Remembering the “When”: Hebbian Memory Models for the Time of Past Events

Johanni Brea[1,2], Alireza Modirshanechi[1,2], Georgios Iatropoulos[1,2], and Wulfram Gerstner[1,2]

[1]School of Computer and Communication Science, École Polytechnique Fédérale de Lausanne, 1015 Lausanne EPFL, Switzerland

[2]School of Life Science, École Polytechnique Fédérale de Lausanne, 1015 Lausanne EPFL, Switzerland

Humans and animals can remember how long ago specific events happened. In contrast to interval-timing on the order of seconds and minutes, little is known about the neural mechanisms that enable remembering the “when” of autobiographical memories stored in the episodic memory system. Based on a systematic exploration of neural coding, association and retrieval schemes, we develop a family of hypotheses about the reconstruction of the time of past events, consistent with Hebbian plasticity in neural networks. We compare several plausible candidate mechanism in simulated experiments and, accordingly, propose how combined behavioral and physiological experiments can be used to pin down the actual neural implementation of the memory for the time of past events.

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 (Addyman et al., 2016; Buhusi & Meck, 2005; Carr & Konishi, 1990; Gerstner et al., 1996; Grothe et al., 2010; Issa et al., 2020; Paton & Buonomano, 2018; Tsao et al., 2022). For autobiographical memories, recall of the “when” information is often an explicit and conscious reconstruction-based process (Friedman, 1993), 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 (Friedman, 2013; Jelbert & Clayton, 2017; Pathman et al., 2013). 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 (Jelbert & Clayton, 2017).

On timescales from milliseconds to minutes, multiple mechanisms based on changing neuronal activity patterns are known to support accurate interval timing (Addyman et al., 2016; Buhusi & Meck, 2005; Issa et al., 2020; Paton & Buonomano, 2018; Tsao et al., 2022). Less 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 synaptic plasticity and possibly on systems consolidation (Moscovitch & Gilboa, 2021). Multiple research communities have developed models of episodic memory to explain recall of past events (Norman et al., 2008). These models focus on different aspects, like replicating behavioral data in free or serial recall of lists (Howard, 2022; Howard & Kahana, 2002; Kahana, 2020; Kahana, 2012; Katkov & Tsodyks, 2022; Polyn et al., 2009; Romani et al., 2013), developing attractor neural networks consistent with anatomical and physiological knowledge of the hippocampal formation (Kesner & Rolls, 2015; Norman & O’Reilly, 2003; Norman et al., 2008), or explaining systems consolidation (Moscovitch & Gilboa, 2021; Remme et al., 2021; Squire et al., 2015). Although attempts at categorizing different theories have been made (Friedman, 1993; Henson, 1998), there exists no systematic exploration of computational models that focus on the time of past events.

Here, we study from a theoretical perspective different neural mechanisms that enable *automatic* reconstruction of the time of past events on long timescales. Whereas theoretical consideration about the representation of temporal information are in principle 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 (Figure 1). Our goal is to develop concrete hypotheses about biologi-

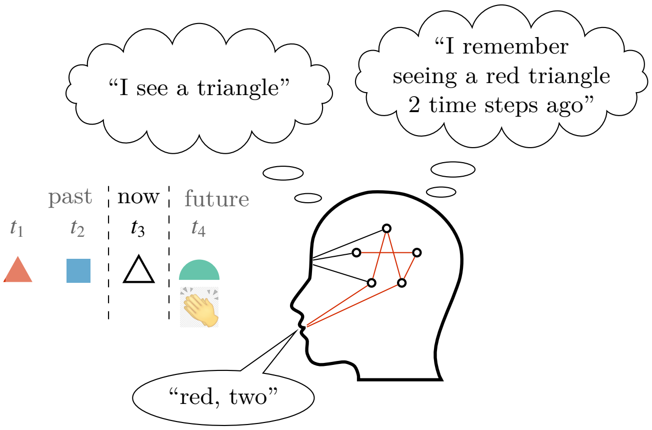


Figure 1

A simple setting to study episodic-like memory. *We consider discrete sensory streams, like 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 $t_1, t_2, \ldots$ 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, like saying "red, two". In the next time step, a new stimulus can be given, together with a reinforcement signal (clapping hands). As a typical example similar to the abstract setting described here, 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 (Clayton & Dickinson, 1998). 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.*

cal neural circuits and Hebbian synaptic plasticity rules that support this behavior.

Results

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, like the sets of colors, shapes and time points in Figure 1. 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 Figure 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, are idealizations that are 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.

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 (Figure 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 it allows to infer the time of a given event and 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 ago". In contrast to timestamp representations of time, age coding implies that the neu-

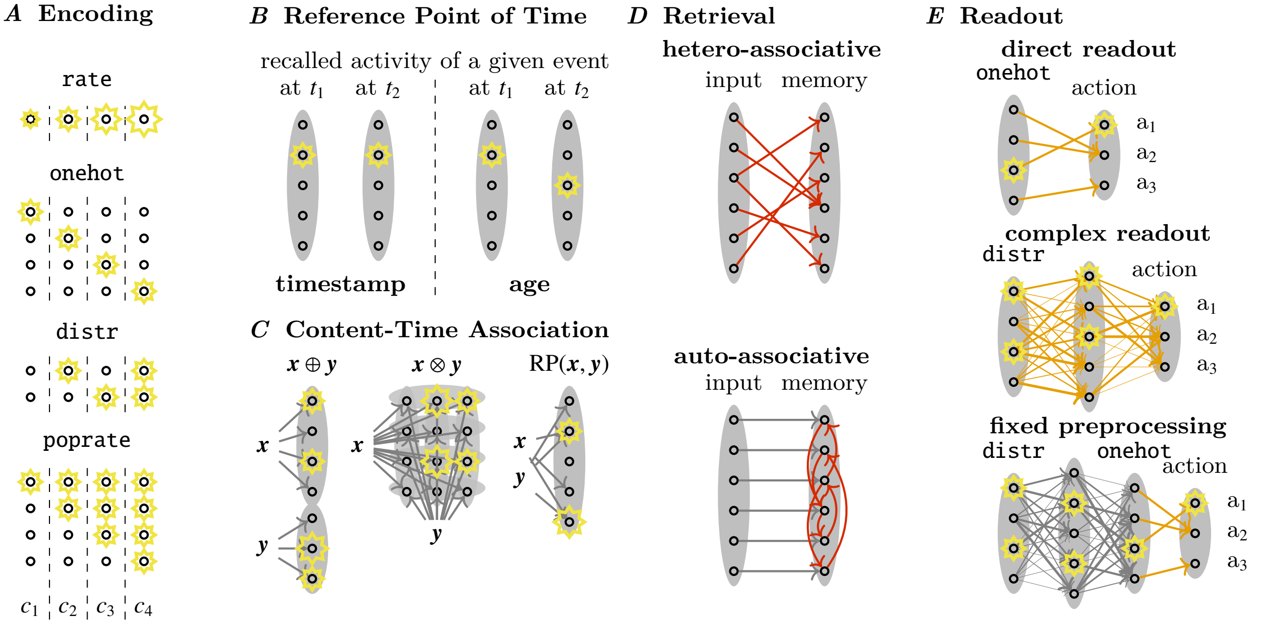


Figure 2

Several distinct "what-where-when" memory systems can be constructed by combining specific choices of the encoding, reference point of time, association, retrieval 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 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 $\boldsymbol{x}$ and $\boldsymbol{y}$, about e.g. the content and the time of an event, can be associated in multiple ways; for example, with concatenation $\boldsymbol{x} \oplus \boldsymbol{y}$ (left) or non-linear mixed codes, like the (outer) product $\boldsymbol{x} \otimes \boldsymbol{y}$ (middle) or random projection* RP($\boldsymbol{x}, \boldsymbol{y}$) *(right) 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*** *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.*

ral 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. An example of an age code is shown in Figure 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 to be memorized. 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 like the ticking of a clock, the day-night cycle or the change of seasons; also processes that involve the agent's actions, like 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.

Associating Information: How to Combine What, Where and When

The be able to remember everything about a given event, the "what", "where" and "when" information need 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 "loose" any information, in the sense that the content and "when" information can be faithfully reconstructed from the momentary neural activity pattern. In the following we focus on three specific examples of association functions: concatenation, (outer) product and random projection codes (Figure 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 (⊕ in Figure 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 allow to represent the content and "when" information in a concatenation code. An example of a *non-linear mixed code* is the *product code* (⊗ in Figure 2C), 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 (Smolensky, 1990). 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 (Kelly et al., 2013) or *random projections* (RP in Figure 2C), 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 "Simulations".

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.

Retrieval

In the field of neural networks, memory retrieval is typically implemented with hetero- or auto-associative memories (Figure 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 (Amit, 1989). 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 (Iatropoulos et al., 2022).

