so the memory capacity, though reduced, still peaks far above the Hopfield limit (now at $\alpha \approx 6.25$ instead of $\alpha \approx 10.0$; Fig. 4B-G, purple). However, in order for branch-specific gating to scale to many contexts, synapses may need to be clustered (see Discussion). Nevertheless, even if the complexity of such context-modular architectures reduce the capacity gains, they have additional benefits for memory organization in stability, retrieval and continual learning, as discussed below.

Memory stability is modulated by context. For the validity of the results presented thus far, it is crucial that as recall occurs in the active context, memories belonging to inactive contexts remain gated. We used an established measure of stability in order to obtain an estimate of the size of each pattern's basin of attraction, i.e., the area of neural activity space around a memory pattern for which the activity will converge to the memory[52] (Fig. 5A; Methods). Memories in the active context exhibit only a modest decrease in average basin of attraction size compared with the standard Hopfield model for most parameter ranges of neuron-specific gating (Fig. 5B; similar for TaSS gating, not shown). For comparison, the low-activity Hopfield network exhibits a substantial reduction in

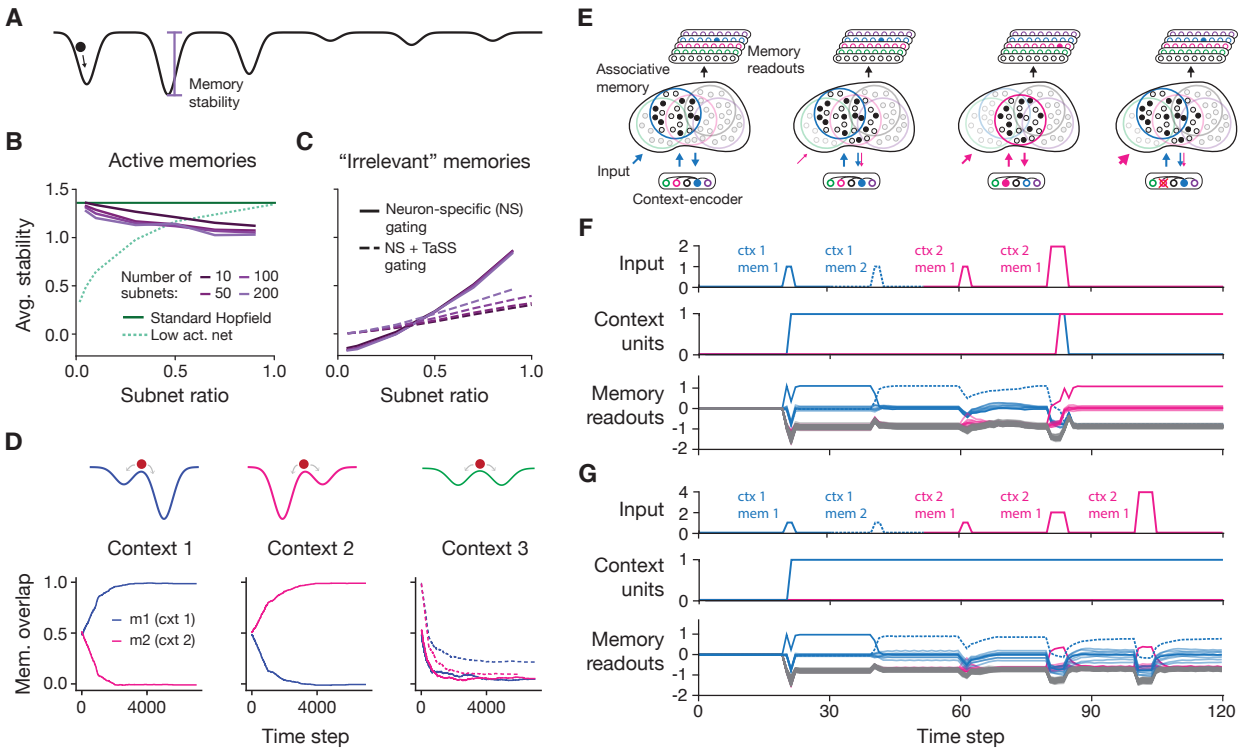

FIG. 5. Memory stability, dynamic switching, and "loss" of memory. **A**, Schematic of memory stability: network state (black ball) evolves according to an energy landscape. Memory stability numerically measures relative size of the basin of attraction for each memory (Methods). **B**, Average memory stability for "active" memories with neuron-specific (NS) gating, for different numbers of subnetworks, s, as a function of relative subnetwork size, a, and compared with standard (green line) and low-activity Hopfield networks (turquoise dashed line). Networks with targeted synapse-specific (TaSS) gating show similar stability levels (not shown). **C**, Average memory stability of "irrelevant" memories (patterns from non-active contexts) for networks with NS gating only (full lines), and including TaSS gating (dashed lines). **D**, Dynamic gating in networks with TaSS gating. Dynamics of memory overlap are shown for two memories belonging to different contexts (cxt 1 and cxt 2) when the network is initialized approximately equidistant from the two memory states for context 1 (left), context 2 (middle) and a third context for which the two patterns do not belong (right). Dotted lines (right panel) show memory overlap when the network begins in each of the two memory states. **E**, Schematic of dynamic gating in networks with NS gating, recurrently connected to a winner-take-all context-encoding network with one unit per context. Linear readouts measure memory activation, one per memory in each context. **F**, Example simulation of dynamic memory control (as illustrated in panel **E**). Top: four inputs are given to the network: two memories from context 1 (solid and dashed blue lines), followed by one memory from context 2, at two different amplitudes (magenta). Middle, bottom: inputs from memories of context 1 (blue) cause the corresponding context and memory readout to activate. The input for the memory from context 2 requires stronger amplitude to switch the context and memory readout (*magenta*). Memory readouts for other contexts are shown in gray. Simulation parameters: $N = 1000$, $N_{\text{cxt}} = 200$, $s = 10$, $p = 10$. **G**, Example simulation showing "loss" of memory (same network and parameters as in panel **F**). Memories from context 2 remain inaccessible when the corresponding context unit is blocked.

stability as the activity level decreases (Fig. 5B, turquoise).

The average stability of "irrelevant" patterns (memories belonging to inactive contexts) increases with the ratio of subnetwork size, but is always lower than that of active patterns (Fig. 5C). For large subnetwork ratios, TaSS gating improves performance by making irrelevant patterns less stable (Fig. 5C, dashed lines). Interestingly, for small subnetwork ratios ($a \approx 0.4$ or less), neuron-specific gating without TaSS gating suppresses irrelevant memories more effectively (Fig. 5C), and even goes to negative values, suggesting that the network is actively repelling these memory states. Therefore, the relative stability of neuron-specific gating alone versus TaSS gating depends upon the subnetwork ratio. Notably, each gating scheme is more stable than the other in high-capacity parameter ranges (cf. Fig. 2E and Fig. 3I). This modulation of stability biases the network dynamics depending on the background context, such that even for the same initial condition the activity state can be pushed towards or away from stored patterns (Fig. 5D).

Controlling memory expression via dynamic context switching. Though it may be beneficial that some memories be inaccessible at any given time, we hypothesized that a strong input corresponding to a memory in an inactive context could cause a switch to that context. We thus consider the complete memory architecture by including an additional context-encoding network which dynamically interacts with the memory network (Fig. 5E). Substantial evidence exists for a representation of context in the brain[1,46], providing experimental support for such an architecture.

For the purposes of simplicity, we model the context-encoding network with winner-take-all dynamics. Importantly, this network has reciprocal connections with the associative memory network – the active context unit provides inhibition to neurons outside of its corresponding contextual subnetwork (Methods). Correspondingly, this unit then receives excitation from neurons inside of the active subnetwork. This loop keeps the current context and memory active, and prevents other areas of the network from being activated (Fig. 5E).

We tested the functionality of this network in an example simulation using neuron-specific gating (Fig. 5F). Here, two

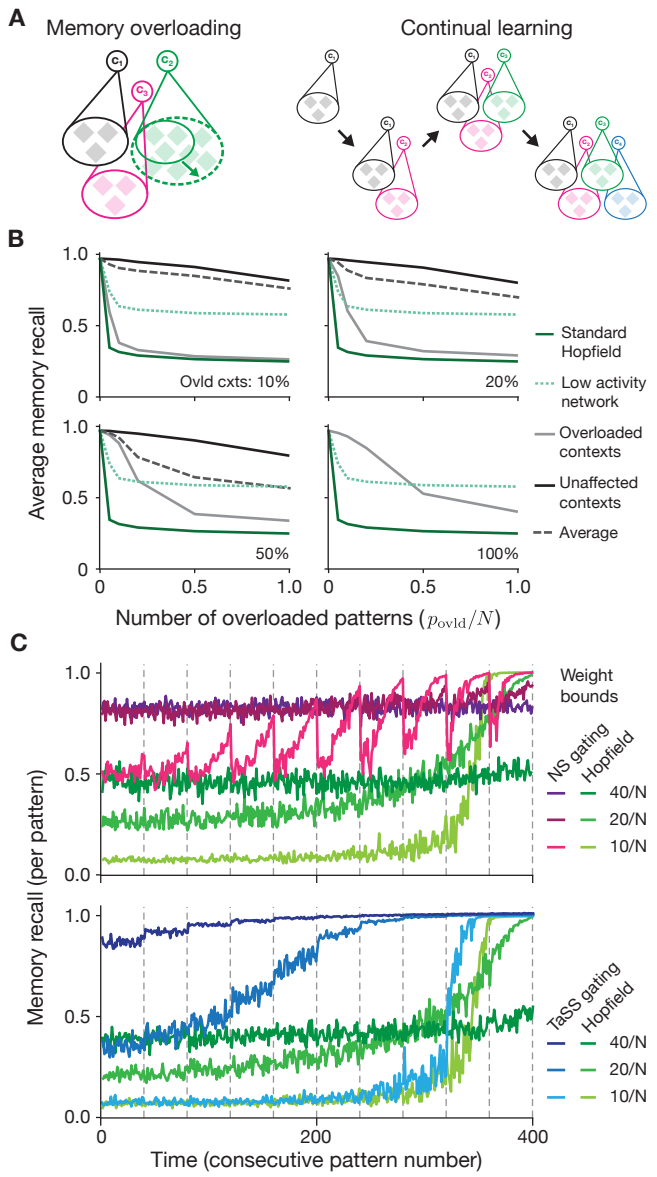

FIG. 6. Memory overloading and continual learning. **A**, In memory overloading (left), extra memory patterns are stored in a subset of contexts. In continual learning (right), contexts are learned sequentially, and synaptic weights are bounded to induce forgetting (Methods). **B**, Robustness to memory overloading. Average recall performance, as measured numerically by the average overlap across patterns and contexts (Methods) as a function of overload amount (p_{ovld}/N, where p_{ovld} is additional patterns stored after maximum capacity has been reached). Only a fraction of contexts are overloaded: 10% (top, left), 20% (top, right), 50% (bottom, left), and 100% (bottom, right). Average recall is plotted separately for overloaded contexts (gray), unaffected contexts (black), and averaged over all contexts (dashed gray), along with comparisons to the standard Hopfield network (green) and low-activity variant (dotted turquoise) with activity level $a/2$. Parameters: $N = 10{,}000$, $a = 0.1$, $s = 100$ (left). **C**,Continual learning with context-modular memory networks. Contexts are created sequentially after every 40 patterns ($p = 40$, $s = 10$ total, $N = 1000$; $\alpha_{cxt} = 0.04$; vertical gray dashed lines). Weights are clipped every time a new memory is added to the network ($w_{ij} \in [-A, A]$; A is indicated in the legends under *Weight bounds*; Methods). Memory recall is shown for each pattern in sequential order for neuron-specific (NS) gating (top) and targeted synapse-specific (TaSS) gating (bottom) versus the standard Hopfield network.

memories from one context were stimulated sequentially, and read out from the network. When a memory from a different context was probed, the context-encoding network suppressed recall. Only when a stronger input is given does the context switch, and the corresponding memory was successfully activated. Thus, this network architecture allows for dynamic memory gating, in which memory expression depends on context. We also explored the effects of context deficits in retrieval by inhibiting the representation of one context, and repeating the same experiment (Fig. 5G). In this case, the memories of the inhibited context became virtually inaccessible, despite the fact that the recurrent weights storing the memories were intact. This provides a potential model of cognitive control of memory access, as well as deficiencies in memory accessibility, such as has been hypothesized in some forms of amnesia[41,43] (see Discussion).

Memory overloading and continual learning. Associative memory networks are normally catastrophically affected by overloading the network with patterns beyond its capacity (Fig. 6A, left) – so-called *blackout interference*[35] (Fig. 6B, green). To probe this in context-modular networks, we first loaded networks with memories up to the maximum storage capacity. Then we chose a subset of contexts (10%, 20%, 50%, or 100%; Fig. 6B) and added additional memories to these configurations, leaving the others at their maximum capacity. When simulating recall, we observe that the architecture of the context-modular memory network makes it resistant to overloading when applied to a subset of contexts – synaptic weights are protected from interference when inactive. The extent of this effect depends upon the number of contexts and subnetwork ratio.

Despite the improved robustness to memory overloading, recall performance still declines as more memories are stored. During continual learning, when memories are added sequentially over time (Fig. 6A, right), catastrophic forgetting can be attenuated by preferentially remembering the most recently acquired memories through weight bounds (strong connections are clipped after each new memory is learned), which is also more biologically plausible[36]. In this scheme, more recent memories will have a large basin of attraction, with a gentle decay of recall performance for older memories[27,31]. We implemented sequential memory acquisition in context-modular memory networks with neuron-specific gating alone (Fig. 6C, top) and TaSS gating alone (Fig. 6C, bottom), over different bound sizes (Methods). Memory recall over time is substantially improved for both schemes, suggesting that contextual gating can also be used to enhance memory retention in continual learning. Our results thus illustrate the potential benefits of organizing memories around context for storage and retrieval over long timescales.

Arbitrary and shared context allocation. In reality, it is desirable that memory patterns be accessible in more than one context (Fig. 7A, top), but in our model thus far, each context contained a separate, non-overlapping set of memories (Fig. 1A). While neuron-specific gating creates fundamentally distinct representations for different contexts TaSS gating defines contexts at the level of active synapses rather than active neurons, and therefore can stabilize identical neuronal activity patterns in different contexts. To test this, we first trained a network with a large number of memory patterns (up to $16N$) in a single context in which all neurons and connections are utilized. Next, we assigned arbitrary (and overlapping) subsets of these memories to be part of new contexts, and determined the

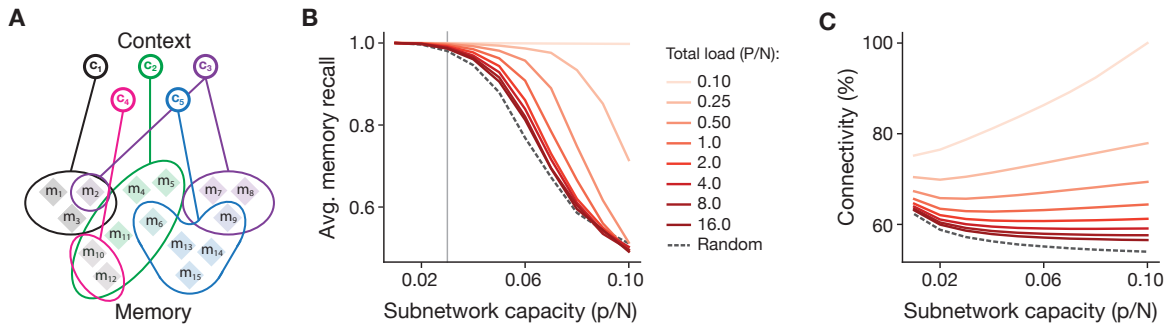

FIG. 7. Arbitrary and shared context allocation. **A**, Schematic of hierarchical memory assignment to context with variable size and overlap. **B**, Network performance as measured numerically by the average overlap across all patterns for the active context, plotted for various total loads, P/N, as a function of the single context capacity, p/N. Simulation with a random weight matrix is shown in the grey dashed line. $N = 1000$ for all simulations. **C**, Network connectivity (non-gated connections) of the simulations in **B**.

appropriate contextual configurations, i.e., which connections should be gated to enable successful memory recall (Methods; Fig. 1C, middle). This was possible, because all neurons remained available in each context, even if individual connections were turned off.

