

RESEARCH ARTICLE

# Remembering the “when”: Hebbian memory models for the time of past events

Johanni Brea<sup>1,2\*</sup>, Alireza Modirshanechi<sup>1,2</sup>, Georgios Iatropoulos<sup>1,2</sup>, Wulfram Gerstner<sup>1,2</sup>

**1** School of Computer and Communication Science, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, **2** Brain Mind Institute, School of Life Science, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland

\* [johanni.brea@epfl.ch](mailto:johanni.brea@epfl.ch)



## OPEN ACCESS

**Citation:** Brea J, Modirshanechi A, Iatropoulos G, Gerstner W (2026) Remembering the “when”: Hebbian memory models for the time of past events. PLoS Comput Biol 22(7): e1014575. <https://doi.org/10.1371/journal.pcbi.1014575>

**Editor:** Emili Balaguer-Ballester, Bournemouth University, UNITED KINGDOM OF GREAT BRITAIN AND NORTHERN IRELAND

**Received:** June 23, 2025

**Accepted:** July 14, 2026

**Published:** July 28, 2026

**Copyright:** © 2026 Brea et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data availability statement:** The code for all simulations is available at <https://github.com/jbrea/RememberingTheWhen.jl>.

**Funding:** This work was supported by the Swiss National Science Foundation with grant 200020\_184615 (J.B., A.M., W.G.). The funders

## Abstract

Humans and animals can remember how long ago specific events happened. Little is known about the neural mechanisms that enable remembering the “when” of memories stored for long durations in the episodic memory system – in contrast to interval-timing on the order of seconds and minutes. Based on a systematic exploration of neural coding, association and retrieval schemes, we develop model classes that span the space of possible mechanisms for the reconstruction of the time of past events. In concrete examples we show how network architecture, Hebbian plasticity, synaptic pruning or systems consolidation allow the retrieval of the time of past events. In a simulation, we demonstrate how these mechanisms would enable food-caching animals such as corvids to remember what they cached, where, and how long ago. To dissociate different hypotheses, we propose three kinds of novel, non-verbal experiments that can be run with humans and animals. Our simulations predict the experimental results for different classes of models. Our study shows that remembering the “when” can be implemented by many biologically plausible mechanisms and that carefully designed experiments are needed to pin down the actual neural implementation of the memory for the time of past events in different species.

## Author summary

Was it yesterday, a week ago, or perhaps even longer since I last ate pizza? Questions like these are typically easy to answer, because humans have a good sense of how long ago specific events happened. Research involving animals, such as food-caching birds, has revealed that they, too, can remember the “when” of past events. So far, little is known about the neuronal mechanisms that enable remembering the “when”. In this study, we explore systematically concrete hypotheses of how neural circuits, synaptic plasticity and ongoing restructuring of neural circuits allow remembering the “when”. We propose experiments

had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing interests:** The authors have declared that no competing interests exist.

that could be done with humans or animals to discriminate different biologically plausible mechanisms for remembering the “when”.

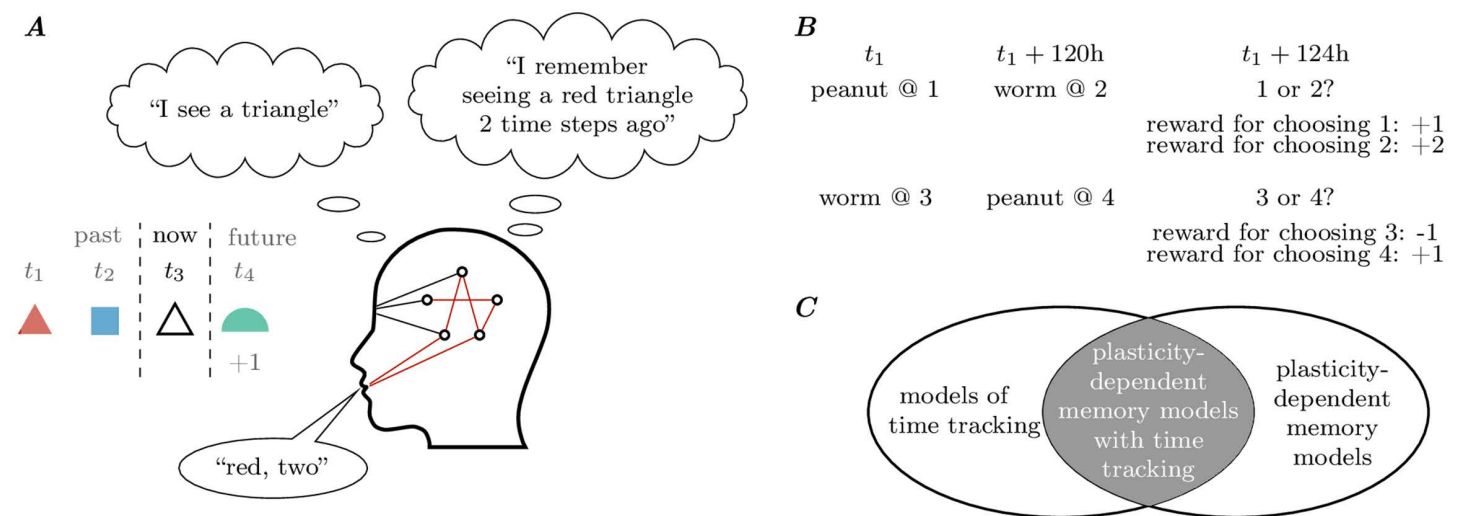
## 1. Introduction

Humans and animals track temporal information on multiple timescales, to estimate, for example, the location of a sound source based on millisecond time differences of sound arrival at the two ears, the interval duration between the perception of lightning and thunder, or the days, months and years that have elapsed since an autobiographical event took place [1–9]. For autobiographical memories, recall of the “when” information is often an explicit and conscious reconstruction-based process [10], for example, “we went to Turkey the year my sister got married, she is five years older than me, got married at the age of 30, and I am now 37 years old, so this must have been 12 years ago.” However, even without explicit reconstruction, healthy human adults usually have a good sense of whether a recalled event happened yesterday, a year ago or decades ago, and there is evidence for automatic processes, in particular in young infants without a fully developed episodic memory system [11–13]. Also corvids, rodents, and other species with an episodic-like “what-where-when” memory can remember the time of past events on timescales of days to months [13,14].

Whereas on timescales from milliseconds to minutes, multiple mechanisms based on changing neuronal activity patterns are known to support accurate interval timing [4–9], little is known about the neuronal mechanisms that support the recall of the time of past events on much longer timescales. Memories on these timescales are likely to rely on long-term synaptic plasticity [15] and possibly on systems consolidation [16].

Multiple research communities have developed models of episodic memory to explain recall of past events [17]. These models focus on different aspects, such as replicating data from laboratory-based memory tasks, such as free or serial recall of lists (reviewed in [18]), embedding episodic memory in cognitive architectures (reviewed in [19]), developing attractor neural networks consistent with anatomical and physiological knowledge of the hippocampal formation (reviewed in [20]), or explaining systems consolidation (reviewed in [16]). However, little work has been devoted to understanding specifically how humans and animals remember the time of past events, although attempts at categorizing different theories of the time of past events have been made [10].

Here, we study from a theoretical perspective different neural mechanisms that enable *automatic* reconstruction of the time of past events on long timescales. Whereas our theoretical considerations about the representation of temporal information are relevant for tracking time on any scale, we focus, in particular, on settings used to investigate episodic-like memory, where a stream of sensory inputs on a timescale of days or months is perceived by an organism that can recall past events and respond with some actions (Fig 1A). As an example of the abstract setting



**Fig 1. A simple setting to study episodic-like memory.** **A** We consider discrete sensory streams, such as perceiving a red triangle at time  $t_1$  and a blue square at time  $t_2$ . In all illustrations we use color, shape and time as abstract analogies of the “what”, “where” and “when” of specific events, respectively. The time points  $t_1, t_2, \dots$  are not necessarily equally spaced and may be separated by hours or days. At time  $t_3$ , the white triangle first triggers activity in the brain that corresponds to the perception of a white triangle and, second, activity that corresponds to remembering the red triangle, including the information of how long ago the red triangle was perceived. An action is performed upon memory retrieval, such as saying “red, two”. In the next time step, a new stimulus can be given, together with a reinforcement signal (+1). Our goal is to find neural network dynamics and synaptic plasticity rules that change the connections between neurons (red lines) such that the sensory stream can be remembered, and action selection rules that depend on the recalled event can be learned. **B** For example, food-caching birds can learn to retrieve caches with fresh worms, and ignore caches with old worms [14]. In trials, where they cache peanuts first (at a given location, indicated by “@ 1”), and worms 120h later, they can retrieve fresh worms (which they prefer over peanuts; indicated with reward +2). In trials where they cache worms first, the worms are unpalatable at retrieval (indicated by reward -1), but the peanuts are still fresh. Clayton et al. [14] showed convincingly that the birds’ ability to learn to ignore caches with old worms depends on memory of the “what” and “how long ago” of caching events. **C** Many models of time tracking in the milliseconds to seconds range, such as pacemaker-accumulator models or state-dependent networks [6], do not depend on (long-term) synaptic plasticity, but on changing neural activity. Most existing models of plasticity-dependent memory, such as models of the hippocampus [20], focus on storage and retrieval of memories (the “what” and the “where”), but not on the age of memories (the “when”). In this work, we investigate plasticity-dependent memory models that keep track of time on long timescales from days to years (indicated by the gray area).

<https://doi.org/10.1371/journal.pcbi.1014575.g001>

described in Fig 1A, one may think of experiments where food-caching animals learn to retrieve from caches they made the same day and ignore caches they made a few days ago (see Fig 1B and Table 1). For memory tasks on this timescale, it is unlikely that some kind of persistent neural activity bridges the gap between storage and recall. Instead, it is commonly believed that memories are stored over long timescales in an activity-silent manner: synaptic connections are altered through Hebbian plasticity mechanisms [15,32] at the moment when an event is experienced. These connections are later reactivated by stimulating neurons upstream of the altered connections, thereby transforming dormant memories back into neural activity patterns.

Our goal is to explore possible Hebbian mechanisms that make it possible to decode the age of such dormant memories at the moment of retrieval (Fig 1C). As the neural mechanisms underlying the retrieval of the age of dormant memories are largely unknown and may vary across different species, we do not want to limit ourselves, a priori, to specific classes of models; rather we aim at spanning the space of possible mechanisms. We discuss in detail some representative examples and propose in simulations specific behavioral experiments that make it possible to distinguish between different model classes. Importantly, and going beyond behavioral predictions, we also show that detailed neural and synaptic recordings are needed to identify the exact mechanisms that allow retrieval of the “when” of past events.

**Table 1. Examples of non-verbal behavioral experiments to investigate memory for the “when” of past events.**

| species  | timescale     | description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| birds    | days          | Birds cache different types of food (what) in different locations (where), at different moments in time. The experimenter changes the availability or the palatability of food items, following rules that depend on the age of the cache and the type of food. Birds have access to their caches after given durations (when). Results: after few training cycles, they avoid inspecting caches where food would be unavailable or unpalatable, which indicates that they can learn behavioral rules that depend on the what-where-when of past caching events [14,21–25] (see also Fig 1B).                                       |
| rats     | hours–days    | Rats explore different arms of a maze (where), containing different types of food (what). The experimenter changes the availability or the palatability of food items, following rules that depend on the type of food item, and when or how long ago the rats explored the maze. Rats have access to the maze after given durations. Results: after training, they prefer arms where they can expect their favorite food [26–28].                                                                                                                                                                                                  |
| apes     | minutes–hours | Great apes observe the experimenter hiding frozen juice and non-perishable food items (e.g., grapes) at different locations (what, where). Apes can choose between the different locations after short or long retention intervals (when). If they chose the location of the frozen juice after the short retention interval, they would get the food; but, if they chose the location of the frozen juice after the long retention interval, when it has melted, they would not get the food. Results: some individuals seem to learn that their preferred frozen juice is available only after short retention intervals [29,30]. |
| children | minutes–hours | Same task with perishable and non-perishable food items as with the apes [31].                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

<https://doi.org/10.1371/journal.pcbi.1014575.t001>

## 2. Results

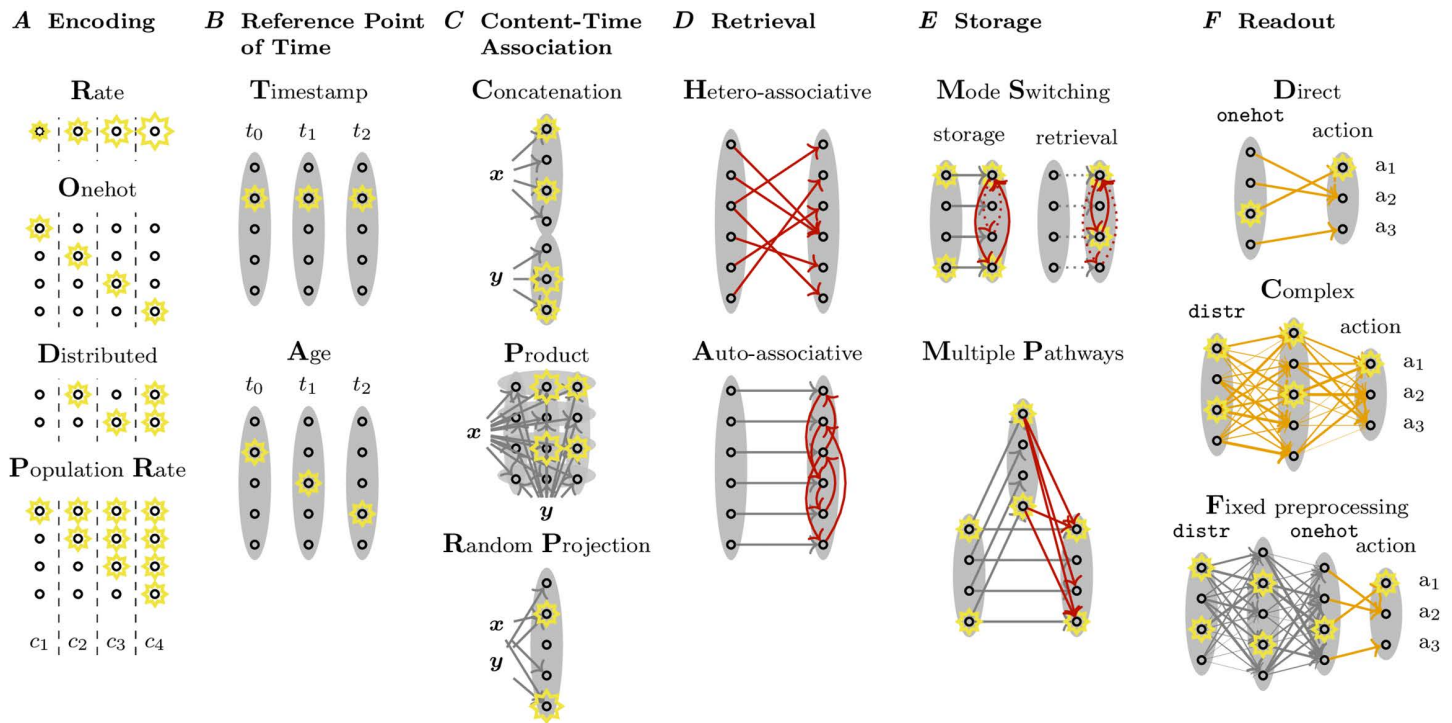
### 2.1. Representing information: The space of possible codes

The activity in a network of neurons can represent a memory in multiple ways. For simplicity, we assume that the “what”, the “where” and the “when” of each memory are elements of discrete sets, such as the sets of colors, shapes and time points in Fig 1A. The value of such discrete variables can be represented with, (i) the firing rate of a neuron (*rate*), (ii) the identity of an active neuron within a group of neurons (*onehot*) or (iii) the distributed activity pattern in a group of neurons (*distr*; see Fig 2A and section “Formal Description of Codes”). For example, in a rate code of color, the sight of red and blue objects evokes different activity levels in the same neuron, whereas in a one-hot code, different neurons are tuned to different colors. A strict rate code with a single neuron or a strict one-hot code, where a given stimulus feature activates a single neuron, is an idealization that is unlikely to be found in any brain. Instead, stimulus features may be represented by a distributed code, where multiple neurons become active, when perceiving the color “red”, for example. However, certain distributed codes can be reduced to rate or one-hot codes by summing the activity of subsets of neurons. Trivial examples are redundant rate or one-hot codes with groups of identical neurons. Another example is the population rate code (*poprate*), where the number of active neurons encodes the value of a variable. We use the term “distributed code” only when such a reduction by summation is impossible.

**2.1.1. Representing time: Timestamp and age codes, internal and external zeitgeber.** Information about the “when” of an event can be represented by any code discussed above, as soon as a reference point for measuring time is defined. We distinguish timestamp and age representations (Fig 2B).