Storage and Retrieval Phases

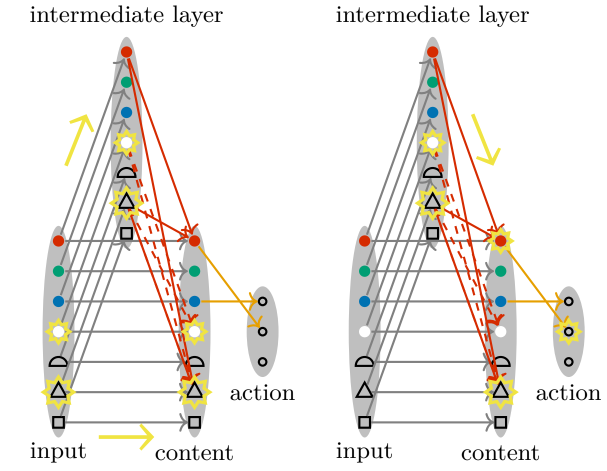
Most models require separate storage and retrieval phases. Suppose "red triangle" has already been stored at time step t_1. At time t_3, the input "white triangle" triggers recall of the stored memory, i.e. the neural code for the "when" information t_1 together with the content information "triangle" 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). Separate storage and retrieval phases could be implemented by 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, like neurotransmitters, could drive such a periodic process (Hasselmo et al., 2002).

Alternatively, the separation into storage and retrieval phases could be implemented with synaptic delays: if multiple pathways exist between two groups of neurons, for example the "input-content" pathway and the "input-intermediate-content" pathway in Figure 3, and if information travels at different speeds along the different pathways, then stimulus-driven activation through one pathway could be used for storage ("input-content" pathway in Figure 3) and through the other pathway for retrieval ("input-intermediate-content" pathway in Figure 3). In the simulated models in section "Simulations" we use this mechanism to distinguish storage and retrieval phases.

Behavioral Readout

Once a previously stored activity pattern is retrieved, it can trigger some behavioural 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 behavioural output, which may have a very different representation than the input that led to the formation of the memory (e.g. visual input and vocal output in Figure 1). 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. How easily different behavioural 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 "Simulations". For a strict one-hot code, for example the product of one-hot codes $\texttt{onehot}(\text{content})\otimes\texttt{onehot}(\text{age})$, any behavioural rule that maps content and age of a recalled event to a given action can be learned with *direct readout*, i.e. plastic connections between the layer of recalled activity to action neurons (Figure 2E). 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 (Hertz et al., 1991), where a certain action is taken if and only if two input neurons are jointly active or jointly inactive, cannot be learned with direct readout, but it can be learned with a multilayer perceptron (Figure 2E). Learning all connections in a multilayer perceptron can be achieved with the backpropagation algorithm or biologically plausible variants thereof (Illing et al., 2019; Lillicrap et al., 2016; Roelfsema & Ooyen, 2005), 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 (Figure 2E).



Storage of white triangle

Recall of red triangle triggered by white triangle

Figure 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 yellow arrows). 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 (yellow arrow). 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).*

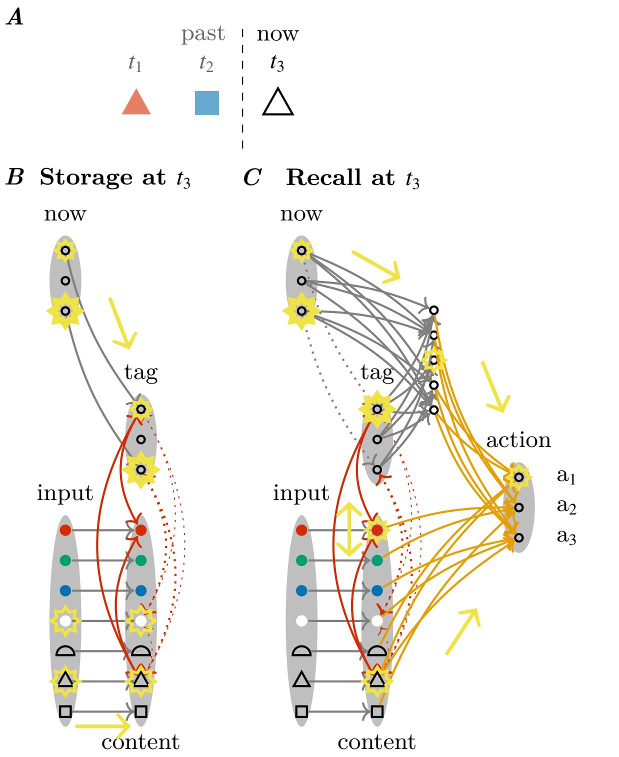


Figure 4

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 current context (activity in layer "now") drives the "tag" neurons (yellow arrow) 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 (yellow arrow), 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. Consequently, behavioural rules that depend on the age of the recalled memory can be learned (orange weights).*

Synaptic Plasticity

Long-term synaptic changes are presumably involved for memorization in the storage phase, for learning actions to indicate successful retrieval and to reflect the passage of time in age representations of time.

If pre- and postsynaptic neurons are jointly active during the storage phase, Hebbian synaptic plasticity is sufficient to memorize and generate a trace of the event in the memory system. Behavioural rules that depend on the content and the age of recalled memories could be learned with neoHebbian synaptic plasticity (Gerstner et al., 2018; Kuśmierz et al., 2017; Lisman et al., 2011; Magee & Grienberger, 2020; Roelfsema & Holtmaat, 2018), where jointly active neurons generate an eligibility trace that is modulated by a subsequent signal. This modulating signal could communicate the reward received after a successful action.

For age representations of time, synaptic growth or decay could reflect the passage of time. Examples of how this could be achieved are discussed in the next section.

For timestamp representations of time, the synaptic changes for memorization and behavioural learning are sufficient. Although this is an appealing advantage of timestamp representations of time, it comes at the cost of an increased complexity to compute 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.

Examples of Episodic-Like Memory Systems

With four different encoding schemes (Figure 2A), two different ways of representing time (Figure 2B), three different codes for associating content and time (Figure 2C), two different memory retrieval mechanisms (Figure 2D), three different readout architectures (Figure 2E), and two different storage-retrieval mechanisms (synaptic delays Figure 3 or periodic processes, like e.g. proposed by Hasselmo et al., 2002) 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 details six specific examples that are representative of the different kinds of models (Table 1). Detailed mathematical descriptions of these models can be found in section "Mathematical Description of the Models".

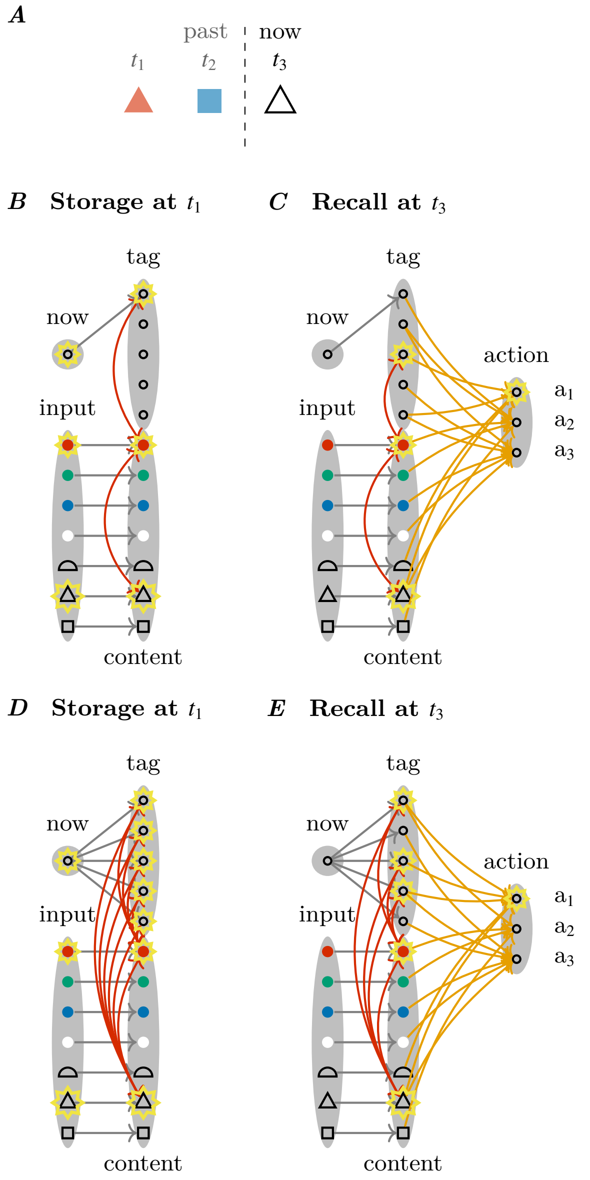


Figure 5

Age tagging models. ***A*** *We consider the same sequence of events as in Figure 4A.* ***B*** *In the* ***Onehot-Age-Tagging*** *model, a connection to the first tag neuron is formed during storage of event "red triangle" 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.*

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 Figure 4) with another subgroup of neurons that represent "when" information ("tag" in Figure 4).