We observed that arbitrary sets of overlapping memory patterns can be reliably retrieved (Fig. 7B), provided that the number of patterns stored in each context is sufficiently small ($0.03N$ or less) – performance begins to degrade when the number of patterns per subnetwork approaches the subnetwork capacity (see Fig. 3E,F). As before, connectivity decreases as the total memory load is increased (Fig. 7C). Remarkably, even a network with a random weight matrix (see Methods) can successfully be used to represent stable context-dependent memories (Fig. 7B,C, dotted gray line), provided that the correct TaSS gating structure is imposed. This is because a random weight matrix will have, on average, half of its elements with the correct sign according to the correlation structure of the desired subset of patterns. This means that TaSS gating can be used to impose any activity pattern as a stable attractor independent of the weight matrix – i.e., groups of memories are stored in the TaSS gating structure. This also suggests that synaptic weights can be corrupted substantially without affecting performance, as long as the synapse-specific gating structure remains intact. Thus, TaSS gating not only allows the network to impose arbitrary and overlapping contextual states, but also produces high robustness to noisy synaptic weights.

Distributions of strengths over memories. Lastly, we extended the context-modular memory network to allow for memories to have different strengths in each context, defined by the memory stability, i.e., basin of attraction size. In other words, we changed the relative *ease* of recalling each memory without gating it completely on or off. Imposing distributions of memory strengths enables more flexible control. An interpretation of this scheme is to consider that the stability of a memory in a given context corresponds to how often this memory is recalled in that context. From this perspective, memory strength reflects the statistics of the external world, thereby enabling the brain to optimize memory access in order to affect behavior, e.g., making decisions more efficiently.

We use TaSS gating in a modified form, in which each particular pattern's stability (i.e., its basin of attraction, Fig. 5A) is manipulated to have a distinct size for each context (Fig. 8A; Methods). As before, we start with a standard Hopfield network whose weights are defined with the standard Hebbian learning rule (Methods), which generates memories with similar stability (Fig. 8B, black lines). Next, in contrast to normal TaSS gating in which a binary choice of (on and off) patterns is used to determine the targets of synaptic gating, patterns are multiplied by an analogue value representing relative strengths. Following this gating, we can impose arbitrary distributions of strengths over the set of memory patterns (Fig. 8B; colored lines, for four examples in top and middle). Note that no explicit learning of synaptic weights is needed to do so.

Such distributions of memory strengths can be used to create new contexts as combinations of previously-defined contextual states (Fig. 8A,B, bottom). This not only increases the flexibility of defining contextual states, but also opens the door towards multi-level hierarchical contextual control with contexts at different levels of specificity. For example, let's say you are deciding which food to eat on a trip to New York – you can bias your memory recall by first selecting memories related to New York, and then selecting memories related to food, thus leading to a new context with food-related memories in New York. As a simple test of network functionality with distributions of strengths, we probed the pattern completion ability of the network in response to noisy patterns (Fig. 8C, bottom left). Noise tolerance closely matches the level of stability of the memories (Fig. 8C, cf. Fig. 8B, bottom), provided the strength is above a certain baseline. For weak memory strengths, noise tolerance decreases to zero, as the memories themselves are in fact no longer stable (Fig. 8C, bottom right). Thus context-modular associative memory may serve as a powerful architecture to combine and distinguish groups of memories from one another.

Discussion

Memory and context are deeply intertwined[1,5,6]. To understand the properties and potential benefits of having context modularity in the brain, we have proposed a novel model of how context and associative memory interact, called the context-modular memory network (Fig. 1). The model provides a mechanistic hypothesis for the basis of context-dependence in recurrent neural circuits through neuron-specific and synapse-specific gating.

Relation to other models of associative memory and context dependence. Context-modular memory networks exhibit enhanced memory capacity (Fig. 2, Fig. 3 and Fig. 4) through the optimized use of sparsity, modularity and hierarchical organization – principles which have been used in previous associative memory models[27,34]. Modularity has been utilized to mimic the architecture of the cortex – dense local connectivity with sparse long-ranged connectivity[53–55] – and several other models have explored a hierarchical organization of memor-

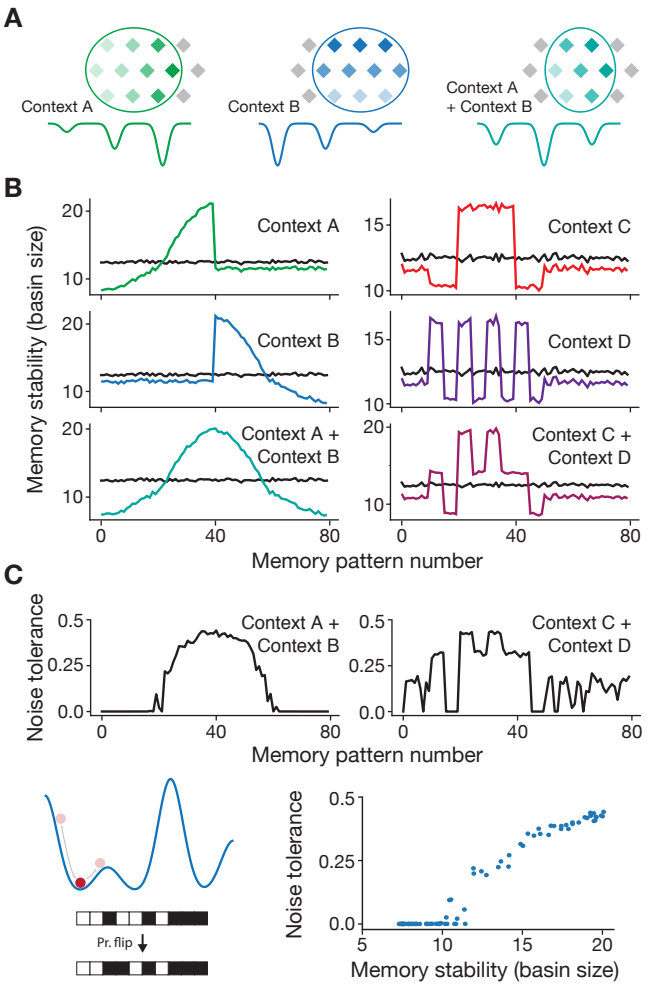

FIG. 8. Imposing distributions of memory stability. **A**, Stability levels of memories within the active context are set with a modified form of targeted synapse-specific (TaSS) gating (Methods). Contexts can be combined to achieve stability proportional to the sum of the stability levels in individual contexts (right). Diamond shaped points represent memory patterns. **B**, Distributions of memory strength and context combination for two example networks, with memory stability measured numerically ($N = 1000$; $p = 80$; Methods). Left: each context set half of the memories to have a monotonically increasing strength, and the other half to baseline. Right: half of the memories were assigned a strong or weak stability value, with the other half set to baseline. Comparison is made with a standard Hopfield network in which each memory has roughly the same stability (black lines). **C**, Noise tolerance of combined contexts in panel **B** measured numerically (top) by initializing the network in a noisy memory state and running the dynamics (schematic, bottom left). Noise tolerance is also plotted as a function of memory stability (bottom right).

ies, in which patterns with common features are stored in a common representation[27,45]. Our model bears a resemblance to these previous models, but is unique in considering context as the main determinant of memory groupings (and therefore sparsity), as well as modelling context as an external signal imposed on the memory network (in contrast to previous hierarchical models). In addition to the evidence for independent context representations in the brain[1], we chose to separate context because it facilitated the analytical estimation of memory capacity, and enabled memories to be shared by different contexts (as in Fig. 7). Furthermore, while we only considered a single hierarchical level between context and memory patterns, we speculate that additional levels could be added in a feedforward or looped architecture. Other models have included context-dependence into associative memory (e.g., Doboli *et al.* [56]), but with different architectures and motivations.

While context-modular networks display high memory capacity, more recent models also achieve similar capacity[27,31], albeit with differences in learning rules, activity, or architecture. Thus, the context-modular architecture should not be interpreted solely as a capacity booster — several other properties of these networks makes them more interesting, e.g., robustness to noise, robustness to memory overloading, and flexible memory access. Additionally, extensions to associative memory networks that improve memory capacity, such as low activity patterns[34], alternative learning rules[33], or complex synapses[57] may be incorporated into the context-modular memory network architecture (e.g., each subnetwork stores low-activity patterns), thus potentially combining their benefits.

We applied our model of context dependence to associative memory, but this architecture may be extended or adapted to other tasks such as context-dependent sensory processing (or multisensory integration[26]), decision making, or motor control, with non-attractor-like dynamics. For example, by applying a similar neuron-specific gating architecture, recent work has shown that problems with continual learning in deep artificial neural networks can be alleviated[37]. Our model also bears a resemblance to several behavioral models of temporal context[7,9], supported by experimental evidence for context as a slowly-drifting process over time[58,59]. Intriguingly, such models include a consistent means of introducing new contexts over time, which could be implemented at the mechanistic level in our model.

Circuit motifs and cell types involved in gating. There is substantial evidence for an input gating motif in the brain[60], implemented through detailed inhibitory control of network state[21,61], which has been linked to contextual processing and learning[18,60]. The diversity of inhibitory cell types and their post-synaptic targets provides a rich basis for such gating, with e.g., parvalbumin-positive interneurons preferentially targeting perisomatic regions and somatostatin-positive interneurons targeting dendritic regions[62]. Alternatively, neuron-specific gating could be implemented through excitatory control – e.g., neurons may be in a suppressed state by default, and only participate in recall if they receive extra contextual excitation – supported by recent work showing that baseline shifts modulate free recall[17]. Finally, recent evidence suggests that gain or excitability changes in individual neurons may play a role in memory allocation[63–66], and computational work has applied this idea to motor learning[22] and sequence learning[67]. Experimental evidence suggests that around 10 – 30% of neurons are allocated for a given engram in the amygdala and hippocampus[65], which would correspond to an area of high capacity in our model of neuron-specific gating.

Neuron-specific versus synapse-specific gating. Neuron-specific and synapse-specific gating have their advantages and limitations in terms of memory capacity, complexity, and flexibility. Synapse-specific gating is inherently more flexible, with many more degrees of freedom (neuronal gating can be seen as a special case of synaptic control, in which all synapses of a given neuron are gated), which is reflected in the larger gains in memory capacity (Fig. 3). However, such a scheme suffers from an expansion in complexity of control, as reflected in

the fact that "harmful" synapses need to be precisely targeted. Targeted synapse-specific gating thus requires many more (external) neurons to implement contextual control (Fig. 4), and it is unfeasible if each synapse is required to be fully and independently controlled (though some experimental evidence exists for individual inhibitory control of dendritic spines[68]).

Subunit (dendritic branch) specific control represents a more realistic option, lying somewhere on the continuum between neuron-specific and synapse-specific control. Recent theoretical work suggests that cortical pyramidal neurons may have around 30 independent subunits[51], which could be targeted according to contextual states. However, we note that having only 30 subunits compared to single-synapse control may limit the maximum number of achievable contexts. We speculate that the most efficient scheme may require taking advantage of multiple connections per pair of neurons[69] as well as clustering of synaptic inputs[70] based upon context. An additional benefit of synapse-specific gating is that it enables arbitrary context allocation (Fig. 7) and control of memory strength (Fig. 8). This flexibility may enable learning of a particular statistical distribution in the memory patterns, thus reflecting relationships between memory items in the world, or their relative values[44,71]. While we did not explore more flexible versions of subnetwork gating here, a more clever choice of subnetwork assignments could result in larger capacity or additional flexibility (e.g., shared memories by including more overlap between particular subnetworks).

The learning problem. Due to the complexity differences in subnetwork gating and targeted synapse-specific gating, these schemes will also likely require very different learning procedures. Learning the memory patterns themselves should be feasible considering the local, correlation-based Hebbian learning rule. Furthermore, random subnetwork gating could be imposed before learning the memory patterns, or indeed after memory patterns have been learned by removing neurons from the representation. Targeted synapse-specific gating relies upon a comparison between the overall synaptic weight of a pair of neurons and the hypothetical weight considering only a single context (Fig. 3), thus making it more complex. However, context may be a slow-changing process in time[72] with discrete shifts[73]. Therefore, the pair of neurons may be able to sample enough patterns to estimate their correlation within the currently active context and compare it with the anatomical weight value.

We found that synapse-specific gating endows a network with a tremendous amount of noise tolerance (Fig. 6), to the point where a random connectivity matrix can be used to retrieve memories provided the contextual configurations are maintained (Fig. 7). These results predict that local recurrent connections may vary quite substantially over time as learning occurs with little detriment to memory performance, consistent with recent data on the volitility of synaptic spines[74]. Given that contextual modulation is likely transmitted through inhibitory neurons, this idea resembles recent theoretical work proposing that inhibitory connectivity is responsible for maintaining information over time in the cortex[75].

In addition to short timescale learning dynamics, context-dependence may also vary over development. A previous experimental study suggested that infantile amnesia arises due to a retrieval failure[43]. In relation to our model, this may imply that a contextualization of memories may be learned and imposed only later on in life, resulting in early memories becoming inaccessible. However, other work suggests that contextual binding of memories decays over time[76], suggesting that some types of memories may also become more general over time.

Capacity vs. accessibility. Our model displays an inverse relationship between capacity and accessibility, as large increases in capacity are achievable provided that the vast majority of memories are not accessible at the same time. This trade-off may be viewed as a limitation of the model However, restricting memory access may also be seen as a benefit, considering that the main purpose of storing memories can be understood as to influence decision making[8,77–79]. We hypothesize that dynamic control of memory availability may act as an efficient means of *tree searching* through memories, enabling the brain to select which memories are currently relevant in order to make faster decisions. In this light, the incorporation of additional layers to such a contextual memory hierarchy may add further benefits. Overall, the integration of associative memory models with retrieval processes and decision making is a promising area of future research.

Furthermore, our model provides a direct mechanistic basis for memory failure due to loss of accessibility rather than forgetting[39]. Such a hypothesis has been put forth in the context of different types of amnesia[39,42,43]. Therefore, our model may have specific implications for the understanding of memory access in healthy and disease states.

Hippocampus, prefrontal cortex, and context representations. While we present the context-modular memory network as a generic architecture without explicit mention to brain areas and circuits, evidence suggests that contextual signals interact with cortical memories through the hippocampus, prefrontal cortex, and amygdala[1,13,80–84]. For example, a recent study finds evidence of inhibitory control of cortical memories through the hippocampus[13], providing direct support for a neuron-specific (Fig. 2) or synapse-specific (Fig. 3) gating motif in the cortex. Some work indicates that hippocampus and prefrontal cortex may play similar and complementary roles in memory retrieval – either that the hippocampus control recent memories and the prefrontal cortex more remote memories[85], or that prefrontal cortex handles active retrieval through executive control, and hippocampus handles more automatic retrieval[86]. Both hypotheses suggest that there may be multiple context-encoder-like networks in the brain. Alternatively, the modularization introduced here is a natural candidate mechanism for *pattern separation*, which is commonly attributed to the dentate gyrus in the hippocampus[87], and could act to control contextual memories in CA3[88] similar to previous models[56]. For each of these cases, our work functions as a useful conceptual model for how to begin studying the underlying circuits of each of these systems.

Materials and Methods

Detailed methods can be found below (after references).

Software and code availability

Code will be made available upon publication.

