Population sparseness determines strength of Hebbian plasticity for maximal memory lifetime in associative networks

Naomi Auer[1*], Lars Chen[1,2], Jakob Stubenrauch[2,3], Benjamin Lindner[2,3], Richard Kempter[1,2*]

1 Institute for Theoretical Biology, Department of Biology, Humboldt-Universität zu Berlin, Berlin, Germany
2 Bernstein Center for Computational Neuroscience Berlin, Berlin, Germany
3 Department of Physics, Humboldt-Universität zu Berlin, Berlin, Germany

* naomi.auer@hu-berlin.de (NA); r.kempter@biologie.hu-berlin.de (RK)

Abstract

The brain can efficiently learn and form memories based on limited exposure to stimuli. One key factor believed to support this ability is sparse coding, which can reduce overlap between representations and minimize interference. It is well known that increased sparseness can enhance memory capacity, yet its impact on the speed of learning remains poorly understood. Here we analyze the relationship between population sparseness and learning speed — specifically, how the learning speed that maximizes memory capacity depends on the sparseness of the neural code, and how this in turn affects the network's maximal capacity. To this end, we study a feedforward network with Hebbian and homeostatic plasticity and a two-state synapse model. The network learns to associate binary input-output pattern pairs, where sparseness corresponds to a small fraction of active neurons per pattern. The learning speed is modeled as the probability of synaptic changes during learning. The results presented in this manuscript are based on both network simulations and an analytical theory that predicts expected memory capacity and optimal learning speed. For both perfect and noisy retrieval cues, we find that the optimal learning speed indeed increases with increasing pattern sparseness — an effect that is more pronounced for input sparseness than for output sparseness. Interestingly, the optimal learning speed stays the same across different network sizes if the number of active units in an input pattern is kept constant. While the capacity obtained at optimal learning speed increases monotonically with output sparseness, its dependence on input sparseness is non-monotonic. Overall, we provide the first detailed investigation of the interactions between population sparseness, learning speed, and storage capacity. Our findings propose that differences in population sparseness across brain regions may underlie observed differences in how quickly those regions adapt and learn.

Author summary

The brain can efficiently learn and form memories based on limited exposure to stimuli. One key factor believed to support this ability is the way neural circuits encode and organize information about the external world. Population sparseness refers to the phenomenon in which, in many brain regions, only a small subset of neurons is active at

any given time or in response to a particular stimulus. Sparse codes are believed to reduce overlap between representations and minimize interference and can thus enhance the storage capacity of a network. Here we investigate the effect of population sparseness on the speed of learning input-output associations in a network model with Hebbian learning. We find that the learning speed that yields the maximal capacity increases with increasing sparseness. The maximal capacity increases for sparser input and output representations but in the first case only up to a certain point before sparseness becomes destructive. These findings could contribute to explaining why some brain regions learn faster than others.

Introduction

Across many regions of the brain, information seems to be represented by the activity of small fractions of the total number of neurons instead of single neurons or whole populations of neurons. In particular, the dentate gyrus (DG) is known to be dependent on very population sparse activity; estimates of population sparseness based on the expression of immediate early genes in the DG of rodents propose a percentage of active cells in the one-digit range [1–3]. Other hippocampal subfields, such as CA1 and CA3, also use sparse coding, but their activity patterns are believed to be less sparse than in DG [2, 4–6]. Outside the hippocampus, population sparseness has been investigated most extensively in the visual system, because it can explain various visual response properties [7, 8]. Estimates for the sparseness level cover a wide range of values, which may be due to different recording techniques, quantification measures, and behavioral paradigms. However, there is a consensus that only a fraction of the neurons available in principle in a particular visual area of the cortex is used to represent a visual stimulus [9–14]. Further, population sparseness has been quantified in the auditory system [15–17], the olfactory system [18–21], the somatosensory system [22], and the amygdala [23]. In general, representations appear to become sparser along the hierarchy of sensory processing [8, 16, 24–27].

In addition to the comprehensive experimental evidence, there are several reasons why sparse coding is favorable from a theoretical point of view. Famously, due to the reduced overlap between patterns, sparse representations can increase the memory capacity of associative neural networks. The dependence of the capacity on the fraction of active neurons was first shown by [28]. Later, [29] found that sparse patterns yield a higher capacity than dense patterns also in the Hopfield net [30, 31]. In 1987, [32] calculated the theoretical maximum capacity, and soon after [33] proposed a weight configuration that can achieve a capacity close to the theoretical optimum. The beneficial effect of sparseness on the storage capacity of different network structures has since been studied extensively and also confirmed for palimpsest-like networks [34–41].

Sparser representations are also favorable because they seem to enable faster learning, which is due to the reduced overlap between representations that reduces interference between patterns. In contrast to the well-established increased storage capacity, higher plasticity or faster learning as benefits of sparser coding are hardly discussed in the literature. For example, the hippocampus, which is well known for its highly sparse representations, is also known to be involved in rapid learning with high plasticity of synapses, serving episodic and spatial memory. A high turnover of dendritic spines as an indicator for high plasticity has been reported for the mouse hippocampus [42]. Studies using single-trial learning paradigms showed that the hippocampus can encode and store information after only a single exposure to an experience [43, 44]. This is often called one-shot learning. Research with rodents [45–47] and humans [48–51] demonstrates that concept cells can develop very quickly (within seconds to minutes), and that hippocampal neurons can rapidly form and reorganize

place fields or episodic representations during single learning episodes.

Compared to the hippocampus, the neocortex seems to have lower population sparseness. According to the theory of complementary learning systems in the context of systems memory consolidation, the neocortex is assumed to be a slow learning system, which needs repetitive exposure to acquire a new memory, for example, in the form of repeated memory replay by the hippocampus during slow-wave sleep [52–54]. This is in line with the observation that plasticity in the neocortex is lower than in the hippocampus [55, 56]. An artificial increase of plasticity in the neocortex has been reported to disrupt memory processes [57]. However, engrams can also form rapidly in the neocortex, either in parallel to engrams in the hippocampus or even without hippocampal support [43, 58, 59]. Fast memory formation in the neocortex was observed, for example, for memories related to prior knowledge in rodents [60, 61] and during the learning of new words in humans [62]. A recent analysis of a large fMRI dataset suggests that under specific conditions the human parahippocampal cortex (PHC), a part of the neocortex, can exhibit even higher plasticity than the hippocampus [63]. Even though various areas of the brain, in particular the hippocampus, are known for exhibiting sparse representations and fast learning, the association between these two features has not been properly analyzed so far.

In [64], it was briefly outlined that a smaller fraction of active cells allows for a higher learning rate and hence faster convergence in a gradient descent learning algorithm, but how this relates to the capacity of the network was not investigated. Furthermore, it is not immediately clear how this translates to biologically more plausible learning paradigms. [15] generated neuronal activity patterns based on sparse or dense firing rate distributions. A sequence of patterns alternating between a fixed target pattern and changing non-target patterns was used to train a single neuron with Hebbian synapses. The ratio of the target pattern response and the non-target pattern responses converges faster and to a larger value if the patterns were sampled from the sparse distribution. To the best of our knowledge, there is no deeper investigation of the dependence of the speed of learning on population sparseness.

In the present paper, we investigate the dependence of the optimal strength of plasticity (interpreted as the learning speed of the network) on the sparseness of representations in a scenario with Hebbian and homeostatic plasticity; furthermore, we explain the impact of learning speed and sparseness on the memory lifetime. To simplify the approach as much as possible, we use a heteroassociative feedforward network of neurons with binary activations, connected by two-state synapses; the network learns pairs of input-output patterns in a supervised fashion. Based on the underlying assumption that the memory capacity of the network should be maximal, we examine the role of fractions of active neurons and probabilities of synaptic changes in numerical simulations and analytical derivations.

This manuscript is organized as follows: We first introduce the model and the learning rule in more detail and explain how the capacity of the network is quantified. Then, we explain the relationship between the optimal learning speed and the sparseness of both the input and the output patterns. We also show how choosing the optimal learning speed for each sparseness value impacts the maximal memory capacity. The next section provides a more intuitive understanding of these results based on analytical considerations. Further, we discuss the effect of a change of the network size and analyze whether the fraction of active neurons or the absolute number of active neurons is the critical parameter. Finally, we add noise to the input pattern during retrieval in order to generalize our results to the biologically more realistic case of non-perfect retrieval cues.

Results

Network model, learning paradigm, and quantification of capacity

We are interested in the effect of population sparseness on the optimal learning speed of an associative network, and we hypothesize that sparse representations facilitate a large capacity while learning fast. This hypothesis is based on the following considerations: In order to establish an association between two neuronal assemblies, the synapses between the neurons that are part of one assembly and the neurons of the second assembly need to be strengthened. If f is the fraction of the total number of neurons that forms an assembly (activation ratio), the number of potential connections between neurons of the two assemblies scales with f^2. A low activation ratio is interpreted as high population sparseness, while a large activation ratio means dense representations and hence a low population sparseness (Fig 1A). Learning an association between two sparser assemblies involves only a smaller fraction of synapses. In this case, the memory capacity can be larger because each association affects fewer synapses. On the other hand, an association between denser assemblies involves the modification of a larger fraction of synapses, which can lead to catastrophic interference and lower capacity. For example, for sparser assemblies where only 1% of the cells is active ($f = 0.01$), only a fraction $f^2 = 1/10^4$ of the synapses is changed, while for denser assemblies with $f = 0.5$, a quarter ($f^2 = 0.25$) of all synapses potentially has to be updated.

We investigate this hypothesis in a hetero-associative one-layer feed-forward network with N_{in} input and N_{out} output units (1B), similar to the architecture of a Willshaw network [28]. A memory is a one-way association between an input pattern $\mathbf{x}$ and an output pattern $\mathbf{y}$. Units have two states and can be either active (1) or inactive (0). The input and output activation ratios f_{in} and f_{out} are defined as the fraction of active units in the input representation and the output representation, respectively. We denote the number of active input units in an input pattern $\mathbf{x}$ as $M_{\text{in}} = f_{\text{in}} N_{\text{in}}$; similarly, the number of active output units in a target output pattern $\mathbf{y}$ is $M_{\text{out}} = f_{\text{out}} N_{\text{out}}$. We typically assume $M_{\text{in}} \gg 1$ and $M_{\text{out}} \gg 1$. Input and output layers of the network are connected by a weight matrix J with binary entries. For the construction of J, we distinguish between morphological and functional connectivity. In biological neural systems, the number of morphological (or structural) synapses per unit is constrained. Therefore, we assume a morphological connectivity $0 < c_m \leq 1$. The morphological connectivity is normalized per output unit, thus each output unit is targeted by exactly $c_m N_{\text{in}}$ morphological connections chosen randomly among all possible connections from input units. Entries of the weight matrix J that do not correspond to a morphological connection are permanently set to zero. We have a total number of $c_m N_{\text{in}} N_{\text{out}}$ morphological connections between the input and the output units, and each connection can be functional (on, value 1) or silent (off, value 0). The weight matrix is initialized such that each output unit is targeted by exactly $c N_{\text{in}}$ connections that are functional (thin grey lines in Fig 1C top), and we call c the functional connectivity. Note that the functional connectivity is bounded by the morphological connectivity: $0 < c \leq c_m$.

The network learns a sequence of uncorrelated input and output pattern pairs $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})_{k \in \{0,1,\ldots\}}$, which are pairs of randomly drawn binary vectors that fulfill the sparseness constraint given by f_{in} or f_{out}. Since there is no explicit time component, presenting one pattern pair to the network can be interpreted as one step in time. For each pattern, the weights of the connection matrix J are updated in the following way: First, we consider a connection between an active input unit (filled grey circle in yellow layer in Fig 1C top) and an active output unit (filled grey circle in green layer in Fig 1C top). Such a connection can only change if there is a morphological connection. A connection that was already functional stays on, i.e., it is not changed (thin blue line originating from last input unit in Fig 1C top). A connection that was silent is turned

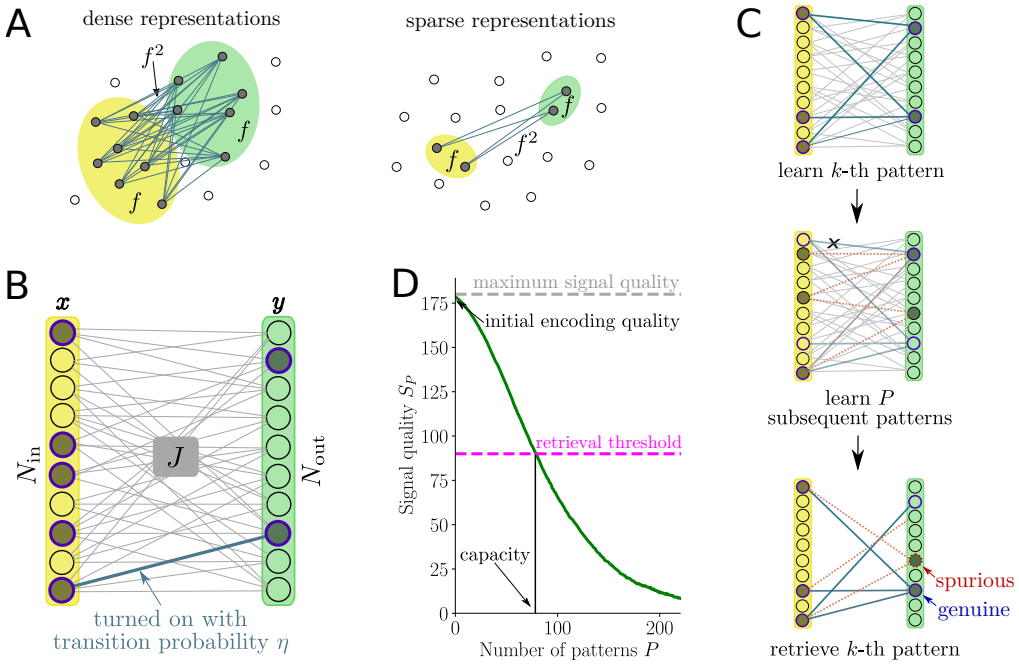

Fig 1. Sparse vs dense representations and quantification of capacity. (**A**) In a network of neurons with an activation ratio f, which is the number of neurons in an assembly divided by the total number of neurons in the network, the number of connections potentially involved in an association between two assemblies scales with f^2. The number of relevant connections is thus smaller for sparse representations (left) than for dense representations (right). (**B**) Illustration of the network model with an input layer (yellow) of size N_{in} (black circles) with input patterns $\mathbf{x}$ (indicated by gray discs and blue circles) of activation ratio f_{in} and an output layer (green) of size N_{out} with output patterns $\mathbf{y}$ (gray discs and blue circles) of activation ratio f_{out} that are connected by a binary weight matrix J with a morphological connectivity c_m (not shown) and a functional connectivity c (lines between input and output layer) (both normalized per output unit). During learning, the connections between active units are turned on with transition probability η; a homeostasis mechanism maintains the functional connectivity c. (**C**) Schematic illustration of the learning and retrieval processes. After learning the k-th pattern, P additional patterns are learned. In order to determine the memory signal quality of the k-th pattern at time P, input pattern $\mathbf{x}^{[k]}$ is presented to the network again to calculate an output pattern $\hat{\mathbf{y}}^{[k]}(P)$, which can then be compared to the target output pattern $\mathbf{y}^{[k]}$. (**D**) The quality of the memory signal (signal quality) is defined as $H_{\text{avg}} - H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P))$ (green line). It decays with the number of subsequently stored patterns P. The average Hamming distance H_{avg} between two random f_{out}-sparse vectors of length N_{out} (dashed grey line) is an upper bound of the average signal quality. The retrieval threshold is marked as T_S (dashed magenta line). The capacity of the network is defined as the number of patterns P for which the signal quality reaches the retrieval threshold T_S.

on with transition probability η. These connections are depicted as thick blue lines in Fig 1C top. We interpret the transition probability $0 \leq \eta \leq 1$ as the learning speed (or learning strength) of the network. For a high transition probability η, many connections are updated at once and the memory is strongly engraved into the network.

Furthermore, a homeostasis mechanism maintains the functional connectivity c. Previously functional connections are turned off in order to balance the additional connections turned on during a learning step. Otherwise all morphological connections would eventually become functional. In principle, there are many possible ways to normalize the number of functional connections and provide a homeostatic stabilization. Inspired by studies suggesting that neurons monitor the total excitation that they receive and the summed synaptic surface area per length of dendritic segment is conserved over time [65, 66], we apply a post-synaptic homeostasis step after each learning step. The functional connectivity c is maintained per output unit by randomly silencing the necessary number of connections from inactive input units to the active output units. To this end, we must assume $f_{\text{in}} \leq c$ (see Methodsfor details). Connections targeting inactive output units are not changed. This combination of Hebbian potentiation and homeostatic depression realizes a learning rule that is similar to the ones suggested by [35] and by [36].

To asses memory capacity, we quantify changes of the memory trace of the k-th pattern pair after learning P subsequent pattern pairs. With increasing P, the memory trace deteriorates more until the memory is not retrievable anymore according to some criterion. We always understand the time point P as relative to the k-th pattern. So after learning the k-th pattern, P additional patterns (illustrated for $P = 1$ in Fig 1C center; units active in the $k + 1$st pattern are shown as filled grey circles, units active in the k-th pattern are shown as circles with dark blue stroke) are learned by turning on connections that are relevant for these patterns and turning others off, potentially also connections that were relevant for the k-th pattern. Connections that are turned on in this step are shown as dashed orange lines and connections that are silenced are shown as light blue lines that are crossed out. The network hence realizes a palimpsest memory system [67] where older memories are erased by newer memories due to the homeostasis mechanism.

In order to quantify memory capacity, we distinguish two types of output units: Output units are either *genuine* with respect to the k-th pattern, hence active in the original target output $\mathbf{y}^{[k]}$ or *spurious* with respect to the k-th pattern, hence inactive in the target output $\mathbf{y}^{[k]}$. During retrieval, genuine output units might become inactive and spurious units might become active. At any time step P after learning the k-th pattern, the retrieval of the k-th input-output pattern pair can be tested by presenting the original input pattern and calculating the output with the current weight matrix $J^{[k+P]}$ (Fig 1C bottom). Some of the relevant connections of the k-th pattern are still functional (blue lines) and provide input to the genuine output units (blue output units in Fig 1C, bottom). Other connections from the same input units might be functional due to learning subsequent patterns (dashed orange lines) or due to the initial weights before learning the k-th pattern. These connections provide input to spurious output units (red output unit in Fig 1C bottom).

Output units are modeled as McCulloch-Pitts neurons [68], which sum the weighted inputs and compare them against an activation threshold $T_{\text{in}}^{[P]}$. Their output is calculated by

$$\hat{y}_j^{[k]}(P) = \Theta\left(\left(\sum_{i=1}^{N_{\text{in}}} J_{ji}^{[k+P]} x_i^{[k]}\right) - T_{\text{in}}^{[P]}\right) \tag{1}$$

where

$$\Theta(x) = \begin{cases} 1, & \text{for } x \geq 0 \\ 0, & \text{for } x < 0 \end{cases} \tag{2}$$

is the Heaviside step function. The activation threshold $T_{\text{in}}^{[P]}$ is adjusted in every time step such that the calculated output pattern has the same activation ratio f_{out} as the target output. This phenomenologically describes latent competition where $M_{\text{out}} = f_{\text{out}} N_{\text{out}}$ winners take all and the rest is deactivated. Note that this choice of threshold is motivated by the comparability of results based on fixed sparseness values. It is not necessarily an optimal activation threshold in that it maximizes capacity or other performance metrics. The difference between the output $\hat{\mathbf{y}}^{[k]}(P)$ obtained with the weight matrix $J^{[k+P]}$ at time (or pattern) P and the original target output $\mathbf{y}^{[k]}$ of this input pattern is measured by the absolute number of bit flips between $\hat{\mathbf{y}}^{[k]}(P)$ and $\mathbf{y}^{[k]}$, termed the Hamming distance

$$H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)) = \sum_{j=1}^{N_{\text{out}}} \left(y_j^{[k]} - \left(\hat{y}^{[k]}(P) \right)_j \right)^2 \tag{3}$$

$$= \# \left\{ j \in \{1, \ldots, N_{\text{out}}\} | y_j^{[k]} \neq \left(\hat{y}^{[k]}(P) \right)_j \right\}. \tag{4}$$

The larger the number of subsequent patterns P that have been learned since the one that is being tested for retrieval, the more the obtained output will differ from the target output because the summed inputs to the genuine units become too small compared to the summed inputs of the spurious units.

To quantify the strength of the memory after P time steps, we define the signal quality

$$S_P := H_{\text{avg}} - H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)) \tag{5}$$

where the first term $H_{\text{avg}} := 2N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}})$ is the average Hamming distance between two random f_{out}-sparse patterns (Fig 1D, dashed grey line) and the second term is the Hamming distance between the calculated output representation $\hat{\mathbf{y}}^{[k]}(P))$ and the target $\mathbf{y}^{[k]}$ (Fig 1D). For two random f_{out}-sparse patterns, an error occurs if either the bit of the first pattern is 1 (with probability f_{out}) and the bit of the second pattern is 0 (with probability $1 - f_{\text{out}}$) or the first bit is 0 (with probability $1 - f_{\text{out}}$) and the second one is 1 (with probability f_{out}); the probability for a deviation between the two entries is thus the sum over the probabilities of the two distinct events. Summed over all entries, this gives H_{avg}.

We define the memory capacity P^* of the network as the lifetime of a memory stored in the network, which is the time that passes until the signal of the memory has decayed to the point that the memory cannot be successfully retrieved anymore. Alternatively, it is the number of patterns that can be stored in addition to the first pattern, without destroying the retrieval of the first pattern. The capacity is determined by introducing a retrieval threshold T_S (Fig 1D, dashed magenta line), which represents the error tolerance between the obtained output and the target. The number of subsequently stored patterns for which the signal quality reaches the retrieval threshold can be interpreted as the lifetime of the memory or the memory capacity P^* of the network, defined by the equation

$$H_{\text{avg}} - H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P^*)) = T_S. \tag{6}$$

We define the threshold

$$T_S = t_S H_{\text{avg}}, \tag{7}$$

with $t_S \in (0,1)$, as a fraction of the maximal signal quality which is the Hamming distance between two random f_{out}-sparse vectors H_{avg}. A fixed number of wrongly activated units thus has a more severe destructive effect if the total number of active output units $M_{\text{out}} = N_{\text{out}} f_{\text{out}}$ is small. Whenever not further specified, we choose $t_S = 0.5$ and thus $T_S = N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}})$.

Theory on distributions of dendritic sums and calculation of capacity

The memory capacity of the network can be calculated numerically in simulations but it can also be described analytically in a probabilistic sense. Given a particular input pattern $\mathbf{x}^{[k]}$, the summed input that an output unit receives is called the dendritic sum of this output unit. Output units can potentially be either genuinely or spuriously active in the calculated output $\hat{\mathbf{y}}^{[k]}(P)$ depending on whether they are active or inactive in the target output $\mathbf{y}^{[k]}$. For each output unit, the dendritic sum hence follows either the distribution of dendritic sums of spurious output units (red distribution in Fig 2A, will be called *spurious distribution*) or the distribution of dendritic sums of genuine output units (blue distributions in Fig 2A, will be called *genuine distribution*).

The probability mass function (PMF) of the dendritic sum of a spurious unit is

$$p_s(x) = \mathcal{B}_{M_{\text{in}},\, c}(x) \tag{8}$$

(see Fig 2A red, derived in Methods, see Eqs. (34) and (42)), where $\mathcal{B}_{n,p}$ denotes a binomial distribution with n trials and success probability p. The spurious distribution is centered at $M_{\text{in}} c$ where $M_{\text{in}} = N_{\text{in}} f_{\text{in}}$. We note that $M_{\text{in}} c$ is constant in P due to the constant functional connectivity c that ensures a fixed number of functional connections per output unit; when a random input pattern is applied, on average, a fraction f_{in} of the cN_{in} connections receives input. The spurious distribution does not change when new uncorrelated patterns are learned, and thus the spurious distribution p_s does not depend on the number of patterns P.

In contrast, the PMF of the distribution of the dendritic sums of genuine units depends on P; it is given by

$$p_g^{[P]}(x) = \sum_{u=0}^{P} \left[\binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u} \cdot \mathcal{B}_{M_{\text{in}},\, \rho_g(u)}(x) \right] \tag{9}$$

(see Fig 2A blue) with the probability for a connection to a genuine unit to be functional given that the output unit is active u times across the whole pattern set

$$\rho_g(u) = (c_m - c)\eta \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right)^u + c \tag{10}$$

(see Fig 2B blue, derived in Methods, see Eqs. (34) and (57)). Immediately after learning a specific pattern, the probability of the relevant connections to a genuine output unit to be functional is larger than the average functional connectivity c, i.e., in Eq. (10) we have $\rho_g(0) = (c_m - c)\eta + c$. Due to the homeostatic plasticity, the probability ρ_g is reduced every time this output unit is active in subsequent patterns, which is reflected in the term $\left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right)^u$ where u is the output unit usage. For $u \to \infty$, we have $\rho_g(u) \to c$. The probability ρ_g enters Eq. (9) in $\mathcal{B}_{M_{\text{in}},\, \rho_g(u)}(x)$, which is the distribution of the dendritic sums if the output unit was active u times in the other patterns. The distribution of the output unit usage u, hence the number of times that an output unit is active across a set of P patterns, explains the weight $\binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u}$ of a binomial distribution in Eq. (9) (see Fig 2B). For $P \to \infty$,

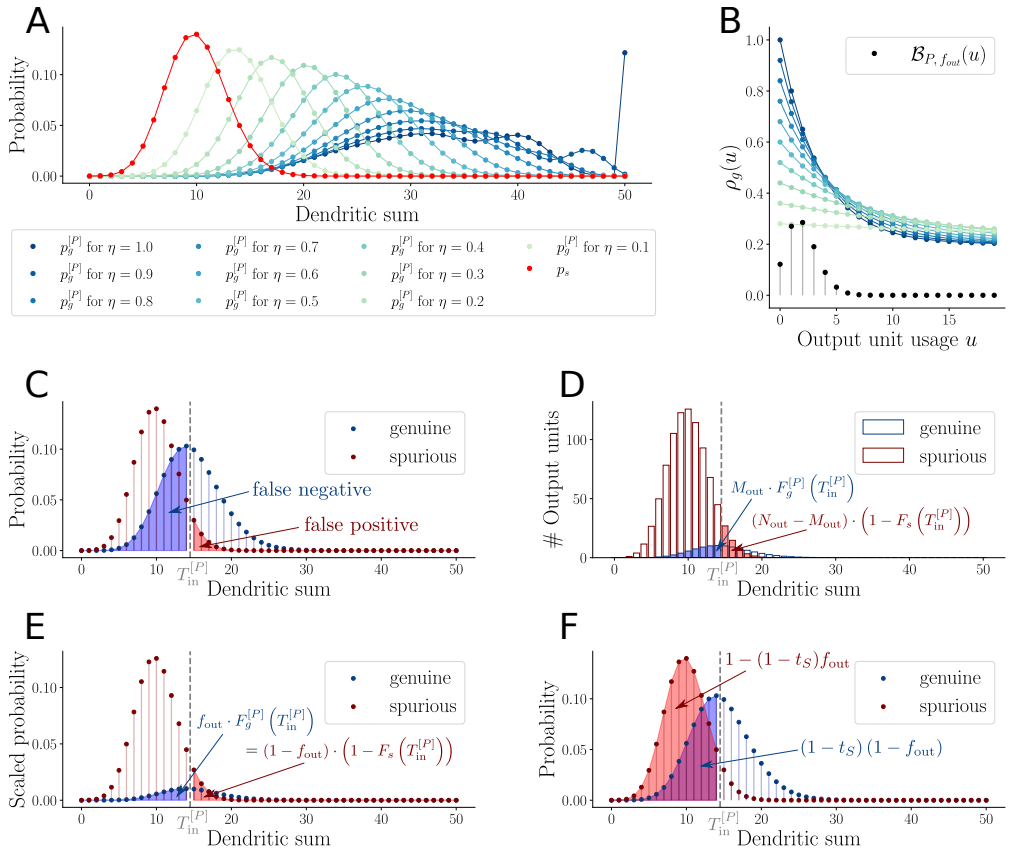

Fig 2. Distributions of dendritic sums and theory on capacity. (**A**) The smaller the transition probability η, the closer the PMF of the dendritic sums of genuine output units (blue) is to the PMF of the dendritic sums of spurious output units (red). (**B**) The probability of a functional connection to a genuine output unit decays with the output unit usage u. The probability $\mathcal{B}_{P,\,f_{\text{out}}}(u)$ that an output unit is active u times across P patterns is the weight of the corresponding binomial PMF in Eq. (9). Different shades of blue represent different transition probabilities η (same as in (**A**)). In (**A**) - (**B**), $P = 20$. (**C**) Spurious units with a dendritic sum larger than the activation threshold $T_{\text{in}}^{[P]}$ (red area) are wrongly activated (false positive); genuine units with a dendritic sum smaller than $T_{\text{in}}^{[P]}$ are wrongly not activated (false negative). (**D**) shows a histogram of the dendritic sums of output units. The expected Hamming distance is the sum of falsely deactivated genuine units (blue area) and falsely activated spurious units (red area). In (**E**), the PMFs of dendritic sums of genuine (blue) and spurious (red) units are scaled by f_{out} and $1 - f_{\text{out}}$, respectively. To respect the Balance Equation, the mass of the part of the genuine distribution left of the activation threshold $T_{\text{in}}^{[P]}$ (blue area) must correspond to the mass of the part of the spurious distribution right of $T_{\text{in}}^{[P]}$ (red area). (**F**) At capacity $P = P^*$, the activation threshold $T_{\text{in}}^{[P]}$ corresponds to the $1 - (1 - t_S) f_{\text{out}}$-quantile of the spurious distribution and to the $(1 - t_S)(1 - f_{\text{out}})$-quantile of the genuine distribution. In (**C**) - (**F**), $P = 108$. Further parameters: $N_{\text{in}} = N_{\text{out}} = 1000, f_{\text{in}} = 0.05, f_{\text{out}} = 0.1, c = 0.2, c_m = 1, t_S = 0.5$.

the genuine distribution converges to the spurious distribution. Distributions of dendritic sums obtained from numerical simulations match the distributions in Eqs. (8) and (9) very well (see example in S5 Fig).