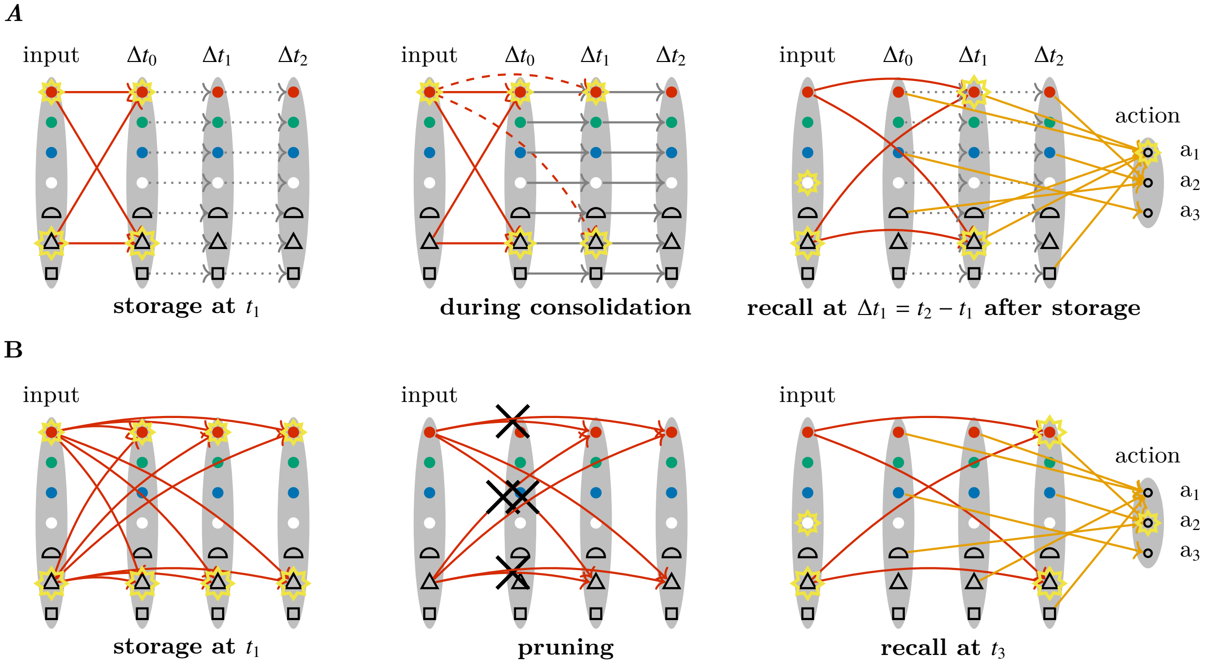
In the spirit of (temporal and random) context models (Howard, 2022; Howard & Kahana, 2002; Polyn et al., 2009), the "when" information could be given implicitly by activity patterns that encode context ("now" in Figure 4). This context could be a trace of recent observations of states that change on different timescales like emotional states, the presence of certain conspecifics, ambient temperature or the weather. 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 (Figure 4B). If an event triggers the recall of an earlier event (Figure 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, Friedman, 1993). Because the estimation of age from the comparison of two context-related activity patterns does, in general, not induce a linearly separable problem, a complex readout network with at least one hidden layer is required, to learn arbitrary readout rules.

Age Tagging Models

Other tagging models can be constructed with one-hot coding or population-rate coding for the age of memories (Figure 2A). In the Onehot-Age-Tagging model, storage leads to the formation of a synaptic connection to the first tag neuron (Figure 5B). It is hypothesised that specific circuits allow to change the representation of the memory by growing new synapses and pruning old ones (Remme et al., 2021; Roxin & Fusi, 2013). 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 (Figure 5C).

In the Poprate-Age-Tagging model, storage leads to the formation of many synaptic connections to several "tag" neurons (Figure 5D). These connections are pruned at different moments in time. Therefore, the number of activated tag neurons during recall is indicative of the elapsed duration between storage and recall (Figure 5E).

In contrast to the Context-Tagging model with a timestamp code (Figure 4), the representation of time changes in the Onehot-Age-Tagging model and the



storage at t_1 **during consolidation** **recall at $\Delta t_1 = t_2 - t_1$ after storage**

storage at t_1 **pruning** **recall at t_3**

Figure 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 Δt_1), such that new, direct-pathway connections (dashed red) can grow between input and Δt_1 neurons. Simultaneously, the original weights between input and Δt_0 neurons decay, such that after consolidation only the newly grown weights remain. During recall (right), all active neurons in layers $\Delta t_0, \Delta t_1, \Delta t_2, \ldots$ 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 (Figure 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.*

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. One can even prove (see Equivalence of One-Hot coding and Deterministic Population-Rate coding) that 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.

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 (⊗ in Figure 2). This leads to a class of models with a chronological organization of memories (Friedman, 1993).

The chronological organization is most obvious, when arbitrarily encoded content information and one-hot encoded age information is 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 (Figure 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

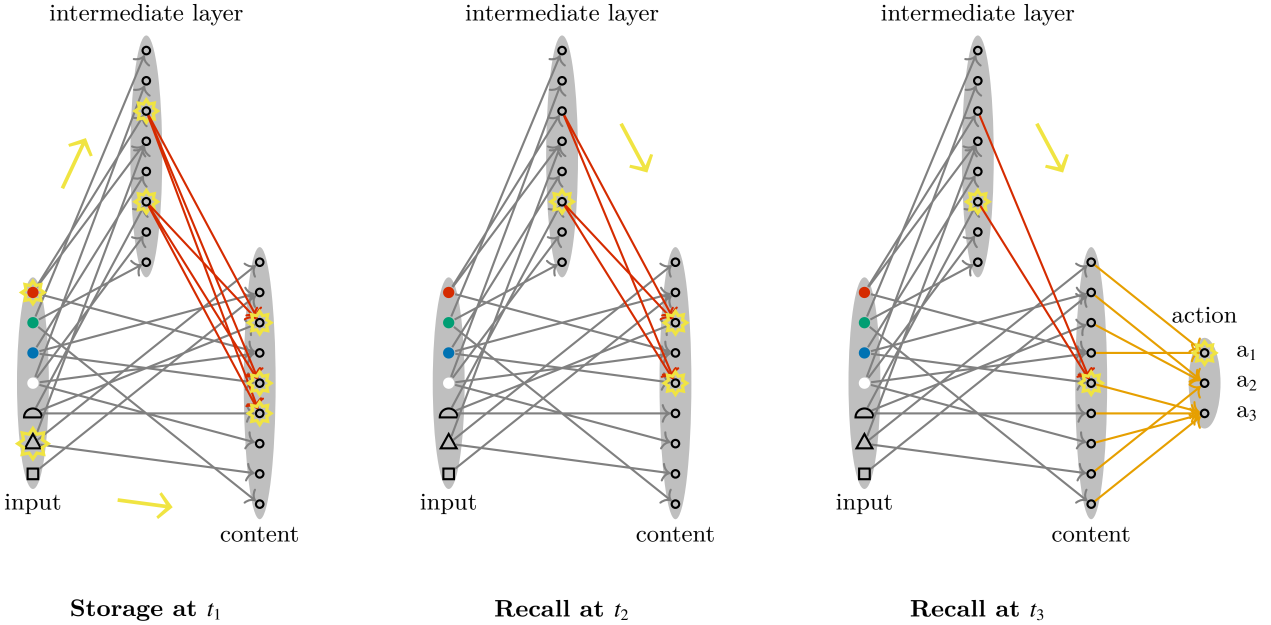


Figure 7

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, Hebbian plasticity connects co-activated neurons (red arrows). 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).*

groups Δt_0, Δt_1, Δt_2 in Figure 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 of the memory: if recall happens some time interval Δt_1 after storage, the content neurons in group Δt_1 become active, whereas the neurons in other groups become active during recall at other times (Figure 6A).

Such an age code requires rewiring 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 allowing to learn direct-pathway connections (Figure 6A, cf. parallel pathway theory, Remme et al., 2021). As a result of synaptic plasticity, the location of the memorized event inside the memory system shifts forward over time (Figure 6A), inducing a "chronological" organization of memories.

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 (Figure 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 Figure 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 (Figure 6B).