Acknowledgements

We thank Helen Barron, Adam Packer, João Sacramento, Andrew Saxe, Misha Tsodyks, and Friedemann Zenke for helpful comments at various stages of this work. This work was supported by a Sir Henry Dale Fellowship by the Wellcome Trust and the Royal Society (WT100000; WFP, EJA and TPV), a Wellcome Trust Senior Research Fellowship (214316/Z/18/Z; EJA and TPV), and a Research Project Grant by the Leverhulme Trust (RPG-2016-446; EJA).

Author contributions

WFP, EJA and TPV designed research; WFP carried out simulations and analysis; WFP, EJA and TPV wrote the manuscript.

Declaration of interests

The authors declare no competing interests.

References

[1] S Maren, KL Phan, and I Liberzon, "The contextual brain: implications for fear conditioning, extinction and psychopathology," Nature Reviews Neuroscience **14**, 417–428 (2013).

[2] V Mante, D Sussillo, KV Shenoy, and WT Newsome, "Context-dependent computation by recurrent dynamics in prefrontal cortex," Nature **503**, 78–84 (2013).

[3] SC Woolley, R Rajan, M Joshua, and AJ Doupe, "Emergence of context-dependent variability across a basal ganglia network," Neuron **82**, 208–223 (2014).

[4] AG Khan and SB Hofer, "Contextual signals in visual cortex," Current Opinion in Neurobiology **52**, 131–138 (2018).

[5] ME Bouton, "Context, time, and memory retrieval in the interference paradigms of pavlovian learning," Psychological Bulletin **114**, 80–99 (1993).

[6] SM Smith, "Remembering in and out of context," Journal of Experimental Psychology: Human Learning and Memory **5**, 460–471 (1979).

[7] MW Howard and MJ Kahana, "A distributed representation of temporal context," Journal of Mathematical Psychology **46**, 269–299 (2002).

[8] SM Polyn, KA Norman, and MJ Kahana, "A context maintenance and retrieval model of organizational processes in free recall," Psychological Review **116**, 129 (2009).

[9] SJ Gershman, M-H Monfils, KA Norman, and Y Niv, "The computational nature of memory modification," eLife **6**, e23763 (2017).

[10] BJ Levy and MC Anderson, "Inhibitory processes and the control of memory retrieval," Trends in Cognitive Sciences **6**, 299–305 (2002).

[11] HC Barron, TP Vogels, UE Emir, TR Makin, J O'shea, S Clare, S Jbabdi, RJ Dolan, and TEJ Behrens, "Unmasking latent inhibitory connections in human cortex to reveal dormant cortical memories," Neuron **90**, 191–203 (2016).

[12] HC Barron, TP Vogels, TEJ Behrens, and M Ramaswami, "Inhibitory engrams in perception and memory," Proceedings of the National Academy of Sciences **114**, 6666–6674 (2017).

[13] RS Koolschijn, UE Emir, AC Pantelides, H Nili, TEJ Behrens, and HC Barron, "The hippocampus and neocortical inhibitory engrams protect against memory interference," Neuron **101**, 528–541 (2019).

[14] H Tomita, M Ohbayashi, K Nakahara, I Hasegawa, and Y Miyashita, "Top-down signal from prefrontal cortex in executive control of memory retrieval," Nature **401**, 699–703 (1999).

[15] CD Gilbert and W Li, "Top-down influences on visual processing," Nature Reviews Neuroscience **14**, 350–363 (2013).

[16] S Zhang, M Xu, T Kamigaki, JPH Do, WC Chang, S Jenvay, K Miyamichi, L Luo, and Y Dan, "Long-range and local circuits for top-down modulation of visual cortex processing," Science **345**, 660–665 (2014).

[17] Y Norman, EM Yeagle, M Harel, AD Mehta, and R Malach, "Neuronal baseline shifts underlying boundary setting during free recall," Nature Communications **8**, 1301 (2017).

[18] KV Kuchibhotla, JV Gill, GW Lindsay, ES Papadoyannis, RE Field, TAH Sten, KD Miller, and RC Froemke, "Parallel processing by cortical inhibition enables context-dependent behavior," Nature Neuroscience **20**, 62–71 (2017).

[19] KC Wood, JM Blackwell, and MN Geffen, "Cortical inhibitory interneurons control sensory processing," Current Opinion in Neurobiology **46**, 200–207 (2017).

[20] DV Buonomano and W Maass, "State-dependent computations: spatiotemporal processing in cortical networks," Nature Reviews Neuroscience **10**, 113–125 (2009).

[21] A Holtmaat and P Caroni, "Functional and structural underpinnings of neuronal assembly formation in learning," Nature Neuroscience **19**, 1553–1562 (2016).

[22] JP Stroud, MA Porter, G Hennequin, and TP Vogels, "Motor primitives in space and time via targeted gain modulation in cortical networks," Nature Neuroscience **21**, 1774–1783 (2018).

[23] EJ Agnes, AI Luppi, and TP Vogels, "Complementary inhibitory weight profiles emerge from plasticity and allow attentional switching of receptive fields," bioRxiv , 729988 (2019).

[24] TP Vogels and LF Abbott, "Gating multiple signals through detailed balance of excitation and inhibition in spiking networks," Nature Neuroscience **12**, 483 (2009).

[25] F Edin, T Klingberg, P Johansson, F McNab, J Tegnér, and A Compte, "Mechanism for top-down control of working memory capacity," Proceedings of the National Academy of Sciences **106**, 6802–6807 (2009).

[26] GR Yang, JD Murray, and X-J Wang, "A dendritic disinhibitory circuit mechanism for pathway-specific gating," Nature Communications **7**, 12815 (2016).

[27] J Hertz, A Krogh, and RG Palmer, *Introduction to the theory of neural computation* (Addison-Wesley/Addison Wesley Longman, 1991).

[28] DO Hebb, *The organization of behavior: A neuropsychological theory* (Psychology Press, 1949).

[29] JJ Hopfield, "Neural networks and physical systems with emergent collective computational abilities," Proceedings of the National Academy of Sciences **79**, 2554–2558 (1982).

[30] G Bi and M Poo, "Synaptic modification by correlated activity: Hebb's postulate revisited," Annual Review of Neuroscience **24**, 139–166 (2001).

[31] R Chaudhuri and I Fiete, "Computational principles of memory," Nature Neuroscience **19**, 394–403 (2016).

[32] DJ Amit, H Gutfreund, and H Sompolinsky, "Storing infinite numbers of patterns in a spin-glass model of neural networks," Physical Review Letters **55**, 1530–1533 (1985).

[33] E Gardner, "The space of interactions in neural network models," Journal of Physics A: Mathematical and General **21**, 257–270 (1988).

[34] MV Tsodyks and MV Feigelman, "The enhanced storage capacity in neural networks with low activity level," Europhysics Letters **6**, 101–105 (1988).

[35] DJ Amit, *Modeling brain function: The world of attractor neural networks* (Cambridge university press, 1992).

[36] S Fusi and LF Abbott, "Limits on the memory storage capacity of bounded synapses," Nature Neuroscience **10**, 485–493 (2007).

[37] NY Masse, GD Grant, and DJ Freedman, "Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization," Proceedings of the National Academy of Sciences **115**, E10467–E10475 (2018).

[38] G Zeng, Y Chen, B Cui, and S Yu, "Continual learning of context-dependent processing in neural networks," Nature Machine Intelligence **1**, 364–372 (2019).

[39] PW Frankland, SA Josselyn, and S Köhler, "The neurobiological foundation of memory retrieval," Nature Neuroscience **22**, 1576–1585 (2019).

[40] M Naim, M Katkov, S Romani, and M Tsodyks, "Fundamental law of memory recall," arXiv , 1905.02403 (2019).

[41] JC Hulbert, RN Henson, and MC Anderson, "Inducing amnesia through systemic suppression," Nature Communications **7**, 11003 (2016).

[42] DS Roy, S Muralidhar, LM Smith, and S Tonegawa, "Silent memory engrams as the basis for retrograde amnesia," Proceedings of the National Academy of Sciences **114**, E9972–E9979 (2017).

[43] A Guskjolen, JW Kenney, J de la Parra, B-rA Yeung, SA Josselyn, and PW Frankland, "Recovery of "lost" infant memories in mice," Current Biology **28**, 2283–2290 (2018).

[44] S Recanatesi, M Katkov, S Romani, and M Tsodyks, "Neural network model of memory retrieval," Frontiers in Computational Neuroscience **9**, 149 (2015).

[45] MV Tsodyks, "Hierarchical associative memory in neural networks with low activity level," Modern Physics Letters B **4**, 259–265 (1990).

[46] AP Yonelinas, C Ranganath, AD Ekstrom, and BJ Wiltgen, "A contextual binding theory of episodic memory: systems consolidation reconsidered," Nature Reviews Neuroscience **20**, 364–375 (2019).

[47] DJ Amit, H Gutfreund, and H Sompolinsky, "Information storage in neural networks with low levels of activity," Physical Review A **35**, 2293–2303 (1987).

[48] H Sompolinsky, "The theory of neural networks: The hebb rule and beyond," Heidelberg colloquium on glassy dynamics , 485–527 (1987).

[49] JL van Hemmen and R Kühn, "Nonlinear neural networks," Physical Review Letters **57**, 913–916 (1986).

[50] P Poirazi, T Brannon, and BW Mel, "Pyramidal neuron as two-layer neural network," Neuron **37**, 989–999 (2003).

[51] WAM Wybo, B Torben-Nielsen, T Nevian, and M-O Gewaltig, "Electrical compartmentalization in neurons," Cell Reports **26**, 1759–1773 (2019).

[52] LF Abbott, "Learning in neural network memories," Network: Computation in Neural Systems **1**, 105–122 (1990).

[53] CF Mari and A Treves, "Modeling neocortical areas with a modular neural network," Biosystems **48**, 47–55 (1998).

[54] A Renart, N Parga, and ET Rolls, "Associative memory properties of multiple cortical modules," Network: Computation in Neural Systems **10**, 237–255 (1999).

[55] AM Dubreuil and N Brunel, "Storing structured sparse memories in a multimodular cortical network model," Journal of Computational Neuroscience **40**, 157–175 (2016).

[56] S Doboli, AA Minai, and PJ Best, "Latent attractors: a model for context-dependent place representations in the hippocampus," Neural Computation **12**, 1009–1043 (2000).

[57] MK Benna and S Fusi, "Computational principles of synaptic memory consolidation," Nature Neuroscience **19**, 1697–170 (2016).

[58] A Rubin, N Geva, L Sheintuch, and Y Ziv, "Hippocampal ensemble dynamics timestamp events in long-term memory," eLife **4**, e12247 (2015).

[59] DJ Cai, D Aharoni, T Shuman, J Shobe, J Biane, W Song, B Wei, M Veshkini, M La-Vu, J Lou, SE Flores, I Kim, Y Sano, M Zhou, K Baumgaertel, A Lavi, M Kamata, M Tuszynski, M Mayford, P Golshani, and AJ Silva, "A shared neural ensemble links distinct contextual memories encoded close in time," Nature **534**, 115 (2016).

[60] JJ Letzkus, SBE Wolff, and A Lüthi, "Disinhibition, a circuit mechanism for associative learning and memory," Neuron **88**, 264–276 (2015).

[61] JS Isaacson and M Scanziani, "How inhibition shapes cortical activity," Neuron **72**, 231–243 (2011).
[62] R Hattori, KV Kuchibhotla, RC Froemke, and T Komiyama, "Functions and dysfunctions of neocortical inhibitory neuron subtypes," Nature Neuroscience **20**, 1199–1208 (2017).
[63] JH Han, SA Kushner, AP Yiu, CJ Cole, A Matynia, RA Brown, RL Neve, JF Guzowski, AJ Silva, and SA Josselyn, "Neuronal competition and selection during memory formation," Science **316**, 457–460 (2007).
[64] AJ Rashid, C Yan, V Mercaldo, HL Hsiang, S Park, CJ Cole, A De Cristofaro, J Yu, C Ramakrishnan, SY Lee, SY Lee, K Deisseroth, PW Frankland, and SA Josselyn, "Competition between engrams influences fear memory formation and recall," Science **353**, 383–387 (2016).
[65] SA Josselyn and PW Frankland, "Memory allocation: mechanisms and function," Annual Review of Neuroscience **41**, 389–413 (2018).
[66] M Pignatelli, TJ Ryan, DS Roy, C Lovett, LM Smith, S Muralidhar, and S Tonegawa, "Engram cell excitability state determines the efficacy of memory retrieval," Neuron **101**, 274–284 (2019).
[67] R Pang and AL Fairhall, "Fast and flexible sequence induction in spiking neural networks via rapid excitability changes," eLife **8**, e44324 (2019).
[68] DM Iascone, Y Li, U Sumbul, M Doron, H Chen, V Andreu, F Goudy, I Segev, H Peng, and F Polleux, "Whole-neuron synaptic mapping reveals local balance between excitatory and inhibitory synapse organization," bioRxiv , 395384 (2018).
[69] E Gal, M London, A Globerson, S Ramaswamy, MW Reimann, E Muller, H Markram, and I Segev, "Rich cell-type-specific network topology in neocortical microcircuitry," Nature Neuroscience **20**, 1004–1013 (2017).
[70] G Kastellakis, DJ Cai, SC Mednick, AJ Silva, and P Poirazi, "Synaptic clustering within dendrites: an emerging theory of memory formation," Progress in Neurobiology **126**, 19–35 (2015).
[71] S Romani, I Pinkoviezky, A Rubin, and M Tsodyks, "Scaling laws of associative memory retrieval," Neural Computation **25**, 2523–2544 (2013).
[72] MW Howard, "Temporal and spatial context in the mind and brain," Current Opinion in Behavioral Sciences **17**, 14–19 (2017).
[73] S DuBrow, N Rouhani, Y Niv, and KA Norman, "Does mental context drift or shift?" Current Opinion in Behavioral Sciences **17**, 141–146 (2017).
[74] G Mongillo, S Rumpel, and Y Loewenstein, "Intrinsic volatility of synaptic connections – a challenge to the synaptic trace theory of memory," Current Opinion in Neurobiology **46**, 7–13 (2017).
[75] G Mongillo, S Rumpel, and Y Loewenstein, "Inhibitory connectivity defines the realm of excitatory plasticity," Nature Neuroscience **21**, 1463–1470 (2018).
[76] BJ Wiltgen and AJ Silva, "Memory for context becomes less specific with time," Learning & memory **14**, 313–317 (2007).
[77] MC Anderson and C Green, "Suppressing unwanted memories by executive control," Nature **410**, 366–369 (2001).
[78] MN Shadlen and D Shohamy, "Decision making and sequential sampling from memory," Neuron **90**, 927–939 (2016).
[79] JE Kragel, GA Worrell, MR Sperling, RE Gross, BC Lega, BC Jobst, SA Sheth, KA Zaghloul, JM Stein, and MJ Kahana, "Distinct cortical systems reinstate content and context information during memory search," Submitted .
[80] C Xu, S Krabbe, J Gründemann, P Botta, JP Fadok, F Osakada, D Saur, BF Grewe, MJ Schnitzer, EM Callaway, and A Lüthi, "Distinct hippocampal pathways mediate dissociable roles of context in memory retrieval," Cell **167**, 961–972 (2016).
[81] B Liang, L Zhang, G Barbera, W Fang, J Zhang, X Chen, R Chen, Y Li, and D Lin, "Distinct and dynamic on and off neural ensembles in the prefrontal cortex code social exploration," Neuron **100**, 700–714 (2018).
[82] KZ Tanaka, A Pevzner, AB Hamidi, Y Nakazawa, J Graham, and BJ Wiltgen, "Cortical representations are reinstated by the hippocampus during memory retrieval," Neuron **84**, 347–354 (2014).
[83] P Rajasethupathy, S Sankaran, JH Marshel, CK Kim, E Ferenczi, SY Lee, A Berndt, C Ramakrishnan, A Jaffe, M Lo, C Liston, and K Deisseroth, "Projections from neocortex mediate top-down control of memory retrieval," Nature **526**, 653–659 (2015).
[84] J Gründemann, Y Bitterman, T Lu, S Krabbe, BF Grewe, MJ Schnitzer, and A Lüthi, "Amygdala ensembles encode behavioral states," Science **364**, eaav8736 (2019).
[85] PW Frankland and B Bontempi, "The organization of recent and remote memories," Nature Reviews Neuroscience **6**, 119–130 (2005).
[86] Y Miyashita, "Cognitive memory: cellular and network machineries and their top-down control," Science **306**, 435–440 (2004).
[87] JK Leutgeb, S Leutgeb, M-B Moser, and EI Moser, "Pattern separation in the dentate gyrus and ca3 of the hippocampus," Science **315**, 961–966 (2007).
[88] AF Lacagnina, ET Brockway, CR Crovetti, F Shue, MJ McCarty, KP Sattler, SC Lim, SL Santos, CA Denny, and MR Drew, "Distinct hippocampal engrams control extinction and relapse of fear memory," Nature Neuroscience **22**, 753–761 (2019).
[89] T Geszti, *Physical models of neural networks* (World Scientific, 1990).
[90] SM Ross, *Introduction to probability models* (Academic press, 2014).
[91] C Sanderson and R Curtin, "Armadillo: a template-based c++ library for linear algebra," Journal of Open Source Software **1**, 26 (2016).
[92] DJ Amit, H Gutfreund, and H Sompolinsky, "Statistical mechanics of neural networks near saturation," Annals of Physics **173**, 30–67 (1987).
[93] N Brunel, "Is cortical connectivity optimized for storing information?" Nature Neuroscience **19**, 749–755 (2016)

Methods of

Context-modular memory networks support high-capacity, flexible, and robust associative memories

William F. Podlaski, Everton J. Agnes, and Tim P. Vogels

Materials and Methods

Model formulation. The context-modular memory network model is a fully connected recurrent network of N binary neurons denoted V_i for neuron i, and taking values $\{0, 1\}$. The modular architecture defines s contextual states, each with a corresponding set of active neurons $N_{\text{cxt}} \leq N$, chosen uniformly at random. We use $\mathcal{S}_k$ to denote the set of active neurons in contextual state k. We also define $a = N_{\text{cxt}}/N$ as the ratio of subnetwork to full network size, which is also the probability that each unit takes part in any given context.