For an analytical description of the expected Hamming distance $\mathbb{E}(H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)))$ between a target output pattern $\mathbf{y}^{[k]}$ and the corresponding output pattern $\hat{\mathbf{y}}^{[k]}(P)$ calculated during retrieval, we need to calculate the expected number of output neurons that are in the wrong state in $\hat{\mathbf{y}}^{[k]}(P)$ compared to $\mathbf{y}^{[k]}$. The activation threshold T_{in} (dashed grey line in Fig 2C) determines which output units are activated and which are not. If the distributions of dendritic sums of genuine and spurious output units are far apart and do not overlap, which can be the case during the first steps after learning a pattern, hence for small P, the activation threshold T_{in} can perfectly separate the two distributions, and then the calculated output is identical with the target output. If the two distributions are close to each other, a part of the spurious output units is wrongly activated (red area under red distribution in Fig 2C, false positive) and a part of the genuine output units is wrongly inactivated (blue area under blue distribution in Fig 2C, false negative). These two errors both contribute to the Hamming distance between the target output pattern and the calculated output pattern but they are weighted differently, as will be explained in the following (see Methods for a detailed derivation). In order to derive an actual error contribution from these distributions, we have to consider how many units have dendritic sums following either of the two distributions. As mentioned, there are M_{out} active units in the target output pattern. There are hence M_{out} output units whose dendritic sum follows the genuine distribution and $N_{\text{out}} - M_{\text{out}}$ output units whose dendritic sum follows the spurious distribution. The false negative error and the false positive error thus have to be weighted by M_{out} and $N_{\text{out}} - M_{\text{out}}$, respectively. It follows that the expected Hamming distance can be expressed as

$$\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) = M_{\text{out}} e_{\text{fn}} + (N_{\text{out}} - M_{\text{out}}) e_{\text{fp}}, \tag{11}$$

where e_{fn} and e_{fp} denote the error probabilities due to false negatives (blue area) and false positives (red area), respectively.

Let $F_g^{[P]}$ and F_s denote the cumulative distribution functions (CDF) of the distributions of dendritic sums of genuine units and spurious units, respectively. Then, we can express the expected Hamming distance as

$$\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) = M_{\text{out}} F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) + (N_{\text{out}} - M_{\text{out}})\left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right), \tag{12}$$

see Fig 2D (derived in Methods).

As discussed in the previous section, we always adjust the activation threshold $T_{\text{in}}^{[P]}$ such that the output activation ratio of the target output pattern $\mathbf{y}^{[k]}$ is maintained in the calculated output $\hat{\mathbf{y}}^{[k]}(P)$. This restriction is respected if the Balance Equation

$$f_{\text{out}} \cdot F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) = (1 - f_{\text{out}}) \cdot \left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right) \tag{13}$$

is fulfilled (see Fig 2E), i.e., if the contribution to the Hamming distance of the wrongly inactive genuine units $M_{\text{out}} e_{\text{fn}}$ equals the contribution of the wrongly active spurious units $(N_{\text{out}} - M_{\text{out}}) e_{\text{fp}}$ (see derivation in Methods).

With the Balance Equation, we can express $F_g^{[P]}\left(T_{\text{in}}^{[P]}\right)$ in terms of $F_s\left(T_{\text{in}}^{[P]}\right)$ and the expected Hamming distance simplifies to

$$\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) = 2N_{\text{out}}(1 - f_{\text{out}})\left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right). \tag{14}$$

For the capacity, we need to determine the number of patterns P for which the signal quality $S_P = H_{\text{avg}} - H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)$ equals the retrieval threshold $T_S = t_S H_{\text{avg}}$ (see Methods for details), i.e., we have to solve the equation

$$H_{\text{avg}} - \mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) = t_S H_{\text{avg}}, \tag{15}$$

which can be simplified to

$$f_{\text{out}}(1 - t_S) = 1 - F_s\left(T_{\text{in}}^{[P]}\right). \tag{16}$$

Using $F_s\left(T_{\text{in}}^{[P]}\right) = 1 - f_{\text{out}}(1 - t_S)$ in the Balance Equation yields

$$F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) = (1 - t_S)(1 - f_{\text{out}}). \tag{17}$$

We now have two Equations (16) and (17) in two variables P and $T_{\text{in}}^{[P]}$. Solving them gives us the number of patterns P (and the corresponding activation threshold $T_{\text{in}}^{[P]}$) for which the signal quality S_P equals the retrieval threshold. This P is the capacity (or memory lifetime) P^*.

Since the dendritic sums follow discrete distributions (see Eqs. (8) and (9)), their CDFs $F_g^{[P]}$ and F_s are not invertible. If we approximate the distributions by appropriate continuous distributions, Eqs. (17) and (16) can be reduced to a single equation

$$F_s^{-1}\left(1 - (1 - t_S) f_{\text{out}}\right) = {F_g^{[P]}}^{-1}\left((1 - t_S)(1 - f_{\text{out}})\right) \tag{18}$$

in t_S. The capacity P^* corresponds thus to the number of patterns P for which the $1 - (1 - t_S) f_{\text{out}}$-quantile of the spurious distribution equals the $(1 - t_S)(1 - f_{\text{out}})$-quantile of the genuine distribution (see Fig 2F). An analytical solution to Eq. (18) is derived in the Methods.

Effect of activation ratio on optimal learning speed and capacity

Let us now quantify how the memory capacity P^* depends on the activation ratio of the input pattern f_{in}, the activation ratio of the output pattern f_{out}, and on the transition probability η. Fig 3A shows the signal quality as a function of the number of patterns P for several transition probabilities. The signal quality always decays with time, i.e., with the number of subsequently stored patterns. The larger the transition probability, the faster the decay because the number of connections that is updated in each time step is larger and the old memories are thus overwritten more quickly. In the long run, the signal quality decays to zero. The initial signal quality (shown at $P = 0$ in Fig 3A) strongly depends on the transition probability because it determines how many connections are used for the initial memory storage. For a higher transition probability, the memory is more strongly engraved into the system and, initially, can be retrieved better. If the transition probability is too low, too few connections are updated to recall a memory even right after storing it (grey circles in Fig 3A). The transition probability that yields the highest capacity, which we call the optimal transition probability, hence depends on a trade-off between a high initial encoding and a slow decay of the memory signal (Fig 3A).

Fig 3B summarizes how the memory capacity P^* depends on the input activation ratio f_{in} and the transition probability η — for fixed f_{out}. The capacity values of Fig 3A are visible in the marked column ($f_{\text{in}} = 0.025$) of Fig 3B. In each column of Fig 3B, hence for a fixed f_{in}, the largest capacity value that thus corresponds to the optimal transition probability is marked by a black square. The corresponding

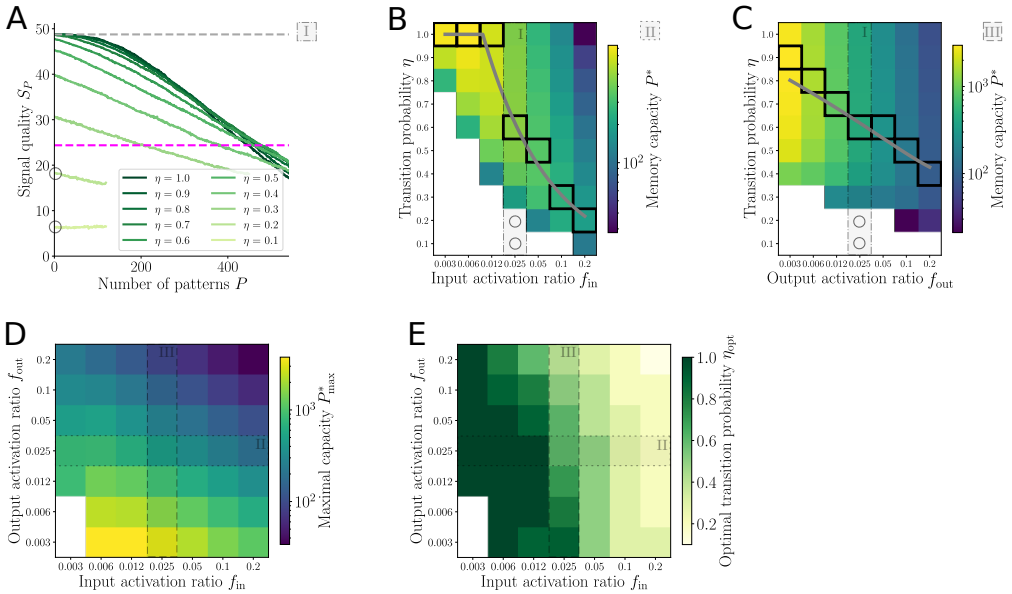

Fig 3. Effect of activation ratio on optimal learning speed and capacity. (**A**) Signal quality decaying with number of patterns P for various transition probabilities η. While small η values cause low initial signal quality, large η values cause a fast decay. A transition probability of $\eta = 0.6$ yields the largest memory capacity because it reaches the retrieval threshold (magenta dashed line), which is chosen as half of the maximal signal quality (grey dashed line), for the largest number of patterns P. For approximately $\eta < 0.25$, the initial signal quality is lower than the threshold and the capacity is zero (grey circles, see also grey circles in (**B**) and (**C**)). (Parameter values: $f_{\text{in}} = f_{\text{out}} = 0.025$.) (**B**) For a given transition probability, the memory capacity increases with decreasing input activation ratio f_{in} until it reaches a maximum and then quickly decays to zero for smaller input activation ratios. For a fixed input activation ratio f_{in}, the optimal transition probability, which is defined as the transition probability that results in the largest memory capacity for the given set of network parameters, increases with decreasing f_{in} until it reaches and stays at its largest value of 1 (see black squares). The solid grey line shows the analytically derived optimal transition probability as a function of the input activation ratio f_{in}. The signal quality traces of the column marked by I are shown in (**A**). White squares indicate a capacity of zero. (Output activation ratio $f_{\text{out}} = 0.025$.) (**C**) Same as (**B**) for output activation ratio f_{out} instead of input activation ratio f_{in}. The memory capacity and the optimal transition probability increase with decreasing output activation ratio f_{out}. Column I is shown in (**A**). (Input activation ratio $f_{\text{in}} = 0.025$.) In (**D**), the maximal capacity (obtained with the optimal transition probability η shown in (**E**)) for each combination of input activation ratio f_{in} and output activation ratio f_{out} is shown. For a fixed input activation ratio f_{in}, the maximal capacity increases monotonically with decreasing f_{out}. For a fixed output activation ratio f_{out}, the maximal capacity has a maximum at a small but nonzero f_{in}. Columns II and III are obtained from the results shown in (**B**) and (**C**), respectively. (**E**) Optimal transition probability increases with decreasing f_{in} and with decreasing f_{out}. The slope is larger for f_{in} than f_{out}. Columns II and III are shown in (**B**) and (**C**), respectively. Further parameter values in (**A**) -(**E**): $N_{\text{in}} = 1000, N_{\text{out}} = 1000, c = 0.2, c_m = 1, t_S = 0.5$, $N_{\text{avg}} = 200$ (see Methods for details on averaging).

analytical estimate of the optimal transition probability is displayed by the grey line (Eq. (165) in Methods) For a fixed transition probability η (single row in Fig 3B), the capacity P^* is non-monotonic as a function of f_{in}, i.e., it increases with increasing input activation ratio f_{in} and then decreases again. As known from the literature [33, 35, 37], denser representations (higher f_{in}) can lead to lower capacity values. However, if the input representations become too sparse (too small f_{in}), the maximal possible capacity also drops and goes to zero for very sparse input patterns. This can be explained by the fact that it is not possible for too few active input units to sufficiently excite the respective output units. In this case, a slight change of the weight matrix due to the storage of additional patterns can lead to very different units being activated in the output. In extreme cases — especially if both the transition probability η and the input activation ratio f_{in} are too small — the pattern is not well enough imprinted from the very beginning (cf. Fig 3A, grey circles, and 3B, white region) and the capacity is zero. This issue is particularly severe if the error tolerance $1 - t_S$ is small, hence if the retrieval threshold T_S is high, i.e., is close to its maximum H_{avg} (not shown here). In this case, the capacity can be zero even for high transition probabilities η if the initial signal quality lies below T_S.

For a fixed transition probability (single rows in Fig 3C), the capacity increases with decreasing output activation ratio f_{out}. In Fig 3C, the capacity is evaluated for several output activation ratios f_{out} and transition probabilities η while the input activation ratio f_{in} is fixed. The marked column in Fig 3C shows the same capacity values as the marked column in Fig 3B. A smaller output activation ratio f_{out} is beneficial for the capacity because a specific output unit is less likely to be active in subsequent patterns; hence relevant connections are turned off with lower probability. Nevertheless, if the transition probability η is too small, as before, the memory is not sufficiently well imprinted from the beginning and the capacity is zero (Fig 3C, white region).

In summary, we find that the sparseness of the input and the output patterns, respectively, have differential effects on the maximal capacity of the network. While the smallest output activation ratio achieves the highest capacity values (Fig 3C), there is a non-monotonic dependence of the maximal capacity on the input activation ratio (Fig 3B).

The non-monotonic dependence of the capacity on f_{in} and its monotonic dependence on f_{out} are also maintained for the maximal capacity. Fig 3D summarizes the maximal capacity for various pairs of input and output activation ratios f_{in} and f_{out}, which is obtained by using the optimal transition probability for each pair. As for the capacity for a fixed η in Fig 3B and C, Fig 3D shows an asymmetric effect of f_{in} and f_{out} on the maximal capacity.

Furthermore, we find that the optimal learning speed depends on the input as well as the output activation ratio. The sparser the patterns (small f_{in} or f_{out}), the larger the transition probability that allows for the largest capacity (depicted by black squares in Fig 3B and C). This effect is monotonic and has a larger gradient for the input activation ratio (Fig 3B) than the output activation ratio (Fig 3C), which is summarized in Fig 3E for various pairs of f_{in} and f_{out}.

In S4 Appendix, we compare our results with post-synaptic homeostasis to a pre-synaptic normalization and a normalization averaged over all connections and find that the qualitative dependences of the optimal transition probability and the maximal capacity on the activation ratios (compare Fig 5D,E to S4 Appendix) remain unchanged.

Activation ratios in the distributions of dendritic sums

The intuitive understanding of these qualitatively different effects of the input and the output activation ratios on the capacity can be strengthened by having a closer look at the distributions of the dendritic sums, which determine the signal quality of the

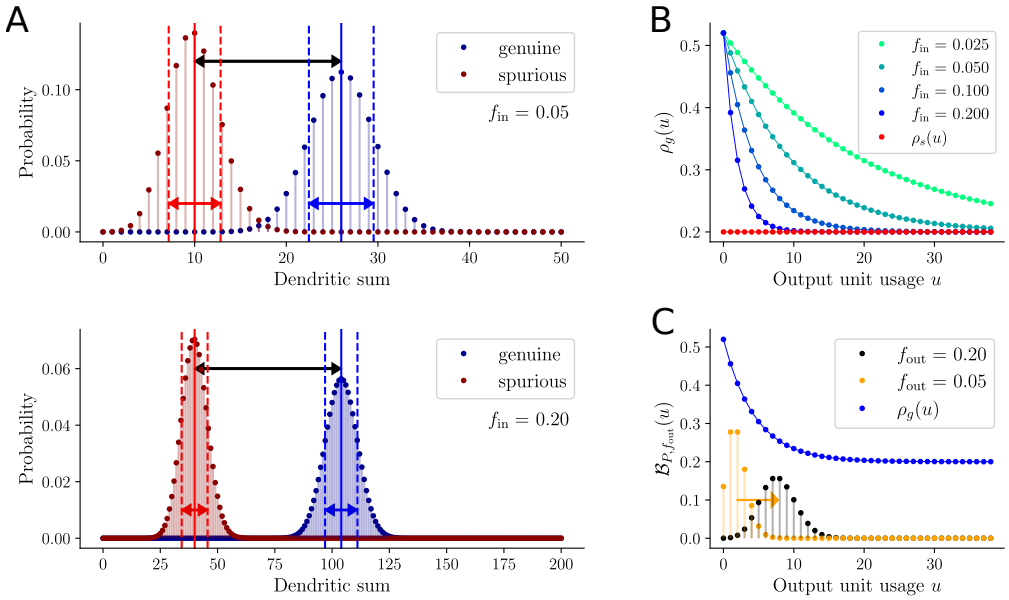

Fig 4. Role of the activation ratios in the distributions of dendritic sums. (**A**) Distributions of dendritic sums of genuine (blue) and spurious (red) output units. In the bottom panel, f_{in} is four times larger than in the top panel. For a larger input activation ratio f_{in}, the difference between the means of the distributions (black arrows) is larger. The standard deviations (red and blue arrows) are also larger for larger f_{in} (note the different scales of the x-axes in the two panels) but the increase is less than for the means. The overall overlap of the two distributions is smaller for larger f_{in}. (**B**) The probability ρ_g of a connection to a genuine output unit to be functional as a function of the output unit usage u decays faster and hence approaches ρ_s (red) faster for larger input activation ratio f_{in}. (**C**) The weight distribution $\mathcal{B}_{P,f_{\text{out}}}$ moves to the right if f_{out} is increased (black: f_{out} four times larger than in orange). The largest weights hence affect binomials $\mathcal{B}_{M_{\text{in}},\rho_g(u)}$ (in Eq. (9)) with smaller $\rho_g(u)$ values (blue). Further parameters: $N_{\text{in}} = 1000, f_{\text{out}} = 0.1, c = 0.2, c_m = 1, P = 0, \eta = 0.4$.

memory. We again distinguish distributions of dendritic sums of spurious and genuine units (red and blue, respectively, in Fig 2A) as given in Eq. (8) and Eq. (9), respectively. These explicit expressions for dendritic sums of spurious and genuine units allow us to understand why f_{in} and f_{out} affect the capacity of the network in different ways.

We first explain the role of f_{in}, while f_{out} is assumed to be fixed: A fixed f_{out} implies that the weighting of the individual binomial mass functions in Eq. (9) is fixed for a given P. We can hence discuss the role of f_{in} in the individual binomials without considering the weighting. The capacity's non-monotonic dependence on f_{in} is a combination of two separate effects working together. In the following, I discuss via what dominating mechanism f_{in} can positively correlate with the capacity, which is the case for very small f_{in} (left of the grey line in Fig 3B), and what mechanism has to dominate such that f_{in} is negatively correlated with the capacity, which is the case for larger f_{in} (right of the grey line in Fig 3B). First, the input activation ratio occurs in the number of trials $M_{\text{in}} = f_{\text{in}} N_{\text{in}}$ of the individual binomial distributions in $p_g^{[P]}$ and the number of trials M_{in} of the spurious distribution p_s. Let us assume for the moment that f_{in} was increased only in the number of trials of the binomials, and we ignore the dependence of ρ_g on f_{in} in Eq. (10). Such an increase of f_{in} in the number of trials pulls the distributions p_s and $p_g^{[P]}$ farther apart since both mean values are proportional to f_{in}. They are also stretched out more (higher standard deviation $\sqrt{M_{\text{in}} p(1-p)}$ with $p = c$ or $p = \rho_g(u)$) but the overall overlap of the two distributions is reduced because the means grow with M_{in} and the standard deviations grow only with $\sqrt{M_{\text{in}}}$ (also see left hand side of Eq. (133) for $P = 0$). This is illustrated in Fig 4A (top: smaller f_{in}, bottom: larger f_{in}). Due to the smaller overlap of the two distributions, the Hamming distance is reduced and this yields a larger capacity. For f_{in} increasing from zero, this increase of the capacity can be seen for the values left of the grey line in Fig 3B. However, the input activation ratio f_{in} also occurs in the probabilities $\rho_g(u)$ (Eq. (10)) of the individual binomials in the genuine distributions. If only changed at this place, an increased f_{in} leads to a faster decay of the probability with increasing u (Fig 4B) and hence with the number of patterns P. For larger f_{in}, each of the individual binomials $\mathcal{B}_{M_{\text{in}},\,\rho_g(u)}$ in Eq. (9) and hence also the total genuine distribution converges faster to the spurious distribution, which implies a deteriorative effect on the capacity. This can be seen for the capacity values right of the grey line in Fig 3B. In summary, these two effects of f_{in} interact with each other, which can explain the intermediate optimal value of f_{in} where the first effect is balanced by the second one to allow for an optimal capacity. For input activation ratios f_{in} close to zero, the effect of the change of f_{in} in the number of trials of the distributions (8) and (9) dominates over the effect of the change of f_{in} in (10). The capacity hence increases with increasing f_{in} if f_{in} is small (derived in S3 Appendix). This confirms the detrimental effect of a very small f_{in} on the network capacity that we discussed in the previous section: If f_{in} becomes very small, the number of active input units becomes too low to reliably activate the respective output units. For very large input activation ratios f_{in}, the effect of a change of f_{in} via the probability ρ_g (Eq. (10)) is stronger than the effect via the number of trials and the capacity decreases with increasing f_{in}.

The upper limit of f_{in} is c (see first section on condition $f_{\text{in}} \leq c$). The capacity for this upper limit of the input activation ratio f_{in} can be derived for the special case $c_m = 1, \eta = 1$. If f_{in} takes its maximal value c, the probability $\rho_g(u)$ (Eq. (10)) equals c and the genuine (Eq. (9)) and spurious distributions (Eq. (8)) become identical. This makes it impossible to differentiate between genuine and spurious units and the capacity is hence zero. In total, the intuition that we have developed here thus matches the numerical results discussed in the previous section.

Let us now turn to the output activation ratio f_{out}, which affects the capacity also in two ways. We now assume a fixed f_{in}. First, f_{out} is the success probability of the

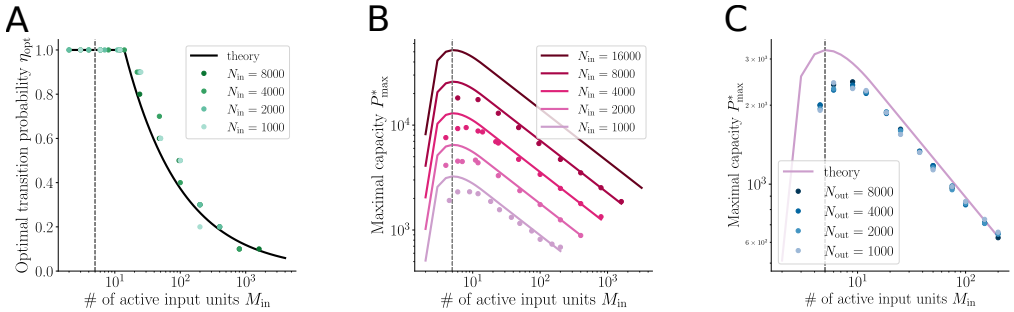

Fig 5. Role of network size and number of active units. Results from numerical simulations are depicted by dots, theoretical results are depicted by solid lines. (**A**) The optimal transition probability η_{opt} as a function of M_{in} monotonically decreases from a plateau at one to zero. It does not depend on the input layer size N_{in}. The overall highest capacity for a fixed network size is obtained for an M_{in} which yields an optimal η of one (grey dashed line). (**B**) shows the maximal capacity P^*_{max} as a function of the number of active input units $M_{\text{in}} = f_{\text{in}} N_{\text{in}}$ for $N_{\text{in}} = 1000, 2000, 4000, 8000$ and 16000. The size of the output layer is fixed to $N_{\text{out}} = 1000$ in the simulations. The maximal capacity as a function of M_{in} scales linearly with N_{in}. The grey dashed line marks the number of active input units M_{in} that yields the highest capacity. The highest capacity for a fixed network size is obtained at the same M_{in} for any network size. In (**C**), the output layer size N_{out} $(= 1000, 2000, 4000, 8000)$ is varied instead of the input layer size N_{in}. The number of input units N_{in} is fixed to 1000. The size of the output layer N_{out} does not have an impact on the maximal memory capacity P^*_{max} of the network. Further parameter values in (**A**)-(**C**): $f_{\text{out}} = 0.006, c = 0.2, c_m = 1, t_S = 0.5, N_{\text{avg}} = 200$.

binomial distribution, which constitutes the weights in the genuine distribution (Eq. (9)). An increased f_{out} pushes the center of mass of the distribution to larger values of u (Fig 4C). The largest weights thus are given to larger u values, which correspond to binomials with smaller probabilities $\rho_g(u)$. This emphasizes the parts of $p_g^{[P]}(x)$ that are more to the left of $x \in [0, M_{\text{in}}]$, which culminates in a shift to the left of the center of mass of the overall genuine distribution. In this sense, increasing f_{out} increases the overlap of the genuine distribution with the spurious distribution and hence decreases the capacity of the network. At the same time, f_{out} directly impacts the placement of the activation threshold because we are always enforcing $M_{\text{out}} = f_{\text{out}} N_{\text{out}}$ active output units. This is discussed in detail in the Methods. The effect on the capacity of a change of the activation threshold due to an increase of f_{out} is unknown (also see Methods). We speculate that this effect is small in the parameter range that we investigated numerically and that the effect of f_{out} via the weight distribution discussed above dominates the dependence of the capacity on f_{out}.

Role of network size and number of active units

So far, we have considered a fixed network size. We have seen how the capacity and the optimal transition probability depend on the input and the output activation ratio. Here, we discuss how these results depend on network sizes.

Assuming that the average number of functional connections per output unit is large but not too large ($M_{\text{in}} c \gg 1$ and $M_{\text{in}}(1 - \rho_g(\lfloor f_{\text{out}}(P+1) \rfloor)) \gg 1$) and $f_{\text{in}} \eta c_m / c \ll 1$, the capacity P^* of the network can be analytically approximated as

$$P^* = \frac{N_{\text{in}} c}{-\eta M_{\text{in}} c_m f_{\text{out}}} \cdot \ln \left(\frac{A(M_{\text{in}}, c, f_{\text{out}}, t_S)}{B(M_{\text{in}}, c_m, c, f_{\text{out}}, t_S) \cdot \eta} \right) \tag{19}$$

with A and B being two functions that are defined in Equations (147) and (148) in the Methods section (derived in Methods, see Eq. (159)). Note that both A and B are independent of the input and output network size N_{in} and N_{out}. They only depend on the number of active input units M_{in} and the output activation ratio f_{out}.

Under the assumption $\frac{f_{\text{in}}\eta c_m}{c} \ll 1$, the optimal transition probability can be approximated by

$$\eta_{\text{opt}} = \min\left(\frac{\mathrm{e}\cdot A(M_{\text{in}}, c, f_{\text{out}}, t_S)}{B(M_{\text{in}}, c_m, c, f_{\text{out}}, t_S)}, 1\right) \tag{20}$$

(black line in Fig 5A, derived in Methods, see Eq. (165)). It is independent of the input layer size N_{in} and the output layer size N_{out}. The maximal capacity P^*_{max} for a fixed set of network parameters is obtained by using the optimal transition probability η_{opt} in Equation (19):

$$P^*_{\text{max}} = P^*\big|_{\eta=\eta_{\text{opt}}}. \tag{21}$$

We find analytically and numerically that the size of the input layer strongly impacts the maximal capacity of the network (see Fig 5B): The capacity P^* scales linearly with the input layer size N_{in}. Since η_{opt} is independent of N_{in}, the maximal capacity P^*_{max} also scales linearly with N_{in} (Eq. (19) and Fig 5B). The maximal capacity P^*_{max} depends non-monotonically on the number of active input units M_{in}. We find that, for any input layer size, the highest capacity is reached for the same number of active input units M_{in} (Fig 5B, grey line). According to Eqs. (19) and (21)), the maximum over M_{in} of the maximal capacity P^*_{max} depends on the output activation ratio f_{out}, the functional and morphological connectivities c and c_m and of course, via t_S, also on the retrieval threshold $T_S = t_S H_{\text{avg}}$. Since, for a fixed network size N_{in}, the input activation ratio $f_{\text{in}} = M_{\text{in}}/N_{\text{in}}$ is simply a scaled version of M_{in}, the maximal capacity P^*_{max} also depends non-monotonically on f_{in}. We have seen examples of this non-monotonic dependence on f_{in} in Fig 3D. An increase of the retrieval threshold T_S yields a larger optimal transition probability and, of course, a smaller capacity (S1 Fig). If the morphological connectivity c_m is increased, η_{opt} decreases while P^*_{max} increases (S2 Fig). An increase of the functional connectivity c increases both η_{opt} and P^*_{max}, at least for large enough M_{in} (S3 Fig).

As a probability, the optimal transition probability is bounded between zero and one. It is a monotonically decreasing function of M_{in}. For small values of M_{in}, it takes the maximal value one. When M_{in} becomes large enough, the expression $\mathrm{e}A/B$ becomes smaller than one and η_{opt} becomes strictly decreasing as a function of M_{in} (Fig 5A). This monotonicity of η_{opt} as a function of M_{in} could not be formally shown with our theory but has been confirmed in all evaluations of Equation (20) and all numerical simulations that were performed. In the range where $\eta_{\text{opt}} < 1$, the maximal capacity P^*_{max} is monotonically decreasing as a function of M_{in} (shown in the Methods) (compare Fig 5A and B). It follows that, if the maximal capacity has a maximum as a function of M_{in} (for any given network size), it is reached within the range where we have $\eta_{\text{opt}} = 1$ (Fig 5A, the grey line marks the M_{in} that yields the largest capacity). Note that, depending on the network parameters and the retrieval criterion, it can occur that there is no $M_{\text{in}} \geq 1$ for which $\eta_{\text{opt}} = 1$. In this case, the dependence of the maximal capacity on M_{in} (as well as f_{in}) is monotonic and the largest capacity is obtained for $M_{\text{in}} = 1$.

As can be seen from Eqs. (19) and (21) and as discussed in previous sections, the capacity depends on the output activation ratio f_{out}. For adequately large N_{out}, the size of the output layer, on the other hand, does not affect the storage capacity of the network (Eq. (19) and Fig 5C, also see Discussion.

