

# Sequence learning, prediction, and replay in networks of spiking neurons

Younes Bouhadjar<sup>1,2,5,\*</sup>, Dirk J. Wouters<sup>4</sup>, Markus Diesmann<sup>1,3</sup>, and Tom Tetzlaff<sup>1</sup>

<sup>1</sup>Institute of Neuroscience and Medicine (INM-6), & Institute for Advanced Simulation (IAS-6), & JARA BRAIN Institute Structure-Function Relationships (INM-10), Jülich Research Centre, Jülich, Germany

<sup>2</sup>Peter Grünberg Institute (PGI-7,10), Jülich Research Centre and JARA, Jülich, Germany

<sup>3</sup>Department of Physics, Faculty 1, & Department of Psychiatry, Psychotherapy, and Psychosomatics, Medical School, RWTH Aachen University, Aachen, Germany

<sup>4</sup>Institute of Electronic Materials (IWE 2) & JARA-FIT, RWTH Aachen University, Aachen, Germany

<sup>5</sup>RWTH Aachen University, Aachen, Germany

\*[y.bouhadjar@fz-juelich.de](mailto:y.bouhadjar@fz-juelich.de)

November 8, 2021

## Abstract

Sequence learning, prediction and replay have been proposed to constitute the universal computations performed by the neocortex. The Hierarchical Temporal Memory (HTM) algorithm realizes these forms of computation. It learns sequences in an unsupervised and continuous manner using local learning rules, permits a context specific prediction of future sequence elements, and generates mismatch signals in case the predictions are not met. While the HTM algorithm accounts for a number of biological features such as topographic receptive fields, nonlinear dendritic processing, and sparse connectivity, it is based on abstract discrete-time neuron and synapse dynamics, as well as on plasticity mechanisms that can only partly be related to known biological mechanisms. Here, we devise a continuous-time implementation of the temporal-memory (TM) component of the HTM algorithm, which is based on a recurrent network of spiking neurons with biophysically interpretable variables and parameters. The model learns high-order sequences by means of a structural Hebbian synaptic plasticity mechanism supplemented with a rate-based homeostatic control. In combination with nonlinear dendritic input integration and local inhibitory feedback, this type of plasticity leads to the dynamic self-organization of narrow sequence-specific feedforward subnetworks. These subnetworks provide the substrate for a faithful propagation of sparse, synchronous activity, and, thereby, for a robust, context specific prediction of future sequence elements as well as for the autonomous replay of previously learned sequences. By strengthening the link to biology, our implementation facilitates the evaluation of the TM hypothesis based on experimentally accessible quantities. The continuous-time implementation of the TM algorithm permits, in particular, an investigation of the role of sequence timing for sequence learning, prediction and replay. We demonstrate this aspect by studying the effect of the sequence speed on the sequence learning performance and on the speed of autonomous sequence replay.

## Author summary

Essentially all data processed by mammals and many other living organisms is sequential. This holds true for all types of sensory input data as well as motor output activity. Being able to form memories of such sequential data, to predict future sequence elements, and to replay learned sequences is a necessary prerequisite for survival. It has been hypothesized that sequence learning, prediction and replay constitute the fundamental computations performed by the neocortex. The Hierarchical Temporal Memory (HTM) constitutes an abstract powerful algorithm implementing this form of computation and has been proposed to serve as a model of neocortical processing. In this study, we are reformulating this algorithm in terms of known biological ingredients and mechanisms to foster the verifiability of the HTM hypothesis based on electrophysiological and behavioral data. The proposed model learns continuously in an unsupervised manner by biologically plausible, local plasticity mechanisms, and successfully predicts and replays complex sequences. Apart from establishing contact to biology, the study sheds light on the mechanisms determining at what speed we can process sequences and provides an explanation of fast sequence replay observed in the hippocampus and in the neocortex.

## Introduction

Learning and processing sequences of events, objects, or percepts are fundamental computational building blocks of cognition [1–4]. Prediction of upcoming sequence elements, mismatch detection and sequence replay in response to a cue signal constitute central components of this form of processing. We are constantly making predictions about

what we are going to hear, see, and feel next. We effortlessly detect surprising, non-anticipated events and adjust our behavior accordingly. Further, we manage to replay learned sequences, for example, when generating motor behavior, or recalling sequential memories. These forms of processing have been studied extensively in a number of experimental works on sensory processing [5, 6], motor production [7], and decision making [8].

The majority of existing biologically motivated models of sequence learning addresses sequence replay [9–12]. Sequence prediction and mismatch detection are rarely discussed. The Hierarchical Temporal Memory (HTM) [13] combines all three aspects: sequence prediction, mismatch detection and replay. Its Temporal Memory (TM) model [14] learns complex context dependent sequences in a continuous and unsupervised manner using local learning rules [15], and is robust against noise and failure in system components. Furthermore, it explains the functional role of dendritic action potentials (dAPs) and proposes a mechanism of how mismatch signals can be generated in cortical circuits [14]. Its capacity benefits from sparsity in the activity, and therefore provides a highly energy efficient sequence learning and prediction mechanism [16].

The original formulation of the TM model is based on abstract models of neurons and synapses with discrete-time dynamics. Moreover, the way the network forms synapses during learning is difficult to reconcile with biology. Here, we propose a continuous-time implementation of the TM model derived from known biological principles such as spiking neurons, dAPs, lateral inhibition, spike-timing-dependent structural plasticity, and homeostatic control of synapse growth. This model successfully learns, predicts and replays high-order sequences, where the prediction of the upcoming element is not only dependent on the current element, but also on the history. Bringing the model closer to biology allows for testing its hypotheses based on experimentally accessible quantities such as synaptic connectivity, synaptic currents, transmembrane potentials, or spike trains. Reformulating the model in terms of continuous-time dynamics moreover enables us to address timing-related questions, such as the role of the sequence speed for the prediction performance and the replay speed.

The study is organized as follows: the [Methods](#) describe the task, the network model, and the performance measures. The [Results](#) illustrate how the interaction of the model’s components gives rise to context dependent predictions and sequence replay, and evaluate the sequence processing speed and prediction performance. The [Discussion](#) finally compares the spiking TM model to other biologically motivated sequence learning models, summarizes limitations, and provides suggestions for future model extensions.

## Methods

In the following, we provide an overview of the task and the training protocol, the network model, and the task performance analysis. A detailed description of the model and parameter values can be found in Tables 1 and 2.

### Task and training protocol

In this study, we develop a neuronal architecture that can learn and process an ensemble of  $S$  sequences  $s_i = \{\zeta_{i,1}, \zeta_{i,2}, \dots, \zeta_{i,C_i}\}$  of ordered discrete items  $\zeta_{i,j}$  with  $C_i \in \mathbb{N}^+$ ,  $i \in [1, \dots, S]$ . The length of sequence  $s_i$  is denoted by  $C_i$ . Throughout this study, the sequence elements  $\zeta_{i,j} \in \{A, B, C, \dots\}$  are represented by Latin characters, serving as placeholders for arbitrary discrete objects or percepts, such as images, numbers, words, musical notes, or movement primitives (Fig. 1A). The order of the sequence elements within a given sequence represents the temporal order of item occurrence.

The tasks to be solved by the network consist of

- i) predicting subsequent sequence elements in response to the presentation of other elements,
- ii) detecting unanticipated stimuli and generating a mismatch signal if the prediction is not met, and
- iii) autonomously replaying sequences in response to a cue signal after learning.

The architecture learns sequences in a continuous manner: the network is exposed to repeated presentations of a given ensemble of sequences (e.g.,  $\{A, D, B, E\}$  and  $\{F, D, B, C\}$  in Fig. 1B). In the *prediction mode* (task i) and ii)), there is no distinction between a “training” and a “testing” phase. At the beginning of the learning process, all presented sequence elements are unanticipated and do not lead to a prediction (diffuse shades in Fig. 1B, left). As a consequence, the network generates mismatch signals (flash symbols in Fig. 1B, left). After successful learning, the presentation of some sequence element leads to a prediction of the subsequent stimulus (colored arrows in Fig. 1B). In case this subsequent stimulus does not match the prediction, the network generates a mismatch signal (red arrow and flash symbol in Fig. 1B, right). The learning process is entirely unsupervised, i.e., the prediction performance does not affect the learning. As described in [Sequence replay](#), the network can be configured into a *replay mode* where the network autonomously replays learned sequences in response to a cue signal (task iii)).

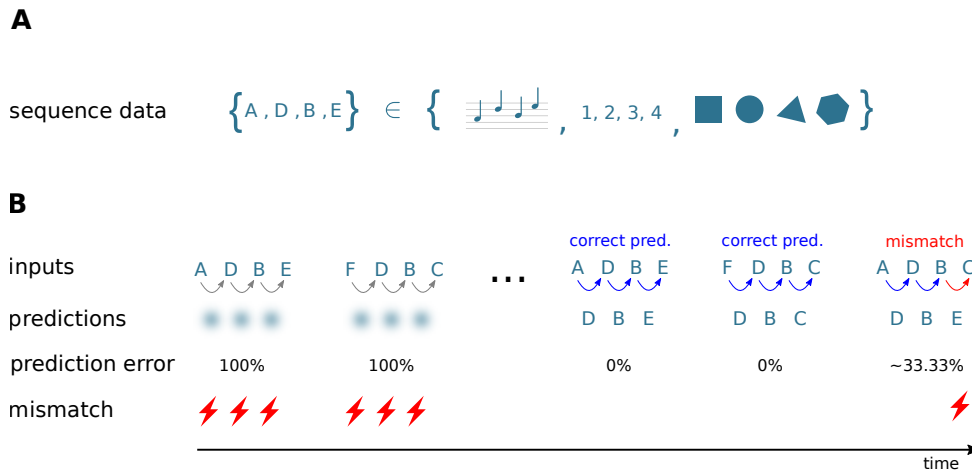
In general, the sequences in this study are “high-order” sequences, similar to those generated by a high-order Markov chain; the prediction of an upcoming sequence element requires accounting for not just the previous element, but for (parts of) the entire sequence history, i.e., the context. Sequences within a given set of training data can

be partially overlapping; they may share certain elements or subsequences (such as in  $\{A,D,B,E\}$  and  $\{F,D,B,C\}$ ). Similarly, the same sequence element (but not the first one, see [Limitations and outlook](#)) may occur multiple times within the same sequence (such as in  $\{A,D,B,D\}$ ). Throughout this work, we use two sequence sets:

**Sequence set I:** For an illustration of the learning process and the network dynamics in the prediction (section [Sequence learning and prediction](#)) and in the replay mode (section [Sequence replay](#)), as well as for the investigation of the sequence processing speed (section [Dependence of prediction performance on the sequence speed](#)), we start with a simple set of two partially overlapping sequences  $s_1 = \{A,D,B,E\}$  and  $s_2 = \{F,D,B,C\}$  (see Fig. 1B).

**Sequence set II:** For a more rigorous evaluation of the sequence prediction performance (section [Prediction performance](#)), we consider a set of  $S = 6$  high-order sequences:  $s_1 = \{E,N,D,I,J\}$ ,  $s_2 = \{L,N,D,I,K\}$ ,  $s_3 = \{G,J,M,C,N\}$ ,  $s_4 = \{F,J,M,C,I\}$ ,  $s_5 = \{B,C,K,H,I\}$ ,  $s_6 = \{A,C,K,H,F\}$ , each consisting of  $C = 5$  elements. The complexity of this sequence ensemble is comparable to the one used in [14], but is more demanding in terms of the high-order context dependence.

Results for a third sequence set composed of sequences with recurring first elements are summarized in Fig. S2.



**Fig 1: Sketch of the task and the learning protocol.** **A)** The neuronal network model developed in this study learns and processes sequences of ordered discrete elements, here represented by characters “A”, “B”, “C”, ... Sequence elements may constitute arbitrary discrete items, such as musical notes, numbers, or images. The order of sequence elements represents the temporal order of item occurrence. **B)** After repeated, consistent presentation of sets of high-order sequences, i.e., sequences with overlapping characters (here,  $\{A,D,B,E\}$  and  $\{F,D,B,C\}$ ), the model learns to predict subsequent elements in response to the presentation of other elements (blue arrows) and to detect unanticipated elements by generating a mismatch signal if the prediction is not met (red arrows and flash symbols). The learning process is continuous and unsupervised. At the beginning of the learning process, all presented elements are unanticipated and hence trigger the generation of a mismatch signal. The learning progress is monitored and quantified by the prediction error (see [Task performance measures](#)).

## Network model

**Algorithmic requirements.** To solve the tasks outlined in [Task and training protocol](#), the network model needs to implement a number of algorithmic components. Here, we provide an overview of these components and their corresponding implementations:

- **Learning and storage of sequences:** in both the original and our model, sequences are represented by specific subnetworks embedded into the recurrent network. During the learning process, these subnetworks are carved out in an unsupervised manner by a form of structural Hebbian plasticity.
- **Context specificity:** in our model, learning of high-order sequences is enabled by a sparse, random potential connectivity, and by a homeostatic regulation of synaptic growth.
- **Generation of predictions:** neurons are equipped with a predictive state, implemented by a nonlinear synaptic integration mimicking the generation of dendritic action potentials (dAPs).
- **Mismatch detection:** only few neurons become active if a prediction matches the stimulus. In our model, this sparsity is realized by a winner-take-all (WTA) dynamics implemented in the form of inhibitory feedback. In

case of non-anticipated stimuli, the WTA dynamics cannot step in, thereby leading to a non-sparse activation of larger neuron populations.

- Sequence replay: autonomous replay of learned sequences in response to a cue signal is enabled by increasing neuronal excitability.

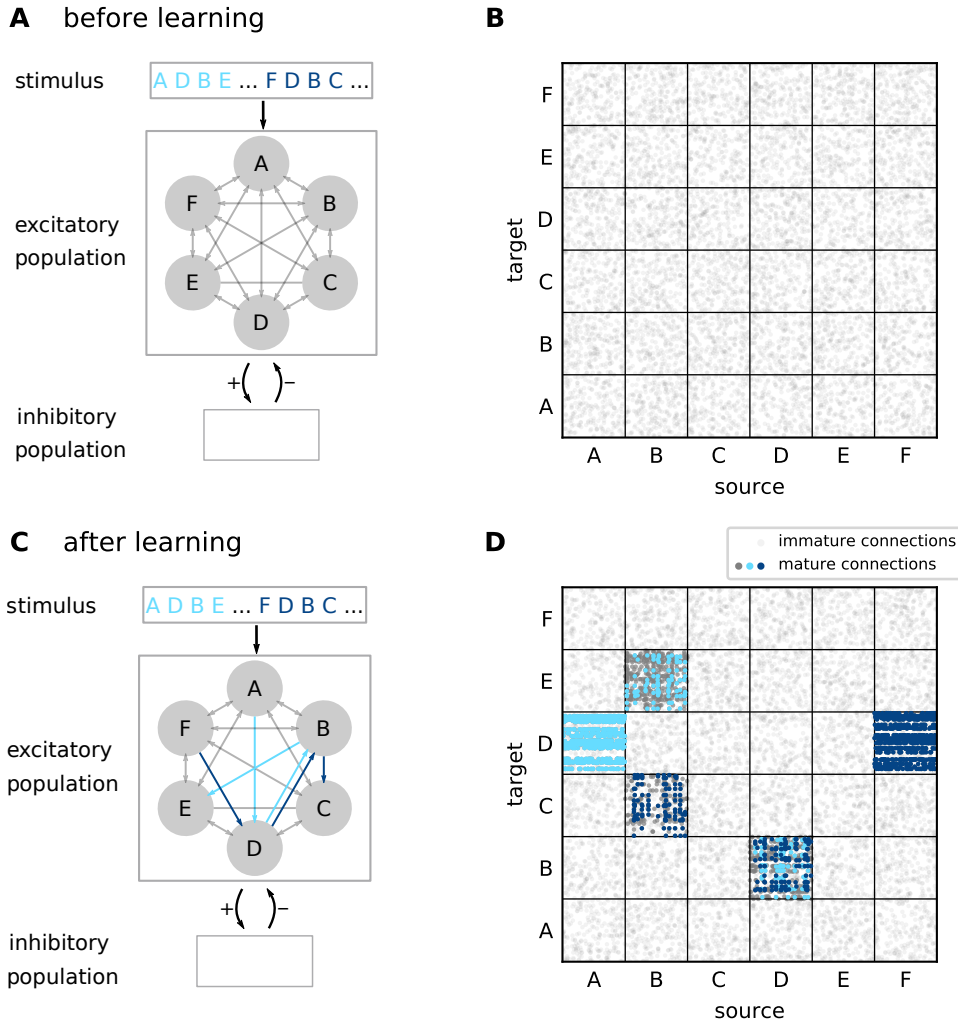
In the following paragraphs, the implementations of these components and the differences between the original and our model are explained in more detail.

