Interleaved training prevents catastrophic forgetting in spiking neural networks

Ryan Golden[1,2,3], Jean Erik Delanois[2,3], Maxim Bazhenov[1,2]

[1] Neurosciences Graduate Program, [2] Department of Medicine, University of California, San Diego, CA 92093

[3] equal contribution

Abstract: 188 words; **Introduction:** 439 words; **Discussion:** 607 words

Pages: 19 (not including references); **Figures:** 7

Acknowledgements: This study was supported by Lifelong Learning Machines program from DARPA/MTO (HR0011-18-2-0021)

Conflicts of Interest: None.

To whom correspondence should be addressed:

Maxim Bazhenov
Department of Medicine
University of California, San Diego
La Jolla, CA 92093
Phone: 858-534-8391
Email: mbazhenov@ucsd.edu

Keywords: Spiking neural network model, catastrophic forgetting, interleaved training, rewarded spike-timing-dependent plasticity

Abstract (300 max)

Artificial neural networks suffer from the inability to learn new tasks sequentially without completely overwriting all memory of the previously learned tasks, a phenomenon known as catastrophic forgetting. However, biological neural networks are able to continuously learn many tasks over the course of the organism's lifetime, and typically learn the best when tasks are trained sequentially. Here we used a multi-layer spiking neural network with biologically plausible dynamics and learning rules to study synaptic mechanisms behind catastrophic forgetting and to discuss possible solutions to overcome the problem. The network utilized reward modulated and non-reward modulated STDP and implemented multiple mechanisms for homeostatic regulation of synaptic efficacy. We found that the network can learn to perform two complementary, complex foraging tasks, but underwent catastrophic forgetting upon sequential training due to a near complete overwriting of synaptic weights. Interleaved training was capable of mitigating catastrophic forgetting by partially preserving synaptic weights from the later task and recovering those from the earlier task. Moreover, interleaved training pushed the final synaptic weight configuration towards the decision boundary between the configurations associated with each task, striking an optimal balance between both task representations.

Author Summary (150 – 200 max)

The human brain is capable of learning an immense number of complex tasks over the course of a lifetime without forgetting previous tasks. Although modern artificial neural networks used in machine learning are getting closer to matching human-level capabilities regarding the complexity of the task to be learned, these networks typically lose all ability to perform this task after being trained sequentially on a second task; a phenomenon known as catastrophic forgetting. To better understand this phenomenon and how the brain overcomes it, we developed a spiking neural network which directly models the functioning of biological neurons and is updated according to biologically plausible learning rules. This network was able to learn to perform complementary complex foraging tasks but suffered from catastrophic forgetting when they were trained

sequentially. Catastrophic forgetting was prevented if the new task was trained by interleaving it with trials from the original task. Significantly, we found that interleaved training accomplishes this by representing the two tasks as a single hybrid-task. This indicates that rather than learn to represent each task independently and appropriately toggle between these, the network compromised and found an optimal hybrid representation of synaptic weights which was sufficient to perform both tasks.

Introduction

Humans are capable of continuously learning to perform novel tasks throughout life without interfering with their ability to perform previous tasks. Conversely, while modern artificial neural networks (ANNs) are capable of learning to perform complicated tasks, ANNs have difficulty learning multiple tasks sequentially [1-3]. Sequential training typically results in catastrophic forgetting, a phenomenon which occurs when training on the new task completely overwrites the synaptic weights learned during the previous task, leaving the ANN incapable of performing a previous task [1-4]. Attempts to solve catastrophic forgetting have drawn on insights from the study of neurobiological learning, leading to the growth of neuroscience-inspired artificial intelligence (AI) [5, 6]. These include equipping synapses with more complicated dynamics to ensure stability [7], or employing an interleaved training paradigm that has been hypothesized to occur during sleep [4, 8, 9]. However, although these approaches are capable of mitigating catastrophic forgetting in certain circumstances, a general solution which can achieve human level performance for continual learning is still an open question [6].

Parallel to the growth of neuroscience-inspired ANNs, there has been an increasing investigation of how spiking neural networks (SNNs) can be trained to perform complex tasks [10-13]. SNNs attempt to provide a realistic model of brain functioning by taking into account the underlying neural dynamics, such as spiking, and using biologically plausible local learning rules. This potentially makes SNNs well suited to study and to reveal specific mechanisms of how the brain avoids catastrophic forgetting. Indeed, there has already been some progress made along these lines showing that interleaved training can also help mitigate catastrophic forgetting in SNNs provided the network has time to settle back to its baseline state between stimulus presentations [10, 11]. However, these studies in SNNs have not focused on analyzing synaptic dynamics, which

will ultimately be critical to understanding how catastrophic forgetting is mitigated in biological systems.

Here we used a multi-layer SNN to investigate the effect of interleaved training on synaptic dynamics. The network was equipped with spike-timing-dependent plasticity (STDP) and biologically plausible homeostatic plasticity rules which have been studied experimentally in the brain. The network could be trained to learn one of two complementary complex foraging tasks involving pattern discrimination, and exhibits graceful degradation following neural pruning. We found that the network exhibits catastrophic forgetting when trained on the tasks sequentially, but interleaved training allowed the network to learn to perform well on both tasks. We further show that interleaved training does this by partially recovering the synaptic weight distributions learned during sequential training for each task and pushes the final synaptic weight configuration towards the decision boundary between the configurations associated with each task.

Results

Architecture of the spiking neural network model

In this study we have used a spiking neural network model which was previously designed to learn and perform a single complex foraging task in a virtual environment [12][13]; see Figure 1. The network was composed of 3 layers with a feed-forward architecture. The size of the input (I), hidden (H), and output (O) layers were 7x7, 28x28, and 3x3 respectively, and each neuron was simulated according to a reduced map-based model [14-16] (see *Methods: Map-based neuron model* for details). Input from the visual field was simulated as a set of suprathreshold inputs to the neurons in layer I which represent the position of particles in an egocentric reference frame relative to the virtual agent (positioned in the center of the visual field). The most active neuron in layer O (size 3x3) determined the direction of the subsequent movement. Each neuron in layer H received an excitatory synapse from 9 randomly selected neurons in layer I, while each neuron in layer O received both an excitatory and an inhibitory synapse from all neurons in layer H. Excitatory synapses between layers I and H were subjected to non-rewarded STDP [17, 18] while those between layers H and O were subjected to rewarded STDP [19-22] (see *Methods: Synaptic plasticity* for details). The strengths of each inhibitory synapse from a given neuron were set to always match the average strength of the excitatory

synapses sent by the same presynaptic neuron. Non-rewarded STDP allowed neurons in layer H to learn to represent and detect the 4 types of particle patterns in various spatial locations, while rewarded STDP allowed the neurons in layer O to make motor decisions based on the particle patterns detected in the visual field [12].

Complementary complex foraging tasks can be robustly learned

Using the network model described above, we began by training it on one of two complementary complex foraging tasks. In either task, the network learned to discriminate between a rewarded and a neutral particle pattern in order to acquire as much of the rewarded pattern as possible. In the following we consider pattern discriminability as a measure of performance, with chance performance being 0.5.

The paradigm for Task 1 is shown in Figure 2A. Task 1 consisted of an unsupervised training period during which 4 types 2-particle patterns (horizontal, vertical, positive diagonal, and negative diagonal) are present in the environment with equal densities. This was followed by a rewarded training period during which the synapses between layers I and H were frozen, and a testing period during which all synapses were frozen. During both the rewarded training and the testing periods only 2 types of patterns were present: horizontal (rewarded) and negative diagonal (neutral). Mean performance during the testing period was 0.686 ± 0.028. Figure 2B shows examples of trajectories of the simulated agent at the beginning of (left) and after (right) rewarded training. Prior to rewarded training the agent moves more or less randomly through the environment, but afterwards it moves in such a way which appears to seek out horizontal patterns and avoid negative diagonal ones.

The complementary paradigm for Task 2 is shown in Figure 2C. Note that Task 2 had the two complementary patterns present during rewarded training and testing: vertical (rewarded) and positive diagonal (neutral). Mean performance was obtained for the testing period of Task 2 was 0.710 ± 0.020, similar to that of Task 1. Figure 2D shows examples of trajectories of the simulated agent at the beginning of (left) and after (right) rewarded training for Task 2. After Task 2 training, the agent moves in such a way which appears to seek out vertical patterns and avoid positive diagonal ones. These results demonstrate that the network is capable of learning and performing either one of the two complementary complex foraging tasks.

Graceful degradation of performance following neural pruning

Previous work [12] suggested that this network learned to perform these types of complex foraging tasks in the following way: (1) neurons in layer H learned to respond only when a specific pair of I layer neurons (representing both particles of a given rewarded pattern) fired together, and (2) neurons in layer H learned to project strongly to a single neuron in layer O to mediate approach behavior. However, it remains unclear to what degree the network learns to use a distributed code. Thus, we tested whether this network model exhibited evidence of a distributed code by implementing a neural pruning protocol.

As shown in Figure 3A, the network was first put through a Task 1 training and testing paradigm identical to that shown in Figure 2A. Following this, we implemented a neural pruning protocol in which a randomly selected neuron in layer H was removed every 10 aeons until the layer H was empty (Figure 3B). It can be seen that performance remained stable until around 225,000 aeons (Figure 3A), or until nearly 700 of the 784 neurons in layer H were removed (Figure 3C). The fact that such a large portion of layer H can be removed before any significant drop in performance occurs is highly suggestive of the network using a distributed code to make decisions rather than relying on the activity of a few select H layer neurons.