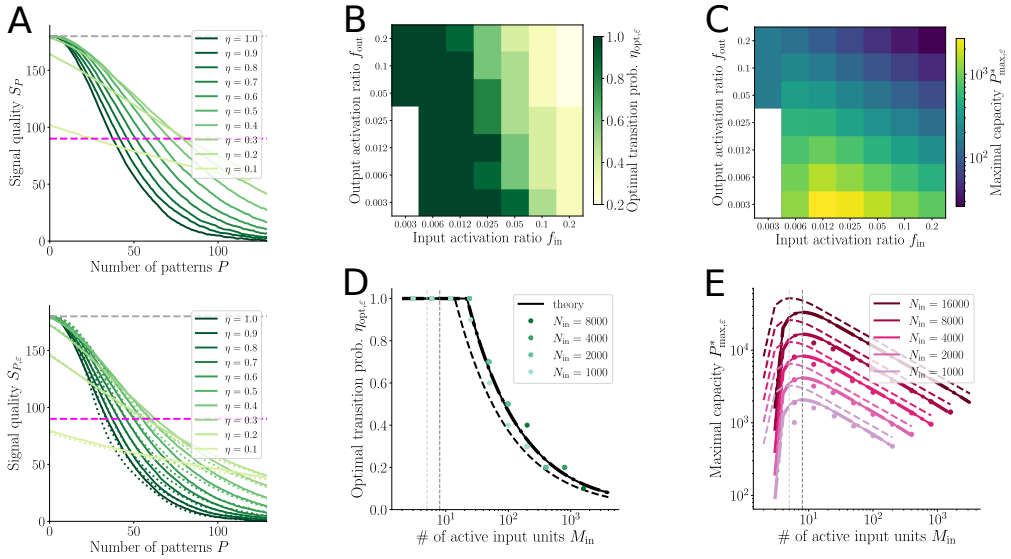

Fig 6. With noisy input patterns during retrieval, maximal capacity decreases and optimal transition probability increases. The noise level is set to $\varepsilon = 0.2$ in the entire figure. (**A**) The memory signal quality decays with the number of patterns learned by the network. For smaller transition probabilities η, the signal quality S_P starts lower but decays more slowly than for larger η. Top: No noise during retrieval. The largest capacity is attained for $\eta = 0.2$. Bottom: With noise on the input patterns during retrieval. With noise, the signal quality $S_{P,\varepsilon}$ is lower than without noise, particularly for small transition probabilities η. All capacity values with noise are slightly smaller than the corresponding ones without noise. The largest capacity is now attained for $\eta = 0.3$. In (**A**) the solid lines show results from simulations. The dotted lines are an approximation of the signal quality with noise $S_{P,\varepsilon}$ where the signal quality without noise S_P is multiplied by a factor obtained from theory (see S2 Appendix). In (**A**), $f_{\text{in}} = f_{\text{out}} = 0.1$. (**B**)-(**C**) The optimal transition probability $\eta_{\text{opt},\varepsilon}$ and the maximal memory capacity $P^*_{\text{max},\varepsilon}$ as functions of input and output activation ratios f_{in} and f_{out} obtained with noise during retrieval behave qualitatively similar to the maximal memory capacity P^*_{max} and the optimal transition probability η_{opt} obtained without noise (cf. Fig 3D and E). In (**A**)-(**C**), $N_{\text{in}} = N_{\text{out}} = 1000$. (**D**) The optimal transition probability $\eta_{\text{opt},\varepsilon}$ with noise (solid line, Eq. (22)) is slightly higher than without noise (dashed line, Eq. (20)). The dash-dotted line shows an approximation of $\eta_{\text{opt},\varepsilon}$ as a multiple of η_{opt} (Eq. (24)). The dark grey and light grey lines mark the M_{in} that yields the highest capacity with and without noise, respectively. With noise, the optimal M_{in} is larger than without noise. All black lines are derived from theoretical considerations, the colored dots show results from simulations (N_{out} fixed to 1000). (**E**) Noise on the input patterns during retrieval reduces the maximal capacity. The maximal capacity scales with the input layer size N_{in}. Dashed lines show the maximal capacity without noise (Eq. (21)), solid lines show the maximal capacity with noise (Eq. (25)), dash-dotted lines show an approximation of the maximal capacity with noise for small ε (Eq. (26)) and dots show numerical results (N_{out} fixed to 1000). Further parameter values in (**A**)-(**E**): $c = 0.2, c_m = 1, t_S = 0.5, N_{\text{avg}} = 200$.

Noisy input patterns during retrieval

We previously (Fig 3A) found that the transition probability that leads to the largest capacity depends on a balance between a large initial signal quality and a slow decay of the signal quality with the number of subsequently stored patterns. The top panel of Fig 6A also shows the signal quality as a function of the number of patterns but for a different set of input and output parameters ($f_{\text{in}} = f_{\text{out}} = 0.1$) than Fig 3A. Let us now investigate the effect of noisy input patterns during retrieval on the optimal transition probability and the maximal capacity of the network.

The input pattern that is presented to the network during the retrieval phase does not necessarily have to be the same pattern that was used for training. From a biological perspective, it makes sense to assume that the input during retrieval is not exactly the same but a noisy version of the original input pattern. The noisy input patterns are created by deactivating a particular fraction ε of the active input units while activating the same number of inactive input units. The input activation ratio f_{in} is hence maintained. Adding noise to the input pattern during retrieval reduces the memory capacity by deteriorating the signal quality right from the beginning (Fig 6A, top vs bottom). This shift in the initial signal quality is particularly strong for small transition probabilities η because the memory relies on few connections to provide a strong input to the output units. If some of the input units that are supposed to be active in a pattern are inactive and others are activated instead, it is likely that many of the genuine output units are not activated anymore. The effect is less drastic for large transition probabilities for which the memory is more reliably imprinted into many connections and does not get lost just because some input units are wrongly activated. However, we find that the change in the signal quality due to noise can neither be explained by a constant shift nor by a constant multiplicative scaling. The change depends on the number of patterns in a more complex way, see S2 Appendix. There we derived an analytical description of the factor that has to be applied to the signal quality S_P calculated without noise during retrieval to obtain an estimate of the signal quality $S_{P,\varepsilon}$ with noise.

Despite the complex relationship between S_P and $S_{P,\varepsilon}$, the qualitative dependence of the optimal transition probability $\eta_{\text{opt},\varepsilon}$ on the input and output activation ratios remains the same as without noise (Fig 6B, compare to Fig 3E) — it increases with decreasing activation ratios. Further, adding noise to the input pattern during retrieval maintains the non-monotonic effect of the input activation ratio and the monotonic effect of the output activation ratio on the maximal capacity (Fig 6C, compare to Fig 3D).

We extend our theory to retrieval with noisy input patterns and find that the optimal transition probability can again be described as

$$\eta_{\text{opt},\varepsilon} = \min\left(\frac{\mathrm{e}\cdot A_\varepsilon(M_{\text{in}}, c, f_{\text{out}}, t_S, \varepsilon)}{B_\varepsilon(M_{\text{in}}, c_m, c, f_{\text{out}}, t_S, \varepsilon)}, 1\right), \tag{22}$$

where A_ε and B_ε (Eqs. (182) and (183)) now additionally depend on the noise level ε (derived in Methods). For a fixed set of parameters, noise during retrieval increases the optimal transition probability:

$$\eta_{\text{opt},\varepsilon} \geq \eta_{\text{opt}} \tag{23}$$

(Fig 6D, compare solid line to dashed line). It can be approximated as a scaled version of the optimal transition probability without noise:

$$\eta_{\text{opt},\varepsilon} \approx \frac{\eta_{\text{opt}}}{1 - \frac{\varepsilon}{1-f_{\text{in}}}} \tag{24}$$

(Fig 6D, dash-dotted lines, also see S4 Fig. As discussed in previously, the optimal transition probability corresponds to a sweet spot that realizes both a good initial encoding of the memory and a slow decay of the memory trace. The additional noise impacts both aspects but since we find that $\eta_{\mathrm{opt},\varepsilon} \geq \eta_{\mathrm{opt}}$, the noise has a more detrimental effect on the initial signal quality than on the decay of the signal quality.

The maximal capacity can again be theoretically described as

$$P^*_{\mathrm{max},\varepsilon} = \frac{N_{\mathrm{in}} c}{-\eta_{\mathrm{opt},\varepsilon} M_{\mathrm{in}} c_m f_{\mathrm{out}}} \cdot \ln\left(\frac{A_\varepsilon(M_{\mathrm{in}}, c, f_{\mathrm{out}}, t_S, \varepsilon)}{B_\varepsilon(M_{\mathrm{in}}, c_m, c, f_{\mathrm{out}}, t_S, \varepsilon) \cdot \eta_{\mathrm{opt},\varepsilon}}\right), \tag{25}$$

(Fig 6E, solid lines, derived in Methods, see also S4 Fig). It can further be approximated by

$$P^*_{\mathrm{max},\varepsilon} \approx P^*\big|_{\eta=\eta_{\mathrm{opt},\varepsilon}} + \frac{\ln\left(1 - \frac{\varepsilon}{1-f_{\mathrm{in}}}\right) c}{f_{\mathrm{in}} f_{\mathrm{out}} \eta_{\mathrm{opt},\varepsilon} c_m} \approx P^*\big|_{\eta=\eta_{\mathrm{opt},\varepsilon}} - \frac{\varepsilon c}{f_{\mathrm{in}} f_{\mathrm{out}} \eta_{\mathrm{opt},\varepsilon} c_m}, \tag{26}$$

where $P^*\big|_{\eta=\eta_{\mathrm{opt},\varepsilon}}$ is the capacity without noise (Eq. (19)) evaluated for $\eta = \eta_{\mathrm{opt},\varepsilon}$ and the last approximation step assumes $\varepsilon \ll 1$ and $f_{\mathrm{in}} \ll 1$ (Fig 6E, dash-dotted lines, see also S4 Fig; derived in Methods). Hence, the maximal capacity with noise during retrieval is the maximal capacity without noise (evaluated at $\eta_{\mathrm{opt},\varepsilon}$) shifted down by a fixed amount that scales linearly with ε. The M_{in} that yields the highest capacity is again independent of N_{in} but it increases with the noise level ε (Fig 6E, vertical dashed lines, see also S4 Fig).

We can conclude that in a system with noisy input patterns, the capacity is lower than in a system without noise. However, with noise, it is beneficial for the capacity to learn with a higher transition probability.

Methods

In the following, we give a detailed account of the analytical derivations and numerical implementations used to obtain the presented results. First, we derive the distributions of dendritic sums, which provides the basis for the subsequent derivation of analytical expressions for the memory capacity. Finally, we describe the numerical implementation of the algorithm. Table 2 provides a summary of the main parameters.

First, we briefly recap the network model and the learning paradigm. The network is presented with a sequence of input/output-pattern pairs $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})$, $k \in \{0, 1, \ldots\}$. Input and output patterns are binary vectors of length N_{in} and N_{out}, respectively, and of activation ratio f_{in} and f_{out}, respectively. Hence, they consist of $M_{\mathrm{in}} = f_{\mathrm{in}} N_{\mathrm{in}}$ or $M_{\mathrm{out}} = f_{\mathrm{out}} N_{\mathrm{out}}$ active units and $N_{\mathrm{in}} - M_{\mathrm{in}}$ or $N_{\mathrm{out}} - M_{\mathrm{out}}$ inactive units, respectively.

The input and the output layer of the network are connected by a weight matrix J. Per output unit, a number $c_m N_{\mathrm{in}}$ connections ($c_m N_{\mathrm{in}}$ entries per row of the weight matrix) are randomly chosen to be morphologically available connections. They can be functional ('on', value 1) or silent ('off', value 0). The functional connectivity is always normalized such that a fixed number $c N_{\mathrm{in}}$ of morphologically available connections per output unit is functional.

In each learning step, the weight matrix is updated according to the presented pattern pair $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})$. Silent connections between active input and active output units are turned on with transition probability η. In addition, we randomly silence the same number of functional connections from inactive input units to active output units such that the total number of functional connections per output unit is maintained. This update step transforms $J^{[k-1]}$ into $J^{[k]}$.

For any pattern pair that has already been learned, retrieval can be tested by presenting the original input $\mathbf{x}^{[k]}$ and calculating the output as

$$\hat{y}_j^{[k]}(P) = \Theta\left(\left(\sum_{i=1}^{N_{\text{in}}} J_{ji}^{[k+P]} x_i^{[k]}\right) - T_{\text{in}}^{[P]}\right) \tag{27}$$

for $j = 1, \ldots, N_{\text{out}}$, where P is the number of subsequent patterns that have been learned since pattern k, and the activation threshold $T_{\text{in}}^{[P]}$ is chosen such that M_{out} output units are activated. To determine the decay of the memory signal, this calculated output $\hat{\mathbf{y}}^{[k]}(P)$ can be compared to the given output pattern $\mathbf{y}^{[k]}$. The larger P, the larger the Hamming distance between the two vectors:

$$H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)) = \sum_{j=1}^{N_{\text{out}}} \left(y_j^{[k]} - \hat{y}_j^{[k]}(P)\right)^2. \tag{28}$$

We define the signal quality after P subsequent patterns as

$$S_P := H_{\text{avg}} - H(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)), \tag{29}$$

where $H_{\text{avg}} = 2N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}})$ is the average Hamming distance between two random f_{out}-sparse vectors of length N_{out}.

The retrieval threshold is defined as

$$T_S := t_S H_{\text{avg}}, \tag{30}$$

for a fixed $t_S \in (0, 1)$. We call a pattern with $S_P \geq T_S$ retrievable and a pattern with $S_P < T_S$ not retrievable. The maximal number P for which $S_P \geq T_S$, if S_P is averaged across many patterns, is defined as the capacity P^* of the network.

Probabilistic description of dendritic sums

To better understand the learning dynamics of the network and the decay of the quality of the memory signal, which is used to quantify the capacity, we derive a probabilistic description of the dendritic sums of output units [69]. The dendritic sum of the j-th output unit is its net input

$$d := \sum_{i=1}^{N_{\text{in}}} J_{ji} x_i. \tag{31}$$

In accordance with [69], we introduce the following nomenclature: In general, a calculated output pattern $\hat{y}_j^{[k]}(P)$ differs from the target output pattern $y_j^{[k]}$. We call output units that are active (1) in the target output *genuine* output units and output units that are inactive (0) in the target output *spurious* units. In the calculated output, on the one hand, genuine units can be correctly active or incorrectly inactive. On the other hand, spurious units can be correctly inactive or incorrectly active. The terms *genuine* and *spurious* do not indicate whether the calculated activity of an output unit is 1 or 0, respectively, but whether it should be 1 or 0, respectively, to match the target output. In this Section, we derive the distributions of the dendritic sums of genuine and spurious output units separately, and these dendritic sums are denoted by symbols d_s (spurious) and d_g (genuine). The shapes of these two distributions (especially their overlap) is related to the Hamming distance between target and calculated output and consequently also to the signal quality. These relationships will become clear later in the Methods.

With respect to a particular pattern pair, we further distinguish between four types of connections:

- genuine-genuine (g-g) connections connect active input to active output units,
- genuine-spurious (g-s) connections connect active input to inactive output units,
- spurious-genuine (s-g) connections connect inactive input to active output units, and
- spurious-spurious (s-s) connections connect inactive input to inactive output units.

Note that the neuronal activity that determines the connection type is here understood as the activity in the original pattern pair $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})$ and not in the calculated output pattern $\hat{\mathbf{y}}^{[k]}(P)$.

Distributions of dendritic sums

In what follows, we calculate the distribution of the net input $d_l^{[P]} = J_{j-}^{[k+P]}\mathbf{x}^{[k]}$ to an output unit j where $l \in \{s, g\}$ denotes a spurious (s) or a genuine (g) output unit. This input $d_l^{[P]}$ is also called the dendritic sum of output unit j. The term $J_{j-}^{[k+P]}$ is the j-th line of the weight matrix obtained for P additional patterns after learning pattern k. To understand how the P subsequent patterns change the weight matrix, we have to take into account how often the output unit j is active across all P patterns, which is also called the output unit usage. As suggested by [69], we assume that the number r of times an output unit is active in P statistically independent patterns follows a binomial distribution $\mathcal{B}(P, f_{\text{out}})$, thus

$$\mathcal{P}(r = u) = \mathcal{B}_{P, f_{\text{out}}}(u) = \binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u}. \tag{32}$$

This describes the output unit usage more accurately than the assumption that each output unit is active exactly $f_{\text{out}} \cdot P$ times across a pattern set of size P, as assumed in the original paper on the Willshaw network [28].

The dendritic sum of the output unit j gets contributions only from connections that originate at one of the M_{in} input units that are active in the k-th pattern. Independently of their state (functional or silent), connections from inactive input units do not contribute to the net input to an output unit. We define $\rho_s(u)$ as the probability that a genuine-spurious connection connecting to output unit j is functional given that the output unit j is active u times in the pattern set. Similarly, the probability that a genuine-genuine connection is functional is called $\rho_g(u)$. The functionality of a connection implicitly takes into account that the connection is morphologically available. Recall that $\rho_l(u)$, for $l \in \{s, g\}$, depends on whether the respective output unit is genuine ($\rho_g(u)$), i.e., active in the target output, or spurious ($\rho_s(u)$), i.e., inactive in the target output. The derivation of $\rho_s(u)$ and $\rho_g(u)$ will be discussed in the following sections.Moreover, the probability that a spurious-spurious or a spurious-genuine connection is functional is less interesting because these connections do not contribute to the dendritic sums due to lack of input. However, these probabilities need to be considered implicitly in the following sections. to be able to determine $\rho_s(u)$ and $\rho_g(u)$. Further, the probability of a functional spurious-genuine connection, called $\rho_n(u)$, is important if the input pattern used for retrieval is noisy. This case will be discussed later.

For a particular output unit usage u and statistically independent output units, the dendritic sums $d_l^{[P]}$ are binomially distributed with

$$\mathcal{P}(d_l^{[P]} = x \mid r = u) = \mathcal{B}_{M_{\text{in}},\, \rho_l(u)}(x) = \binom{M_{\text{in}}}{x} \rho_l(u)^x (1 - \rho_l(u))^{M_{\text{in}} - x}, \tag{33}$$

for $l \in \{s, g\}$. In the extreme case where all connections from input units that are active in the k-th pattern to an output unit are functional, its dendritic sum equals the number of active input units M_{in}. Combining Eq. (32) and Eq. (33), we obtain

$$\mathcal{P}(d_l^{[P]} = x) = \sum_{u=0}^{P} \binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u} \binom{M_{\text{in}}}{x} \rho_l(u)^x (1 - \rho_l(u))^{M_{\text{in}}-x}, \tag{34}$$

for $l \in \{s, g\}$, as the distribution of the dendritic sums $d_l^{[P]}$. We abbreviate

$$p_s^{[P]}(x) := \mathcal{P}(d_s^{[P]} = x) \tag{35}$$

and

$$p_g^{[P]}(x) := \mathcal{P}(d_g^{[P]} = x). \tag{36}$$

Before calculating the probability $\rho_l(u)$ of a single genuine-genuine or a single genuine-spurious connection being functional after storing a set of additional patterns in which the output unit was active u times, we review the learning rule from a probabilistic perspective and determine important fractions of connections.

Hebbian and homeostatic update rules from a probabilistic perspective

We assume a morphological connectivity of $c_m \cdot N_{\text{in}}$ connections per output unit and start from a random choice of $c \cdot N_{\text{in}}$ functional connections targeting each output unit. In each learning step k, only connections targeting output units that are active in the current pattern, i.e., genuine-genuine and spurious-genuine connections, are updated. We further can distinguish between four types of connections: They can be either silent or functional before learning the k-th pattern and they can either originate from an active or an inactive input unit in the k-th pattern. Two types of connections are not updated: If the connection is already functional and the corresponding input unit is active (in addition to the corresponding output unit being active), the connection is protected and remains functional. If the connection is previously silent and the input unit is inactive, the connection always remains silent. The other two types can transition between the two states (functional/silent): Genuine-genuine connections (a fraction f_{in} of all connections to one output unit) of the k-th pattern that are previously silent (a fraction $c_m - c$) are made functional with a transition probability η, i.e., the fraction $f_{\text{in}}(c_m - c)\eta$ of all connections becomes functional in one learning step.

For each genuine output unit in the k-th pattern, the same number of connections that were made functional is also silenced such that there is a constant fraction c of functional connections per output unit after Hebbian learning and homeostatic normalization. Only spurious-genuine connections of the k-th pattern (fraction $1 - f_{\text{in}}$) that are previously functional (fraction c) could be silenced due to the homeostasis mechanism. This is a total fraction $c(1 - f_{\text{in}})$ of all connections. The probability that such a connection is silenced depends on the number of additional connections that became functional due to the Hebbian mechanism in the same learning step. As discussed above, a fraction $\eta \cdot (c_m - c) f_{\text{in}}$ becomes functional in each step. The probability for a previously functional connection from an inactive input unit to be silenced due to homeostasis is

$$\frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}. \tag{37}$$

Normalization is possible in this way if this probability is less than or equal to one, which is always fulfilled if we assume that $c \geq f_{\text{in}}$ (regardless of the values of η and c_m).

In summary, the numerator in Eq. 37 is the fraction of connections that are made functional in this step and the denominator is the fraction of connections available to be silenced; together, this is the *probability* to be newly silenced. Since the *fraction* of connections made functional, $\eta f_{\text{in}}(c_m - c)$, equals the *fraction* of connections newly silenced, $\frac{\eta f_{\text{in}}(c_m - c)}{c(1-f_{\text{in}})} \cdot c(1 - f_{\text{in}})$, the overall fraction c of functional connections is maintained. Since patterns are uncorrelated, these probabilities remain the same for every pattern that is learned, and they will be used several times in the derivations in the following subsections.

Probability of a functional connection to a spurious unit

First, we consider an output unit that is spurious for the k-th pattern. As we show below, the probability $\rho_s(u)$ for genuine-spurious connections to be functional given that the respective output unit was active u times across the patterns $1, \ldots, P$ does not change with u. Due to the initialization of turning on cN_{in} connections per output unit (blue part in Fig 7A) and the fact that storing the k-th pattern does not affect the connections to spurious units, we have that $\rho_s(0) = c$ (orange part in Fig 7B). Only connections to genuine units in each pattern are turned on or off. Hence, we do not need to distinguish between genuine-spurious and spurious-spurious connections in the k-th pattern (as we will have to in the next section where we discuss the probability of a functional connection to a genuine unit). While storing additional patterns, connections targeting output units that are spurious with respect to the k-th pattern will be updated because they might not be spurious with respect to the other patterns. Since the k-th input pattern is uncorrelated with the other input patterns and since the homeostasis mechanism maintains the functional connectivity level c at all times, the probability that a connection to a spurious output unit (with respect to the k-th pattern) is on remains c.

Let us explain this result in more detail: We assume that the spurious output unit that we are investigating is active for one additional pattern $k + \kappa$ with $\kappa \in \mathbb{N}^+$ fixed. The probability $\rho_s(1)$ of a connection to this unit to be functional after the update step can be decomposed using the law of total probability

$$\mathcal{P}(A) = \sum_{i \in I} \mathcal{P}(A|B_i)\mathcal{P}(B_i), \tag{38}$$

where $\{B_i\}_{i \in I}$ is a partition of the sample space. As conditions B_i, we distinguish between the four possible combinations that arise from two initial states of the connection and two activity states of the input unit: either the connection is already functional before learning pattern $k + \kappa$ (with probability c, orange part in Fig 7B) or it is previously silent (but morphologically available) (with probability $c_m - c$, grey part in Fig 7B); furthermore, the originating input unit is either active (with probability f_{in}) or inactive (with probability $1 - f_{\text{in}}$) in pattern $k + \kappa$. These four conditions have the following probabilities:

(i) $\mathcal{P}($*'connection previously functional and input unit active'*$) = cf_{\text{in}}$

(ii) $\mathcal{P}($*'connection previously silent and input unit active'*$) = (c_m - c)f_{\text{in}}$

(iii) $\mathcal{P}($*'connection previously silent and input unit inactive'*$) = (c_m - c)(1 - f_{\text{in}})$

(iv) $\mathcal{P}($*'connection previously functional and input unit inactive'*$) = c(1 - f_{\text{in}})$

Remember that the output unit that we are investigating is spurious with respect to the k-th pattern but it is active in pattern $k + \kappa$. For each of the four cases above, we derive the conditional probability of a functional connection based on the update rule that consists of Hebbian and homeostatic learning:

(i) If the connection is already functional and the corresponding input unit is active, the connection remains functional:
$\mathcal{P}(\text{'connection functional'} \mid \text{'connection previously functional and input unit active'}) = 1$
(yielding the green frame in Figure 7C)

(ii) If the connection is silent and the corresponding input unit is active, the connection is made functional with transition probability η:
$\mathcal{P}(\text{'connection functional'} \mid \text{'connection previously silent and input unit active'}) = \eta$
(yielding the green area in Figure 7C)

(iii) If the connection is silent and the input unit is inactive, the connection remains silent:
$\mathcal{P}(\text{'connection functional'} \mid \text{'connection previously silent and input unit inactive'}) = 0$

(iv) If the connection is functional and the input unit is inactive, the connection could be silenced due to the homeostasis mechanism (shown by the red frame in Figure 7C) or it could remain functional. As derived in Eq. (37), the probability that it is silenced is $\eta f_{\text{in}}(c_m - c)/\left(c(1 - f_{\text{in}})\right)$ and the probability that it remains functional is thus
$\mathcal{P}(\text{'connection functional'} \mid \text{'connection previously functional and input unit inactive'}) = 1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}$
(represented by the orange area without the red frame and without the green frame in Figure 7C).

In total, according to Eq. (38), the probability of the connection to be functional can be expressed as

$$\begin{aligned}
\rho_s(1) &= \mathcal{P}(\text{'connection functional'}) && (39)\\
&= 1 \cdot c f_{\text{in}} + \eta \cdot (c_m - c) f_{\text{in}} + 0 \cdot (c_m - c)(1 - f_{\text{in}}) \\
&\quad + \left(1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}\right) \cdot c(1 - f_{\text{in}}) && (40)\\
&= c && (41)
\end{aligned}$$

which is represented by the purple parts in Fig 7C.

The preservation of $\rho_s(k)$ relies on the statistical identity of units in the distribution of patterns. Specifically, every additional pattern in which a particular spurious output unit is active manipulates the connections to the spurious output unit in the same way. It follows that

$$\rho_s(u) = c, \quad \text{for all } u \in \mathbb{N} \tag{42}$$

and, in the following, we denote the distribution of the dendritic sums of spurious units for any P as

$$p_s(x) := p_s^{[0]}(x). \tag{43}$$

Probability of a functional connection to a genuine unit

We have seen that the probability ρ_s that a connection to a spurious unit is functional does not change with the output unit usage u. The probability ρ_g that a connection to a genuine unit is functional, however, does strongly depend on u. In the following, we first derive $\rho_g(0)$ and $\rho_g(1)$ and then generalize this derivation recursively to $\rho_g(u)$ for $u > 1$.

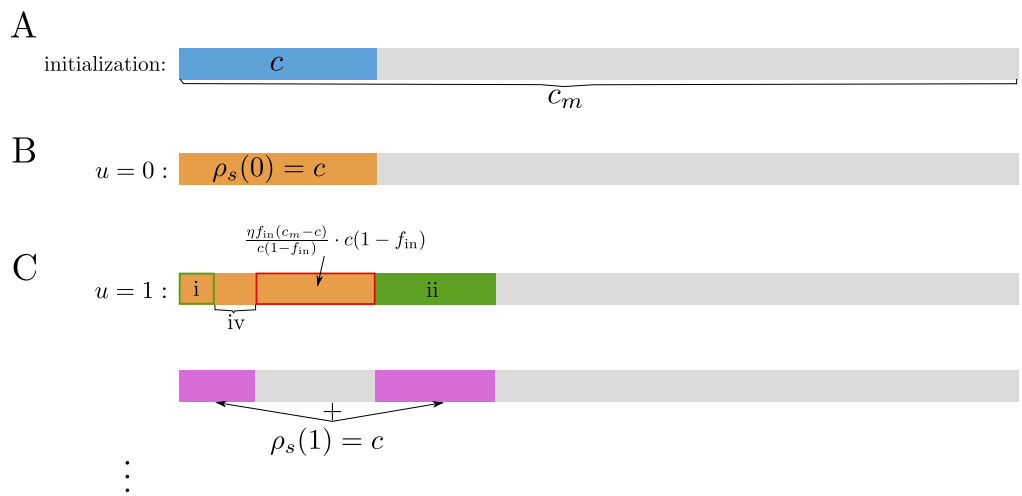

Fig 7. Illustration of the probability of a functional connection to a spurious unit. Fractions of functional connections targeting a spurious output unit if the output unit is active u times across the pattern set. Grey parts represent fractions of silent connections. Colored parts represent fractions of functional connections. (**A**) Configuration after initialization. (**B**) Configuration for $u = 0$. (**C**) Configuration for $u = 1$.

Derivation of $\rho_g(0)$ and $\rho_g(1)$ We aim at calculating the probability $\rho_g(u)$ that a genuine-genuine connection of the k-th pattern (Fig 8 left, labeled by f_{in}) is functional under the condition that the respective output unit is active u times in the subsequent patterns $k+1, \ldots, k+P$. Here, we compute $\rho_g(u)$ for $u = 0, 1$. For large numbers of such connections, i.e. for $f_{\text{in}} N_{\text{in}} \cdot f_{\text{out}} N_{\text{out}} \gg 1$, the probabilities $\rho_g(0)$ and $\rho_g(1)$ can, by the law of large numbers, be interpreted as fractions. Similarly to describing $\rho_g(u)$, we can describe the fraction of functional connections originating from input units that are inactive in the k-th pattern and targeting one specific genuine output unit, which are called spurious-genuine connections (Fig 8 right, labelled by $1 - f_{\text{in}}$). For the output unit being active never or once, we name these fractions $\rho_n(0)$ and $\rho_n(1)$, respectively. These probabilities of functional spurious-genuine connections are needed for deriving ρ_g but they do not explicitly contribute to the dendritic sum of the output unit because they do not receive any input.