In *timestamp* representations, time is measured relative to a fixed reference point in time. The fixed reference point could be the birth of an individual and the “when” information of an event could be represented as “5 months since birth”. In timestamp representations of time, the neural activity representing the “when” information during recall of a given event is always the same, no matter when recall happens; this neural activity code can thus be seen as representing a timestamp attached to each memory. Importantly, we do not assume that this timestamp representation encodes literally the date and time of an event. In fact, any neural activity pattern can be a timestamp, if the time of a given event can be inferred from it and it does not change with the age of the memory.

In contrast, in *age* representations, time is measured relative to changing moments in time. The changing reference point could be the current moment in time and the “when” information of an event could be represented as “8 months



**Fig 2. Several distinct “what-where-when” memory systems can be constructed by combining specific choices of the encoding, reference point of time, association, retrieval, storage and readout schemes.** **A** The elements  $c_1, c_2, c_3, c_4$  of a set  $C$  can be encoded, for example, (from top to bottom) in a rate code, one-hot code, distributed code or a population rate code (see Formal Description of Codes). **B** In timestamp memory systems, time is measured relative to a fixed reference point and the “what-where-when” information of a specific event is, therefore, encoded with the same activity pattern at storage  $t_0$ , and at any time of recall  $t_1$  or  $t_2$ . In age memory systems, the retrieved memory pattern of a given event changes with time, because time is measured relative to a changing reference point. **C** Information  $x$  and  $y$ , about, e.g., the content and the time of an event, can be associated in multiple ways; for example, with concatenation, or non-linear mixed codes, such as the (outer) product, or random projection code. The two-dimensional arrangement of the twelve neurons in the product code is for visualization purposes; they could also be arranged in a vector with 12 elements. **D** Memory retrieval can be based on hetero-associative recall with learned feed-forward synaptic connections (red) or on auto-associative recall with learned recurrent synaptic connections (red). **E** Separation of storage and retrieval can be achieved with external modulators that switch the network mode from being input-driven in storage mode, to being recurrently driven in retrieval mode. Alternatively, multiple pathways allow a separation of storage and retrieval through different signal propagation properties along the different pathways. **F** The difficulty of learning flexible rules depends on the code. Direct readout: For one-hot coding, any rule is learnable with direct connections to action neurons. Complex readout: Learning arbitrary rules based on distributed or rate coding can be achieved with plastic connections in multilayer perceptrons or complex readout with fixed preprocessing: with hard-wired transformations into, for example, one-hot codes that allow flexible learning. See Table 2 for an overview of the models that can be constructed with different combinations.

<https://doi.org/10.1371/journal.pcbi.1014575.g002>

ago”. In contrast to timestamp representations of time, age coding implies that the neural activity during recall of a given event is not the same at different moments, because the “when” information depends on how much time has elapsed between storage and recall; this neural activity code can thus be seen as representing the age of each memory. The distinction between timestamp and age coding is similar to an important difference in McTaggart’s A and B series [41]: in age coding the representation of the “when” of an event changes continually, like the position of an event in the A series, whereas in timestamp coding the representation is constant, like the position of an event in the B series. An example of an age code is shown in Fig 2B, where the elapsed time between storage and recall is represented by the location of the activity peak.

We use the term *zeitgeber* to refer to the process that generates either timestamps or changes the age code. Unlike a clock, that is synchronized with physical time, a *zeitgeber* may drive the representation of time with variable speed that

depends, for example, on the frequency of events that are worth memorizing. The distinction between physical time and the temporal code produced by the zeitgeber is reminiscent of the contrast between physical time and “subjective time” explored, for example, in the philosophical works of Henri Bergson and Maurice Merleau-Ponty. Note, however, that the “timestamp” or “age” code generated by the zeitgeber is – in principle – objectively measurable, and does not need to coincide with the subjective perception of time.

The zeitgeber can either be an internal process that runs almost autonomously inside the time-perceiving agent or it can depend mostly on the agent’s interaction with the external world. Internal zeitgebers can be any biological process inside the agent with a fairly stable time constant such as ramping or decreasing synaptic strengths, neurogenesis, spine turnover, the circadian rhythm (in the absence of exposure to sunlight) or even changes in satiety, thirst or tiredness level. Examples of external zeitgebers are processes such as the ticking of a clock, the day-night cycle or the change of seasons; also processes that involve the agent’s actions, such as changes of context (leaving or entering a house) or changes of the main activity (switching from working to eating lunch), could act as external zeitgebers.

## 2.2. Associating information: How to combine what, where and when

To remember everything about a given event, the “what”, “where” and “when” information needs to be associated in some way. In the following, we assume the “what” and the “where” are given as a content variable in some code and focus exclusively on how the content is associated with the “when” information. The question of associating the “what” (and “where”) with the “when” becomes the question of building an “association function” that produces a neural activity pattern in response to the “when” information on one side and the encoded content (“what” and “where”) information on the other side (see section “Formal Description of Association Schemes”).

The number of possibilities to write down such an association function is huge, even if we restrict ourselves to those functions that do not “lose” any information, in the sense that the content and “when” information can be faithfully reconstructed from the neural activity pattern during recall. In the following we focus on three specific examples of association functions: concatenation, (outer) product and random projection codes (Fig 2C).

In a *concatenation* code, neurons can be split into two separate groups: one representing the “when” information and the other one the content information (cf. Fig 2C). Closely related is a *linear mixed code*, where such a split is not directly possible, because single neurons contribute to the representation of both content and “when” information, but a linear transformation of the neural population activity would make it possible to represent the content and “when” information in a concatenation code. An example of a *non-linear mixed code* is the *product code*, where the neural population activity is given by the outer product of the content and the “when” code. This product code is a special case of tensor product variable binding [42]. Such a product code requires, in general, more neurons than a concatenation code: if content and “when” information could be represented separately by  $N$  and  $M$  neurons, respectively, their association with a product code requires  $N \times M$  neurons, whereas  $N + M$  neurons would be sufficient for a concatenation code. Other non-linear mixed codes can be constructed with (circular) convolutions [43,44] or *random projections*, where the activity of each neuron in a group depends non-linearly on a randomly weighted mixture of content and “when” information.

The way a neuronal population represents the association of content and “when” information has important implications for the readout of retrieved memories, as we will discuss in the next section and demonstrate in the section “Experimental Predictions”.

## 2.3. Storage, retrieval and readout of memories

So far, we considered only the representation of information. However, the description of a memory system is incomplete without a characterization of the storage, retrieval and readout mechanism.

**2.3.1 Retrieval.** In the field of neural networks, memory retrieval is typically implemented with hetero- or auto-associative memories (Fig 2D). In both hetero- and auto-associative networks, the output activity of a neural network in

response to an input cue represents the retrieved memory. In a hetero-associative memory, retrieval is performed in a single step whereas a recurrent auto-associative network requires convergence to a fixed point [45]. However, a single update step is often sufficient to almost reach the fixed point and retrieve a memory almost perfectly, in particular in kernel memory networks [46].

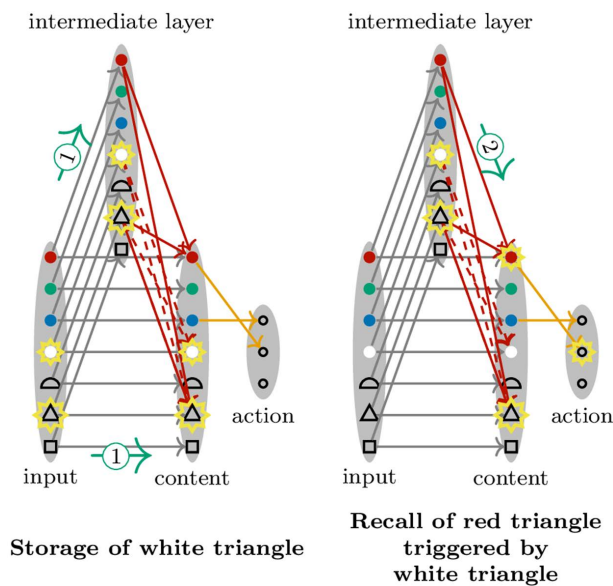
**2.3.2. Separation of storage and retrieval.** Suppose “red triangle” has already been stored at time step  $t_1$ . At time  $t_3$ , the input “white triangle” triggers recall of the stored memory, i.e., the neural code for the “when” information  $t_1$  together with the content information “triangle” and “red” should be accessible at time  $t_3$  and become active while the remembered event “red triangle” is retrieved from memory (retrieval phase). At the same time step  $t_3$ , however, it must also be possible to store the new event “white triangle”, as an event that happens at time  $t_3$  (storage phase).

We distinguish two basic mechanisms to separate storage and retrieval: *mode switching* and *multiple pathways* (Fig 2E). A prominent example of mode switching is the Hasselmo et al. model [35], of a periodic process in which external input drives the memory network during the storage phase and recurrent connectivity in the memory network dominates during the retrieval phase. The theta rhythm in the hippocampus, or some modulatory factors, such as neurotransmitters, could drive such a periodic process [35,47], although the plausibility of this mechanism across species is debated [48,49].

Alternatively, the separation of storage and retrieval could be implemented with separate pathways. For example, Treves and Rolls [33] suggested that the strong connections of the mossy fiber system from dentate gyrus (DG) to the CA3 hippocampal network clamp the CA3 neurons to an input-driven activity pattern during storage, whereas the perforant path to CA3 is used to initiate the retrieval process. In hetero-associative retrieval, strong clamping may not be needed; instead, different signal transmission delays along different pathways may be sufficient to separate storage from retrieval. Suppose multiple pathways exist between two groups of neurons, for example the “input-content” pathway and the “input-intermediate-content” pathway in Fig 3, and information travels at different speeds along them. Then stimulus-driven activation through one pathway could be used for storage (“input-content” pathway in Fig 3) and through the other pathway for retrieval (“input-intermediate-content” pathway in Fig 3). In the simulated models in section “Experimental Predictions” we use this mechanism to distinguish storage and retrieval phases. In contrast to mode switching with 4–8 Hz theta frequency, multiple pathways allow for a quicker alternation between storage and retrieval on a timescale of a few milliseconds. If plasticity were to be induced only during the storage phase, this would require mechanisms that are rather precisely timed. It remains to be seen, whether this is possible with fast modulation of Hebbian learning rules with a third factor [50–54]. However, ongoing plasticity may not be harmful, in particular with prediction-error reducing learning rules [55,56].

**2.3.3. Behavioral readout.** Once a previously stored activity pattern is retrieved, it can trigger some behavioral output. In contrast to computer memory, where recall success can be measured by the number of bits lost between storage and recall, successful retrieval of a memory in humans and animals is usually inferred from some behavioral output, which may have a different representation from the input that led to the formation of the memory (e.g., visual input and vocal output in Fig 1A). Therefore, we must include a discussion of action selection that subjects perform in response to retrieved memories.

Behavioral rules control which action to perform in response to a specific retrieved memory. Whereas learning is often irrelevant for memory tasks with humans – as the behavioral rule is usually instructed – it is crucial for memory experiments with animals. How easily different behavioral rules can be learned depends on the representation of recalled memories. This can be used to design experiments that discriminate between different kinds of “what-where-when” memory systems, as we will show in section “Experimental Predictions”. For a strict one-hot code, for example the product of one-hot codes  $\text{onehot}(\text{content}) \otimes \text{onehot}(\text{age})$ , any behavioral rule that maps content and age of a recalled event to a given action can be learned with *direct readout*, i.e., plastic connections from the layer of recalled activity to action neurons (Fig 2F). For distributed or rate coding, direct readout allows learning of some rules, but *complex readout* is needed to learn any rule. For example, the notorious XOR rule [57], where a certain action is taken if and only if two input neurons



**Fig 3. Example of a Storage and Retrieval Mechanism with Synaptic Delays.** Signals take more time to travel in long pathways with multiple intermittent synapses than in short pathways, because signal transmission across chemical synapses takes time. Therefore, multiple pathways of different lengths between two groups of neurons can be used to separate storage and retrieval phases. During storage, the input drives the activity in the intermediate layer; the content layer receives input through the direct “input-content” pathway (indicated by green arrows with label 1). To simplify the exposition, the outgoing connections of the content layer are one-to-one in this example, but the same principle works in less artificial settings with random connectivity (see Fig 7). Connections between the intermediate and the content layer (red dashed arrows) are selected for growth with a Hebbian plasticity rule. Shortly thereafter, because of more synaptic delays along the longer pathway, the content neurons receive input through the “input-intermediate-content” pathway, i.e., the intermediate layer is the main input of the content layer (green arrow with label 2). Already grown connections (red arrows) enable recall of previous events. The actual growth of the connections selected in the storage phase (dashed red arrows) is not instantaneous and does therefore not interfere with the recall phase. Once the content neurons received input through the “input-intermediate-content” pathway, the content layer drives the action selection through weights that implement some learned rule (orange arrows).

<https://doi.org/10.1371/journal.pcbi.1014575.g003>

are jointly active or jointly inactive, cannot be learned with direct readout, but it can be learned with a multilayer perceptron (Fig 2F). Learning all connections in a multilayer perceptron can be achieved with the backpropagation algorithm or biologically plausible variants thereof [58–60], but it is rather slow if learning happens in an online fashion, where each example is used just once. An alternative is to rely on fixed weights in most layers to transform the input into a useful feature representation and quickly learn flexible mappings with biologically plausible Hebbian plasticity in the last layer [61] (Fig 2F).

**2.3.4. Synaptic plasticity.** Long-term synaptic changes are presumably involved in memorization in the storage phase, in learning actions to indicate successful retrieval, and in reflecting the passage of time in age representations of time.

If pre- and postsynaptic neurons are jointly active during the storage phase, one-shot Hebbian synaptic plasticity (e.g., [62]) is sufficient to memorize and generate a trace of the event in the memory system. Behavioral rules that depend on the content and the age of recalled memories could be learned with neoHebbian synaptic plasticity [50–54], where jointly active neurons generate an eligibility trace that is modulated by a signal that arrives up to a few seconds later. This modulating signal could communicate the reward received after a successful action.

For age representations of time, synaptic turnover (growth or decay) could reflect the passage of time [63–67]. If the rate of synaptic pruning depends on the postsynaptic neuron, neurons with a low pruning rate would be active when young and old memories are retrieved, whereas neurons with a high pruning rate would only be active when young

memories are retrieved. This would allow for a straightforward readout of the age of memories, as demonstrated in the examples below. A broad distribution of neuron-dependent pruning rates over timescales from hours and days to months and years would enable approximate recall of the age of memories (see also Fig 8). While such a neuron-dependent pruning rate is simple and plausible, we are not yet aware of clear experimental support for it.

For timestamp representations of time, the synaptic changes for memorization and behavioral learning are sufficient. Although this is an appealing advantage of timestamp representations of time, it comes at the cost of increased complexity in computing the age of recalled memories, because a representation of the current moment in time needs to be compared with the time of storage of the recalled event.

## 2.4. Examples of episodic-like memory systems

With four encoding schemes (Fig 2A), two ways of representing time (Fig 2B), three codes for associating content and time (Fig 2C), two memory retrieval mechanisms (Fig 2D), three readout architectures (Fig 2F), and two storage-retrieval mechanisms (synaptic delays Fig 3 or periodic processes, such as the one proposed by [35]) we have  $4 \times 2 \times 3 \times 2 \times 3 \times 2 = 288$  concrete hypotheses about Hebbian “what-where-when” memory systems. This is a lower bound, because even more association, storage-retrieval and readout mechanisms are conceivable. The number 288 looks daunting. However, in this section we discuss in more detail six specific examples that are representative of the different kinds of models (Table 2). As storage, retrieval and readout are general aspects of synaptic memory systems, irrespective of time, we selected our examples primarily to highlight advantages and disadvantages of different encodings of time, content-time associations, and timestamp versus age organization of time. Detailed mathematical descriptions of these models can be found in section “Mathematical Description of the Models”.

**2.4.1. Timestamp tagging with auto-associative retrieval and complex readout.** Time tagging models are characterized by a concatenation code that combines one subgroup of neurons that represent content information (“content” in Fig 4) with another subgroup of neurons that represent “when” information (“tag” in Fig 4).

In the spirit of (temporal and random) context models [36–38], the “when” information could be given implicitly by activity patterns that encode context (“now” in Fig 4). This context could be a trace of recent observations of states that change on different timescales, such as emotional states, the presence of certain conspecifics, ambient temperature or the weather. On an implementation level and for timescales from seconds to hours, it could be given by so-called hippocampal time cells [9,69–74] that fire at successive moments in temporally structured experiences. In the following, we call this the “Context-Tagging model”. At the moment of storage, this context information is bound together, e.g., by a Hebbian plasticity rule, with the specific event under consideration (Fig 4B). If an event triggers the recall of an earlier event (Fig 4C), the associated context, i.e., the “when” information, is also recalled. A comparison of the current context with the recalled context may allow a rough estimate of the age of the recalled memory (cf. contextual overlap theory, [10]). Because the estimation of age from the comparison of two context-related activity patterns does not, in general, induce a linearly separable problem, a complex readout network with at least one hidden layer is required, to learn arbitrary readout rules.