Sequential training of both tasks leads to catastrophic forgetting

We next tested whether our network model could exhibit catastrophic forgetting by training sequentially on Task 1 followed by Task 2 (Figure 4A). As shown in Figure 4B, following Task 1 training, performance on Task 1 was 0.708 ± 0.035 , while, as expected, performance on Task 2 was no better than chance. Conversely, following Task 2 training, performance on Task 1 was now no better than chance, while performance on Task 2 improved to 0.697 ± 0.031. Thus, sequential training on a complementary task caused the network to undergo catastrophic forgetting of the task trained earlier.

To better understand the source of catastrophic forgetting in our network, we next analyzed the synaptic weights between layers H and O. The synaptic weights were found to be exponentially distributed, and no obvious differences could be seen in the overall distributional structure of these synaptic weights when compared after training on each task, (Figure 4C). However, important differences were observed if task-relevant synapses (i.e. synapses in the top 10% following training on that task) were analyzed (Figure 4D). The structure in the distribution

of Task 1-relevant synapses following Task 1 training (top-left) was destroyed following Task 2 training (top-right). Similarly, the structure in the distribution of Task 2-relevant synapses following Task 2 training (bottom-right) was not present following Task 1 training (bottom-left). Therefore, it suggests that, in our network, catastrophic forgetting results from a nearly complete overwriting of the synaptic weight matrix between layers H and O after each new task learning.

Interleaved training facilitates sequential learning without catastrophic forgetting

After confirming that our network exhibited catastrophic forgetting when trained sequentially, we added an interleaved training phase to our simulation (Figure 5A) to test whether it was a capable of rescuing performance on Task 1 without overwriting Task 2. Figure 5B shows that, following interleaved training, the network achieved a performance of 0.666 ± 0.048 on Task 1 and a performance of 0.679 ± 0.024 on Task 2. Moreover, from the very onset of interleaved training the performance was significantly above chance (Figure 5A). Therefore, interleaved training allowed the network to relearn Task 1 without forgetting what the network had just learned during training on Task 2. We next analyzed the synaptic weight distributions learned after each of the training phases. As before, the distributional structure of the synaptic weights was not noticeably different following training on Task 1 or Task 2 (Figure 4C). However, following interleaved training, the distribution became bimodal, with a new peak forming at intermediate values of synaptic weights centered around 0.1 (Figure 5C).

We hypothesized that interleaved training relearned Task 1-relevant synapses while simultaneously preserving Task 2-relevant weights. Figure 6A shows that this hypothesis is partially correct. The left and middle columns show analogous results to that of Figure 4D while the right column considers task-relevant synapses following interleaved training. In other words, here we identified task relevant synapses after training the task alone (Figure 6A, top/left for Task 1 and Figure 6A, middle/bottom for Task 2) and we then traced the same set of synapses after another task training or after interleaved training (we then plot the same set of synapses throughout different points in the simulation). It can be seen that for both tasks, the structure of the task-relevant synapses was partially preserved following interleaved training. For Task 2, while some of the task relevant synapses decayed after interleaved training, a significant fraction remained (compare Figure 6A, bottom/right to bottom/middle). Importantly, even though Task 1 relevant synapses have been almost completely wiped out by Task 2 training (compare Figure

6A, top/middle to top/left), interleaved training relearned a subset (possibly critical subset) of the Task 1-relevant synapses (compare Figure 6A, top/right to top/left). To summarize this effect across trials, we used the normalized Kullback – Leibler divergence (nKL_D), which provides a notion of how different one distribution is compared to a reference distribution. Figure 6B shows the mean nKL_D across trails for Task 1 and Task 2-relevant synapses after various training paradigms using, as reference distributions, the synaptic distributions learned by training on Task 1 or 2, respectively. The fact that interleaved training consistently resulted in roughly half the nKL_D compared to the training on the complementary task indicates that interleaved training partially recovered the task-relevant distributions. Given that the network was able to perform both Task 1 and Task 2, this suggests that interleaved training allowed the network (a) to relearn Task 1 by strengthening a subset of synapses which were trained originally for Task 1 while preserving a subset of the Task-2 relevant synapses; (b) to possibly strengthen an additional new set of synapses necessary for performing both tasks after interleaved training.

Interleaved training pushes the network to the decision boundary between Task 1 and Task 2 in synaptic weight space

To better understand the effect of interleaved training on the synaptic weights, we trained a support vector machine (SVM) with a radial basis function kernel to classify the synaptic weights between layers H and O according to whether they serve to perform Task 1 or Task 2. To train the SVM, for each task (Task 1 and Task 2), we selected multiple "snapshots" of the synaptic weight matrices obtained from the last fifth of the Task 1 and Task 2 training phases (i.e. after performance appeared to reach an asymptote). We then used this SVM to classify the synaptic weight matrices which evolved during the second half of interleaved training. Figure 7A shows the average classification value across trials for synaptic weights associated with Task 1, Task 2, and interleaved training. While the SVM robustly classified the synaptic weight matrices from Task 1 and Task 2, the distance of interleaved weight states to the decision boundary were significantly closer to the decision boundary (typically on the task 2 side). This indicates that the synaptic weight matrices from interleaved training are an amalgam of Task 1 and Task 2 states.

Since this classification occurred in a 784-dimensional synaptic weight space, we used principle components analysis (PCA) to reduce the dimensionality of the data in order to

visualize the trajectory of the synaptic weights at the network level over the course of a simulation (Figure 7B). Here the beginning of the grey trajectory represents the initial weight distribution, that evolved to the Task 1 (Task 2) specific distribution indicated by red (blue) dots, and eventually to the interleaved training specific distribution (green dots). It can be seen that while synaptic weight matrices associated with Task 1 and Task 2 cluster in distinct regions of PC space, interleaved training pushes the synaptic weights to an intermediate location between Task 1 and Task 2. Our goal of visualization limited us to using only 3 PCs, which combined only explained 40% of the total variance. Therefore, we repeated our SVM classification using the data projected in PC space as input. Figure 7D demonstrates that this gave the same qualitative result as using the high-dimensional data suggesting that PC space captures the major characteristics of weight space which we are concerned with.

Discussion

In this study we report that a multi-layer SNN may exhibit catastrophic forgetting upon sequential training of two complementary complex foraging tasks. Subsequent interleaved training of these tasks allowed the network to relearn Task 1 without forgetting Task 2, mitigating catastrophic forgetting. At the synaptic level, training a new task alone led to complete overwriting of synaptic weights responsible for the previous task. In contrast, interleaved training strengthened a set of previously weak synapses which were capable of performing Task 1 while preserving the synapses which were learned during Task 2 training. Interestingly, a set of synapses strengthened by interleaved training overlapped but was not identical to that strengthened by Task 1 or Task 2 training alone. At a network level, interleaved training resulted in the synaptic weight configuration being pushed towards the decision boundary between the configurations associated with each task. Thus, interleaved training caused the network to converge towards a synaptic weight configuration which was equally representative of each task - an optimal compromise for performing both tasks.

These results are in line with a large body of literature suggesting that interleaved training is capable of mitigating catastrophic forgetting in ANNs [4, 8, 9] and SNNs [10, 11]. Interestingly, it has been shown that humans actually learn to perform multiple tasks better under a sequential, batch training paradigm than they do under an interleaved paradigm [9]. This was because

sequential, batch training allowed human brains to develop a factorized representation which optimally segregated both tasks, thus reducing interference between the memories for each task. However, when trained under an interleaved paradigm, human brains failed to separate the two tasks, and instead formed a single linear boundary for a hybrid task [9]. Therefore, it appears that the SNN presented in this study was able to capture some aspects of how humans learn under an interleaved paradigm, but not under a sequential batch paradigm, suggesting additional mechanisms need to be incorporated into the model to capture the richness of human learning across contexts.

It has previously been suggested that attentional mechanisms might be able to assist in task segregation by increasing the gain on certain cell populations that shift over time depending on how autocorrelated the context is [9, 23]. Another possibility is that the hippocampus assists with task segregation by biasing memory allocation in the cortex according to episodic context. Indeed, it has been shown that the hippocampus allocates memories in overlapping neural ensembles if they occurred close in time, regardless of the spatial context [24, 25]. Additionally, the hippocampus is thought to index to cortical memories during sleep to bias memory reactivation and consolidation [26-29], possibly in an interleaved manner [8].

Importantly, while early psychology studies of learning and memory cast doubt on the idea that catastrophic forgetting was a phenomenon that could be exhibited by human brains, more recent work has shown that catastrophic forgetting can occur for hippocampal-independent memories under certain circumstances [1, 30, 31]. For instance, a learning protocol known as Fast Mapping is thought to be able to bypass the typical hippocampal-dependent encoding pathway. This is evidenced by the fact that amnesiacs with medial temporal lobe damage can learn novel word-meaning associations through a Fast Mapping protocol that they are incapable of encoding under standard learning paradigms [31]. By using this Fast Mapping protocol, researchers have been able to exhibit catastrophic forgetting in both amnesiacs and healthy subjects [30, 31]. Therefore, it may be possible to avoid catastrophic forgetting by incorporating a mechanism which takes into account the interactions between the cortex and hippocampus, thus gaining the benefits of sequential, block training during online learning and interleaved training during offline consolidation.