Furthermore, each contextual state also determines a particular set of active inputs per neuron. We explore two variants of this type of contextual control – one random and one algorithmically targeted (see section on *Targeted synapse-specific gating*). In the first case, active inputs are chosen randomly with probability $b = K/N_{\text{cxt}}$, such that each neuron receives $K \leq N_{\text{cxt}}$ inputs on average. We define a symmetric matrix C^k with elements $c^k_{ij} = c^k_{ji} = 1$ if connections ij and ji are present in contextual state k, i.e., $\Pr(c^k_{ij} = 1) = b$ and $\Pr(c^k_{ij} = 0) = (1-b)$. Given the symmetry of the standard Hopfield model, we only consider input configurations which are also symmetric.

Importantly, at any given time, only one contextual state is "active". Considering contextual state k is active, the dynamics of each unit in the network are defined as

$$V_i = H(h_i) = \begin{cases} H\left(\sum_j c^k_{ij} w_{ij} V_j - \theta^k_i\right) & \text{if } i \in \mathcal{S}_k, \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

where $H(\cdot)$ is the heaviside step function, h_i is the total input to unit i, w_{ij} is the synaptic weight between neurons i and j (symmetric), and θ^k_i is the threshold for neuron i when contextual state k is active (defined below).

The network stores p memory patterns per contextual state, making $P = sp$ total memories. Memory patterns are denoted $\eta^{\sigma\mu}_i$ as the configuration of neuron i for memory μ of state σ, taking values $\{0, 1\}$ with equal probability provided unit i is in $\mathcal{S}_\sigma$, and 0 otherwise. The connectivity matrix is defined using a variant of a "Hebbian" learning rule[29,34]

$$w_{ij} = \frac{8}{bN_{\text{cxt}}} \sum_{\sigma=1}^{s} \sum_{\mu=1}^{p} c^\sigma_{ij} \tilde{\eta}^{\sigma\mu}_i \tilde{\eta}^{\sigma\mu}_j, \tag{10}$$

where

$$\tilde{\eta}^{\sigma\mu}_i = \begin{cases} \eta^{\sigma\mu}_i - \frac{1}{2} & \text{if } i \in \mathcal{S}_\sigma, \\ 0 & \text{otherwise.} \end{cases} \tag{11}$$

Note that the factor $1/bN_{\text{cxt}}$ in Eq. 10 ensures that the average total synaptic input to each neuron is around unity regardless of the choice of b[48] (see *Signal-to-noise analysis* below). The threshold is defined as

$$\theta^k_i = \frac{1}{2} \sum_{j \in \mathcal{S}_k} c^k_{ij} w_{ij}. \tag{12}$$

This choice of threshold, along with the factor 8 outside of the sum in Eq. 10, ensures that this model is equivalent to the standard Hopfield model with $\{\pm 1\}$ units (see *Supplementary methods* for further elaboration). Note that this threshold is different for each neuron and also changes with the contextual state of the network, whereas the connectivity matrix remains constant, with some weights being effectively set to zero through the c^k_{ij} terms in Eq. 9.

The activity level, or coding level, of the patterns denotes the fraction of active neurons for any given pattern. We define two measures of activity level – per individual context, and for the network as a whole. The activity level per individual context is set to 1/2, since each pattern unit is chosen to be 0 or 1 with uniform probability. The activity level as defined from the perspective of the entire network is $a_{\text{LA}} = a/2$ ("LA" for low-activity, see the following section), where a is the relative subnetwork size. We note that for the sake of analysis, we impose that all subnetworks contain the same number of active neurons and store the same number of memory patterns, though this is not necessary in practice.

Comparison with standard Hopfield network and low-activity network. We compare the results obtained in this study with that of the standard Hopfield model[29,32], as well as a standard variant with low-activity patterns[34]. For these models, we consider a fully connected network of N neurons which store a set of P memory patterns. Again, units in these networks are binary and take on values $\{0, 1\}$ (see *Supplementary Methods* for a comparison of $\{0, 1\}$ and $\{\pm 1\}$ formulations). Patterns are denoted $\eta^\mu_i \in \{0, 1\}$ with $\Pr(\eta^\mu_i = 1) = a_{\text{LA}}$ and $\Pr(\eta^\mu_i = 0) = (1 - a_{\text{LA}})$, where a_{LA} is the activity level.

Dynamics of both models are as follows

$$V_i = H\left(\sum_{j=1}^{N} w_{ij}V_j - \theta_i - \theta_0\right), \tag{13}$$

where w_{ij} is the connectivity matrix defined using the "Hebbian" rule[34]

$$w_{ij} = \frac{1}{N}\sum_{\mu=1}^{N}\left(\eta_i^\mu - a_{\mathrm{LA}}\right)\left(\eta_j^\mu - a_{\mathrm{LA}}\right), \tag{14}$$

θ_i is a neuron-specific threshold, defined as

$$\theta_i = a_{\mathrm{LA}}\sum_j w_{ij}, \tag{15}$$

and θ_0 is a constant threshold defined as

$$\theta_0 = a_{\mathrm{LA}}(1-a_{\mathrm{LA}})^2 - a_{\mathrm{LA}}^2(1-a_{\mathrm{LA}}). \tag{16}$$

The standard Hopfield model is obtained by setting $a_{\mathrm{LA}} = 1/2$, which then makes $\theta_0 = 0$.

Several previous works have studied the theoretical memory capacity limits of these networks[32,34]. For the standard Hopfield model with a Hebbian learning rule, this is approximately $\alpha_{\mathrm{H}} = P/N \approx 0.138$ (in the zero temperature limit). For the low-activity model, capacity scales with the activity level as:

$$\alpha_{\mathrm{TF}}(a_{\mathrm{LA}}) = \frac{1}{2a_{\mathrm{LA}}|\ln a_{\mathrm{LA}}|}, \tag{17}$$

for $a_{\mathrm{LA}} \ll 1$ ("TF" stands for Tsodyks-Feigelman[34]). The comparison with the context-modular memory network is only relevant at intermediate activity levels, where this estimate does not hold. We thus use numerical simulations to obtain a more accurate estimate of memory capacity (Fig. S3).

Analytical capacity estimation. In the following, we adapt a signal-to-noise analysis and a heuristic mean-field theory of memory capacity for the standard Hopfield network[27,35,89] to the case of the context-modular memory network.

Wald's equation. We will make use of a result in statistics known as Wald's equation[90], which we summarize here. Consider a Binomial random variable $K \sim \mathcal{B}(N_K, p_K)$. Let Y be the sum of a sequence of independent identically distributed random variables X_i of length K, i.e., $Y = \sum_{i=1}^{K} X_i$. Assuming that each X_i is independent of K, then the mean and variance of Y can then be written as

$$\mathbb{E}[Y] = \mathbb{E}[K]\mathbb{E}[X] \tag{18}$$

and

$$\mathrm{Var}[Y] = \mathbb{E}[K]\mathrm{Var}[X] + (\mathbb{E}[X])^2\mathrm{Var}[K], \tag{19}$$

where we have dropped the index on X because each random variable X_i comes from an identical distribution.

Signal-to-noise analysis. We aim to estimate the stability of an arbitrary neuron i with respect to a particular pattern ν of context k, $\eta_i^{k\nu}$, which generalizes to ensuring that $H(h_i^{k\nu}) = \eta_i^{k\nu}$, where $h_i^{k\nu}$ is the total input to neuron i when the network's state is exactly at pattern ν of context k (i.e., this neuron will not change activity, given the input it receives at the stored pattern state). Plugging Eqs. 10 and 12 into Eq. 9, we obtain

$$h_i^{k\nu} = \sum_{j=1}^{N} c_{ij}^k w_{ij}\eta_j^{k\nu} - \theta_i^k = \sum_{j=1}^{N} c_{ij}^k w_{ij}(\eta_j^{k\nu} - \tfrac{1}{2}) \tag{20}$$

$$= \frac{8}{bN_{\mathrm{cxt}}}\sum_{j=1}^{N} c_{ij}^k\left[\sum_\sigma\sum_\mu c_{ij}^\sigma \tilde{\eta}_i^{\sigma\mu}\tilde{\eta}_j^{\sigma\mu}\right]\tilde{\eta}_j^{k\nu} \tag{21}$$

$$= 2\tilde{\eta}_i^{k\nu} + \frac{8}{bN_{\mathrm{cxt}}}\sum_{j=1}^{N}\sum_{\mu\neq\nu} c_{ij}^k\tilde{\eta}_i^{k\mu}\tilde{\eta}_j^{k\mu}\tilde{\eta}_j^{k\nu} + \frac{8}{bN_{\mathrm{cxt}}}\sum_{j=1}^{N}\sum_{\sigma\neq k}\sum_{\mu=1}^{N} c_{ij}^k c_{ij}^\sigma\tilde{\eta}_i^{\sigma\mu}\tilde{\eta}_j^{\sigma\mu}\tilde{\eta}_j^{k\nu} \tag{22}$$

The right two terms in Eq. 22 are two different "crosstalk" terms that may disrupt the stability of memories. The first one is analogous to the standard Hopfield crosstalk term[27], and accounts for other patterns stored in the *same* context. The second term accounts for patterns stored in *different* contexts. We expect patterns stored in the same versus different subnetworks to affect stability differently, which will become apparent in the next section. As shown in Eq. 3 and Eq. 6, if the desired state of neuron i is 1 ($2\tilde{\eta}_i^{k\nu} = 1$), then we see that this state will be stable providing that the sum of the two crosstalk terms are greater than -1.

The input described in Eq. 22 depends upon the actual value of $\tilde{\eta}_i^{k\nu}$, which complicates the analysis. In order to generalize to the case in which the desired state of the neuron can be either 0 or 1, we follow Hertz *et al.* [27] and multiply Eq. 22 by $-2\tilde{\eta}_i^{k\nu}$, turning

the first term into −1. We now study the effect of the two crosstalk terms by considering the quantity

$$C_i^{k\nu} \triangleq -\frac{16\tilde{\eta}_i^{k\nu}}{bN_{\text{cxt}}} \sum_{j=1}^{N} \sum_{\mu \neq \nu} c_{ij}^{k} \tilde{\eta}_i^{k\mu} \tilde{\eta}_j^{k\mu} \tilde{\eta}_j^{k\nu} - \frac{16\tilde{\eta}_i^{k\nu}}{bN_{\text{cxt}}} \sum_{j=1}^{N} \sum_{\sigma \neq k} \sum_{\mu=1}^{N} c_{ij}^{k} c_{ij}^{\sigma} \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu} \tilde{\eta}_j^{k\nu} \tag{23}$$

which is the two crosstalk terms from Eq. 22 multiplied by $-2\tilde{\eta}_i^{k\nu}$. We can now define the probability that any given bit will flip as

$$P_{\text{err}} = \Pr\left(C_i^{k\nu} > 1\right). \tag{24}$$

The aim is now to approximate the two terms in Eq. 23 with Gaussian random variables.

Let's start with the first term. We first notice that $c_{ij}^{k} \in \{0, 1\}$ will set some terms in the sum to zero. The four pattern variables, $\tilde{\eta}_i^{k\nu} \tilde{\eta}_i^{k\mu} \tilde{\eta}_j^{k\mu} \tilde{\eta}_j^{k\nu}$, combined with the total factor of 16 outside of the sum, act as a shifted and scaled Bernoulli random variable that takes on values ±1 with equal probability. This sum can be interpreted as a random sum of random variables of the form $Y = \frac{1}{bN_{\text{cxt}}} \sum_{i=1}^{K} (2X_i - 1)$, with $K \sim \mathcal{B}(N_{\text{cxt}}(p-1), b)$ and $X_i \sim \text{Bern}(\frac{1}{2})$. Since this is a sum of Bernoulli random variables, we know that it will take the form of a Binomial distribution. We can then use Wald's equation (see above section) to determine the statistics of this distribution. Given the symmetry of the Bernoulli random variables (±1), it is easy to see that $\mathbb{E}[X] = 0$ and thus $\mathbb{E}[Y] = 0$. We use Eq. 19 to estimate the variance of Y, first ignoring the factor $\frac{1}{bN_{\text{cxt}}}$, to obtain $\text{Var}[Y] = \mathbb{E}[K]\text{Var}[(2X-1)] = N_{\text{cxt}}(p-1)b$. Dividing by bN_{cxt}, we obtain $\text{Var}[Y] = \frac{p}{bN_{\text{cxt}}} = \frac{\alpha_{\text{cxt}}}{b}$.

The second term can now be approximated analogously to the first. It again takes the form $Z = \frac{1}{bN_{\text{cxt}}} \sum_{i=1}^{K} (2X_i - 1)$, but now with $K \sim \mathcal{B}(N_{\text{cxt}}(s-1)p, b^2a^2)$ and $X_i \sim \text{Bern}(\frac{1}{2})$. The reason for the form of K in this term is that now we have both c_{ij}^{k} and c_{ij}^{σ}, independent and each with non-zero probability b, as well as $\tilde{\eta}_i^{\sigma\mu}$ and $\tilde{\eta}_i^{\sigma\mu}$, each with non-zero probability a (see Eq. 11). The mean of the resulting distribution is again zero, but now the variance is $\text{Var}[Z] = \frac{(s-1)p}{N} a = \alpha a$.