**2.4.2. Age tagging models.** Other tagging models can be constructed with one-hot coding or population-rate coding for the age of memories (Fig 2A). In the Onehot-Age-Tagging model, storage leads to the formation of a synaptic connection to the first tag neuron (Fig 5A and Fig 5B). A systems consolidation mechanism can change the representation of the memory by growing new synapses and pruning old ones [75,76]. Such a mechanism could implement a one-hot time code, where the identity of the activated tag neuron during recall indicates the age of the memory (Fig 5C). Note that, for simplicity, we illustrated the age tagging mechanism using single neurons and single synapses per memory age; one-hot coding should be understood as described in Representing Information: the Space of Possible Codes.

In the Poprate-Age-Tagging model, storage leads to the formation of many synaptic connections to several “tag” neurons (Fig 5D). These connections are pruned at different moments in time. Therefore, the number of activated tag neurons

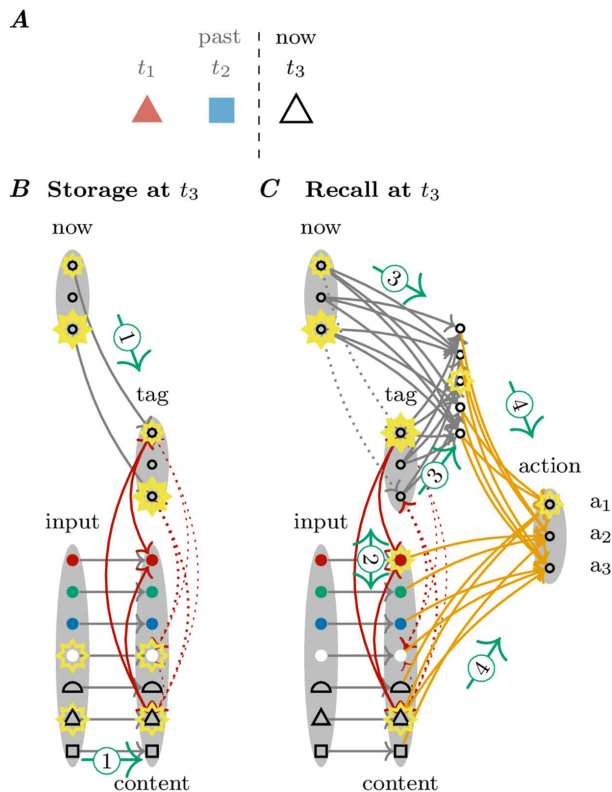
**Table 2. Comparison of Computational Models.** The labels in columns “Encoding” to “Readout” correspond to the labels used in Fig 2. The label used in column “Memory Location” indicates whether the memory is stored synaptically (S), or in the activity pattern (A) of some neurons. A dot “.” indicates that the model is agnostic about this feature.

| name                     | Encoding | Time Reference | Association | Retrieval | Storage | Readout | Memory Location | comments                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------------|----------|----------------|-------------|-----------|---------|---------|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Context-Tagging          | D        | T              | C           | A         | MS      | F       | S               | see Fig 4                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Onehot-Age-Tagging       | O        | A              | C           | A         | MS      | D       | S               | see Fig 5C                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Poprate-Age-Tagging      | PR       | A              | C           | A         | MS      | D       | S               | see Fig 5E                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Age-Organization         | O        | A              | P           | H         | MS      | D       | S               | see Fig 6A                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Poprate-Age-Organization | PR       | A              | P           | H         | MS      | D       | S               | see Fig 6B                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| Random-Pruning           | PR       | A              | RP          | H         | MP      | D       | S               | see Fig 7                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| Treves and Rolls         | .        | .              | .           | A         | MP      | .       | S               | A model of associative learning and recall in the hippocampus without explicit tracking of time [33,34].                                                                                                                                                                                                                                                                                                                                              |
| Hasselmo                 | .        | .              | .           | A         | MS      | .       | S               | Example of a mode-switching storage model of the hippocampal circuit using theta cycles, without explicit tracking of time [35].                                                                                                                                                                                                                                                                                                                      |
| TCM                      | D        | T              | C           | H         | .       | .       | S               | The age of memories could be retrieved using, either, the current context and reconstructing the contextual overlap, or                                                                                                                                                                                                                                                                                                                               |
|                          | R        | A              | C           | H         | .       | .       | A               | the explicit exponential decay of memory traces in the context tags [36–38].                                                                                                                                                                                                                                                                                                                                                                          |
| OSCAR                    | D        | T              | C           | .         | .       | .       | S               | A clock-context model similar to TCM, except that the context tags are constructed from oscillators of different frequencies, instead of exponentially decaying memory traces [39].                                                                                                                                                                                                                                                                   |
| TILT                     | D        | A              | P           | .         | .       | .       | A               | Instead of exponentially decaying context tags with <i>one</i> given timescale as in TCM, this model maintains for each feature dimension <i>multiple</i> exponentially decaying context tags at different timescales [40]. Note that TILT is embedded in a network with associative, synaptic-plasticity-dependent learning. The memory of the age of the memory is, however, encoded in the decaying activities and linear transformations thereof. |

<https://doi.org/10.1371/journal.pcbi.1014575.t002>

during recall is indicative of the elapsed duration between storage and recall, and a simple readout mechanism that compares the sum of tag neuron activities to a fixed threshold – each action neuron in Fig 5E may have a different threshold – can implement behavioral rules that depend on the age of memories. We emphasize, however, that certain rules are impossible to learn with this simple readout mechanism (see, e.g., Fig 11).

In contrast to the Context-Tagging model with a timestamp code (Fig 4), the representation of time changes in the Onehot-Age-Tagging model and the Poprate-Age-Tagging model, because the activity pattern of the tag neurons during retrieval depends on the elapsed time since storage. Age tagging allows simple readout learning of age-dependent behavioral rules, in particular, when the order of pruning synaptic connections is fixed, i.e., whenever it is possible to order pairs of tag neurons  $i$  and  $j$ , such that connections to tag neuron  $i$  are consistently lost earlier than simultaneously grown connections to tag neuron  $j$ . This could be implemented with neuron-dependent pruning rates as described in section Synaptic Plasticity. One can even prove (see Equivalence of One-Hot coding and Deterministic Population-Rate coding) that



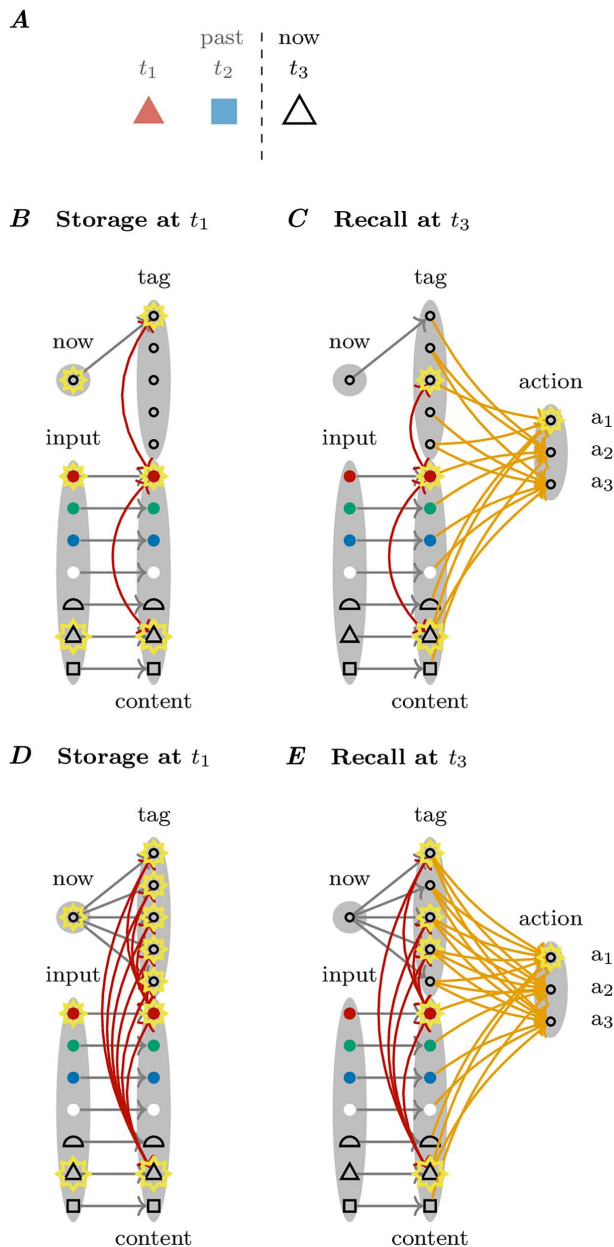
**Fig 4. The Context-Tagging model: Timestamp Tagging with Auto-Associative Retrieval and Complex Readout.** **A** The events “red triangle” and “blue square” were observed at times  $t_1$  and  $t_2$ , respectively. At time  $t_3$  the event “white triangle” is observed. **B** During storage, the input drives the content layer, and the current context (activity in layer “now”) drives the “tag” neurons (green arrows with label 1) such that content and context can be bound together (dotted red lines). **C** Subsequently, the previously grown synaptic connections (red lines) allow auto-associative recall of the event “red triangle”. During recall, the tag neurons are no longer driven by input from the “now” neurons, but, through auto-associative recall (green arrow with label 2), the activity of the tag neurons encodes the context at time  $t_1$ . The comparison of the current context (represented by the “now” neurons) with the recalled context (represented by the “tag” neurons) allows the readout network to estimate the age of the retrieved memory (green arrows with label 3). This comparison could be implemented in different ways, e.g., as a vector subtraction, or in a similar way as in linear prediction error networks [68]. Consequently, behavioral rules that depend on the age or the content of the recalled memory can be learned (orange weights; green arrows with label 4).

<https://doi.org/10.1371/journal.pcbi.1014575.g004>

this population rate code leads to the same action selection policy as a model with one-hot coding of “when” information, if the readout connections follow a special synaptic plasticity rule.

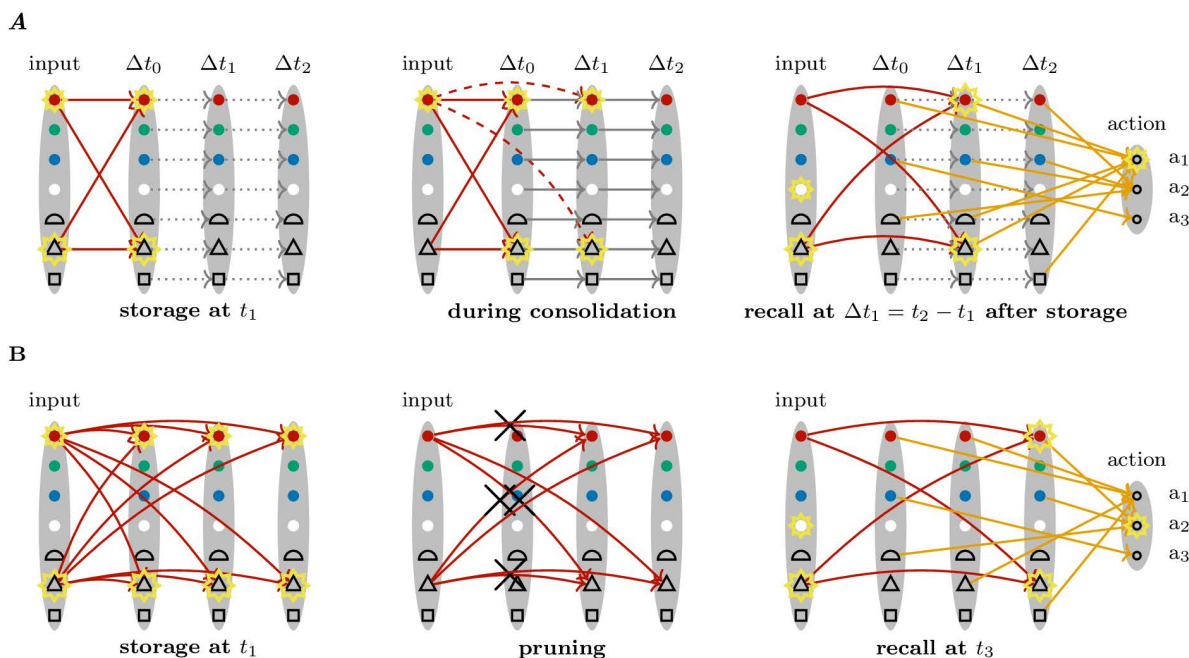
**2.4.3. Age organization with systems consolidation.** Instead of using concatenation, as in tagging models, the content and “when” information could be associated with the (outer) product operation (cf. Fig 2). Variants of models constructed in this way include the chronological organization of memories [10], which is also known as a shift register in engineering [77]. For example, this approach is implemented in the TILT model [40], an activity-based model of time-tracking memory (see also Table 2).

The chronological organization is most obvious, when arbitrarily encoded content information and one-hot encoded age information are associated with the product operation. We call this the Age-Organization model. In this case, the configuration of active neurons during recall of a given event depends on the time of recall (Fig 6), similarly to how suitcases on a conveyor belt change their position relative to a fixed observation point. Because of the product operation, there are multiple groups of neurons that code for content (e.g., the groups  $\Delta t_0$ ,  $\Delta t_1$ ,  $\Delta t_2$  in Fig 6A), but the neurons in only one of these groups become active during recall of a specific event. The identity of the active group encodes implicitly the age



**Fig 5. Age tagging models.** **A** We consider the same sequence of events as in Fig 4A. **B** In the **Onehot-Age-Tagging** model, the first tag neuron is activated by the “now” neuron, which is only active during the storage phase, such that Hebbian one-shot learning establishes a connection between the first tag neuron and the active content neurons, when event “red triangle” is perceived at time  $t_1$ . **C** Thanks to a process that prunes some synapses and grows new ones, tag neuron 3 is activated during recall at time  $t_3$ , indicating how long ago the red triangle was observed. **D** In the **Poprate-Age-Tagging** model, connections to all tag neurons are formed during storage. **E** These connections are pruned at different moments in time, such that during recall at time  $t_3$ , fewer connections are present than at time  $t_1$ . The number of active “tag” neurons during recall encodes the elapsed time since storage: if many “tag” neurons are active during recall, the recalled event happened recently and if few “tag” neurons are active, it happened long ago.

<https://doi.org/10.1371/journal.pcbi.1014575.g005>



**Fig 6. Examples of Chronological Organization Models.** **A** In this age model, a systems consolidation mechanism shifts the location where a memory is stored. During storage (left), the red connections are strengthened, whereas the feedforward weights (gray dotted arrows) are inactive. During consolidation (middle), input neurons are randomly active and activity is forward propagated (gray arrows; indirect pathway from input to  $\Delta t_1$ ), such that new, direct-pathway connections (dashed red) can grow between input and  $\Delta t_1$  neurons. Simultaneously, the original weights between input and  $\Delta t_0$  neurons decay, such that after consolidation only the newly grown weights remain. During recall (right), all active neurons in layers  $\Delta t_0$ ,  $\Delta t_1$ ,  $\Delta t_2$ , ... give input to the action neurons (orange connections). **B** Another age model relies on pruning of synapses at different moments in time, similar to the Poprate-Age-Tagging model (Fig 5E). Synapses onto neurons in the first memory layer have a faster decay rate than those onto the last layer. During recall, the number of active neurons across all layers is indicative of the age of a memory: at  $t_1$  more “red” and “triangle” neurons are activated than at  $t_3$ .

<https://doi.org/10.1371/journal.pcbi.1014575.g006>

of the memory: if recall happens some time interval  $\Delta t_1$  after storage, the content neurons in group  $\Delta t_1$  become active, whereas the neurons in other groups become active during recall at other times (Fig 6A).

In contrast to activity-based models of time tracking such as the TILT model [40], a plasticity-dependent model of time tracking with an age code requires growing and pruning of synaptic connections. Similarly to the Onehot-Age-Tagging model, this could be mediated by a systems consolidation process, where new synapses are grown to groups of neurons that code for older memories. For example, in consolidation phases during sleep, randomly activated input neurons could trigger recall of past events in neurons connected to the input by an indirect pathway, thereby creating new direct-pathway connections (Fig 6A). A concrete example for how this could be implemented is given by the parallel pathway theory of systems consolidation [76]. The idea that the location of memorized events changes over time appears also in the multiple trace and trace transformation theory [16,78].