As discussed in the previous section, connections targeting a spurious output unit are not modified during learning the k-th pattern, and the probability of a functional connection to a spurious unit after learning the k-th pattern is the same as the probability at the initialization. Since there is no correlation between input patterns, the probability of functionality is identical for genuine-spurious and spurious-spurious connections. In contrast, the probability of a functional connection to a genuine output unit after learning the k-th pattern (i.e., for $u = 0$) depends on whether the connection originates at an input unit that is active in the k-th pattern or not. Thus, we separately examine the probability of a functional genuine-genuine connection, ρ_g, and the probability of a spurious-genuine connection, ρ_n. Note that the two probabilities ρ_g and ρ_n, when weighted by the probabilities $\mathcal{P}(\text{'input unit active'}) = f_{\text{in}}$ and $\mathcal{P}(\text{'input unit inactive'}) = 1 - f_{\text{in}}$, respectively, should, nevertheless, sum up to the functional connectivity c:

$$\begin{aligned} &\mathcal{P}(\textit{'connection functional'}) \\ &= \rho_g(u) \cdot \mathcal{P}(\textit{'input unit active'}) + \rho_n(u) \cdot \mathcal{P}(\textit{'input unit inactive'}) = c \quad \forall u. \end{aligned} \tag{44}$$

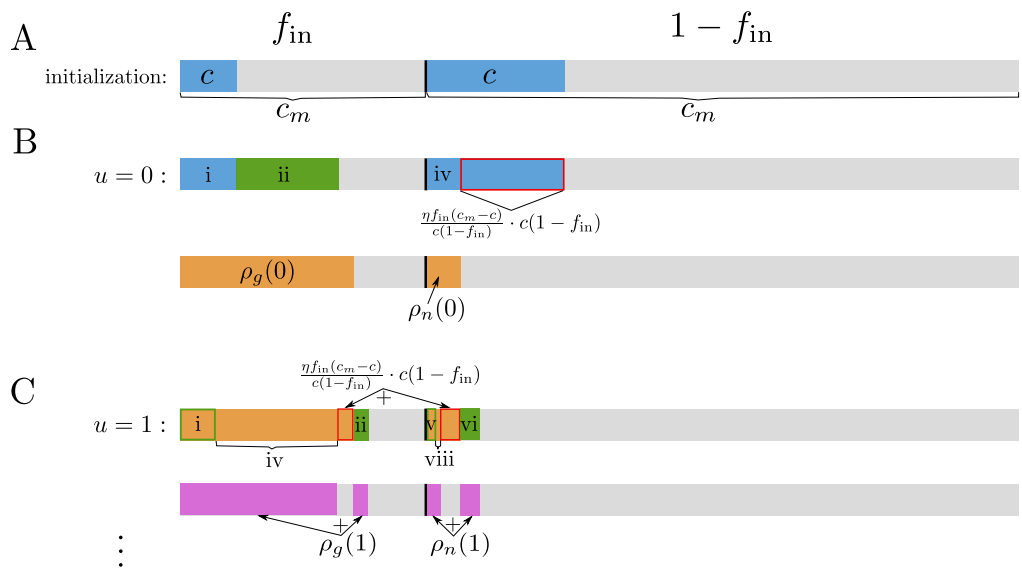

Fig 8. Illustration of the probability of a functional connection to a genuine unit. Fractions of functional connections targeting a genuine output unit if the output unit is active u times across the pattern set. Left (below "f_{in}"): fraction f_{in} of connections that originate from an input unit that is active in the k-th pattern and will thus contribute to the dendritic sum of the output unit when the k-th input pattern is applied — the genuine-genuine connections; right (below "$1 - f_{\text{in}}$"): fraction $1 - f_{\text{in}}$ of connections that originate from an input unit that is inactive in the k-th pattern — the spurious-genuine connections. Grey areas represent fractions of silent connections. Colored areas represent fractions of functional connections. (**A**) Configuration after initialization. (**B**) Configuration for $u = 0$. (**C**) Configuration for $u = 1$.

Output unit never active after learning the k-th pattern ($\mathbf{u = 0}$): For $\rho_g(0)$ and $\rho_n(0)$, only the storage of the k-th pattern plays a role. Starting from a random fraction c (thus a total number of cN_{in}) of functional connections targeting each output unit (blue parts in Fig 8A), we first consider connections only from an active input unit (left part, titled f_{in}, in Fig 8). We can split up the probability of a genuine-genuine connection to become functional into the probabilities of two mutually exclusive conditions and their corresponding conditional probabilities (see Eq. (38)). We have the two conditions

(i) $\mathcal{P}($*'g-g connection previously functional'*$) = c$

(ii) $\mathcal{P}($*'g-g connection previously silent'*$) = c_m - c$,

for each of which we compute the conditional probability of a functional genuine-genuine connection:

(i) If the connection is already functional and the corresponding input unit is active, the connection is protected and remains functional:
$\mathcal{P}($*'g-g connection functional'* $\mid$ *'g-g connection previously functional'*$) = 1$
(yielding the blue part in Figure 8B left)

(ii) If the connection is silent and the corresponding input unit is active, the connection is made functional with transition probability η:
$\mathcal{P}($*'g-g connection functional'* $\mid$ *'g-g connection previously silent'*$) = \eta$
(yielding the green part in Figure 8B)

According to the law of total probability, the probability of a functional genuine-genuine connection is thus

$$\rho_g(0) = c + \eta \cdot (c_m - c), \tag{45}$$

which is depicted by the left orange part in Fig 8B.

Analogously, the probability of a functional spurious-genuine connection is derived (right part, titled $1 - f_{\text{in}}$, in Fig 8). Given that the corresponding input unit is inactive, the two conditions have the probabilities

(iii) $\mathcal{P}(\text{'s-g connection previously silent'}) = c_m - c$

(iv) $\mathcal{P}(\text{'s-g connection previously functional'}) = c$,

and the corresponding conditional probabilities are the following:

(iii) If the connection is silent and the input unit is inactive, the connection remains silent:
$\mathcal{P}(\text{'s-g connection functional'} \mid \text{'s-g connection previously silent'}) = 0$

(iv) If the connection is functional and the input unit is inactive, the connection could either be silenced due to the homeostasis mechanism (represented by the red frame in Figure 8B) or it could remain functional. As before, the probability for case (iv) connections to remain functional is
$\mathcal{P}(\text{'s-g connection functional'} \mid \text{'s-g connection previously functional'})$
$= 1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}$
(yielding the blue part without the red frame in Figure 8B right).

Then, the total probability of a functional spurious-genuine connection is

$$\rho_n(0) = 0 \cdot (c_m - c) + \left(1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}\right) \cdot c = c - \frac{c_m - c}{1 - f_{\text{in}}} \eta f_{\text{in}}, \tag{46}$$

which is represented by the right orange part in Fig 8B. By using Eqs. (45) and (46) in Eq.(44), we see that the total probability of a functional connection to a genuine unit after learning the k-th pattern is again

$$\mathcal{P}(\text{'connection functional'}) = \rho_g(0) \cdot f_{\text{in}} + \rho_n(0) \cdot (1 - f_{\text{in}}) = c. \quad \checkmark \tag{47}$$

Output unit active once after learning the k-th pattern ($\mathbf{u} = 1$): Next, we derive $\rho_g(1)$ and $\rho_n(1)$, which correspond to the probabilities that a genuine-genuine or a spurious-genuine connection (with respect to the k-th pattern) is functional under the condition that the respective output unit is active exactly once among all subsequent patterns. The derivation of $\rho_g(1)$ and $\rho_n(1)$ is more involved because functional connections that originated from an active input unit in the k-th pattern (genuine-genuine connections) can be silenced while storing additional patterns, and in turn silent connections originating from inactive input units in the k-th pattern (spurious-genuine connections) can be made functional. We assume that the genuine output unit that we are investigating is active only for one pattern $k + \kappa$ with $\kappa \in \mathbb{N}^+$ fixed.

Let us start with the derivation of $\rho_g(1)$. The fraction of genuine-genuine connections being functional after storing pattern $k + \kappa$ (in addition to pattern k) can be split into four conditions depending on the previous state of the connection (i.e., depending on $\rho_g(0)$) and on the activity state of the corresponding input unit in pattern $k + \kappa$. The four conditions have the probabilities

(i) $\mathcal{P}($*'g-g connection previously functional and input unit active'* $) = \rho_g(0) f_{\text{in}}$

(ii) $\mathcal{P}($*'g-g connection previously silent and input unit active'* $) = (c_m - \rho_g(0)) f_{\text{in}}$

(iii) $\mathcal{P}($*'g-g connection previously silent and input unit inactive'* $)$
$= (c_m - \rho_g(0))(1 - f_{\text{in}})$

(iv) $\mathcal{P}($*'g-g connection previously functional and input unit inactive'* $) = \rho_g(0)(1 - f_{\text{in}})$

and the corresponding conditional probabilities are

(i) If the connection is already functional before storing pattern $k + \kappa$ and the corresponding input unit is active, the connection is protected and remains functional:
$\mathcal{P}($*'g-g connection functional'* $\mid$ *'g-g connection previously functional and input unit active'* $) = 1$
(represented by the green frame in Figure 8C left)

(ii) If the connection is silent and the corresponding input unit is active in pattern $k + \kappa$, the connection is made functional with transition probability η:
$\mathcal{P}($*'g-g connection functional'* $\mid$ *'g-g connection previously silent and input unit active'* $) = \eta$
(yielding the green area in Figure 8C left)

(iii) If the connection is silent and the input unit is inactive, the connection remains silent:
$\mathcal{P}($*'g-g connection functional'* $\mid$ *'g-g connection previously silent and input unit inactive'* $) = 0$

(iv) If the connection is functional and the input unit is inactive, the connection remains functional with probability
$\mathcal{P}($*'g-g syn. functional'* $\mid$ *'g-g syn. previously functional and input unit inactive'* $)$
$= 1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}$
(shown by the orange area without the red frame and without the green frame in Figure 8C left).

Using the law of total probability, we thus have

$$\rho_g(1) := 1 \cdot \rho_g(0) f_{\text{in}} + \eta \left(c_m - \rho_g(0)\right) f_{\text{in}} + 0 \cdot \left(c_m - \rho_g(0)\right)(1 - f_{\text{in}}) \tag{48}$$

$$+ \left(1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}\right) \rho_g(0)(1 - f_{\text{in}}) \tag{49}$$

$$= \rho_g(0) \left(1 - \frac{\eta f_{\text{in}} c_m}{c}\right) + \eta f_{\text{in}} c_m, \tag{50}$$

shown by the purple parts in Figure 8C left.

Analogously, the probability $\rho_n(1)$ of spurious-genuine connections being functional after storing pattern $k + \kappa$ consists of the four conditions

(v) $\mathcal{P}($*'s-g connection previously functional and input unit active'* $) = \rho_n(0) f_{\text{in}}$

(vi) $\mathcal{P}($*'s-g connection previously silent and input unit active'* $) = (c_m - \rho_n(0)) f_{\text{in}}$

(vii) $\mathcal{P}($*'s-g connection previously silent and input unit inactive'* $)$
$= (c_m - \rho_n(0))(1 - f_{\text{in}})$

(viii) $\mathcal{P}($*'s-g connection previously functional and input unit inactive'* $) = \rho_n(0)(1 - f_{\text{in}})$

and the four corresponding conditional probabilities

(v) $\mathcal{P}($*'s-g connection functional'* | *'s-g connection previously functional and input unit active'* $) = 1$
(represented by the green frame in Figure 8C right)

(vi) $\mathcal{P}($*'s-g connection functional'* | *'s-g connection previously silent and input unit active'* $) = \eta$
(yielding the green area in Figure 8C right)

(vii) $\mathcal{P}($*'s-g connection functional'* | *'s-g connection previously silent and input unit inactive'* $) = 0$

(viii) $\mathcal{P}($*'s-g syn. functional'* | *'s-g syn. previously functional and input unit inactive'* $)$
$= 1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1-f_{\text{in}})}$
(represented by the orange area without the red frame and without the green frame in Figure 8C right)

According to the law of total probability, the probability of a functional spurious-genuine connection amounts to

$$\rho_n(1) := 1 \cdot \rho_n(0) f_{\text{in}} + \eta\,(c_m - \rho_n(0))\, f_{\text{in}} + 0 \cdot (c_m - \rho_n(0))\,(1 - f_{\text{in}}) \tag{51}$$

$$+ \left(1 - \frac{\eta f_{\text{in}}(c_m - c)}{c(1 - f_{\text{in}})}\right) \rho_n(0)(1 - f_{\text{in}}) \tag{52}$$

$$= \rho_n(0) \left(1 - \frac{\eta f_{\text{in}} c_m}{c}\right) + \eta f_{\text{in}} c_m, \tag{53}$$

which corresponds to the purple parts in Fig 8C right.

Generalization for any output unit usage $u > 0$ The probability of a functional connection, if the corresponding output unit is active u times across the whole pattern set, can be computed recursively. The conditional probabilities remain the same in every step, while the four conditions always depend on the probabilities $\rho_g(u-1)$ and $\rho_n(u-1)$ of the previous step. We thus obtain $\rho_g(2)$ from $\rho_g(1)$ the same way we obtained $\rho_g(1)$ from $\rho_g(0)$,

$$\rho_g(2) = \rho_g(1) \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right) + f_{\text{in}} \eta c_m\ , \tag{54}$$

and in general we get

$$\rho_g(u) = \rho_g(u-1) \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right) + f_{\text{in}} \eta c_m. \tag{55}$$

$\rho_n(u)$ can be derived in the same way and we have

$$\rho_n(u) = \rho_n(u-1) \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right) + f_{\text{in}} \eta c_m, \text{ with } \rho_n(0) = c - \frac{c_m - c}{1 - f_{\text{in}}} \eta f_{\text{in}}, \tag{56}$$

where the only difference from $\rho_g(u)$ is the initial value $\rho_n(0) \neq \rho_g(0)$ (see Eqs. (45) and (46)).

From the recursive description of $\rho_g(u)$ (Eqs. (45) and (55)), we can derive an explicit description

$$\rho_g(u) = (c_m - c)\eta \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right)^u + c. \tag{57}$$

Examples of the probability for a genuine-genuine connection to be functional $\rho_g(u)$ as a function of the output unit usage u are shown in Fig 9. Analogously, the explicit description of ρ_n is

$$\rho_n(u) = \frac{-f_{\text{in}}\eta(c_m - c)}{1 - f_{\text{in}}}\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^u + c, \tag{58}$$

and it will come up again later when we discuss retrieval with noisy cues.

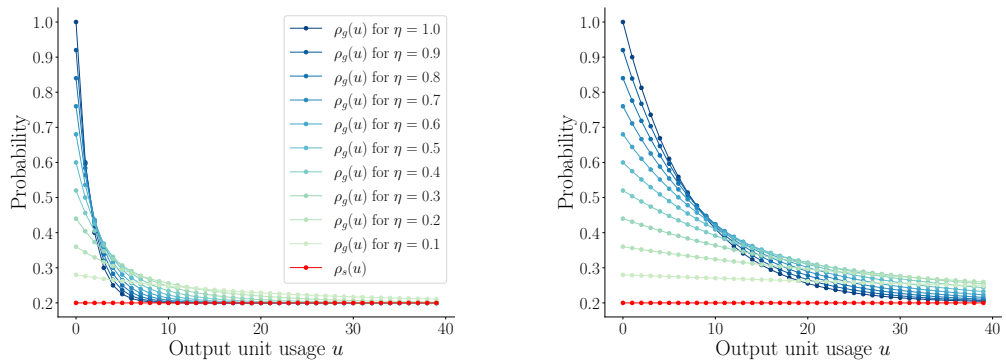

Fig 9. Probabilities of functional connections $\rho_g(u)$ and $\rho_s(u)$. Probability $\rho_l(u)$, $l \in \{s, g\}$, of a connection to be functional depending on output unit usage u for various transition probabilities η. Blue dots: genuine-genuine connections (Eq. (57)); red dots: genuine-spurious connections (Eq. (42)). Left: $f_{\text{in}} = 0.1$, right: $f_{\text{in}} = 0.025$. Probability ρ_s of genuine-spurious connections to be functional is constant as a function of u. Probability ρ_g of genuine-genuine connections to be functional decreases exponentially with u. (Other parameters: $c = 0.2, c_m = 1$.)

To summarize, we have now fully characterized the distributions of dendritic sums for spurious and for genuine output units. We insert the probability of a functional connection to a spurious unit $\rho_s(u) = c$ into Eq. (34) and obtain the probability mass function (PMF) of the dendritic sum of a spurious output unit

$$p_s(x) = \mathcal{P}(d_s = x) = \sum_{u=0}^{P} \binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u} \binom{M_{\text{in}}}{x} c^x (1-c)^{M_{\text{in}}-x} \tag{59}$$

$$= \binom{M_{\text{in}}}{x} c^x (1-c)^{M_{\text{in}}-x} \tag{60}$$

$$= \mathcal{B}_{M_{\text{in}},\, c}(x). \tag{61}$$

Since $\rho_s(u)$ is constant as a function of the output unit usage u, the distribution of dendritic sums for spurious output units p_s simplifies to a single binomial distribution $\mathcal{B}_{n,\,p}$ with parameters $n = M_{\text{in}}$ and $p = c$, and p_s does not depend on the number of subsequently learned patterns P.

Analogously, we insert $\rho_g(u) = (c_m - c)\eta\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^u + c$ into Eq. (34) and obtain

the PMF of the dendritic sum of a genuine output unit

$$
\begin{aligned}
p_g^{[P]}(x) &= \mathcal{P}(d_g^{[P]} = x) \\
&= \sum_{u=0}^{P} \left[\binom{P}{u} {f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u} \binom{M_{\text{in}}}{x} \left((c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c} \right)^u + c \right)^x \right. \\
&\qquad \left. \cdot \left(1 - (c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c} \right)^u - c \right)^{M_{\text{in}}-x} \right] \qquad (62) \\
&= \sum_{u=0}^{P} \left[\mathcal{B}_{P,\, f_{\text{out}}}(u) \cdot \mathcal{B}_{M_{\text{in}},\, \rho_g(u)}(x) \right]. \qquad (63)
\end{aligned}
$$

Thus, $p_g^{[P]}$ depends on the number of subsequent patterns P (in contrast to p_s). It is not a single binomial but a linear combination of binomial PMFs $\mathcal{B}_{M_{\text{in}},\, \rho_g(u)}(x)$, for $u \in \{0, \ldots, P\}$, see Fig 10. This multimodality is analyzed in more detail in S1 Appendix. The relation between p_s and $p_g^{[P]}$ and their dependence on the activation ratios f_{in} and f_{out} is discussed in the Results. S5 Fig shows a comparison of Eqs. (61) and (63) to distributions of dendritic sums obtained from numerical simulations.

Let us finally discuss some limiting cases. We have

$$f_{\text{in}}\eta c_m \leq f_{\text{in}} c_m \leq f_{\text{in}} \leq c \qquad (64)$$

because $0 < \eta \leq 1$, $0 < c_m \leq 1$, $0 < f_{\text{in}}$, and $f_{\text{in}} \leq c$. Note that the constraint $f_{\text{in}} \leq c$ stems from the normalization condition as discussed previously. It thus follows that we either have

$$0 < \left(1 - \frac{f_{\text{in}}\eta c_m}{c} \right) < 1, \qquad (65)$$

which implies

$$\rho_g(u) \xrightarrow{u \to \infty} c \qquad (66)$$

or, for $c_m = 1$, $\eta = 1$, and $f_{\text{in}} = c$,

$$\left(1 - \frac{f_{\text{in}}\eta c_m}{c} \right) = 0, \qquad (67)$$

which implies

$$\rho_g(0) = 1 \qquad (68)$$

$$\rho_g(u) = c, \quad \text{for } u > 0. \qquad (69)$$

In the second case, the part of the memory stored in the connections targeting one output unit is completely lost as soon as this output unit is active one additional time for another pattern.

Noisy input patterns during retrieval

So far we always assumed that during recall the original input pattern is shown to the network. Due to the inherent presence of noise in biological systems, it makes sense to assume that the input pattern that is presented to the network during retrieval is not exactly the same as the input pattern used for training. In the following, we discuss the consequences on the previous derivations when a noisy cue is presented to the network

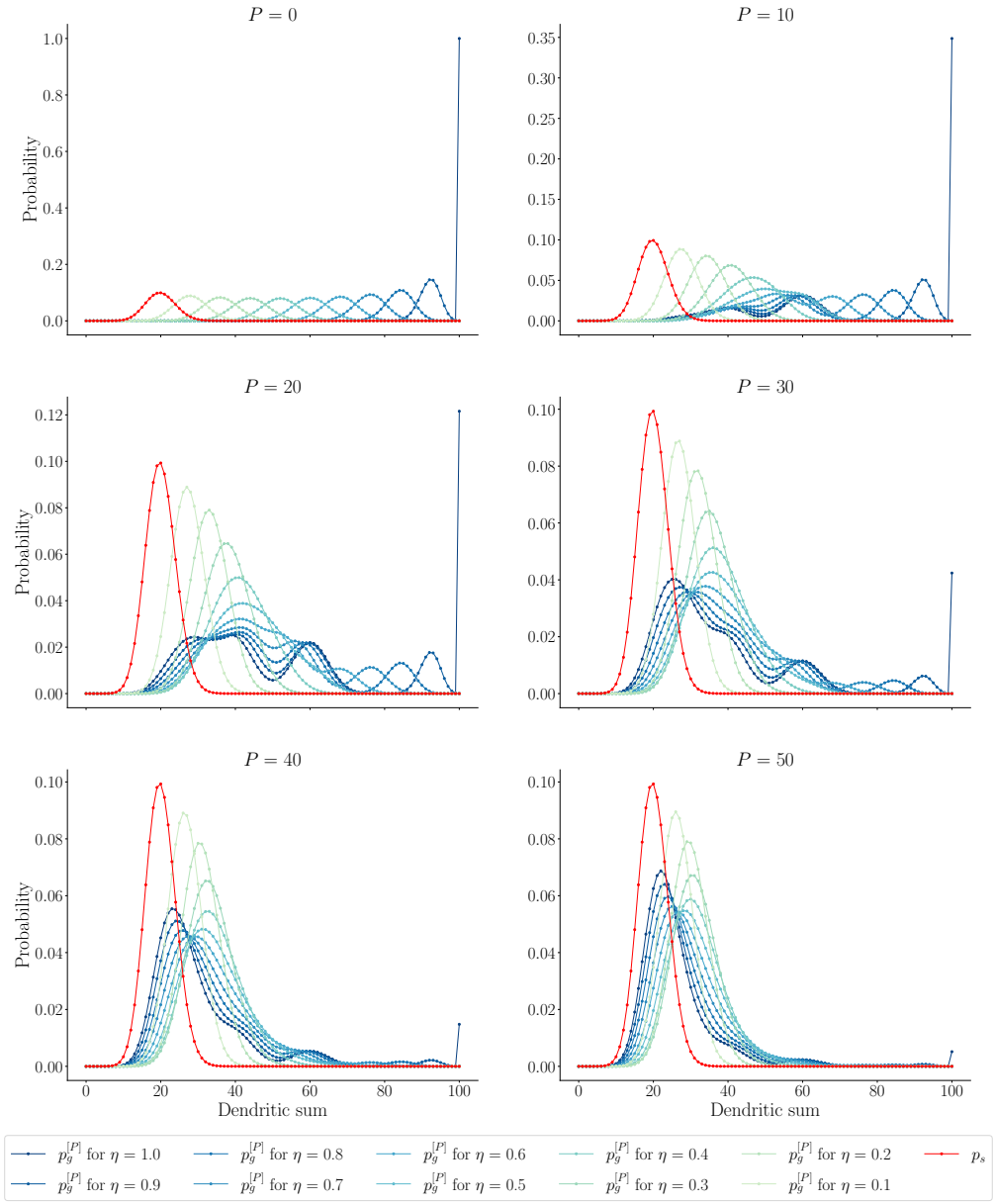

Fig 10. Distributions of dendritic sums. Analytical probability mass functions of dendritic sums after storing $P = 0, 10, 20, 30, 40, 50$ patterns (Eqs. (61) and (63)). Spurious units in red; genuine units in blue shades, for several transition probabilities η. With an increasing number of subsequent patterns P, the distributions of the dendritic sums of genuine units approach the distribution of the dendritic sums of spurious units. The genuine distributions are multimodal, in particular for large transition probabilities η and small numbers of patterns $P > 0$. Other parameters: $N_{\text{in}} = N_{\text{out}} = 1000, f_{\text{in}} = f_{\text{out}} = 0.1, c = 0.2, c_m = 1$.

by putting noise onto the input pattern. To be able to compare the results in a fair fashion, the input activation ratio f_{in} should not be changed, hence a fraction ε of the input units that are active in the original pattern (genuine input units) is deactivated while the same absolute number of input units that are inactive in the original pattern (spurious input units) is activated. Thus there is a number $m_g = M_{\text{in}} \cdot (1-\varepsilon)$ of original input units (genuine input units) and a number $m_s = M_{\text{in}} \cdot \varepsilon$ of wrong input units (spurious input units) active, in total we have $M_{\text{in}} = m_g + m_s$ active units [69]. The parameter ε is restricted to values that yield $m_g, m_s \in \mathbb{N}$. Note that, in the numerical simulations, we set each of the genuine input units to zero with probability ε, count how many have changed and flip the same amount of spurious input units to one. This means m_g and m_s are not fixed exact numbers but follow distributions. Only *on average*, $M_{\text{in}}\varepsilon$ genuine units are deactivated. The additional variability coming from deactivating each genuine input unit with probability ε would make the analytical description more involved while hardly affecting the expected result. It is therefore neglected in the theory derived in this section. Instead, we assume that, for generating a noisy cue, exactly $M_{\text{in}}\varepsilon$ of the genuine input units are deactivated and $M_{\text{in}}\varepsilon$ of the spurious input units are activated. Note that, nevertheless, in both the numerical simulations and the theory, it is always ensured that the absolute number of active units M_{in} in the noisy input pattern that is presented to the network does not change.

Noisy cues alter the distribution of the dendritic sums of genuine output units but not of spurious output units. The probability of a functional genuine-genuine connection is the same as before, ρ_g. The probability of a functional spurious-genuine connection has already been introduced as $\rho_n(u)$ (Eq. (58)). Analogously to $\rho_g(u)$ in Eq. (55) and Eq. (57), $\rho_n(u)$ can be expressed recursively as

$$\rho_n(u) = \rho_n(u-1)\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^u + f_{\text{in}}\eta c_m, \text{ with } \rho_n(0) = c - \frac{c_m - c}{1 - f_{\text{in}}}\eta f_{\text{in}}, \tag{70}$$

or explicitly as

$$\rho_n(u) = \frac{-f_{\text{in}}\eta(c_m - c)}{1 - f_{\text{in}}}\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^u + c, \tag{71}$$

and is shown in Fig 11. Just like $\rho_g(u)$, $\rho_n(u)$ depends on the number of times u an output unit has been active across the pattern set. The probability of a functional spurious-genuine connection ρ_n is lower than the probability of a functional genuine-genuine connection ρ_g (and even lower than the functional connectivity c, see Figs 11 and 9).

For a particular output unit usage u, the dendritic sums $d_g^{[P]}$ are distributed following

$$\begin{aligned}\mathcal{P}(d_g^{[P]} = x \mid r = u) = \sum_{x_g=0}^{x} \Bigg(&\binom{m_g}{x_g} \rho_g(u)^{x_g}(1-\rho_g(u))^{m_g - x_g} \\ &\cdot \binom{m_s}{x_s} \rho_n(u)^{x_s}(1-\rho_n(u))^{m_s - x_s}\Bigg),\end{aligned} \tag{72}$$

where $x_s = x - x_g$. In total, the probability of the dendritic sum $d_g^{[P]}$ of a genuine

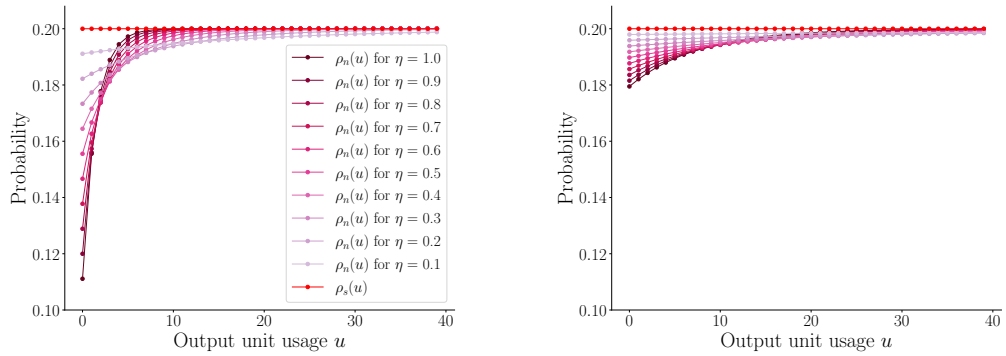

Fig 11. Probabilities of functional connections $\rho_n(u)$ and $\rho_s(u)$. Probability $\rho_l(u)$, $l \in n, s$, of a connection to be functional depending on output unit usage u. Pink shades: spurious-genuine connections ($\rho_n(u)$, Eq. (71)); red: connections to spurious units ($\rho_s(u)$, Eq. (42)). Left: $f_{\text{in}} = 0.1$, right: $f_{\text{in}} = 0.025$. Probability ρ_s of genuine-spurious connections to be functional is constant as a function of u. Probability ρ_n of spurious-genuine connections to be functional increases exponentially with u. (Other parameters: $c = 0.2, c_m = 1$.)

output unit to have a particular value x can be described by

$$\begin{aligned} p_{g,\varepsilon}^{[P]}(x) &= \mathcal{P}(d_g^{[P]} = x) \\ &= \sum_{u=0}^{P} \binom{P}{u} \Bigg[{f_{\text{out}}}^u (1 - f_{\text{out}})^{P-u} \sum_{x_g=0}^{x} \Bigg(\binom{m_g}{x_g} \rho_g(u)^{x_g} (1 - \rho_g(u))^{m_g - x_g} \\ &\qquad \cdot \binom{m_s}{x_s} \rho_n(u)^{x_s} (1 - \rho_n(u))^{m_s - x_s} \Bigg) \Bigg] \qquad (73) \\ &= \sum_{u=0}^{P} \left[\mathcal{B}_{P, f_{\text{out}}}(u) \sum_{x_g=0}^{x} \left(\mathcal{B}_{m_g, \rho_g(u)}(x_g) \; \mathcal{B}_{m_s, \rho_n(u)}(x_s) \right) \right] \qquad (74) \end{aligned}$$

[69]. Compared to the distribution of the dendritic sums for the noise-less input pattern, this distribution is shifted to the left. Thus, it is closer to the distribution of the dendritic sums of the spurious units, the Hamming distance between the target output and the calculated output pattern is larger, and the capacity thus decreases (compare Fig 12 and S3 Fig to Fig 10).