Methods

Environment. Foraging behavior took place in a virtual environment consisting of a 50x50 grid with randomly distributed "food" particles. Each particle was two pixels in length and could be classified into one of four types depending on its orientation: vertical, horizontal, positively-sloped diagonal, or negatively-sloped diagonal. During the initial unsupervised training period, the particles are distributed at random with the constraints that each of the four types are equally represented and no two particles can be directly adjacent. During training and testing periods only the task-relevant particles were distributed. When a particle was acquired as a result of the virtual agent moving, it was removed from its current location and randomly assigned to a new location on the grid, again with the constraint that it not be directly adjacent to another particle. This ensures a continuously changing environment with a constant particle density. The density of particles in the environment was set to 10%. The virtual agent can see a 7x7 grid of squares (the "visual field") centered on its current location and it could move to any adjacent square, including diagonally, for a total of eight directions.

Network structure. The network was composed of 842 spiking map-based neurons (see *Methods: Map-based neuron model* below) [16, 32], arranged into three feed-forward layers to mimic a basic biological circuit: a 7x7 input layer (I), a 28x28 middle (hidden, H) layer, and a 3x3 output layer (O) (Fig 1). This structure included a basic feed-forward inhibitory circuit [33] found in many biological structures [33-38].

Each neuron in layer H received synaptic input from nine random neurons in layer I. These connections initially had random strengths drawn from a normal distribution. Each neuron in layer H connected to every neuron in layer O with both an excitatory (W_{ij}) and an inhibitory (WI_{ij}) synapse. This provided an all-to-all connectivity pattern between these two layers and accomplished a balanced feed-forward inhibition. Initially, all these connections had uniform strengths and the responses in layer O were due to the random synaptic variability. Random variability was a property of all synaptic interactions between neurons and was implemented as variability in the magnitude of the individual synaptic events.

Movement cycle. Simulation time was divided up into epochs of 600 timesteps, each roughly equivalent to 300 ms. At the start of each epoch the virtual agent received input corresponding to locations of nearby particles within the 7x7 "visual field". Thus 48 of the 49 neurons in layer I

received input from a unique location relative to the virtual agent. At the end of the epoch the virtual agent made a single move based on the activity in layer O. If the virtual agent moved to a grid location with a "food" particle present, the particle was removed and assigned to a randomly selected new location.

Each epoch was of sufficient duration for the network to receive inputs, propagate activity forward, produce outputs, and return to a resting state. Neurons in layer I which represent locations in the visual field containing particles received a brief pulse of excitatory stimulation sufficient to trigger a spike; this stimulation was applied at the start of each movement cycle (epoch). At the end of each epoch the virtual agent moved according to the activity which has occurred in layer O.

The activity in layer O controlled the direction of the virtual agent's movement. Each of the neurons in layer O mapped onto a specific direction (i.e. one of the eight adjacent locations or the current location). The neuron in layer O which spiked the greatest number of times during the first half of the epoch defined the direction of movement for that epoch. If there was a tie, the direction was chosen at random from the set of tied directions. If no neurons in layer O spiked, the virtual agent continued in the direction it had moved during the previous epoch.

There was a 1% chance on every move that the virtual agent would ignore the activity in layer O and instead move in a random direction. Moreover, for every movement cycle that passed without the virtual agent acquiring a particle, this probability was increased by 1%. The random variability prevented the virtual agent from getting stuck in movement patterns corresponding to infinite loops. Synaptic noise was not sufficient to break out of all infinite loops as some loops were the result of forming strong connections was would facilitate the same spiking pattern despite the noise. Other times, the probability of escape from a loop due to noise was simply so low that it would take an impractical amount of time to break the loop. While biological systems could utilize various different mechanisms to achieve the same goal, the method we implemented was efficient and effective for the scope of our study.

Map-based neuron model. The underlying reduced model of a fast-spiking neuron was identical to the model used in [12, 13] and can be described by the following set of difference equations [14-16]:

$$V_{n+1} = f_{\alpha}(V_n, I_n + \beta_n),$$

$$I_{n+1} = I_n - \mu(V_n + 1) + \mu\sigma + \mu\sigma_n,$$

where V_n is the membrane potential, I_n is a slow dynamical variable describing the effects of slow conductances, and n is a discrete time-step (0.5 ms). Slow temporal evolution of I_n was achieved by using small values of the parameter $\mu << 1$. Input variables β_n and σ_n were used to incorporate external current I^{ext} (e.g. background synaptic input): $\beta_n = \beta^e I^{ext}$, $\sigma_n = \sigma^e I^{ext}$. Parameter values were set to $\sigma = 0.06$, $\beta^e = 0.133$, $\sigma^e = 1$, and $\mu = 0.0005$. The nonlinearity $f_\alpha(V_n, I_n)$ was defined in the form of the piece-wise continuous function:

$$f_\alpha(V_n, I_n) = \begin{cases} \alpha(1 - V_n)^{-1} + I_n, & V_n \leq 0 \\ \alpha + I_n, & 0 < V_n < \alpha + I_n \,\&\, V_{n-1} \leq 0 \\ -1 & \alpha + I_n \leq V_n \; or \; V_{n-1} > 0, \end{cases}$$

where $\alpha = 3.65$.

This model is very computationally efficient, and, despite its intrinsic low dimensionality, produces a rich repertoire of dynamics capable of mimicking the dynamics of Hodgkin-Huxley type neurons both at the single neuron level and in the context of network dynamics [14, 16].

To model the synaptic interactions, we used conventional first-order kinetic models of synaptic conductances rewritten in the form of the piece-wise difference equation:

$$g_{n+1}^{syn} = \gamma g_n^{syn} + \begin{cases} (1 + XR) g_{syn}/W_j, & spike_{pre} \\ 0, & \text{otherwise}, \end{cases}$$

and the synaptic current is computed as:

$$I_n^{syn} = - g_n^{syn}(V_n^{post} - V_{rp}).$$

Here g_{syn} is the strength of the synaptic coupling, modulated by the target rate W_j of receiving neuron j. Indices *pre* and *post* stand for the pre- and post-synaptic variables, respectively. The first condition, $spike_{pre}$, is satisfied when the pre-synaptic spikes are generated. Parameter γ controls the relaxation rate of synaptic conductance after a presynaptic spike is received ($0 \leq \gamma < 1$). The parameter R is the coefficient of variability in synaptic release. The standard value of R is 0.12. X is a random variable sampled from a uniform distribution with range [-1, 1]. Parameter V_{rp} defines the reversal potential and, therefore, the type of synapse (i.e. excitatory or inhibitory). The term $(1+XR)$ introduces a variability in synaptic release such that the effect of any synaptic

interaction has an amplitude that is pulled from a uniform distribution with range [1-*R*, 1+*R*] multiplied by the average value of the synapse.

Synaptic plasticity. Synaptic plasticity closely followed the rules introduced in [12, 13]. A rewarded STDP rule [19-22] was operated on synapses between layers H and O while a standard STDP rule operated on synapses between layers I and H. A spike in a post-synaptic neuron that directly followed a spike in a pre-synaptic neuron created a *pre before post* event while the converse created a *post before pre* event. Each new post-synaptic (pre-synaptic) spike was compared to all pre-synaptic (post-synaptic) spikes with a time window of 120 iterations.

The value of an STDP event (trace) was calculated using the following equation [17, 18]:

$$p = \frac{-|t_r - t_p|}{T_c},$$

$$tr_k = Ke^p$$

where t_r and t_p are the times at which the pre- and post-synaptic spike events occurred respectively, T_c is the time constant and is set to 40 ms, and K is maximum value of the trace tr_k and is set to -0.04 for a *post before pre* event and 0.04 for a *pre before post* event.

A trace was immediately applied to synapse between neurons in layers I and H. However, for synapses between neurons in layers H and O the traces were stored for 6 epochs after its creation before being erased. During storage, a trace had an effect whenever there was a rewarding or punishing event. In such a case, the synaptic weights are updated as follows:

$$W_{ij} \leftarrow W_{ij} \prod_{k}^{traces} \left(1 + \frac{W_{i0}}{W_i} * \Delta_k\right),$$

$$\Delta_k = S_{rp}\left(\frac{tr_k}{t - t_k + c}\right)\frac{Sum_{tr}}{Avg_{tr}},$$

$$Sum_{tr} = \sum_{k}^{traces} \frac{tr_k}{t - t_k + c},$$

$$Avg_{tr} \leftarrow (1 - \delta)Avg_{tr} + \delta Sum_{tr},$$

where t is the current timestep, S_{rp} is a scaling factor for reward/punishment, tr_k is the magnitude of the trace, t_k is the time of the trace event, c is a constant (=1 epoch) used for decreasing sensitivity to very recent spikes, $W_i = \Sigma_j W_{ij}$ is the total synaptic strength of all connections from the neuron i in layer H to all neurons in layer O, W_{i0} is a constant that is set to the initial value (*target value*) of W_i at the beginning of the simulation. The term W_{i0}/W_i helped to keep the output weight sum close to the initial target value. The effect of these rules was that neurons with lower total output strength could increase their output strength more easily.

The network was rewarded when the virtual agent moved to a location which contained a particle from a "food" pattern (horizontal in Task 1, vertical in Task 2) and $S_{rp} = 1$, but not when it moved to a location with a particle from a neutral pattern (negative/positive diagonal in Task 1/2). A small punishment of $S_{rp} = -0.01$ was applied if the agent moved to a location without a particle present to help the virtual agent learn to acquire "food" as rapidly as possible.