Since the two Binomial distributions Y and Z are symmetric and feature a large number of trials, they are well approximated by Gaussian distributions. Plugging these two approximations into Eq. 23 gives us

$$\begin{aligned} C_i^{k\nu} &\approx \mathcal{N}\left(0, \frac{p}{bN_{\text{cxt}}}\right) + \mathcal{N}\left(0, \frac{(s-1)p}{N} a\right) \\ &= \mathcal{N}\left(0, \frac{\alpha_{\text{cxt}}}{b}\right) + \mathcal{N}(0, \alpha a) \qquad (25) \\ &= \mathcal{N}\left(0, \frac{\alpha_{\text{cxt}}}{b} + \alpha a\right) \qquad (26) \end{aligned}$$

We can now obtain a rough estimate of the memory capacity by ensuring that P_{err} remains low[27]. Alternatively, we can also fomulate the memory capacity in terms of the standard Hopfield model, in which the crosstalk takes the form of a single Gaussian centered around zero with variance $\alpha_{\text{H}} \approx 0.138$ (see Appendix). We thus set the variance of Eq. 26 equal to that of the standard Hopfield model

$$\frac{\alpha_{\text{cxt}}}{b} + \alpha a = \alpha_H. \tag{27}$$

Rearranging terms, and using the relationship $\alpha = \alpha_{\text{cxt}} sa$, we arrive at the relationship

$$\alpha_{\text{cxt}} = \frac{\alpha_{\text{H}}}{\frac{1}{b} + (s-1)a^2}. \tag{28}$$

This equation applies generally for cases of arbitrary numbers of contexts s, relative subnetwork size a and relative input size b. To obtain the expression for context modulation by neuron-specific gating only (Eq. 4), we set $b = 1$, and for context modulation by synapse-specific gating only (Eq. 7), we set $a = 1$. Finally, to obtain an expression for the full network capacity α, we need to multiply Eq. 28 by sa (Eq. 5 and Eq. 8).

Information content. The information content of the standard Hopfield model can be calculated as the total entropy (average Shannon information) across all patterns in the following way[33]. Considering that each neuron of each pattern is randomly chosen to be 0 or 1 with probability $\frac{1}{2}$, the entropy of each bit of each pattern is equal to the binary entropy function evaluated at $\frac{1}{2}$, which we will denote $H_{\text{B}}(\frac{1}{2})$. Multiplying this by the number of neurons in each pattern, and the number of patterns, we arrive at the total entropy across all patterns

$$I_{\text{H}} = NpH_{\text{B}}\left(\tfrac{1}{2}\right) = N^2\alpha_{\text{H}}. \tag{29}$$

This can be extended to the low-activity variant of the Hopfield network simply by replacing α_{H} in the equation above with the corresponding low-activity capacity for a particular activity level a_{LA}: $\alpha_{\text{TF}}(a_{\text{LA}})$ (see section *Comparison with standard Hopfield network and low-activity network*), and evaluating the binary entropy function at the activity level a_{LA}. Together, this means that the information content of the low-activity Hopfield network for activity level a_{LA} is

$$I_{\text{TF}} = NpH_{\text{B}}(a_{\text{LA}}) = N^2\alpha_{\text{TF}}(a_{\text{LA}})H_{\text{B}}(a_{\text{LA}}). \tag{30}$$

For the context-modular network with neuron-specific gating, we consider the following. For each subnetwork with N_{cxt} neurons, there are $2^{N_{\text{cxt}}}$ possible patterns, each equally likely, leading to an information content of

$$I_\mu = -\log_2\left(1/2^{N_{\text{cxt}}}\right) = N_{\text{cxt}}\log_2(2) \tag{31}$$

for each pattern. Multiplying this by the number of patterns, we obtain

$$I = spI_\mu = spN_{\text{cxt}}\log_2(2) = spN_{\text{cxt}}N^2/N^2 = N^2\alpha a. \tag{32}$$

We can write this as a ratio with the information content of the standard Hopfield network to obtain

$$\frac{I}{I_H} = \frac{N^2\alpha a}{N^2\alpha_{\text{H}}} = \frac{\alpha a}{\alpha_{\text{H}}}. \tag{33}$$

Note that this equation for information content also holds for targeted synapse-specific (TaSS) gating, with the only difference being that the memory capacity α is different with and without TaSS gating.

Mean-field theory. The memory capacity of the context-modular memory network was calculated using established mean-field methods[27,32,89]. See supplementary methods for details.

Targeted synapse-specific gating. Targeted synapse-specific (TaSS) gating was implemented algorithmically in the following way. Given a context-modular memory network with parameters N, s, a, and p, we define the full weight matrix $\mathbf{W}$ from Eq. 10. We also define the hypothetical *isolated weight matrix* for each individual context k considering only patterns assigned to that subnetwork as $\tilde{\mathbf{W}}^k$. The resulting weight matrix for each context after applying TaSS gating is

$$\mathbf{W}^k = \mathbf{D}^k \odot \mathbf{W}, \tag{34}$$

where $\mathbf{D}^k$ is a binary matrix of zeros and ones (with elements d_{ij}^k), defined as

$$d_{ij}^k = \begin{cases} 0 & \text{if } \left(w_{ij}\tilde{w}_{ij}^k\right) < 0 \\ 1 & \text{otherwise} \end{cases} \tag{35}$$

and $\odot$ is the Hadamard product (element-wise product). This algorithm ensures that for each context k, the sign of each weight w_{ij}^k will reflect the correlation between neurons i and j over patterns in context k, but not over all other patterns.

This selective input gating substantially complicates the analytical methods for memory capacity estimation. Considering Eq. 23, it is now the case that the two crosstalk terms are no longer independent. We instead propose a means of obtaining a rough estimate of the memory capacity with TaSS gating by comparing the network to results from Hopfield networks with binary weights[27,49] (see main text). Essentially, when overlap is low ($a < 0.1$ and $s < 10$), no connections are gated (full connectivity), and each subnetwork has a capacity close to the standard Hopfield network ($\alpha_H \approx 0.138$). However, when there is a large amount of overlap between contexts (e.g., $a > 0.5$ and $s > 100$), connectivity drops to 50%, and each subnetwork is well approximated by a Hopfield network with binary synapses ($\alpha_B \approx 0.1$[27]), scaled linearly by the sparsity (Eq. **??**). Therefore, to a first approximation, we can estimate the capacity in between these extremes by linearly interpolating between α_H and α_B as a function of network sparsity, $f(s,a)$, which denotes the estimated network connectivity following synapse-specific gating. This capacity estimate can be written as $\alpha_{\text{cxt}} \approx c_H\alpha_H + (1-c_H)\alpha_B$, where $c_H = 2f(s,a) - 1$ indicates how much the network behaves like a standard Hopfield network, and $(1-c_H) = 2(1-f(s,a))$ denotes how much the network behaves as the network with binary weights. This interpolation then can be linearly scaled by the sparsity, $f(s,a)$ to obtain

$$\alpha_{\text{cxt}} \approx f(s,a)\Big([2f(s,a)-1]\alpha_H + 2[1-f(s,a)]\alpha_{\text{B}}\Big), \tag{36}$$

The expected amount of sparsity, $f(s,a)$, for each contextual configuration can be estimated assuming that the weight distributions for a single context and across all contexts are Gaussian with zero mean and variances proportional to the amount of crosstalk that they contribute. Given this, the probability that a particular weight will be removed can be approximated as:

$$\Pr(w_{ij} \leftarrow 0) = \int_0^\infty \frac{1}{\sqrt{2\pi\sigma_1^2}}\exp\left(-\frac{x^2}{2\sigma_1^2}\right)\int_x^\infty \frac{1}{\sqrt{2\pi\sigma_A^2}}\exp\left(-\frac{y^2}{2\sigma_A^2}\right)dydx \tag{37}$$

$$= \frac{1}{2\pi}\arctan\left(\sigma_A/\sigma_1\right), \tag{38}$$

where σ_1^2 is the variance of the weight distribution for a single context, and σ_A^2 is the variance of the weight distribution across all contexts. Based on the mean field results described below (see Eq. 105), we can approximate σ_1^2 and σ_A^2 as $\alpha_{\text{cxt}}r$ and $\frac{1}{2}\alpha r_n(a+\alpha+a^2)$, respectively. We compare this estimate with numerical simulations in Fig. 3B,C, where $f(s,a) = 1 - \Pr(w_{ij} \leftarrow 0)$.

Inclusion of complexity factors in capacity estimation. We consider the addition of context-encoding neurons in the estimation of memory capacity in order to make a fairer comparison with other networks. To do so, we simply multiply the original capacity

equations (Eq. 1 and Eq. 2) by a factor $N/(N+M)$, where M is the number of context neurons. This converts the subnet capacity to

$$\tilde{\alpha}_{\text{cxt}} = \alpha_{\text{cxt}} \frac{N}{N+M} = \frac{p}{N_{\text{cxt}} + aM} \tag{39}$$

and the full network capacity to

$$\tilde{\alpha} = \alpha \frac{N}{N+M} = \frac{sp}{N+M}. \tag{40}$$

Importantly, we assume that each neuron should have on the order of N pre- and post-synaptic connections, with an absolute maximum of N for both cases. We refer to the main text as discussed in the results for more details about each individual case for M.

Numerical simulations. We briefly describe the details of all numerical simulations here. More information can be found in the supplementary methods. Code was written in C++ with the help of *Armadillo*, a linear algebra library[91].

Memory capacity was estimated numerically by building finite-sized networks ($N = 10000$) initialized with a set of random patterns, connectivity and thresholds as defined above in section *Model formulation*. Dynamics were run *synchronously* according to Eq. 9, i.e., all units were updated simultaneously. Synchronous dynamics were chosen for efficiency reasons despite potential convergence issues. To test stability of the patterns, we initialized the network in each memory state and simulated the dynamics until they either reached a steady-state or they reached the maximum number of allowed time steps (100). We then calculated the overlap of the network state with the original pattern state as:

$$m^{\sigma\mu} = \frac{4}{N_{\text{cxt}}} \sum_i \tilde{\eta}_i^{\sigma\mu} \left(V_i - \tfrac{1}{2}\right), \tag{41}$$

where $m^{\sigma\mu} \in [-1, 1]$ ($m^{\sigma\mu} = 1$ when the state is sitting exactly at the pattern, and $m^{\sigma\mu} = 0$ when it is uncorrelated with the pattern). This was done for all patterns in a particular subnetwork to obtain an average overlap $\bar{m}^{\sigma} = N_{\text{cxt}}^{-1} \sum_\mu m^{\sigma\mu}$. A small amount of noise was allowed in the overlap, such that memory retrieval was deemed successful as long as $\bar{m}^{\sigma} \geq 0.97$[92].

Binomial test of proportions. Due to the finite network size, random choices of patterns may have non-zero correlations, which affects recall performance (weight correlations scale as $1/\sqrt{N}$). This may lead to networks which are able to store a larger or smaller number of patterns (relative to network size) stably compared to a network storing patterns with absolutely zero correlations. We thus ran several trials for each configuration and treated the average overlap from each trial as an estimate of the parameter p_m from a Binomial distribution. The average overlap was corrected to fall between 0 and 1, such that the estimate of p_m was formulated as: $\hat{p}_m = N_{tr}^{-1} \sum_n \frac{1}{2}(i\bar{m}_n + 1)$, where N_{tr} is the total number of trials. To determine the maximum number of patterns, we used a Binomial test of proportions with test statistic

$$T = \frac{p_0 - \hat{p}_m}{\sqrt{p_0(1-p_0)/N_{tr}}}, \tag{42}$$

where $p_0 = 0.97$ is the cutoff value. The p-value for this test was set to 10%, such that the overlap was considered too low if $T > 1.281$. This allowed us to obtain a more accurate estimate of the average overlap with fewer trials. For all simulations shown here, $N_{tr} = 10$.

Estimating capacity from a single subnetwork. To further speed up numerical simulations, we also took advantage of the fact that all contextual configurations in the context-modular Hopfield network should be identical in memory storage properties. We therefore constructed a single contextual configuration and then mimicked the effect of the other contexts by adding noise to the weight matrix in the following way

$$w_{ij} = \frac{1}{N_{\text{cxt}}} \sum_{\mu=1}^{N_{\text{cxt}}} c_{ij} \tilde{\eta}_i^{\mu} \tilde{\eta}_j^{\mu} + \tfrac{1}{N_{\text{cxt}}} \left(\delta - \tfrac{p}{2}\right), \tag{43}$$

with

$$\delta = \begin{cases} \mathcal{B}(p, 0.5) & \text{with Pr} = a \\ 0 & \text{otherwise,} \end{cases} \tag{44}$$

where $\mathcal{B}(n_B, p_B)$ is a binomial distribution with parameters n_B and p_B.

We confirmed the equivalence of this method with the simulation of all contextual configurations, and the two methods correspond well for most parameter ranges (not shown).

Robustness to noise. Robustness to noise was measured using so-called *stability parameters*[52]. We adapt this notion here, and write the stability parameter of a particular pattern ν of a context k, for neuron i as

$$\kappa_i^{k\nu} = \frac{2h_i^{k\nu} \tilde{\eta}_i^{k\nu}}{|W|_i}, \tag{45}$$

where

$$|W|_i = \left(\sum_{j=1}^{N} W_{ij}^2 \right)^{1/2} \tag{46}$$

We average this across all neurons and patterns to obtain the average stability parameter κ.

Dynamic gating model. The dynamic memory gating model (Fig. 5E,F) featured an associative memory network with $N = 1000$ units (again denoted V_i). The network was composed of $s = 10$ contexts, each defined by a subnetwork of $N_{\text{cxt}} = 200$ neurons, storing $p = 10$ random patterns each. Note that we do not consider sparse connectivity (and so $b = 1$), and targeted synapse-specific gating was not applied to the network.

The memory network was connected to a second network of s context-encoding units, denoted c_k for unit k, corresponding to context k. Units in both networks are binary, taking on values of 0 and 1. Finally, we found it necessary to add a third component, which was a global inhibitory neuron, denoted by y, which provides inhibition proportional to the population activity in the memory network (similar to Brunel [93]). We also defined a linear readout for each memory of each context, denoted $z^{k\nu}$ for the readout for pattern ν of context k.

The connectivity matrix for the memory network, denoted w_{ij}^{MM} (for units i and j; M for memory), was defined according to the Hebbian rule in Eq. 10.

Connectivity from the memory network to the context-encoding network was defined as:

$$w_{kj}^{CM} = \begin{cases} 1.5/N_{\text{cxt}} & \text{if } j \in \mathcal{S}_k, \\ 0 & \text{otherwise,} \end{cases} \tag{47}$$

which means that memory units that belong to context k will excite context unit k. The connectivity from the context-encoding network back to the memory network was defined as

$$w_{ik}^{MC} = \begin{cases} 4 \max\limits_{\nu} \left(\sum_j w_{ij}^{MM} \eta_j^{k\nu} \right) & \text{if } i \notin \mathcal{S}_k, \\ 0 & \text{otherwise.} \end{cases} \tag{48}$$

Context units will inhibit memory neurons outside of their corresponding subnetwork proportional to the maximum activation that these memory units receive across all patterns in that context (the factor of 4 was added for stability purposes). This ensures that these neurons remain in the suppressed state.

The recurrent connectivity of the context-encoding network was implemented as

$$w_{kl}^{CC} = \begin{cases} 0.4 & \text{if } k = l, \\ -\max_{\nu} \sum_{i=1}^{N} w_{ki}^{CM} \eta_i^{l\nu} & \text{otherwise,} \end{cases} \tag{49}$$

which means that a particular context unit l will inhibit other context units proportional to the maximum input they receive across all patterns in context l. This ensures that the dynamics in the context-encoding network are approximately winner-take-all. The global inhibitory neuron, y, has a synaptic weight of $w^{MG} = 0.05$ for all memory units.