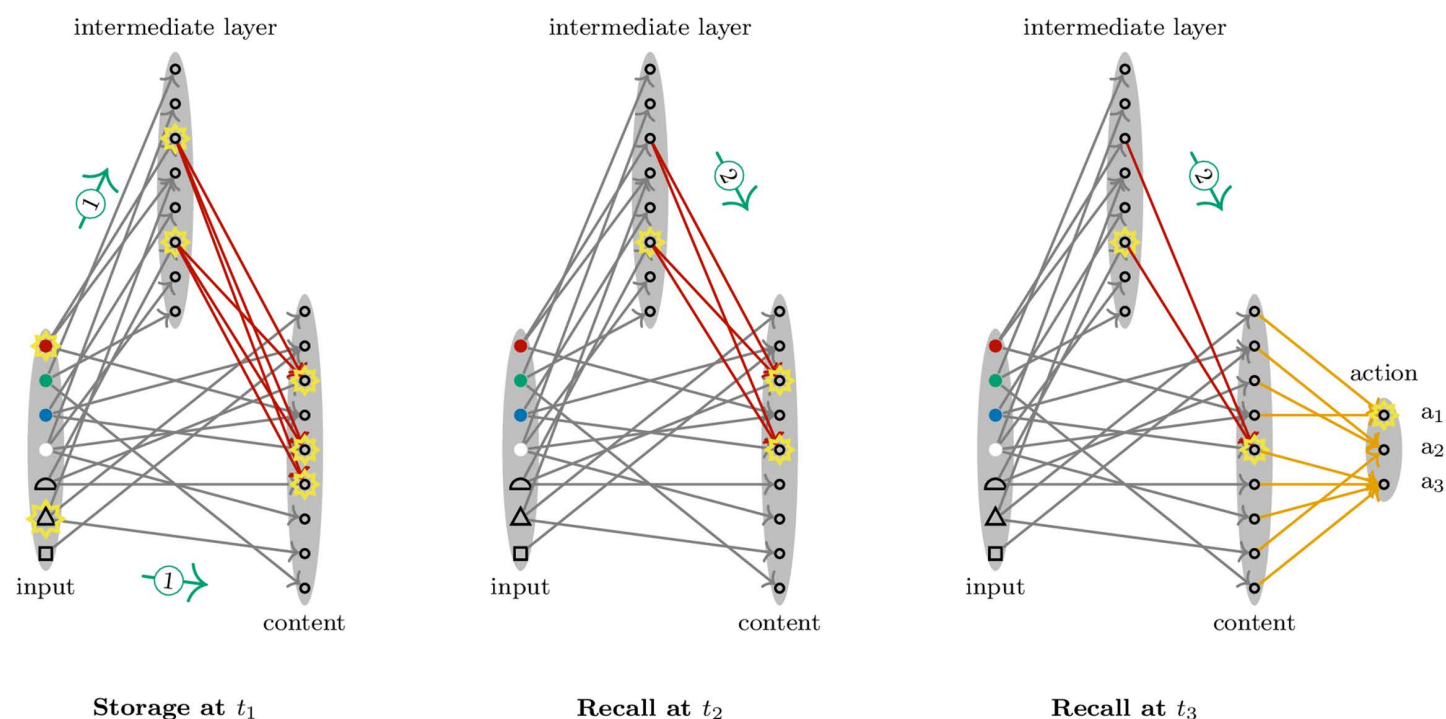
The readout connections do not need to change during systems consolidation, but they make it possible to learn flexible rules. An example is the rule “perform action  $a_1$  if the memory for a red object is younger than  $\Delta t_1$ , and  $a_2$ , if it is older than  $\Delta t_2$ , whereas for blue objects, perform action  $a_3$ , if they are younger than  $\Delta t_1$ , and action  $a_2$ , if they are older than  $\Delta t_1$ ” (see orange lines in Fig 6A). If the time it takes to move memories from one layer to another depends on the age of the memory, such that in early layers memories stay for a shorter amount of time than in later layers, this memory system could approximate a Weber-Fechner law, where the precision of determining the exact age of a memory decreases with its age.

If something like a Age-Organization model is indeed implemented in some species, a fixed number of layers and the indirect pathways (gray arrows in Fig 6A) are probably established during development. An exact one-hot code for time and an exact copy mechanism from one layer to the next seem unrealistic, but approximations thereof with bump-like time coding and linear transformations of the content from one layer to the next would lead to approximately the same behavior as the Age-Organization model.

**2.4.4. Age organization with synaptic pruning.** Another instantiation of a chronological organization model arises when considering the product between content and population-rate encoded “when” information (Fig 6B). This model is similar to the Poprate-Age-Tagging model. In both models, many synapses are grown during storage (storage at  $t_1$  in Fig 6B) and pruned at different moments in time, such that the age of a recalled memory can be decoded from the identities or numbers of active neurons during recall. The chronological organization across the memory system arises from the fact that synapses onto neurons in the first layer of the memory decay more quickly than those in the last layer (Fig 6B).

**2.4.5. Sparse encoding with random synaptic pruning and simple readout.** Despite the frequent appearance of the special association schemes “concatenation” and “product” (Fig 2C) in the literature (Table 2), it is unclear why brains should favor them over other non-linear mixed association schemes [79, 80].

Combining sparse random projections for association (Fig 2C) with synaptic delays for storage and recall (Fig 3) and synaptic pruning (Fig 5E and Fig 6B) leads to the Random-Pruning model (Fig 7). During storage, the input triggers distributed activity patterns in the intermediate and content neurons. Because of the random projections, these activity



**Fig 7. Sparse Encoding with Random Synaptic Pruning and Simple Readout.** The input is sparsely and randomly connected to an intermediate and a content layer (gray arrows). During storage, the content and intermediate layers are driven by the input layer (green arrows with label 1), and Hebbian plasticity connects co-activated neurons (red arrows). Once the activity during recall with the triangle cue has reached the intermediate layer, the neurons in this layer drive the content layer through the red connections (green arrows with label 2; cf. Fig 3). These synaptic connections are pruned after random, postsynaptic-neuron-specific durations, such that during recall at time  $t_2$  more neurons are activated in the content layer than during recall at time  $t_3 > t_2$ . This is an example of a non-linear mixed code of content and time, because the activity of a given neuron in the content layer can mean, for example, “a red triangle was observed at most so-and-so long ago”. The neurons in the content layer link directly to action neurons (orange arrows).

<https://doi.org/10.1371/journal.pcbi.1014575.g007>

patterns encode the input information implicitly, i.e., the neurons in these groups are not necessarily “tuned” to a single feature, such as the redness of an object, but a given neuron may specialize to specific combinations of features and become active, for example, only when a red triangle is shown. During the storage phase, a Hebbian plasticity rule can initiate the growth of synaptic connections between the intermediate layer and the content neurons (storage in Fig 7).

In the model of Fig 7, the sparse random projection code of the content is combined with a population rate code of the “when” information, similarly to the Poprate-Age-Tagging model (Fig 5E) and the chronological organization model with synaptic pruning (Fig 6B): synapses to the content layer are pruned after random durations that depend on the identity of the post-synaptic neuron, such that more neurons become active when recalling a recent event (recall at  $t_2$  in Fig 7), than when recalling an old event (recall at  $t_3$  in Fig 7).

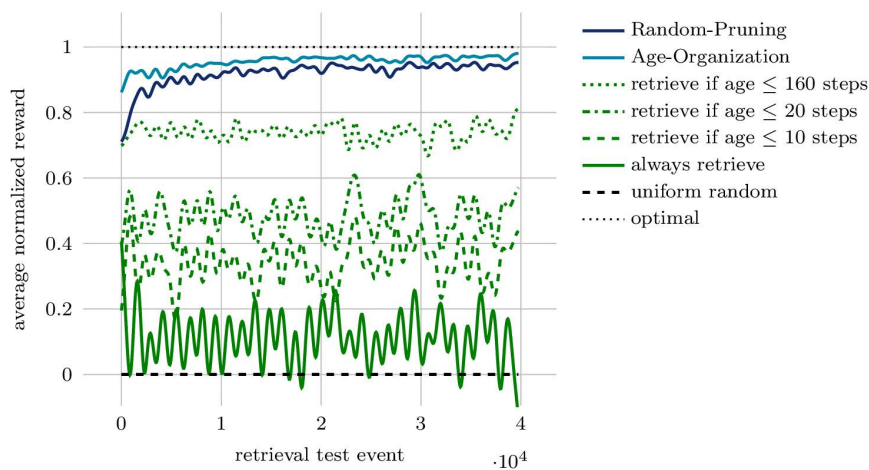
**2.4.6. A what-where-when memory system for food caching animals.** As an example of where such a memory system could be beneficial in natural settings, let us examine food-caching behavior (Fig 8). Scatter hoarders, such as nutcrackers or jays, store small food items in hundreds of locations each season [81]. There is good evidence that retrieval of their own caches is memory-dependent: olfactory cues are unnecessary; visual cues matter; and random search at preferred locations is inconsistent with the observed behavior [81]. As different types of food degrade at different rates, it is beneficial for these animals to keep memories not only of what they cached where, but also of how long ago the caching events happened. In the laboratory, *California scrub-jays* were observed to remember what they cached how long ago, and adapt their behavior, if they learn that certain types of food degrade faster or slower than others [14,22,23,82].

To demonstrate that the memory systems proposed in this article enable good performance in these kinds of settings, we ran simulations with approximately 20,000 caching events of 3 types of food at 1000 possible cache locations (Fig 8; in the abstract setting of Fig 1A this corresponds to 1000 distinct colors and 3 distinct shapes). Location and food type are given as two one-hot coded vectors to the input of the models. Randomly interleaved with the caching events, approximately 40,000 retrieval test events probed the memory system with possible cache locations. Response action  $a_1$  is interpreted as an active retrieval event at the test location (in the wild, a jay may start digging with its bill), whereas action  $a_2$  is interpreted as ignoring the test location (reducing waste of energy and time of an active retrieval event at a location where no food is cached or the food has already degraded). A positive reward for successful retrieval is given, if the retrieval action  $a_1$  is chosen at a location where food of the first type was cached less than 10 steps ago, or food of the second type less than 20 steps ago, or food of the third type less than 160 steps ago. A small negative reward for the wasted effort is given, if the retrieval action  $a_1$  is chosen otherwise. No reward is given for ignoring retrieval with action  $a_2$ .

The Age-Organization model in Fig 8 relies on synaptic pruning after different delays as in Fig 6B. These delays are chosen artificially at 10, 20, and 160 steps, matching the degradation duration of the three types of food. At the beginning of the simulation, the readout weights are initialized such that action  $a_1$  is only chosen, when a caching event at the given test location happened less than 160 steps ago (best baseline condition in Fig 8), i.e., the network reacts to all food types in the same way. Within only a few tens of retrieval test events, a food-type-specialization is learned, and the memory system approaches the optimal policy. Conceptually related, but less artificial is the Random-Pruning model in Fig 8, which relies on sparse fixed random connections to an intermediate layer of almost 5000 neurons, sparse fixed random connections to a content layer of 60 neurons, and synaptic pruning after random, postsynaptic-neuron-specific durations in the range of 1–200 steps, as in Fig 7. The readout weights of this model are initialized such that action  $a_1$  is chosen, whenever a content neuron becomes active during recall, and action  $a_2$  otherwise. Although performing slightly worse than the artificial Age-Organization model specifically designed for this task, the Random-Pruning model performs considerably better than all baselines, and reaches close to optimal performance.

## 2.5. Experimental predictions

To highlight advantages and disadvantages of different systems and explore the limitations of purely behavioral experiments as a tool to learn about how brains remember the “when” of past events, we simulated several “what-where-when”



**Fig 8. A scaled-up version of a variant of the Age-Organization and the Random-Pruning model yields good performance in a simulated food-caching and retrieval setting.** Average normalized rewards were computed with Gaussian smoothing (with standard deviation  $\sigma = 500$  steps), and dynamic rescaling, such that the optimal policy always has reward 1, and uniformly random choices of actions  $a_1$  and  $a_2$  have reward 0. The performances of baseline policies (in green) are computed with full access to the history of caching events. Choosing always the retrieval action  $a_1$  (solid green line) incurs high costs for all locations, where the food item is not retrievable, whereas choosing the retrieval action only, if there was a cache event at the test location within the last 10 steps (dashed green line) misses many opportunities with retrievable items. The best baseline policy without food-type-specificity is to choose retrieval action  $a_1$  whenever at the given test location a food item was cached within the last 160 steps (dotted green line). The Age-Organization and the Random-Pruning model learn quickly to outperform the baseline policies and approach the optimal performance.

<https://doi.org/10.1371/journal.pcbi.1014575.g008>

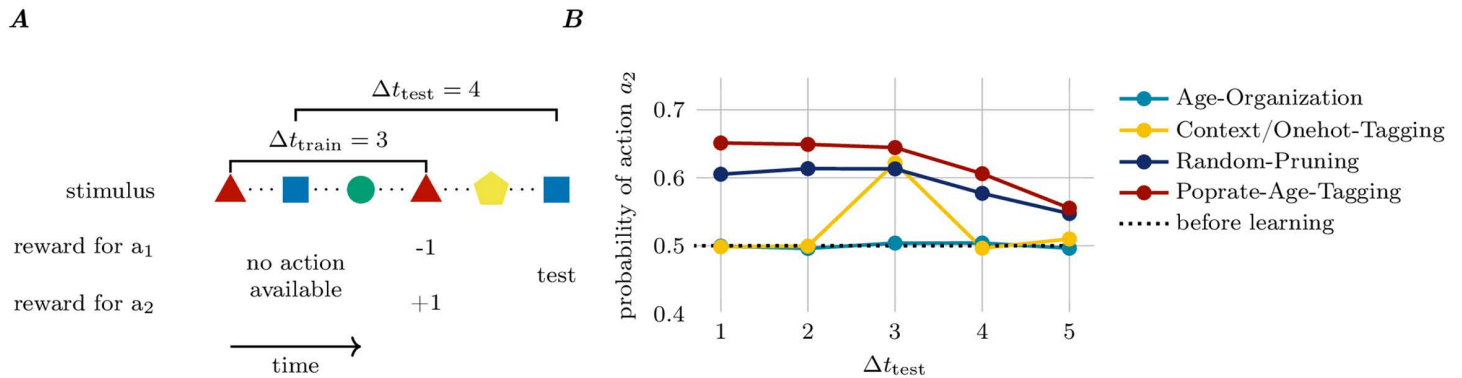
memory models. For these simulations we assume that the subjects have to learn the behavioral rule through reinforcement learning, as is typically the case in experiments with animals.

For the Context-Tagging model we assume a fixed preprocessing to a one-hot intermediate representation of the age of a memory (Fig 4C), which is identical to the one-hot representation of time in the Onehot-Age-Tagging model. Because of their similarities, we do not simulate these two models separately and refer to them as Context/Onehot-Tagging model.

Although the discrimination of some models requires recordings of neural or synaptic dynamics, purely behavioral experiments can provide valuable insights. Suppose, for example, that a subject has learned how to respond to recalling a memory with a certain content and age, such as performing action  $a_2$  when the event “red triangle” is remembered to have happened  $\Delta t_{\text{train}}$  ago (Fig 9A, see also section “Protocols of Simulated Experiments”). In a similar task jays learned to avoid food caches containing crickets that they cached 4 days ago [23]. If one tests the subject on untrained content-age combinations, for example “blue square” after  $\Delta t_{\text{test}}$  (Fig 9A), different representations and associations of content and memory make different predictions.

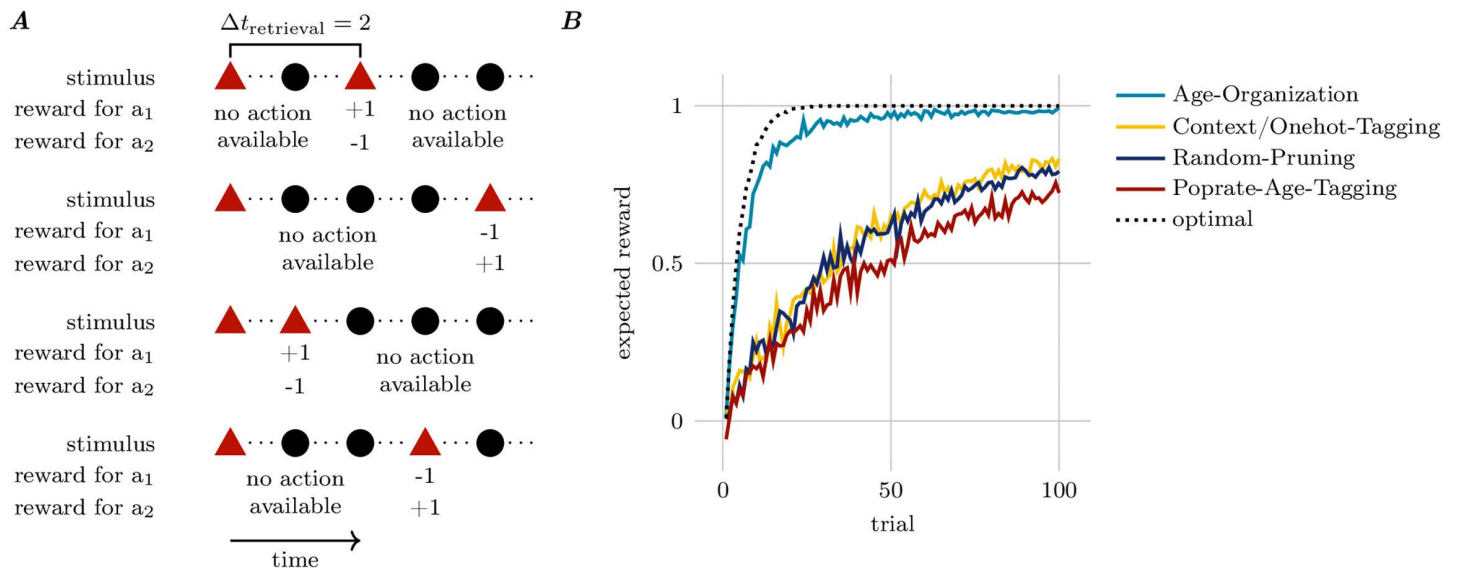
How a model generalizes depends mostly on the overlap of the recalled memories. For the one-hot coded memories in the Age-Organization model, there is no generalization from training to test settings, because distinct neurons are active during the recall of “red triangle  $\Delta t_{\text{train}}$  ago” and “blue square  $\Delta t_{\text{test}}$  ago” for any  $\Delta t_{\text{test}}$  (light blue curve in Fig 9B). In the Context/Onehot-Tagging model, the tags for “red triangle  $\Delta t_{\text{train}}$  ago” and “blue triangle  $\Delta t_{\text{test}} = \Delta t_{\text{train}}$  ago” are identical, despite the contents being different, and therefore there is some generalization to other memories of the same age (yellow curve in Fig 9B). Even more generalization occurs with the Poprate-Age-Tagging and the Random-Pruning model, because there is also some overlap in the recalled activity patterns for  $\Delta t_{\text{test}} \neq \Delta t_{\text{train}}$ .