Sparse Encoding with Random Synaptic Pruning and Simple Readout

Despite the frequent appearance of the special association schemes "concatenation" and "product" in the literature (⊕ and ⊗ in Figure 2C), it is unclear why brains should favor them over other non-linear mixed association schemes.

Combining sparse random projections for association (RP in Figure 2C) with synaptic delays for storage and recall (Figure 3) and synaptic pruning (Figure 5E and Figure 6B) leads to the Random-Pruning model (Figure 7). During storage, the input triggers distributed activity patterns in the intermediate and content neurons. Because of the random projections, these activity patterns encode the input information implicitly, i.e. the neurons in these groups are not necessarily "tuned" to a single feature, like 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 Figure 7).

In the model of Figure 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 (Figure 5E) and the chronological organization model with synaptic pruning (Figure 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 Figure 7), than when recalling an old event (recall at t_3 in Figure 7).

Simulations

To highlight advantages and disadvantages of different systems and explore the limitations of purely behavioural experiments as a tool to learn about how brains allow to remember the "when" of past events, we simulated different "what-where-when" memory models.

For the Context-Tagging model we assume a fixed preprocessing to a one-hot intermediate representation of the age of a memory (Figure 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, like performing action a_2 when the event "red triangle" is remembered to have happened Δt_{train} ago (Figure 8A, 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 (Clayton et al., 2003). If one tests the subject on untrained content-age combinations, for example "blue square" after Δt_{test} (Figure 8A), 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 Δt_{train} ago" and "blue square Δt_{test} ago" for any Δt_{test} (light blue curve in Figure 8B). In the Context/Onehot-Tagging model, the tags for "red triangle Δt_{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 Figure 8B). 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 (Figure 9A, see also section "Protocols of Simulated Experiments"). Such a task can be learned with all the models considered here, but the Age-Organization model has potentially an advantage, because the one-hot encoding permits faster learning with higher learning rates than other representations (Figure 9B; learning rates for all models are optimized for best final performance in the tasks in Figure 9A and Figure 10A). However, if an experiment would show 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.

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 (Figure 10A, 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 (Figure 10B). We find that tagging models systematically fail on this task (yellow and red curve in Figure 10B). 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 Figure 9B).

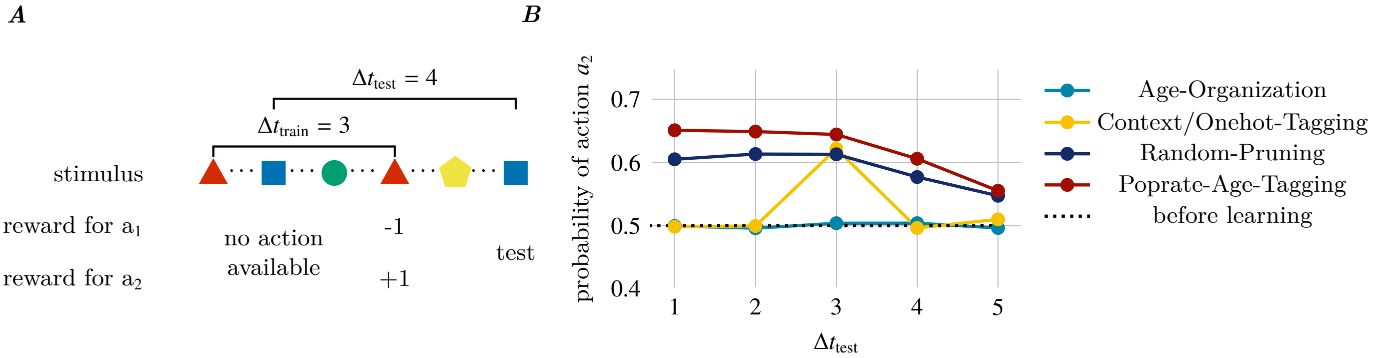


Figure 8

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 Δt_{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 Δt_{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 Δt_{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 $\Delta_{\text{test}} = 3$). Conversely, for distributed representations (Poprate-Age-Tagging and Random-Pruning) there is generalization to other stimuli and test intervals different than 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.*

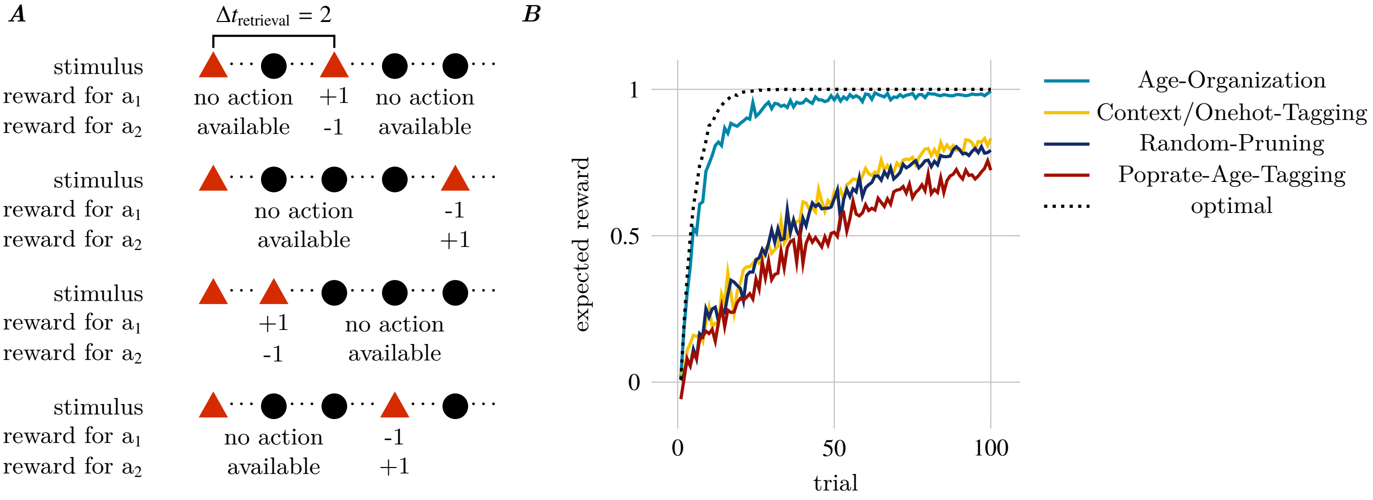


Figure 9

Learning to take actions that depend only on the age of memories. A *The experiment consists of multiple trials with random retrieval intervals $\Delta_{\text{retrieval}}$. If the retrieval interval satisfies $\Delta_{\text{retrieval}} \leq 2$, 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 10^4 agents that make for each interval at most one mistake and always select the correct action afterwards.*

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.

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. The concrete implementations are idealized "toy"-models, that illustrate the central ideas succinctly.

The considered neural codes (rate, one-hot, distributed and poprate, Figure 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 spatial-temporal codes, more sophisticated readout and learning mechanisms than the ones in Figure 2E 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 "Simulations") would further increase and include models with spatio-temporal retrieval (e.g. Jensen and Lisman, 2005).

Memory storage and learning of new tasks in the proposed models rely on Hebbian and neoHebbian synaptic plasticity, for which there is ample experimental evidence (Gerstner et al., 2018; Kuśmierz et al., 2017; Lisman et al., 2011; Magee & Grienberger, 2020; Roelfsema & Holtmaat, 2018). 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 (Abraham, 2003; Statman et al., 2014). Rewiring of networks in one-hot or distributed encoding of the age of memories is consistent with the observed phenomena of rewiring of connections (Bennett et al., 2018) and systems memory consolidation (Moscovitch & Gilboa, 2021; Squire et al., 2015) and could be achieved with the hypothesised mechanisms of parallel synaptic pathways (Remme et al., 2021) or memory transfer (Roxin & Fusi, 2013). Although there is experimental evidence for all the synaptic processes needed to implement the above models, further experiments should be done to determine which processes are actually used to remember the time of past events.

Chronological organization models can be generalized to include models where the different groups of content neurons are not just copies of one another, but the representations of the memory content in each group differ from on 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 (Moscovitch & Gilboa, 2021), which postulates that recall of details requires a functional hippocampus, whereas the gist can be recalled without hippocampus.

On a conceptual level, multiple theories of the processing of "when" information have been proposed. For example, Friedman, 1993 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 on readout of temporal and content information.

Theory-ladenness of observations (Kuhn, 1996) together with the richness of biological phenomena allows to find support in experimental data for different theories, such as chronological organization as proposed by theories of systems consolidation (Moscovitch & Gilboa, 2021; Squire et al., 2015), time tagging models with context tags that decay on different timescales (Bright et al., 2020; Tsao et al., 2018), time tagging models based on intrinsic oscillators (Rolls & Mills, 2019), time tagging models with hippocampal CA1 time cells (Eichenbaum, 2017) and hippocampus-dependent reconstruction-based theories (Bellmund et al., 2022).