**Network structure.** The network consists of a population  $\mathcal{E}$  of  $N_E$  excitatory (“E”) and a population  $\mathcal{I}$  of  $N_I$  inhibitory (“I”) neurons. The neurons in  $\mathcal{E}$  are randomly and recurrently connected, such that each neuron in  $\mathcal{E}$  receives  $K_{EE}$  excitatory inputs from other neurons in  $\mathcal{E}$ . Note that these “EE” connections can be either “mature” or “immature” (“effective” or “potential”; see below). In the neocortex, the degree of potential connectivity depends on the distance between the neurons [17]. It can reach probabilities as high as 90% for neighboring neurons, and decays to 0% for neurons that are farther apart. In this work, the connection probability is chosen such that the connectivity is sufficiently dense allowing for the formation of feedforward subnetworks, and sufficiently sparse for increasing the network capacity (see paragraph “Constraints on potential connectivity” below). The excitatory population  $\mathcal{E}$  is subdivided into  $M$  non-overlapping subpopulations  $\mathcal{M}_1, \dots, \mathcal{M}_M$ , each of them containing neurons with identical stimulus preference (“receptive field”; see below). Each subpopulation  $\mathcal{M}_k$  thereby represents a specific element within a sequence (Fig. 2A,B). In the original TM model [14], a single sequence element is represented by multiple ( $L$ ) subpopulations (“minicolumns”). For simplicity, we identify the number  $M$  of subpopulations with the number of elements required for a specific set of sequences, such that each sequence element is encoded by just one subpopulation ( $L = 1$ ). All neurons within a subpopulation  $\mathcal{M}_k$  are recurrently connected to a subpopulation-specific inhibitory neuron  $k \in \mathcal{I}$ . The inhibitory neurons in  $\mathcal{I}$  are mutually unconnected.

**External inputs.** During the prediction mode, the network is driven by an ensemble  $\mathcal{X} = \{x_1, \dots, x_M\}$  of  $M$  external inputs, representing inputs from other brain areas, such as thalamic sources or other cortical areas. Each of these external inputs  $x_k$  represents a specific sequence element (“A”, “B”, ...), and feeds all neurons in the subpopulation  $\mathcal{M}_k$  with the corresponding stimulus preference. The occurrence of a specific sequence element  $\zeta_{i,j}$  at time  $t_{i,j}$  is modeled by a single spike  $x_k(t) = \delta(t - t_{i,j})$  generated by the corresponding external source  $x_k$ . Subsequent sequence elements  $\zeta_{i,j}$  and  $\zeta_{i,j+1}$  within a sequence  $s_i$  are presented with an inter-stimulus interval  $\Delta T = t_{i,j+1} - t_{i,j}$ . Subsequent sequences  $s_i$  and  $s_{i+1}$  are separated in time by an inter-sequence time interval  $\Delta T_{\text{seq}} = t_{i+1,1} - t_{i,C_i}$ . During the replay mode, we present only a cue signal encoding for first sequence elements  $\zeta_{i,1}$  at times  $t_{i,1}$ . Subsequent cues are separated in time with an inter-cue time interval  $\Delta T_{\text{cue}} = t_{i+1,1} - t_{i,1}$ . In the absence of any other (inhibitory) inputs, each external input spike is strong enough to evoke an immediate response spike in all target neurons  $i \in \mathcal{M}_k$ . Sparse activation of the subpopulations in response to the external inputs is achieved by a winner-take-all mechanism implemented in the form of inhibitory feedback (see [Sequence learning and prediction](#)).

**Neuron and synapse model.** In the original TM model [14], excitatory (pyramidal) neurons are described as abstract three-state systems that can assume an active, a predictive, or a non-active state. State updates are performed in discrete time. The current state is fully determined by the external input in the current time step and the network state in the previous step. Each TM neuron is equipped with a number of dendrites (segments), modeled as coincidence detectors. The dendrites are grouped into distal and proximal dendrites. Distal dendrites receive inputs from other neurons in the local network, whereas proximal dendrites are activated by external sources. Inputs to proximal dendrites have a large effect on the soma and trigger the generation of action potentials. Individual synaptic inputs to a distal dendrite, in contrast, have no direct effect on the soma. If the total synaptic input to a distal dendritic branch at a given time step is sufficiently large, the neuron becomes predictive. This dynamic mimics the generation of dendritic action potentials (dAPs), NMDA spikes [18–20]), which result in a long-lasting depolarization (~50-500ms) of the somata of neocortical pyramidal neurons.

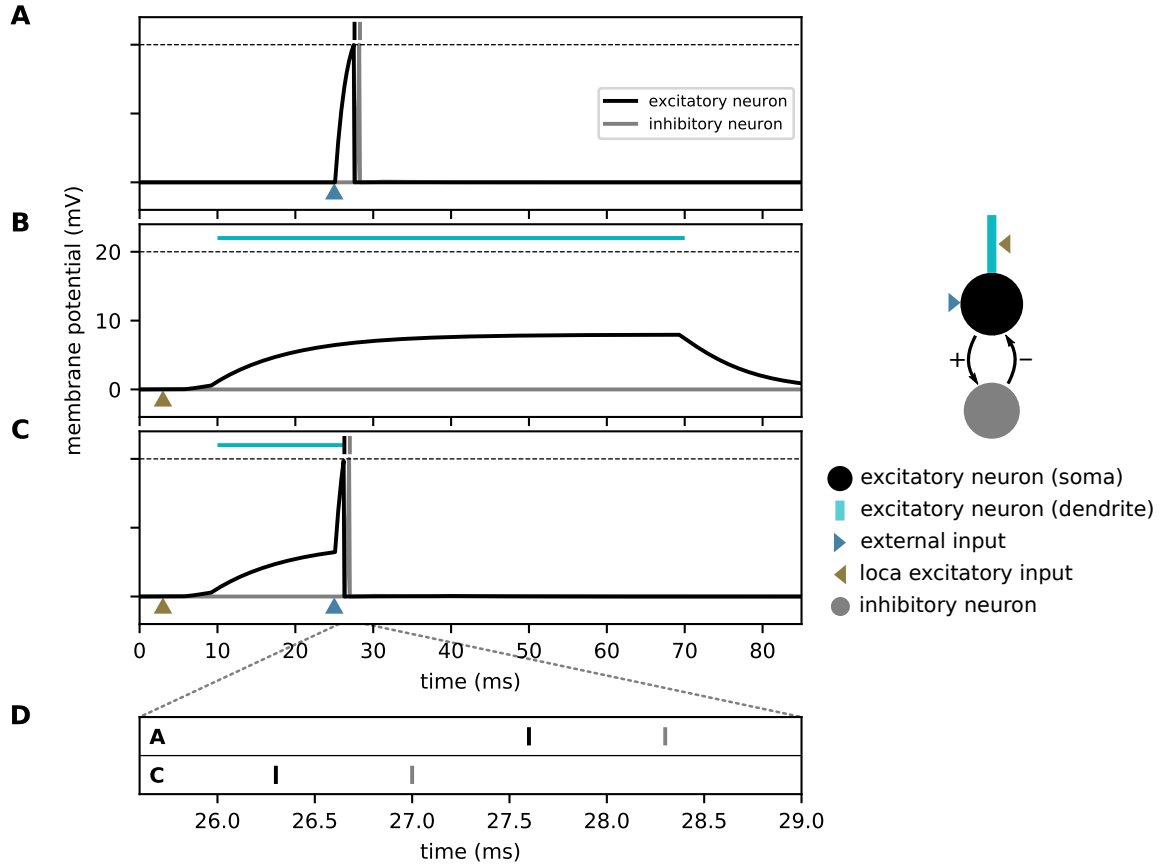
In contrast to the original study, the model proposed here employs neurons with continuous-time dynamics. For all types of neurons, the temporal evolution of the membrane potential is given by the leaky integrate-and-fire model Eq (10). The total synaptic input current of excitatory neurons is composed of currents in distal dendritic branches, inhibitory currents, and currents from external sources. Inhibitory neurons receive only inputs from excitatory neurons in the same subpopulation. Individual spikes arriving at dendritic branches evoke alpha-shaped postsynaptic currents, see Eq (12). The dendritic current includes an additional nonlinearity describing the generation of dAPs: if the dendritic current  $I_{ED}$  exceeds a threshold  $\theta_{dAP}$ , it is instantly set to the dAP plateau current  $I_{dAP}$ , and clamped to this value for a period of duration  $\tau_{dAP}$ , see Eq (16). This plateau current leads to a long lasting depolarization of the soma (see Fig. 3B). In this work, we use a single dendritic branch per neuron. However, the model could easily be extended to include multiple dendritic branches. External and inhibitory inputs to excitatory neurons as well as excitatory inputs to inhibitory neurons trigger exponential postsynaptic currents, see Eq (13–15). Similar



**Fig 2: Sketch of the network structure.** **A)** The architecture constitutes a recurrent network of excitatory and inhibitory neurons. Excitatory neurons are stimulated by external sources providing sequence-element specific inputs “A”, “D”, etc. The excitatory neuron population is composed of subpopulations containing neurons with identical stimulus preference (gray circles). Connections between and within the excitatory subpopulations are random and sparse. Inhibitory neurons are recurrently connected to excitatory neurons, but mutually unconnected. **B)** Initial connectivity matrix for excitatory connections to excitatory neurons (EE connections). Target and source neurons are grouped into stimulus-specific subpopulations (“A”, ..., “F”). Before learning, the excitatory neurons are sparsely and randomly connected via immature (potential) synapses (light gray dots). **C)** During learning, sequence specific, sparsely connected feedforward subnetworks with mature (effective) synapses are formed (light blue arrows: {A,D,B,E}, dark blue arrows: {F,D,B,C}). **D)** EE connectivity matrix after learning. During the learning process, subsets of connections between subpopulations corresponding to subsequent sequence elements become mature and effective (light and dark blue dots). Mature connections are context specific (see distinct connectivity between subpopulations “D” and “B” corresponding to different sequences), thereby providing the backbone for a reliable propagation of sequence-specific activity. In panels B and D, only 10% of all EE connections are shown for clarity. Dark gray dots in panel D correspond to mature connections between neurons that remain silent after learning. For details on the network structure, see Table 1 and Table 2.

to the original implementation, an external input strongly depolarizes the neurons and causes them to fire. To this end, the external weights  $J_{EX}$  are chosen to be supra-threshold (see Fig. 3A). Inhibitory interactions implement the WTA described in [Sequence learning and prediction](#). The weights  $J_{IE}$  of excitatory synapses on inhibitory neurons are chosen such that the collective firing of a subset of  $\rho$  excitatory neurons in the corresponding subpopulation causes the inhibitory neuron to fire. The weights  $J_{EI}$  of inhibitory synapses on excitatory neurons are strong such that each inhibitory spike prevents all excitatory neurons in the same subpopulation that have not generated a spike yet from firing. All synaptic time constants, delays and weights are connection-type specific (see Table 1).

**Plasticity dynamics.** Both in the original [14] and in our model, the lateral excitatory connectivity between excitatory neurons (EE connectivity) is dynamic and shaped by a Hebbian structural plasticity mechanism mimicking principles known from the neuroscience literature [21–25]. All other connections are static. The dynamics of the EE



**Fig 3: Effect of dendritic action potentials (dAP) on the firing response to an external stimulus.** Membrane-potential responses to an external input (blue arrow, **A**), a strong dendritic input (brown arrow, **B**) triggering a dAP, and a combination of both (**C**). Black and gray vertical bars mark times of excitatory and inhibitory spikes, respectively. The horizontal light blue lines depict the dAP plateau. **D**) Magnified view of spike times from panels A and C. A dAP preceding the external input (as in panel C) can speed up somatic, and hence, inhibitory firing, provided the time interval between the dAP and the external input is in the right range. The excitatory neuron is connected bidirectionally to an inhibitory neuron (see sketch on the right).

connectivity is determined by the time evolution of the permanences  $P_{ij}$  ( $i, j \in \mathcal{E}$ ), representing the synapse maturity, and the synaptic weights  $J_{ij}$ . Unless the permanence  $P_{ij}$  exceeds a threshold  $\theta_P$ , the synapse  $\{j \rightarrow i\}$  is “immature” or “potential”, with zero synaptic weight  $J_{ij} = 0$ . Upon threshold crossing,  $P_{ij} \geq \theta_P$ , the synapse becomes “mature” or “effective”, and its weight is assigned a fixed value  $J_{ij} = W$  ( $\forall i, j$ ). Overall, the permanences evolve according to a Hebbian plasticity rule: the synapse  $\{j \rightarrow i\}$  is potentiated, i.e.,  $P_{ij}$  is increased, if the activation of the postsynaptic cell  $i$  is immediately preceded by an activation of the presynaptic cell  $j$ . Otherwise, the synapse is depressed, i.e.,  $P_{ij}$  is decreased. At the beginning of the learning process or during relearning, the activity in the individual subpopulations is non-sparse. Hebbian learning alone would therefore lead to the strengthening of all existing synapses between two subsequently activated subpopulations, irrespective of the context these two subpopulations participate in. After learning, the subsets of neurons that are activated by a sequence element recurring in different sequences would therefore largely overlap. As a consequence, it becomes harder to distinguish between different contexts (histories) based on the activation patterns of these subsets. The original TM model [14] avoids this loss of context sensitivity by restricting synaptic potentiation to a small subset of synapses between a given pair of source and target subpopulations: if there are no predictive target neurons, the original algorithm selects a “matched” neuron from the set of active postsynaptic cells as the one being closest to becoming predictive, i.e., the neuron receiving the largest number of synaptic inputs on a given dendritic branch from the set of active presynaptic cells (provided this number is sufficiently large). Synapse potentiation is then restricted to this set of matched neurons. In case there are no potential synapses, the “least used” neuron or a randomly chosen neuron is selected as the “matched” cell, and connected to the winner cell of the previously active subpopulation. Restricting synaptic potentiation to synapses targeting such a subset of “matched” neurons is difficult to reconcile with biology. It is known that inhibitory inputs targeting the dendrites of pyramidal cells can locally suppress backpropagating action potentials and, hence, synaptic potentiation [26]. A selection mechanism based on such local inhibitory circuits would however involve extremely



fast synapses and require fine-tuning of parameters. The model presented in this work circumvents the selection of “matched” neurons and replaces this with a homeostatic mechanism controlled by the postsynaptic dAP rate. In the following, the specifics of the plasticity dynamics used in this study are described in detail.