Experiments that probe the learnability of different tasks can provide further evidence in favor or against specific models. Consider a task, where subjects are repeatedly trained to respond with action  $a_1$  if the age of a remembered event is less than some threshold and respond with action  $a_2$  otherwise (Fig 10A, see also section “Protocols of Simulated Experiments”). Such a task can be learned with all the models considered here, but the Age-Organization model potentially has



**Fig 9. Generalization of actions that depend on the age of memories, based on a single rewarded trial.** **A** A subject learns in a binary forced-choice task that action  $a_2$  is rewarded (or action  $a_1$  is punished), when recalling an event that happened  $\Delta t_{\text{train}}$  ago. Training occurs with the same stimulus (red triangle). After training, the subject is tested once with a different stimulus (e.g., blue square) and a retention interval  $\Delta t_{\text{test}}$  which may differ from the training interval. **B** Average probability across  $10^4$  simulated subjects of taking action  $a_2$  as a function of  $\Delta t_{\text{test}}$ . For the sparsest code (Age-Organization) there may not be any generalization to other stimuli, even when the test interval is the same as the training interval (blue dot at  $\Delta t_{\text{test}} = 3$ ). Conversely, for distributed representations (Poprate-Age-Tagging and Random-Pruning) there is generalization to other stimuli and test intervals different from the training interval. Quantitatively the results would be different for other learning rates or other values of the probability of  $a_2$  before learning, but qualitatively the results stay the same.

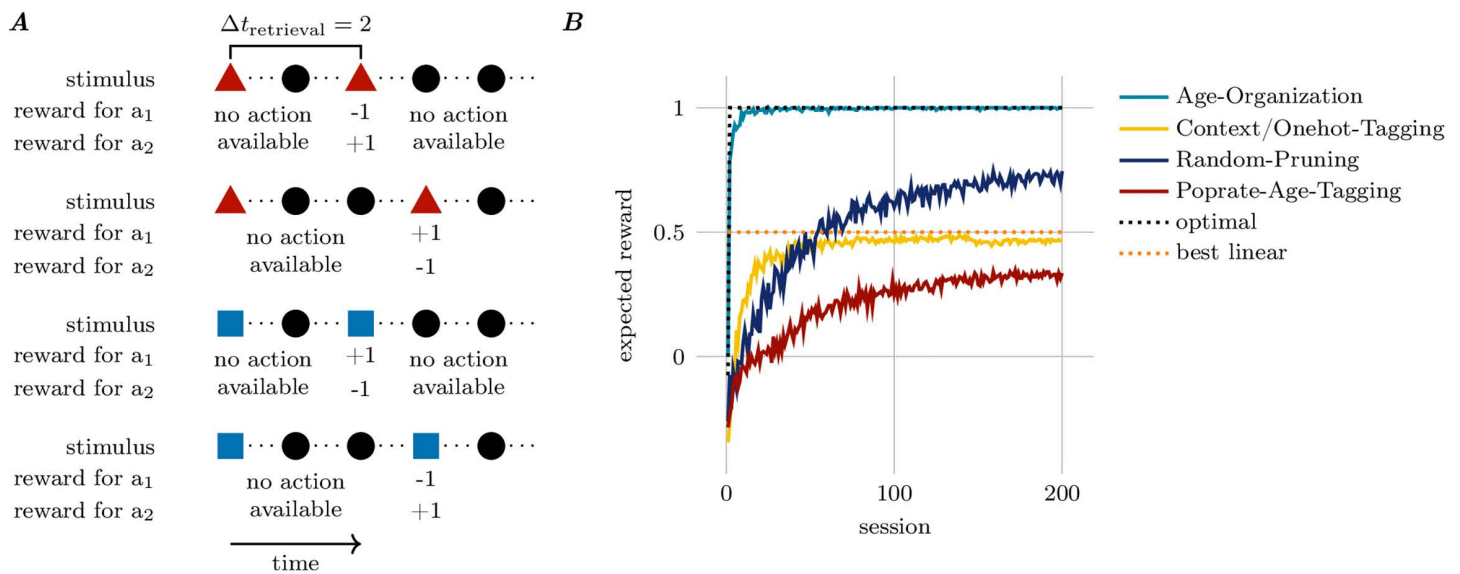
<https://doi.org/10.1371/journal.pcbi.1014575.g009>



**Fig 10. Learning to take actions that depend only on the age of memories.** **A** The experiment consists of multiple trials with random retrieval intervals  $\Delta t_{\text{retrieval}}$ . If the retrieval interval satisfies  $\Delta t_{\text{retrieval}} \leq 2$ , action  $a_1$  is rewarded (+1) and action  $a_2$  is punished (reward -1); reward contingencies are reversed, if the retrieval interval is larger than 2. **B** The expected reward per trial is measured over  $10^3$  simulated agents. All models can learn this task, but learning with one-hot codes can be faster than with other codes, because large learning rates can be chosen. The optimal performance (dashed line) was computed by averaging over  $10^4$  agents that make for each interval at most one mistake and always select the correct action afterwards.

<https://doi.org/10.1371/journal.pcbi.1014575.g010>

an advantage, because the one-hot encoding permits faster learning with higher learning rates than other representations (Fig 10B; learning rates for all models are optimized for best final performance in the tasks in Fig 10A and Fig 11A). However, if an experiment showed slow learning, this should not be taken as evidence against the Age-Organization model, because a suboptimal learning rate would induce slow learning also in the Age-Organization model.



**Fig 11. Learning to take actions that depend on the age and the content of memories.** **A** The experiment consists of multiple trials with different retrieval intervals  $\Delta t_{\text{retrieval}}$  and objects. If the object is red and the retrieval interval is  $\Delta t_{\text{retrieval}} = 2$  or the object is blue and  $\Delta t_{\text{retrieval}} = 3$ , action  $a_2$  is rewarded (+1) and action  $a_1$  is punished (reward -1); reward contingencies are reversed, otherwise. We call the sequence of these four trials one session. **B** The expected reward per session is measured over  $10^3$  simulated agents. Because this is an XOR-like task, models with Context/Onehot-Tagging or Poprate-Age-Tagging encoding cannot reach better performance than the best linear model (correct in 3 and wrong in one condition leads to an expected reward of  $(+3 - 1)/4 = 0.5$ ). The Random-Pruning model eventually learns the task, but it learns more slowly than the Age-Organization model with its one-hot encoding and a large learning rate. The optimal performance (dashed line) was computed by averaging over  $10^2$  agents that make for each interval and content at most one mistake and always select the correct action afterwards.

<https://doi.org/10.1371/journal.pcbi.1014575.g011>

If the correct responses of the subjects depend not only on the age of the recalled events, but also on their content, XOR-like tasks can be constructed (Fig 11A, see also section “Protocols of Simulated Experiments”). For example, consider a task where action  $a_2$  is rewarded, whenever the memory of a red triangle is at most 2 time units old or the memory of a blue square is older than 2 time units, but otherwise action  $a_1$  is rewarded. A linear readout cannot correctly learn this rule, if the content and the “when” information are given in a concatenation code (Fig 11B). We find that tagging models systematically fail on this task (yellow and red curve in Fig 11B). Both the Age-Organization and the Random-Pruning can learn this task, but the Age-Organization model could learn it much faster (optimized learning rates, same as in Fig 10B).

The results in these simulated experiments depend crucially on the representation of the recalled information that arrives as input to the final linear decision-making layer and the plasticity rule that changes the synaptic weights of this decision-making layer. With this insight, it is straightforward to design similar experiments that investigate, for example, the representation of different aspects of “what” and “where” information. However, this dependence upon the representation of recalled information also implies that purely behavioral experiments cannot discriminate between different models that exhibit the same activity pattern as input to the final decision-making layer. Therefore it may, for example, be almost impossible to discriminate timestamp from age representations of time, unless one can selectively manipulate the zeitgeber or rely on neural recordings.

### 3. Discussion

We showed that different choices of neural encoding, reference point of time, content-time associations, retrieval and readout mechanisms lead to a family of models, where the “what”, “where” and “when” of events can be stored and retrieved through automatic processes and Hebbian plasticity. Although the concrete implementations are idealized

“toy”-models that illustrate the central ideas succinctly, the same ideas generalize to bigger networks and more difficult tasks, as demonstrated in our simulation of food-caching behavior.

The considered neural codes (rate, one-hot, distributed and poprate, [Fig 2A](#)) are spatial codes, in the sense that all relevant information about the “what”, “where” and “when” of an event is given by the activity pattern of a group of neurons in a single time step. This activity pattern could be, for example, the average firing rates of neurons in a time window of 100 milliseconds. In addition, one could consider spatio-temporal codes, where some information is encoded in the temporal evolution of activity patterns. For example, a single neuron could implement a temporal one-hot code, where the information is encoded by the duration between some fixed reference point in time and a spike (time-to-spike code). For spatio-temporal codes, more sophisticated readout and learning mechanisms than the ones in [Fig 2F](#) would be needed to extract information from the temporal evolution of activity patterns. With spatio-temporal codes, the already large lower bound of 288 models (see section “Experimental Predictions”) would further increase and include models with spatio-temporal retrieval (e.g., [\[83\]](#)).

Chronological organization models can be generalized to include models where the groups of content neurons are not just copies of one another, but the representations of the memory content in each group differ from one another. For example, a lossy, age organized model, similar to the one described in section “Age Organization with Synaptic Pruning”, could store the gist of an event in some groups of neurons together with a detailed representation in other groups of neurons and forget the detailed representation faster than the gist. This could be a simple model of the trace transformation theory [\[16\]](#), which postulates that recall of details requires a functional hippocampus, whereas the gist can be recalled without hippocampus.

### 3.1. Comparison to other models

On a conceptual level, multiple theories of the processing of “when” information have been proposed. For example, [\[10\]](#) described eight theories: strength, chronological organization, time tagging, contextual overlap, encoding perturbation, associative chaining, reconstruction and order codes. The last four theories of this list are beyond the scope of this article. Our work, however, provides concrete hypotheses for neural implementations of the first four theories and discusses their implications for the readout of temporal and content information.

Although we focused on long-term memory with behavioral readout, we briefly summarize here known results where storage and retrieval are separated by less than a few minutes, including those with human subjects where the behavior is instructed. In psychology, there is a huge literature on laboratory-based memory tasks on short timescales [\[84\]](#). Among the most popular computational models to explain these memory experiments are different variants of temporal context models [\[18,36–38\]](#) (see also [Table 2](#)), which keep memories of the recent past with decaying activity traces (temporal context) and learn associations between temporal context and individual memory items with fast synaptic changes. These models can thus be seen as implementing both a rate code of age information and a distributed timestamp code ([Table 2](#)). Whereas the decaying activity traces are unlikely to extend to timescales of days or years, there is some experimental support for the contextual overlap theory also for long timescales. For example, recency judgments were found to be context-dependent [\[85\]](#). Furthermore, the contiguity effect of remembering multiple events jointly, if they happened at similar moments in time, seems to generalize to autobiographical memory and can be explained with temporal context models [\[86\]](#). However, there is also evidence for hippocampus-dependent reconstruction-based theories in humans [\[87\]](#).

Our models are rooted in the tradition of Hebbian memory models. Notably, there are interesting similarities between the Random-Pruning model and the palimpsest multi-state synapse model of Amit and Fusi [\[88\]](#): in both, synapses strengthen according to a Hebbian learning rule, and older memories are gradually forgotten. However, there are also significant differences: in the Random-Pruning model, synapses are pruned after durations that depend on the post-synaptic neurons, whereas in the Amit and Fusi model, old memories are overwritten through a stimulus-driven stochastic process. Although the overlap between stored and retrieved memory patterns may be indicative of the age of a memory in the Amit

and Fusi model, it is unclear whether with partial retrieval cues it is possible to decode the age of the memory in the Amit and Fusi model, and solve the memory tasks described in Experimental Predictions.

In machine learning, attention-based or memory-augmented differentiable neural networks are known to solve difficult memory tasks [89–92]. At the heart of these models are key-value memory systems that have a long tradition in computational neuroscience [46,93–97]. Simple key-value memories, including standard attention layers, are invariant under temporal permutation, that is, querying a memory system with temporally-permuted key and value matrices results in the same output as querying the unpermuted memory. Therefore, simple key-value memories would not be able to solve the behavioral tasks in Experimental Predictions. To retain temporal information, machine learning models rely usually on positional encoding [89,90]. The standard practice of adding positional encoding vectors to content vectors can be seen as a special case of a distributed timestamp code with (random) projection for the association of content and time.

### 3.2. Experimental evidence

The richness of biological phenomena in general, and memory phenomena in particular, in combination with the “theory-ladenness” of observations [98], allows for finding support in experimental data for different theories. Memory storage in the proposed models relies on one-shot Hebbian synaptic plasticity, for which there is experimental evidence [62,99]. Simple reinforcement learning of new tasks requires neoHebbian synaptic plasticity (see section “Synaptic Plasticity”), where plasticity depends on a modulatory signal that arrives up to a few seconds after Hebbian tagging; there is ample experimental evidence for this kind of plasticity [50–54]. Age representations of time in synaptic memories require ongoing changes of synaptic strengths or connections. An implementation of age representations of time with a rate code could rely on synapses that decay at different rates on a timescale of days to weeks [100,101]. One-hot or distributed encoding of the age of memories requires continual growing and pruning of synapses. Synaptic turnover has been widely observed [63–67,102], but we are not aware of experimental results that would support or refute the rewiring mechanisms proposed in our models. The specific pruning mechanism involving neuron-dependent pruning rates on a timescale from hours to years (Synaptic Plasticity) is not inconsistent with current knowledge of synaptic turnover, but direct experimental evidence for this hypothesis is lacking. Ongoing synaptic changes are also required in other theories and models of systems memory consolidation [16,103], and could potentially be mediated with mechanisms such as parallel synaptic pathways [76] or memory transfer [75]. Although there is experimental evidence – of varying strength – for all the synaptic processes needed to implement the above models, further experiments are clearly necessary to determine which processes are actually used by different species to remember the time of past events.

On timescales from a few hundred milliseconds to tens of seconds, and potentially up to a few hours, activity-based age tracking of memories may be mediated by time cells [9,69–74]. Time cells are neurons that fire at successive moments in temporally structured experiences. They were found in region CA1 of the hippocampus of rodents and in the entorhinal cortex of macaque monkeys [104] (but see [105], for an example where no time cells were found in CA1). Time cells could be seen as evidence for activity-based models of age tracking with concatenation or product associations (see TCM or TILT model in Table 2), but a random spatio-temporal feature model, akin to the Random-Pruning model, would probably lead to similar results. XOR-like experiments, such as the one suggested in Fig 11, could potentially be used to distinguish these hypotheses for age tracking on small timescales.

More relevant for the topic of this paper are experiments involving longer timescales. For example, neural responses to the same context in the CA1 region of the rodent hippocampus have been shown to drift over hours and days [106–110]. If the recorded activity in these studies reflects a temporal context tag used for memory storage, such slow representational drift would be consistent with a timestamp mechanism similar to the Context-Tagging model. On the other hand, if the measured activity primarily reflects the recall of a salient past experience – such as the first exposure to that context – these findings would favor an age-coding interpretation of past events. While it appears less likely that recall dominates the recorded activity, further experiments that clearly separate storage and recall phases are necessary to distinguish

between these interpretations. Lastly, it is also possible that the observed neural activity is unrelated to the storage or recall of specific events and instead reflects factors such as behavioral variability [111].