Analytical calculation of the capacity of the network

In what follows, our aim is to analytically quantify the capacity of the network. The capacity is defined as the number of subsequent patterns P for which the signal quality has declined to the retrieval threshold. Therefore, we first discuss the signal quality as a function of P. Before we enter the details, we briefly recap basic quantities: The difference between a calculated output pattern $\hat{\mathbf{y}}^{[k]}(P)$ given a particular input pattern $\mathbf{x}^{[k]}$ after learning P subsequent pattern pairs and the corresponding target output pattern $\mathbf{y}^{[k]}$ is measured in terms of the Hamming distance between the two, denoted by $H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)$. This Hamming distance is always averaged across a large number of patterns, and it lies between 0 and the average Hamming distance between two random patterns of length N_{out} and activation ratio f_{out}, which is

$$H_{\text{avg}} := 2 N_{\text{out}} f_{\text{out}} (1 - f_{\text{out}}). \qquad (75)$$

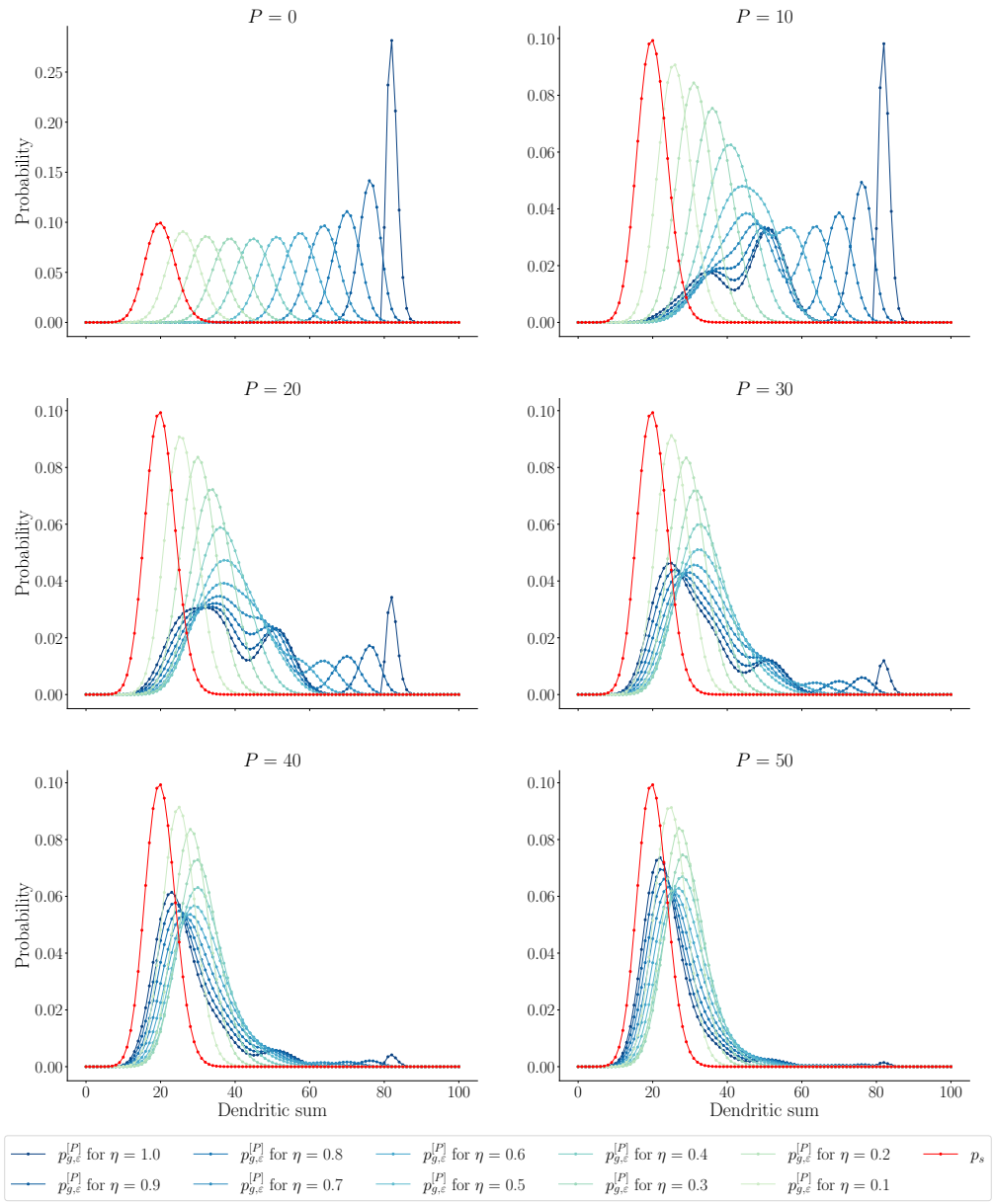

Fig 12. Distributions of dendritic sums with noise. Analytical probability mass functions of dendritic sums with noise on the input patterns during retrieval after storing $P = 0, 10, 20, 30, 40, 50$ patterns (Eq. (73) with (57) and (58); Eq. (34) with (42)). Spurious units in red; genuine units in blue shades, for several transition probabilities η. With an increasing number of subsequent patterns P, the distributions of the dendritic sums of genuine units approach the distribution of the dendritic sums of spurious units. Compared to the noise-less case (see Fig 10), the genuine distributions are closer to the spurious distribution. Other parameters:
$N_{\text{in}} = 1000, f_{\text{in}} = 0.1, N_{\text{out}} = 1000, f_{\text{out}} = 0.1, c = 0.2, c_m = 1, \varepsilon = 0.2$. The same figure but with $\varepsilon = 0.4$ can be found in S3 Fig.

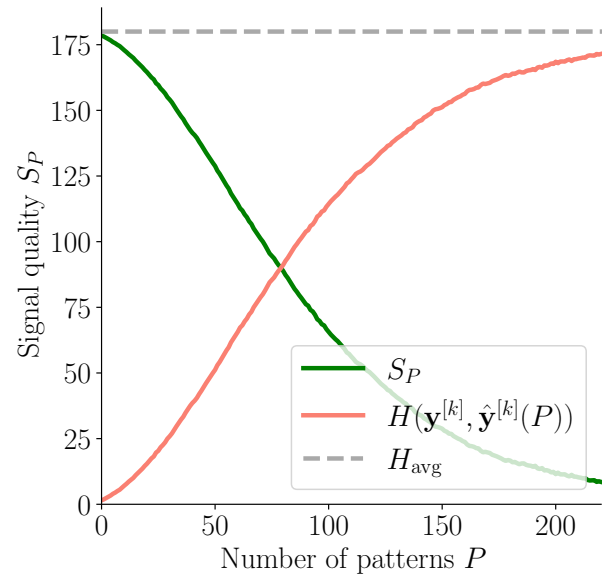

Fig 13. Hamming distance and signal quality. The Hamming distance $H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)$ between the target and the calculated output pattern (salmon line) increases with an increasing number of patterns P and approaches H_{avg} (grey line) for $P \to \infty$. The signal quality S_P (green line) decreases with increasing P and approaches 0.

We define the memory signal quality S_P at P patterns as the difference between H_{avg} and the Hamming distance between the target output and the calculated output $H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)$:

$$S_P := H_{\text{avg}} - H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right). \tag{76}$$

The signal quality can be expressed as a fraction s_P of H_{avg}

$$S_P = s_P \cdot H_{\text{avg}}, \quad \text{where } s_P \in [0, 1], \tag{77}$$

see Fig 13, green line. We call s_P the normalized signal quality. With s_P, the Hamming distance reads as

$$H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right) = (1 - s_P) \cdot H_{\text{avg}} = \overline{s_P} \cdot H_{\text{avg}} \tag{78}$$

(see Fig 13, salmon line), where we introduce the additional symbol

$$\overline{s_P} := 1 - s_P \tag{79}$$

out of convenience for the derivations in this section.

Analytical Hamming distance

In this section, we derive the expected Hamming distance $\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right)$ between the target output $\mathbf{y}^{[k]}$ and the output $\hat{\mathbf{y}}^{[k]}(P)$ calculated after learning P additional patterns.

The activation threshold and balancing tails of the distributions Since this Hamming distance depends on the activation threshold $T_{\text{in}}^{[P]}$, we first discuss the choice of $T_{\text{in}}^{[P]}$, which is self-consistently defined such that the correct number of output units, M_{out}, is activated:

$$\sum_{j=1}^{N_{\text{out}}} \hat{y}_j^{[k]}(P) \stackrel{!}{=} M_{\text{out}}. \tag{80}$$

The activation of output unit j is calculated as

$$\hat{y}_j^{[k]}(P) = \Theta\left(\left(\sum_{i=1}^{N_{\text{in}}} J_{ji}^{[k+P]} x_i^{[k]}\right) - T_{\text{in}}^{[P]}\right) = \Theta\left(\left(J^{[k+P]}\mathbf{x}^{[k]}\right)_j - T_{\text{in}}^{[P]}\right), \tag{81}$$

where $\Theta(x) = \begin{cases} 1, & \text{for } x \geq 0 \\ 0, & \text{for } x < 0 \end{cases}$ is the Heaviside step function. The number of active output units is hence

$$\sum_{j=1}^{N_{\text{out}}} \hat{y}_j^{[k]}(P) = \sum_{j=1}^{N_{\text{out}}} \Theta\left(\left(J^{[k+P]}\mathbf{x}^{[k]}\right)_j - T_{\text{in}}^{[P]}\right). \tag{82}$$

Since the weight matrix $J^{[k+P]}$ is updated in every learning step and hence depends on the number of subsequently learned patterns P, the activation threshold $T_{\text{in}}^{[P]}$ also depends on P. It thus has to be newly determined in every update step in order to obtain the given number of active output units M_{out}.

The active output units can either be genuine, hence active in the target output pattern, or spurious, hence inactive in the target output pattern. The dendritic sum

$$\left(d_g^{[P]}\right)_j = \left(J^{[k+P]}\mathbf{x}^{[k]}\right)_j \sim p_g^{[P]} \tag{83}$$

of a genuine output unit j follows the genuine distribution $p_g^{[P]}$ (Eq. (63)). A genuine output unit is activated if its dendritic sum is at least as large as the activation threshold $T_{\text{in}}^{[P]}$, hence if

$$\left(J^{[k+P]}\mathbf{x}^{[k]}\right)_j \geq T_{\text{in}}^{[P]} \tag{84}$$

(blue area in Fig 14A). There is a total amount of M_{out} genuine output units. Thus, if we assume $M_{\text{out}} \gg 1$, the number of genuine output units that are activated for a given threshold $T_{\text{in}}^{[P]}$ can be approximated by using the law of large numbers as

$$\sum_{j \in G} \hat{y}_j^{[k]}(P) \approx M_{\text{out}} \cdot \left\langle \Theta\left(\left(d_g^{[P]}\right)_j - T_{\text{in}}^{[P]}\right)\right\rangle_{\left(d_g^{[P]}\right)_j \sim p_g^{[P]}} \tag{85}$$

$$= M_{\text{out}} \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]}\right\rceil}^{M_{\text{in}}} p_g^{[P]}(x), \tag{86}$$

where G is the set of indices of genuine output units. Analogously, assuming $N_{\text{out}} - M_{\text{out}} \gg 1$, the number of spurious units that are activated can be approximated as

$$\sum_{j \in S} (y_j^{[P]}) \approx (N_{\text{out}} - M_{\text{out}}) \cdot \left\langle \Theta\left((d_s)_j - T_{\text{in}}^{[P]}\right)\right\rangle_{(d_s)_j \sim p_s} \tag{87}$$

$$= (N_{\text{out}} - M_{\text{out}}) \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]}\right\rceil}^{M_{\text{in}}} p_s(x), \tag{88}$$

where $(d_s)_j$ denotes the dendritic sum of the j-th output unit, which is spurious, and S is the set of indices of spurious output units (cf. red area in Fig 14A). In order to

activate the right amount of output units, the activation threshold $T_{\text{in}}^{[P]}$ thus has to be chosen such that

$$M_{\text{out}} \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]} \right\rceil}^{M_{\text{in}}} p_g^{[P]}(x) + (N_{\text{out}} - M_{\text{out}}) \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]} \right\rceil}^{M_{\text{in}}} p_s(x) = M_{\text{out}}, \tag{89}$$

see Fig 14B. By basic computations, we have that

$$M_{\text{out}} \cdot \sum_{x=0}^{\left\lfloor T_{\text{in}}^{[P]} \right\rfloor} p_g^{[P]}(x) = (N_{\text{out}} - M_{\text{out}}) \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]} \right\rceil}^{M_{\text{in}}} p_s(x) \tag{90}$$

$$\Leftrightarrow M_{\text{out}} \cdot F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) = (N_{\text{out}} - M_{\text{out}}) \cdot \left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right) \tag{91}$$

$$\Leftrightarrow f_{\text{out}} \cdot F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) = (1 - f_{\text{out}}) \cdot \left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right), \tag{92}$$

where $F_g^{[P]}$ and F_s are the cumulative distribution functions (CDF) of $p_g^{[P]}$ and p_s, respectively (Fig 14C). We call the last equation the Balance Equation because it balances the false positive and the false negative errors.

With an increasing number of patterns P, the center of mass of the genuine distribution $p_g^{[P]}$ moves to the left, while the spurious distribution stays constant. For a fixed T_{in}, the left-hand side of Eq. (92) would hence grow while the right-hand side would not, which would lead to an output activation ratio that is smaller than f_{out}. In order to keep the Balance Equation fulfilled, the activation threshold $T_{\text{in}}^{[P]}$ has to move to the left for increasing P (Fig 14C, $P = 68$, and 14D, $P = 114$).

Deriving the Hamming distance from dendritic sums The Hamming distance

$$H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right) := \#\left\{j \in \{0, \ldots, N_{\text{out}}\} | y_j^{[k]} \neq \left(\hat{y}^{[k]}(P)\right)_j\right\} \tag{93}$$

is the sum of the number of genuine units that are wrongly set to 0 and the number of spurious units that are wrongly set to 1 in the calculated output $\hat{\mathbf{y}}^{[k]}(P)$:

$$H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right) = \sum_{j \in G}\left(1 - \hat{y}_j^{[k]}(P)\right) + \sum_{j \in S} \hat{y}_j^{[k]}(P). \tag{94}$$

In terms of the distributions p_s and $p_g^{[P]}$, the expected Hamming distance reads as

$$\begin{aligned}
\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) &= M_{\text{out}} \cdot \left\langle 1 - \Theta\left(\left(d_g^{[P]}\right)_j - T_{\text{in}}^{[P]}\right)\right\rangle_{\left(d_g^{[P]}\right)_j \sim p_g^{[P]}} \\
&\quad + (N_{\text{out}} - M_{\text{out}}) \cdot \left\langle \Theta\left((d_s)_j - T_{\text{in}}^{[P]}\right)\right\rangle_{(d_s)_j \sim p_s} && (95) \\
&= M_{\text{out}} \cdot \sum_{x=0}^{\left\lfloor T_{\text{in}}^{[P]} \right\rfloor} p_g^{[P]}(x) + (N_{\text{out}} - M_{\text{out}}) \cdot \sum_{x=\left\lceil T_{\text{in}}^{[P]} \right\rceil}^{M_{\text{in}}} p_s(x) && (96) \\
&= M_{\text{out}} F_g^{[P]}\left(T_{\text{in}}^{[P]}\right) + (N_{\text{out}} - M_{\text{out}})\left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right), && (97)
\end{aligned}$$

see Fig 15A. The Hamming distance is hence the sum of the part of the genuine distribution that lies below the activation threshold $T_{\text{in}}^{[P]}$ and the part of the spurious

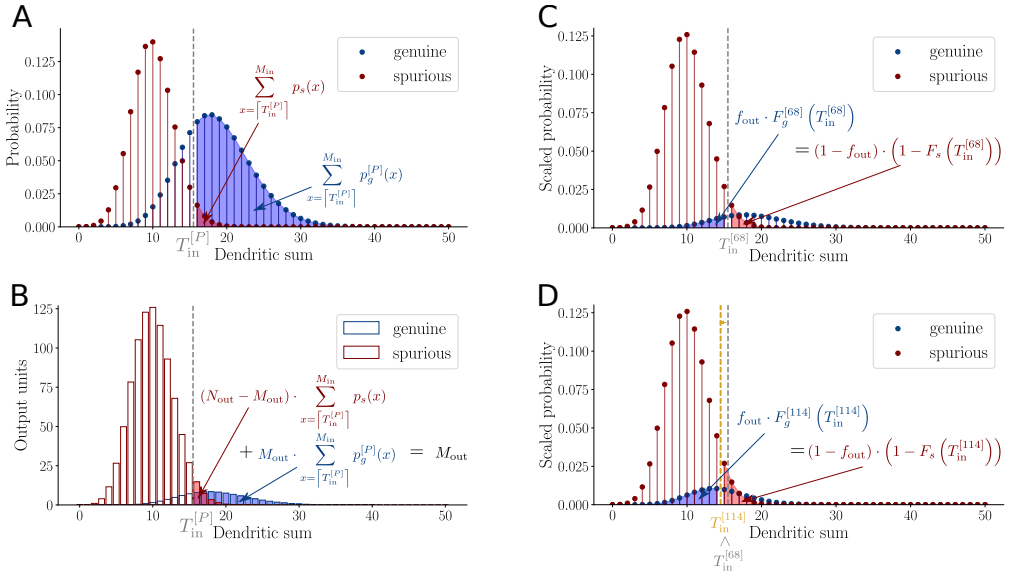

Fig 14. The activation threshold and balancing tails of the distributions. (**A**) Distributions of dendritic sums of genuine (blue) and spurious (red) output units. Output units with a dendritic sum larger than the activation threshold $T_{\text{in}}^{[P]}$ are activated. (**B**) Distributions of dendritic sums scaled by the number of units whose dendritic sums follow the respective distributions. The sum of the number of genuine (blue) and the number of spurious (red) units that have a dendritic sum larger than the threshold $T_{\text{in}}^{[P]}$ corresponds to the predefined number of active output units M_{out}. (**C**) Distributions of dendritic sums of genuine (blue) and spurious (red) units scaled by f_{out} and $1 - f_{\text{out}}$, respectively. The mass of the part of the genuine distribution left of the activation threshold $T_{\text{in}}^{[P]}$ (blue area) must correspond to the mass of the part of the spurious distribution right of $T_{\text{in}}^{[P]}$ (red area). (**D**) Same as (**C**) but for larger number of patterns $P = 114$ instead of $P = 68$. The threshold $T_{\text{in}}^{[114]}$ ($< T_{\text{in}}^{[68]}$) moves to the left in order to maintain the balance of the tails, which ensures the right amount of active output units.

distribution that lies above $T_{\text{in}}^{[P]}$ — weighted by the corresponding number of units. Using the Balance Equation (92), we can express $F_g^{[P]}\left(T_{\text{in}}^{[P]}\right)$ in terms of $F_s\left(T_{\text{in}}^{[P]}\right)$, and the expected Hamming distance simplifies to

$$\mathbb{E}\left(H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right)\right) = 2N_{\text{out}}(1 - f_{\text{out}})\left(1 - F_s\left(T_{\text{in}}^{[P]}\right)\right). \tag{98}$$

The dependence on the number of patterns P is now solely contained in the activation threshold $T_{\text{in}}^{[P]}$ because F_s is constant in P. The Hamming distance increases with decreasing $T_{\text{in}}^{[P]}$ and, as discussed previously, $T_{\text{in}}^{[P]}$ decreases with increasing P. Thus, the larger the number of subsequently learned patterns P, the larger the Hamming distance between target output and calculated output and, according to Eq. (76), the smaller the signal quality S_P.

Note that the dependence of the Hamming distance on the output activation ratio f_{out} is less straightforward. On the one hand, if f_{out} is increased, the factor $1 - f_{\text{out}}$ in Eq. (98) yields a decrease in the Hamming distance. On the other hand, an increase of f_{out} also decreases the activation threshold $T_{\text{in}}^{[P]}$ and thus increases the factor $1 - F_s(T_{\text{in}}^{[P]})$ in Eq. (98), which yields an increase in the Hamming distance (see Fig

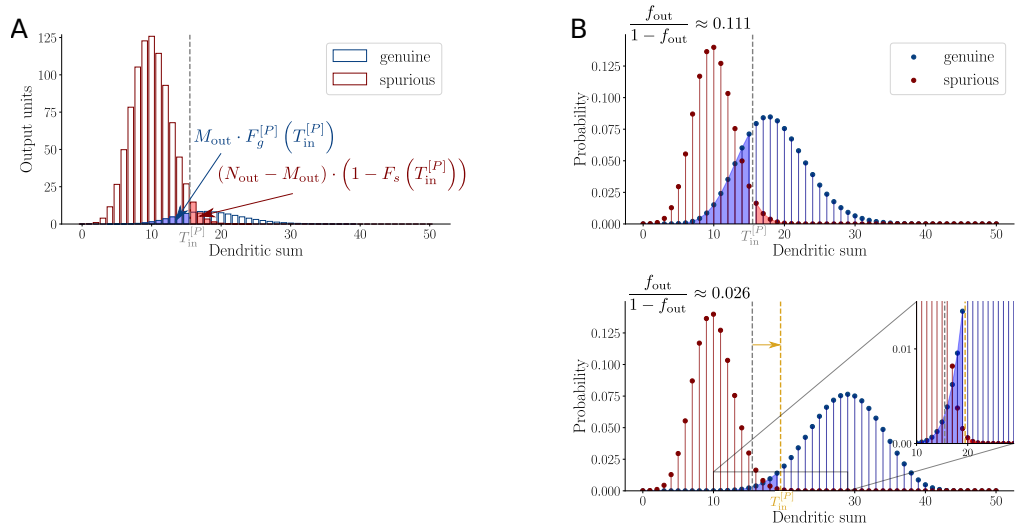

Fig 15. Deriving the Hamming distance from dendritic sums. (**A**) Hamming distance is the sum of falsely deactivated genuine units (blue area) and falsely activated spurious units (red area). (**B**) If f_{out} is changed, the threshold $T_{\text{in}}^{[P]}$ needs to be adapted such that the ratio of the red area to the blue area corresponds to the ratio $\frac{f_{\text{out}}}{1-f_{\text{out}}}$. Upper: $f_{\text{out}} = 0.1$, $\frac{f_{\text{out}}}{1-f_{\text{out}}} \approx 0.111$, lower: $f_{\text{out}} = 0.025$, $\frac{f_{\text{out}}}{1-f_{\text{out}}} \approx 0.026$ and hence the red area is much smaller compared to the blue area in the lower panel than in the upper panel. This is achieved by a larger $T_{\text{in}}^{[P]}$, which can also affect the Hamming distance.

15B). It is not easy to determine analytically which of these opposing effects is stronger. As mentioned in the Results, we speculate that the combined effect on the Hamming distance is small in the parameter range that we investigated numerically.

Signal quality after P subsequent patterns

In order to derive the signal quality $S_P = s_P H_{\text{avg}}$ (see Eqs. (76) and (77)) after P additional learning steps, we combine Eqs. (78) and (98) and obtain

$$\overline{s_P} \cdot H_{\text{avg}} = 2N_{\text{out}}(1 - f_{\text{out}})(1 - F_s(T_{\text{in}}^{[P]})). \tag{99}$$

Inserting $H_{\text{avg}} := 2N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}})$ yields

$$\overline{s_P} f_{\text{out}} = 1 - F_s(T_{\text{in}}^{[P]}) \tag{100}$$

$$\Leftrightarrow T_{\text{in}}^{[P]} = F_s^{-1}\left(1 - \overline{s_P} f_{\text{out}}\right). \tag{101}$$

Using Eq. (100), we find that the Balance Equation (92) is fulfilled if and only if

$$F_g^{[P]}(T_{\text{in}}^{[P]}) = \overline{s_P}(1 - f_{\text{out}}) \tag{102}$$

$$\Leftrightarrow T_{\text{in}}^{[P]} = {F_g^{[P]}}^{-1}\left(\overline{s_P}(1 - f_{\text{out}})\right). \tag{103}$$

Fig 16 illustrates the two equations. We can equate the right-hand sides of Eqs. (101) and (103) to get rid of the explicit value of $T_{\text{in}}^{[P]}$ and obtain a single equation

$$F_s^{-1}\left(1 - \overline{s_P} f_{\text{out}}\right) = {F_g^{[P]}}^{-1}\left(\overline{s_P}(1 - f_{\text{out}})\right), \tag{104}$$

which we call the Signal Quality Equation. If this equation is solved for $\overline{s_P}$, inserting its solution into Eq. (77) with $s_P = 1 - \overline{s_P}$ determines the signal quality S_P. There is an

explicit solution if F_s^{-1} and F_g^{-1} are easy to calculate. An analytical solution to this equation, which requires several approximation steps, is outlined in S2 Appendix. Alternatively, the Signal Quality Equation can be solved numerically (see next subsection).

Notably, since the CDF of a discrete probability distribution is a piecewise constant function, the CDFs in Eqs. (102) and (102) are not generally invertible. A strict inverse $F^{-1}(y)$ such that $F(F^{-1}(y)) = y$ does not always exist because the same CDF value $F(x)$ can correspond to multiple arguments x. Being aware of this fact, we use the notation F_s^{-1} and $F_g^{[P]\,-1}$ here anyway. Later in this section, the binomial distributions (see Eqs. (61) and (63)) are approximated by normal distributions, and their approximated CDFs become strictly increasing and continuous, and thus invertible.

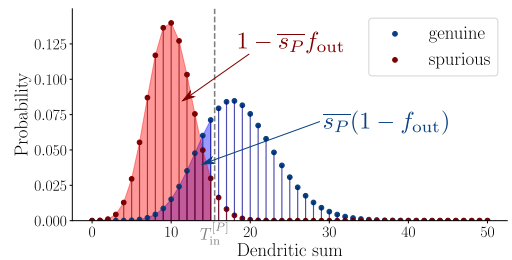

Fig 16. Signal Quality Equation. The activation threshold $T_{\text{in}}^{[P]}$ corresponds to the $1 - \overline{s_P} f_{\text{out}}$-quantile of the spurious distribution and to the $\overline{s_P}(1 - f_{\text{out}})$-quantile of the genuine distribution.

Numerical solution of the Signal Quality Equation The Signal Quality Equation (104) could be solved numerically. As discussed above, the CDFs F_s and $F_g^{[P]}$ are not invertible. Thus, we approximate the discrete distributions $p_s = \mathcal{B}_{M_{\text{in}},c}$ and $p_g = \sum_{u=0}^{P} \left[\mathcal{B}_{P,\, f_{\text{out}}}(u) \cdot \mathcal{B}_{M_{\text{in}},\, \rho_g(u)}(x)\right]$ by continuous distributions

$$p_s(x) \approx \mathcal{N}\left(M_{\text{in}}c, M_{\text{in}}c(1-c)\right)(x) \tag{105}$$

$$p_g^{[P]}(x) \approx \sum_{u=0}^{P} \left[\mathcal{B}_{P,\, f_{\text{out}}}(u) \cdot \mathcal{N}\left(M_{\text{in}}\rho_g(u), M_{\text{in}}\rho_g(u)(1-\rho_g(u))\right)(x)\right]. \tag{106}$$

This is a good approximation for large enough M_{in} and sparse but not too sparse connectivity c. Specifically, we have to assume

$$M_{\text{in}}c \gg 1,\ M_{\text{in}}(1-c) \gg 1 \text{ and} \tag{107}$$

$$M_{\text{in}}\rho_g(u) \gg 1,\ M_{\text{in}}(1-\rho_g(u)) \gg 1, \tag{108}$$

for all $u \in \{0, \ldots, P\}$. These four assumptions reduce to the two assumptions

$$M_{\text{in}}c \gg 1 \text{ and } M_{\text{in}}(1-\rho_g(u) \gg 1 \tag{109}$$

because $\rho_g(u) \geq c$. Then, the Signal Quality Equation (104) can be written as

$$\frac{F_g^{[P]}\left(F_s^{-1}\left(1 - \overline{s_P} f_{\text{out}}\right)\right)}{1 - f_{\text{out}}} = \overline{s_P} \tag{110}$$

and easily solved numerically without any additional approximation steps. This semi-analytical strategy yields a better approximation of the results from network simulations (see Fig 17) than the signal quality based on a fully analytical description (see S2 Appendix).

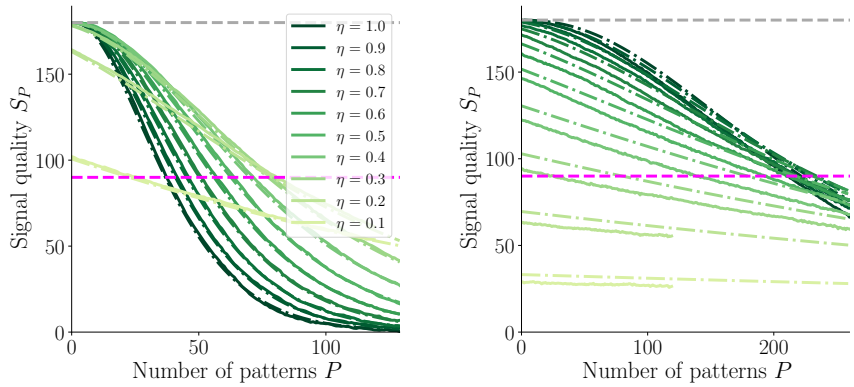

Fig 17. Comparison of signal quality obtained from network simulations and from a semi-analytical approximation. Signal quality decays with number P of subsequently learned patterns P for various values of transition probability η. Solid lines: numerical network simulations dash-dotted lines: semi-analytical estimate (as discussed in this subsection) of the signal quality; dashed grey line: maximal signal quality, which is the average Hamming distance between two random f_{out}-sparse patterns; dashed blue line: retrieval threshold. Left: $f_{\text{in}} = 0.1$, right: $f_{\text{in}} = 0.012$. Other parameters: $N_{\text{in}} = N_{\text{out}} = 1000, f_{\text{out}} = 0.1, c_m = 1, c = 0.2$.

Memory capacity of the network

Instead of calculating the signal quality at any P subsequent patterns (see previous section), which can then be used for finding the P for which the signal quality reaches the retrieval threshold, the network capacity can be determined by directly equating the signal quality to the retrieval threshold (expressed in terms of s_P) and solving the resulting equation for P.

We are searching for the number of patterns P for which the signal quality as defined in Eq. (76) equals the retrieval threshold T_S, which can be expressed as a fraction t_S of H_{avg}:

$$S_P := H_{\text{avg}} - H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right) \overset{!}{=} t_S H_{\text{avg}} =: T_S, \tag{111}$$

with $t_S \in (0, 1)$. In analogy to $\overline{s_P} = 1 - s_P$, we define $\overline{t_S} := 1 - t_S$ for convenience. Instead of calculating $\overline{s_P}$ for a given P, we now use the same Signal Quality Equation (104) to find the number of patterns P for which $\overline{s_P} \equiv \overline{t_S}$ (which is equivalent to $s_P \equiv t_S$) and $\overline{t_S}$ is given. We thus want to solve the Capacity Equation

$$F_s^{-1}\left(1 - \overline{t_S} f_{\text{out}}\right) = F_g^{[P]^{-1}}\left(\overline{t_S}(1 - f_{\text{out}})\right) \tag{112}$$

for P, which is the only unknown in this equation for a fixed set of network parameters. Notably, the Signal Quality Equation (104) and the Capacity Equation (112) are of the same form but, in the first, the number of patterns P is given and the fraction $\overline{s_P}$, which determines the signal quality $S_P = (1 - \overline{s_P} H_{\text{avg}}$, is unknown while, in the second, the retrieval ratio t_S is given and the number of patterns P, which corresponds to the capacity, is unknown.

Approximation of $p_g^{[P]}$ by a single binomial distribution In the following, we will approximate the distributions of dendritic sums such that the inverses of their distribution functions F_s and $F_g^{[P]}$ can be calculated. First, the linear combination of

binomial PMFs in $p_g^{[P]}$ in Eq. (63) is approximated by a single binomial PMF $\bar{p}_g^{[P]}$ (note the overline of the symbol p in the approximation).