Within the interval  $[P_{\min,ij}, P_{\max}]$ , the dimensionless permanences  $P_{ij}(t)$  evolve according to a combination of an additive spike-timing-dependent plasticity (STDP) rule [27] and a homeostatic component [28, 29]:

$$P_{\max}^{-1} \frac{dP_{ij}}{dt} = \lambda_+ \sum_{\{t_i^*\}'} x_j(t) \delta(t - [t_i^* + d_{EE}]) - \lambda_- y_i \sum_{\{t_j^*\}} \delta(t - t_j^*) + \lambda_h \sum_{\{t_i^*\}'} (z^* - z_i(t)) \delta(t - t_i^*). \quad (1)$$

At the boundaries  $P_{\min,ij}$  and  $P_{\max}$ ,  $P_{ij}(t)$  is clipped. While the maximum permanences  $P_{\max}$  are identical for all EE connections, the minimal permanences  $P_{\min,ij}$  are uniformly distributed in the interval  $[P_{0,\min}, P_{0,\max}]$ . The first term on the right-hand side of Eq (1) corresponds to the spike-timing dependent synaptic potentiation triggered by the postsynaptic spikes at times  $t_i^* \in \{t_i^*\}'$ . Here,  $\{t_i^*\}' = \{t_i^* | \exists t_j^* \Delta t_{\min} < t_i^* - t_j^* + d_{EE} < \Delta t_{\max}\}$  denotes the set of all postsynaptic spike times  $t_i^*$  for which there exists a (previous) presynaptic spike  $t_j^*$  such that the time lag  $t_i^* - t_j^* + d_{EE}$  is larger than  $\Delta t_{\min}$  and smaller than  $\Delta t_{\max}$ . The potentiation is restricted to these time lags to avoid a growth of synapses between synchronously active neurons belonging to the same subpopulation, and between neurons encoding for the first elements in different sequences. Note that the potentiation update times lag the somatic postsynaptic spike times by the delay  $d_{EE}$ , which is here interpreted as a purely dendritic delay [27, 30]. The potentiation increment is determined by the dimensionless potentiation rate  $\lambda_+$ , and the spiking trace  $x_j(t)$  of the presynaptic neuron  $j$ , which is updated according to

$$\frac{dx_j}{dt} = -\tau_+^{-1} x_j(t) + \sum_{t_j^*} \delta(t - t_j^*). \quad (2)$$

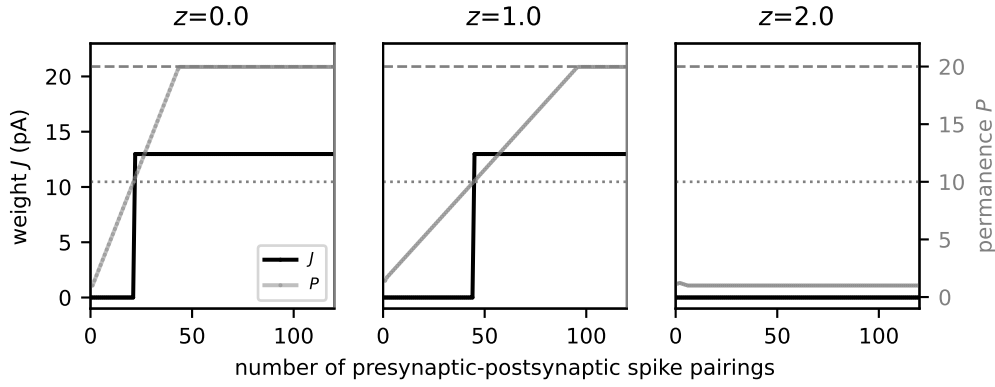
The trace  $x_j(t)$  is incremented by unity at each spike time  $t_j^*$ , followed by an exponential decay with time constant  $\tau_+$ . The potentiation increment  $\Delta P_{ij}$  at time  $t_i^*$  therefore depends on the temporal distance between the postsynaptic spike time  $t_i^*$  and all presynaptic spike times  $t_j^* \leq t_i^*$  (STDP with all-to-all spike pairing; [27]). The second term in Eq (1) represents synaptic depression, and is triggered by each presynaptic spike at times  $t_j^* \in \{t_j^*\}$ . The depression decrement  $y_i = 1$  is treated as a constant, independently of the postsynaptic spike history. The depression magnitude is parameterized by the dimensionless depression rate  $\lambda_-$ . The third term in Eq (1) corresponds to a homeostatic control triggered by postsynaptic spikes at times  $t_i^* \in \{t_i^*\}'$ . Its overall impact is parameterized by the dimensionless homeostasis rate  $\lambda_h$ . The homeostatic control enhances or reduces the synapse growth depending on the dAP trace  $z_i(t)$  of neuron  $i$ , the low-pass filtered dAP activity updated according to

$$\frac{dz_i}{dt} = -\tau_h^{-1} z_i(t) + \sum_k \delta(t - t_{\text{dAP},i}^k). \quad (3)$$

Here,  $\tau_h$  represents the homeostasis time constant, and  $t_{\text{dAP},i}^k$  the onset time of the  $k$ th dAP in neuron  $i$ . According to Eq (1), synapse growth is boosted if the dAP activity  $z_i(t)$  is below a target dAP activity  $z^*$ . Conversely, high dAP activity exceeding  $z^*$  reduces the synapse growth (Fig. 4). This homeostatic regulation of the synaptic maturity controlled by the postsynaptic dAP activity constitutes a variation of previous models [28, 29] describing ‘synaptic scaling’ [31–33]. It counteracts excessive synapse formation during learning driven by Hebbian structural plasticity. In addition, the combination of Hebbian plasticity and synaptic scaling can introduce a competition between synapses [28, 29]. Here, we exploit this effect to ensure that synapses are generated in a context specific manner, and thereby reduce the overlap between neuronal subpopulations activated by the same sequence element occurring in different sequences. To this end, the homeostasis parameters  $z^* = 1$  and  $\tau_h$  are chosen such that each neuron tends to become predictive, i.e., generate a dAP, at most once during the presentation of a single sequence ensemble of total duration  $((C-1)\Delta T + \Delta T_{\text{seq}})S$  (see [Network model](#)). The time constant  $\tau_h$  is hence adapted to the parameters of the task. For sequence sets I and II and the default inter-stimulus interval  $\Delta T = 40$  ms, it is set to  $\tau_h = 440$  ms and  $\tau_h = 1560$  ms, respectively. In section [Dependence of prediction performance on the sequence speed](#), we study the effect of the sequence speed (inter-stimulus interval  $\Delta T$ ) on the prediction performance for a given network parameterization. For these experiments,  $\tau_h = 440$  ms is therefore fixed even though the inter-stimulus interval  $\Delta T$  is varied.

The prefactor  $P_{\max}^{-1}$  in Eq (1) ensures that all learning rates  $\lambda_+$ ,  $\lambda_-$  and  $\lambda_h$  are measured in units of the maximum permanence  $P_{\max}$ .

**Constraints on potential connectivity.** The sequence processing capabilities of the proposed network model rely on its ability to form sequence specific feed-forward subnetworks based on the skeleton provided by the random potential connectivity. On the one hand, the potential connectivity must not be too diluted to ensure that a subset of neurons representing a given sequence element can establish sufficiently many mature connections to a second subset of neurons representing the subsequent element. On the other hand, a dense potential connectivity would promote overlap between subnetworks representing different sequences, and thereby slow down the formation of



**Fig 4: Homeostatic regulation of the spike-timing-dependent structural plasticity by the dAP activity.** Evolution of the synaptic permanence (gray) and weight (black) during repetitive presynaptic-postsynaptic spike pairing for different levels of the dAP activity. In the depicted example, presynaptic spikes precede the postsynaptic spikes by 40 ms for each spike pairing. Consecutive spike pairs are separated by a 200 ms interval. In each panel, the postsynaptic dAP trace is clamped at a different value:  $z = 0$  (left),  $z = 1$  (middle),  $z = 2$  (right). The dAP target activity is fixed at  $z^* = 1$ . The horizontal dashed and dotted lines mark the maximum permanence  $P_{\max}$  and the maturity threshold  $\theta_P$ , respectively.

context specific subnetworks during learning (see [Sequence learning and prediction](#)). Here, we therefore identify the minimal potential connection probability  $p$  guaranteeing the existence of network motifs with a sufficient degree of divergent-convergent connectivity.

Consider the subset  $\mathcal{P}_{ij}$  of  $\rho$  excitatory neurons representing the  $j$ th sequence element  $\zeta_{ij}$  in sequence  $s_i$  (see [Task and training protocol](#) and [Network model](#)). During the learning process, the plasticity dynamics needs to establish mature connections from  $\mathcal{P}_{ij}$  to a second subset  $\mathcal{P}_{i,j+1}$  of neurons in another subpopulation representing the subsequent element  $\zeta_{i,j+1}$ . Each neuron in  $\mathcal{P}_{i,j+1}$  must receive at least  $c = \lceil \theta_{\text{dAP}}/W \rceil$  inputs from  $\mathcal{P}_{ij}$  to ensure that synchronous firing of the neurons in  $\mathcal{P}_{ij}$  can evoke a dAP after synapse maturing. For a random, homogeneous potential connectivity with connection probability  $p$ , the probability of finding these  $c$  potential connections for some arbitrary target neuron is given by

$$q(c; \rho, p) = \sum_{k=c}^{\rho} \binom{\rho}{k} p^k (1-p)^{\rho-k}. \quad (4)$$

For a successful formation of sequence specific subnetworks during learning, the sparse subset  $\mathcal{P}_{ij}$  of presynaptic neurons needs to recruit at least  $\rho$  targets in the set of  $n_E$  neurons representing the subsequent sequence element (Fig. 5A). The probability of observing such a divergent-convergent connectivity motif is given by

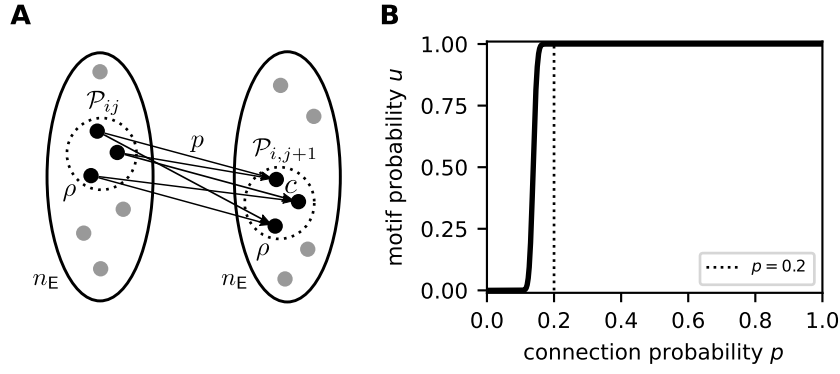
$$u(\rho; c, p, n_E) = \sum_{l=\rho}^{n_E} \binom{n_E}{l} q^l (1-q)^{n_E-l}. \quad (5)$$

Note that the above described motif does not require the size of the postsynaptic subset  $\mathcal{P}_{i,j+1}$  to be exactly  $\rho$ . Eq (5) constrains the parameters  $p$ ,  $c$ ,  $n_E$  and  $\rho$  to ensure such motifs exist in a random network. Fig. 5B illustrates the dependence of the motif probability  $u$  on the connection probability  $p$  for our choice of parameters  $n_E$ ,  $c$ , and  $\rho$ . For  $p \geq 0.2$ , the existence of the divergent-convergent connectivity motif is almost certain ( $u \approx 1$ ). For smaller connection probabilities  $p < 0.2$ , the motif probability quickly vanishes. Hence,  $p = 0.2$  constitutes a reasonable choice for the potential connection probability.

**Network realizations and initial conditions.** For every network realization, the potential connectivity and the initial permanences are drawn randomly and independently. All other parameters are identical for different network realizations. The initial values of all state variables are given in Table 1 and Table 2.

**Simulation details.** The network simulations are performed in the neural simulator NEST [34] under version 3.0 [35]. The differential equations and state transitions defining the excitatory neuron dynamics are expressed in the domain specific language NESTML [36, 37] which generates the required C++ code for the dynamic loading into NEST. Network states are synchronously updated using exact integration of the system dynamics on a discrete-time grid with step size  $\Delta t$  [38]. The full source code for the implementation with a list of other software requirements is available at Zenodo: <https://doi.org/10.5281/zenodo.5578212>.





**Fig 5: Existence of divergent-convergent connectivity motifs in a random network.** **A)** Sketch of the divergent-convergent potential connectivity motif required for the formation of sequence specific subnetworks during learning. See main text for details. **B)** Dependence of the motif probability  $u$  on the connection probability  $p$  for  $n_E = 150$ ,  $c = 5$ , and  $\rho = 20$  (see Table 2). The dotted vertical line marks the potential connection probability  $p = 0.2$  used in this study.

### Task performance measures

To assess the network performance, we monitor the dendritic currents reporting predictions (dAPs) as well as the somatic spike times of excitatory neurons. To quantify the prediction error, we identify for each last element  $\zeta_{i,C_i}$  in a sequence  $s_i$  all excitatory neurons that have generated a dAP in the time interval  $(t_{i,C_i} - \Delta T, t_{i,C_i})$ , where  $t_{i,C_i}$  and  $\Delta T$  denote the time of the external input corresponding to the last sequence element  $\zeta_{i,C_i}$  and the inter-stimulus interval, respectively (see [Task and training protocol](#) and [Network model](#)). All subpopulations  $\mathcal{M}_k$  with at least  $\rho/2$  neurons generating a dAP are considered “predictive”. The prediction state of the network is encoded in an  $M$  dimensional binary vector  $\mathbf{o}$ , where  $o_k = 1$  if the  $k$ th subpopulation is predictive, and  $o_k = 0$  else. The

$$\text{prediction error} = \frac{1}{L} \sqrt{\sum_{k=1}^M (o_k - v_k)^2} \quad (6)$$

is defined as the Euclidean distance between  $\mathbf{o}$  and the binary target vector  $\mathbf{v}$  representing the pattern of external inputs for each last element  $\zeta_{i,C_i}$ , normalized by the number  $L$  of subpopulations per sequence element. Furthermore, we assess the

$$\text{false positive rate} = \frac{1}{L} \sum_{k=1}^M \Theta(o_k - v_k) \quad (7)$$

and the

$$\text{false negative rate} = \frac{1}{L} \sum_{k=1}^M \Theta(v_k - o_k), \quad (8)$$

where  $\Theta(\cdot)$  denotes the Heaviside function. In addition to these performance measures, we monitor for each last sequence element the level of sparsity by measuring the ratio between the number of active neurons and the total number  $Ln_E$  of neurons representing this element. During learning, we expose the network repetitively to the same set  $\{s_1, \dots, s_S\}$  of sequences for a number of training episodes  $K$ . To obtain the total prediction performance in each episode, we average the prediction error, the false negative and false positive rates, as well as the level of sparsity across the set of sequences.

Summary		
Populations	excitatory neurons ( $\mathcal{E}$ ), inhibitory neurons ( $\mathcal{I}$ ), external spike sources ( $\mathcal{X}$ ); $\mathcal{E}$ and $\mathcal{I}$ composed of $M$ disjoint subpopulations $\mathcal{M}_k$ and $\mathcal{I}_k$ ( $k = 1, \dots, M$ )	
Connectivity	<ul style="list-style-type: none"><li>• sparse random connectivity between excitatory neurons (plastic)</li><li>• local recurrent connectivity between excitatory and inhibitory neurons (static)</li></ul>	
Neuron model	<ul style="list-style-type: none"><li>• excitatory neurons: leaky integrate-and-fire (LIF) with nonlinear input integration (dendritic action potentials)</li><li>• inhibitory neurons: leaky integrate-and-fire (LIF)</li></ul>	
Synapse model	exponential or alpha-shaped postsynaptic currents (PSCs)	
Plasticity	homeostatic spike-timing dependent structural plasticity in excitatory-to-excitatory connections	

Populations		
Name	Elements	Size
$\mathcal{E} = \cup_{i=k}^M \mathcal{M}_k$	excitatory (E) neurons	$N_E$
$\mathcal{I} = \cup_{i=k}^M \mathcal{I}_k$	inhibitory (I) neurons	$N_I$
$\mathcal{M}_k$	excitatory neurons in subpopulation $k$ , $\mathcal{M}_k \cap \mathcal{M}_l = \emptyset$ ( $\forall k \neq l \in [1, M]$ )	$n_E$
$\mathcal{I}_k$	inhibitory neurons in subpopulation $k$ , $\mathcal{I}_k \cap \mathcal{I}_l = \emptyset$ ( $\forall k \neq l \in [1, M]$ )	$n_I$
$\mathcal{X} = \{x_1, \dots, x_M\}$	external spike sources	$M$

Connectivity		
Source pop-ulation	Target popu-lation	Pattern
$\mathcal{E}$	$\mathcal{E}$	random; fixed in-degrees $K_i = K_{EE}$ , delays $d_{ij} = d_{EE}$ , synaptic time constants $\tau_{ij} = d_{EE}$ ; plastic weights $J_{ij}$ ( $\forall i \in \mathcal{E}, \forall j \in \mathcal{E}$ ; "EE connections")
$\mathcal{M}_k$	$\mathcal{I}_k$	all-to-all; fixed delays $d_{ij} = d_{IE}$ , synaptic time constants $\tau_{ij} = \tau_{IE}$ , and weights $J_{ij} = J_{IE}$ ( $\forall i \in \mathcal{M}_k, \forall j \in \mathcal{I}_k, \forall k \in [1, M]$ ; "IE connections")
$\mathcal{I}_k$	$\mathcal{M}_k$	all-to-all; fixed delays $d_{ij} = d_{EI}$ , synaptic time constants $\tau_{ij} = \tau_{EI}$ , and weights $J_{ij} = J_{EI}$ ( $\forall i \in \mathcal{I}_k, \forall j \in \mathcal{M}_k, \forall k \in [1, M]$ ; "EI connections")
$\mathcal{I}_k$	$\mathcal{I}_k$	none ( $\forall k \in [1, M]$ ; "II connections")
$\mathcal{X}_k = x_k$	$\mathcal{M}_k$	one-to-all; fixed delays $d_{ik} = d_{EX}$ , synaptic time constants $\tau_{ij} = \tau_{EX}$ , and weights $J_{ik} = J_{EX}$ ( $\forall i \in \mathcal{M}_k, \forall k \in [1, M]$ ; "EX connections")
no self-connections ("autapses"), no multiple connections ("multapses")		
all unmentioned connections $\mathcal{M}_k \rightarrow \mathcal{I}_l, \mathcal{I}_k \rightarrow \mathcal{M}_l, \mathcal{I}_k \rightarrow \mathcal{I}_l, \mathcal{X}_k \rightarrow \mathcal{M}_l$ ( $\forall k \neq l$ ) are absent		