Sparse codes enable fast and flexible learning of behavioral rules, as demonstrated by the Age-Organization model. While explicit one-hot coding of content and time with single neurons does not seem plausible, relaxed versions of our models with redundant, sparse, and almost non-overlapping codes are consistent with experimental findings of sparse activity patterns in granule cells of the dentate gyrus (DG) of rodents [112–114] or the recently discovered “barcode” activity patterns in the hippocampus of food-caching birds [115]. Sparse activity patterns in the intermediate layer of the Random-Pruning model (Fig 7) were also crucial to scaling up our simulations (Fig 8). Models with sparse codes also get indirect support from behavioral studies: experiments with *California scrub-jays* showed convincingly that these birds have a flexible “what-where-when” memory system [14,21–24,116]. Although the existing experimental results cannot discriminate between the models considered here, the observation that they can learn different behavioral rules that depend on the content and age of memories within a few trials speaks in favor of an Age-Organization model or a Random-Pruning model, which allow fast and flexible learning. To get more behavioral evidence for the models, the suggested experiments in Fig 9–10 and Fig 11 could be run in a similar spirit to these food-caching experiments or the non-verbal experiments with other species mentioned in Table 1.

### 3.3. Conclusions

Remembering the “when” is an idiosyncratic feature of episodic and episodic-like memory. Thus, revealing the mechanisms that underlie the ability to estimate the age of memories is a crucial step towards a better understanding of episodic memory systems. The different models discussed here can serve as concrete hypotheses and the simulated experiments as inspirations for future experiments that combine behavioral and physiological recordings to learn more about how humans and animals remember the “when” of past events.

## 4. Methods

### 4.1. Formal description of codes

We consider a discretized version of the sensory stream in  $\prod_{i=1}^N \mathcal{C}_i \times \mathcal{T}$ , where  $\mathcal{C}_i$  denotes the finite set of values that sensor  $i$  can take and  $\mathcal{T}$  denotes the finite set of possible time points.

For a single finite set  $\mathcal{C} = \{c_1, \dots, c_{|\mathcal{C}|}\}$  with elements  $c_i$  and cardinality  $|\mathcal{C}|$ , we define three elementary coding schemes for representing element  $c_i$ :

- *rate code*:  $\text{rate}(c_i) := r(i) \in \mathbb{R}$ , where  $r$  is an arbitrary, one-to-one function.
- *one-hot code*: a  $|\mathcal{C}|$ -tuple-valued function

$\text{onehot}(c_i) := (\text{onehot}_1(c_i), \dots, \text{onehot}_{|\mathcal{C}|}(c_i))$  with  $\text{onehot}_j(c_i) := A_i \delta_{ij}$ , amplitude  $A_i > 0$  and Kronecker delta  $\delta_{ij} := 1$  if  $i=j$  and  $\delta_{ij} := 0$  otherwise.

- *distributed code*: a one-to-one  $M$ -tuple-valued function or random vector

$\text{distr}(c_i) := (\text{distr}_1(c_i), \dots, \text{distr}_M(c_i))$  with  $M > 1$  and  $0 < \text{distr}_j(c_i) \leq \text{distr}_{j'}(c_i)$  for at least one pair  $j \neq j'$  and at least one  $i$  (Fig 2A).

In addition to these elementary coding schemes, we consider population rate codes, where the value  $c_i$  is encoded by the number of active neurons. An example of a deterministic population rate code is given by  $\text{distr}_j(c_i) = 1$  if  $j \leq i$  and  $\text{distr}_j(c_i) = 0$  otherwise (poprate in Fig 2A). Stochastic population rate codes satisfy the condition  $\Pr(\sum_j \text{distr}_j(c_i) = i) = 1$ . These codes can be reduced to a rate code by computing the population activity  $\text{rate}(c_i) = i = \sum_{j=1}^{|\mathcal{C}|} \text{distr}_j(c_i)$ .

We write  $\ell(\text{code}(\mathcal{C}))$  or  $\ell(\text{code})$  for the length of an element of  $\mathcal{C}$  in code-coding, e.g.,  $\ell(\text{rate}(\mathcal{C})) = 1$ ,  $\ell(\text{onehot}(\mathcal{C})) = |\mathcal{C}|$ .

A generalization to continuous variables  $x$  (space, time, color, etc.) can easily be found with, e.g., continuous rate coding  $\text{rate}(x) = x$ , generalized one-hot coding with, e.g., radial basis functions  $\text{rbf}_j(x) = e^{-(j-x)^2}$  for a population of neurons with indices  $j = 1, \dots, M$  or generalized distributed code with, e.g., mixtures of radial basis functions.

## 4.2. Formal description of association schemes

Any function  $f: \mathcal{X} \times \mathcal{Y} \rightarrow \mathcal{Z}$  defines an association  $f(x, y) \in \mathcal{Z}$  between elements  $x \in \mathcal{X}$  and  $y \in \mathcal{Y}$ . If the function  $f$  is one-to-one, the association  $f(x, y)$  keeps the full information about the associated elements  $x$  and  $y$ .

We define

- *concatenation of codes*:

$$\mathbf{x} \oplus \mathbf{y} := (x_1, \dots, x_{\ell(\mathbf{x})}, y_1, \dots, y_{\ell(\mathbf{y})}) \quad (\text{Fig 2C}).$$

- *linear mixed codes*:  $\mathbf{x} \oplus_M \mathbf{y} := M(\mathbf{x} \oplus \mathbf{y})$ , where  $M: \mathbb{R}^{\ell(\mathbf{x})+\ell(\mathbf{y})} \rightarrow \mathbb{R}^M$  is a linear map.
- *product of codes*:  $\mathbf{x} \otimes \mathbf{y} := \mathbf{x} \cdot y_1 \oplus \mathbf{x} \cdot y_2 \oplus \dots \oplus \mathbf{x} \cdot y_{\ell(\mathbf{y})}$ , where the product  $\mathbf{x} \cdot y_i$  between a tuple  $\mathbf{x}$  and a scalar  $y_i$  has the standard meaning  $\mathbf{x} \cdot y_i = (x_1 y_i, \dots, x_{\ell(\mathbf{x})} y_i)$  (Fig 2C)
- *random projections*:  $\text{RP}(\mathbf{x}, \mathbf{y}) = \sigma(W_1 \mathbf{x} + W_2 \mathbf{y})$ , where  $W_1$  and  $W_2$  are fixed random matrices and  $\sigma$  is some non-linear function that is applied element-wise.

## 4.3. Mathematical description of the models

We model brains that observe sensory states  $\mathbf{x}_t, \mathbf{x}_{t+1}, \dots$  and take actions (or decisions)  $\mathbf{a}_t, \mathbf{a}_{t+1}, \dots$  on a slow timescale. These brains have internal neural states  $\mathbf{z}_\tau$  and synaptic connection parameters  $\mathbf{W}_\tau$  that evolve on a faster timescale, indicated by the time index  $\tau$ .

For all the models we used the storage and recall mechanism with synaptic delays (described in Fig 3) and hetero-associative recall. For the tagging models, this implementation differs from the descriptions with auto-associative recall in Fig 4 and Fig 5, but it leads to the same predictions for the behavioral experiments. The hyperparameters used in the simulations are summarized in Table 3.

The internal neural states  $\mathbf{z}_\tau$  are organized into groups of neurons. Tagging models have sensor, intermediate, content, tag and actuator neurons and we write the neural state  $\mathbf{z}_\tau = \mathbf{z}_\tau^{\text{sensor}} \oplus \mathbf{z}_\tau^{\text{intermediate}} \oplus \mathbf{z}_\tau^{\text{content}} \oplus \mathbf{z}_\tau^{\text{tag}} \oplus \mathbf{z}_\tau^{\text{actuator}}$ . The Age-Organization model and the Random-Pruning model have the same groups of neurons, except that the group of tag neurons is lacking and the group of content neurons is larger than in the tagging models. The group of sensory neurons is further divided into two subgroups that receive color and shape as one-hot coded input. The activity in these sensory neurons propagates along the synaptic connections to down-stream neurons, until an action is taken. Once an action was taken, the next sensory input is provided to the sensory neurons.

The activity propagation along synaptic connections can be described in terms of the update of the neural state of neurons in group  $\mu$ , which is given by

$$\mathbf{z}_\tau^{(\mu)} = \sigma^{(\mu)}(\mathbf{W}_{\tau-1}^{\nu \rightarrow \mu} \mathbf{z}_{\tau-1}^{(\nu)}), \quad (1)$$

where  $\sigma^{(\mu)}$ , the activation function of neurons in group  $\mu$ , is applied element-wise to the matrix-vector product of synaptic weight matrix  $\mathbf{W}_{\tau-1}^{\nu \rightarrow \mu}$  and activity state  $\mathbf{z}_{\tau-1}^{(\nu)}$  of group  $\nu$  in the previous time-step  $\tau - 1$ . The activation function of the actuator group is the soft-max function  $\sigma^{\text{actuator}}(\mathbf{x})_i = e^{x_i} / \sum_j e^{x_j}$ . For all other groups of neurons we use the Heaviside function  $\sigma^{(\mu)}(\mathbf{x})_i = H(x_i - b) = 1$  if  $x_i > b$  and  $H(x_i - b) = 0$ , otherwise, where bias  $b=0$  for all groups of neurons except for the

**Table 3. Hyperparameters used in all simulations.**

| Hyperparameter                                               | Age-Organization | Context/Onehot-Tagging | Random-Pruning                | Poprate-Age-Tagging |
|--------------------------------------------------------------|------------------|------------------------|-------------------------------|---------------------|
| <i>Readout learning</i>                                      |                  |                        |                               |                     |
| Learning rate $\eta$                                         | 1.0              | 0.5                    | 0.02                          | 0.2                 |
| <i>Activation functions</i>                                  |                  |                        |                               |                     |
| Bias $b$ (content layer)                                     | 0                | 0                      | 1.5                           | 0                   |
| Bias $b$ (all other layers)                                  | 0                | 0                      | 0                             | 0                   |
| <i>Hebbian Latent-State-Decay (intermediate→content/tag)</i> |                  |                        |                               |                     |
| Decay term $\Delta s$                                        | —                | —                      | $\frac{1}{3}$                 | $\frac{1}{3}$       |
| $s_i^{\max}$ (tasks, Fig 9 – 11)                             | —                | —                      | $\sim \text{Uniform}(0, 6)$   | $i$                 |
| $s_i^{\max}$ (food caching, Fig 8)                           | —                | —                      | $\sim \text{Uniform}(0, 200)$ | —                   |
| <i>Network architecture (tasks, Fig 9–11)</i>                |                  |                        |                               |                     |
| Sensor neurons                                               | 12               | 12                     | 12                            | 12                  |
| Intermediate neurons                                         | 12               | 12                     | 50                            | 12                  |
| Content neurons                                              | 72               | 12                     | 200                           | 12                  |
| Tag neurons                                                  | —                | 6                      | —                             | 6                   |
| Actuator neurons                                             | 2                | 2                      | 2                             | 2                   |
| <i>Network architecture (food caching, Fig 8)</i>            |                  |                        |                               |                     |
| Sensor neurons                                               | —                | —                      | 1002                          | —                   |
| Intermediate neurons                                         | —                | —                      | 4995                          | —                   |
| Content neurons                                              | —                | —                      | 60                            | —                   |
| Actuator neurons                                             | —                | —                      | 2                             | —                   |

<https://doi.org/10.1371/journal.pcbi.1014575.t003>

content group in the Random-Pruning model, where  $b=1.5$ . The non-zero bias in the Random-Pruning model ensures sparse activity in the content layer. Action  $a_t$  is sampled according to the distribution given by  $\mathbf{z}_t^{\text{actuator}}$  after recall has happened.

Because we used the recall mechanism with synaptic delays (Fig 3) and it takes three time steps for sensory activity to propagate along the “input-intermediate-content-actuator” pathway, there is a simple relationship between the slow timescale indexed by  $t$  and the fast timescale indexed by  $\tau$ : if  $\mathbf{z}_t^{\text{sensor}} = \text{onehot}(\text{color}(x_t)) \oplus \text{onehot}(\text{shape}(x_t))$ , action  $a_t$  will be sampled with probability  $\mathbf{z}_{\tau+3}^{\text{actuator}}$ . The sensory neurons are inactive during propagation of the activity through the neural network, i.e.,  $\mathbf{z}_{\tau+1}^{\text{sensor}} = \mathbf{z}_{\tau+2}^{\text{sensor}} = \mathbf{z}_{\tau+3}^{\text{sensor}} = \mathbf{0}$ ; the sensory neurons are reactivated, once action  $a_t$  has been taken, i.e.,  $\mathbf{z}_{\tau+4}^{\text{sensor}} = \text{onehot}(\text{color}(x_{t+1})) \oplus \text{onehot}(\text{shape}(x_{t+1}))$ .

Synaptic weight matrices are static or evolve according to one of the following plasticity rules:

#### Hebbian

$$\Delta W_{\tau,ij}^{\nu \rightarrow \mu} = W_{\tau,ij}^{\nu \rightarrow \mu} - W_{\tau-1,ij}^{\nu \rightarrow \mu} = z_{\tau-1,i}^{(\mu)} z_{\tau-1,j}^{(\nu)}. \quad (2)$$

where  $i$  is the postsynaptic neuron in group  $\mu$  and  $j$  the presynaptic neuron in group  $\nu$ .

#### Reward-Modulated Hebbian

$$E_{\tau,ij}^{\nu \rightarrow \text{actuator}} = \tilde{z}_{\tau,i}^{\text{actuator}} z_{\tau-1,j}^{(\nu)} \quad (3)$$

$$\Delta W_{\tau,ij}^{\nu \rightarrow \text{actuator}} = W_{\tau,ij}^{\nu \rightarrow \text{actuator}} - W_{\tau-1,ij}^{\nu \rightarrow \text{actuator}} = \eta r_t E_{\tau-1,ij}^{\nu \rightarrow \text{actuator}} \quad (4)$$

where  $E_{\tau}^{\nu \rightarrow \text{actuator}}$  is an eligibility trace that depends on  $\tilde{z}_{\tau,i}^{\text{actuator}} = 1 - z_{\tau,i}^{\text{actuator}}$  for postsynaptic neuron  $i = a_t$  and  $\tilde{z}_{\tau,i}^{\text{actuator}} = -z_{\tau,i}^{\text{actuator}}$  for postsynaptic neurons  $i \neq a_t$ ,  $\eta$  a learning rate and  $r_t \in \{-1, 1\}$  is the reward obtained after performing action  $a_{t-1}$ . This plasticity rule can be seen as a policy gradient (REINFORCE) rule [117], where the terms  $\tilde{z}_{\tau,i}^{\text{actuator}}$  arise as a consequence of taking the derivative of the logarithm of the soft-max policy function in the derivation of the REINFORCE rule

$$\frac{\partial}{\partial W_{ij}} \log \Pr(a_t = k) = \frac{\partial}{\partial W_{ij}} \log \sigma^{\text{actuator}}(Wz^{(\nu)})_k \quad (5)$$

$$= \frac{\partial}{\partial W_{ij}} \left[ (Wz^{(\nu)})_k - \log \left( \sum_l e^{(Wz^{(\nu)})_l} \right) \right] \quad (6)$$

$$= (\delta_{ik} - z_i^{\text{actuator}}) z_j^{(\nu)} \quad (7)$$

$$= \tilde{z}_i^{\text{actuator}} z_j^{(\nu)}. \quad (8)$$

This plasticity rule is used in all models for the readout weights (orange in the figures). We did not extensively optimize the learning rates, but ran the experiments in Fig 10 and Fig 11 for

$$\eta \in \{0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1.0, 2.0\},$$

and picked for each model a value that lead to good performances on both tasks (Age-Organization:  $\eta = 1.0$ ; Context/Onehot-Tagging:  $\eta = 0.5$ ; Random-Pruning:  $\eta = 0.02$ ; Poprate-Age-Tagging:  $\eta = 0.2$ ). We used the same learning rates in all simulations reported in Fig 8, Fig 9, Fig 10, Fig 11.