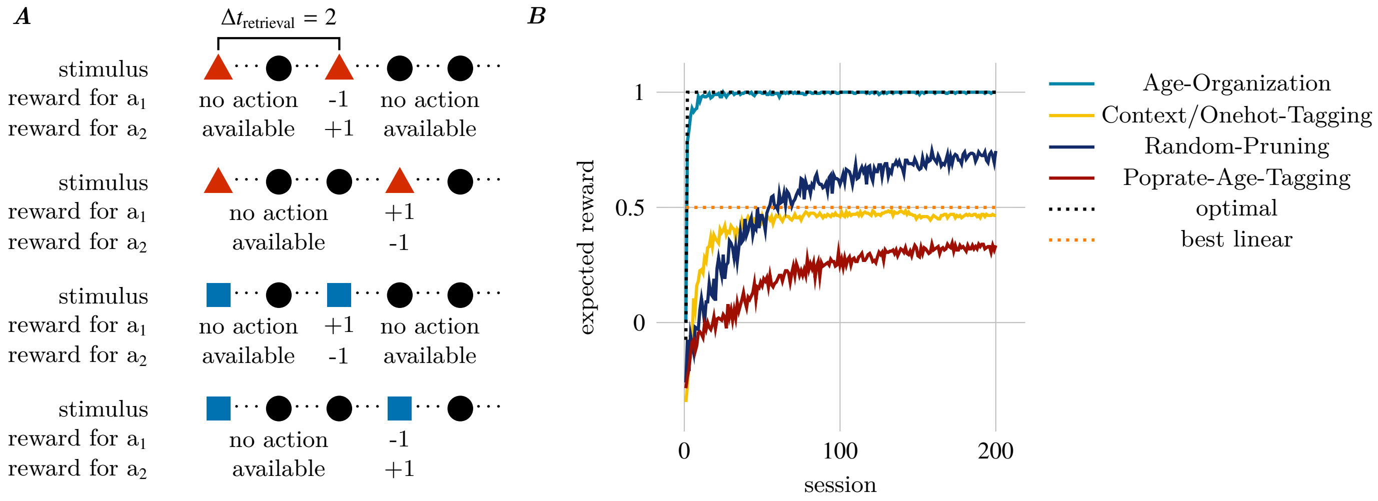


Figure 10

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* $\Delta_{\text{retrieval}}$ *and objects. If the object is red and the retrieval interval is* $\Delta_{\text{retrieval}} = 2$ *or the object is blue and* $\Delta_{\text{retrieval}} = 3$*, 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 slower than the Age-Organization model encoding and sufficiently large learning rate. The optimal performance (dashed line) was computed by averaging* 10^2 *agents that make for each interval and content at most one mistake and always select the correct action afterwards.*

name	time code	association	retrieval & readout	comments
Context-Tagging	timestamp distributed	concatenation	auto-associative & complex	see Figure 4
Onehot-Age-Tagging	age one-hot	concatenation	auto-associative & simple	see Figure 5C
Poprate-Age-Tagging	age pop-rate	concatenation	auto-associative & simple	see Figure 5E
Age-Organization	age one-hot	(outer) product	hetero-associative & simple	see Figure 6A
Poprate-Age-Organization	age pop-rate	(outer) product	hetero-associative & simple	see Figure 6B
Random-Pruning	age pop-rate	random projection	hetero-associative & simple	see Figure 7
Temporal Context (TCM)	timestamp distributed	concatenation	Recall is hetero-associative with a stochastic winner-takes-all mechanism. Readout learning is not modelled.	see Howard, 2022; Howard and Kahana, 2002; Polyn et al., 2009. The age of memories could be retrieved using, either, the current context and reconstructing the contextual overlap, or, the explicit exponential decay of memory traces in the context tags.
Complementary Learning Systems	age distributed	(lossy outer) product: at storage the content is redundantly encoded in hippocampal and cortical activity	Recall is auto-associative. Readout learning is not modelled.	See e.g. Kesner and Rolls, 2015; McClelland et al., 1995; Moscovitch and Gilboa, 2021; Remme et al., 2021; Squire et al., 2015; Treves and Rolls, 1994. The age of a memory can roughly be estimated by how much the hippocampus is involved and needed for memory retrieval.

Table 1

Examples of episodic-like memory systems. *Models above the horizontal line are discussed in the text. Below the horizontal line are two broad classes of episodic-memory models that allow to reconstruct the "when" information of past events, although they do not specifically focus on this aspect.*

It is reasonable to assume that different mechanisms are at play in different species and even within the same individual for tasks on different timescales (Wiener et al., 2011).

Although it is unclear, what exactly should be considered as single events that are stored in episodic memory (Zacks, 2020), the popular hypothesis that events are stored together with context information has led to numerous studies. Most experiments with physiological recordings focus on short timescales in the range of seconds to minutes, where memory for the "when" can be supported by neural activity mechanisms which are presumably independent of long-term synaptic plasticity (Bright et al., 2020; Tsao et al., 2018). More relevant for the topic of this paper are experiments involving longer timescales. For example, Rubin et al., 2015 recorded for two weeks calcium levels in more than 1000 neurons in the CA1 of mice that were placed every day in two environments with distinct features. They found that the correlation of population activity between different contexts on subsequent days is higher than between the same context on distant days (Fig. 2A in Rubin et al., 2015; the opposite is observed, however, if the analysis is restricted to a subset of cells that they identified as pure place cells). This surprising result would be consistent with a timestamp mechanism similar to the Context-Tagging model. However, it is unclear, if the population activity reflects recalled events or if the measured calcium signal is dominated by stimulus-driven input. The surprising result could simply be a consequence of representational drift (Rule et al., 2019), which may be unrelated to the storage and recall of events and depend, for example, on behavioral variability (Sadeh & Clopath, 2022).

Also some purely behavioral studies with human participants are consistent with the contextual overlap theory implemented in the Context-Tagging model. For example, Taub et al., 2022 asked human participants during the first Covid-lockdown in April 2020 in Israel, to retrospectively estimate the time passed since prominent news events happened that were related or unrelated to Covid. They found that participants underestimated the elapsed duration since Covid-related events more than other durations, which is consistent with the contextual overlap theory, as the context at the time of recall during the lockdown was Covid-dominated (Taub et al., 2022). Whether this should be taken as evidence for the simple Context-Tagging model is unclear, however, as healthy adult humans are believed to rely on concious reconstruction-based estimates of the time of past events (Friedman, 1993), as illustrated in the introduction with the example of the journey to Turkey.

Behavioral experiments with *California scrub-jays* showed convincingly that these birds have a flexible "what-where-when" memory system (Brea et al., 2023; Clayton & Dickinson, 1999; Clayton et al., 2001, 2003, 2005). Although the existing experimental results cannot discriminate between the different models considered here, the observation that they can learn different behavioral rules that depend on the content and age of memories within few trials, speaks in favor of the Age-Organization model, that allows fast and flexible learning.

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.

Methods

Formal Description of Codes

We consider a discretized version of the sensory stream in $\prod_{i=1}^{N} C_i \times \mathcal{T}$, where 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 $C = \{c_1, \ldots, c_{|C|}\}$ with elements c_i and cardinality $|C|$, we define three different elementary coding schemes for representing element c_i:

- *rate code*: $\mathtt{rate}(c_i) := r(i) \in \mathbb{R}$, where r is an arbitrary, one-to-one function.
- *one-hot code*: a $|C|$-tuple-valued function $\mathtt{onehot}(c_i) := (\mathtt{onehot}_1(c_i), \ldots, \mathtt{onehot}_{|C|}(c_i))$ with $\mathtt{onehot}_j(c_i) := A_i \delta_{ij}$, 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 $\mathtt{distr}(c_i) := (\mathtt{distr}_1(c_i), \ldots, \mathtt{distr}_M(c_i))$ with $M > 1$ and $0 < \mathtt{distr}_j(c_i) \leq \mathtt{distr}_{j'}(c_i)$ for at least one pair $j \neq j'$ and at least one i (Figure 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 $\mathtt{distr}_j(c_i) = 1$ if $j \leq i$ and $\mathtt{distr}_j(c_i) = 0$ otherwise ($\mathtt{poprate}$ in Figure 2A). Stochastic population rate codes satisfy the condition $\Pr(\sum_j \mathtt{distr}_j(c_i) = i) = 1$.

These codes can be reduced to a rate code by computing the population activity $\mathtt{rate}(c_i) = i = \sum_{j=1}^{|C|} \mathtt{distr}_j(c_i)$.