**Table 1:** Description of the network model (continued on next page). Parameter values are given in Table 2.

Neuron and synapse	
Neuron	
Type	leaky integrate-and-fire (LIF) dynamics
Description	<p>dynamics of membrane potential <math>V_i(t)</math> and spiking activity <math>s_i(t)</math> of neuron <math>i</math>:</p> <ul style="list-style-type: none"> <li>emission of the <math>k</math>th spike of neuron <math>i</math> at time <math>t_i^k</math> if <math display="block">V_i(t_i^k) \geq \theta_i \quad (9)</math> <p>with somatic spike threshold <math>\theta_i</math></p> </li> <li>spike train: <math>s_i(t) = \sum_k \delta(t - t_i^k)</math></li> <li>reset and refractoriness: <math display="block">V_i(t) = V_r \quad \forall k, \forall t \in \left( t_i^k, t_i^k + \tau_{\text{ref},i} \right]</math> <p>with refractory time <math>\tau_{\text{ref},i}</math> and reset potential <math>V_r</math></p> </li> <li>subthreshold dynamics: <math display="block">\tau_{m,i} \dot{V}_i(t) = -V_i(t) + R_{m,i} I_i(t) \quad (10)</math> <p>with membrane resistance <math>R_{m,i} = \frac{\tau_{m,i}}{C_{m,i}}</math>, membrane time constant <math>\tau_{m,i}</math>, and total synaptic input current <math>I_i(t)</math> (see Synapse)</p> </li> <li>excitatory neurons: <math>\tau_{m,i} = \tau_{m,E}</math>, <math>C_{m,i} = C_m</math>, <math>\theta_i = \theta_E</math>, <math>\tau_{\text{ref},i} = \tau_{\text{ref},E}</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>inhibitory neurons: <math>\tau_{m,i} = \tau_{m,I}</math>, <math>C_{m,i} = C_m</math>, <math>\theta_i = \theta_I</math>, <math>\tau_{\text{ref},i} = \tau_{\text{ref},I}</math> (<math>\forall i \in \mathcal{I}</math>)</li> </ul>
Synapse	
Type	exponential or alpha-shaped postsynaptic currents (PSCs)
Description	<ul style="list-style-type: none"> <li>total synaptic input currents: <math display="block">\begin{aligned} \text{excitatory neurons: } I_i(t) &amp;= I_{\text{ED},i}(t) + I_{\text{EX},i}(t) + I_{\text{EI},i}(t), \quad \forall i \in \mathcal{E} \\ \text{inhibitory neurons: } I_i(t) &amp;= I_{\text{IE},i}(t), \quad \forall i \in \mathcal{I} \end{aligned} \quad (11)</math> <p>with dendritic, external, inhibitory and excitatory input currents <math>I_{\text{ED},i}(t)</math>, <math>I_{\text{EX},i}(t)</math>, <math>I_{\text{EI},i}(t)</math>, <math>I_{\text{IE},i}(t)</math> evolving according to</p> <math display="block">I_{\text{ED},i}(t) = \sum_{j \in \mathcal{E}} (\alpha_{ij} * s_j)(t - d_{ij}) \quad (12)</math> <p>with <math>\alpha_{ij}(t) = J_{ij} \frac{e}{\tau_{\text{ED}}} t e^{-t/\tau_{\text{ED}}} \Theta(t)</math> and <math>\Theta(t) = \begin{cases} 1 &amp; t \geq 0 \\ 0 &amp; \text{else} \end{cases}</math>,</p> <math display="block">\tau_{\text{EX}} \dot{I}_{\text{EX},i} = -I_{\text{EX},i}(t) + \sum_{j \in \mathcal{X}} J_{ij} s_j(t - d_{ij}), \quad (13)</math> <math display="block">\tau_{\text{EI}} \dot{I}_{\text{EI},i} = -I_{\text{EI},i}(t) + \sum_{j \in \mathcal{I}} J_{ij} s_j(t - d_{ij}), \quad (14)</math> <math display="block">\tau_{\text{IE}} \dot{I}_{\text{IE},i} = -I_{\text{IE},i}(t) + \sum_{j \in \mathcal{E}} J_{ij} s_j(t - d_{ij}) \quad (15)</math> </li> <li>suprathreshold dynamics of dendritic currents (dAP generation): <ul style="list-style-type: none"> <li>emission of <math>k</math>th dAP of neuron <math>i</math> at time <math>t_{\text{dAP},i}^k</math> if <math>I_{\text{ED},i}(t_{\text{dAP},i}^k) \geq \theta_{\text{dAP}}</math></li> <li>dAP current plateau: <math display="block">I_{\text{ED},i}(t) = I_{\text{dAP}} \quad \forall k, \forall t \in \left( t_{\text{dAP},i}^k, t_{\text{dAP},i}^k + \tau_{\text{dAP}} \right) \quad (16)</math> <p>with dAP current plateau amplitude <math>I_{\text{dAP}}</math>, dAP current duration <math>\tau_{\text{dAP}}</math>, and dAP activation threshold <math>\theta_{\text{dAP}}</math></p> </li> <li>reset: <math>I_{\text{ED},i}(t_{\text{dAP},i}^k + \tau_{\text{dAP}}) = 0</math> (<math>\forall k</math>)</li> <li>reset and refractoriness in response to emission of <math>l</math>th somatic spike of neuron <math>i</math> at time <math>t_i^l</math>: <math display="block">I_{\text{ED},i}(t) = 0 \quad \forall l, \forall t \in \left( t_i^l, t_i^l + \tau_{\text{ref},i} \right) \quad (17)</math> </li> </ul> </li> </ul>

**Table 1:** Description of the network model (continued on next page). Parameter values are given in Table 2.

Plasticity	
Type	spike-timing dependent structural plasticity and dAP-rate homeostasis
EE synapses	<ul style="list-style-type: none"> <li>• dynamics of synaptic permanence <math>P_{ij}(t)</math> (synapse maturity): <math display="block">P_{\max}^{-1} \frac{dP_{ij}}{dt} = \lambda_+ \sum_{\{t_i^*\}'} x_j(t) \delta(t - [t_i^* + d_{EE}]) - \lambda_- y_i \sum_{\{t_j^*\}} \delta(t - t_j^*) + \lambda_h \sum_{\{t_i^*\}'} (z^* - z_i(t)) \delta(t - t_i^*)</math> </li> </ul> <p>with</p> <ul style="list-style-type: none"> <li>– list of presynaptic spike times <math>\{t_j^*\}</math>,</li> <li>– list of postsynaptic spike times <math>\{t_i^*\}' = \{t_i^*   \exists t_j^* \Delta t_{\min} &lt; t_i^* - t_j^* + d_{EE} &lt; \Delta t_{\max}\}</math>,</li> <li>– maximum permanence <math>P_{\max}</math>, potentiation and depression rates <math>\lambda_+</math>, <math>\lambda_-</math>, homeostasis rate <math>\lambda_h</math>, delay <math>d_{EE}</math>, depression decrement <math>y_i</math>,</li> <li>– spike trace <math>x_j(t)</math> of presynaptic neuron <math>j</math>, evolving according to <math display="block">\frac{dx_j}{dt} = -\tau_+^{-1} x_j(t) + \sum_{t_j^*} \delta(t - t_j^*)</math> </li> </ul> <p>with presynaptic spike times <math>t_j^*</math> and potentiation time constant <math>\tau_+</math>,</p> <ul style="list-style-type: none"> <li>– dAP trace <math>z_i(t)</math> of postsynaptic neuron <math>i</math>, evolving according to <math display="block">\frac{dz_i}{dt} = -\tau_h^{-1} z_i(t) + \sum_k \delta(t - t_{\text{dAP},i}^k)</math> </li> </ul> <p>with onset time <math>t_{\text{dAP},i}^k</math> of the <math>k</math>th dAP, homeostasis time constant <math>\tau_h</math>, and</p> <ul style="list-style-type: none"> <li>– target dAP activity <math>z^*</math></li> </ul> <ul style="list-style-type: none"> <li>• dynamics of synaptic weights <math>J_{EE,ij}</math>: <math display="block">J_{EE,ij}(t) = \begin{cases} W &amp; \text{if } P_{ij}(t) \geq \theta_P \quad (\text{mature synapse}) \\ 0 &amp; \text{if } P_{ij}(t) &lt; \theta_P \quad (\text{immature synapse}) \end{cases}</math> </li> </ul> <p>with weight of mature EE connections <math>W</math> and synapse maturity threshold <math>\theta_P</math></p> <p>(for an algorithmic implementation of the plasticity dynamics, see Algorithm S1)</p>
all other synapses	non-plastic

**Table 1:** Description of the network model (continued on next page). Parameter values are given in Table 2.

Input
<ul style="list-style-type: none"> <li>• prediction mode <ul style="list-style-type: none"> <li>– repetitive stimulation with the same set <math>\mathcal{S} = \{s_1, \dots, s_S\}</math> of sequences <math>s_i = \{\zeta_{i,1}, \zeta_{i,2}, \dots, \zeta_{i,C_i}\}</math> of ordered discrete items <math>\zeta_{i,j}</math> with number of sequences <math>S</math> and length <math>C_i</math> of <math>i</math>th sequence</li> <li>– presentation of sequence element <math>\zeta_{i,j}</math> at time <math>t_{i,j}</math> modeled by single spike <math>x_k(t) = \delta(t - t_{i,j})</math>, generated by the corresponding external source <math>x_k</math></li> <li>– inter-stimulus interval <math>\Delta T = t_{i,j+1} - t_{i,j}</math> between subsequent sequence elements <math>\zeta_{i,j}</math> and <math>\zeta_{i,j+1}</math> within a sequence <math>s_i</math></li> <li>– inter-sequence time interval <math>\Delta T_{\text{seq}} = t_{i+1,1} - t_{i,C_i}</math> between subsequent sequences <math>s_i</math> and <math>s_{i+1}</math></li> <li>– example sequence sets: <ul style="list-style-type: none"> <li>* sequence set I: <math>\mathcal{S} = \{\{A,D,B,E\} \text{ and } \{F,D,B,C\}\}</math></li> <li>* sequence set II: <math>\mathcal{S} = \{\{E,N,D,I,J\}, \{L,N,D,I,K\}, \{G,J,M,C,N\}, \{F,J,M,C,I\}, \{B,C,K,H,I\}, \{A,C,K,H,F\}\}</math></li> </ul> </li> </ul> </li> <li>• replay mode <ul style="list-style-type: none"> <li>– presentation of a cue encoding for first sequence elements <math>\zeta_{i,1}</math> at <math>t_{i,1}</math></li> <li>– inter-cue time interval <math>\Delta T_{\text{cue}} = t_{i+1,1} - t_{i,1}</math> between subsequent cues <math>\zeta_{i,1}</math> and <math>\zeta_{i+1,1}</math></li> </ul> </li> </ul>
Output
<ul style="list-style-type: none"> <li>• somatic spike times <math>\{t_i^k   \forall i \in \mathcal{E}, k = 1, 2, \dots\}</math></li> <li>• dendritic currents <math>I_{\text{ED},i}(t)</math> (<math>\forall i \in \mathcal{E}</math>)</li> </ul>
Initial conditions and network realizations
<ul style="list-style-type: none"> <li>• membrane potentials: <math>V_i(0) = V_r</math> (<math>\forall i \in \mathcal{E} \cup \mathcal{I}</math>)</li> <li>• dendritic currents: <math>I_{\text{ED},i}(0) = 0</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>• external currents: <math>I_{\text{EX},i}(0) = 0</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>• inhibitory currents: <math>I_{\text{EI},i}(0) = 0</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>• excitatory currents: <math>I_{\text{IE},i}(0) = 0</math> (<math>\forall i \in \mathcal{I}</math>)</li> <li>• synaptic permanences: <math>P_{ij}(0) = P_{\min,ij}</math> with <math>P_{\min,ij} \sim \mathcal{U}(P_{0,\min}, P_{0,\max})</math> (<math>\forall i, j \in \mathcal{E}</math>)</li> <li>• synaptic weights: <math>J_{\text{EE},ij}(0) = 0</math> (<math>\forall i, j \in \mathcal{E}</math>)</li> <li>• spike traces: <math>x_i(0) = 0</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>• dAP traces: <math>z_i(0) = 0</math> (<math>\forall i \in \mathcal{E}</math>)</li> <li>• potential connectivity and initial permanences randomly and independently drawn for each network realization</li> </ul>
Simulation details
<ul style="list-style-type: none"> <li>• network simulations performed in NEST [34] version 3.0 [35]</li> <li>• definition of excitatory neuron model using NESTML [36, 37]</li> <li>• synchronous update using exact integration of system dynamics on discrete-time grid with step size <math>\Delta t</math> [38]</li> <li>• source code underlying this study: <a href="https://doi.org/10.5281/zenodo.5578212">https://doi.org/10.5281/zenodo.5578212</a></li> </ul>

**Table 1:** Description of the network model. Parameter values are given in Table 2.

Name	Value	Description
<b>Network</b>		
$N_E$	2100	total number of excitatory neurons
$N_I$	14	total number of inhibitory neurons
$M$	$A = 14$	number of excitatory subpopulations (= number of external spike sources)
$n_E$	$N_E/M = 150$	number of excitatory neurons per subpopulation
$n_I$	$N_I/M = 1$	number of inhibitory neurons per subpopulation
$\rho$	20	(target) number of active neurons per subpopulation after learning = minimal number of coincident excitatory inputs required to trigger a spike in postsynaptic inhibitory neurons
<b>(Potential) Connectivity</b>		
$K_{EE}$	420	number of excitatory inputs per excitatory neuron (EE in-degree)
$p$	$K_{EE}/N_E = 0.2$	probability of potential (excitatory) connections
$K_{EI}$	$n_I = 1$	number of inhibitory inputs per excitatory neuron (EI in-degree)
$K_{IE}$	$n_E$	number of excitatory inputs per inhibitory neuron (IE in-degree)
$K_{II}$	0	number of inhibitory inputs per inhibitory neuron (II in-degree)
<b>Excitatory neurons</b>		
$\tau_{m,E}$	10 ms	membrane time constant
$\tau_{ref,E}$	10 ms	absolute refractory period
$C_m$	250 pF	membrane capacity
$V_r$	0.0 mV	reset potential
$\theta_E$	20 mV (predictive mode), 5 mV (replay mode)	somatic spike threshold
$I_{dAP}$	200 pA	dAP current plateau amplitude
$\tau_{dAP}$	60 ms	dAP duration
$\theta_{dAP}$	59 pA	dAP threshold
<b>Inhibitory neurons</b>		
$\tau_{m,I}$	5 ms	membrane time constant
$\tau_{ref,I}$	2 ms	absolute refractory period
$C_m$	250 pF	membrane capacity
$V_r$	0.0 mV	reset potential
$\theta_I$	15 mV	spike threshold

**Table 2:** Model and simulation parameters (continued on next page). Parameters derived from other parameters are marked in gray.