A simple way to do so is by choosing the binomial distribution $\mathcal{B}_{M_{\text{in}},\,\rho_g(u)}$ that provides the largest contribution because it has the largest weight. The largest weight is $\binom{P}{u_{\text{mod}}} f_{\text{out}}^{u_{\text{mod}}}(1-f_{\text{out}})^{P-u_{\text{mod}}}$ with $u_{\text{mod}} = \lfloor f_{\text{out}}(P+1)\rfloor$. We hence obtain the approximation

$$p_g^{[P]}(x) = \sum_{k=0}^{P} \binom{P}{k} {f_{\text{out}}}^k (1-f_{\text{out}})^{P-k} \binom{M_{\text{in}}}{x} \rho_g(k)^x (1-\rho_g(k))^{M_{\text{in}}-x} \tag{113}$$

$$\approx \binom{M_{\text{in}}}{x} \rho_g(u_{\text{mod}})^x (1-\rho_g(u_{\text{mod}}))^{M_{\text{in}}-x} =: \bar{p}_g^{[P]}(x). \tag{114}$$

We are ultimately interested in a good approximation of the cumulative distribution function (see Capacity Equation (112) and Fig 16). We denote the cumulative distribution function of the approximation (114) as $\bar{F}_g^{[P]}$ (note, again, the overline). Fig 18 shows a numerical comparison of the summed probabilities

$$\sum_{x=0}^{X} \mathcal{P}(d_g^{[P]} = x) \tag{115}$$

up to various values of X for the original distribution $p_g^{[P]}$ and the approximation $\bar{p}_g^{[P]}$ (Eq. (114)) in terms of the absolute difference between the two

$$\left| F_g^{[P]}(X) - \bar{F}_g^{[P]}(X) \right| \tag{116}$$

and of the relative error

$$\frac{\left| F_g^{[P]}(X) - \bar{F}_g^{[P]}(X) \right|}{F_g^{[P]}(X)} \tag{117}$$

as functions of P for several values of X between $M_{\text{in}}c$ and $M_{\text{in}}c_m$. The non-continuities in the errors as functions of P occur due to the discrete jumps in $u_{\text{mod}} = \lfloor f_{\text{out}}(P+1)\rfloor$. The approximation by an individual binomial distribution is better for smaller transition probabilities η (compare first to second and third to fourth row in Fig 18). As discussed in S1 Appendix, a large f_{in} (above all if combined with a transition probability η that is close to one) leads to a more pronounced multimodality of the distribution for small P values which fades out more quickly with increasing P due to the steeper slope of $\rho_g(u)$ for small values of u (compare Fig 18 upper left to upper right). The more pronounced the multimodality, the less accurate the approximation by a single binomial distribution can be. The output activation ratio f_{out} has a different effect on the quality of the approximation than the input parameters. Increasing f_{out} increases u_{mod}. A large u_{mod} allows for a low multimodality already for small values of P because the strongest weights are quickly in a range where $\rho_g(k)$ is not extremely steep anymore but approaching the asymptotic value c (compare Fig 18 upper left and upper center). We would like to emphasize that decreasing the morphological connectivity c_m strongly improves the match of $F_g^{[P]}$ and $\bar{F}_g^{[P]}$ (compare Fig 18 upper left to lower right). Most figures in this manuscript show results for $c_m = 1$. In terms of the approximation of the distributions of dendritic sums by a binomial distribution, this is the worst-case scenario. In biological nervous systems, there is typically no all-to-all morphological connectivity observed. The analytical approximation derived in this section hence improves significantly for more realistic values of $c_m \ll 1$.

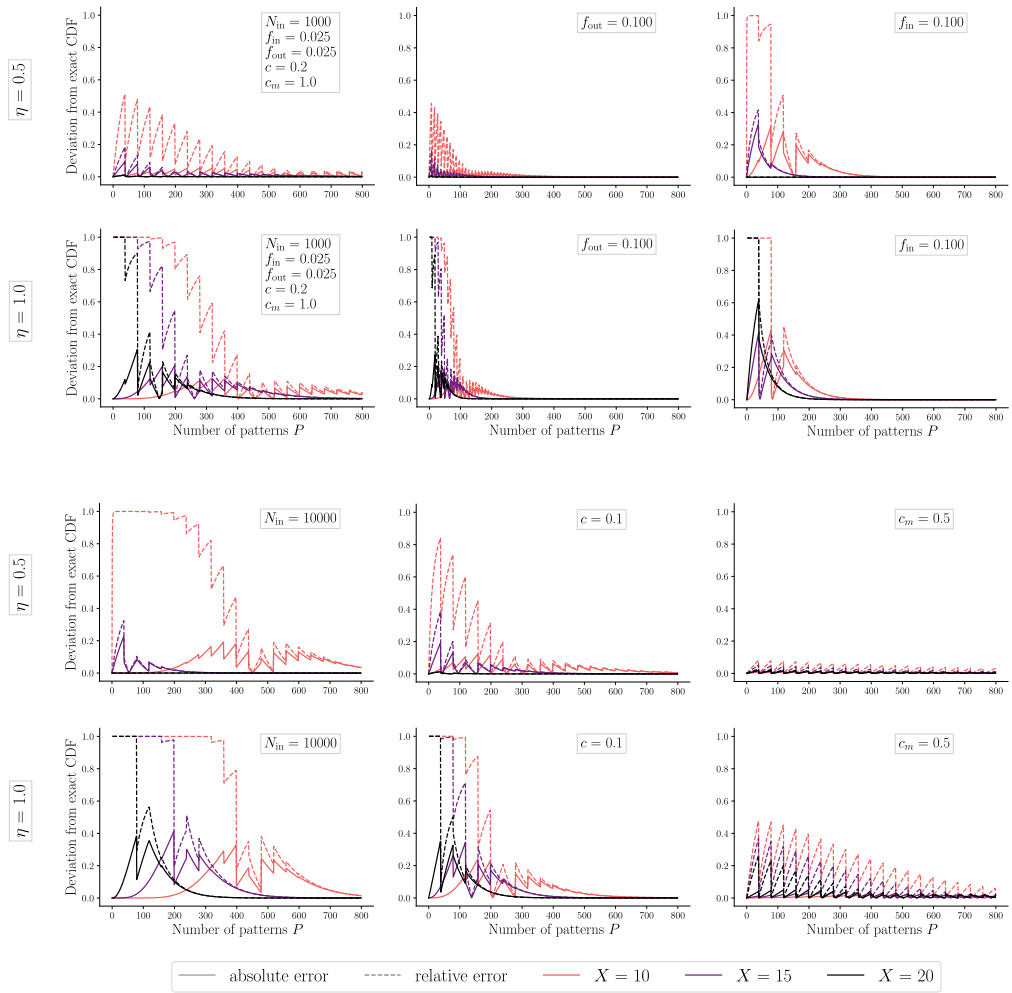

Fig 18. Error of approximation of $p_g^{[P]}$ by a single binomial distribution. Absolute (solid lines, Eq. (116)) and relative (dashed lines, Eq. (117)) error of the approximation of $F_g^{[P]}$ by the cumulative distribution function $\bar{F}_g^{[P]}$ of a binomial distribution (Eq. (114)) for various parameter combinations. For three different values of X (argument of the CDF) between cM_{in} and $c_m M_{\text{in}}$. In general, the errors decrease with an increase of X and with an increase of the number of patterns P. Default parameters: $N_{\text{in}} = 1000, f_{\text{in}} = f_{\text{out}} = 0.025, c = 0.2, c_m = 1$ (upper left). First and third row: $\eta = 0.5$, second and fourth row: $\eta = 1$. Upper center: $f_{\text{out}} = 0.1$, upper right: $f_{\text{in}} = 0.1$, lower left: $N_{\text{in}} = 10^4$, lower center: $c = 0.1$, lower right: $c_m = 0.5$.

Approximation of binomial distributions by normal distributions In what follows, we approximate the binomial distributions p_s (Eq. (61)) and $\bar{p}_g^{[P]}(x)$ (Eq. (114)) by normal distributions to be able to solve the Capacity Equation (112) analytically. This assumption allows us to derive explicit expressions, which are excellent approximations for the regime of sparse input patterns but large input layer sizes and not too sparse connectivity. Similar to the assumptions for the numerical solution of the Signal Quality Equation, we assume

$$M_{\text{in}} c \gg 1,\ M_{\text{in}}(1-c) \gg 1 \text{ and} \tag{118}$$

$$M_{\text{in}} \rho_g(\lfloor f_{\text{out}}(P+1) \rfloor) \gg 1,\ M_{\text{in}}(1-\rho_g(\lfloor f_{\text{out}}(P+1) \rfloor)) \gg 1, \tag{119}$$

for all P, which can be reduced to the two assumptions

$$M_{\text{in}}c \gg 1 \text{ and } M_{\text{in}}(1-\rho_g(\lfloor f_{\text{out}}(P+1)\rfloor)) \gg 1 \tag{120}$$

because $\rho_g(\lfloor f_{\text{out}}(P+1)\rfloor) \geq c$. (Note that $M_{\text{in}}(1-\rho_g(\lfloor f_{\text{out}}(P+1)\rfloor)) \gg 1$ is not true for $\eta \approx 1, c_m \approx 1$ and small P.)

We approximate the binomial PMFs $p_s = \mathcal{B}(M_{\text{in}}, c)$ and $\bar{p}_g^{[P]}$, as derived in the previous subsection, by normal probability densities

$$\tilde{p}_s = \mathcal{N}\left(\mu_s, \sigma_s^2\right) = \mathcal{N}\left(M_{\text{in}}c, M_{\text{in}}c(1-c)\right), \tag{121}$$

$$\begin{aligned}\tilde{p}_g^{[P]} &= \mathcal{N}\left(\bar{\mu}_g^{[P]}, \bar{\sigma}_g^{[P]2}\right) \\ &= \mathcal{N}\left(M_{\text{in}}\rho_g\left(\lfloor f_{\text{out}}(P+1)\rfloor\right), M_{\text{in}}\rho_g\left(\lfloor f_{\text{out}}(P+1)\rfloor\right)\left(1-\rho_g\left(\lfloor f_{\text{out}}(P+1)\rfloor\right)\right)\right),\end{aligned} \tag{122}$$

respectively, with cumulative distribution functions

$$\tilde{F}_s(x) = \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{x-\mu_s}{\sigma_s\sqrt{2}}\right)\right), \tag{123}$$

$$\tilde{F}_g^{[P]}(x) = \frac{1}{2}\left(1+\operatorname{erf}\left(\frac{x-\bar{\mu}_g^{[P]}}{\bar{\sigma}_g^{[P]}\sqrt{2}}\right)\right), \tag{124}$$

respectively. Note that, in the following, we will often ignore the floor function in the probability density $\tilde{p}_g^{[P]}$ and use $\rho_g(f_{\text{out}}P)$ instead of $\rho_g(\lfloor f_{\text{out}}(P+1)\rfloor)$ because this simplifies some calculations and gives rise to a smooth dependence of the signal quality on P. This implies that the capacity can take any non-negative real value and not only integer values that can be interpreted as a number of patterns.

The quantile functions in the Capacity Equation (112) are hence approximated by

$$F_s^{-1}(x) \approx \mu_s + \sigma_s\sqrt{2}\operatorname{erf}^{-1}(2x-1), \tag{125}$$

$$F_g^{[P]^{-1}}(x) \approx \bar{\mu}_g^{[P]} + \bar{\sigma}_g^{[P]}\sqrt{2}\operatorname{erf}^{-1}(2x-1) \tag{126}$$

and we obtain

$$F_s^{-1}\left(1-\overline{t_S}f_{\text{out}}\right) = F_g^{[P]^{-1}}\left(\overline{t_S}(1-f_{\text{out}})\right) \tag{127}$$

$$\Leftrightarrow \quad \mu_s + \sigma_s\sqrt{2}\operatorname{erf}^{-1}\left(2(1-\overline{t_S}f_{\text{out}})-1\right) = \bar{\mu}_g^{[P]} + \bar{\sigma}_g^{[P]}\sqrt{2}\operatorname{erf}^{-1}\left(2(\overline{t_S}(1-f_{\text{out}})-1\right) \tag{128}$$

$$\Leftrightarrow \quad \frac{\bar{\mu}_g^{[P]} - \mu_s}{\sqrt{2}\left(\sigma_s \cdot \operatorname{erf}^{-1}\left(1-2\overline{t_S}f_{\text{out}}\right) - \bar{\sigma}_g^{[P]} \cdot \operatorname{erf}^{-1}\left(2\overline{t_S}(1-f_{\text{out}})-1\right)\right)} = 1. \tag{129}$$

We define the function

$$R(x,y) := \operatorname{erf}^{-1}(1-2xy) \tag{130}$$

and introduce the abbreviations

$$R_s := R(\overline{t_S}, f_{\text{out}}) = \operatorname{erf}^{-1}\left(1-2\overline{t_S}f_{\text{out}}\right) \tag{131}$$

and

$$R_g := R(\overline{t_S}, 1-f_{\text{out}}) = -\operatorname{erf}^{-1}\left(2\overline{t_S}(1-f_{\text{out}})-1\right) \tag{132}$$

for the weights of the standard deviations in Eq. (129) (Fig 19). Then we have

$$\frac{\bar{\mu}_g^{[P]} - \mu_s}{\sqrt{2}\left(\sigma_s \cdot R_s + \bar{\sigma}_g^{[P]} \cdot R_g\right)} = 1. \tag{133}$$

Note that, for fixed $\overline{t_S}$, the weights' dependence on f_{out} is symmetric:

$$R_s\Big|_{f_{\text{out}}=f} = R_g\Big|_{f_{\text{out}}=1-f} \tag{134}$$

For $f_{\text{out}} < 0.5$, the prefactor of the spurious standard deviation, R_s, is larger than the prefactor of the genuine distribution, R_g. For $\overline{t_S} \leq 0.5$, both R_g and R_s are non-negative for all f_{out}.

The left hand side of Eq. (133) can be interpreted as a generalized version of the sensitivity index

$$d' = \frac{|\mu_1 - \mu_2|}{\frac{1}{2}(\sigma_1 + \sigma_2)} \tag{135}$$

which can be used to quantify the discriminability of two distributions with means μ_1 and μ_2 and standard deviations σ_1 and σ_2. In our case, the additional weighting of the standard deviations by R_g and R_s takes into account the dissimilarly weighted contributions of the false positive and false negative activations due to the unequal amount of genuine units M_{out} and spurious units $N_{\text{out}} - M_{\text{out}}$ reflected in Eq. (92).

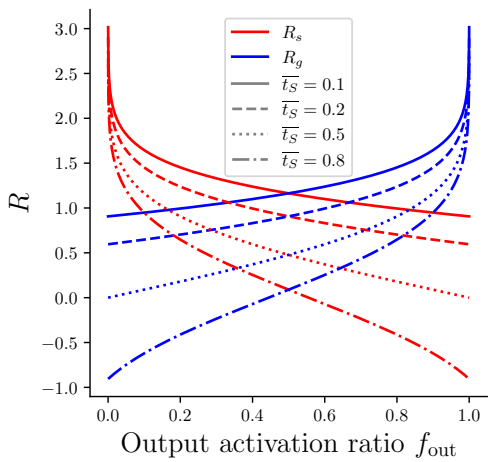

Fig 19. Weights of standard deviations. R_g and R_s as functions of f_{out} for $\overline{t_S} = 0.1, 0.2, 0.5$ and 0.8. R_g is R_s mirrored at $f_{\text{out}} = 0.5$. While R_s is decreasing as a function of f_{out}, R_g is hence increasing as a function of f_{out}. Both decrease with increasing $\overline{t_S}$. For $\overline{t_S} \leq 0.5$, R_g and R_s are non-negative. For $f_{\text{out}} \leq 0.5$, we have $R_s \geq 0$ and, for $f_{\text{out}} \geq 0.5$, we have $R_g \geq 0$.

Solving the Capacity Equation With the approximations discussed in the previous subsection,the Capacity Equation (112) is equivalent to Eq. (133). Now, we solve Eq. (133) for P with the means and standard deviations from the approximate distributions in Eq. (121) and (123):

$$\mu_s = M_{\text{in}} c \tag{136}$$

$$\sigma_s = \sqrt{M_{\text{in}} c(1-c)} \tag{137}$$

$$\bar{\mu}_g^{[P]} = M_{\text{in}} \left[(c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} + c \right] \tag{138}$$

$$\bar{\sigma}_g^{[P]} = \sqrt{\begin{aligned} & M_{\text{in}} \left[(c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} + c \right] \\ & \cdot \left[1 - (c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} - c \right] \end{aligned}}. \tag{139}$$

We obtain

$$\bar{\mu}_g^{[P]} - \mu_s = M_{\text{in}}(c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} \tag{140}$$

and

$$\sigma_s R_s + \bar{\sigma}_g^{[P]} R_g = \tag{141}$$
$$\sqrt{M_{\text{in}}}\left(\sqrt{c(1-c)}R_s + \sqrt{\begin{array}{l}\left[(c_m - c)\eta\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} + c\right] \\ \cdot \left[1 - (c_m - c)\eta\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} - c\right]\end{array}} R_g\right).$$

Using the abbreviation $\alpha := (c_m - c)\eta\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} > 0$, Eq.(133) becomes

$$\frac{\sqrt{M_{\text{in}}}\alpha}{\sqrt{2}\left(\sqrt{c(1-c)}R_s + \sqrt{[\alpha + c]\,[1 - \alpha - c]}R_g\right)} = 1 \tag{142}$$
$$\Leftrightarrow \sqrt{M_{\text{in}}}\alpha - \sqrt{c(1-c)}\sqrt{2}R_s = \sqrt{[\alpha + c]\,[1 - \alpha - c]}\sqrt{2}R_g \tag{143}$$
$$\Leftrightarrow M_{\text{in}}\alpha^2 - 2\sqrt{2M_{\text{in}}c(1-c)}R_s\alpha + 2c(1-c)R_s^2 = 2\,[\alpha + c]\,[1 - \alpha - c]\,R_g^2 \tag{144}$$
$$\Leftrightarrow \alpha^2\left[M_{\text{in}} + 2R_g^2\right] + \alpha\left[2(2c-1)R_g^2 - 2\sqrt{2M_{\text{in}}c(1-c)}R_s\right] + 2c(1-c)(R_s^2 - R_g^2) = 0. \tag{145}$$

Solving the quadratic equation gives

$$\alpha = \frac{2(1-2c)R_g^2 + 2\sqrt{2M_{\text{in}}c(1-c)}R_s + \sqrt{\begin{array}{l}\left[2(1-2c)R_g^2 + 2\sqrt{2M_{\text{in}}c(1-c)}R_s\right]^2 \\ +8c(1-c)(M_{\text{in}} + 2R_g^2)(R_g^2 - R_s^2)\end{array}}}{2(M_{\text{in}} + 2R_g^2)}. \tag{146}$$

For convenience, we introduce the abbreviations

$$A := (1-2c)R_g^2 + \sqrt{2M_{\text{in}}c(1-c)}R_s + |R_g| \cdot \sqrt{R_g^2 + 2c(1-c)(M_{\text{in}} - 2R_s^2) + 2(1-2c)\sqrt{2M_{\text{in}}c(1-c)}R_s}, \tag{147}$$
$$B := (c_m - c)(M_{\text{in}} + 2R_g^2) \tag{148}$$

and have

$$\alpha = \frac{(c_m - c)A}{B}. \tag{149}$$

The leading coefficient of the quadratic equation (145) in α is positive. For increasing P, the term $\alpha := (c_m - c)\eta\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor}$ is decreasing. This means that, by increasing P, we are approaching zero from the right and we are hence interested in the

larger solution of the equation. Therefore, we have to choose the positive sign in Eq. (146). Since $c_m > c, \eta > 0$ and $\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{\lfloor f_{\text{out}}(P+1)\rfloor} > 0$, α must be positive. Using the positive sign in front of the square root ensures a positive α. Note that the expression under the second square root could become negative if R_s becomes too large compared to R_g. For $M_{\text{in}} \gg 1$, this is only the case for very small f_{out} values combined with a small $\overline{t_S}$. These cases cannot be treated analytically in this way.

Next, we insert $\alpha := (c_m - c)\eta \left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right)^{f_{\text{out}}(P+1)}$ and approximate $\lfloor f_{\text{out}}(P+1)\rfloor \approx f_{\text{out}}P$ to obtain

$$f_{\text{out}}P \ln\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right) = \ln\left(\frac{A}{\eta B}\right) \tag{150}$$

$$\Leftrightarrow \quad P = \frac{\ln\left(\frac{A}{\eta B}\right)}{\ln\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right) f_{\text{out}}} \tag{151}$$

Since P is a number of patterns, we round the obtained expression to the nearest integer and obtain the capacity of the network

$$P^* = \left\lfloor \frac{\ln\left(\frac{A}{\eta B}\right)}{\ln\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right) f_{\text{out}}} + \frac{1}{2} \right\rfloor . \tag{152}$$

If R_g and R_s are known, this expression allows us to directly calculate the capacity of the network.

Approximation of the error function At this point, the constants R_s and R_g (see Eqs. (131) and (132)) could be approximated with the help of any invertible approximation of the error function. We choose the highly accurate global Padé approximation

$$\text{erf}(x) \approx \text{sign}(x)\sqrt{1 - \exp\left(-x^2 \frac{\frac{4}{\pi} + ax^2}{1 + ax^2}\right)}, \tag{153}$$

with $a = \frac{8(\pi-3)}{3\pi(4-\pi)}$ [70], see Fig 20, which allows for the approximation

$$\text{erf}^{-1}(x) \approx \text{sign}(x)\sqrt{\frac{-\frac{4}{\pi} - a\ln(1-x^2) + \sqrt{\left(\frac{4}{\pi} + a\ln(1-x^2)\right)^2 - 4a\ln(1-x^2)}}{2a}} \tag{154}$$

to use for R_s and R_g in Eq. (133). All analytical approximations of the network capacity shown in figures are calculated with this approximation unless stated otherwise.

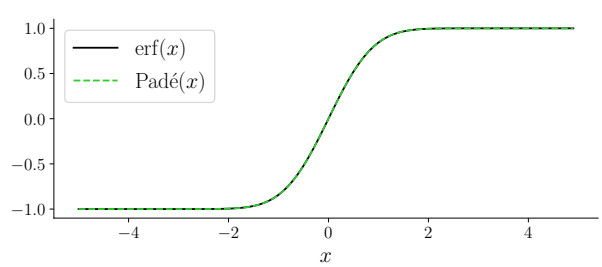

Fig 20. Approximation of the error function by a global Padé approximation. The maximal absolute error is 0.00033, the maximal relative error is 0.00035.

Optimal transition probability and maximal capacity

From Eq. (152), we can derive the optimal transition probability η_{opt} and therewith the maximal capacity P^*_{max} for a fixed set of network parameters.

We assume $\frac{f_{\text{in}}\eta c_m}{c} \ll 1$ and use the Taylor series of $\ln(1-x)$ at $x=0$

$$\ln(1-x) = -x + \mathcal{O}(x^2) \tag{155}$$

to approximate

$$\ln\left(1 - \frac{f_{\text{in}}\eta c_m}{c}\right) \approx -\frac{f_{\text{in}}\eta c_m}{c}. \tag{156}$$

The capacity as a function of η then reads as

$$P^*(\eta) \approx \left\lfloor \frac{\ln(A) - \ln(B) - \ln(\eta)}{-\eta C} + \frac{1}{2} \right\rfloor, \tag{157}$$

with A and B as defined in Eqs. (147) and (148) and

$$C := \frac{M_{\text{in}} c_m f_{\text{out}}}{N_{\text{in}} c}. \tag{158}$$

In the following, we will ignore the rounding of P^* to an integer value in (157) and allow the capacity to take any (non-negative) real value. We then have

$$P^*(\eta) \approx \frac{\ln(A) - \ln(B) - \ln(\eta)}{-\eta C} \tag{159}$$

as also stated in the Results in Eq. (19). Next, we calculate the optimal transition probability η_{opt} by differentiating P^* with respect to η

$$P^{*\prime}(\eta) = \frac{-\ln(\eta) + \ln(A) - \ln(B) + 1}{\eta^2 C} \tag{160}$$

and solving

$$P^{*\prime}(\eta_0) = 0 \tag{161}$$

$$\Leftrightarrow \frac{-\ln(\eta_0) + \ln(A) - \ln(B) + 1}{\eta_0^2} = 0 \tag{162}$$

$$\Leftrightarrow \eta_0 = \exp\left(1 + \ln(A) - \ln(B)\right) = \frac{\mathrm{e}A}{B}. \tag{163}$$

This is a global maximum of $P^*(\eta)$ because

$$P^{*\prime\prime}(\eta_0) = \frac{2\ln(\frac{\mathrm{e}A}{B}) - 2\ln(A) + 2\ln(B) - 3}{\left(\frac{\mathrm{e}A}{B}\right)^3 C} = \frac{-1}{\left(\frac{\mathrm{e}A}{B}\right)^3 C} < 0. \tag{164}$$

The optimal transition probability must be bounded to $(0,1]$. We can assume that $\frac{\mathrm{e}A}{B} > 0$ (otherwise there is no real solution for (152)). If $\frac{\mathrm{e}A}{B} \leq 1$, we have $\eta_{\text{opt}} = \frac{\mathrm{e}A}{B}$. If $\frac{\mathrm{e}A}{B} > 1$, the capacity as a function of the transition probability $P^*(\eta)$ is monotonically increasing in the range $\eta \in (0,1]$ that we are interested in because we have shown that P^* has a global maximum at $\mathrm{e}A/B > 1$. We thus have

$$\eta_{\text{opt}} = \min\left(\frac{\mathrm{e}A}{B}, 1\right), \tag{165}$$

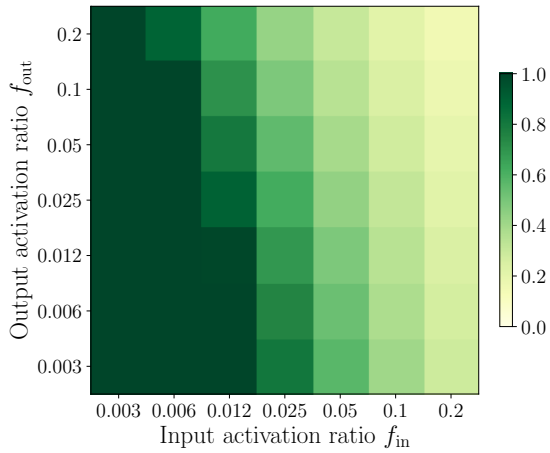

Fig 21. Analytical optimal transition probability. Optimal transition probability η_{opt} (Eq. (165)) as a function of f_{in} and f_{out}. η_{opt} decreases both as a function of f_{in} and as a function of f_{out} but the slope is larger for f_{in}. In Fig 3E in the Results, the corresponding optimal transition probability η_{opt} obtained from numerical simulations is shown. (Other parameters: $N_{\text{in}} = 1000, c = 0.2, c_m = 1, t_S = 0.5$.)

as also displayed in Eq. (20) in the Results. Fig 21 shows η_{opt} as a function of the input and output activation ratios f_{in} and f_{out}.

The maximal capacity that can be obtained for a given set of network parameters by optimizing the transition probability is then defined as

$$P^*_{\text{max}} := P^*(\eta_{\text{opt}}) \approx \frac{\ln(A) - \ln(B) - \ln(\eta_{\text{opt}})}{-\eta_{\text{opt}} C}. \tag{166}$$

Comparisons of the analytical approximation of the optimal transition probability η_{opt} (Eq. (165)) and the maximal capacity P^*_{max} (Eq. (166)) with results obtained from numerical network simulations are presented in Fig 3B,C and Fig 5.

Dependence on input parameters Since only $C = M_{\text{in}} c_m f_{\text{out}}/(N_{\text{in}} c)$ depends on N_{in} while A and B do not (see Eqs. (147)–(158)), the capacity is proportional to the input layer size N_{in} (Fig 5B). Further, we note that the optimal transition probability η_{opt} (see Eq. (165)) does not depend on the input layer size N_{in}, but it does depend on the number of active input units $M_{\text{in}} = f_{\text{in}} N_{\text{in}}$ (Fig 5A).

We further observe that the maximal capacity as a function of the number of active input units M_{in} has a maximum that is independent of N_{in}. In the range where $\eta_{\text{opt}} = \mathrm{e}A/B$, the maximal capacity as a function of M_{in}

$$P^*_{\text{max}}(M_{\text{in}}) = \frac{\ln(A(M_{\text{in}})) - \ln(B(M_{\text{in}})) - \ln\left(\frac{\mathrm{e}A(M_{\text{in}})}{B(M_{\text{in}})}\right)}{-\frac{\mathrm{e}A(M_{\text{in}})}{B(M_{\text{in}})} C(M_{\text{in}})} = \frac{B(M_{\text{in}})}{\mathrm{e}A(M_{\text{in}}) C(M_{\text{in}})} \tag{167}$$

is strictly monotonically decreasing: For $M_1 < M_2$, we have

$$P^*_{\text{max}}(M_1) > P^*_{\text{max}}(M_2) \Leftrightarrow \frac{B(M_1)}{\mathrm{e}A(M_1)C(M_1)} > \frac{B(M_2)}{\mathrm{e}A(M_2)C(M_2)} \tag{168}$$

$$\Leftarrow \frac{B(M_1)}{A(M_1)C(M_1)} > \frac{B(M_2)}{A(M_1)C(M_2)} \Leftrightarrow \frac{B(M_1)}{C(M_1)} > \frac{B(M_2)}{C(M_2)} \tag{169}$$

$$\Leftrightarrow \frac{M_1 + 2R_g^2}{M_1} > \frac{M_2 + 2R_g^2}{M_2} \Leftrightarrow M_2 > M_1, \tag{170}$$

where we used that $A(M_1) < A(M_2)$ because A is monotonically increasing as a function of M_{in} (due to $R_s > 0$ for $f_{\text{out}} < 0.5$) and that A has to be positive in order to yield a real valued number for the capacity in Eq. (152). Thus, if there is a maximum of

$P^*_{\max}(M_{\text{in}})$, it must occur within the range of M_{in} values that have an optimal transition probability of 1 (see Fig 22A-C). In order to find this maximum, we set $\eta = 1$ in Eq. (159), and then we set the derivative (with respect to M_{in}) of the maximal capacity

$$P^*_{\max}(M_{\text{in}}) = \frac{\ln(A(M_{\text{in}})) - \ln(B(M_{\text{in}}))}{-C(M_{\text{in}})} \tag{171}$$

to zero:

$$P^*_{\max}{}'(M_{\text{in}}) \stackrel{!}{=} 0. \tag{172}$$

This equation can be solved numerically for fixed values of c, c_m, f_{out}, and $\overline{t_S}$. For example, for $c = 0.2, c_m = 1, f_{\text{out}} = 0.006$ and $\overline{t_S} = 0.1$, we obtain an optimal number of active input units $M_{\text{in}} = 13$, which yields a maximal capacity $P_{\max}(13) \approx 1.11172 \cdot N_{\text{in}}$ (see Fig 22C).