Finally, output connectivity was defined as

$$w_{k\nu i}^{OM} = \eta_i^{k\nu} \tag{50}$$

Dynamics for the memory network were defined as:

$$V_i(t+1) = H\left(\sum_{j=1}^{N} w_{ij}^{MM} V_j(t) - \sum_{j=1}^{s} w_{ij}^{MC} c_j(t) - w^{MG} y(t) + I_i(t) \right), \tag{51}$$

where $H(\cdot)$ denotes the heaviside step function, $I_i(t)$ is the time-dependent input to unit i, and t denotes the time step. For the context-encoding network, dynamics were implemented as:

$$c_i(t+1) = H\left(\sum_{j=1}^{s} w_{ij}^{CC} c_j(t) + \sum_{j=1}^{N} w_{ij}^{CM} V_j(t) - \theta_{\text{c}} \right), \tag{52}$$

where θ_{c} is a threshold set to 0.5. Finally, the global inhibitory neuron, y, was implemented as a linear neuron, defined as:

$$y(t+1) = \frac{1}{N_{\text{cxt}}} \sum_{j=1}^{N} V_j(t). \tag{53}$$

Memory overloading and continual learning. Memory overloading experiments were carried out numerically for networks of size $N = 10000$. Networks were trained at maximum capacity, as determined numerically. Then, a subset of contexts, denoted as "overloaded" contexts, were allocated additional memories, divided equally among the overloaded contexts, and making

up 10%, 20%, 50% or 100% of the total number of contexts. Performance was assessed numerically by measuring the average memory recall, as described above. In Fig. 6B, performance is plotted as a function of "overload amount", which is the number of additional overloaded memories divided by the network size, such that an overload amount of 1 indicates that N additional memories past the maximum capacity were stored in the network. Comparisons are made with the standard Hopfield network and low-activity Hopfield network, in which additional memories were stored in the standard way.

Continual learning experiments were also done numerically, for networks of size $N = 1000$. In this setting, memories were trained sequentially (following Eq. 10, but with the sum over patterns only including a single pattern), with an additional clipping step (threshold parameter A) following the storage of each new pattern. The clipping step is described as follows:

$$w_{ij} = \begin{cases} A & \text{if } w_{ij} > A \\ -A & \text{if } w_{ij} < -A \\ w_{ij} & \text{otherwise} \end{cases} \tag{54}$$

Additionally, contexts were defined sequentially, and changed more slowly than the memory patterns themselves. Importantly, it was assumed that contexts were noiseless and did not deteriorate as new memories were added. Memory performance was again measured as above, and averaged over 20 independent trials.

Arbitrary context allocation. A context-modular memory network with TaSS gating and no neuron-specific gating ($a = 1$) was trained with an overall number of memory patterns, i.e., as a single context (Fig. 7). Following this, contexts were assigned by choosing an arbitrary group of memory patterns (with replacement), and the appropriated targeted gating configuration was determined as defined above (as in Eqs. 34 and 35). This was done for various numbers of memory patterns per context, and memory recall performance was measured numerically. Importantly, this means that a single memory pattern could be found in multiple contexts.

Distributions of memory strengths and noise tolerance. A modified TaSS gating scheme was devised in order to assign memory strengths for each pattern (Fig. 8). In a standard Hopfield network, all stored patterns have roughly equal stabilities (Fig. 8B, black lines). We then define a set of strong memories, whose stabilities are larger than the standard Hopfield network, and a set of weak memories, who stabilities are less than the standard Hopfield network. In the examples shown, one quarter of the memory patterns is strong, one quarter is weak, and one half remains close to the standard Hopfield network.

As a proof of concept, we generated two *classes* of distributions of strengths for the strong and weak memory patterns: one class with sequential strengths (contexts "A" and "B" in Fig. 7B,C, left) and another with discrete values (contexts "C" and "D" in Fig. 7B,C, right). The set of strong and weak memories in context k are denoted $\mathcal{T}_k^S$ and $\mathcal{T}_k^W$, respectively. Memories in the strong and weak groups were then assigned particular stability values, through the auxiliary variables x_μ^S and x_μ^W, respectively. These were then combined with the pattern values to obtain weight matrices $\tilde{S}^k$ and $\tilde{W}^k$ with elements

$$\tilde{s}_{ij}^k = \sum_{\mu \in \mathcal{T}_k^S} (x_\mu^S \tilde{\eta}_i^{k\mu})(x_\mu^S \tilde{\eta}_j^{k\mu}) \tag{55}$$

and

$$\tilde{w}_{ij}^k = \sum_{\mu \in \mathcal{T}_k^W} (x_\mu^W \tilde{\eta}_i^{k\mu})(x_\mu^W \tilde{\eta}_j^{k\mu}). \tag{56}$$

Finally, to obtain the mask for each context (as in Eq. 35), we compared these weight matrices with an overall weight matrix, W, obtained with the standard Hebbian learning rule applied to all memory patterns (Eq. 14):

$$d_{ij}^k = \begin{cases} 0 & \text{if } \left(\tilde{s}_{ij}^k \tilde{w}_{ij}^k\right) < 0 \ \ \& \ \ \left(w_{ij} \tilde{s}_{ij}^k\right) < 0 \\ 1 & \text{otherwise.} \end{cases} \tag{57}$$

In other words, a given synapse is gated ($d_{ij}^k = 0$) if $\text{sign}(\tilde{s}_{ij}) \neq \text{sign}(w_{ij})$, $\text{sign}(\tilde{w}_{ij}) = \text{sign}(w_{ij})$, and $\text{sign}(\tilde{s}_{ij}) \neq \text{sign}(\tilde{w}_{ij})$.

To obtain the distributions found in Fig. 8B, the following auxiliary values, x^S and x^W, were used. In Context "A", memories 1 to 20 were assigned to be weak, and memories 21 to 40 were assigned to be strong, with the remaining 40 memories being neutral. The stability values were set to

$$x_\mu^W = \begin{cases} 1 - \exp[-3\mu/20] & \text{for } \ \mu = 1, ..., 20. \\ 0 & \text{otherwise} \end{cases} \tag{58}$$

and

$$x_\mu^S = \begin{cases} 1 - \exp[-3(\mu - 20)/20] & \text{for } \ \mu = 21, ..., 40. \\ 0 & \text{otherwise.} \end{cases} \tag{59}$$

Context "B" featured the same memory stability values, but applied to different sets of memories: memories 41 to 60 were

assigned to be strong, and memories 61 to 80 were assigned to be weak:

$$x_\mu^S = \begin{cases} 1 - \exp[-3(\mu - 20)/20] & \text{for } \mu = 41, ..., 60. \\ 0 & \text{otherwise} \end{cases} \tag{60}$$

and

$$x_\mu^W = \begin{cases} 1 - \exp[-3\mu/20] & \text{for } \mu = 61, ..., 80. \\ 0 & \text{otherwise.} \end{cases} \tag{61}$$

Next, for context "C", memories 21 to 40 were assigned to be strong ($x_\mu^S = 1$), and memories 11 to 20 and 41 to 50 were assigned to be weak ($x_\mu^W = 1$). Finally, for context "D", memories 11 to 15, 21 to 25, 31 to 35 and 41 to 45 were assigned to be strong ($x_\mu^S = 1$), and memories 15 to 20, 25 to 30, 35 to 40 and 45 to 50 were assigned to be weak ($x_\mu^W = 1$).

Memory strength was measured using a stability parameter (Eq. 45). Noise tolerance was measured numerically by assessing memory recall performance for each pattern when the network is initialized in a noisy version the pattern. This noisy version was initialized by flipping each bit (neuron's activity) of the pattern with a probability f. The noise tolerance (as plotted in Fig. 8C) was defined as the maximum value of f for which the recall (averaged over 20 trials with random noisy pattern initializations) for a particular pattern became significantly less than 0.97 (as measured by binomial test of proportions; see *Binomial test of proportions*).

Supplementary methods of

Context-modular memory networks support high-capacity, flexible, and robust associative memories

William F. Podlaski, Everton J. Agnes, and Tim P. Vogels

Equivalence of {-1,+1} and {0,1} formulations. The $\{0,1\}$ formulation for the networks used in this study was designed such that it is equivalent to the standard $\{\pm 1\}$ formulation of the original Hopfield network[29]. To see this, it is sufficient to show that the dynamics in either case are equivalent. We consider the context-modular memory network as defined in section *Model formulation*, but with units $S_i \in \{\pm 1\}$, and patterns $\xi_i^{\sigma\mu} \in \{\pm 1\}$ within the active context ($S_i = 0$ and $\xi_i^{\sigma\mu} = 0$ for any unit i not contained within the subnetwork). The synaptic weights are defined as

$$J_{ij} = \frac{1}{bN_{\text{cxt}}} \sum_{\sigma} \sum_{\mu} c_{ij}^k \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} \tag{62}$$

and the dynamics of this network are described by the following equation (assuming context k is active)

$$S_i = \begin{cases} \text{sgn}\left(\sum_j c_{ij}^k J_{ij} S_j\right) & \text{if } i \in \mathcal{S}_k, \\ 0 & \text{otherwise.} \end{cases} \tag{63}$$

Now, starting from the dynamics of the $\{0,1\}$ network, it is simple to show the equivalence to Eq. 63 using the relationships between the two formulations: $2V_i - 1 = S_i$, $\tilde{\eta}^\mu = \frac{1}{2}\xi_i^\mu$, and $2H(x) - 1 = \text{sgn}(x)$ for all x. Assuming that context k is active, the dynamics of the $\{0,1\}$ fomulation for a particular unit i within subnetwork k follow

$$\begin{aligned}
V_i &= H\left(\sum_j c_{ij}^k w_{ij} V_j - \theta_i^k\right), \\
2V_i - 1 &= 2H\left(\sum_j c_{ij}^k w_{ij} V_j - \tfrac{1}{2}\sum_j c_{ij}^k w_{ij}\right) - 1, \\
S_i &= \text{sgn}\left(\sum_j c_{ij}^k w_{ij}\left(V_j - \tfrac{1}{2}\right)\right), \\
S_i &= \text{sgn}\left(\frac{8}{bN_{\text{cxt}}} \sum_j c_{ij}^k \sum_\sigma \sum_\mu \tilde{\eta}_i^{\sigma\mu} \tilde{\eta}_j^{\sigma\mu}\left(V_j - \tfrac{1}{2}\right)\right), \\
S_i &= \text{sgn}\left(\frac{1}{bN_{\text{cxt}}} \sum_j c_{ij}^k \sum_\sigma \sum_\mu \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} S_i\right), \\
S_i &= \text{sgn}\left(\sum_j c_{ij}^k J_{ij} S_j\right).
\end{aligned}$$

Thus, the deterministic dynamics of the two formulations are exactly the same. We note that this holds also for the stochastic version of the dynamics, in which $V_i = \sigma(h_i)$ and $S_i = \tanh(h_i)$, due to the identity $\tanh(x) = 2\sigma(x) - 1$. This is used in the following section.

Heuristic mean field analysis. In this section, we follow the heuristic mean-field analysis as described previously[27,89] and apply it to the context-modular memory network. We derive this here for the case of ± 1 units, as defined above (*Equivalence of* $\{-1,+1\}$ *and* $\{0,1\}$ *formulations*). Due to the equivalence with the $\{0,1\}$ formulation, this result holds for both cases.

Incorporating context-dependence into the mean-field theory. In order to consider context-dependence through neuron-specific gating and synapse-specific gating, we must appropriately incorporate them into the mean-field theory. For neuron-specific gating, as described in the main text, we can take the perspective of a single subnetwork k, and assume that the effect of the other contexts is just to add noise to the connectivity matrix relative to the subnetwork size $a = N_{\text{cxt}}/N$ and the number of contexts s. Thus, we do not need to do anything in addition to using the full weight matrix formulation as described in equation Eq. 62, which we separate into two terms here since we are taking the perspective of context k:

$$J_{ij} = \frac{1}{bN_{\text{cxt}}} \sum_\mu c_{ij}^k \xi_i^{k\mu} \xi_j^{k\mu} + \frac{1}{bN_{\text{cxt}}} \sum_{\sigma \neq k} \sum_\mu c_{ij}^\sigma \xi_i^{k\mu} \xi_j^{k\mu} \tag{64}$$

As for dendritic input gating, this is a bit trickier. Essentially, this creates a so-called diluted Hopfield network, with connection

sparsity controlled by the parameter $b = K/N$, the average number of inputs per neuron. It turns out that for a standard Hopfield network, symmetric dilution of synapses can be approximated by a fully-connected weight matrix with independent Gaussian noise added to each element with variance $N^{-1}\alpha_{\text{cxt}}(1-c)/c$, where N is the network size, $\alpha_{\text{cxt}} = p/N$ is the memory capacity, and c is the connection probability[48]. In this case, we can approximate the dilution in the first term in this way, to obtain

$$J_{ij} = \frac{1}{N_{\text{cxt}}} \sum_{\mu} \xi_i^{k\mu} \xi_j^{k\mu} + \frac{1}{bN_{\text{cxt}}} \sum_{\sigma \neq k} \sum_{\mu} c_{ij}^{\sigma} \xi_i^{k\mu} \xi_j^{k\mu} + \delta_{ij} \tag{65}$$

where δ_{ij} is the independent symmetric Gaussian noise with zero mean and variance $N_{\text{cxt}}^{-1}\alpha_{\text{cxt}}(1-b)/b$. The dilution in the second term should be treated differently. Since the second term only concerns patterns that we are not interested in recalling, the dilution does not add extra noise to the weights, but instead reduces the crosstalk effects. Thus we can treat this term as undiluted, but scaled by the amount of dilution b, since on average bN connections will be present. Thus we are left with:

$$J_{ij} = \frac{1}{N_{\text{cxt}}} \sum_{\mu} \xi_i^{k\mu} \xi_j^{k\mu} + \frac{b}{N_{\text{cxt}}} \sum_{\sigma \neq k} \sum_{\mu} \xi_i^{k\mu} \xi_j^{k\mu} + \delta_{ij}, \tag{66}$$

Setup. We consider a stochastic version of the Hopfield network, with updates to the units S_i taking the form:

$$\Pr(S_i = \pm 1) = \sigma_\beta(\pm h_i) = \frac{1}{1 + \exp(\mp 2\beta h_i)}, \tag{67}$$

where h_i is the input to neuron i, β is the inverse temperature parameter, which effectively controls the amount of noise in the updates, and σ_β is a variant of a sigmoid function, parameterized by β.

In mean field theory, we replace the true fluctuating input h_i by its average value $\langle h_i \rangle = \sum_j w_{ij} \langle S_j \rangle$, which allows us to compute the average activation of each neuron as

$$\langle S_i \rangle = \sum_{S_i = \{\pm 1\}} S_i \Pr(S_i) = \sigma(-2\beta h_i) - \sigma(2\beta h_i) = \tanh(\beta h_i) \tag{68}$$

We can then combine these two equations to get an expression for the average activation of each neuron as a function of the average activity of the rest of the network:

$$\langle S_i \rangle = \tanh(\beta \langle h_i \rangle) = \tanh\left(\beta \sum_j J_{ij} \langle S_j \rangle\right). \tag{69}$$

This allows us to describe the dynamics of the system using so-called order parameters. First of all, we use the two measures of capacity as defined in the main text, which are repeated here for convenience:

$$\begin{aligned} \alpha_{\text{cxt}} &= \frac{p}{N_{\text{cxt}}} && \text{(single context capacity)} \\ \alpha &= \frac{sp}{N} = \alpha_{\text{cxt}} sa && \text{(overall capacity)} \end{aligned} \tag{70}$$

where p is the number of patterns stored per context, N is the total network size, N_{cxt} is the number of active neurons per context, s is the number of contexts, and $a = N_{\text{cxt}}/N$ is the relative subnetwork size. Throughout the analysis, it is important to determine the nearness of the current network state to the different memory patterns, both in the current context, as well as across contexts. Consider that the currently active context is k. We use m_ν^k and n_μ^σ to denote the *overlap* between the average network state $\langle S_i \rangle$ and a particular pattern ν of context k ($\xi_i^{k\nu}$), or a pattern μ of context σ ($\xi_i^{\sigma\mu}$), respectively, when k is active:

$$m_\nu^k = \frac{1}{N_{\text{cxt}}} \sum_{i \in \mathcal{S}_k} \xi_i^{k\nu} \langle S_i \rangle \tag{71}$$

$$n_\mu^\sigma = \frac{1}{N_{\text{cxt}}} \sum_{i \in \mathcal{S}_k} \xi_i^{\sigma\mu} \langle S_i \rangle \tag{72}$$

where $\mathcal{S}_k$ is the set of active neurons in context k (as defined in the section *Model formulation*). Note that both m_ν^k and n_μ^σ are sums of N_{cxt} terms and are normalized by $1/N_{\text{cxt}}$. However, the $\xi_i^{\sigma\mu}$ terms of Eq. 72 may be zero depending upon the subnetwork size, so we expect Eq. 71 and Eq. 72 to have different variances. Importantly, we suppose that the network is close to one of the patterns, say pattern 1 of context k. Thus m_1^k will be of order unity, and the rest of m_ν^k's for $\nu \neq 1$ are small, of order $1/\sqrt{N_{\text{cxt}}}$. To see this, consider the fact that m_ν^k is a sum of N_{cxt} terms, each of which will be ± 1 with equal probability. This can be approximated by a zero-mean Gaussian with variance N_{cxt}. The normalizer $1/N_{\text{cxt}}$ makes the variance $1/N_{\text{cxt}}$, and thus the standard deviation $1/\sqrt{N_{\text{cxt}}}$. As for n_μ^σ, it is a sum of N_{cxt} terms, each of which will be non-zero with probability a, due to $\xi_i^{\sigma\mu}$. The non-zero terms will

then be ±1 with equal probability. This can be approximated by a Gaussian with variance N_{cxt}^2/N, which, divided by N_{cxt}, yields $1/N$. Thus the standard deviation is $1/\sqrt{N}$.