### Hebbian Latent-State-Decay

$$S_{\tau,ij}^{\nu \rightarrow \mu} = \left[ S_{\tau-1,ij}^{\nu \rightarrow \mu} + s_i^{\max} z_{\tau-1,i}^{(\mu)} z_{\tau-1,j}^{(\nu)} - \Delta s \right]_0^{s_i^{\max}} \quad (9)$$

$$W_{\tau,ij}^{\nu \rightarrow \mu} = H(S_{\tau,ij}^{\nu \rightarrow \mu}) \quad (10)$$

where  $S_{\tau}^{\nu \rightarrow \mu}$  is a latent synaptic state matrix,  $[x]_a^b = \min(b, \max(a, x))$ ,  $s_i^{\max}$  is the maximal value the latent state for post-synaptic neuron  $i$  can achieve,  $\Delta s$  is a decay term and  $H$  is the Heaviside function. After a Hebbian growth, synapses of this kind are pruned after  $s_i^{\max}/\Delta s$  time-steps, unless they are restrengthened meanwhile. This plasticity rule is used in the Poprate-Age-Tagging model for the connections from intermediate to tag neurons and in the Random-Pruning model for connections from intermediate to content neurons. In the Poprate-Age-Tagging, we set  $s_i^{\max} = i$  for the six tag neurons. In the Random-Pruning, we sampled  $s_i^{\max} \sim \text{Uniform}(0, s^{\max})$ , where  $s^{\max} = 6$  for the simulations in Fig 9, Fig 10 and Fig 11, and  $s^{\max} = 200$  for the larger model in Fig 8. We set  $\Delta s = 1/3$  for all models and simulations, implying that the latent states decay by one, for one slow time step indexed by  $t$ .

### Postsynaptic Rewiring

$$W_{\tau,ij}^{\nu \rightarrow \mu} = W_{\tau-\Delta\tau, i-1, j}^{\nu \rightarrow \mu} \quad (11)$$

where  $\Delta\tau$  is such that the synaptic change happens always when perceiving a new input  $x_t$ . This rule is an abstract implementation of a hypothetical systems consolidation process, where the active neuron encodes the age of a memory (e.g., Fig 4C or Fig 6A). This plasticity rule is used in the Onehot-Age-Tagging model for the connections between the intermediate neurons and the tag neurons and in the Age-Organization model for the connections between the intermediate neurons and the content neurons.

#### 4.4. Equivalence of one-hot coding and deterministic population-rate coding

Let  $\mathbf{x}$  be a one-hot coded neural activity pattern,  $\mathbf{w} \in \mathbb{R}^N$  a weight vector,  $y = \mathbf{w}^T \mathbf{x}$  a linear readout, and  $\tilde{\mathbf{x}} = P\mathbf{x}$ , with  $P$  such that  $\tilde{x}_i = \sum_{j=1}^N x_j$ , the re-parametrization from one-hot coding to the deterministic population rate code at the bottom of Fig 2A. The corresponding transformation  $\tilde{\mathbf{w}} = (P^{-1})^T \mathbf{w}$  leaves the response invariant, i.e.,  $\tilde{y} = \tilde{\mathbf{w}}^T \tilde{\mathbf{x}} = y$ . The gradient descent learning rule  $\Delta w_i = \eta \frac{\partial}{\partial w_i} F(y) = \eta f(y) x_i$ , for some  $f(x) = \frac{\partial}{\partial x} F(x)$ , transforms under  $P$  to  $\Delta \tilde{w}_i = \eta f(\tilde{y}) (2\tilde{x}_i - (\tilde{x}_{i-1} + \tilde{x}_{i+1}))$  for  $i = 2, \dots, N-1$  and  $\Delta \tilde{w}_1 = \eta f(\tilde{y}) (\tilde{x}_1 - \tilde{x}_2)$ ,  $\Delta \tilde{w}_N = \eta f(\tilde{y}) (2\tilde{x}_N - \tilde{x}_{N-1})$  as can be seen by computing  $(P^{-1})^T (P)^{-1}$  [118]. If the synapses are spatially organized such that the inputs of presynaptic neurons  $i-1, i, i+1$  are neighboring, the resulting plasticity rule features cross-talk between neighboring synapses. In particular, a synaptic weight  $\tilde{w}_i$  should only be changed, if the inputs at the neighboring synapses  $i-1$  and  $i+1$  differ from the input at synapse  $i$ .

#### 4.5. Protocols of simulated experiments

The experimental protocols of the simulated experiments are reported from the perspective of the experimenter. All protocols are constructed using four basic actions that an experimenter performs: show to the subject some stimulus (show\_to!), provide a choice of multiple actions and observe the action taken by the subject (force\_to\_choose\_and\_observe\_action!), reward or punish the subject (reward!) and keep track of the relevant observations in the results table (push!). It is assumed, but not explicitly modelled, that in between each action taken by the experimenter there is some waiting time. These waiting times should be sufficiently long, for example on the order of hours or days, such that subjects cannot solve the tasks with working memory alone, but need to rely on long-term “what-where-when” memory. The code for all simulations is available at <https://github.com/jbrea/RememberingTheWhen.jl>

#### Author contributions

**Conceptualization:** Johanni Brea.

**Funding acquisition:** Wulfram Gerstner.

**Investigation:** Johanni Brea.

**Software:** Johanni Brea.

**Supervision:** Wulfram Gerstner.

**Visualization:** Johanni Brea.

**Writing – original draft:** Johanni Brea.

**Writing – review & editing:** Johanni Brea, Alireza Modirshanechi, Georgios Iatropoulos, Wulfram Gerstner.

#### References

1. Carr CE, Konishi M. A circuit for detection of interaural time differences in the brain stem of the barn owl. *J Neurosci*. 1990;10(10):3227–46. <https://doi.org/10.1523/JNEUROSCI.10-10-03227.1990> PMID: 2213141
2. Gerstner W, Kempter R, van Hemmen JL, Wagner H. A neuronal learning rule for sub-millisecond temporal coding. *Nature*. 1996;383(6595):76–81. <https://doi.org/10.1038/383076a0> PMID: 8779718
3. Grothe B, Pecka M, McAlpine D. Mechanisms of sound localization in mammals. *Physiol Rev*. 2010;90(3):983–1012. <https://doi.org/10.1152/physrev.00026.2009> PMID: 20664077

4. Buhusi CV, Meck WH. What makes us tick? Functional and neural mechanisms of interval timing. *Nat Rev Neurosci*. 2005;6(10):755–65. <https://doi.org/10.1038/nrn1764> PMID: [16163383](#)
5. Addyman C, French RM, Thomas E. Computational models of interval timing. *Current Opinion in Behavioral Sciences*. 2016;8:140–6. <https://doi.org/10.1016/j.cobeha.2016.01.004>
6. Paton JJ, Buonomano DV. The Neural Basis of Timing: Distributed Mechanisms for Diverse Functions. *Neuron*. 2018;98(4):687–705. <https://doi.org/10.1016/j.neuron.2018.03.045> PMID: [29772201](#)
7. Issa JB, Tocker G, Hasselmo ME, Heys JG, Dombeck DA. Navigating Through Time: A Spatial Navigation Perspective on How the Brain May Encode Time. *Annu Rev Neurosci*. 2020;43:73–93. <https://doi.org/10.1146/annurev-neuro-201419-011117> PMID: [31961765](#)
8. Lee ACH, Thavabalasingam S, Alushaj D, Çavdaroglu B, Ito R. The hippocampus contributes to temporal duration memory in the context of event sequences: A cross-species perspective. *Neuropsychologia*. 2020;137:107300. <https://doi.org/10.1016/j.neuropsychologia.2019.107300> PMID: [31836410](#)
9. Tsao A, Yousefzadeh SA, Meck WH, Moser M-B, Moser EI. The neural bases for timing of durations. *Nat Rev Neurosci*. 2022;23(11):646–65. <https://doi.org/10.1038/s41583-022-00623-3> PMID: [36097049](#)
10. Friedman WJ. Memory for the time of past events. *Psychological Bulletin*. 1993;113(1):44–66. <https://doi.org/10.1037/0033-2909.113.1.44>
11. Friedman WJ. The Development of Memory for the Times of Past Events. *The Wiley Handbook on the Development of Children's Memory*. Wiley. 2013. 394–407. <https://doi.org/10.1002/9781118597705.ch17>
12. Pathman T, Larkina M, Burch M, Bauer PJ. Young Children's Memory for the Times of Personal Past Events. *J Cogn Dev*. 2013;14(1):120–40. <https://doi.org/10.1080/15248372.2011.641185> PMID: [23687467](#)
13. Jelbert SA, Clayton NS. Comparing the non-linguistic hallmarks of episodic memory systems in corvids and children. *Current Opinion in Behavioral Sciences*. 2017;17:99–106. <https://doi.org/10.1016/j.cobeha.2017.07.011>
14. Clayton NS, Dickinson A. Episodic-like memory during cache recovery by scrub jays. *Nature*. 1998;395(6699):272–4. <https://doi.org/10.1038/26216> PMID: [9751053](#)
15. Abraham WC, Jones OD, Glanzman DL. Is plasticity of synapses the mechanism of long-term memory storage?. *npj Science of Learning*. 2019;4(1). <https://doi.org/10.1038/s41539-019-0048-y>
16. Moscovitch M, Gilboa A. Systems consolidation, transformation and reorganization: Multiple trace theory, trace transformation theory and their competitors. *PsyArXiv Preprints*. 2021. <https://doi.org/10.31234/osf.io/yxbrs>
17. Norman KA, Detre G, Polyn SM. Computational Models of Episodic Memory. *The Cambridge Handbook of Computational Psychology*. 2008; p. 189–225. <https://doi.org/10.1017/cbo9780511816772.011>
18. Sederberg PB, Darby KP. Computational Models of Episodic Memory. *The Cambridge Handbook of Computational Cognitive Sciences*. Cambridge University Press. 2023. p. 567–610. <https://doi.org/10.1017/9781108755610.022>
19. Laird JE, Lebiere C, Rosenbloom PS. A Standard Model of the Mind: Toward a Common Computational Framework across Artificial Intelligence, Cognitive Science, Neuroscience, and Robotics. *AI Magazine*. 2017;38(4):13–26. <https://doi.org/10.1609/aimag.v38i4.2744>
20. Rolls ET, Treves A. A theory of hippocampal function: New developments. *Prog Neurobiol*. 2024;238:102636. <https://doi.org/10.1016/j.pneurobio.2024.102636> PMID: [38834132](#)
21. Clayton NS, Dickinson A. Scrub jays (*Aphelocoma coerulescens*) remember the relative time of caching as well as the location and content of their caches. *J Comp Psychol*. 1999;113(4):403–16. <https://doi.org/10.1037/0735-7036.113.4.403> PMID: [10608564](#)
22. Clayton NS, Yu KS, Dickinson A. Scrub jays (*Aphelocoma coerulescens*) form integrated memories of the multiple features of caching episodes. *J Exp Psychol Anim Behav Process*. 2001;27(1):17–29. <https://doi.org/10.1037/0097-7403.27.1.17> PMID: [11199511](#)
23. Clayton NS, Yu KS, Dickinson A. Interacting Cache memories: evidence for flexible memory use by Western Scrub-Jays (*Aphelocoma californica*). *J Exp Psychol Anim Behav Process*. 2003;29(1):14–22. <https://doi.org/10.1037/0097-7403.29.1.14> PMID: [12561130](#)
24. Clayton NS, Dally J, Gilbert J, Dickinson A. Food caching by western scrub-jays (*Aphelocoma californica*) is sensitive to the conditions at recovery. *J Exp Psychol Anim Behav Process*. 2005;31(2):115–24. <https://doi.org/10.1037/0097-7403.31.2.115> PMID: [15839770](#)
25. Feeney MC, Roberts WA, Sherry DF. Memory for what, where, and when in the black-capped chickadee (*Parus atricapillus*). *Anim Cogn*. 2009;12(6):767–77. <https://doi.org/10.1007/s10071-009-0236-x> PMID: [19466468](#)
26. Babb SJ, Crystal JD. Episodic-like memory in the rat. *Curr Biol*. 2006;16(13):1317–21. <https://doi.org/10.1016/j.cub.2006.05.025> PMID: [16824919](#)
27. Roberts WA, Feeney MC, Macpherson K, Petter M, McMillan N, Musolino E. Episodic-like memory in rats: is it based on when or how long ago?. *Science*. 2008;320(5872):113–5. <https://doi.org/10.1126/science.1152709> PMID: [18388296](#)
28. Zhou W, Crystal JD. Evidence for remembering when events occurred in a rodent model of episodic memory. *Proc Natl Acad Sci U S A*. 2009;106(23):9525–9. <https://doi.org/10.1073/pnas.0904360106> PMID: [19458264](#)
29. Martin-Ordas G, Haun D, Colmenares F, Call J. Keeping track of time: evidence for episodic-like memory in great apes. *Anim Cogn*. 2010;13(2):331–40. <https://doi.org/10.1007/s10071-009-0282-4> PMID: [19784852](#)
30. Pladevall J, Mendes N, Riba D, Llorente M, Amici F. No evidence of what-where-when memory in great apes (*Pan troglodytes*, *Pan paniscus*, *Pongo abelii*, and *Gorilla gorilla*). *J Comp Psychol*. 2020;134(2):252–61. <https://doi.org/10.1037/com0000215> PMID: [32052981](#)

31. Martin-Ordas G, Atance CM, Caza J. Did the popsicle melt? Preschoolers' performance in an episodic-like memory task. *Memory*. 2017;25(9):1260–71. <https://doi.org/10.1080/09658211.2017.1285940> PMID: [28276982](#)
32. Hebb DO. *The Organization of Behavior*. New York: Wiley. 1949.
33. Treves A, Rolls ET. Computational constraints suggest the need for two distinct input systems to the hippocampal CA3 network. *Hippocampus*. 1992;2(2):189–99. <https://doi.org/10.1002/hipo.450020209> PMID: [1308182](#)
34. Treves A, Rolls ET. Computational analysis of the role of the hippocampus in memory. *Hippocampus*. 1994;4(3):374–91. <https://doi.org/10.1002/hipo.450040319> PMID: [7842058](#)
35. Hasselmo ME, Bodelón C, Wyble BP. A proposed function for hippocampal theta rhythm: separate phases of encoding and retrieval enhance reversal of prior learning. *Neural Comput*. 2002;14(4):793–817. <https://doi.org/10.1162/089976602317318965> PMID: [11936962](#)
36. Howard MW, Kahana MJ. A Distributed Representation of Temporal Context. *Journal of Mathematical Psychology*. 2002;46(3):269–99. <https://doi.org/10.1006/jmps.2001.1388>
37. Polyn SM, Norman KA, Kahana MJ. A context maintenance and retrieval model of organizational processes in free recall. *Psychol Rev*. 2009;116(1):129–56. <https://doi.org/10.1037/a0014420> PMID: [19159151](#)
38. Howard MW. Formal models of memory based on temporally-varying representations. *arXiv e-prints*. 2022. arXiv:2201.01796.
39. Brown GD, Preece T, Hulme C. Oscillator-based memory for serial order. *Psychol Rev*. 2000;107(1):127–81. <https://doi.org/10.1037/0033-295x.107.1.127> PMID: [10687405](#)
40. Shankar KH, Howard MW. A scale-invariant internal representation of time. *Neural Comput*. 2012;24(1):134–93. [https://doi.org/10.1162/NECO\\_a\\_00212](https://doi.org/10.1162/NECO_a_00212) PMID: [21919782](#)
41. McTaggart JE. The unreality of time. *Mind*. 1908;17(68):457–74.
42. Smolensky P. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*. 1990;46(1–2):159–216. [https://doi.org/10.1016/0004-3702\(90\)90007-m](https://doi.org/10.1016/0004-3702(90)90007-m)
43. Plate TA. Holographic reduced representations. *IEEE Trans Neural Netw*. 1995;6(3):623–41. <https://doi.org/10.1109/72.377968> PMID: [18263348](#)
44. Kelly MA, Blöstein D, Mewhort DJK. Encoding structure in holographic reduced representations. *Can J Exp Psychol*. 2013;67(2):79–93. <https://doi.org/10.1037/a0030301> PMID: [23205508](#)
45. Amit DJ. *Modeling Brain Function: The World of Attractor Neural Networks*. Cambridge University Press. 1989.
46. Iatropoulos G, Brea J, Gerstner W. Kernel Memory Networks: A Unifying Framework for Memory Modeling. In: *Advances in Neural Information Processing Systems* 35, 2022. 35326–38. <https://doi.org/10.52202/068431-2560>
47. Stella F, Treves A. Associative memory storage and retrieval: involvement of theta oscillations in hippocampal information processing. *Neural Plast*. 2011;2011:683961. <https://doi.org/10.1155/2011/683961> PMID: [21961072](#)
48. Jacobs J. Hippocampal theta oscillations are slower in humans than in rodents: implications for models of spatial navigation and memory. *Philos Trans R Soc Lond B Biol Sci*. 2013;369(1635):20130304. <https://doi.org/10.1098/rstb.2013.0304> PMID: [24366145](#)
49. Herweg NA, Solomon EA, Kahana MJ. Theta Oscillations in Human Memory. *Trends Cogn Sci*. 2020;24(3):208–27. <https://doi.org/10.1016/j.tics.2019.12.006> PMID: [32029359](#)
50. Lisman J, Grace AA, Duzel E. A neoHebbian framework for episodic memory; role of dopamine-dependent late LTP. *Trends Neurosci*. 2011;34(10):536–47. <https://doi.org/10.1016/j.tins.2011.07.006> PMID: [21851992](#)
51. Gerstner W, Lehmann M, Liakoni V, Corneil D, Brea J. Eligibility Traces and Plasticity on Behavioral Time Scales: Experimental Support of Neo-Hebbian Three-Factor Learning Rules. *Front Neural Circuits*. 2018;12:53. <https://doi.org/10.3389/fncir.2018.00053> PMID: [30108488](#)
52. Roelfsema PR, Holtmaat A. Control of synaptic plasticity in deep cortical networks. *Nat Rev Neurosci*. 2018;19(3):166–80. <https://doi.org/10.1038/nrn.2018.6> PMID: [29449713](#)
53. Kuśmiercz Ł, Isomura T, Toyozumi T. Learning with three factors: modulating Hebbian plasticity with errors. *Curr Opin Neurobiol*. 2017;46:170–7. <https://doi.org/10.1016/j.conb.2017.08.020> PMID: [28918313](#)
54. Magee JC, Grienberger C. Synaptic Plasticity Forms and Functions. *Annu Rev Neurosci*. 2020;43:95–117. <https://doi.org/10.1146/annurev-neuro-090919-022842> PMID: [32075520](#)
55. Pfister J-P, Toyozumi T, Barber D, Gerstner W. Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. *Neural Comput*. 2006;18(6):1318–48. <https://doi.org/10.1162/neco.2006.18.6.1318> PMID: [16764506](#)
56. Urbanczik R, Senn W. Learning by the dendritic prediction of somatic spiking. *Neuron*. 2014;81(3):521–8. <https://doi.org/10.1016/j.neuron.2013.11.030> PMID: [24507189](#)
57. Hertz JA, Krogh AS, Palmer RG. *Introduction to the Theory of Neural Computation*. Westview Press. 1991.
58. Lillicrap TP, Cownden D, Tweed DB, Akerman CJ. Random synaptic feedback weights support error backpropagation for deep learning. *Nat Commun*. 2016;7:13276. <https://doi.org/10.1038/ncomms13276> PMID: [27824044](#)
59. Illing B, Gerstner W, Brea J. Biologically plausible deep learning - But how far can we go with shallow networks?. *Neural Netw*. 2019;118:90–101. <https://doi.org/10.1016/j.neunet.2019.06.001> PMID: [31254771](#)