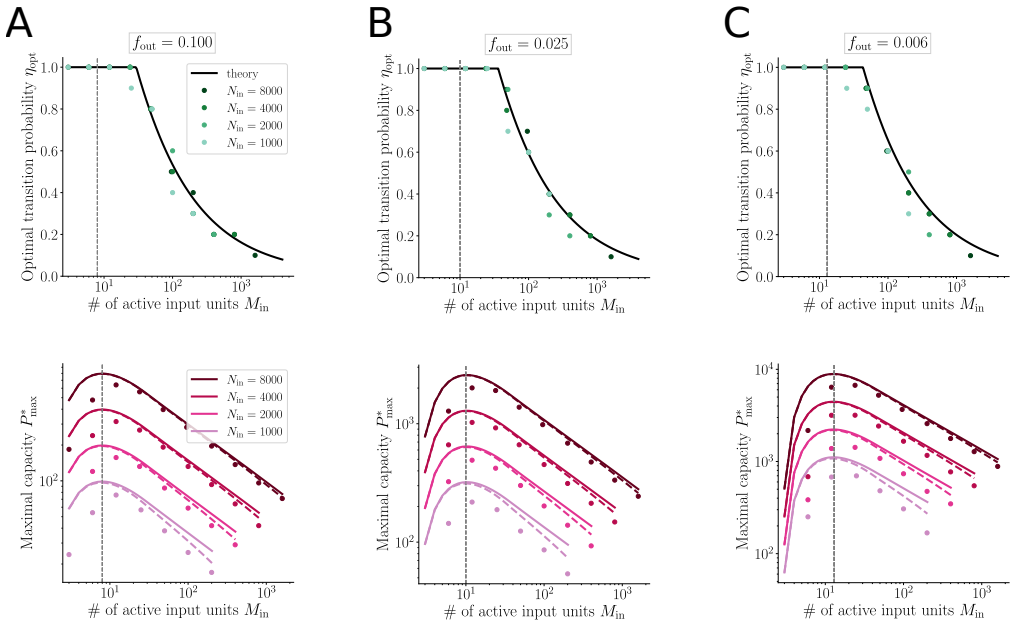

Fig 22. Optimal transition probability and maximal capacity for various N_{in}. Dependence of optimal transition probability (top) and maximal capacity (bottom) on number of active input units ($M_{\text{in}} = f_{\text{in}} N_{\text{in}}$) for various input layer sizes ($N_{\text{in}} \in \{1000, 2000, 4000, 8000\}$). Dots: numerical simulations; solid lines: optimal η obtained from Eq. (165) and maximal capacity calculated with Eq. (159); dashed lines: maximal capacity calculated with Eq. (152), i.e. without the Taylor approximation (156). In (**A**), $f_{\text{out}} = 0.1$, in (**B**), $f_{\text{out}} = 0.025$ and in (**C**), $f_{\text{out}} = 0.006$. Other parameters: $N_{\text{out}} = 1000, c = 0.2, c_m = 1, \overline{t_S} = 0.1$.

The fit between the theoretical approximation and the actual capacity obtained in numerical simulations improves for increasing input layer size N_{in}, increasing input activation ratio f_{in}, and increasing output activation ratio f_{out} (see comparison in Table 1). As discussed before, we need $M_{\text{in}} \cdot c \gg 1$ and $M_{\text{in}} \cdot (1 - \rho_g(u_{\text{mod}})) \gg 1$ in order to approximate the distributions of the dendritic sums by Gaussian distributions. A large f_{out} reduces the multimodality of the genuine distribution and hence also allows for a better approximation.

	$f_{\text{out}} = 0.1$	$f_{\text{out}} = 0.025$	$f_{\text{out}} = 0.006$
$N_{\text{in}} = 1000$	1.21 (1.36)	1.34 (1.53)	1.44 (1.67)
$N_{\text{in}} = 2000$	1.13 (1.21)	1.22 (1.32)	1.33 (1.45)
$N_{\text{in}} = 4000$	1.11 (1.16)	1.16 (1.22)	1.22 (1.28)
$N_{\text{in}} = 8000$	1.06 (1.09)	1.11 (1.14)	1.24 (1.18)

Table 1. Deviation between theory and simulations: Maximal capacity obtained from theory divided by maximal capacity obtained from simulations for various values of N_{in} and f_{out} averaged across all M_{in} values shown in Fig 22 except the two smallest $M_{\text{in}} = 3$ and $M_{\text{in}} = 6$ (which does not fulfill the condition $M_{\text{in}} \gg 1$ and for which the fit is thus not accurate enough). Deviation calculated based on Eq. (152) (corresponding to dashed lines in Fig 22). In brackets: Deviation calculated based on (159), i.e. with the Taylor approximation (156) (corresponding to solid lines in Fig 22).

Extension to noisy input patterns during retrieval

In this section, we extend the derivation of the memory capacity and therewith the derivation of the maximal memory capacity and the optimal transition probability to the case of noisy input patterns during retrieval. We assume a noise level $\varepsilon \in (0, 1)$. See earlier sections of the Methods for details on the distributions of dendritic sums with noisy cues; some of the expressions stated there are repeated in what follows in order to simplify the argument.

Mean and standard deviation First, we approximate the distribution of dendritic sums of genuine output units

$$p_{g,\varepsilon}^{[P]}(x) = \sum_{u=0}^{P} \left[\binom{P}{u} {f_{\text{out}}}^{u} (1 - f_{\text{out}})^{P-u} \right. \tag{173}$$
$$\left. \cdot \sum_{x_g=0}^{x} \left(\binom{m_g}{x_g} \rho_g(u)^{x_g} (1 - \rho_g(u))^{m_g - x_g} \binom{m_s}{x_s} \rho_n(u)^{x_s} (1 - \rho_n(u))^{m_s - x_s} \right) \right]$$

by the renormalized maximum summand ($u = u_{\text{mod}}$) and have

$$p_{g,\varepsilon}^{[P]}(x) \approx \bar{p}_{g,\varepsilon}^{[P]}(x) = \sum_{x_g=0}^{x} \left(\binom{m_g}{x_g} \rho_g(u_{\text{mod}})^{x_g} (1 - \rho_g(u_{\text{mod}}))^{m_g - x_g} \right.$$
$$\left. \cdot \binom{m_s}{x_s} \rho_n(u_{\text{mod}})^{x_s} (1 - \rho_n(u_{\text{mod}}))^{m_s - x_s} \right) \tag{174}$$

with $u_{\text{mod}} = \lfloor (P + 1) f_{\text{out}} \rfloor$.

We use again the abbreviation $\alpha := (c_m - c)\eta \left(1 - \frac{f_{\text{in}} \eta c_m}{c}\right)^{u_{\text{mod}}}$. As suggested by [69], the mean and standard deviation of $\bar{p}_{g,\varepsilon}^{[P]}$ are approximated as

$$\bar{\mu}_{g,\varepsilon}^{[P]} \approx m_g \rho_g(u_{\text{mod}}) + m_s \rho_n(u_{\text{mod}}) = m_g (\alpha + c) + m_s \left(\frac{-f_{\text{in}}}{1 - f_{\text{in}}} \alpha + c \right) \tag{175}$$
$$= M_{\text{in}} \left[\alpha \left(1 - \varepsilon + \varepsilon \frac{-f_{\text{in}}}{1 - f_{\text{in}}} \right) + c \right] = M_{\text{in}} \left[\alpha \left(1 - \frac{\varepsilon}{1 - f_{\text{in}}} \right) + c \right] \tag{176}$$

and

$$\bar{\sigma}_{g,\varepsilon}^{[P]} \approx \sqrt{m_g \rho_g(u_{\text{mod}})(1-\rho_g(u_{\text{mod}})) + m_s \rho_n(u_{\text{mod}})(1-\rho_n(u_{\text{mod}}))} \tag{177}$$

$$= \sqrt{M_{\text{in}}\left[(1-2c)\alpha\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right) - \alpha^2\left(1-\frac{\varepsilon(1-2f_{\text{in}})}{(1-f_{\text{in}})^2}\right) + c(1-c)\right]}. \tag{178}$$

Memory capacity of the network As for the case without noise, we solve Eq. (133) for P to obtain the capacity of the network. With $\bar{\mu}_{g,\varepsilon}^{[P]}$ and $\bar{\sigma}_{g,\varepsilon}^{[P]}$, I have

$$\frac{M_{\text{in}}\left[\alpha\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)+c\right] - M_{\text{in}}c}{\sqrt{2}\left(R_g\sqrt{M_{\text{in}}\left[(1-2c)\alpha\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right) - \alpha^2\left(1-\frac{\varepsilon(1-2f_{\text{in}})}{(1-f_{\text{in}})^2}\right)+c(1-c)\right]} + R_s\sqrt{M_{\text{in}}c(1-c)}\right)} = 1 \tag{179}$$

$$\Leftrightarrow \alpha^2\left[M_{\text{in}}\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)^2 + 2R_g^2\left(1-\frac{\varepsilon(1-2f_{\text{in}})}{(1-f_{\text{in}})^2}\right)\right]$$
$$- \alpha 2\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)\left[\sqrt{2M_{\text{in}}c(1-c)}R_s + (1-2c)R_g^2\right]$$
$$+ 2c(1-c)(R_s^2 - R_g^2) = 0 \tag{180}$$

$$\Leftrightarrow \alpha = \frac{(c_m - c)A_\varepsilon}{B_\varepsilon}, \tag{181}$$

with the abbreviations

$$A_\varepsilon := \left((1-2c)R_g^2 + \sqrt{2M_{\text{in}}c(1-c)}R_s\right)\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right) + |R_g| \tag{182}$$
$$\cdot\sqrt{\left(2\sqrt{2M_{\text{in}}c(1-c)}(1-2c)R_s + (1-2c)^2R_g^2 + 2c(1-c)M_{\text{in}}\right)\cdot\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)^2 - 4c(1-c)(R_s^2-R_g^2)\left(1-\frac{\varepsilon(1-2f_{\text{in}})}{(1-f_{\text{in}})^2}\right)}$$

and

$$B_\varepsilon := (c_m - c)\left[M_{\text{in}}\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)^2 + 2R_g^2\left(1-\frac{\varepsilon(1-2f_{\text{in}})}{(1-f_{\text{in}})^2}\right)\right]. \tag{183}$$

These results generalize the expressions for A and B in Eqs. (147) and (148), which were derived for $\varepsilon = 0$. Expanding $\ln(1-x)$, we obtain the capacity

$$P_\varepsilon^* = \left\lfloor \frac{\ln\left(\frac{A_\varepsilon}{\eta B_\varepsilon}\right)}{\ln\left(1-\frac{f_{\text{in}}\eta c_m}{c}\right)f_{\text{out}}} + \frac{1}{2}\right\rfloor \approx \frac{c}{-f_{\text{in}}f_{\text{out}}\eta c_m}\ln\left(\frac{A_\varepsilon}{\eta B_\varepsilon}\right). \tag{184}$$

Optimal transition probability and maximal capacity The optimal transition probability η_{opt} and hence the maximal capacity P_{max}^* of the network can be calculated in the same way as without noise.The only difference lies in the expressions A_ε and B_ε

that were previously called A and B. Expression $C = M_{\text{in}} c_m f_{\text{out}}/(N_{\text{in}} c)$ remains the same as in Eq. (158). We have

$$\eta_{\text{opt},\varepsilon} \approx \min\left(\frac{\text{e}A_\varepsilon}{B_\varepsilon}, 1\right) \tag{185}$$

and

$$P^*_{\text{max},\varepsilon} \approx \frac{\ln(A_\varepsilon) - \ln(B_\varepsilon) - \ln(\eta_{\text{opt},\varepsilon})}{-\eta_{\text{opt},\varepsilon} C}, \tag{186}$$

as also stated in Eqs. (22) and (25) in the Results.In the range where $\eta_{\text{opt},\varepsilon} < 1$, we thus can express the maximal capacity as

$$P^*_{\text{max},\varepsilon} = \frac{B_\varepsilon}{\text{e}A_\varepsilon C}. \tag{187}$$

We can assume that $P^*_{\text{max}} > P^*_{\text{max},\varepsilon}$, where P^*_{max} is the maximal capacity for $\varepsilon = 0$ defined in Eq. (159), because the additional noise on the input patterns during retrieval naturally has a detrimental effect on the capacity. In the range of parameters where $\eta_{\text{opt}} < 1$ (see Eq. (167)) and $\eta_{\text{opt},\varepsilon} < 1$, we thus have

$$P^*_{\text{max}} > P^*_{\text{max},\varepsilon} \Leftrightarrow \frac{B}{\text{e}AC} > \frac{B_\varepsilon}{\text{e}A_\varepsilon C} \Leftrightarrow \frac{\text{e}A}{B} < \frac{\text{e}A_\varepsilon}{B_\varepsilon} \Leftrightarrow \eta_{\text{opt}} < \eta_{\text{opt},\varepsilon}, \tag{188}$$

since $C > 0$ and we can assume $A/B > 0$ and $A_\varepsilon/B_\varepsilon > 0$ in order to obtain a real-valued capacity. The optimal transition probability hence increases with added noise compared to the noise-less case by a factor that will be further approximated in the next section. A comparison of P^*_{max} and $P^*_{\text{max},\varepsilon}$ as well as of η_{opt} and $\eta_{\text{opt},\varepsilon}$ is shown in Fig 6D,E and in S4 Fig.

Approximation of $\bar{p}^{[P]}_{g,\varepsilon}$ by a single binomial distribution In order to find a relationship between the capacity P^*_ε with noise and the capacity P^* without noise that is easier to interpret, we can further approximate $\bar{p}^{[P]}_{g,\varepsilon}$ by a single binomial distribution $\bar{p}^{[P],\text{approx}}_{g,\varepsilon}$ with the same mean

$$\bar{\mu}^{[P]}_{g,\varepsilon} = M_{\text{in}}\left[\alpha\left(1 - \frac{\varepsilon}{1 - f_{\text{in}}}\right) + c\right] \tag{189}$$

as in Eq. (175). If $\bar{\mu}^{[P]}_{g,\varepsilon}$ is interpreted as the mean of a binomial distribution $\mathcal{B}_{M_{\text{in}},p}$, its success probability is $p = \alpha\left(1 - \frac{\varepsilon}{1-f_{\text{in}}}\right) + c$ and the corresponding effective standard deviation has to be

$$\bar{\sigma}^{[P],\text{approx}}_{g,\varepsilon} = \sqrt{M_{\text{in}}\left[\alpha\left(1 - \frac{\varepsilon}{1 - f_{\text{in}}}\right) + c\right] \cdot \left[1 - \alpha\left(1 - \frac{\varepsilon}{1 - f_{\text{in}}}\right) - c\right]}. \tag{190}$$

This approximation is exact if the two variances

$$m_g \rho_g(u_{\text{mod}})(1 - \rho_g(u_{\text{mod}})) \qquad \text{and} \tag{191}$$

$$m_s \rho_n(u_{\text{mod}})(1 - \rho_n(u_{\text{mod}})) \tag{192}$$

that contribute to $\bar{\sigma}^{[P]}_{g,\varepsilon}$ (Eq. (177)) are the same. Inserting $\bar{\mu}^{[P]}_{g,\varepsilon}$ and $\bar{\sigma}^{[P],\text{approx}}_{g,\varepsilon}$ in Eq. (133) and using the abbreviation $\tilde{\alpha} := \alpha \cdot \left(1 - \frac{\varepsilon}{1-f_{\text{in}}}\right)$ yields the equation

$$\frac{M_{\text{in}}\left(\tilde{\alpha} + c\right) - M_{\text{in}} c}{\sqrt{2}\left(R_g\sqrt{2\left[(1-2c)\tilde{\alpha} - \tilde{\alpha}^2 + c(1-c)\right]} + R_s\sqrt{c(1-c)}\right)} = 1. \tag{193}$$

This equation in $\tilde{\alpha}$ is identical with Eq. (142) in α and thus has the solution

$$\tilde{\alpha} = \frac{(c_m - c)A}{B}, \tag{194}$$

with A and B being exactly the same as without noisy input patterns (Eqs. (147) and (148)). From this we obtain the capacity

$$P_\varepsilon^{*\text{approx}} = \left\lfloor \frac{\ln\left(\frac{A}{\eta\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)B}\right)}{\ln\left(1-\frac{f_{\text{in}}\eta c_m}{c}\right)f_{\text{out}}} + \frac{1}{2} \right\rfloor \tag{195}$$

$$\approx \frac{c}{-f_{\text{in}}f_{\text{out}}\eta c_m}\left[\ln\left(\frac{A}{\eta B}\right) - \ln\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)\right] \tag{196}$$

$$= P^* + \frac{\ln\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)c}{f_{\text{in}}f_{\text{out}}\eta c_m}. \tag{197}$$

The optimal transition probability with noise during retrieval $\eta_{\text{opt},\varepsilon}$ can thus be approximated as a multiple of the optimal transition probability without noise η_{opt} by

$$\eta_{\text{opt},\varepsilon} \approx \eta_{\text{opt},\varepsilon}^{\text{approx}} = \min\left(\frac{\mathrm{e}A}{\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)B}, 1\right) = \frac{\eta_{\text{opt}}}{\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)} \tag{198}$$

and the maximal capacity $P_{\text{max},\varepsilon}^*$ by

$$P_{\text{max},\varepsilon}^* \approx P^*\big|_{\eta=\eta_{\text{opt},\varepsilon}} + \frac{\ln\left(1-\frac{\varepsilon}{1-f_{\text{in}}}\right)c}{f_{\text{in}}f_{\text{out}}\eta_{\text{opt},\varepsilon}c_m}. \tag{199}$$

For $\varepsilon \ll 1$ and $f_{\text{in}} \ll 1$, we further obtain the approximation

$$P_{\text{max},\varepsilon}^* \approx P^*\big|_{\eta=\eta_{\text{opt},\varepsilon}} - \frac{\varepsilon c}{f_{\text{in}}f_{\text{out}}\eta_{\text{opt},\varepsilon}c_m}. \tag{200}$$

See Fig 6D,E and S4 Fig for these approximations of $\eta_{\text{opt},\varepsilon}$ and $P_{\text{max},\varepsilon}^*$.

Network simulations

In this section, we describe the network simulations performed to obtain the numerical results presented in this paper. First, we report how the main parameters were sampled. Then, we explain the simulation paradigm in detail. Last, we clarify how we average across results for different patterns before briefly broaching the issue of enforcing the output activation ratio.

All simulations were performed in `python 3.9`. All code written in support of this publication will be made publicly available at `https://itbgit.biologie.hu-berlin.de/auer/sparseness_plasticity_lifetime` before publication.

Parameter sampling

Unless stated otherwise we perform all simulations for transition probabilities

$$\eta \in \{1, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1\} \tag{201}$$

and define the value for which the capacity is the largest as the optimal transition probability. For the reference network size of $N_{\text{in}} = N_{\text{out}} = 1000$, the activation ratios are typically sampled as

$$f_{\text{in}}, f_{\text{out}} \in \{0.2, 0.1, 0.05, 0.025, 0.012, 0.006, 0.003\}. \tag{202}$$

For larger network sizes, smaller values f_{in} and f_{out} are added to achieve minimal M_{in} and M_{out} values comparable to those of the small reference network.

The simulation paradigm

We now describe how patterns are sampled and how the weight matrix is initialized as well as how a simulation step is executed and how many steps are carried out in total. As also explained in the Results, we aim at quantifying the memory capacity of the network by tracking the signal quality, which depends on the Hamming distance between a calculated output vector and the corresponding target output vector. The capacity is defined as the number of subsequently learned patterns for which the signal quality reaches a specific value. In each simulation step the association between one pattern pair is learned, and in order to calculate the capacity we need a number of simulation steps that is larger than the capacity itself.

Input and output patterns Input patterns are generated as vectors $\mathbf{x}$ of length N_{in} by randomly choosing M_{in} entries to be one while the others are zero. Analogously, output patterns are random vectors $\mathbf{y}$ of length N_{out} with M_{out} entries equal to one. One random input pattern $\mathbf{x}$ together with one random output pattern $\mathbf{y}$ constitute a pattern pair $(\mathbf{x}, \mathbf{y})$. Note that we do not explicitly exclude the possibility of generating the same vector twice as an input or as an output pattern, but for $M_{\text{in}}, M_{\text{out}} \gg 1$ and $N_{\text{in}} \gg M_{\text{in}}$, $N_{\text{out}} \gg M_{\text{out}}$, this is very unlikely.

If noisy input patterns are used as retrieval cues, they are generated based on the noise-less input patterns in the following way: Each originally active input unit is deactivated with probability ε. We count the number of deactivated units and randomly choose the same number of originally inactive input units to be activated. The number of active input units M_{in} is thus maintained.

Initialization of the weight matrix First, we generate a matrix $J_m \in \{0, 1\}^{N_{\text{out}} \times N_{\text{in}}}$ that represents the morphological connections with connectivity c_m. In every row of J_m, we set $c_m N_{\text{in}}$ randomly chosen entries to one, and the other entries are zero. The binary matrix J_m acts as a mask on the actual weight matrix J in every update step such that only the entries in J that correspond to a one in J_m can be altered. The weight matrix J of size $N_{\text{out}} \times N_{\text{in}}$ is initialized such that per row cN_{in} entries of those that are morphologically available (i.e., have a value of one in J_m) are set to one, the rest is set to zero.

One simulation step Each simulation step k corresponds to one pattern pair $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})$ being learned. In one simulation step k, the weights of the matrix in the previous step, $J^{[k-1]}$, are first updated according to Hebbian learning and then updated according to the homeostatic mechanism (described in the Results).This corresponds to transforming $J^{[k-1]}$ into $J^{[k]}$.

Then, for a fixed $J^{[k]}$, we evaluate the output of the network for all input patterns of pairs $0, \ldots, k$ that have been learned up to this point; the input patterns are presented to the network again one after the other with plasticity off, and the corresponding

output patterns are calculated. By $\hat{\mathbf{y}}^{[i]}(j)$ we denote the output of the i-th input pattern calculated after learning j subsequent pattern pairs. To summarize, the outputs

$$\hat{\mathbf{y}}^{[0]}, \ldots, \hat{\mathbf{y}}^{[k]} \tag{203}$$

for each of the input patterns

$$\mathbf{x}^{[0]}, \ldots, \mathbf{x}^{[k]} \tag{204}$$

are calculated and compared to the original target outputs

$$\mathbf{y}^{[0]}, \ldots, \mathbf{y}^{[k]}, \tag{205}$$

respectively. The $k+1$ Hamming distances

$$H\left(\mathbf{y}^{[0]}, \hat{\mathbf{y}}^{[0]}(k)\right), H\left(\mathbf{y}^{[1]}, \hat{\mathbf{y}}^{[1]}(k-1)\right), \ldots, H\left(\mathbf{y}^{[k-1]}, \hat{\mathbf{y}}^{[k-1]}(1)\right), H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(0)\right) \tag{206}$$

between target and calculated output are stored. Note that these Hamming distances that are stored after one particular simulation step belong to different numbers of additional pattern pairs that are stored after the pattern that is tested. With respect to pattern 0, the maximum number k of additional pattern pairs have been stored, while with respect to pattern k, no additional pattern pair has been stored.

Number of simulation steps The average Hamming distance between two random f_{out}-sparse vectors of length N_{out} is

$$H_{\text{avg}} = 2N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}}) \ . \tag{207}$$

To define a threshold T_S for the successful retrieval of an output pattern, we use a certain fraction of H_{avg},

$$T_S = t_S H_{\text{avg}} \ , \tag{208}$$

where $t_S \in (0, 1)$. If not stated otherwise, the retrieval threshold is chosen as

$$T_S = N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}}) \tag{209}$$

and hence, $t_S = 0.5$.

The degradation of the calculated output vector is quantified by means of the signal quality S, which is defined as the difference between the average Hamming distance of two random vectors, H_{avg}, and $H(\mathbf{y}, \hat{\mathbf{y}})$:

$$S := H_{\text{avg}} - H(\mathbf{y}, \hat{\mathbf{y}}) \ . \tag{210}$$

For a successful retrieval of the calculated output, we then require $S > t_S H_{\text{avg}}$. This translates to the assumption that an output pattern can be recovered if the Hamming distance $H(\mathbf{y}, \hat{\mathbf{y}})$ between the target output $\mathbf{y}$ and the calculated output $\hat{\mathbf{y}}$ is smaller than the threshold $(1 - t_S)H_{\text{avg}}$. In this case, the output, which is typically degraded by subsequent storage of other input-output pairs, is still close enough to the original output pattern. To be more specific, let us consider the example of input-pattern pair 0 and k subsequent learning steps. The signal quality is then

$$S_k := H_{\text{avg}} - H\left(\mathbf{y}^{[0]}, \hat{\mathbf{y}}^{[0]}(k)\right), \tag{211}$$

which is based on the Hamming distance $H\left(\mathbf{y}^{[0]}, \hat{\mathbf{y}}^{[0]}(k)\right)$ of the target output $\mathbf{y}^{[0]}$ of the very first pattern pair and its output $\hat{\mathbf{y}}^{[0]}(k)$ calculated with weight matrix $J^{[k]}$.

The maximum number of simulation steps for a particular parameter combination was determined by the observed degradation of calculated output patterns. Once the signal quality for the first time is less than $0.8 \cdot H_{\text{avg}}$ (or equivalently, when the Hamming distance is above $0.2 \cdot H_{\text{avg}}$), 200 additional patterns are learned before the simulation is stopped.

We denote the total number of patterns that have been learned by the network at the end of a simulation by K.

Averaging

The signal quality that is shown in various figures in this manuscript is always averaged across many patterns. Recall that the number of subsequently learned patterns P is understood as relative to the pattern number $k \in \{0, \ldots, K\}$. For each pattern $(\mathbf{x}^{[k]}, \mathbf{y}^{[k]})$ that the network learns, we compute the Hamming distance of the target output and the calculated output after $P = 0, 1, 2, \ldots$ patterns additionally learned after the k-th pattern, hence with weight matrix $J^{[k+P]}$. For a fixed P (always relative to a particular pattern), we average across all patterns for which we have learned at least P subsequent patterns:

$$H_K\left(\mathbf{y}, \hat{\mathbf{y}}(P)\right) := \frac{1}{K-P} \sum_{k=0}^{K-P} H\left(\mathbf{y}^{[k]}, \hat{\mathbf{y}}^{[k]}(P)\right). \tag{212}$$

All simulation data shown in this manuscript are obtained by averaging across at least $N_{\text{avg}} = 200$ patterns. This means, if the largest P shown in a figure is N_P, simulations ran for at least $K = N_P + N_{\text{avg}}$ steps.

Whenever we investigate the distributions of dendritic sums for a particular number of subsequent patterns P, two kinds of averaging are done: First, we pool the dendritic sums of all genuine output units and of all spurious output units of one pattern after learning P subsequent patterns, respectively. In addition, we pool a certain number of patterns. A comparison of distributions of dendritic sums obtained from numerical simulations and from theory (Eqs. (61) and (63)) is shown in S5 Fig.

A note on the output sparseness

During retrieval, the activation of the output units is calculated as

$$\hat{y}_j^{[k]}(P) = \Theta\left(\left(\sum_{i=1}^{N_{\text{in}}} J_{ji}^{[k+P]} x_i^{[k]}\right) - T_{\text{in}}^{[P]}\right) \tag{213}$$

where Θ is the Heaviside step function. Note that, due to the discrete nature of the dendritic sums of the output units, it is unlikely that an activation threshold $T_{\text{in}}^{[P]}$ exists that allows to activate exactly M_{out} output units. Most of the times, there is a group of units with identical dendritic sums that are exactly at the critical value. If the threshold is chosen larger than these dendritic sums, too few units are activated, if it is chosen smaller, too many units are activated. Since we want to keep M_{out} fixed, we have to find a way to enforce the exact output sparseness. Hence, we typically randomly choose the right number of these units to be activated and the rest to be inactive. On average, this has a slightly deteriorative effect on the capacity of the network because spurious units might be chosen to be activated even though some genuine units have the same dendritic sum. In S5 Appendix, we discuss in more detail for which parameters this might have significant effects on the capacity of the network and how the enforcement of the output sparseness level could be handled differently in order to avoid this problem.

Variable	Description	Default Value
N_{in}	Number of input units	1000
N_{out}	Number of output units	1000
f_{in}	Input activation ratio	0.025
f_{out}	Output activation ratio	0.025
M_{in}	Number of active input units	25
M_{out}	Number of active output units	25
$J^{[k]}$	Binary weight matrix after learning k patterns	
c_m	Morphological connectivity	1
c	Functional connectivity	0.2
η	Transition probability	
η_{opt}	Optimal transition probability	
$\mathbf{x}^{[k]}$	k-th input pattern	
$\mathbf{y}^{[k]}$	k-th output pattern	
$\hat{\mathbf{y}}^{[k]}(l)$	Output given the k-th input pattern calculated after learning l additional patterns	
P	Number of subsequent patterns learned since a particular pattern	
K	Total number of patterns learned at the end of a simulation	
N_{avg}	Minimal number of patterns used for averaging in simulations	200
$T_S = t_S H_{\text{avg}}$	Retrieval threshold	
t_S	Retrieval ratio	0.5
H_{avg}	Average Hamming distance between random patterns with given size and activation ratio	
$H(a, b)$	Hamming distance between a and b	
$S_P = s_P H_{\text{avg}}$	Signal quality after P subsequent patterns	
s_P	Normalized signal quality after P subsequent patterns	
P^*	Memory capacity	
P^*_{max}	Maximal memory capacity (with optimal transition probability)	
ε	Noise level on input patterns	0.2
$d_g^{[P]}$	Dendritic sum of genuine unit after P subsequent patterns	
d_s	Dendritic sum of spurious unit	
$p_g^{[P]}$	PMF of dendritic sums of genuine units after P subsequent patterns	
p_s	PMF of dendritic sums of spurious units	
$F_g^{[P]}$	CDF of dendritic sums of genuine units after P subsequent patterns	
F_s	CDF of dendritic sums of spurious units	
ρ_g	Probability of a functional genuine-genuine connection	
ρ_s	Probability of a functional genuine-spurious connection	
ρ_n	Probability of a functional spurious-genuine connection	
d'_R	Generalized sensitivity index	

Table 2. Description of variable names and their default values.