To ensure that neurons in layer O maintained a relatively constant long-term firing rate, the model incorporated homeostatic synaptic scaling which was applied every epoch. Each timestep, the total strength of synaptic inputs $W_j = \Sigma_i W_{ij}$ to a given neuron in layer O was set equal to the target synaptic input W_{j0} – a slow variable which varied over many epochs depending on the activity of the given neuron in layer O – which was updated according to:

$$W_{j0} \leftarrow \begin{cases} W_{j0}(1 + D_{tar}) & \text{spike rate} < \text{target rate} \\ W_{j0}(1 - D_{tar}) & \text{spike rate} > \text{target rate} \end{cases}$$

To ensure that the net synaptic input W_j to any neuron was unaffected by plasticity events at the individual synapses at distinct timesteps and equal to W_{j0}, we implemented a scaling process akin to heterosynaptic plasticity which occurs after each STDP event. When any excitatory synapse of neuron in layer O changed in strength, all other excitatory synapses received by that neuron were updated according to:

$$W_{ij} \leftarrow W_{ij} \frac{W_{j0}}{\sum_i W_{ij}}$$

Figure legends

Figure 1. Network architecture. The network is composed of three layers of neurons and follows a feed-forward connectivity scheme. Input from the "visual field" was simulated as a set of excitatory inputs to input layer neurons representing the position of food particles in an egocentric reference frame relative to the virtual agent (positioned in the center of the visual field). Each neuron in the hidden layer received one excitatory and one inhibitory synapse from 9 randomly selected input layer neurons. Excitatory synapses between these neurons are subject to unsupervised STDP which allowed the hidden layer to learn object representations. Each neuron in the output layer received one excitatory and one inhibitory synapse from each hidden layer neuron. The most active neuron in the output layer (size 3x3) determines the direction of the subsequent movement. Excitatory synapses between the hidden and output layer were subject to rewarded STDP where reward depends on acquiring a "food" particle as a result of movement. The strengths of each inhibitory synapse from a given neuron always matches the average strength of the excitatory synapses sent by the same presynaptic neuron.

Figure 2. Two complementary complex foraging tasks can be learned independently. (A) Trace of the mean performance (blue line) and standard deviation (red lines) across time for a Task 1 paradigm: 10,000 aeons of unsupervised training (white), 5000 aeons of Task 1 training (blue), and 1,000 aeons of Task 1 testing (green). The y-axis is the probability of the network acquiring the rewarded particle pattern as opposed to the neutral particle pattern calculated as an exponential moving average. The x-axis is time in aeons (each aeon consisted of 1000 epochs, which in turn consisted of 600 timesteps). Mean performance during testing on Task 1 was 0.686 ± 0.028. (B) Examples of trajectories through the environment at the beginning (left) and at the end (right) of training on Task 1. (C). The same as shown in (A) except now for a Task 2 paradigm: 10,000 aeons of unsupervised training (white), 5000 aeons of Task 2 training (red), and 1,000 aeons of Task 2 testing (yellow). Mean performance during testing on Task 2 was 0.710 ± 0.020. (D). The same as shown in (B) except now for Task 2.

Figure 3. Graceful degradation of performance resulting from random hidden layer neural pruning. (A) Trace of the mean performance (blue line) and standard deviation (red lines) of the network across time. The network undergoes a Task 1 paradigm followed by a neural pruning protocol (cyan) during which the synaptic weights remain frozen while performance is

continuously monitored with an exponential moving average. (B) The number of neurons present in the hidden layer across time. The neural pruning protocol begins at 16,000 aeons, during which 1 randomly selected hidden layer neuron is removed every 10 aeons until the hidden layer is empty. (C) Network performance as a function of the number of neurons removed from the hidden layer. Performance remains stable until nearly 700 of the 784 neurons in the hidden layer are removed.

Figure 4. Sequential training on complementary task induces catastrophic forgetting of initial complex foraging task. (A) Trace of the mean performance (blue line) and standard deviation (red lines) of the network across time for a catastrophic forgetting paradigm: 10,000 aeons of unsupervised training (white), 5,000 aeons of Task 1 training (blue), 1,000 aeons of Task 1 testing (green), 1,000 aeons of Task 2 testing (yellow), 5,000 aeons of Task 2 testing, 1,000 aeons of Task 1 testing (green), and 1,000 aeons of Task 2 testing (yellow). (B) Bar plot of the mean performance during testing on Task 1 (blue) and Task 2 (red) after training on each task. Error bars indicate the standard deviation. Following Task 1 training, the mean performance on Task 1 was 0.708 ± 0.035 while that on Task 2 was 0.526 ± 0.014. Conversely, following Task 2 training, the mean performance on Task 1 was 0.509 ± 0.038 while that on Task 2 was 0.697 ± 0.031. (C) Distributions of synaptic weights following training on Task 1 (left) and Task 2 (right). The y-axis shows the number of synapses between the hidden and output layers which have the corresponding synaptic weight displayed on the x-axis. (D) Distributions of task-relevant synaptic weights. The structure in the distribution of Task 1-relevant synapses following Task 1 training (top-left) is destroyed following Task 2 training (top-right). Similarly, the structure in the distribution of Task 2-relevant synapses following Task 2 training (bottom-right) was not present following Task 1 training (bottom-left). Task-relevant synapses were considered to be those which had a synaptic weight of at least 0.1 following training on that task.

Figure 5. Interleaved training prevents catastrophic forgetting. (A) Trace of the mean performance (blue line) and standard deviation (red lines) of the network across time. The network undergoes a catastrophic forgetting paradigm followed by 5,000 aeons of interleaved training (purple), 1,000 aeons of Task 1 testing (green), and 1,000 aeons of Task 2 testing (yellow). During interleaved training the network is trained on Task 1 and Task 2 in a rapid succession of 1 aeon intervals. (B) Bar plot of the mean performance during testing on Task 1 (blue) and Task 2 (red) after each training period (Task 1, Task 2, and interleaved). Error bars

indicate the standard deviation. Following Task 1 training, the mean performance on Task 1 was 0.708 ± 0.035 while that on Task 2 was 0.526 ± 0.014. Conversely, following Task 2 training, the mean performance on Task 1 was 0.509 ± 0.038 while that on Task 2 was 0.697 ± 0.031. However, following interleaved training, the mean performance on Task 1 was 0.666 ± 0.048 while that on Task 2 was 0.679 ± 0.024. (C) Distribution of synaptic weights following interleaved training. The y-axis shows the number of synapses between the hidden and output layers which have the corresponding synaptic weight displayed on the x-axis. The distribution is bimodal with peaks at 0 and ~0.1.

Figure 6. Interleaved training partially recovers the distributional structure of task-relevant synapses for both tasks. (A) Distributions of task-relevant synaptic weights. The structure in the distribution of Task 1-relevant synapses following Task 1 training (top-left) is destroyed following Task 2 training (top-middle), but partially recovered following interleaved training (top-right). Similarly, the structure in the distribution of Task 2-relevant synapses following Task 2 training (bottom-middle), which was not present following Task 1 training (bottom-left), was partially preserved following interleaved training (bottom-right). Task-relevant synapses were considered to be those which had a synaptic weight of at least 0.1 following training on that task. (B) Normalized Kullback – Leibler divergence of the task-relevant synaptic weight distributions for Task 1 and Task 2 following training on the opposing tasks (blue bars) and interleaved training (orange bars). The reference distributions in each case were the task-relevant distributions which resulted after training on the associated task (e.g. the distribution following training on Task 1 for the Task 1-relevant synaptic distributions). For both Task 1 and Task 2-relevant synaptic weight distributions, the normalized Kullback – Leibler divergence is smaller after interleaved training compared to after training on the opposing task.

Figure 7. Interleaved training pushes the synapses closer to the decision boundary between the two tasks in synaptic weight space. (A) Box plots of SVM classification value of synaptic weight matrices for Task 1, Task 2, and interleaved training across trials. Dashed Green line indicates mean, orange line denotes median. Classification was tested after training the SVM on data from only Task 1 and Task 2. Task 1 and Task 2 synaptic weight matrices had mean classification values of -0.0070 and 0.00685 respectively, while that of interleaved training was 0.00141. (B) Trajectory of H to O layer synaptic weights through PC space. Synaptic weights which evolved during interleaved training clustered in a location of PC space intermediary

between the clusters of synaptic weights which evolved during training on Task 1 and Task 2. (C) Mean percent of the variance in synaptic weights explained by the top three PCs used for visualization. Error bars indicate standard deviation. PC1, PC2, and PC3 explained 21.5% ± 4.17%, 11.1% ± 2.0%, and 7.7% ± 1.3% of the variance respectively, and 40.4% ± 4.47% of the variance cumulatively. (D) Boxplots of SVM classification value of synaptic weight matrices for Task 1, Task 2, and interleaved training across trials after projecting the data into 3-dimensional PC space. Task 1 and Task 2 synaptic weight matrices had mean classification values of -1.02 and 1.03 respectively, while that of interleaved training was 0.34, preserving the qualitative result obtained with the high dimensional data.