Name	Value	Description
<b>Synapse</b>		
$W$	12 pA	weight of mature EE connections (EPSC amplitude)
$\tilde{J}_{IE}$	$\theta_1 = 0.75$ mV	weight of IE connections (EPSP amplitude)
$J_{IE}$	581.19 pA	weight of IE connections (EPSC amplitude)
$\tilde{J}_{EI}$	-22.5 mV	weight of EI connections (IPSP amplitude)
$J_{EI}$	-12915.49 pA	weight of EI connections (IPSC amplitude)
$\tilde{J}_{EX}$	22 mV	weight of EX connections (EPSP amplitude)
$J_{EX}$	4112.20 pA	weight of EX connections (EPSC amplitude)
$\tau_{EE}$	5 ms	synaptic time constant of EE connections
$\tau_{IE}$	0.5 ms	synaptic time constant of IE connections
$\tau_{EI}$	1 ms	synaptic time constant of EI connections
$\tau_{EX}$	2 ms	synaptic time constant of EX connection
$d_{EE}$	2 ms	delay of EE connections (dendritic)
$d_{IE}$	0.1 ms	delay of IE connections
$d_{EI}$	0.1 ms	delay of EI connections
$d_{EX}$	0.1 ms	delay of EX connections
<b>Plasticity</b>		
$\lambda_+$	0.08 (sequence set I), 0.28 (sequence set II)	potentiation rate
$\lambda_-$	0.0015 (sequence set I), 0.0061 (sequence set II)	depression rate
$\theta_P$	20	synapse maturity threshold
$P_{\min,ij}$	$\sim \mathcal{U}(P_{0,\min}, P_{0,\max})$	minimum permanence
$P_{\max}$	20	maximum permanence
$P_{0,\min}$	0	minimal initial permanence
$P_{0,\max}$	8	maximal initial permanence
$\tau_+$	20 ms	potentiation time constant
$z^*$	1	target dAP activity
$\lambda_h$	0.014 (sequence set I), 0.024 (sequence set II)	homeostasis rate
$\tau_h$	440 ms (sequence set I), 1560 ms (sequence set II)	homeostasis time constant
$y_i$	1	depression decrement
$\Delta t_{\min}$	4 ms	minimum time lag between pairs of pre- and postsynaptic spikes at which synapses are potentiated
$\Delta t_{\max}$	$2\Delta T$	maximum time lag between pairs of pre- and postsynaptic spikes at which synapses are potentiated
<b>Input</b>		
$L$	1	number of subpopulations per sequence element = number of target subpopulations per spike source
$S$	2 (sequence set I), 6 (sequence set II)	number of sequences per set
$C$	4 (sequence set I), 5 (sequence set II)	number of elements per sequence
$A$	14	alphabet length (total number of distinct sequence elements)
$\Delta T$	$\{2, \dots, 40, \dots, 90\}$ ms	inter-stimulus interval
$\Delta T_{\text{seq}}$	$2.5\Delta T$	inter-sequence interval
$\Delta T_{\text{cue}}$	80 ms	inter-cue interval
<b>Simulation</b>		
$\Delta t$	0.1 ms	time resolution
$K$	$\{80, 150\}$	number of training episodes

**Table 2:** Model and simulation parameters (continued). Parameters derived from other parameters are marked in gray. Bold numbers depict default values.

## Results

### Sequence learning and prediction

According to the Temporal Memory (TM) model, sequences are stored in the form of specific paths through the network. Prediction and replay of sequences correspond to a sequential sparse activation of small groups of neurons along these paths. Non-anticipated stimuli are signaled in the form of non-sparse firing of these groups. This subsection describes how the model components introduced in [Network model](#) interact and give rise to the network structure and behavior postulated by TM. For illustration, we here consider a simple set of two partly overlapping sequences  $\{A, D, B, E\}$  and  $\{F, D, B, C\}$  corresponding to the sequence set I (see Fig. 1B).

The initial sparse, random and immature network connectivity (Fig. 2A,B) constitutes the skeleton on which the sequence-specific paths will be carved out during the learning process. To guarantee a successful learning, this initial skeleton must be neither too sparse nor too dense (see [Methods](#)). Before learning, the presentation of a particular sequence element causes all neurons with the corresponding stimulus preference to reliably and synchronously fire a somatic action potential due to the strong, suprathreshold external stimulus (Fig. 3A). All other subpopulations remain silent (see Fig. 6A,B). The lateral connectivity between excitatory neurons belonging to the different subpopulations is subject to a form of Hebbian structural plasticity. Repetitive and consistent sequential presentation of sequence elements turns immature connections between successively activated subpopulations into mature connections, and hence leads to the formation of sequence-specific feed-forward subnetworks (see Fig. 2C,D). Synaptic depression prunes connections not supporting the learned pattern, thereby reducing the chance of predicting wrong sequence items (false positives).

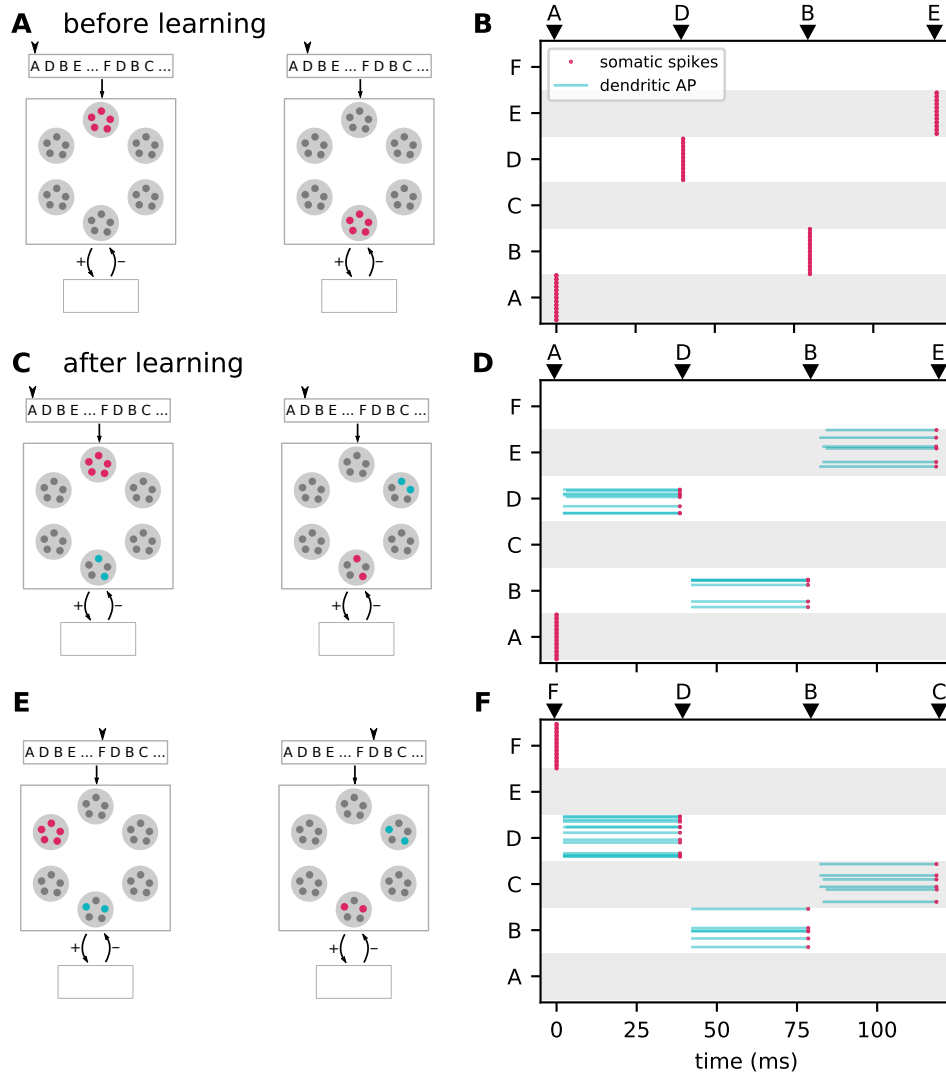
During the learning process, the number of mature connections grows to a point where the activation of a certain subpopulation by an external input generates dendritic action potentials (dAPs), a “prediction”, in a subset of neurons in the subsequent subpopulation (blue neurons in Fig. 6C). The dAPs generate a long-lasting depolarization of the soma (Fig. 3B). When receiving an external input, these depolarized neurons fire slightly earlier as compared to non-depolarized (non-predictive) neurons (Fig. 3A,B,D). If the number of predictive neurons within a subpopulation is sufficiently large, their advanced spikes (Fig. 3C) initiate a fast and strong inhibitory feedback to the entire subpopulation, and thereby suppress subsequent firing of non-predictive neurons in this population (Fig. 6C,D). Owing to this winner-take-all dynamics, the network generates sparse spiking in response to predicted stimuli, i.e., if the external input coincides with a dAP-triggered somatic depolarization. In the presence of a non-anticipated, non-predicted stimulus, the neurons in the corresponding subpopulation fire collectively in a non-sparse manner, thereby signaling a “mismatch”.

In the model presented in this study, the initial synapse maturity levels, the permanences, are randomly chosen within certain bounds. During learning, connections with a higher initial permanence mature first. This heterogeneity in the initial permanences enables the generation of sequence specific sparse connectivity patterns between subsequently activated neuronal subpopulations (Fig. 2D). For each pair of sequence elements in a given sequence ensemble, there is a unique set of postsynaptic neurons generating dAPs (Fig. 6D). These different activation patterns capture the context specificity of predictions. When exposing a network that has learned the two sequences  $\{A, D, B, E\}$  and  $\{F, D, B, C\}$  to the elements “A” and “F”, different subsets of neurons are activated in “D” and “B”. By virtue of these sequence specific activation patterns, stimulation by  $\{A, D, B\}$  or  $\{F, D, B\}$  leads to correct predictions “E” or “C”, respectively (Fig. 6C–F).

Heterogeneity in the permanences alone, however, is not sufficient to guarantee context specificity. The subsets of neurons activated in different contexts may still exhibit a considerable overlap. This overlap is promoted by Hebbian plasticity in the face of the initial non-sparse activity, which leads to a strengthening of connections to neurons in the postsynaptic population in an unspecific manner (Fig. 7A,B). Moreover, the reoccurrence of the same sequence elements in different co-learned sequences initially causes higher firing rates of the neurons in the respective populations (“D” and “B” in Fig. 7). As a result, the formation of unspecific connections would even be accelerated if synapse formation was driven by Hebbian plasticity alone. The model in this study counteracts this loss of context specificity by supplementing the plasticity dynamics with a homeostatic component, which regulates synapse growth based on the rate of postsynaptic dAPs. This form of homeostasis prevents the same neuron from becoming predictive multiple times within the same set of sequences, and thereby reduces the overlap between subsets of neurons activated within different contexts (Fig. 7C, Fig. S3). To further aid the formation of context specific paths, the density of the initial potential connectivity skeleton is set close to the minimum value ensuring existence of the connectivity motifs required for a faithful prediction (see [Methods](#)).

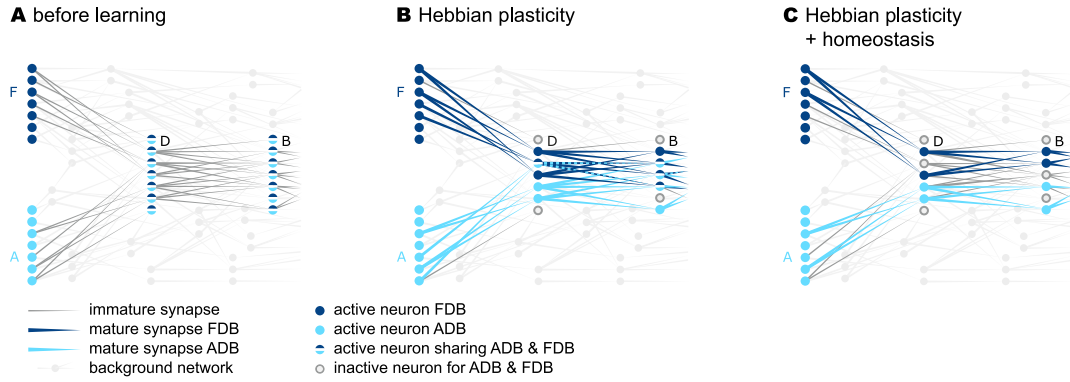
### Prediction performance

To quantify the sequence prediction performance, we repetitively stimulate the network with the sequences in sequence set I (see [Task and training protocol](#)), and continuously monitor the prediction error, the false-positive and false-negative rates, as well as the fraction of active stimulated neurons as a measure of encoding sparsity (Fig. 8; [Task performance measures](#)). To ensure the performance results are not specific to a single network, the evaluation is



**Fig 6: Context specific predictions.** Sketches (left column) and raster plots of network activity (right column) before (top row) and after learning of the two sequences  $\{A,D,B,E\}$  and  $\{F,D,B,C\}$  (middle and bottom rows). In the left column, large light gray circles depict the excitatory subpopulations (same arrangement as in Fig. 2). Red, blue and gray circles mark active, predictive and silent neurons, respectively. In the right column, red dots and blue lines mark somatic spikes and dAP plateaus, respectively. Type and timing of presented stimuli are depicted by black arrows. **A,B**) Snapshots of network activity upon subsequent presentation of the sequence elements “A” and “D” (panel A), and network activity in response to presentation of the entire sequence  $\{A,D,B,E\}$  (panel B) before learning. All neurons in the stimulated subpopulations become active. **C,D**) Same as panels A and B, but after learning. Presenting the first element “A” causes all neurons in the corresponding subpopulations to fire. Activation of these neurons triggers dAPs (predictions) in a subset of neurons representing the subsequent element “D”. When the next element “D” is presented, only these predictive neurons become active, leading to predictions in the subpopulation representing the subsequent subpopulation (“B”), etc. **E,F**) Same as panels C and D, but for sequence  $\{F,D,B,C\}$ . The subsets of neurons representing “D” and “B” activated during sequences  $\{A,D,B,E\}$  and  $\{F,D,B,C\}$  are distinct, i.e., context specific. For clarity, panels B, D, and F show only a fraction of excitatory neurons (30%).

repeated for a number of randomly instantiated network realizations with different initial potential connectivities. At the beginning of the learning process, all neurons of a stimulated subpopulation collectively fire in response to the external input. Non-stimulated neurons remain silent. As the connectivity is still immature at this point, no dAPs are triggered in postsynaptic neurons, and, hence, no predictions are generated. As a consequence, the prediction error, the false-negative rate and the number of active neurons (in stimulated populations) are at their maximum, and the false positive rate is zero (Fig. 8). During the first training episodes, the consistent collective firing of subsequently activated populations leads to the formation of mature connections as a result of the Hebbian structural plasticity. Upon reaching of a critical number of mature synapse, first dAPs (predictions) are generated in postsynaptic cells (in Fig. 8, this happens after about 10 learning episodes). As a consequence, the false negative rate decreases, and the stimulus responses become more sparse. At this early phase of the learning, the predictions of upcoming sequence elements are not yet context specific (for sequence set I, non-sparse activity in “B” triggers a prediction in both “E”



**Fig 7: dAP-rate homeostasis enhances context specificity.** **A)** Sketch of subpopulations of excitatory neurons representing the elements of the two sequences  $\{F, D, B\}$  and  $\{A, D, B\}$ , depicted by light and dark blue colors, respectively. Before learning, the connections between the subpopulations are immature (gray lines). Hence, for each element presentation, all neurons in the respective subpopulations fire (filled circles). **B)** Hebbian plasticity drives the formation of mature connections between subpopulations representing successive sequence elements (colored lines), and leads to sparse firing. The sets of neurons contributing to the two sequences partly overlap. **C)** Incorporating dAP-rate homeostasis reduces this overlap in the activation patterns.

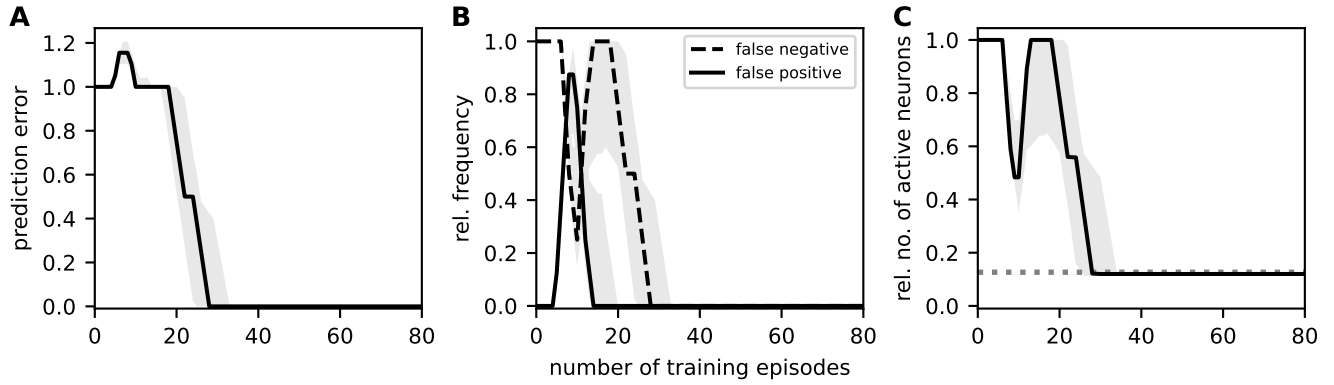
and “C”, irrespective of the context). Hence, the false-positive rate transiently increases. As the context specific connectivity is not consolidated at this point, more and more presynaptic subpopulations fail at triggering dAPs in their postsynaptic targets when they switch to sparse firing. Therefore, the false-positive rate decreases again, and the false-negative rate increases. In other words, there exists a negative feedback loop in the interim learning dynamics where the generation of predictions leads to an increase in sparsity which, in turn, causes prediction failures (and, hence, non-sparse firing). With an increasing number of training episodes, synaptic depression and homeostatic regulation increase context selectivity and thereby break this loop. Eventually, sparse firing of presynaptic populations is sufficient to reliably trigger predictions in their postsynaptic targets. For sequence set I, the total prediction error becomes zero and the stimulus responses maximally sparse after about 30 training episodes (Fig. 8). For a time resolved visualization of the learning dynamics, see Video S1.