Discussion

We studied the effect of population sparseness on the optimal learning speed in an associative network. We quantified the storage capacity of a heteroassociative feedforward network, where the capacity is based on a threshold of Hamming distance between the target patterns and the calculated output patterns.Furthermore, we determined the transition probability that maximizes the capacity (see Eqs. (19)-(21)).Analytical derivations and numerical simulations show that the optimal transition probability increases with the sparseness of both the input and the output patterns, but the impact of the input sparseness is stronger (see Fig 3B,C,F).The maximum capacity is largest for the lowest *output* activation ratio, whereas it depends non-monotonically on the *input* activation ratio (Fig 3B-D).We found that the maximum capacity is extensive with respect to the input layer size but that the optimal transition probability is independent of the network size (Fig 5).As already hinted at by [71] in a gradient descent framework, we confirm that not the input activation ratio but the actual number of active input units is the crucial parameter that determines the optimal transition probability — independent of the network size (Fig 5A).The number of active input units that yields the largest possible capacity is also independent of the network size (Fig 5). All results hold for recall with perfect input patterns as well as with noisy input patterns. As expected, the maximal capacity decreases as a function of the noise level but, interestingly, the optimal transition probability increases (Fig 6).In contrast to previous works based on single neurons [15] or biologically implausible algorithms such as gradient descent [64, 71], which is non-local, our in-depth numerical and analytical analysis based on Hebbian and homeostatic learning presents a substantial advancement as it reveals the distinct effects of input and output sparseness as well as network size and input noise on the optimal learning speed while relying on a very simple network and pattern structure.

Palimpsest memory and homeostatic plasticity

The architecture of the network used in this manuscript is similar to the Willshaw network [28] in that it is a feedforward one-layer network that learns associations between binary input and output patterns in a Hebbian way.In contrast to a classical Willshaw network in which all memories disappear when the capacity is reached, our network realizes a palimpsest memory system in which older memories are erased by newer memories due to a homeostasis mechanism (see e.g. 3A). This degradation of older memories allows for an online learning regime where an in principle infinite sequence of patterns can be learned — new patterns always gradually replacing older ones [35, 37]. The signal quality of a memory pattern hence does not only passively decay over time due to the interference between patterns, but also actively due to the deactivation or depression of synaptic connections. The underlying heterosynaptic plasticity, which is characterized by an input-unspecific change of synaptic efficacies [72, 73], has been found, e.g., in the hippocampus and the neocortex [74, 75]. Heterosynaptic plasticity can provide an effective homeostatic constraint on Hebbian plasticity [76, 77]. In the presented work, a combination of Hebbian homosynaptic and homeostatic heterosynaptic learning was used. Hebbian plasticity increases connection weights with a predefined probability if both the pre- and the postsynaptic units are active, and homeostatic plasticity decreases weights with another predefined probability if only the postsynaptic unit is active (details in Methods).The update probabilities, which can be interpreted as learning rates, are balanced such that the total number of functional connections per output unit remains constant. This special heterosynaptic plasticity rule is inspired by experimental evidence of a constant total summed synaptic weight on a dendritic branch [66, 78].

In general, variations of the implemented homeostatic plasticity are possible. For example, in contrast to our approach, the total number of functional connections per input unit could be conserved to realize a presynaptic homeostasis by depressing connections from active input neurons to inactive output neurons (as suggested by [35], e.g., reported by [79]). Alternatively, the number of functional connections could only be conserved across the entire network, i.e. not be normalized per (input or output) unit; such a normalization could be achieved by randomly silencing connections that are not involved in a given association of patterns [36]. Numerical simulations based on either of these two variations suggest that the qualitative dependence of the optimal learning speed on the sparseness of the patterns is the same as for the learning rule analyzed in detail in this paper (see S4 Appendix). Finally, a decrease in synaptic efficacies could occur due to synaptic aging where connections that have not been involved in a pre- and postsynaptic co-activity for a long time might be more likely to be depressed than those that have been strengthened more recently [80].

Input sparseness vs output sparseness

In our heteroassociative network, both a change in the input and the output sparseness value affect the optimal learning speed and the corresponding maximum memory capacity (Figs 3, 5 and 6, Eqs. (19)-(21), details in Methods).The optimal transition probability monotonically decreases with an increase in the input activation ratio as well as with an increase in the output activation ratio but the gradient is steeper for the input activation ratio (see Figs 3E, 6B and 21). In order to learn fast, sparseness in the input representations is thus more beneficial than sparseness in the output representations. As known from the literature [28, 32, 35], the capacity of an associative network benefits from sparse representations. The same is of course true for the maximum capacity, which is obtained by learning with the optimal transition probability for each combination of parameters. While enforcing sparser output patterns decreases the overlap in the output space and hence decreases the likelihood of retrieval errors up to extreme sparseness values of only one active output unit, using too sparse input patterns can be detrimental to the capacity. If the input representations contain a too small number of active units M_{in}, it is difficult for the signal, which scales as M_{in}, to overcome the noise, which scales as $\sqrt{M_{\mathrm{in}}}$ (Figs 3A,B and 5B). In other words, a very small number of active input units cannot reliably drive the respective output units. If M_{in} is smaller than some optimal value, the initial signal quality approaches the retrieval threshold, and if M_{in} becomes very small, the capacity can be zero because the initial signal quality lies below the threshold [36, 37]. This collapse of the maximal capacity for too sparse input patterns only occurs in the range of M_{in}-values where the optimal transition probability has already reached its maximal value 1 (Fig 5A, details in Methods). Note that, while the assumption of only binary synaptic weights combined with a transition probability is a simplification of the state transitions occurring in real synapses, assuming a bounded number of discrete states is realistic [81, 82]. It hence makes sense to assume that the learning speed in terms of the amount of potentiation experienced by synapses combined with a probability of potentiation is bound to reach an upper limit.

Choice of the activation threshold and optimized capacity

The central purpose of this work is to rigorously establish the relationship between the population sparseness of the patterns and the optimal learning speed of the network. In this spirit, for a given output activation ratio, the setup needs to enable a meaningful comparison of target and calculated output patterns in order to quantify the capacity and therewith the optimal transition probability as functions of the activation ratio.

Thus, it is of crucial interest to enforce a particular number of active output units M_{out}, which is implemented by choosing the activation threshold of the McCulloch-Pitts output neurons accordingly (Eq. (1)).The maximum capacity calculated in this framework (Eq. (21), details in Methods) represents the highest capacity obtained under the given constraints using the optimal transition probability. In other approaches, the beneficial effect of sparseness on the capacity has been studied extensively in different settings. For example, [33] could increase the capacity by using an optimized, covariance-based learning rule. Moreover, [83] suggested the use of a numerically optimized activation threshold in a setup that is very similar to ours. In this manuscript, such aspects have been excluded, in an attempt to keep the model as simple as possible to allow for a detailed analysis of the effect of sparseness on the learning speed. The behavior of the capacity that is reported in the current work is always focused on its relation to the optimal transition probability.

Role of the output size and retrieval criterion

The analytical description of the memory capacity relies on the distributions of dendritic sums which are based on the synaptic connections of single output units (Eqs. (8) and (9)). For the Hamming distance (Eq. (12)) and thus also for the signal quality S_P (Eq. (5), also see Methods), the number of output units N_{out} has to be taken into account as the probabilities of output units being in the wrong state are scaled by the respective number of units. We defined the memory capacity as the number of subsequent patterns that can be learned until the signal quality S_P reaches the retrieval threshold T_S (Eq. (6) and Eq. (15)). The retrieval threshold $T_S = t_S H_{\text{avg}}$, with a fixed $t_S \in (0, 1)$, is chosen as a fraction of $H_{\text{avg}} = 2N_{\text{out}} f_{\text{out}}(1 - f_{\text{out}})$, the average Hamming distance between two random f_{out}-sparse patterns of size N_{out} (Eq. (7)). Hence, the number of wrong output units that are accepted for the pattern to be classified as retrievable depends on N_{out} and f_{out}. This dependence accounts for the fact that a fixed number of wrongly activated units has a more severe destructive effect if the total number of active output units is small. For example, if in a pattern with $M_{\text{out}} = 100$ active output units nine are wrong, the pattern is still very similar to the original one; if $M_{\text{out}} = 10$, nine wrong units are detrimental. The capacity is obtained by comparing the signal quality S_P to the retrieval threshold T_S or, equivalently, by comparing the normalized signal quality S_P/H_{avg} to the retrieval ratio t_S. Since the normalized signal quality S_P/H_{avg} does not depend on N_{out} (derived in Methods) but only on the distributions of the dendritic sums and f_{out} (see Eqs. (77) and (79) as well as S2 Appendix), the capacity does not depend on N_{out} either.

As an alternative, which has not been explored in this manuscript, the retrieval threshold could be defined independent of H_{avg} and thus independent of N_{out}, e.g., implementing the retrieval criterion that a maximum of two units may be wrongly activated or that at least ten units need to be correctly activated. In such alternative cases, the capacity could depend on the output network size N_{out}.

Furthermore, the output size as well as the input size in combination with the output and input activation ratios, respectively, pose an upper limit to the representational capacity, i.e., the maximal number of distinct patterns. If every input pattern and every output pattern should occur maximally once, the number of distinct input-output pattern pairs is bounded by $\min\left(\binom{N_{\text{in}}}{M_{\text{in}}}, \binom{N_{\text{out}}}{M_{\text{out}}}\right)$.

In general, however, even if the size of the output layer is negligible for the storage capacity under specific circumstances, it plays an important role in a potential additional readout layer, e.g. a classification of the output representations. In [84], it has been shown that an increase in the number of output units increases the performance of a readout unit that performs a binary classification.

Fast learning

In the context of fast learning, several distinct but related notions are used. Here, it seems useful to explain and compare the concepts of transition probability, learning rate, and convergence time. [64] and [15] are the only two references that discuss the impact of sparseness on learning speed. They build their arguments on analyzing learning rate and convergence, while we focus on the transition probability.

A high transition probability refers to the likelihood that a synaptic weight is increased during learning, while a large learning rate refers to the magnitude of synaptic weight updates (per time unit) during learning. These two are closely related as they both affect changes in weights, and the average weight change in one learning step is proportional to the product of the learning rate and the transition probability. In the present work, connections can only be silent (0) or functional (1). Although the update steps from 0 to 1 or from 1 to 0 could be interpreted as a fixed learning rate, the learning speed is variable and represented by the transition probability η.

Another possibility to vary the learning speed would be to use a fixed transition probability while varying the learning rate, e.g., by weight changes that are continuous and within the interval between 0 and 1. We believe that such an alternative model may well yield qualitatively similar results for the effect of sparseness on the optimal learning speed. It would also be interesting to analyze more complex and more realistic synapse models in this context, but all this is beyond the scope of the present work.

In contrast, the term "convergence" and its magnitude do not refer to how many or by how much weights are altered, but it is a system-level outcome. It refers to how quickly a neural network stabilizes or reaches its learning objective, e.g., a minimum in a gradient-descent learning algorithm. While a high learning rate and a high transition probability can both speed up weight updates, fast convergence is achieved by balancing these factors to ensure that the network efficiently settles into an optimal or stable solution. A high learning rate can accelerate updates towards a solution, but it does not necessarily relate to fast convergence. On the contrary, if the update steps are too large, they risk overshooting or instability. [64] argue that sparse representations yield faster convergence because the learning rate can be increased without risking instabilities. In our model, convergence is not evaluated because every pattern is only presented once to do one weight update step, and output patterns are always calculated from the input in one step. It would be interesting to investigate the effect of sparseness on transition probability (and/or learning rate) in combination with convergence speed in a network model that allows for both. Building on the results of this manuscript and [64]'s reasoning, we believe that high sparseness would yield a high optimal transition probability (or learning rate) while converging fast because sparseness allows for large update steps without instability.

Optimal sparseness values

In terms of the capacity obtained with an optimized transition probability, our study predicts an optimal small but not too small input activation ratio f_{in} and an optimal output activation ratio $f_{\text{out}} = 1/N_{\text{out}}$ (Fig 3). The optimal f_{in} depends on the input size N_{in} while the optimal number of active input units M_{in} does not (Fig 5). Nevertheless, the optimal M_{in} cannot be interpreted as a quantitative prediction of the optimal number of active units because it depends on the retrieval threshold T_S (see Methods).If T_S is increased, i.e., if the retrieval criterion becomes more strict, the optimal M_{in} increases (and the corresponding maximum capacity decreases).

Further, the quantification of the optimal learning speed in our model is solely focused on optimizing the capacity of the network.In reality, the brain might favor a trade-off of various properties instead of optimization for one purpose [71]. Even though

sparseness can be advantageous, e.g., in terms of energy efficiency [85], storage capacity [86] and downstream classification [84], it might be beneficial to avoid extreme sparseness because it can reduce the representational capacity [71, 87], the information capacity per pattern [88, 89] and the generalization capacity [71, 90] and it is less robust to noise or damage of neuronal structures [71, 87]. Moreover, sparse connectivity, which is also ubiquitous in the brain, seems to yield similar favorable effects as population sparseness and could be optimized in combination with sparse activity [71, 91].

In general, it is thus not reasonable to predict *one* optimal activation ratio or *one* optimal number of active neurons. Optimal (and actual) values are probably different for different brain regions, cell types, tasks, etc. First steps towards more concrete optimal sparseness values could combine experimentally measured values, obtained with modern sophisticated techniques such as Neuropixels recordings, and computational models that take into account the specific properties and presumed role(s) of particular brain regions. [2]

Supporting information

S1 Fig. Comparison of optimal transition probability $\eta_{\mathbf{opt}}$ and maximal capacity $P^*_{\mathbf{max}}$ for different retrieval thresholds T_S.

S2 Fig. Comparison of optimal transition probability $\eta_{\mathbf{opt}}$ and maximal capacity $P^*_{\mathbf{max}}$ for different morphological connectivity levels c_m.

S3 Fig. Comparison of optimal transition probability $\eta_{\mathbf{opt}}$ and maximal capacity $P^*_{\mathbf{max}}$ for different functional connectivity levels c.

S4 Fig. Comparison of optimal transition probability $\eta_{\mathbf{opt},\varepsilon}$ and maximal capacity $P^*_{\mathbf{max},\varepsilon}$ for different input noise levels ε during retrieval.

S5 Fig. Comparison of distributions of dendritic sums obtained from network simulations and from theory.

S3 Fig. Distributions of dendritic sums with noise ($\varepsilon = 0.4$).

S1 Appendix. Analysis of shape of distributions.

S2 Appendix. Analytical calculation of the signal quality.

S3 Appendix. Monotonicity of capacity for small $f_{\mathbf{in}}$.

S4 Appendix. Comparison to other versions of homeostasis.

S5 Appendix. Can the output activation ratio be enforced?

Acknowledgments

We thank Natalie Schieferstein for valuable comments on the manuscript.

References

1. Chawla M, Guzowski J, Ramirez-Amaya V, Lipa P, Hoffman K, Marriott L, et al. Sparse, environmentally selective expression of Arc RNA in the upper blade of the rodent fascia dentata by brief spatial experience. Hippocampus. 2005;15(5):579–586.

2. GoodSmith D, Chen X, Wang C, Kim SH, Song H, Burgalossi A, et al. Spatial representations of granule cells and mossy cells of the dentate gyrus. Neuron. 2017;93(3):677–690.

3. Liu X, Ramirez S, Pang PT, Puryear CB, Govindarajan A, Deisseroth K, et al. Optogenetic stimulation of a hippocampal engram activates fear memory recall. Nature. 2012;484(7394):381–385.

4. Rao-Ruiz P, Yu J, Kushner SA, Josselyn SA. Neuronal competition: microcircuit mechanisms define the sparsity of the engram. Current Opinion in Neurobiology. 2019;54:163–170.

5. Tayler KK, Tanaka KZ, Reijmers LG, Wiltgen BJ. Reactivation of neural ensembles during the retrieval of recent and remote memory. Current Biology. 2013;23(2):99–106.

6. Ramsaran AI, Wang Y, Golbabaei A, Aleshin S, De Snoo ML, Yeung BrA, et al. A shift in the mechanisms controlling hippocampal engram formation during brain maturation. Science. 2023;380(6644):543–551.

7. Beyeler M, Dutt N, Krichmar JL. 3D visual response properties of MSTd emerge from an efficient, sparse population code. Journal of Neuroscience. 2016;36(32):8399–8415.

8. Olshausen BA, Field DJ. Sparse coding of sensory inputs. Current Opinion in Neurobiology. 2004;14(4):481–487.

9. Froudarakis E, Berens P, Ecker AS, Cotton RJ, Sinz FH, Yatsenko D, et al. Population code in mouse V1 facilitates readout of natural scenes through increased sparseness. Nature Neuroscience. 2014;17(6):851–857.

10. Tang S, Zhang Y, Li Z, Li M, Liu F, Jiang H, et al. Large-scale two-photon imaging revealed super-sparse population codes in the V1 superficial layer of awake monkeys. eLife. 2018;7:e33370.

11. Yoshida T, Ohki K. Natural images are reliably represented by sparse and variable populations of neurons in visual cortex. Nature Communications. 2020;11(1):872.

12. Chen X, Zhu S, Bai K, Xia R, Kong NC, Norcia AM, et al. Rapid Selectivity to Natural Images Across Layers of Primate V1. bioRxiv. 2022; p. 2022–01.

13. Weliky M, Fiser J, Hunt RH, Wagner DN. Coding of natural scenes in primary visual cortex. Neuron. 2003;37(4):703–718.

14. Yen SC, Baker J, Gray CM. Heterogeneity in the responses of adjacent neurons to natural stimuli in cat striate cortex. Journal of Neurophysiology. 2007;97(2):1326–1341.

15. Hromádka T, DeWeese MR, Zador AM. Sparse representation of sounds in the unanesthetized auditory cortex. PLoS Biology. 2008;6(1):e16.

16. Feigin L, Tasaka G, Maor I, Mizrahi A. Sparse coding in temporal association cortex improves complex sound discriminability. Journal of Neuroscience. 2021;41(33):7048–7064.

17. Clemens J, Wohlgemuth S, Ronacher B. Nonlinear computations underlying temporal and population sparseness in the auditory system of the grasshopper. Journal of Neuroscience. 2012;32(29):10053–10062.

18. Perez-Orive J, Mazor O, Turner GC, Cassenaer S, Wilson RI, Laurent G. Oscillations and sparsening of odor representations in the mushroom body. Science. 2002;297(5580):359–365.

19. Turner GC, Bazhenov M, Laurent G. Olfactory representations by Drosophila mushroom body neurons. Journal of Neurophysiology. 2008;99(2):734–746.

20. Ito I, Ong RCy, Raman B, Stopfer M. Sparse odor representation and olfactory learning. Nature Neuroscience. 2008;11(10):1177–1184.

21. Miura K, Mainen ZF, Uchida N. Odor representations in olfactory cortex: distributed rate coding and decorrelated population activity. Neuron. 2012;74(6):1087–1098.

22. Ranjbar-Slamloo Y, Arabzadeh E. Diverse tuning underlies sparse activity in layer 2/3 vibrissal cortex of awake mice. The Journal of Physiology. 2019;597(10):2803–2817.

23. Gouty-Colomer L, Hosseini B, Marcelo I, Schreiber J, Slump DE, Yamaguchi S, et al. Arc expression identifies the lateral amygdala fear memory trace. Molecular Psychiatry. 2016;21(3):364–375.

24. Lehky SR, Kiani R, Esteky H, Tanaka K. Statistics of visual responses in primate inferotemporal cortex to object stimuli. Journal of Neurophysiology. 2011;106(3):1097–1117.

25. Rust NC, DiCarlo JJ. Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area V4 to IT. Journal of Neuroscience. 2010;30(39):12978–12995.

26. Poo C, Isaacson JS. Odor representations in olfactory cortex: "sparse" coding, global inhibition, and oscillations. Neuron. 2009;62(6):850–861.

27. Markind C, Kundu P, Barak M, Haddad R. Increased pattern similarity in two major olfactory cortices despite higher sparseness levels. bioRxiv. 2021; p. 2021–04.

28. Willshaw DJ, Buneman OP, Longuet-Higgins HC. Non-holographic associative memory. Nature. 1969;222(5197):960–962.

29. Amit DJ, Gutfreund H, Sompolinsky H. Information storage in neural networks with low levels of activity. Physical Review A. 1987;35(5):2293.

30. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences. 1982;79(8):2554–2558.

31. Amari SI. Learning patterns and pattern sequences by self-organizing nets of threshold elements. IEEE Transactions on Computers. 1972;100(11):1197–1206.

32. Gardner E. Maximum storage capacity in neural networks. Europhysics Letters. 1987;4(4):481.

33. Tsodyks MV, Feigel'man MV. The enhanced storage capacity in neural networks with low activity level. Europhysics Letters. 1988;6(2):101.

34. Treves A, Rolls ET. What determines the capacity of autoassociative memories in the brain? Network: Computation in Neural Systems. 1991;2(4):371.

35. Amit DJ, Fusi S. Learning in neural networks with material synapses. Neural Computation. 1994;6(5):957–982.

36. Ben Dayan Rubin DD, Fusi S. Long memory lifetimes require complex synapses and limited sparseness. Frontiers in Computational Neuroscience. 2007;1:117.

37. Leibold C, Kempter R. Sparseness constrains the prolongation of memory lifetime via synaptic metaplasticity. Cerebral Cortex. 2008;18(1):67–77.

38. Dubreuil AM, Amit Y, Brunel N. Memory capacity of networks with stochastic binary synapses. PLoS Computational Biology. 2014;10(8):e1003727.

39. Benna MK, Fusi S. Computational principles of synaptic memory consolidation. Nature Neuroscience. 2016;19(12):1697–1706.

40. Schönsberg F, Roudi Y, Treves A. Efficiency of local learning rules in threshold-linear associative networks. Physical Review Letters. 2021;126(1):018301.

41. Feng Y, Brunel N. Storage capacity of networks with discrete synapses and sparsely encoded memories. Physical Review E. 2022;105(5):054408.

42. Attardo A, Fitzgerald JE, Schnitzer MJ. Impermanence of dendritic spines in live adult CA1 hippocampus. Nature. 2015;523(7562):592–596.

43. Piette C, Touboul J, Venance L. Engrams of fast learning. Frontiers in Cellular Neuroscience. 2020;14:575915.

44. Liao Z, Losonczy A. Learning, Fast and Slow: Single-and Many-Shot Learning in the Hippocampus. Annual Review of Neuroscience. 2024;47:187–209.

45. Lee D, Lin BJ, Lee AK. Hippocampal place fields emerge upon single-cell manipulation of excitability during behavior. Science. 2012;337(6096):849–853.

46. Bittner KC, Milstein AD, Grienberger C, Romani S, Magee JC. Behavioral time scale synaptic plasticity underlies CA1 place fields. Science. 2017;357(6355):1033–1036.

47. Milstein AD, Li Y, Bittner KC, Grienberger C, Soltesz I, Magee JC, et al. Bidirectional synaptic plasticity rapidly modifies hippocampal representations. eLife. 2021;10:e73046.

48. Viskontas IV, Quiroga RQ, Fried I. Human medial temporal lobe neurons respond preferentially to personally relevant images. Proceedings of the National Academy of Sciences. 2009;106(50):21329–21334.

49. Rutishauser U, Mamelak AN, Schuman EM. Single-trial learning of novel stimuli by individual neurons of the human hippocampus-amygdala complex. Neuron. 2006;49(6):805–813.

50. Ison MJ, Quiroga RQ, Fried I. Rapid encoding of new memories by individual neurons in the human brain. Neuron. 2015;87(1):220–230.

51. Horner AJ, Doeller CF. Plasticity of hippocampal memories in humans. Current Opinion in Neurobiology. 2017;43:102–109.

52. McClelland JL, McNaughton BL, O'Reilly RC. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. Psychological Review. 1995;102(3):419.

53. Remme MW, Bergmann U, Alevi D, Schreiber S, Sprekeler H, Kempter R. Hebbian plasticity in parallel synaptic pathways: A circuit mechanism for systems memory consolidation. PLOS Computational Biology. 2021;17(12):e1009681.

54. Squire LR, Genzel L, Wixted JT, Morris RG. Memory consolidation. Cold Spring Harbor Perspectives in Biology. 2015;7(8):a021766.

55. Yang G, Pan F, Gan WB. Stably maintained dendritic spines are associated with lifelong memories. Nature. 2009;462(7275):920–924.

56. Castro-Alamancos MA, Connors BW. Distinct forms of short-term plasticity at excitatory synapses of hippocampus and neocortex. Proceedings of the National Academy of Sciences. 1997;94(8):4161–4166.

57. Navarro Lobato I, Aleman-Zapata A, Samanta A, Bogers M, Narayanan S, Rayan A, et al. Increased cortical plasticity leads to memory interference and enhanced hippocampal-cortical interactions. eLife. 2023;12:e84911.

58. Hebscher M, Wing E, Ryan J, Gilboa A. Rapid cortical plasticity supports long-term memory formation. Trends in Cognitive Sciences. 2019;23(12):989–1002.

59. Brodt S, Gais S, Beck J, Erb M, Scheffler K, Schönauer M. Fast track to the neocortex: A memory engram in the posterior parietal cortex. Science. 2018;362(6418):1045–1048.

60. Tse D, Langston RF, Kakeyama M, Bethus I, Spooner PA, Wood ER, et al. Schemas and memory consolidation. Science. 2007;316(5821):76–82.

61. Tse D, Takeuchi T, Kakeyama M, Kajii Y, Okuno H, Tohyama C, et al. Schema-dependent gene activation and memory encoding in neocortex. Science. 2011;333(6044):891–895.

62. Hofstetter S, Friedmann N, Assaf Y. Rapid language-related plasticity: microstructural changes in the cortex after a short session of new word learning. Brain Structure and Function. 2017;222:1231–1241.

63. Solomon SH, Kay K, Schapiro AC. Semantic plasticity across timescales in the human brain. eLife. 2024;13.

64. Schweighofer N, Doya K, Lay F. Unsupervised learning of granule cell sparse codes enhances cerebellar adaptive control. Neuroscience. 2001;103(1):35–50.

65. Davis GW, Bezprozvanny I. Maintaining the stability of neural function: a homeostatic hypothesis. Annual Review of Physiology. 2001;63(1):847–869.

66. Bourne JN, Harris KM. Coordination of size and number of excitatory and inhibitory synapses results in a balanced structural plasticity along mature hippocampal CA1 dendrites during LTP. Hippocampus. 2011;21(4):354–373.

67. Nadal J, Toulouse G, Changeux J, Dehaene S. Networks of formal neurons and memory palimpsests. Europhysics Letters. 1986;1(10):535.

68. McCulloch WS, Pitts W. A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics. 1943;5:115–133.

69. Buckingham JT. Delicate Nets, Faint Recollections: A Study of Partially Connected Associative Network Memories. University of Edinburgh. 1991;.

70. Winitzki S. Uniform approximations for transcendental functions. In: Computational Science and Its Applications—ICCSA 2003: International Conference Montreal, Canada, May 18–21, 2003 Proceedings, Part I 3. Springer; 2003. p. 780–789.

71. Spanne A, Jörntell H. Questioning the role of sparse coding in the brain. Trends in Neurosciences. 2015;38(7):417–427.

72. Lynch GS, Dunwiddie T, Gribkoff V. Heterosynaptic depression: a postsynaptic correlate of long-term potentiation. Nature. 1977;266(5604):737–739.

73. Abraham W, Goddard GV. Asymmetric relationships between homosynaptic long-term potentiation and heterosynaptic long-term depression. Nature. 1983;305(5936):717–719.

74. Jenks KR, Tsimring K, Ip JPK, Zepeda JC, Sur M. Heterosynaptic plasticity and the experience-dependent refinement of developing neuronal circuits. Frontiers in Neural Circuits. 2021;15:803401.

75. Chistiakova M, Volgushev M. Heterosynaptic plasticity in the neocortex. Experimental Brain Research. 2009;199:377–390.

76. Chen JY, Lonjers P, Lee C, Chistiakova M, Volgushev M, Bazhenov M. Heterosynaptic plasticity prevents runaway synaptic dynamics. Journal of Neuroscience. 2013;33(40):15915–15929.

77. Chistiakova M, Bannon NM, Chen JY, Bazhenov M, Volgushev M. Homeostatic role of heterosynaptic plasticity: models and experiments. Frontiers in Computational Neuroscience. 2015;9:89.

78. Keck T, Toyoizumi T, Chen L, Doiron B, Feldman DE, Fox K, et al. Integrating Hebbian and homeostatic plasticity: the current state of the field and future research directions. Philosophical Transactions of the Royal Society B: Biological Sciences. 2017;372(1715):20160158.

79. Stanton PK, Sejnowski TJ. Associative long-term depression in the hippocampus induced by hebbian covariance. Nature. 1989;339(6221):215–218.

80. Henson R. Short-Term Associative Memories. University of Edinburgh. 1993;.

81. Petersen CC, Malenka RC, Nicoll RA, Hopfield JJ. All-or-none potentiation at CA3-CA1 synapses. Proceedings of the National Academy of Sciences. 1998;95(8):4732–4737.

82. Bartol Jr TM, Bromer C, Kinney J, Chirillo MA, Bourne JN, Harris KM, et al. Nanoconnectomic upper bound on the variability of synaptic plasticity. eLife. 2015;4:e10778.

83. Wu Y, Maass W. A simple model for Behavioral Time Scale Synaptic Plasticity (BTSP) provides content addressable memory with binary synapses and one-shot learning. Nature Communications. 2025;16(1):342.

84. Babadi B, Sompolinsky H. Sparseness and expansion in sensory representations. Neuron. 2014;83(5):1213–1226.

85. Yu L, Yu Y. Energy-efficient neural information processing in individual neurons and neuronal networks. Journal of Neuroscience Research. 2017;95(11):2253–2266.

86. Fusi S. Memory capacity of neural network models. In: Kahana MJ, Wagner AD, editors. The Oxford Handbook of Human Memory, Two Volume Pack: Foundations and Applications. Oxford University Press; 2024. p. 740–764.

87. Földiak P. Forming sparse representations by local anti-Hebbian learning. Biological Cybernetics. 1990;64(2):165–170.

88. Rolls E, Treves A. Neural Networks and Brain Function. Oxford University Press; 1997.

89. Sharma S, Chandra S, Fiete I. Content addressable memory without catastrophic forgetting by heteroassociation with a fixed scaffold. In: International Conference on Machine Learning. PMLR; 2022. p. 19658–19682.

90. Beyeler M, Rounds EL, Carlson KD, Dutt N, Krichmar JL. Neural correlates of sparse coding and dimensionality reduction. PLoS Computational Biology. 2019;15(6):e1006908.

91. Thom M, Palm G. Sparse activity and sparse connectivity in supervised learning. The Journal of Machine Learning Research. 2013;14(1):1091–1143.