References:

1. French RM. Catastrophic forgetting in connectionist networks. Trends Cogn Sci. 1999;3(4):128-35. Epub 1999/05/14. PubMed PMID: 10322466.
2. Mccloskey M, Cohen NJ. CATASTROPHIC INTERFERENCE IN CONNECTIONIST NETWORKS: THE SEQUENTIAL LEARNING PROBLEM. The Psychology of Learning and Motivation. 1989;24:109-65.
3. Ratcliff R. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. Psychol Rev. 1990;97(2):285-308. Epub 1990/04/01. PubMed PMID: 2186426.
4. Hasselmo ME. Avoiding Catastrophic forgetting. Trends Cogn Sci. 2017;21(6):407-8. Epub 2017/04/27. doi: 10.1016/j.tics.2017.04.001. PubMed PMID: 28442279.
5. Hassabis D, Kumaran D, Summerfield C, Botvinick M. Neuroscience-Inspired Artificial Intelligence. Neuron. 2017;95(2):245-58. Epub 2017/07/21. doi: 10.1016/j.neuron.2017.06.011. PubMed PMID: 28728020.
6. Kemker R, McClure M, Abitino A, Hayes T, Kanan C. Measuring Catastrophic forgetting in Neural Networks. arXiv e-prints [Internet]. 2017 August 01, 2017. Available from: https://ui.adsabs.harvard.edu/\#abs/2017arXiv170802072K.
7. Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, Rusu AA, et al. Overcoming catastrophic forgetting in neural networks. Proc Natl Acad Sci U S A. 2017;114(13):3521-6. Epub 2017/03/16. doi: 10.1073/pnas.1611835114. PubMed PMID: 28292907; PubMed Central PMCID: PMCPMC5380101.
8. McClelland JL, McNaughton BL, O'Reilly RC. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. Psychol Rev. 1995;102(3):419-57. Epub 1995/07/01. PubMed PMID: 7624455.
9. Flesch T, Balaguer J, Dekker R, Nili H, Summerfield C. Comparing continual task learning in minds and machines. Proc Natl Acad Sci U S A. 2018;115(44):E10313-E22. Epub 2018/10/17. doi: 10.1073/pnas.1800755115. PubMed PMID: 30322916; PubMed Central PMCID: PMCPMC6217400.
10. Evans BD, Stringer SM. Transformation-invariant visual representations in self-organizing spiking neural networks. Front Comput Neurosci. 2012;6:46. Epub 2012/08/01. doi: 10.3389/fncom.2012.00046. PubMed PMID: 22848199; PubMed Central PMCID: PMCPMC3404434.
11. Higgins I, Stringer S, Schnupp J. Unsupervised learning of temporal features for word categorization in a spiking neural network model of the auditory brain. PLoS One. 2017;12(8):e0180174. Epub 2017/08/11. doi: 10.1371/journal.pone.0180174. PubMed PMID: 28797034; PubMed Central PMCID: PMCPMC5552261.
12. Sanda P, Skorheim S, Bazhenov M. Multi-layer network utilizing rewarded spike time dependent plasticity to learn a foraging task. PLoS Comput Biol. 2017;13(9):e1005705. Epub 2017/09/30. doi: 10.1371/journal.pcbi.1005705. PubMed PMID: 28961245; PubMed Central PMCID: PMCPMC5636167.
13. Skorheim S, Lonjers P, Bazhenov M. A spiking network model of decision making employing rewarded STDP. PLoS One. 2014;9(3):e90821. Epub 2014/03/19. doi: 10.1371/journal.pone.0090821. PubMed PMID: 24632858; PubMed Central PMCID: PMCPMC3954625.
14. Bazhenov M, Rulkov NF, Fellous JM, Timofeev I. Role of network dynamics in shaping spike timing reliability. Phys Rev E Stat Nonlin Soft Matter Phys. 2005;72(4 Pt 1):041903. Epub 2005/12/31. doi: 10.1103/PhysRevE.72.041903. PubMed PMID: 16383416.
15. Rulkov NF. Modeling of spiking-bursting neural behavior using two-dimensional map. Phys Rev E Stat Nonlin Soft Matter Phys. 2002;65(4 Pt 1):041922. Epub 2002/05/15. doi: 10.1103/PhysRevE.65.041922. PubMed PMID: 12005888.
16. Rulkov NF, Timofeev I, Bazhenov M. Oscillations in large-scale cortical networks: map-based model. J Comput Neurosci. 2004;17(2):203-23. Epub 2004/08/13. doi: 10.1023/B:JCNS.0000037683.55688.7e. PubMed PMID: 15306740.

17. Bi GQ, Poo MM. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J Neurosci. 1998;18(24):10464-72. Epub 1998/12/16. PubMed PMID: 9852584.
18. Markram H, Lubke J, Frotscher M, Sakmann B. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. Science. 1997;275(5297):213-5. Epub 1997/01/10. PubMed PMID: 8985014.
19. Farries MA, Fairhall AL. Reinforcement learning with modulated spike timing dependent synaptic plasticity. J Neurophysiol. 2007;98(6):3648-65. Epub 2007/10/12. doi: 10.1152/jn.00364.2007. PubMed PMID: 17928565.
20. Florian RV. Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. Neural Comput. 2007;19(6):1468-502. Epub 2007/04/21. doi: 10.1162/neco.2007.19.6.1468. PubMed PMID: 17444757.
21. Izhikevich EM. Solving the distal reward problem through linkage of STDP and dopamine signaling. Cereb Cortex. 2007;17(10):2443-52. Epub 2007/01/16. doi: 10.1093/cercor/bhl152. PubMed PMID: 17220510.
22. Legenstein R, Pecevski D, Maass W. A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. PLoS Comput Biol. 2008;4(10):e1000180. Epub 2008/10/11. doi: 10.1371/journal.pcbi.1000180. PubMed PMID: 18846203; PubMed Central PMCID: PMCPMC2543108.
23. Goldstone RL, Gerganov A, Landy D, Roberts ME. Learning to See and Conceive. Vienna Ser Theor Bio. 2009:163-88. PubMed PMID: WOS:000279130800013.
24. Cai DJ, Aharoni D, Shuman T, Shobe J, Biane J, Song W, et al. A shared neural ensemble links distinct contextual memories encoded close in time. Nature. 2016;534(7605):115-8. Epub 2016/06/03. doi: 10.1038/nature17955. PubMed PMID: 27251287; PubMed Central PMCID: PMCPMC5063500.
25. Rubin A, Geva N, Sheintuch L, Ziv Y. Hippocampal ensemble dynamics timestamp events in long-term memory. Elife. 2015;4. Epub 2015/12/20. doi: 10.7554/eLife.12247. PubMed PMID: 26682652; PubMed Central PMCID: PMCPMC4749549.
26. McNaughton BL. Cortical hierarchies, sleep, and the extraction of knowledge from memory. Artif Intell. 2010;174(2):205-14. doi: 10.1016/j.artint.2009.11.013. PubMed PMID: WOS:000274867000013.
27. Skelin I, Kilianski S, McNaughton BL. Hippocampal coupling with cortical and subcortical structures in the context of memory consolidation. Neurobiol Learn Mem. 2018. Epub 2018/04/17. doi: 10.1016/j.nlm.2018.04.004. PubMed PMID: 29660400.
28. Teyler TJ, DiScenna P. The hippocampal memory indexing theory. Behav Neurosci. 1986;100(2):147-54. Epub 1986/04/01. PubMed PMID: 3008780.
29. Teyler TJ, Rudy JW. The hippocampal indexing theory and episodic memory: updating the index. Hippocampus. 2007;17(12):1158-69. Epub 2007/08/19. doi: 10.1002/hipo.20350. PubMed PMID: 17696170.
30. Borragan G, Urbain C, Schmitz R, Mary A, Peigneux P. Sleep and memory consolidation: motor performance and proactive interference effects in sequence learning. Brain Cogn. 2015;95:54-61. Epub 2015/02/16. doi: 10.1016/j.bandc.2015.01.011. PubMed PMID: 25682352.
31. Merhav M, Karni A, Gilboa A. Neocortical catastrophic interference in healthy and amnesic adults: a paradoxical matter of time. Hippocampus. 2014;24(12):1653-62. Epub 2014/08/27. doi: 10.1002/hipo.22353. PubMed PMID: 25154723.
32. Rulkov NF, Bazhenov M. Oscillations and synchrony in large-scale cortical network models. J Biol Phys. 2008;34(3-4):279-99. Epub 2009/08/12. doi: 10.1007/s10867-008-9079-y. PubMed PMID: 19669478; PubMed Central PMCID: PMCPMC2585628.

622 33. Bazhenov M, Stopfer M. Forward and back: motifs of inhibition in olfactory processing. Neuron.
623 2010;67(3):357-8. Epub 2010/08/11. doi: 10.1016/j.neuron.2010.07.023. PubMed PMID: 20696373;
624 PubMed Central PMCID: PMCPMC5337941.
625 34. Bruno RM. Synchrony in sensation. Curr Opin Neurobiol. 2011;21(5):701-8. Epub 2011/07/05.
626 doi: 10.1016/j.conb.2011.06.003. PubMed PMID: 21723114; PubMed Central PMCID: PMCPMC3191231.
627 35. Dong H, Shao Z, Nerbonne JM, Burkhalter A. Differential depression of inhibitory synaptic
628 responses in feedforward and feedback circuits between different areas of mouse visual cortex. J Comp
629 Neurol. 2004;475(3):361-73. Epub 2004/06/29. doi: 10.1002/cne.20164. PubMed PMID: 15221951.
630 36. Pouille F, Scanziani M. Enforcement of temporal fidelity in pyramidal cells by somatic feed-
631 forward inhibition. Science. 2001;293(5532):1159-63. Epub 2001/08/11. doi: 10.1126/science.1060342.
632 PubMed PMID: 11498596.
633 37. Shao Z, Burkhalter A. Different balance of excitation and inhibition in forward and feedback
634 circuits of rat visual cortex. J Neurosci. 1996;16(22):7353-65. Epub 1996/11/15. PubMed PMID:
635 8929442.
636 38. Silberberg G. Polysynaptic subcircuits in the neocortex: spatial and temporal diversity. Curr Opin
637 Neurobiol. 2008;18(3):332-7. Epub 2008/09/20. doi: 10.1016/j.conb.2008.08.009. PubMed PMID:
638 18801433.
639