Up to this point, we illustrated the model’s sequence learning dynamics and performance for a simple set of two sequences (sequence set I). In the following, we assess the network’s sequence prediction performance for a more complex sequence set (II) composed of five high-order sequences (see [Task and training protocol](#)), each consisting of five elements. This sequence set is comparable to the one used in [14], but contains a larger amount of overlap between sequences. The overall pattern of the learning dynamics resembles the one reported for sequence set I (Fig. 9). The prediction error, the false-positive and false-negative rates as well as the sparsity measure vary more smoothly, and eventually converge at minimal levels after about 40 training episodes. To compare the spiking TM model with the original, non-spiking TM model, we repeat the experiment based on the simulation code provided in [14], see Table S1. With our parameterization, the learning rates  $\lambda_+$  and  $\lambda_-$  of the spiking model are by a factor of about 10 smaller than in the original model. As a consequence, learning sequence set II with the original model converges faster than with the spiking model (compare black and gray curves in Fig. 9). The ratio in learning speeds, however, is not larger than about 2. Increasing the learning rates, i.e., the permanence increments, would speed up the learning process in the spiking model, but bears the risk that a large fraction of potential connections mature simultaneously. This would effectively overwrite the permanence heterogeneity which is essential to form context specific connectivity patterns (see [Sequence learning and prediction](#)). As a result, the network performance would decrease. The original model avoids this problem by limiting the number of potentiated synapses in each update step (see “Plasticity dynamics” in [Network model](#)).

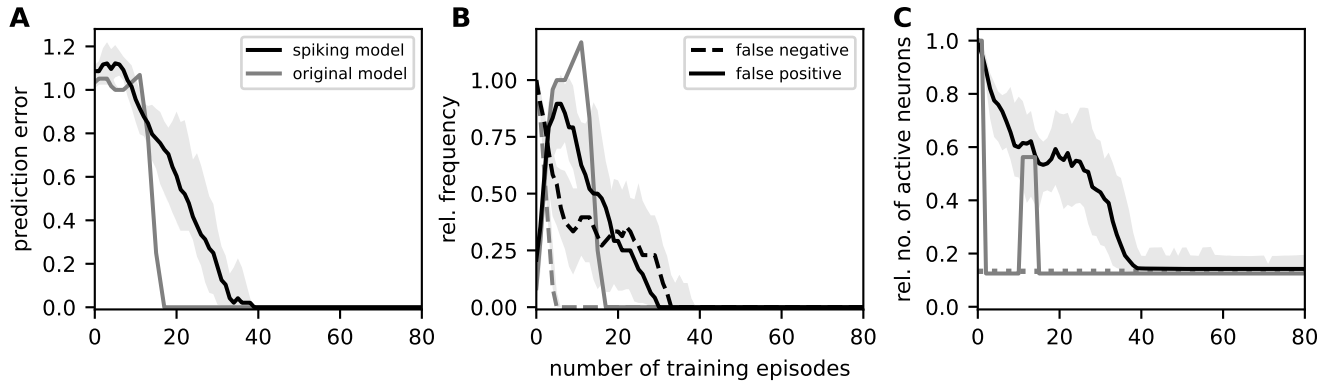
## Dependence of prediction performance on the sequence speed

The reformulation of the original TM model in terms of continuous-time dynamics allows us to ask questions related to timing aspects. Here, we investigate the sequence processing speed by identifying the range of inter-stimulus intervals  $\Delta T$  that permit a successful prediction performance (Fig. 10). The timing of the external inputs affects the dynamics of the network in two respects. First, reliable predictions of sequence elements can only be made if the time interval  $\Delta T$  between two consecutive stimulus presentations is such that the second input coincides with the somatic depolarization caused by the dAP triggered by the first stimulus. Second, the formation of sequence specific connections by means of the spike-timing-dependent structural plasticity dynamics depends on  $\Delta T$ .

If the external input does not coincide with the somatic dAP depolarization, i.e., if  $\Delta T$  is too small or too large, the respective target population responds in a non-sparse, non-selective manner (mismatch signal; Fig. 10C), and in turn, generates false positives (Fig. 10B). For small  $\Delta T$ , the external stimulus arrives before the dAP onset, i.e.,



**Fig 8: Sequence prediction performance for sequence set I.** Dependence of the sequence prediction error (A), the false-positive and false-negative rates (B), and the relative number of active neurons (C) on the number of training episodes during repetitive stimulation with sequence set I (see [Task and training protocol](#)). Curves and error bands indicate the median as well as the 5% and 95% percentiles across an ensemble of 5 different network realizations, respectively. All prediction performance measures are calculated as a moving average over the last 4 training episodes. The dashed gray horizontal line in panel C depicts the target sparsity level  $\rho/(Ln_E)$ . Inter-stimulus interval  $\Delta T = 40$  ms. See Table 2 for remaining parameters.



**Fig 9: Sequence prediction performance for sequence set II and comparison with original model.** Same figure arrangement, training and measurement protocol as in Fig. 8. Data obtained during repetitive stimulation of the network with sequence set II (see [Task and training protocol](#)). Gray curves depict results obtained using the original (non-spiking) TM model from [14] with adapted parameters (see Table S1). The dashed gray horizontal line in panel C depicts the target sparsity level  $\rho/(Ln_E)$ .

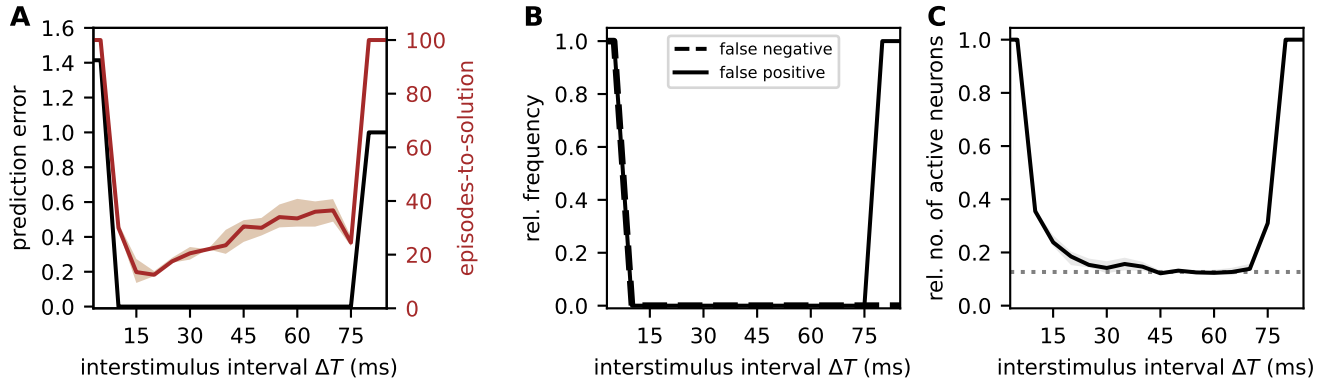
before it is predicted. In consequence, the false negative rate is high. For large  $\Delta T$ , the false negative rate remains low as the network is still generating predictions (Fig. 10B). The inter-stimulus interval  $\Delta T$  in addition affects the formation of sequence specific connections due to the dependence of the plasticity dynamics on the timing of pre- and postsynaptic spikes, see Eq (1) and Eq (2). Larger  $\Delta T$  result in smaller permanence increments, and thereby a slow-down of the learning process (red curve in Fig. 10A).

Taken together, the model predicts a range of optimal inter-stimulus interval  $\Delta T$  (Fig. 10A). For our choice of network parameters, this range spans intervals between 10 ms and 75 ms. The lower bound depends primarily on the synaptic time constant  $\tau_{EE}$ , the spike transmission delay  $d_{EE}$ , and the membrane time constant  $\tau_m$ . The upper bound is mainly determined by the dAP plateau duration  $\tau_{dAP}$ .

## Sequence replay

So far, we studied the network in the predictive mode, where the network is driven by external inputs and generates predictions of upcoming sequence elements. Another essential component of sequence processing is sequence replay, i.e., the autonomous generation of sequences in response to a cue signal (see [Task and training protocol](#)). After successful learning, the network model presented in this study is easily configured into the replay mode by increasing the neuronal excitability, such that the somatic depolarization caused by a dAP alone makes the neuron fire a somatic spike. Here, this is implemented by lowering the somatic spike threshold  $\theta_E$  of the excitatory neurons. In the biological system, this increase in excitability could, for example, be caused by the effect of neuromodulators [39, 40], additional excitatory inputs from other brain regions implementing a top-down control, e.g., attention [41, 42], or propagating waves during sleep [43, 44].

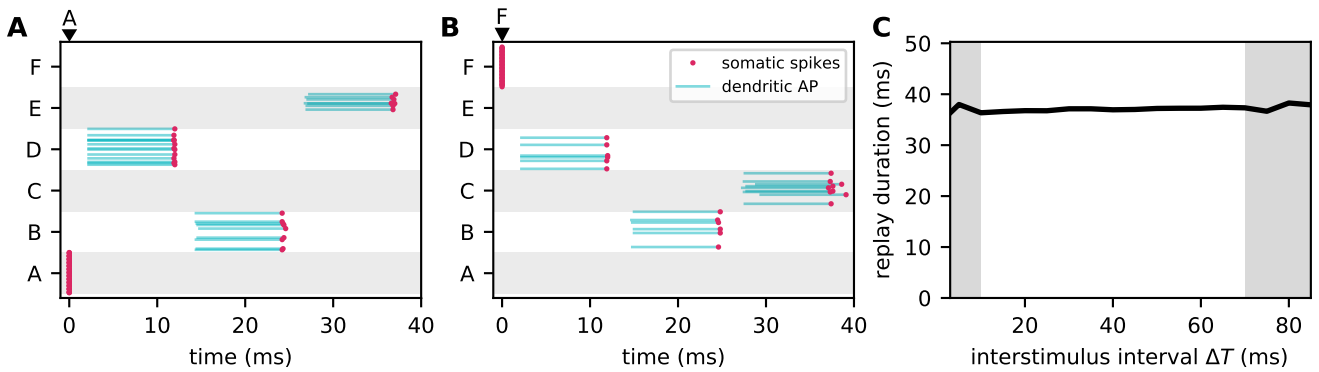
The presentation of the first sequence element activates dAPs in the subpopulation corresponding to the expected next element in a previously learned sequence. Due to the reduced firing threshold in the replay mode, the somatic



**Fig 10: Effect of sequence speed on network performance.** Dependence of the sequence prediction error, the learning speed (episodes-to-solution; **A**), the false-positive and false-negative rates (**B**), and the relative number of active neurons (**C**) on the inter-stimulus interval  $\Delta T$  after 150 training episodes. Curves and error bands indicate the median as well as the 5% and 95% percentiles across an ensemble of 5 different network realizations, respectively. Same task and network as in Fig. 8.

depolarization caused by these dAPs is sufficient to trigger somatic spikes during the rising phase of this depolarization. These spikes, in turn, activate the subsequent element. This process repeats, such that the network autonomously reactivates all sequence elements in the correct order, with the same context specificity and sparsity level as in the predictive mode (see Fig. 11A,B). The latency between the activation of subsequent sequence elements is determined by the spike transmission delay  $d_{EE}$ , the synaptic time constant  $\tau_{EE}$ , the membrane time constant  $\tau_{m,E}$ , the synaptic weights  $J_{EE,ij}$ , the dAP current plateau amplitude  $I_{dAP}$ , and the somatic firing threshold  $\theta_E$ . For sequences that can be successfully learned (see previous section), the time required for replaying the entire sequence is independent of the inter-stimulus interval  $\Delta T$  employed during learning (Fig. 11C).

As shown in the previous section, sequences cannot be learned if the inter-stimulus interval  $\Delta T$  is too small or too large. For small  $\Delta T$ , connections between subpopulations corresponding to subsequent elements are strongly potentiated by the Hebbian plasticity due to the consistent firing of pre- and postsynaptic populations during the learning process. The network responses are, however, non-sparse, as the winner-take-all mechanism cannot be invoked during the learning (Fig. 10C). In the replay mode, sequences are therefore replayed in a non-sparse and non-context specific manner (left gray region in Fig. 11C). Similarly, connections between subsequent populations are slowly potentiated for very large  $\Delta T$ . With sufficiently long learning, sequences can still be replayed in the right order, but the activity is non-sparse and therefore not context specific (right gray region in Fig. 11C).



**Fig 11: Sequence replay dynamics and speed.** Autonomous replay of the sequences {A,D,B,E} (**A**) and {F,D,B,C} (**B**), initiated by stimulating the subpopulations "A" and "F", respectively. Red dots and blue lines mark somatic spikes and dAP plateaus, respectively, for a fraction of neurons (30%) within each subpopulation. During learning, the inter-stimulus interval  $\Delta T$  is set to 40 ms. **C**) Dependence of the sequence replay duration on the inter-stimulus interval  $\Delta T$  during learning. Replay duration is measured as the difference between the mean firing times of the populations representing the first and last elements in a given sequence. Gray areas mark regions with low prediction performance (see [Dependence of prediction performance on the sequence speed](#)). Error bands represent the mean  $\pm$  standard deviation of the prediction error across 5 different network realizations. Same network and training set as in Fig. 8.



## Discussion

### Summary

In this work, we reformulate the Temporal Memory (TM) model [14] in terms of biophysical principles and parameters. We replace the original discrete-time neuronal and synaptic dynamics with continuous-time models with biologically interpretable parameters such as membrane and synaptic time constants and synaptic weights. We further substitute the original plasticity algorithm with a more biologically plausible mechanism, relying on a form of Hebbian structural plasticity, homeostatic control, and sparse random connectivity. Moreover, our model implements a winner-take-all dynamics based on lateral inhibition that is compatible with the continuous-time neuron and synapse models. We show that the revised TM model supports successful learning and processing of high-order sequences with a performance similar to the one of the original model [14].

A new aspect that we investigated in the context of our work is sequence replay. After learning, the model is able to replay sequences in response to a cue signal. The duration of sequence replay is independent of the sequence speed during training, and determined by the intrinsic parameters of the network. In general, sequence replay is faster than the sequence presentation during learning, consistent with sequence compression and fast replay observed in hippocampus [45–47] and neocortex [6, 48].

Finally, we identified the range of possible sequence speeds that guarantee a successful learning and prediction. Our model predicts an optimal range of processing speeds (inter-stimulus intervals) with lower and upper bounds constrained by neuronal and synaptic parameters (e.g., firing threshold, neuronal and synaptic time constants, coupling strengths, potentiation time constants). Within this range, the number of required training episodes is proportional to the inter-stimulus interval  $\Delta T$ .

### Relationship to other models

The model presented in this work constitutes a recurrent, randomly connected network of neurons with predefined stimulus preferences. The model learns sequences in an unsupervised manner using local learning rules. This is in essence similar to several other spiking neuronal network models for sequence learning [9–12]. The new components employed in this work are dendritic action potentials (dAPs) and Hebbian structural plasticity. We use structural plasticity to be as close as possible to the original model, and Hebbian forms of this are also known from the literature [21, 22, 25]. However, preliminary results show that classical (non-structural) spike-timing-dependent plasticity (STDP) can yield similar performance (see Fig. S1). Dendritic action potentials are instrumental for our model for two reasons. First, they effectively lower the threshold for coincidence detection and thereby permit a reliable and robust propagation of sparse activity [49, 50]. In essence, our model bears similarities to the classical synfire chain [51], one difference being that our "mature" network is not a simple feed-forward network but has an abundance of recurrent connections. As shown in [52], a stable propagation of synchronous activity requires a minimal number of neurons in each synfire group. Without active dendrites, this minimal number is in the range of  $\sim 100$  for plausible single-cell and synaptic parameters. In our (and in the original TM) model, coincidence detection happens in the dendrites. The number of presynaptic spikes needed to trigger a dAP is small, of the order of  $\sim 10$  [53–55]. This helps to reduce redundancy (only a small number of neurons needs to become active) and to increase the capacity of the network (the number of different patterns that can be learned is increased with pattern sparsity; [56]). Second, dAPs equip neurons with a third type of state (next to the quiescent and the firing state): the predictive state, i.e., a long lasting ( $\sim 50$ – $200$  ms) strong depolarization of the soma. Due to the prolonged depolarization of the soma, the inter-stimulus interval can be much larger than the synaptic time constants and delays. An additional benefit of dAPs, which is not exploited in the current version of our model, is that they equip individual neurons with more possible states if they comprise more than one dendritic branch. Each branch constitutes an independent pattern detector. The response of the soma may depend on the collective predictions in different dendritic branches. A single neuron could hence perform the types of computations that are usually assigned to multilayer perceptrons, i.e., small networks [57, 58].