We next introduce r and r_n to denote the *mean square overlap* of the system configuration with the nonretrieved patterns in subnetwork k, and all other subnetworks, respectively:

$$r = \frac{1}{\alpha_{\text{cxt}}} \sum_{\nu \neq 1} \left(m_\nu^k\right)^2 \tag{73}$$

$$r_n = \frac{1}{\alpha} \sum_{\sigma \neq k} \sum_{\mu} \left(n_\mu^\sigma\right)^2. \tag{74}$$

Both of these quantities should be approximately of order unity. For the case of r, we have a sum of p terms, each with variance $1/N_{\text{cxt}}$, and normalized by $1/\alpha_{\text{cxt}} = N_{\text{cxt}}/p$. For r_n, we have a sum of $(s-1)p$ terms, each with variance of approximately $1/N$, and normalized by $1/\alpha = N/sp$. *Our task is now to get a self-consistent calculation of r, r_n, and m_1.*

Derivation Part A: finding an expression for r. We begin by plugging the mean field equations for $\langle S_i \rangle$ from Eq. 69 into equation Eq. 71, and expanding the weight matrix using the expression in Eq. 66:

$$m_\nu^k = \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \tanh\left(\frac{\beta}{N_{\text{cxt}}} \sum_{j \in T_k} \sum_\mu \xi_i^{k\mu} \xi_j^{k\mu} \langle S_j \rangle + \frac{\beta b}{N_{\text{cxt}}} \sum_{j \in T_k} \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} \langle S_j \rangle + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \tag{75}$$

We then rearrange terms and substitute the other m_μ^k and n_μ^σ terms into the equation:

$$m_\nu^k = \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \tanh\left(\beta \sum_\mu \xi_i^{k\mu} \frac{1}{N_{\text{cxt}}} \sum_{j \in T_k} \xi_j^{k\mu} \langle S_j \rangle + \beta b \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \frac{1}{N_{\text{cxt}}} \sum_{j \in T_k} \xi_j^{\sigma\mu} \langle S_j \rangle + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \tag{76}$$

$$= \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \tanh\left(\beta \sum_\mu \xi_i^{k\mu} m_\mu^k + \beta b \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} n_\mu^\sigma + \beta \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \tag{77}$$

Next, we pull out the terms with m_1^k and m_ν^k, and then use a trick to multiply the equation by $\xi_i^{k1} \xi_i^{k1} = 1$. We can put one of these terms inside of the tanh due to the fact that $\tanh(-x) = -\tanh(x)$:

$$m_\nu^k = \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \tanh\left(\beta \left(\xi_i^{k1} m_1^k + \xi_i^{k\nu} m_\nu^k + \sum_{\mu \neq 1,\nu} \xi_i^{k\mu} m_\mu^k + b \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} n_\mu^\sigma + \sum_{j \in T_k} \delta_{ij} \langle S_j \rangle \right) \right)$$

$$= \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\left(\beta \left(m_1^k + \xi_i^{k\nu} \xi_i^{k1} m_\nu^k + \sum_{\mu \neq 1,\nu} \xi_i^{k\mu} \xi_i^{k1} m_\mu^k + b \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_i^{k1} n_\mu^\sigma + \sum_{j \in T_k} \xi_i^{k1} \delta_{ij} \langle S_j \rangle \right) \right) \tag{78}$$

We then use another trick, and replace the right-hand side of Eq. 78 with the first two terms of the Taylor expansion with respect to m_ν^k, i.e., $f(m_\nu^k) = f(a) + f'(a)(m_\nu^k - a)$, where $f(m_\nu^k)$ is the right-hand side of Eq. 78. We assume that m_ν^k is small, of order $1/\sqrt{N}$, and so we take $a = 0$. This expansion yields:

$$m_\nu^k \approx \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right) + \beta m_\nu^k + \beta d m_\nu^k \tag{79}$$

where we use c_i to denote the crosstalk terms:

$$c_i = \sum_{\mu \neq 1} \xi_i^{k\mu} \xi_i^{k1} m_\mu^k + b \sum_{\sigma \neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_i^{k1} n_\mu^\sigma + \sum_{j \in T_k} \xi_i^{k1} \delta_{ij} \langle S_j \rangle \tag{80}$$

and

$$d = \frac{1}{N_{\text{cxt}}} \sum_i \tanh^2\left(\beta(m_1^k + c_i)\right) \tag{81}$$

Note that we write the crosstalk terms c_i to include all patterns except for pattern 1 of context k ($k1$). Based on the expression in Eq. 79, c_i should also exclude the term $\xi_i^{k\nu} \xi_i^{k1} m_\nu^k$, but this will be a negligible addition assuming large N and p. This term is left in for generality purposes, as the expression c_i will reappear later in the derivation. We now approximate d as an average of a function $f(z)$ of a Gaussian random variable z. Since N is large (and therefore N_{cxt} too), we can replace the average $\frac{1}{N_{\text{cxt}}} \sum_i \tanh^2$ with the integral over the distribution of z, which we will call q:

$$d \approx q = \int \frac{dz}{\sqrt{2\pi}\sigma} \exp\left(\frac{-(z - \bar{z})^2}{2\sigma^2} \right) f(z) \tag{82}$$

The mean and variance of this distribution will depend upon the architecture of the system, i.e., the choice of a, b, s, etc. There may also be several ways of devising appropriate approximations for the terms inside. We thus remain agnostic to this for the moment, and write the mean and variance as $m_1^k + \mu_*$ and σ_*^2, respectively. We are thus left with:

$$m_\nu^k = \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right) + \beta m_\nu^k + \beta q m_\nu^k \tag{83}$$

where

$$q = \int \frac{dz}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}z^2\right) \tanh^2\left(\beta(m_1 + \mu_* + \sigma_* z)\right). \tag{84}$$

Moving all terms with m_ν^k to the left, this can be rewritten as:

$$m_\nu^k = \frac{N_{\text{cxt}}^{-1} \sum_i \xi_i^{k\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right)}{1 - \beta(1-q)} \tag{85}$$

Finally, we can obtain an expression for r by squaring and averaging Eq. 85 (since this is the definition of r, see Eq. 73). First, squaring Eq. 85 yields

$$\left(m_\nu^k\right)^2 = \left(\frac{1}{1-\beta(1-q)}\right)^2 \frac{1}{N_{\text{cxt}}^2} \sum_{ij} \xi_i^{k\nu} \xi_i^{k1} \xi_j^{k\nu} \xi_j^{k1} \tanh\left(\beta(m_1^k + c_i)\right) \tanh\left(\beta(m_1^k + c_j)\right) \tag{86}$$

Then averaging, we get

$$\left\langle \left(m_\nu^k\right)^2 \right\rangle = \left(\frac{1}{1-\beta(1-q)}\right)^2 \frac{1}{N_{\text{cxt}}^2} \sum_{ij} \xi_i^{k1} \xi_j^{k1} \left\langle \xi_i^{k\nu} \xi_j^{k\nu} \right\rangle \tanh\left(\beta(m_1^k + c_i)\right) \tanh\left(\beta(m_1^k + c_j)\right) \tag{87}$$

We see that the only terms that are affected by the averaging are $\xi_i^{k\nu}$ and $\xi_j^{k\nu}$. It is easy to see that if $i \neq j$, these two terms are independent and so the expected value will be zero, which removes most terms from the sum. For $i = j$, both $\xi_i^{k\nu}\xi_j^{k\nu}$ and $\xi_i^{k1}\xi_j^{k1}$ will equal 1. This leaves us with

$$\left\langle \left(m_\nu^k\right)^2 \right\rangle = \left(\frac{1}{1-\beta(1-q)}\right)^2 \frac{1}{N_{\text{cxt}}} \sum_i \tanh^2\left(\beta(m_1^k + c_i)\right). \tag{88}$$

Lastly, we can replace the sum of $\tanh^2$ functions with the same q variable defined in Eq. 84, to get an expression for r:

$$r = \left\langle \left(m_\nu^k\right)^2 \right\rangle = \frac{q}{[1-\beta(1-q)]^2}. \tag{89}$$

Note that this expression is equivalent to the expression for r in the standard Hopfield model, except that q has changed.

Derivation Part B: finding an expression for r_n. We again start by using the mean field equations Eq. 69, but now plug them in to Eq. 72 for a particular n_ν^ϕ. We again rearrange terms so as to plug m_μ^k and n_μ^σ into the equation:

$$n_\nu^\phi = \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{\phi\nu} \tanh\left(\frac{\beta}{N_{\text{cxt}}} \sum_{j=1}^{N} \sum_\mu \xi_i^{k\mu} \xi_j^{k\mu} \langle S_j\rangle + \frac{\beta b}{N_{\text{cxt}}} \sum_{j\in T_k} \sum_{\sigma\neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_j^{\sigma\mu} \langle S_j\rangle + \beta \sum_{j\in T_k} \delta_{ij} \langle S_j\rangle\right) \tag{90}$$

$$= \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{\phi\nu} \tanh\left(\beta \sum_\mu \xi_i^{k\mu} \frac{1}{N_{\text{cxt}}} \sum_{j\in T_k} \xi_j^{k\mu} \langle S_j\rangle + \beta b \sum_{\sigma\neq k} \sum_\mu \xi_i^{\sigma\mu} \frac{1}{N_{\text{cxt}}} \sum_{j\in T_k} \xi_j^{\sigma\mu} \langle S_j\rangle + \beta \sum_{j\in T_k} \delta_{ij} \langle S_j\rangle\right) \tag{91}$$

$$= \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{\phi\nu} \tanh\left(\beta \sum_\mu \xi_i^{k\mu} m_\mu^k + \beta b \sum_{\sigma\neq k} \sum_\mu \xi_i^{\sigma\mu} n_\mu^\sigma + \beta \sum_{j\in T_k} \delta_{ij} \langle S_j\rangle\right) \tag{92}$$

We then follow the same steps as the previous section, by pulling out the terms with m_1^k and n_ν^ϕ, multiplying the equation by $\xi_i^{k1}\xi_i^{k1} = 1$, and finally taking the Taylor expansion with respect to n_ν^ϕ around zero, to get:

$$n_\nu^\phi \approx \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{\phi\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right) + \beta n_\nu^\phi + \beta d n_\nu^\phi, \tag{93}$$

where c_i and d are as defined above in Eq. 80 and Eq. 81, respectively. The reason for the approximate equality is because c_i in this case does not include pattern $\phi\nu$. Again, because we are considering the case of large N and p, this change is negligible. We then

approximate d using the very same method as before in Eq. 82 and Eq. 83. We can thus approximate Eq. 93 with the following:

$$n_\nu^\phi \approx \frac{1}{N_{\text{cxt}}} \sum_i \xi_i^{\phi\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right) + \beta n_\nu^\phi + \beta q n_\nu^\phi \tag{94}$$

$$= \frac{N^{-1} \sum_i \xi_i^{\phi\nu} \xi_i^{k1} \tanh\left(\beta(m_1^k + c_i)\right)}{1 - \beta(1-q)} \tag{95}$$

where q is defined as in Eq. 84. Finally, following Eq. 86 and Eq. 87, we can obtain an expression for r_n by squaring and averaging Eq. 95. This yields

$$\left\langle \left(n_\nu^\phi\right)^2 \right\rangle = \left(\frac{1}{1-\beta(1-q)}\right)^2 \frac{1}{N_{\text{cxt}}^2} \sum_{ij} \xi_i^{k1} \xi_j^{k1} \left\langle \xi_i^{\phi\nu} \xi_j^{\phi\nu} \right\rangle \tanh\left(\beta(m_1^k + c_i)\right) \tanh\left(\beta(m_1^k + c_j)\right) \tag{96}$$

Again, only the terms with $i = j$ will be non-zero (see description following Eq. 87). However, in this case the terms $\langle \xi_i^{\phi\nu} \xi_j^{\phi\nu} \rangle$ for $i = j$ do not equal 1, but equal $a = N_{\text{cxt}}/N$. This then leaves us with

$$\left\langle \left(n_\nu^\phi\right)^2 \right\rangle = \frac{a}{[1-\beta(1-q)]^2} \frac{1}{N_{\text{cxt}}} \sum_i \tanh^2\left(\beta(m_1^k + c_i)\right) \tag{97}$$

Lastly, we can replace the sum of $\tanh^2$ functions with the same q variable defined in Eq. 86, to get an expression for r_n:

$$r_n = \left\langle \left(n_\nu^\phi\right)^2 \right\rangle = \frac{qb}{[1-\beta(1-q)]^2}. \tag{98}$$

Derivation Part C: finding an expression for m_1^k. We assume the network configuration is close to pattern 1 of subnetwork k, and so m_1^k should be much larger than the other overlaps. We follow the steps from Part A up to Eq. 78, but now using m_1^k instead of an arbitrary m_ν^k to get:

$$m_1^k = \frac{1}{N_{\text{cxt}}} \sum_i \tanh\left(\beta\left(m_1^k + \sum_{\mu\neq 1} \xi_i^{k\mu} \xi_i^{k1} m_\mu^k + b \sum_{\sigma\neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_i^{k1} n_\mu^\sigma + \sum_{j\in T_k} \xi_i^{k1} \delta_{ij} \langle S_j \rangle\right)\right) \tag{99}$$

$$= \frac{1}{N_{\text{cxt}}} \sum_i \tanh\left(\beta(m_1^k + c_i)\right) \tag{100}$$

Finally, we use the same trick as in Eq. 82, and treat m_1^k as an average of a function $f(z)$ over the Gaussian random variable z:

$$m_1^k \approx \int \frac{dz}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}z^2\right) \tanh\left(\beta(m_1^k + \mu_* + \sigma_* z)\right), \tag{101}$$

again remaining agnostic as to the exact form of the mean and variance of the distribution, which will be addressed in the next section.

Derivation Part D: approximating the distributions of q and m_1^k. We now focus on determining an appropriate approximating distribution for q and m_1^k. To do this, we will take a more in-depth look at the expression for c_i in Eq. 80, which we repeat here for convenience:

$$c_i = \sum_{\mu\neq 1} \xi_i^{k\mu} \xi_i^{k1} m_\mu^k + b \sum_{\sigma\neq k} \sum_\mu \xi_i^{\sigma\mu} \xi_i^{k1} n_\mu^\sigma + \sum_{j\in T_k} \xi_i^{k1} \delta_{ij} \langle S_j \rangle \tag{102}$$

The first term of Eq. 102 is a sum over the product of three independent random variables $\xi_i^{k\mu}$, ξ_i^{k1} and m_μ^k. Based on Eq. 73, we know that m_μ^k should have zero mean and variance $\alpha_{\text{cxt}} r/p$. Due to the symmetry of m_ν^k around zero, the other two variables $\xi_i^{k\mu} \xi_i^{k1} = \pm 1$ will not have any effect. We thus have the sum of $p-1$ random variables, each with variance $\alpha_{\text{cxt}} r/p$, which leads to a single random variable with zero mean and variance $\alpha_{\text{cxt}} r$.