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

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

Formal Description of Association Schemes

Any function $f : \mathcal{X} \times \mathcal{Y} \to \mathcal{Z}$ 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*: $\boldsymbol{x} \oplus \boldsymbol{y} := (x_1, \dots, x_{\ell(x)}, y_1, \dots, y_{\ell(y)})$ (Figure 2C).
- *linear mixed codes*: $\boldsymbol{x} \oplus_M \boldsymbol{y} := M(\boldsymbol{x} \oplus \boldsymbol{y})$, where $M : \mathbb{R}^{\ell(\boldsymbol{x})+\ell(\boldsymbol{y})} \to \mathbb{R}^M$ is a linear map.
- *product of codes*: $\boldsymbol{x} \otimes \boldsymbol{y} := \boldsymbol{x} \cdot y_1 \oplus \boldsymbol{x} \cdot y_2 \oplus \cdots \oplus \boldsymbol{x} \cdot y_{\ell(x)}$, where the product $\boldsymbol{x} \cdot y_i$ between a tuple $\boldsymbol{x}$ and a scalar y_i has the standard meaning $\boldsymbol{x} \cdot y_i = (x_1 y_i, \dots, x_{\ell(x)} y_i)$ (Figure 2C)
- *random projections*: $\mathrm{RP}(\boldsymbol{x}, \boldsymbol{y}) = \sigma(W_1 \boldsymbol{x} + W_2 \boldsymbol{y})$, where W_1 and W_2 are fixed random matrices and σ is some non-linear function that is applied element-wise.

Mathematical Description of the Models

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

For all the models we used the storage and recall mechanism with synaptic delays, described in Figure 3 and hetero-associative recall. For the tagging models, this implementation differs from the descriptions with auto-associative recall in Figure 4 and Figure 5, but it leads to the same predictions for the behavioral experiments.

The internal neural states z_τ are organized into groups of neurons. Tagging models have sensor, intermediate, content, tag and actuator neurons and we write the neural state $z_\tau = z_\tau^{\text{sensor}} \oplus z_\tau^{\text{intermediate}} \oplus z_\tau^{\text{content}} \oplus z_\tau^{\text{tag}} \oplus z_\tau^{\text{actuator}}$. 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 μ, which is given by

$$z_\tau^{(\mu)} = \sigma^{(\mu)}(W_{\tau-1}^{\nu\to\mu} z_{\tau-1}^{(\nu)}), \tag{1}$$

where $\sigma^{(\mu)}$, the activation function of neurons in group μ, is applied element-wise to the matrix-vector product of synaptic weight matrix $W_{\tau-1}^{\nu\to\mu}$ and activity state $z_{\tau-1}^{(\nu)}$ of group ν in the previous time-step $\tau-1$. The activation function of the actuator group is the soft-max function $\sigma^{\text{actuator}}(\boldsymbol{x})_i = e^{x_i} / \sum_j e^{x_j}$. For all other groups of neurons we use the Heaviside function $\sigma^{(i)}(\boldsymbol{x})_i = H(x_i - b) = 1$ if $x_i > b$ and $H(x_i - b) = 0$, otherwise, where bias $b = 0$ for all groups of neurons except for the content group in the Random-Pruning model, where $b = 1.5$. The non-zero bias in the Random-Pruning assures sparse activity in the content layer. Action a_t is sampled with probability z_τ^{actuator} after recall has happened.

Because we used the recall mechanism with synaptic delays (Figure 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 τ: if $z_\tau^{\text{sensor}} = \mathtt{onehot}(\mathtt{color}(x_t)) \oplus \mathtt{onehot}(\mathtt{shape}(x_t))$, action a_t will be sampled with probability $z_{\tau+3}^{\text{actuator}}$. The sensory neurons are inactive during propagation of the activity through the neural network, i.e. $z_{\tau+1}^{\text{sensor}} = z_{\tau+2}^{\text{sensor}} = z_{\tau+3}^{\text{sensor}} = \mathbf{0}$; the sensory neurons are reactivated, once action a_t has been taken, i.e. $z_{\tau+4}^{\text{sensor}} = \mathtt{onehot}(\mathtt{color}(x_{t+1})) \oplus \mathtt{onehot}(\mathtt{shape}(x_{t+1}))$.

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

Hebbian

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

where i is the postsynaptic neuron in group μ and j the presynaptic neuron in group ν.

Reward-Modulated Hebbian

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

$$\Delta W_{\tau,i,j}^{\nu\to\text{actuator}} = W_{\tau,i,j}^{\nu\to\text{actuator}} - W_{\tau-1,i,j}^{\nu\to\text{actuator}} = \eta r_t E_{\tau-1,i.j}^{\nu\to\text{actuator}} \tag{4}$$

where $E_\tau^{\nu\to\text{actuator}}$ 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 $\tilde{z}^{\text{actuator}}_{\tau,j} = -z^{\text{actuator}}_{\tau,j}$, otherwise, η 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 (Williams, 1992), where the terms $\tilde{z}^{\text{actuator}}_{\tau,i}$ arise as a consequence of taking the derivative of the logarithm of the soft-max policy function in the derivation of the REINFORCE rule. This plasticity rule is used in all models for the readout weights (orange in the figures).

Hebbian Latent-State-Decay

$$S^{\nu\to\mu}_{\tau,i,j} = \left[S^{\nu\to\mu}_{\tau-1,i,j} + s_i^{\max} z^{(\mu)}_{\tau-1,i} z^{(\nu)}_{\tau-1,j} - \Delta s\right]_0^{s_i^{\max}} \tag{5}$$

$$W^{\nu\to\mu}_{\tau,i,j} = H(S^{\nu\to\mu}_{\tau,i,j}) \tag{6}$$

where $S^{\nu\to\mu}_{\tau}$ 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, Δs 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.

Postsynaptic Rewiring

$$W^{\nu\to\mu}_{\tau,i,j} = W^{\nu\to\mu}_{\tau-\Delta\tau,i-1,j}, \tag{7}$$

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. Figure 4C or Figure 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.

Equivalence of One-Hot coding and Deterministic Population-Rate coding

Let $\boldsymbol{x}$ be a one-hot coded neural activity pattern, $\boldsymbol{w} \in \mathbb{R}^N$ a weight vector, $y = \boldsymbol{w}^T\boldsymbol{x}$ a linear readout, and $\tilde{\boldsymbol{x}} = P\boldsymbol{x}$, with P such that $\tilde{x}_i = \sum_{j=1}^N x_j$, the reparametrization from one-hot coding to the deterministic population rate code at the bottom of Figure 2A. The corresponding transformation $\tilde{\boldsymbol{w}} = (P^{-1})^T\boldsymbol{w}$ leaves the response invariant, i.e. $\tilde{y} = \tilde{\boldsymbol{w}}^T\tilde{\boldsymbol{x}} = y$. 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 $i = 2, \ldots, N-1$ 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 $(P^{-1})^T(P)^{-1}$ (Surace et al., 2020). If the synapses are spatially organized such that the inputs of presynaptic neurons $i-1, i, i+1$ are neighbouring, the resulting plasticity rule features cross-talk between neighbouring synapses. In particular, a synaptic weight $\tilde{w}_i$ should only be changed, if the inputs at the neighbouring synapses $i-1$ and $i+1$ differ from the input at synapse i.

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.

Generalization of actions (Figure 8)

```
results = DataFrame(a2_test = [])
for subject in subjects
    show_to!(subject, red_triangle)
    show_to!(subject, blue_square)
    show_to!(subject, green_circle)
    show_to!(subject, red_triangle)
    a_train = force_to_choose_and_observe_action!(subject)
    if a_train == a_1
        reward!(subject, -1)
    else
        reward!(subject, +1)
    end
    delta_test = rand([1, 2, 3, 4, 5])
    if delta_test == 1
        show_to!(subject, yellow_pentagon)
        show_to!(subject, yellow_pentagon)
    elseif delta_test == 2
        show_to!(subject, green_circle)
    elseif delta_test == 3
        show_to!(subject, blue_square)
    elseif delta_test == 4
        show_to!(subject, yellow_pentagon)
        show_to!(subject, blue_square)
    else
        show_to!(subject, yellow_pentagon)
        show_to!(subject, orange_halfcircle)
        show_to!(subject, blue_square)
    end
    a_test = force_to_choose_and_observe_action!(subject)
    push!(results, is_equal(a_test, a2))
end
# compute average number of a2 responses
```

Learning to take age-dependent actions (Figure 9)

```
results = DataFrame(trial = [], reward = [])
for subject in subjects
    for trial in trials
        show_to!(subject, red_triangle)
        delta = rand([1, 2, 3, 4, 5])
        for i in 1:delta-1
            show_to!(subject, black_circle)
        end
        show_to!(subject, red_triangle)
        a = force_to_choose_and_observe_action!(subject)
        if (delta <= 2 && a == a_2) || (delta > 2 && a == a_1)
            reward!(subject, +1)
            push!(results, [trial, +1])
        else
            reward!(subject, -1)
            push!(results, [trial, -1])
        end
        # do something else before new trial starts
    end
end
# compute average reward per trial
```