60. Roelfsema PR, van Ooyen A. Attention-gated reinforcement learning of internal representations for classification. *Neural Comput*. 2005;17(10):2176–214. <https://doi.org/10.1162/0899766054615699> PMID: [16105222](#)
61. Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev*. 1958;65(6):386–408. <https://doi.org/10.1037/h0042519> PMID: [13602029](#)
62. Petersen CC, Malenka RC, Nicoll RA, Hopfield JJ. All-or-none potentiation at CA3-CA1 synapses. *Proc Natl Acad Sci U S A*. 1998;95(8):4732–7. <https://doi.org/10.1073/pnas.95.8.4732> PMID: [9539807](#)
63. Holtmaat AJGD, Trachtenberg JT, Wilbrecht L, Shepherd GM, Zhang X, Knott GW, et al. Transient and persistent dendritic spines in the neocortex in vivo. *Neuron*. 2005;45(2):279–91. <https://doi.org/10.1016/j.neuron.2005.01.003> PMID: [15664179](#)
64. Loewenstein Y, Yanover U, Rumpel S. Predicting the Dynamics of Network Connectivity in the Neocortex. *J Neurosci*. 2015;35(36):12535–44. <https://doi.org/10.1523/JNEUROSCI.2917-14.2015> PMID: [26354919](#)
65. Kasai H, Ziv NE, Okazaki H, Yagishita S, Toyozumi T. Nature Reviews Neuroscience. 2021;22(7):407–22. <https://doi.org/10.1038/s41583-021-00467-3>
66. Attardo A, Fitzgerald JE, Schnitzer MJ. Impermanence of dendritic spines in live adult CA1 hippocampus. *Nature*. 2015;523(7562):592–6. <https://doi.org/10.1038/nature14467> PMID: [26098371](#)
67. Chen JL, Villa KL, Cha JW, So PTC, Kubota Y, Nedivi E. Clustered dynamics of inhibitory synapses and dendritic spines in the adult neocortex. *Neuron*. 2012;74(2):361–73. <https://doi.org/10.1016/j.neuron.2012.02.030> PMID: [22542188](#)
68. Barry MLLR, Gerstner W. Fast adaptation to rule switching using neuronal surprise. *PLoS Comput Biol*. 2024;20(2):e1011839. <https://doi.org/10.1371/journal.pcbi.1011839> PMID: [38377112](#)
69. Kraus BJ, Robinson RJ 2nd, White JA, Eichenbaum H, Hasselmo ME. Hippocampal “time cells”: time versus path integration. *Neuron*. 2013;78(6):1090–101. <https://doi.org/10.1016/j.neuron.2013.04.015> PMID: [23707613](#)
70. MacDonald CJ, Lepage KQ, Eden UT, Eichenbaum H. Hippocampal “time cells” bridge the gap in memory for discontinuous events. *Neuron*. 2011;71(4):737–49. <https://doi.org/10.1016/j.neuron.2011.07.012> PMID: [21867888](#)
71. Eichenbaum H. Time cells in the hippocampus: a new dimension for mapping memories. *Nat Rev Neurosci*. 2014;15(11):732–44. <https://doi.org/10.1038/nrn3827> PMID: [25269553](#)
72. Eichenbaum H. On the Integration of Space, Time, and Memory. *Neuron*. 2017;95(5):1007–18. <https://doi.org/10.1016/j.neuron.2017.06.036> PMID: [28858612](#)
73. Tsao A, Sugar J, Lu L, Wang C, Knierim JJ, Moser M-B, et al. Integrating time from experience in the lateral entorhinal cortex. *Nature*. 2018;561(7721):57–62. <https://doi.org/10.1038/s41586-018-0459-6> PMID: [30158699](#)
74. Taxis J, Pnevmatikakis EA, Dorian CC, Mylavarapu AL, Arora JS, Samadian KD, et al. Differential Emergence and Stability of Sensory and Temporal Representations in Context-Specific Hippocampal Sequences. *Neuron*. 2020;108(5):984–998.e9. <https://doi.org/10.1016/j.neuron.2020.08.028> PMID: [32949502](#)
75. Roxin A, Fusi S. Efficient partitioning of memory systems and its importance for memory consolidation. *PLoS Comput Biol*. 2013;9(7):e1003146. <https://doi.org/10.1371/journal.pcbi.1003146> PMID: [23935470](#)
76. Remme MWH, Bergmann U, Alevi D, Schreiber S, Sprekeler H, Kempter R. Hebbian plasticity in parallel synaptic pathways: A circuit mechanism for systems memory consolidation. *PLoS Comput Biol*. 2021;17(12):e1009681. <https://doi.org/10.1371/journal.pcbi.1009681> PMID: [34874938](#)
77. Howard MW, Shankar KH, Aue WR, Criss AH. A distributed representation of internal time. *Psychol Rev*. 2015;122(1):24–53. <https://doi.org/10.1037/a0037840> PMID: [25330329](#)
78. Nadel L, Samsonovich A, Ryan L, Moscovitch M. Multiple trace theory of human memory: computational, neuroimaging, and neuropsychological results. *Hippocampus*. 2000;10(4):352–68. [https://doi.org/10.1002/1098-1063\(2000\)10:4<352::AID-HIPO2>3.0.CO;2-D](https://doi.org/10.1002/1098-1063(2000)10:4<352::AID-HIPO2>3.0.CO;2-D) PMID: [10985275](#)
79. Rigotti M, Barak O, Warden MR, Wang X-J, Daw ND, Miller EK, et al. The importance of mixed selectivity in complex cognitive tasks. *Nature*. 2013;497(7451):585–90. <https://doi.org/10.1038/nature12160> PMID: [23685452](#)
80. Cazettes F. Mixed selectivity: when neurons stopped looking like specialists. *Nat Rev Neurosci*. 2026;27(2):84. <https://doi.org/10.1038/s41583-025-01009-x> PMID: [41291071](#)
81. Vander Wall SB. Food Hoarding in Animals. The University of Chicago Press. 1990.
82. Clayton NS, Dickinson A. Memory for the content of caches by scrub jays (*Apelocoma coerulescens*). *J Exp Psychol Anim Behav Process*. 1999;25(1):82–91. <https://doi.org/10.1037/0097-7403.25.1.82> PMID: [9987859](#)
83. Jensen O, Lisman JE. Hippocampal sequence-encoding driven by a cortical multi-item working memory buffer. *Trends Neurosci*. 2005;28(2):67–72. <https://doi.org/10.1016/j.tins.2004.12.001> PMID: [15667928](#)
84. Kahana MJ, Wagner AD. Oxford Handbook of Human Memory. Oxford University Press. 2023.
85. Taub K, Abeles D, Yuval-Greenberg S. Evidence for content-dependent timing of real-life events during COVID-19 crisis. *Sci Rep*. 2022;12(1):9220. <https://doi.org/10.1038/s41598-022-13076-6> PMID: [35654909](#)
86. Kahana MJ, Diamond NB, Aka A. Laws of human memory. *PsyArXiv Preprints*. 2022. <https://doi.org/10.31234/osf.io/aczu9>

87. Bellmund JLS, Deuker L, Montijn ND, Doeller CF. Mnemonic construction and representation of temporal structure in the hippocampal formation. *Nat Commun.* 2022;13(1):3395. <https://doi.org/10.1038/s41467-022-30984-3> PMID: 35739096
88. Amit DJ, Fusi S. Learning in Neural Networks with Material Synapses. *Neural Computation.* 1994;6(5):957–82. <https://doi.org/10.1162/neco.1994.6.5.957>
89. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN. Attention Is All You Need. *ArXiv e-prints.* 2017.
90. Sukhbaatar S, Szlam A, Weston J, Fergus R. End-to-End Memory Networks. In: 2015. <http://arxiv.org/abs/1503.08895>
91. Miconi T, Clune J, Stanley KO. Differentiable Plasticity: Training Plastic Neural Networks with Backpropagation. 2018. <http://arxiv.org/abs/1804.02464>
92. Graves A, Wayne G, Reynolds M, Harley T, Danihelka I, Grabska-Barwińska A, et al. Hybrid computing using a neural network with dynamic external memory. *Nature.* 2016;538(7626):471–6. <https://doi.org/10.1038/nature20101> PMID: 27732574
93. Gershman SJ, Fiete I, Irie K. Key-value memory in the brain. *Neuron.* 2025;113(11):1694–1707.e1. <https://doi.org/10.1016/j.neuron.2025.02.029> PMID: 40147436
94. Limbacher T, Legenstein R. H-Mem: Harnessing Synaptic Plasticity with Hebbian Memory Networks. In: *Advances in Neural Information Processing Systems*, 2020. 21627–37. <https://proceedings.neurips.cc/paper/2020/hash/f6876a9f998f6472cc26708e27444456-Abstract.html>
95. Ferrand R, Baronig M, Unger F, Legenstein R. Non-synaptic plasticity enables memory-dependent local learning. *PLoS One.* 2025;20(3):e0313331. <https://doi.org/10.1371/journal.pone.0313331> PMID: 40096655
96. Chandra S, Sharma S, Chaudhuri R, Fiete I. Episodic and associative memory from spatial scaffolds in the hippocampus. *Nature.* 2025;638(8051):739–51. <https://doi.org/10.1038/s41586-024-08392-y> PMID: 39814883
97. Kohonen T. Correlation matrix memories. *IEEE Transactions on Computers.* 1972;C–21(4):353–9. <https://doi.org/10.1109/TC.1972.5008975>
98. Kuhn TS. *The structure of scientific revolutions.* Third edition ed. University of Chicago Press. 1996.
99. Liao Z, Losonczy A. Learning, Fast and Slow: Single- and Many-Shot Learning in the Hippocampus. *Annu Rev Neurosci.* 2024;47(1):187–209. <https://doi.org/10.1146/annurev-neuro-102423-100258> PMID: 38663090
100. Abraham WC. How long will long-term potentiation last?. *Philos Trans R Soc Lond B Biol Sci.* 2003;358(1432):735–44. <https://doi.org/10.1098/rstb.2002.1222> PMID: 12740120
101. Statman A, Kaufman M, Minerbi A, Ziv NE, Brenner N. Synaptic size dynamics as an effectively stochastic process. *PLoS Comput Biol.* 2014;10(10):e1003846. <https://doi.org/10.1371/journal.pcbi.1003846> PMID: 25275505
102. Bennett SH, Kirby AJ, Finnerty GT. Rewiring the connectome: Evidence and effects. *Neurosci Biobehav Rev.* 2018;88:51–62. <https://doi.org/10.1016/j.neubiorev.2018.03.001> PMID: 29540321
103. Squire LR, Genzel L, Wixted JT, Morris RG. Memory Consolidation. *Cold Spring Harbor Perspectives in Biology.* 2015;7(8):a021766. <https://doi.org/10.1101/cshperspect.a021766>
104. Bright IM, Meister MLR, Cruzado NA, Tiganj Z, Buffalo EA, Howard MW. A temporal record of the past with a spectrum of time constants in the monkey entorhinal cortex. *Proc Natl Acad Sci U S A.* 2020;117(33):20274–83. <https://doi.org/10.1073/pnas.1917197117> PMID: 32747574
105. Ahmed MS, Priestley JB, Castro A, Stefanini F, Solis Canales AS, Balough EM, et al. Hippocampal Network Reorganization Underlies the Formation of a Temporal Association Memory. *Neuron.* 2020;107(2):283–291.e6. <https://doi.org/10.1016/j.neuron.2020.04.013> PMID: 32392472
106. Mankin EA, Sparks FT, Slayyeh B, Sutherland RJ, Leutgeb S, Leutgeb JK. Neuronal code for extended time in the hippocampus. *Proc Natl Acad Sci U S A.* 2012;109(47):19462–7. <https://doi.org/10.1073/pnas.1214107109> PMID: 23132944
107. Ziv Y, Burns LD, Cocker ED, Hamel EO, Ghosh KK, Kitch LJ, et al. Long-term dynamics of CA1 hippocampal place codes. *Nat Neurosci.* 2013;16(3):264–6. <https://doi.org/10.1038/nn.3329> PMID: 23396101
108. Rubin A, Geva N, Sheintuch L, Ziv Y. Hippocampal ensemble dynamics timestamp events in long-term memory. *Elife.* 2015;4:e12247. <https://doi.org/10.7554/eLife.12247> PMID: 26682652
109. Mau W, Sullivan DW, Kinsky NR, Hasselmo ME, Howard MW, Eichenbaum H. *Current Biology.* 2018;28(10):1499–1508.e4. <https://doi.org/10.1016/j.cub.2018.03.051>
110. Rule ME, O'Leary T, Harvey CD. Causes and consequences of representational drift. *Curr Opin Neurobiol.* 2019;58:141–7. <https://doi.org/10.1016/j.conb.2019.08.005> PMID: 31569062
111. Sadeh S, Clopath C. Contribution of behavioural variability to representational drift. *Elife.* 2022;11:e77907. <https://doi.org/10.7554/eLife.77907> PMID: 36040010
112. 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.e5. <https://doi.org/10.1016/j.neuron.2016.12.026> PMID: 28132828
113. Senzai Y, Buzsáki G. Physiological Properties and Behavioral Correlates of Hippocampal Granule Cells and Mossy Cells. *Neuron.* 2017;93(3):691–704.e5. <https://doi.org/10.1016/j.neuron.2016.12.011> PMID: 28132824
114. Diamantaki M, Frey M, Berens P, Preston-Ferrer PF, Burgalossi A. Sparse activity of identified dentate granule cells during spatial exploration. *eLife.* 2016;5:e20252. <https://doi.org/10.7554/eLife.20252>

115. Chettih SN, Mackevicius EL, Hale S, Aronov D. Barcoding of episodic memories in the hippocampus of a food-caching bird. *Cell*. 2024;187(8):1922–1935.e20. <https://doi.org/10.1016/j.cell.2024.02.032> PMID: [38554707](https://pubmed.ncbi.nlm.nih.gov/38554707/)
116. Brea J, Clayton NS, Gerstner W. Computational models of episodic-like memory in food-caching birds. *Nat Commun*. 2023;14(1):2979. <https://doi.org/10.1038/s41467-023-38570-x> PMID: [37221167](https://pubmed.ncbi.nlm.nih.gov/37221167/)
117. Williams RJ. Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. *Machine Learning*. 1992;8(3–4):229–56. <https://doi.org/10.1023/a:1022672621406>
118. Surace SC, Pfister J-P, Gerstner W, Brea J. On the choice of metric in gradient-based theories of brain function. *PLoS Comput Biol*. 2020;16(4):e1007640. <https://doi.org/10.1371/journal.pcbi.1007640> PMID: [32271761](https://pubmed.ncbi.nlm.nih.gov/32271761/)