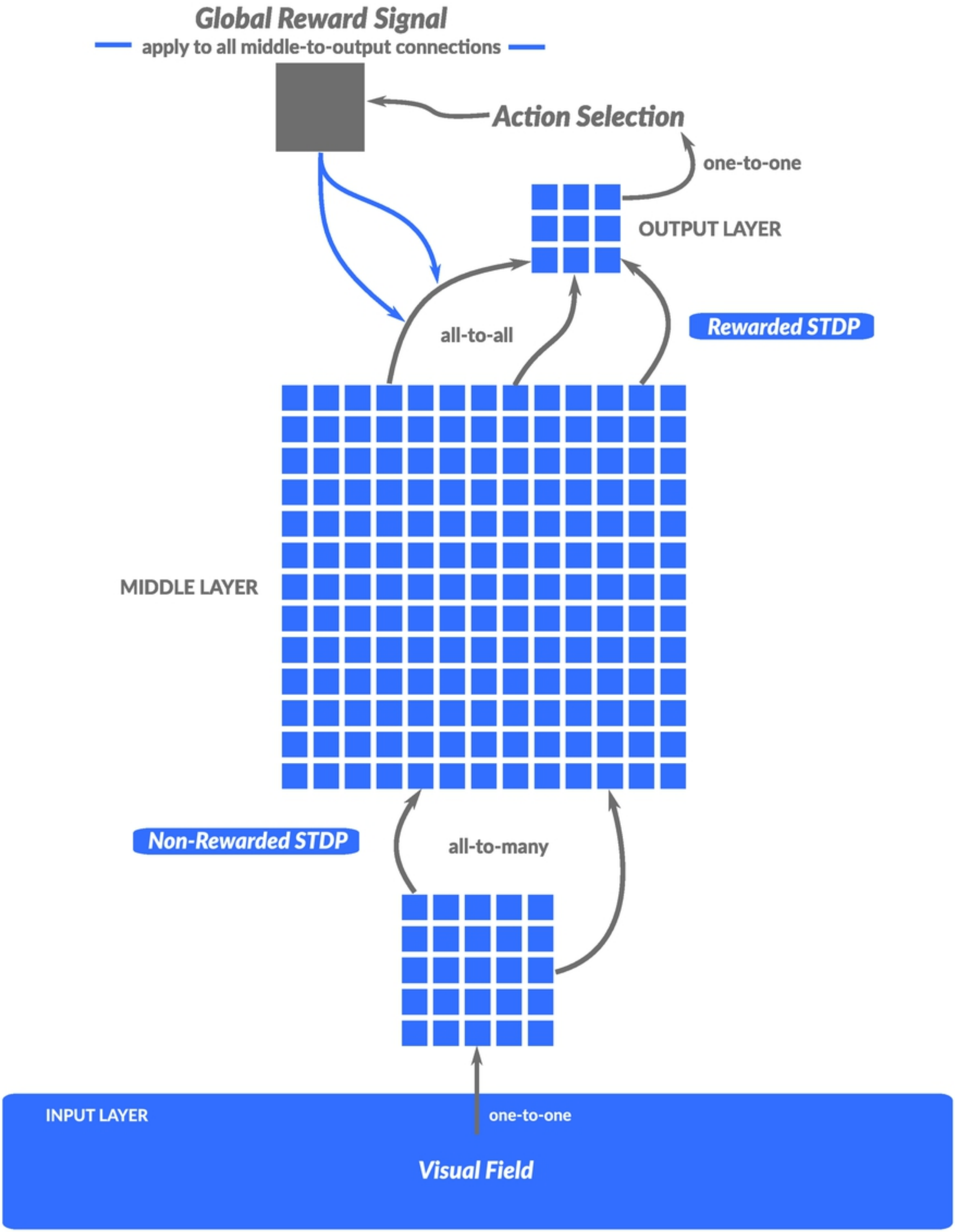

Global Reward Signal
apply to all middle-to-output connections
Action Selection
one-to-one
OUTPUT LAYER
all-to-all
Rewarded STDP
MIDDLE LAYER
Non-Rewarded STDP
all-to-many
INPUT LAYER
one-to-one
Visual Field

Figure 1

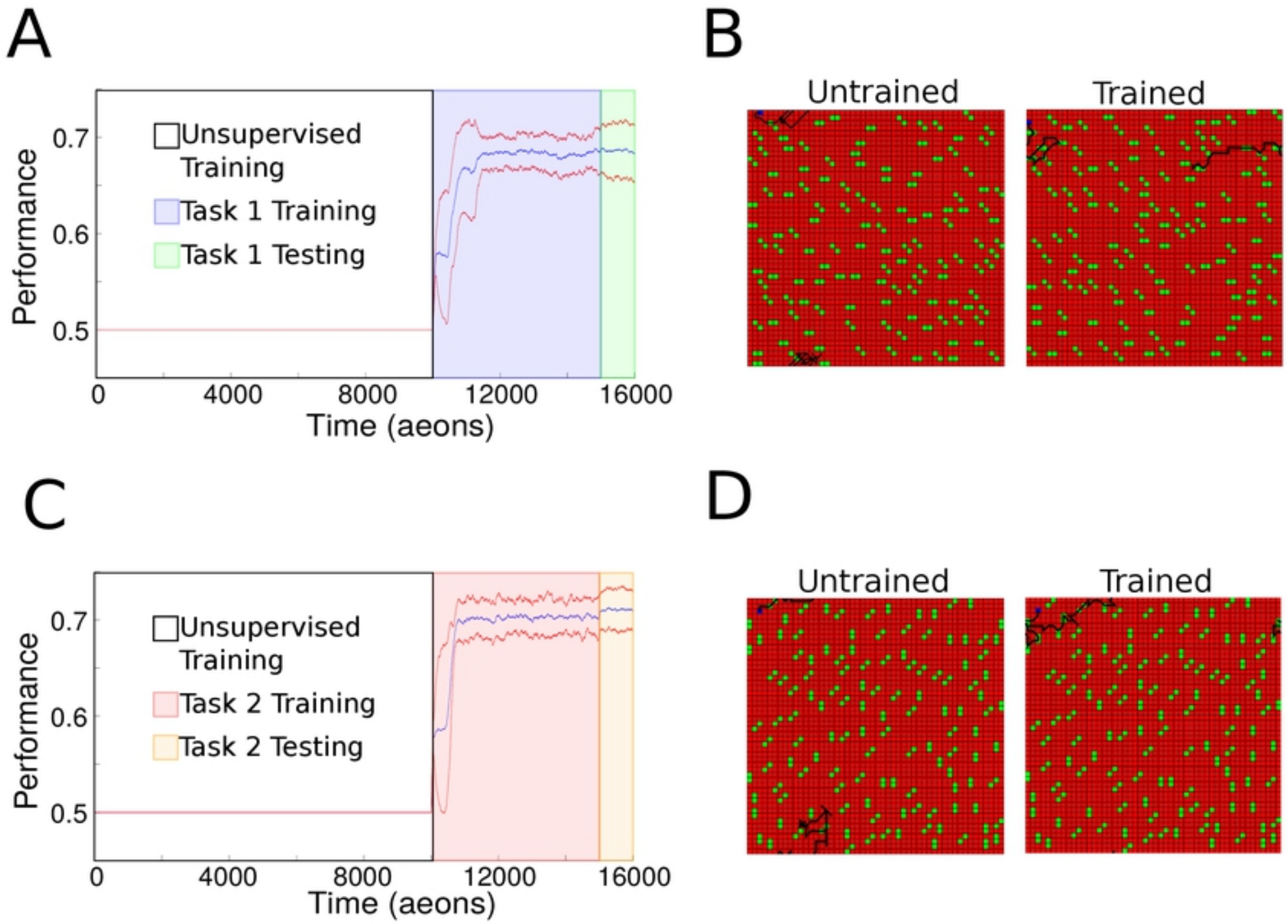

A
Performance
0.7
0.6
0.5
Unsupervised Training
Task 1 Training
Task 1 Testing
0
4000
8000
12000
16000
Time (aeons)
B
Untrained
Trained
C
Performance
0.7
0.6
0.5
Unsupervised Training
Task 2 Training
Task 2 Testing
0
4000
8000
12000
16000
Time (aeons)
D
Untrained
Trained