Now we move on to the second term of Eq. 102, which is again a sum over the product of three random variables. From Eq. 74, we know that n_μ^σ should be zero mean with variance $\alpha r_n/(s-1)p$. The term ξ_i^{k1} again does not have an effect due to the symmetry of n_μ^σ. However, the term $\xi_i^{\sigma\mu}$ will be different – it will be nonzero with probability a, and ± 1 otherwise. If we assume that $\xi_i^{\sigma\mu}$ and n_μ^σ are independent, then we get a random sum of random variables of the form $b \sum_{i=1}^K X_i$, with $K \sim \mathcal{B}((s-1)p, a)$ and $X_i \sim \mathcal{N}(0, \alpha r_n/(s-1)p)$. Using Wald's equation, we arrive at a Gaussian distribution with zero mean and variance $\alpha r_n a b^2$.

However, it turns out that the assumption that $\xi_i^{\sigma\mu}$ and n_μ^σ are independent is not true. They are slightly correlated, which shifts the mean to α (shown in Fig. S4A). Thus, as the total capacity increases, the mean-field approximation gets worse. In fact, we even face this issue with the first term, as m_μ^k and $\xi_i^{k\mu}$ are also correlated, which causes the mean to scale with α_{cxt}. However, since α_{cxt} never goes above 0.138, this fact can be safely ignored. For the second term, we can solve this by setting $\mu_* = \alpha$.

Adding a term to the mean turns out to complicate the solution quite a bit, so we propose an alternative method as well. This

is based on two tricks. First of all, we can calculate the variance assuming zero mean by computing $\mathbb{E}[X^2] = \mathrm{Var}[X] + \mathbb{E}[X]^2$. Second of all, since each neuron has equal probability of being 1 and -1, we can split up the contribution to the variance from the positive and negative components of the distribution, each having an effect half of the time. It turns out that we get the following expression for the variance:

$$\begin{aligned}\mathrm{Var} &= \tfrac{1}{2}\mathbb{E}\left[X_+^2\right] + \tfrac{1}{2}\mathbb{E}\left[X_-^2\right] && (103)\\ &= \tfrac{1}{2}\left(\alpha a + \alpha^2\right) b^2 r_n + \tfrac{1}{2}\alpha a^2 b^2 r_n && (104)\\ &= \tfrac{1}{2}\alpha b^2 r_n \left(a + \alpha + a^2\right) && (105)\end{aligned}$$

This was confirmed empirically by simulating several networks over different realizations of a, b and s (Fig. S4B).

Finally, for the last term of Eq. 102, it can be shown that this will inject Gaussian noise with variance $\Delta^2 q$, where $\Delta^2 = \alpha_{\mathrm{cxt}}(1-b)/b$[48]. To see this, we first note that all three terms in this sum are independent random variables with zero mean, which means that the sum should also have zero mean. The term ξ_i^{k1}, being ± 1, will also have no effect due to the symmetry of the sum. Now, we can estimate the variance of this term, which we will call Z:

$$\begin{aligned}\mathrm{Var}[Z] &= \mathbb{E}\left[\left(\sum_{j\in T_k} \delta_{ij}\langle S_j\rangle\right)^2\right] && (106)\\ &= \mathbb{E}\left[\sum_{j,l\in T_k} \delta_{ij}\delta_{il}\langle S_j\rangle\langle S_l\rangle\right] && (107)\end{aligned}$$

Note that for $j \neq l$, the expected value of this expression is zero because each item is idependent and has zero mean. We are thus only left with terms for $j = l$:

$$\begin{aligned}\mathrm{Var}[Z] &= \mathbb{E}\left[\sum_{j\in T_k} \delta_{ij}^2\langle S_j\rangle^2\right] && (108)\\ &= \mathbb{E}\left[\delta^2\right]\mathbb{E}\left[\sum_{j\in T_k}\langle S_j\rangle^2\right] && (109)\\ &= \frac{\alpha_{\mathrm{cxt}}(1-b)}{b}\mathbb{E}\left[\frac{1}{N_{\mathrm{cxt}}}\sum_{j\in T_k}\langle S_j\rangle^2\right] && (110)\\ &= \Delta^2 q. && (111)\end{aligned}$$

As a whole we thus have the following:

$$\begin{aligned}\mu_* &= 0 && (112)\\ \sigma_*^2 &= \alpha_{\mathrm{cxt}} r + \tfrac{1}{2}\alpha b^2 r_n\left(a + \alpha + a^2\right) + \Delta^2 q && (113)\end{aligned}$$

Derivation Part E: solving for α_{cxt}. We can now solve simultaneously for q, r, r_n, and m_1^k (from here on, we will refer to m_1^k as m). We list the four equations of interest [Eq. 84, Eq. 89, Eq. 98, and Eq. 101] here again for convenience:

$$\begin{aligned}q &= \int \frac{dz}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}z^2\right)\tanh^2\left(\beta(m + \sigma_* z)\right) && (114)\\ r &= \frac{q}{[1-\beta(1-q)]^2} && (115)\\ r_n &= \frac{qb}{[1-\beta(1-q)]^2} && (116)\\ m &= \int \frac{dz}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}z^2\right)\tanh\left(\beta(m + \sigma_* z)\right), && (117)\end{aligned}$$

where $\sigma_*^2 = \alpha_{\mathrm{cxt}} r + \frac{1}{2}\alpha b^2 r_n(a + \alpha + a^2) + \Delta^2 q$. These equations resemble the resulting equations of the standard Hopfield network very closely. The only difference is that the variance of z has changed in Eq. 114 and Eq. 115 and there is an extra equation for r_n in Eq. 116. We will take the same approach as for the standard Hopfield network, following Hertz *et al.*[27], and solve the equations

in the limit of $\beta \to \infty$. Given this limit, we can make use of the following two integral identities:

$$\int \frac{dz}{\sqrt{2\pi}} \exp\left(-z^2/2\right)\left(1 - \tanh^2\left(\beta(az+b)\right)\right) \approx \sqrt{\frac{2}{\pi}} \frac{1}{a\beta} \exp\left(-b^2/2a^2\right) \tag{118}$$

$$\int \frac{dz}{\sqrt{2\pi}} \exp\left(-z^2/2\right) \tanh\left(\beta(az+b)\right) \approx \operatorname{erf}\left(\frac{b}{\sqrt{2}a}\right) \tag{119}$$

This enables us to write the four equations as:

$$C \triangleq \beta(1-q) = \sqrt{\frac{2}{\pi\sigma_*^2}} \exp\left(-\frac{m^2}{2\sigma_*^2}\right) \tag{120}$$

$$r = \frac{q}{(1-C)^2} \tag{121}$$

$$r_a = \frac{qb}{(1-C)^2} \tag{122}$$

$$m = \operatorname{erf}\left(\frac{m}{\sqrt{2\sigma_*^2}}\right) \tag{123}$$

We see that $q \to 1$ as $\beta \to \infty$, and the expression $C \triangleq \beta(1-q)$ is undetermined for $\beta \to \infty$. We can, however, express its limit with respect to m and r. Given that $q \to 1$, we can rewrite equations Eq. 115 and Eq. 116 as:

$$r = \frac{1}{(1-C)^2} \tag{124}$$

$$r_n = \frac{b}{(1-C)^2} \tag{125}$$

which can be rearranged to get $C = 1 - \frac{1}{\sqrt{r}}$, and therefore $r_n = rb$. We can thus write the solution as two equations which can be simultaneously solved for m and r, given a particular capacity α_{cxt} as well as the other parameters of σ_*^2:

$$1 - \frac{1}{\sqrt{r}} = \sqrt{\frac{2}{\pi\sigma_*^2}} \exp\left(-\frac{m^2}{2\sigma_*^2}\right) \tag{126}$$

$$m = \operatorname{erf}\left(\frac{m}{\sqrt{2\sigma_*^2}}\right) \tag{127}$$

Supplementary figures of

Context-modular memory networks support high-capacity, flexible, and robust associative memories

William F. Podlaski, Everton J. Agnes, and Tim P. Vogels

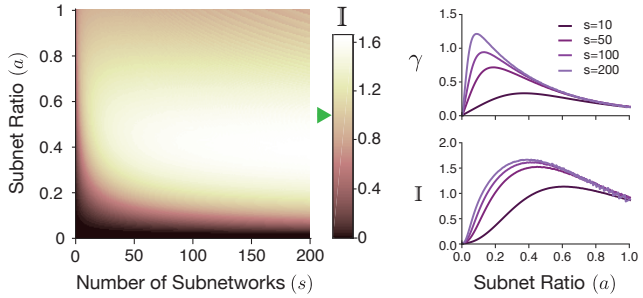

FIG. S1. Relative information content of context-modular memory network with neuron-specific gating as compared to the standard Hopfield network (green arrowhead; Methods). Vertical slices through the 2D plot are shown (lower right) and compared with full network memory capacity (upper right).

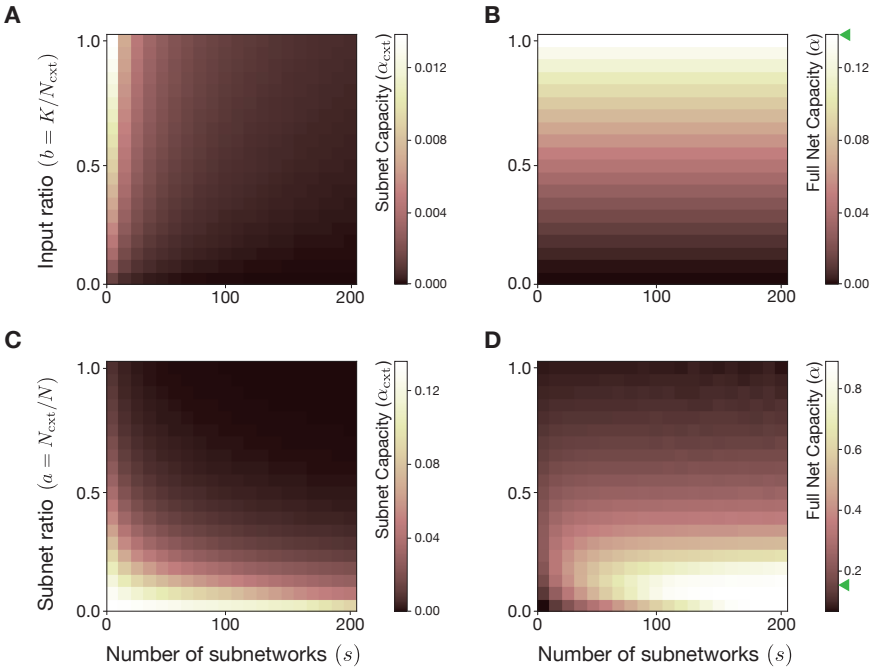

FIG. S2. **Memory capacity of the context-modular memory network with random synapse-specific gating and with both neuron-specific and synapse-specific gating. A,B**, Analytical capacity estimation for random synapse-specific gating for a single contextual configuration, α_{cxt} (**A**) and for total network, α (**B**), plotted as a function of the number of subnetworks, s, and the relative input ratio, b. **C,D**, Analytical capacity estimation for combined neuron-specific and synapse-specific gating (shown for fixed $b = 0.5$) for a single contextual configuration (α_{cxt}; **C**) and for total network (α; D), plotted as a function of the number of subnetworks, s, and the ratio of subnetwork size, a. Green arrowhead indicates the standard Hopfield network capacity, α_{H}.

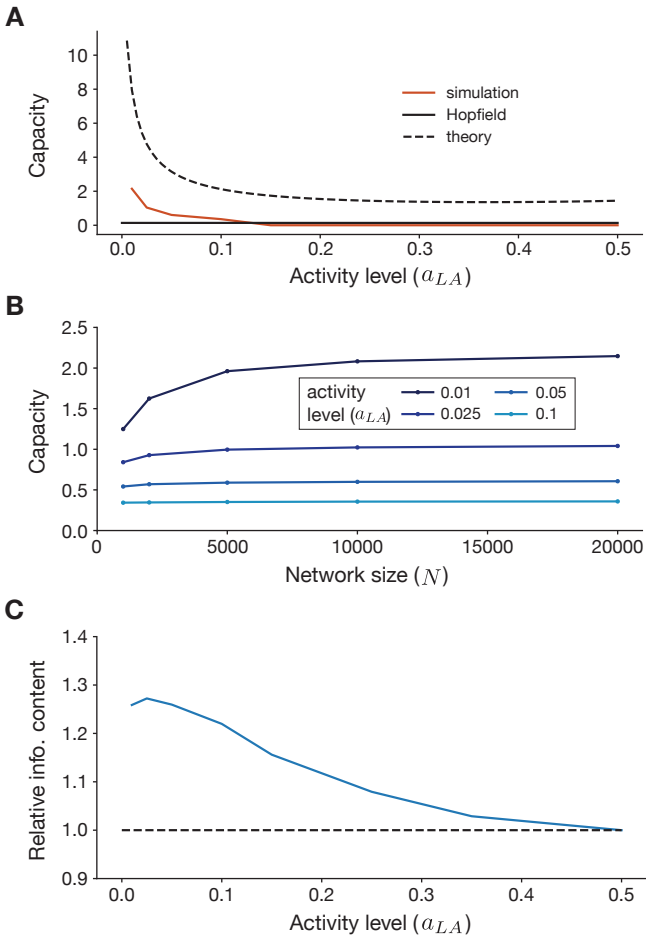

FIG. S3. Memory capacity of low-activity Hopfield network. **A**, Comparison of theoretical (dotted black) and numerical (red) estimations of memory capacity for the low-activity Hopfield network with $\{0,1\}$ units as in[34] as a function of activity level a_{LA}. Standard Hopfield capacity (0.138) is plotted as solid black line for reference. **B**, Numerical estimation of memory capacity as a function of network size, for different activity levels a_{LA}. Plot in **A** used $N = 20000$. **C**, Information content (relative to the standard Hopfield network) as determined from numerical simulations (red curve in **A**). See methods section *Analytical capacity estimation* for details.

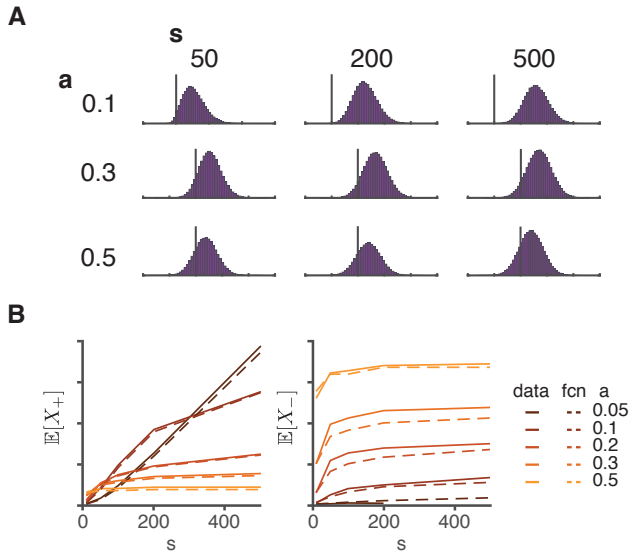

FIG. S4. Noise distribution and variance estimation for mean-field method. **A**, Distributions of crosstalk noise for "irrelevant" memories over different numbers of contexts (s) and relative subnetwork sizes (a). **B**, Due to non-zero mean, variance of the distribution was measured using the second moment of all positive values of the distribution ($\mathbb{E}[X_+]$) and the second moment of all negative values of the distribution ($\mathbb{E}[X_-]$). See Eq. 105.