Most of the existing models have been developed to replay learned sequences in response to a cue signal. The TM model can perform this type of pattern completion, too. In addition, it can act as a quiet, sparsely active observer of the world that becomes highly active only in the case of unforeseen, non-anticipated events. In this work, we didn't directly analyze the network's mismatch detection performance. However, this could be easily achieved by equipping each population with a "mismatch" neuron that fires if a certain fraction of neurons in the population fires (threshold detectors). Moreover, the TM model presents an efficient way of learning and encoding the context in high-order sequences, without prior assignment of context specificity to individual neuron populations [9], and without additional network components (such as reservoir networks in [11]).

An earlier spiking neural network version of the HTM model has already been devised in [59]. It constitutes a proof-of-concept study demonstrating that the HTM model can be ported to an analog-digital neuromorphic hardware system. It is restricted to small simplistic sequences and does not address the biological plausibility of the TM model. In particular, it does not offer a solution to the question of how the model can perform online learning by known biological ingredients. Our study delivers a solution for this based on local plasticity rules and permits a direct

implementation on a neuromorphic hardware system.

## Limitations and outlook

Our results on the sequence processing speed revealed that the model presented here can process fast sequences with inter-stimulus intervals  $\Delta T$  up to  $\sim 75$  ms. This range of processing speeds is relevant in many behavioral contexts such as motor generation, vision (saccades), music perception and generation, language, and many others [60]. However, slow sequences with inter-stimulus intervals beyond several hundreds of milliseconds cannot be learned by this model with biologically plausible parameters. This is problematic as behavioral time scales are often larger [60, 61]. By increasing the duration  $\tau_{\text{dAP}}$  of the dAP plateau, the upper bound for  $\Delta T$  could be extended to 500 ms, and maybe beyond [62]. However, for such long intervals, the synaptic potentiation would be very slow, unless the time constant  $\tau_+$  of the structural STDP is increased and the depression rate  $\lambda_-$  is adapted accordingly. Furthermore, while our model explains the fast replay observed in the hippocampus and cortex, it is not able to learn an element specific timing and duration of sequence elements [5, 63, 64]. This could be overcome by equipping the model with a working memory mechanism, which maintains the activity of the subpopulations for behaviorally relevant time scales [9, 65].

The network model underlying our study is still simplistic. The number of subpopulations, the number of neurons within each subpopulation, the number of dendritic branches per neuron, as well as the number of synapses per neuron are far from realistic [14]. The number of sequences that can be successfully learned in this network is hence rather small. Here, the model serves as a proof of concept demonstrating that the TM algorithm proposed in [14] can be implemented using biological ingredient. In addition, the current work is focusing on sequence processing at a single abstraction level, not accounting for a hierarchical network and task structure with both bottom-up and top-down projections. A further simplification in this work is that the lateral inhibition within a subpopulation is mediated by a single interneuron with unrealistically strong and fast connections to and from the pool of excitatory neurons. In future versions of this model, this interneuron could be replaced by a recurrently connected network of inhibitory neurons, thereby permitting more realistic weights, and simultaneously speeding up the interaction between inhibitory and excitatory cells by virtue of the fast-tracking property of such networks [66].

Similar to the original TM model, the response of the population representing the first element in a sequence is non-sparse, indicating that the first sequence element is not anticipated and can therefore not be predicted. If a given first sequence element reoccurs within the same sequence (say, "A" in {A,B,A,C}) or in other sequences (e.g., in {D,E,A,F}), the non-sparse response of the respective population to a first sequence element leads to a simultaneous prediction of all possible subsequent elements, i.e., the generation of false positives. These false predictions would lead to a pruning of functional synapses as a response of the homeostatic regulation to the increased dAP activity. This could be overcome by three possible mechanisms: a) synaptic normalization avoiding excessive synapse growth [67, 68], b) removing breaks between sequences, or c) sparse, sequence specific firing of subpopulations representing first elements. Results of applying the last mechanism are shown in Fig. S2, where dAPs are externally activated in random subsets of neurons in the populations representing first elements. In a more realistic hierarchical network, a similar effect could be achieved by top-down projections from a higher level predicting sequences of sequences.

In the original model, synapses targeting silent postsynaptic cells are depressed, even if the presynaptic neuron is inactive. This pruning process, the freeing of unused synaptic resources, increases the network capacity while ensuring context sensitivity. According to the structural plasticity dynamics employed in our study, synapse depression is bound to presynaptic spiking, similar to other implementations of (non-structural) STDP [30]. As a consequence, strong connections originating from silent presynaptic neurons are not depressed (dark gray dots in Fig. 2D). This may complicate or slow down the learning of new sequences, and could be overcome by synaptic normalization.

For the dAP-rate homeostasis used in this study, the target dAP rate is set to one to make sure that each neuron contributes at most one dAP during each training episode. As a consequence, the time constant of the dAP-rate homeostasis is adapted to the duration of a training episode, which is in the range of few seconds in this work. We are not aware of any biological mechanism that could account for such an adaptation. dAP-rate homeostasis is mediated by the intracellular calcium concentration, which, in turn, controls the synthesis of synaptic receptors, and hence, the synaptic strength. It is therefore known to be rather slow, acting on timescales of many minutes, hours or days [32, 33]. It is unclear to what extent the use of long homeostatic time constants and increased dAP target rates would alter the model performance. Alternatively, the dAP-rate homeostasis could be replaced by other mechanisms such as synaptic normalization.

## Conclusion

Our work demonstrates that the principle mechanisms underlying sequence learning, prediction, and replay in the TM model can be implemented using biologically plausible ingredients. By strengthening the link to biology, our implementation permits a more direct evaluation of the TM model predictions based on electrophysiological and

behavioral data. Furthermore, this implementation allows for a direct mapping of the TM model on neuromorphic hardware systems.

## Acknowledgments

This project was funded by the Helmholtz Association Initiative and Networking Fund (project number SO-092, Advanced Computing Architectures), and the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 785907 (HBP SGA2) and No. 945539 (Human Brain Project SGA3). The authors thank Charl Linssen for help with the neuron model implementation using NESTML, Abigail Morrison for suggestions on the plasticity dynamics, Danylo Ulianych and Dennis Terhorst for code review, and Sebastian B.C. Lehmann for assistance with graphic design. All network simulations were carried out with NEST (<http://www.nest-simulator.org>).

## Author contributions

YB, DJW and TT conceived and designed the work. YB performed the simulations, and analyzed and visualized the data. All authors jointly wrote the paper, reviewed the manuscript and approved it for publication. YB was supervised by TT and DJW.

## References

1. Lashley KS. The problem of serial order in behavior. vol. 21. Bobbs-Merrill; 1951.
2. Hawkins J, Blakeslee S. On intelligence: How a new understanding of the brain will lead to the creation of truly intelligent machines. Macmillan; 2007.
3. Dehaene S, Meyniel F, Wacongne C, Wang L, Pallier C. The neural representation of sequences: from transition probabilities to algebraic patterns and linguistic trees. *Neuron*. 2015;88(1):2–19.
4. Clegg BA, DiGirolamo GJ, Keele SW. Sequence learning. *Trends in cognitive sciences*. 1998;2(8):275–281.
5. Gavnornik JP, Bear MF. Learned spatiotemporal sequence recognition and prediction in primary visual cortex. *Nature neuroscience*. 2014;17(5):732.
6. Xu S, Jiang W, Poo MM, Dan Y. Activity recall in a visual cortical ensemble. *Nature Neuroscience*. 2012 mar;15(3):449–455.
7. Hahnloser RH, Kozhevnikov AA, Fee MS. An ultra-sparse code underlies the generation of neural sequences in a songbird. *Nature*. 2002 Sep;419(6902):65–70.
8. Harvey CD, Coen P, Tank DW. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*. 2012;484(7392):62–68.
9. Maes A, Barahona M, Clopath C. Learning spatiotemporal signals using a recurrent spiking network that discretizes time. *PLoS computational biology*. 2020;16(1):e1007606.
10. Klos C, Miner D, Triesch J. Bridging structure and function: A model of sequence learning and prediction in primary visual cortex. *PLoS computational biology*. 2018;14(6):e1006187.
11. Cone I, Shouval HZ. Learning precise spatiotemporal sequences via biophysically realistic learning rules in a modular, spiking network. *Elife*. 2021;10:e63751.
12. Klampfl S, Maass W. Emergence of dynamic memory traces in cortical microcircuit models through STDP. *Journal of Neuroscience*. 2013;33(28):11515–11529.
13. Hawkins J, Ahmad S, Dubinsky D. Cortical Learning Algorithm and Hierarchical Temporal Memory. Numenta Whitepaper. 2011;p. 1–68. Available from: [http://numenta.org/resources/HTM\\_CorticalLearningAlgorithms.pdf](http://numenta.org/resources/HTM_CorticalLearningAlgorithms.pdf).
14. Hawkins J, Ahmad S. Why neurons have thousands of synapses, a theory of sequence memory in neocortex. *Frontiers in neural circuits*. 2016;10:23.
15. Cui Y, Ahmad S, Hawkins J. Continuous online sequence learning with an unsupervised neural network model. *Neural computation*. 2016;28(11):2474–2504.

16. Ahmad S, Hawkins J. Properties of sparse distributed representations and their application to hierarchical temporal memory. arXiv preprint arXiv:150307469. 2015;.
17. Stepanyants A, Hirsch JA, Martinez LM, Kisvárdy ZF, Ferecskó AS, Chklovskii DB. Local Potential Connectivity in Cat Primary Visual Cortex. *Cerebral Cortex*. 2007 April;18(1):13–28. Available from: <https://doi.org/10.1093/cercor/bhm027>.
18. Antic SD, Zhou WL, Moore AR, Short SM, Ikonomu KD. The decade of the dendritic NMDA spike. *Journal of neuroscience research*. 2010;88(14):2991–3001.
19. Schiller J, Major G, Koester HJ, Schiller Y. NMDA spikes in basal dendrites of cortical pyramidal neurons. *nat*. 2000 Mar;404(6775):285–289. Available from: <http://dx.doi.org/10.1038/35005094>.
20. Larkum ME, Nevian T, Sandler M, Polsky A, Schiller J. Synaptic Integration in Tuft Dendrites of Layer 5 Pyramidal Neurons: A New Unifying Principle. *Science*. 2009 August;325(5941):756–760. Available from: <https://doi.org/10.1126/science.1171958>.
21. Liao D, Hessler NA, Malinow R. Activation of postsynaptically silent synapses during pairing-induced LTP in CA1 region of hippocampal slice. *Nature*. 1995;375(6530):400–404.
22. Wu GY, Malinow R, Cline HT. Maturation of a Central Glutamatergic Synapse. *Science*. 1996 November;274(5289):972–976. Available from: <https://doi.org/10.1126/science.274.5289.972>.
23. Lüscher C, Noicoll RA, Malenka RC, Muller D. Synaptic plasticity and dynamic modulation of the postsynaptic membrane. *Nat Neurosci*. 2000 June;3(6):545–550.
24. Nevian T, Sakmann B. Single Spine  $\text{Ca}^{2+}$  Signals Evoked by Coincident EPSPs and Backpropagating Action Potentials in Spiny Stellate Cells of Layer 4 in the Juvenile Rat Somatosensory Barrel Cortex. *J Neurosci*. 2004;24(7):1689–1699.
25. Deger M, Helias M, Rotter S, Diesmann M. Spike-Timing Dependence of Structural Plasticity Explains Cooperative Synapse Formation in the Neocortex. *PLoS Comput Biol*. 2012;8(9):e1002689.
26. Müllner FE, Wierenga CJ, Bonhoeffer T. Precision of inhibition: dendritic inhibition by individual GABAergic synapses on hippocampal pyramidal cells is confined in space and time. *Neuron*. 2015;87(3):576–589.
27. Morrison A, Diesmann M, Gerstner W. Phenomenological models of synaptic plasticity based on spike-timing. *Biol Cybern*. 2008;98(6):459–478. Available from: <https://doi.org/10.1007/s00422-008-0233-1>.
28. Abbott LF, Nelson SB. Synaptic plasticity: taming the beast. *Nat Neurosci*. 2000 November;3:1178–1183. Available from: <https://doi.org/10.1038/81453>.
29. Tetzlaff C, Kolodziejcki C, Timme M, Wörgötter F. Synaptic scaling in combination with many generic plasticity mechanisms stabilizes circuit connectivity. *Frontiers in computational neuroscience*. 2011;5:47.
30. Morrison A, Aertsen A, Diesmann M. Spike-Timing Dependent Plasticity in Balanced Random Networks. *Neural Comput*. 2007 June;19:1437–1467. Available from: <https://doi.org/10.1162/neco.2007.19.6.1437>.
31. Turrigiano GG, Leslie KR, Desai NS, Rutherford LC, Nelson SB. Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*. 1998;391:892–896.
32. Turrigiano GG, Nelson SB. Homeostatic plasticity in the developing nervous system. *natrevnsi*. 2004;5:97–107.
33. Turrigiano GG. The self-tuning neuron: synaptic scaling of excitatory synapses. *Cell*. 2008 October;135(3):422–435.
34. Gewaltig MO, Diesmann M. NEST (NEural Simulation Tool). *Scholarpedia*. 2007;2(4):1430. Available from: <https://doi.org/10.4249/scholarpedia.1430>.
35. Hahne J, Diaz S, Patronis A, Schenck W, Peyser A, Graber S, et al.. NEST 3.0. Zenodo; 2021. Available from: <https://doi.org/10.5281/zenodo.4739103>.
36. Plotnikov D, Blundell I, Ippen T, Eppler JM, Rumpel B, Morrison A. NESTML: a modeling language for spiking neurons. In: Oberweis A, Reussner R, editors. *Modellierung 2016*. vol. P-254 of Lecture Notes in Informatics (LNI). Modellierung 2016, Karlsruhe (Germany), 17 Mar 2016 - 19 Mar 2016. Gesellschaft für Informatik e.V. (GI); 2016. p. 93–108. Available from: <http://juser.fz-juelich.de/record/826510>.

37. Nagendra Babu P, Linssen C, Eppler JM, Schulte to Brinke T, Ziaemehr A, Fardet T, et al.. NESTML 4.0. Zenodo; 2021. Available from: <https://doi.org/10.5281/zenodo.4740083>.
38. Rotter S, Diesmann M. Exact digital simulation of time-invariant linear systems with applications to neuronal modeling. *Biol Cybern.* 1999;81(5-6):381–402. Available from: <https://doi.org/10.1007/s004220050570>.
39. Atherton LA, Dupret D, Mellor JR. Memory trace replay: the shaping of memory consolidation by neuromodulation. *Trends Neurosci.* 2015 September;38(9):560–570. Available from: <https://doi.org/10.1016/j.tins.2015.07.004>.
40. Thomas SA. Neuromodulatory signaling in hippocampus-dependent memory retrieval. *Hippocampus.* 2015;25(4):415–431.
41. Baluch F, Itti L. Mechanisms of top-down attention. *Trends Neurosci.* 2011 April;34(4):210–224. Available from: <https://doi.org/10.1016/j.tins.2011.02.003>.
42. Noudoost B, Chang MH, Steinmetz NA, Moore T. Top-down control of visual attention. *Current Opinion in Neurobiology.* 2010 April;20(2):183–190. Available from: <https://doi.org/10.1016/j.conb.2010.02.003>.
43. Buzsáki G. *Rhythms of the Brain.* Oxford University Press; 2006. Available from: <https://doi.org/10.1093/acprof:oso/9780195301069.001.0001>.
44. Grosmark AD, Mizuseki K, Pastalkova E, Diba K, Buzsáki G. REM Sleep Reorganizes Hippocampal Excitability. *Neuron.* 2012 September;75(6):1001–1007. Available from: <https://doi.org/10.1016/j.neuron.2012.08.015>.
45. Nádasdy Z, Hirase H, Czurkó A, Csicsvari J, Buzsáki G. Replay and Time Compression of Recurring Spike Sequences in the Hippocampus. *The Journal of Neuroscience.* 1999 November;19(21):9497–9507. Available from: <https://doi.org/10.1523/jneurosci.19-21-09497.1999>.
46. Lee AK, Wilson MA. Memory of Sequential Experience in the Hippocampus during Slow Wave Sleep. *Neuron.* 2002 December;36(6):1183–1194. Available from: [https://doi.org/10.1016/s0896-6273\(02\)01096-6](https://doi.org/10.1016/s0896-6273(02)01096-6).
47. Davidson TJ, Kloosterman F, Wilson MA. Hippocampal replay of extended experience. *Neuron.* 2009;63(4):497–507.
48. Euston DR, Tatsuno M, McNaughton BL. Fast-Forward Playback of Recent Memory Sequences in Prefrontal Cortex During Sleep. *Science.* 2007 November;318(5853):1147–1150. Available from: <https://doi.org/10.1126/science.1148979>.
49. Jahnke S, Timme M, Memmesheimer RM. Guiding Synchrony through Random Networks. *Phys Rev X.* 2012;2(4):041016.
50. Breuer D, Timme M, Memmesheimer RM. Statistical physics of neural systems with nonadditive dendritic coupling. *Phys Rev X.* 2014;4(1):011053.
51. Abeles M. *Corticonics: Neural Circuits of the Cerebral Cortex.* Cambridge: Cambridge University Press; 1991.
52. Diesmann M, Gewaltig MO, Aertsen A. Stable propagation of synchronous spiking in cortical neural networks. *Nature.* 1999;402(6761):529–533.
53. Major G, Larkum M, Schiller J. Active Properties of Neocortical Pyramidal Neuron Dendrites. *Annu Rev Neurosci.* 2013;36:1–24.
54. Mengual UM, Wybo WA, Spierenburg LJ, Santello M, Senn W, Nevian T. Efficient low-pass dendro-somatic coupling in the apical dendrite of layer 5 pyramidal neurons in the anterior cingulate cortex. *Journal of neuroscience.* 2020;40(46):8799–8815.
55. Diesmann M. Conditions for Stable Propagation of Synchronous Spiking in Cortical Neural Networks: Single Neuron Dynamics and Network Properties; 2002. Available from: <https://hss-opus.ub.rub.de/opus4/frontdoor/index/index/docId/3292>.
56. Ahmad S, Hawkins J. How do neurons operate on sparse distributed representations? A mathematical theory of sparsity, neurons and active dendrites. *arXiv preprint arXiv:160100720.* 2016;.