Learning to take age- and content-dependent actions (Figure 10)

```
results = DataFrame(session = [], reward = [])
for subject in subjects
    for session in sessions
        r_per_session = 0
        # trial 1
        show_to!(subject, red_triangle)
        show_to!(subject, black_circle)
        show_to!(subject, red_triangle)
        a = force_to_choose_and_observe_action!(subject)
        if a == a_1
            reward!(subject, -1)
            r_per_session -= 1
        else
            reward!(subject, +1)
            r_per_session += 1
        end
        # do something else before trial 2 starts
        show_to!(subject, red_triangle)
        show_to!(subject, black_circle)
        show_to!(subject, black_circle)
        show_to!(subject, red_triangle)
        a = force_to_choose_and_observe_action!(subject)
        if a == a_1
            reward!(subject, +1)
            r_per_session += 1
        else
            reward!(subject, -1)
            r_per_session -= 1
        end
        # do something else before trial 3 starts
        show_to!(subject, blue_square)
        show_to!(subject, black_circle)
        show_to!(subject, blue_square)
        a = force_to_choose_and_observe_action!(subject)
        if a == a_1
            reward!(subject, +1)
            r_per_session += 1
        else
            reward!(subject, -1)
            r_per_session -= 1
        end
        # do something else before trial 4 starts
        show_to!(subject, blue_square)
        show_to!(subject, black_circle)
        show_to!(subject, black_circle)
        show_to!(subject, blue_square)
        a = force_to_choose_and_observe_action!(subject)
        if a == a_1
            reward!(subject, -1)
            r_per_session -= 1
        else
            reward!(subject, +1)
            r_per_session += 1
        end
        push!(results, [session, r_per_session/4])
        # do something else before new session starts
    end
end
# compute average reward per session
```

References

Abraham, W. C. (2003). How long will long-term potentiation last? (T. V. P. Bliss, G. L. Collingridge, & R. G. M. Morris, Eds.). *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, *358*(1432), 735–744. https://doi.org/10.1098/rstb.2002.1222

Addyman, C., French, R. M., & Thomas, E. (2016). Computational models of interval timing. *Current Opinion in Behavioral Sciences*, *8*, 140–146. https://doi.org/10.1016/j.cobeha.2016.01.004

Amit, D. J. (1989). *Modeling brain function: The world of attractor neural networks*. Cambridge University Press.

Bellmund, J. L. S., Deuker, L., Montijn, N. D., & Doeller, C. F. (2022). Mnemonic construction and representation of temporal structure in the hippocampal formation. *Nature Communications*, *13*(1). https://doi.org/10.1038/s41467-022-30984-3

Bennett, S. H., Kirby, A. J., & Finnerty, G. T. (2018). Rewiring the connectome: Evidence and effects. *Neuroscience & Biobehavioral Reviews*, *88*, 51–62. https://doi.org/10.1016/j.neubiorev.2018.03.001

Brea, J., Clayton, N. S., & Gerstner, W. (2023). Computational models of episodic-like memory in food-caching birds. *Nature Communications*, *14*(1). https://doi.org/10.1038/s41467-023-38570-x

Bright, I. M., Meister, M. L. R., Cruzado, N. A., Tiganj, Z., Buffalo, E. A., & Howard, M. W. (2020). A temporal record of the past with a spectrum of time constants in the monkey entorhinal cortex. *Proceedings of the National Academy of Sciences*, *117*(33), 20274–20283. https://doi.org/10.1073/pnas.1917197117

Buhusi, C. V., & Meck, W. H. (2005). What makes us tick? functional and neural mechanisms of interval timing. *Nature Reviews Neuroscience*, *6*(10), 755–765. https://doi.org/10.1038/nrn1764

Carr, C., & Konishi, M. (1990). A circuit for detection of interaural time differences in the brain stem of the barn owl. *The Journal of Neuroscience*, *10*(10), 3227–3246. https://doi.org/10.1523/jneurosci.10-10-03227.1990

Clayton, N. S., Dally, J., Gilbert, J., & Dickinson, A. (2005). Food caching by western scrub-jays (aphelocoma californica) is sensitive to the conditions at recovery. *Journal of Experimental Psychology: Animal Behavior Processes*, *31*(2), 115–124. https://doi.org/10.1037/0097-7403.31.2.115

Clayton, N. S., & Dickinson, A. (1998). Episodic-like memory during cache recovery by scrub jays. *Nature*, *395*(6699), 272–274. https://doi.org/10.1038/26216

Clayton, N. S., & Dickinson, A. (1999). Scrub jays (aphelocoma coerulescens) remember the relative time of caching as well as the location and content of their caches. *Journal of Comparative Psychology*, *113*(4), 403–416. https://doi.org/10.1037/0735-7036.113.4.403

Clayton, N. S., Yu, K. S., & Dickinson, A. (2001). Scrub jays (aphelocoma coerulescens) form integrated memories of the multiple features of caching episodes. *Journal of Experimental Psychology: Animal Behavior Processes*, *27*(1), 17–29. https://doi.org/10.1037/0097-7403.27.1.17

Clayton, N. S., Yu, K. S., & Dickinson, A. (2003). Interacting cache memories: Evidence for flexible memory use by western scrub-jays (aphelocoma californica). *Journal of Experimental Psychology: Animal Behavior Processes*, *29*(1), 14–22. https://doi.org/10.1037/0097-7403.29.1.14

Eichenbaum, H. (2017). On the integration of space, time, and memory. *Neuron*, *95*(5), 1007–1018. https://doi.org/10.1016/j.neuron.2017.06.036

Friedman, W. J. (1993). Memory for the time of past events. *Psychological Bulletin*, *113*(1), 44–66. https://doi.org/10.1037/0033-2909.113.1.44

Friedman, W. J. (2013). The development of memory for the times of past events. *The Wiley Handbook on the Development of Children's Memory*, 394–407. https://doi.org/10.1002/9781118597705.ch17

Gerstner, W., Kempter, R., van Hemmen, J. L., & Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Nature*, *383*(6595), 76–78. https://doi.org/10.1038/383076a0

Gerstner, W., Lehmann, M., Liakoni, V., Corneil, D., & Brea, J. (2018). Eligibility traces and plasticity on behavioral time scales: Experimental support of neohebbian three-factor learning rules. *Frontiers in Neural Circuits*, *12*, 53. https://doi.org/10.3389/fncir.2018.00053

Grothe, B., Pecka, M., & McAlpine, D. (2010). Mechanisms of sound localization in mammals. *Physiological Reviews*, *90*(3), 983–1012. https://doi.org/10.1152/physrev.00026.2009

Hasselmo, M. E., Bodelón, C., & Wyble, B. P. (2002). A proposed function for hippocampal theta rhythm: Separate phases of encoding and retrieval enhance reversal of prior learning. *Neu-*

ral Computation, *14*(4), 793–817. https://doi.org/10.1162/089976602317318965

Henson, R. N. (1998). Short-term memory for serial order: The start-end model. *Cognitive Psychology*, *36*(2), 73–137. https://doi.org/10.1006/cogp.1998.0685

Hertz, J. A., Krogh, A. S., & Palmer, R. G. (1991). *Introduction to the theory of neural computation*. Westview Press.

Howard, M. W. (2022). Formal models of memory based on temporally-varying representations. *arXiv e-prints*, Article arXiv:2201.01796, arXiv:2201.01796.

Howard, M. W., & Kahana, M. J. (2002). A distributed representation of temporal context. *Journal of Mathematical Psychology*, *46*(3), 269–299. https://doi.org/10.1006/jmps.2001.1388

Iatropoulos, G., Brea, J., & Gerstner, W. (2022). Kernel memory networks: A unifying framework for memory modeling. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh (Eds.), *Advances in neural information processing systems* (pp. 35326–35338, Vol. 35). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2022/file/e55d081280e79e714debf2902e18eb69-Paper-Conference.pdf

Illing, B., Gerstner, W., & Brea, J. (2019). Biologically plausible deep learning — but how far can we go with shallow networks? *Neural Networks*, *118*, 90–101. https://doi.org/10.1016/j.neunet.2019.06.001

Issa, J. B., Tocker, G., Hasselmo, M. E., Heys, J. G., & Dombeck, D. A. (2020). Navigating through time: A spatial navigation perspective on how the brain may encode time. *Annual Review of Neuroscience*, *43*(1), 73–93. https://doi.org/10.1146/annurev-neuro-101419-011117

Jelbert, S. A., & Clayton, N. S. (2017). Comparing the non-linguistic hallmarks of episodic memory systems in corvids and children. *Current Opinion in Behavioral Sciences*, *17*, 99–106. https://doi.org/10.1016/j.cobeha.2017.07.011

Jensen, O., & Lisman, J. E. (2005). Hippocampal sequence-encoding driven by a cortical multi-item working memory buffer. *Trends in Neurosciences*, *28*(2), 67–72. https://doi.org/10.1016/j.tins.2004.12.001

Kahana, M. J. (2020). Computational models of memory search. *Annual Review of Psychology*, *71*(1), 107–138. https://doi.org/10.1146/annurev-psych-010418-103358

Kahana, M. J. (2012). *Foundations of human memory*. Oxford University Press.