Figure 2

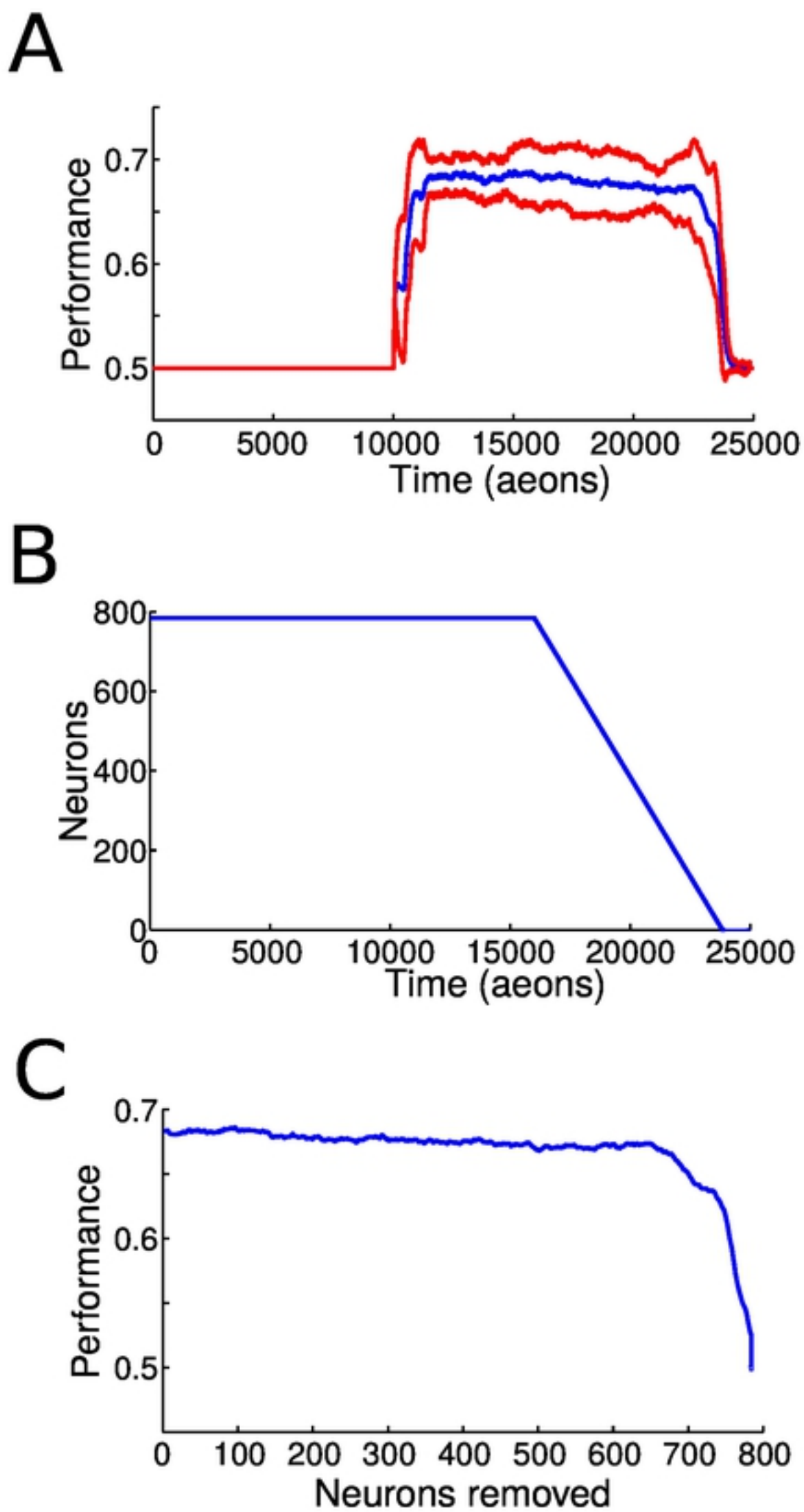

A
Performance
0.7
0.6
0.5
0
5000
10000
15000
20000
25000
Time (aeons)
B
Neurons
800
600
400
200
0
0
5000
10000
15000
20000
25000
Time (aeons)
C
Performance
0.7
0.6
0.5
0
100
200
300
400
500
600
700
800
Neurons removed

Figure 3

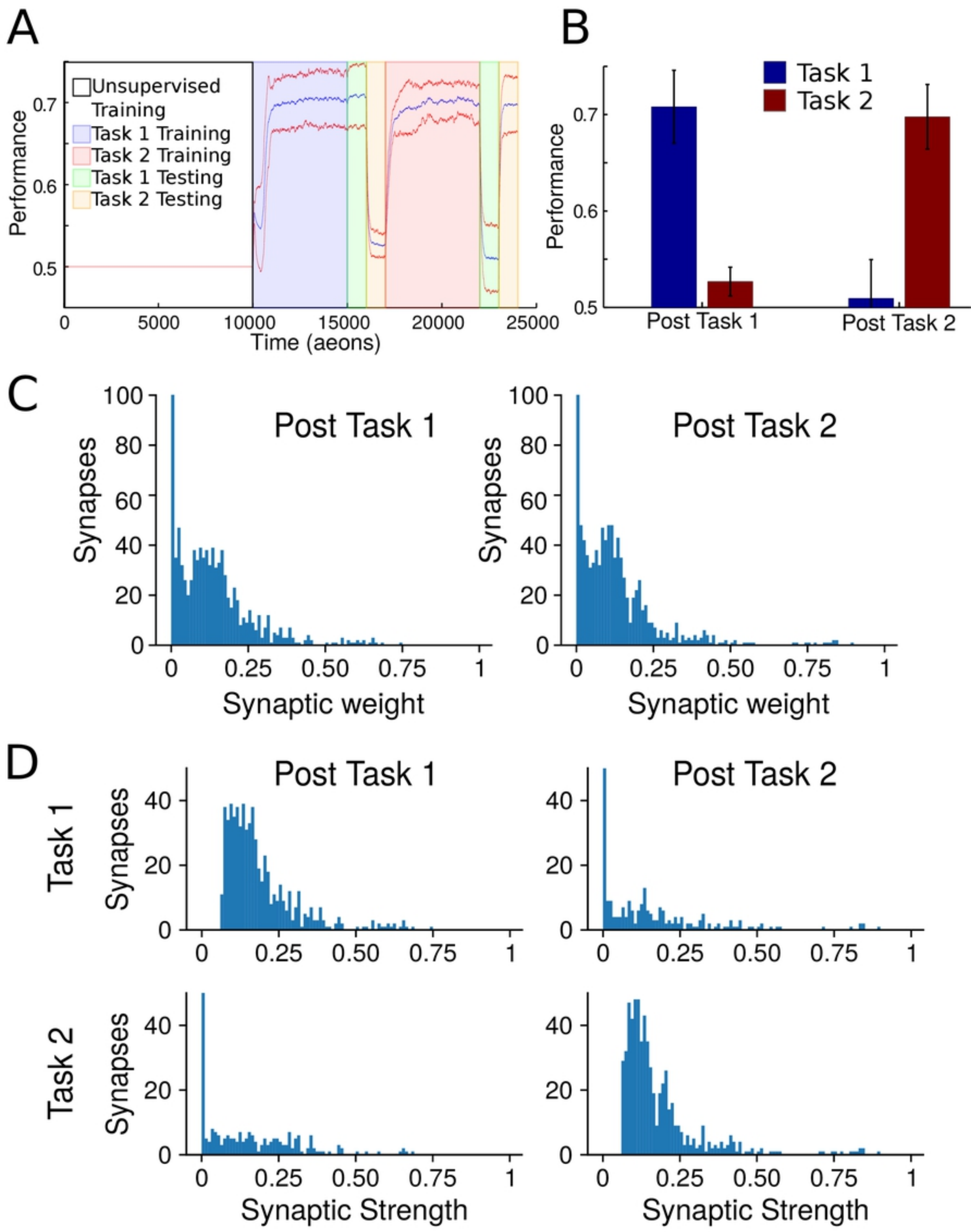
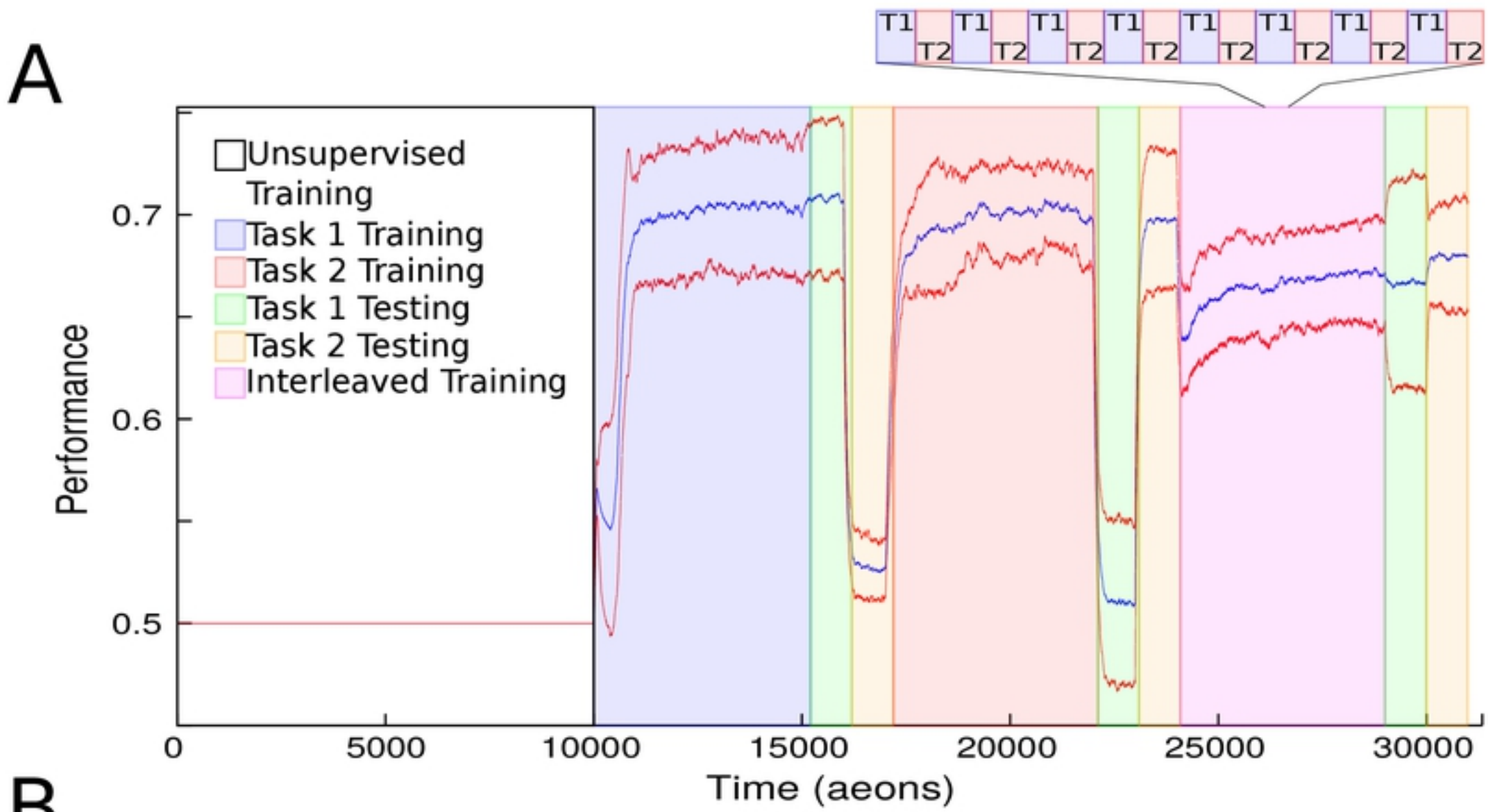