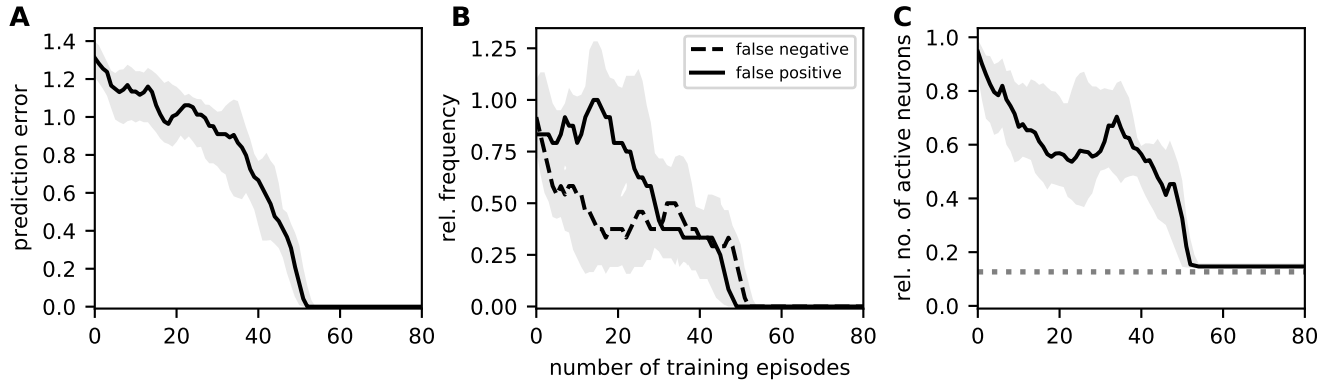
57. Gidon A, Zolnik TA, Fidzinski P, Bolduan F, Papoutsi A, Poirazi P, et al. Dendritic action potentials and computation in human layer 2/3 cortical neurons. *Science*. 2020 January;367(6473):83–87. Available from: <https://doi.org/10.1126/science.aax6239>.
58. Poirazi P, Brannon T, Mel BW. Pyramidal Neuron as Two-Layer Neural Network. *Neuron*. 2003 March;37(6):989–999. Available from: [https://doi.org/10.1016/s0896-6273\(03\)00149-1](https://doi.org/10.1016/s0896-6273(03)00149-1).
59. Billaudelle S, Ahmad S. Porting HTM models to the Heidelberg neuromorphic computing platform. *arXiv preprint arXiv:150502142*. 2015;.
60. Mauk MD, Buonomano DV. The Neural Basis of Temporal Processing. *Annu Rev Neurosci*. 2004;27:307–340.
61. Paton JJ, Buonomano DV. The neural basis of timing: Distributed mechanisms for diverse functions. *Neuron*. 2018;98(4):687–705.
62. Milojkovic BA. A Strict Correlation between Dendritic and Somatic Plateau Depolarizations in the Rat Prefrontal Cortex Pyramidal Neurons. *Journal of Neuroscience*. 2005 April;25(15):3940–3951. Available from: <https://doi.org/10.1523/jneurosci.5314-04.2005>.
63. Dave AS, Margoliash D. Song replay during sleep and computational rules for sensorimotor vocal learning. *Science*. 2000;290(5492):812–816.
64. Louie K, Wilson MA. Temporally structured replay of awake hippocampal ensemble activity during rapid eye movement sleep. *Neuron*. 2001;29(1):145–156.
65. Tully PJ, Lindén H, Hennig MH, Lansner A. Spike-Based Bayesian-Hebbian Learning of Temporal Sequences. *PLOS Computational Biology*. 2016 May;12(5):e1004954. Available from: <https://doi.org/10.1371/journal.pcbi.1004954>.
66. van Vreeswijk C, Sompolinsky H. Chaotic Balanced State in a Model of Cortical Circuits. *Neural computation*. 1998 August;10(6):1321–1371. Available from: <https://doi.org/10.1162/089976698300017214>.
67. Turrigiano GG, Leslie KR, Desai NS, Rutherford LC, Nelson SB. Activity-dependent scaling of quantal amplitude in neocortical neurons. *Nature*. 1998 February;391(6670):892–896. Available from: <https://doi.org/10.1038/36103>.
68. Elliott T. An Analysis of Synaptic Normalization in a General Class of Hebbian Models. *Neural Computation*. 2003 April;15(4):937–963. Available from: <https://doi.org/10.1162/08997660360581967>.



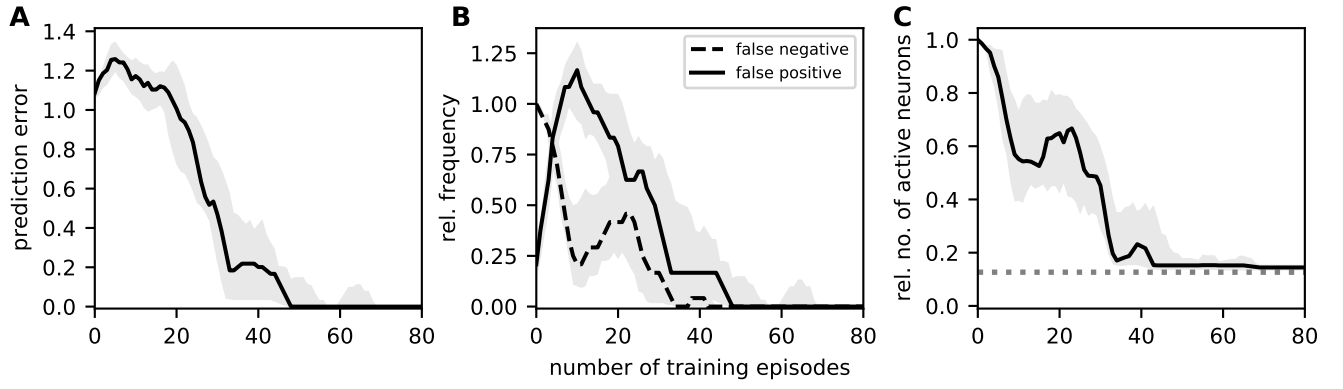
## Supplementary Materials

Name	Value	Description
columnDimensions ( $M$ )	280	number of columns
numColumnsPerElement ( $L$ )	20	number of columns per element
cellsPerColumn ( $n_E$ )	8	number of cells per column
initialPermanence ( $P_0$ )	[0.1, 0.3]	initial permanence
connectedPermanence ( $\theta_P$ )	0.5	threshold at which a synapses is considered connected
minThreshold	15	if the number of potential synapses active on a segment is at least this threshold, it is said to be “matching” and is eligible for learning.
maxNewSynapseCount	40	the maximum number of synapses added to a segment during learning
permanenceIncrement ( $\lambda_+$ , $\lambda_h$ )	0.1	amount by which permanences of synapses are incremented during learning.
permanenceDecrement ( $\lambda_-$ )	0.3	amount by which permanences of synapses are decremented during learning.
activationThreshold ( $\theta_{dAP}$ )	15	if the number of active connected synapses on a segment is at least this threshold, the segment is said to be active.
predictedSegmentDecrement	0.01	amount by which permanences of synapses are decremented during learning.

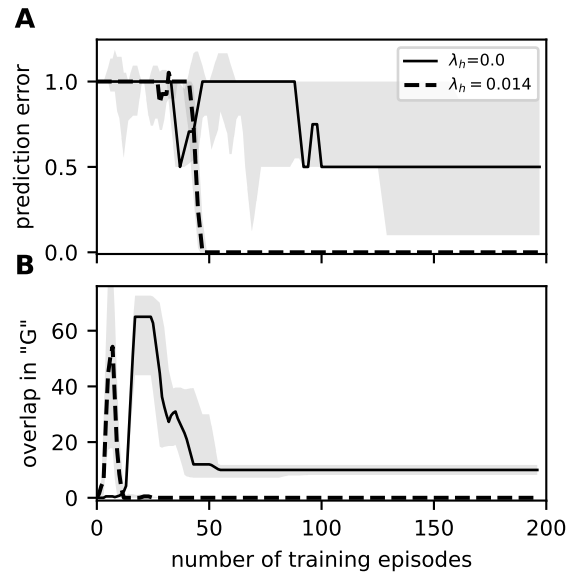
**Table S1: Adapted parameters of the original TM model used for Fig.9.** Parameter names match those used in the original simulation code ([https://github.com/numenta/htmpapers/tree/master/frontiers/why\\_neurons\\_have\\_thousands\\_of\\_synapses](https://github.com/numenta/htmpapers/tree/master/frontiers/why_neurons_have_thousands_of_synapses)). Gray parameter names are those used in the spiking TM model.



**Fig S1: Sequence prediction performance in the presence of conventional (non-structural) spike-timing dependent plasticity (STDP).** Dependence of the sequence prediction error (A), the false-positive and false-negative rates (B), and the relative number of active neurons (C) on the number of training episodes for sequence set II. Curves and error bands indicate the median as well as the 5% and 95% percentiles across an ensemble of 5 different network realizations, respectively. All prediction performance measures are calculated as a moving average over the last 4 training episodes. In this experiment, structural STDP is replaced by conventional STDP, i.e., the permanences  $P_{ij}(t)$  and  $P_{\max}$  in Eq (1) are replaced by the synaptic weights  $J_{EE,ij}(t)$  and  $J_{\max}$ . The weights  $J_{EE,ij}$  are restricted to the interval  $[J_{\min,ij}, J_{\max}]$ , and clipped at the boundaries. The minimal weights  $J_{\min,ij}$  are randomly and independently drawn from a uniform distribution between  $J_{0,\min}$  and  $J_{0,\max}$ . The performance characteristics shown here are comparable to those obtained with structural STDP (see Fig. 9 in [Prediction performance](#)). Parameters:  $\Delta T = 40$  ms,  $\lambda_+ = 0.43$ ,  $\lambda_- = 0.006$ ,  $\lambda_h = 0.03$ ,  $J_{0,\min} = 0$ ,  $J_{0,\max} = 1$ ,  $J_{\max} = 12$ . See Table 2 for remaining parameters.



**Fig S2: Prediction performance for a sequence set with recurring first items.** Dependence of the sequence prediction error (A), the false positive frequency, the false negative frequency (B), and the relative number of active neurons (C) on the number of training episodes for a set of sequences  $s_1 = \{B, D, I, C, H\}$ ,  $s_2 = \{E, D, I, C, F\}$ ,  $s_3 = \{F, B, C, A, H\}$ ,  $s_4 = \{G, B, C, A, D\}$ ,  $s_5 = \{E, C, I, H, A\}$ ,  $s_6 = \{D, C, I, H, G\}$ . with recurring first items. Curves and error bands indicate the median as well as the 5% and 95% percentiles across 5 different network realizations, respectively. As a solution to the issue discussed in [Limitations and outlook](#) concerning the recurring of first sequence elements in other sequences or within the same sequence, the dAPs are externally activated in a random subsets of neurons in the populations representing first elements. Inter-stimulus interval  $\Delta T = 40$  ms. All prediction performance measures are calculated as a moving average over the last 4 training episodes. Parameters:  $\Delta T = 40$  ms,  $\lambda_+ = 0.39$ ,  $\lambda_- = 0.0057$ ,  $\lambda_h = 0.034$ . See Table 2 for remaining parameters.



**Fig S3: Effect of the dAP-rate homeostasis on the prediction performance.** Dependence of the prediction error (**A**) and the overlap in the activation pattern between the neurons representing the sequence element "G" in the context of sequences {A,D,B,G,H,E} and {F,D,B,G,H,C} (**B**) on the number of training episodes explored for two values of the homeostasis rate ( $\lambda_h$ ). Curves and error bands indicate the median as well as the 5% and 95% percentiles across 5 different network realizations, respectively. Disabling the homeostasis control ( $\lambda_h = 0.0$ ) increases the overlap in the "G" activation pattern, which leads to a lost of context specificity and hence an increase in the prediction error (see [Sequence learning and prediction](#)). Parameters:  $\Delta T = 40$  ms,  $\lambda_+ = 0.08$ ,  $\lambda_- = 0.0015$ ,  $\lambda_h = 0.014$ . See Table 2 for remaining parameters.

---

**Algorithm S1** Algorithmic description of the plasticity model, based on the algorithm proposed in [30].

---

**Update of permanence  $P_{ij}$  and synaptic weight  $J_{EE,ij}$  at time  $t_j^k$  of the  $k$ th spike of presynaptic neuron  $j$ :**

```

 $x_j \leftarrow$  get trace of presynaptic neuron  $j$  ▷ last update at time  $t_j^{k-1}$ 
 $L_{\text{post}}, Z_{\text{post}} \leftarrow$  get lists of postsynaptic spike times and corresponding dAP traces in the interval  $(t_j^{k-1}, t_j^k]$ 

for  $t_i, z_i$  in  $L_{\text{post}}, Z_{\text{post}}$  do
  if  $\Delta t_{\min} < t_i - t_j^{k-1} < \Delta t_{\max}$  then
     $P_{ij} += x_j \cdot \lambda_+ \cdot P_{\max} \cdot \exp(-(t_i - t_j^{k-1} + d_{EE})/\tau_+)$  ▷ potentiation
     $P_{ij} += \lambda_h \cdot P_{\max} \cdot (z^* - z_i)$  ▷ homeostasis
  end if
end for

 $P_{ij} -= y_i \cdot \lambda_- \cdot P_{\max}$  ▷ depression

if  $P_{ij} > \theta_P$  then
   $J_{EE,ij} \leftarrow W$  ▷ mature synapse
else
   $J_{EE,ij} \leftarrow 0$  ▷ immature synapse
end if

 $x_j = x_j \cdot \exp((t_{k-1}^j - t_k^j)/\tau_+) + 1.0$  ▷ update of presynaptic spike trace

```

Note: the clipping of the permanence  $P_{ij}$  at the boundaries of the interval  $[P_{\min,ij}, P_{\max}]$  is not included here for clarity.

---

**Video S1: Time resolved visualization of the learning dynamics:** <https://fz-juelich.sciebo.de/s/CoptaUMS6ufRW3t>. Network activity (top) and connectivity (bottom) of the network during one learning episode. Each frame corresponds to a new training episode. In each learning episode, each of the two sequences  $\{A,D,B,E\}$  and  $\{F,D,B,C\}$  is presented once (black arrows in the top panel). **Top panel:** Red dots and blue bars mark spike and dAP times for each neuron. Neurons are sorted according to stimulus preference (vertical axis). **Bottom panel:** Network connectivity before learning (left) and during the current training episode (right). Light gray and black dots represent immature potential and mature connections, respectively, for each pair of source and target neurons (sorted according to stimulus preference; see [Sequence learning and prediction](#)).