Katkov, M., & Tsodyks, M. (2022). Statistics of free memory recall. *Physical Review Research*, *4*(3). https://doi.org/10.1103/physrevresearch.4.033090

Kelly, M. A., Blostein, D., & Mewhort, D. J. K. (2013). Encoding structure in holographic reduced representations. *Canadian Journal of Experimental Psychology / Revue canadienne de psychologie expérimentale*, *67*(2), 79–93. https://doi.org/10.1037/a0030301

Kesner, R. P., & Rolls, E. T. (2015). A computational theory of hippocampal function, and tests of the theory: New developments. *Neuroscience & Biobehavioral Reviews*, *48*, 92–147. https://doi.org/10.1016/j.neubiorev.2014.11.009

Kuhn, T. S. (1996). *The structure of scientific revolutions* (Third Edition). University of Chicago Press. https://doi.org/10.7208/chicago/9780226458106.001.0001

Kuśmierz, Isomura, T., & Toyoizumi, T. (2017). Learning with three factors: Modulating hebbian plasticity with errors. *Current Opinion in Neurobiology*, *46*, 170–177. https://doi.org/10.1016/j.conb.2017.08.020

Lillicrap, T. P., Cownden, D., Tweed, D. B., & Akerman, C. J. (2016). Random synaptic feedback weights support error backpropagation for deep learning. *Nature Communications*, *7*, 13276. https://doi.org/10.1038/ncomms13276

Lisman, J., Grace, A. A., & Duzel, E. (2011). A neohebbian framework for episodic memory; role of dopamine-dependent late ltp. *Trends in Neurosciences*, *34*(10), 536–547. https://doi.org/10.1016/j.tins.2011.07.006

Magee, J. C., & Grienberger, C. (2020). Synaptic plasticity forms and functions. *Annual Review of Neuroscience*, *43*(1), 95–117. https://doi.org/10.1146/annurev-neuro-090919-022842

McClelland, J. L., McNaughton, B. L., & O’Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3), 419–457. https://doi.org/10.1037/0033-295x.102.3.419

Moscovitch, M., & Gilboa, A. (2021). Systems consolidation, transformation and reorganization: Multiple trace theory, trace transformation theory and their competitors. *PsyArXiv Preprints*. https://doi.org/10.31234/osf.io/yxbrs

Norman, K. A., Detre, G., & Polyn, S. M. (2008). Computational models of episodic memory (R. Sun, Ed.). *The Cambridge Handbook of Computational Psychology*, 189–225. https://doi.org/10.1017/cbo9780511816772.011

Norman, K. A., & O'Reilly, R. C. (2003). Modeling hippocampal and neocortical contributions to recognition memory: A complementary-learning-systems approach. *Psychological Review*, *110*(4), 611–646. https://doi.org/10.1037/0033-295x.110.4.611

Pathman, T., Larkina, M., Burch, M. M., & Bauer, P. J. (2013). Young children's memory for the times of personal past events. *Journal of Cognition and Development*, *14*(1), 120–140. https://doi.org/10.1080/15248372.2011.641185

Paton, J. J., & Buonomano, D. V. (2018). The neural basis of timing: Distributed mechanisms for diverse functions. *Neuron*, *98*(4), 687–705. https://doi.org/10.1016/j.neuron.2018.03.045

Polyn, S. M., Norman, K. A., & Kahana, M. J. (2009). A context maintenance and retrieval model of organizational processes in free recall. *Psychological Review*, *116*(1), 129–156. https://doi.org/10.1037/a0014420

Remme, M. W. H., Bergmann, U., Alevi, D., Schreiber, S., Sprekeler, H., & Kempter, R. (2021). Hebbian plasticity in parallel synaptic pathways: A circuit mechanism for systems memory consolidation (D. Bush, Ed.). *PLOS Computational Biology*, *17*(12), e1009681. https://doi.org/10.1371/journal.pcbi.1009681

Roelfsema, P. R., & Holtmaat, A. (2018). Control of synaptic plasticity in deep cortical networks. *Nature Reviews Neuroscience*, *19*(3), 166–180. https://doi.org/10.1038/nrn.2018.6

Roelfsema, P. R., & Ooyen, A. v. (2005). Attention-gated reinforcement learning of internal representations for classification. *Neural Computation*, *17*(10), 2176–2214. https://doi.org/10.1162/0899766054615699

Rolls, E. T., & Mills, P. (2019). The generation of time in the hippocampal memory system. *Cell Reports*, *28*(7), 1649–1658.e6. https://doi.org/10.1016/j.celrep.2019.07.042

Romani, S., Pinkoviezky, I., Rubin, A., & Tsodyks, M. (2013). Scaling laws of associative memory retrieval. *Neural Computation*, *25*(10), 2523–2544. https://doi.org/10.1162/neco_a_00499

Roxin, A., & Fusi, S. (2013). Efficient partitioning of memory systems and its importance for memory consolidation (J. Beck, Ed.). *PLoS Computational Biology*, *9*(7), e1003146. https://doi.org/10.1371/journal.pcbi.1003146

Rubin, A., Geva, N., Sheintuch, L., & Ziv, Y. (2015). Hippocampal ensemble dynamics timestamp events in long-term memory. *eLife*, *4*. https://doi.org/10.7554/elife.12247

Rule, M. E., O'Leary, T., & Harvey, C. D. (2019). Causes and consequences of representational drift. *Current Opinion in Neurobiology*, *58*, 141–147. https://doi.org/10.1016/j.conb.2019.08.005

Sadeh, S., & Clopath, C. (2022). Contribution of behavioural variability to representational drift. *eLife*, *11*. https://doi.org/10.7554/elife.77907

Smolensky, P. (1990). Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, *46*(1-2), 159–216. https://doi.org/10.1016/0004-3702(90)90007-m

Squire, L. R., Genzel, L., Wixted, J. T., & Morris, R. G. (2015). Memory consolidation. *Cold Spring Harbor Perspectives in Biology*, *7*(8), a021766. https://doi.org/10.1101/cshperspect.a021766

Statman, A., Kaufman, M., Minerbi, A., Ziv, N. E., & Brenner, N. (2014). Synaptic size dynamics as an effectively stochastic process (Y. Loewenstein, Ed.). *PLoS Computational Biology*, *10*(10), e1003846. https://doi.org/10.1371/journal.pcbi.1003846

Surace, S. C., Pfister, J.-P., Gerstner, W., & Brea, J. (2020). On the choice of metric in gradient-based theories of brain function (F. Ouellette, Ed.). *PLOS Computational Biology*, *16*(4), e1007640. https://doi.org/10.1371/journal.pcbi.1007640

Taub, K., Abeles, D., & Yuval-Greenberg, S. (2022). Evidence for content-dependent timing of real-life events during covid-19 crisis. *Scientific Reports*, *12*(1). https://doi.org/10.1038/s41598-022-13076-6

Treves, A., & Rolls, E. T. (1994). Computational analysis of the role of the hippocampus in memory. *Hippocampus*, *4*(3), 374–391. https://doi.org/10.1002/hipo.450040319

Tsao, A., Sugar, J., Lu, L., Wang, C., Knierim, J. J., Moser, M.-B., & Moser, E. I. (2018). Integrating time from experience in the lateral entorhinal cortex. *Nature*, *561*(7721), 57–62. https://doi.org/10.1038/s41586-018-0459-6

Tsao, A., Yousefzadeh, S. A., Meck, W. H., Moser, M.-B., & Moser, E. I. (2022). The neural bases for timing of durations. *Nature Reviews Neuro-*

science, *23*(11), 646–665. https://doi.org/10.1038/s41583-022-00623-3

Wiener, M., Matell, M. S., & Coslett, H. B. (2011). Multiple mechanisms for temporal processing. *Frontiers in Integrative Neuroscience*, *5*. https://doi.org/10.3389/fnint.2011.00031

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, *8*(3-4), 229–256. https://doi.org/10.1007/bf00992696

Zacks, J. M. (2020). Event perception and memory. *Annual Review of Psychology*, *71*(1), 165–191. https://doi.org/10.1146/annurev-psych-010419-051101