A
Unsupervised Training
Task 1 Training
Task 2 Training
Task 1 Testing
Task 2 Testing
Performance
Time (aeons)
B
Task 1
Task 2
Performance
Post Task 1
Post Task 2
C
Post Task 1
Post Task 2
Synapses
Synaptic weight
D
Task 1
Task 2
Post Task 1
Post Task 2
Synapses
Synaptic Strength

Figure 4


A
T1
T2
Unsupervised Training
Task 1 Training
Task 2 Training
Task 1 Testing
Task 2 Testing
Interleaved Training
Performance
0.7
0.6
0.5
0
5000
10000
15000
20000
25000
30000
Time (aeons)

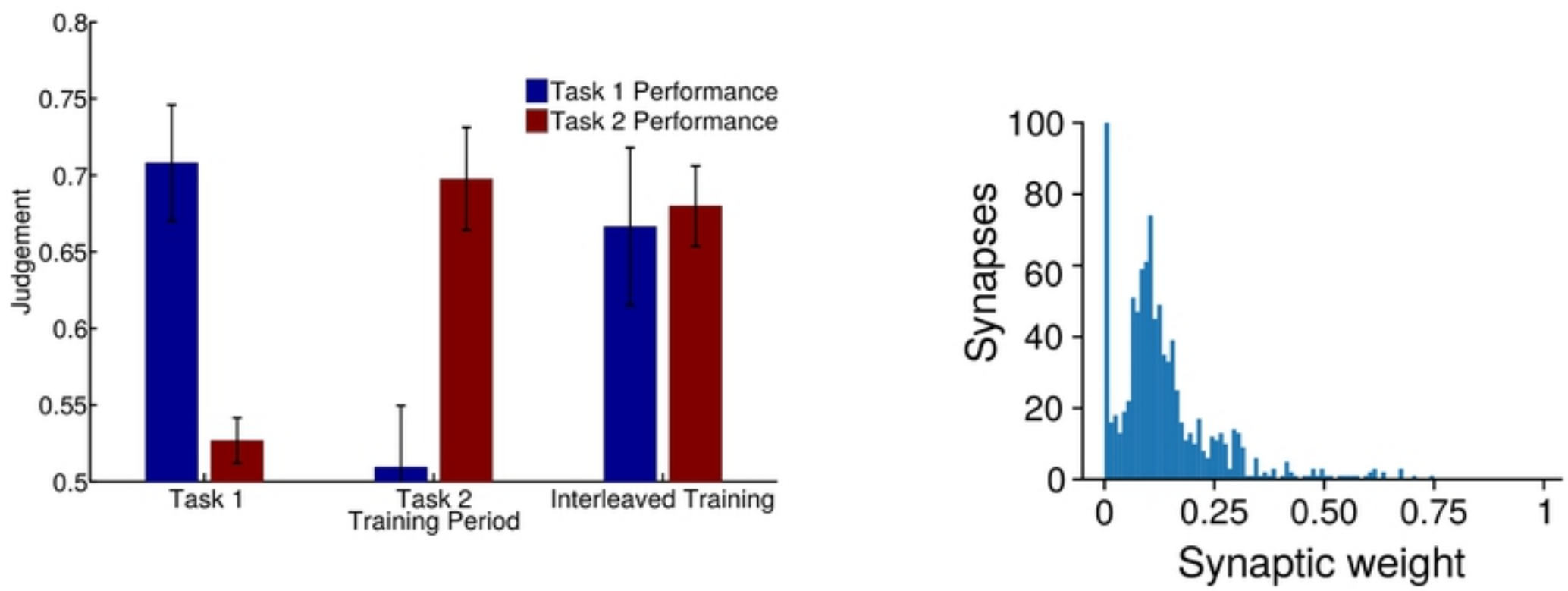

B
Task 1 Performance
Task 2 Performance
Judgement
0.8
0.75
0.7
0.65
0.6
0.55
0.5
Task 1
Task 2
Training Period
Interleaved Training
Synapses
100
80
60
40
20
0
0
0.25
0.50
0.75
1
Synaptic weight

Figure 5

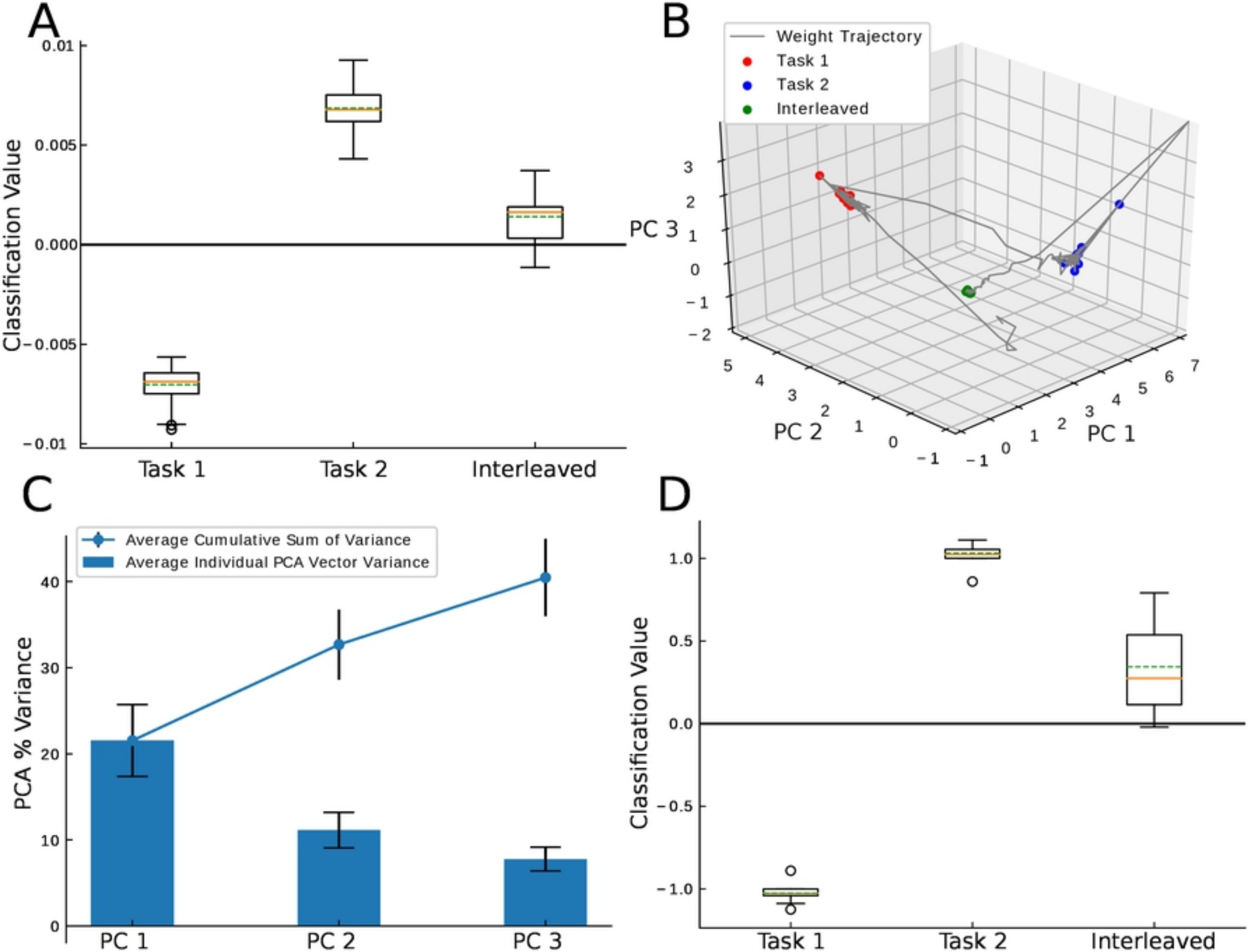

A
Classification Value
0.01
0.005
0.000
−0.005
−0.01
Task 1
Task 2
Interleaved
B
Weight Trajectory
Task 1
Task 2
Interleaved
PC 3
PC 2
PC 1
C
Average Cumulative Sum of Variance
Average Individual PCA Vector Variance
PCA % Variance
PC 1
PC 2
PC 3
D
Classification Value
1.0
0.5
0.0
−0.5
−1.0
Task 1
Task 2
Interleaved

Figure 7

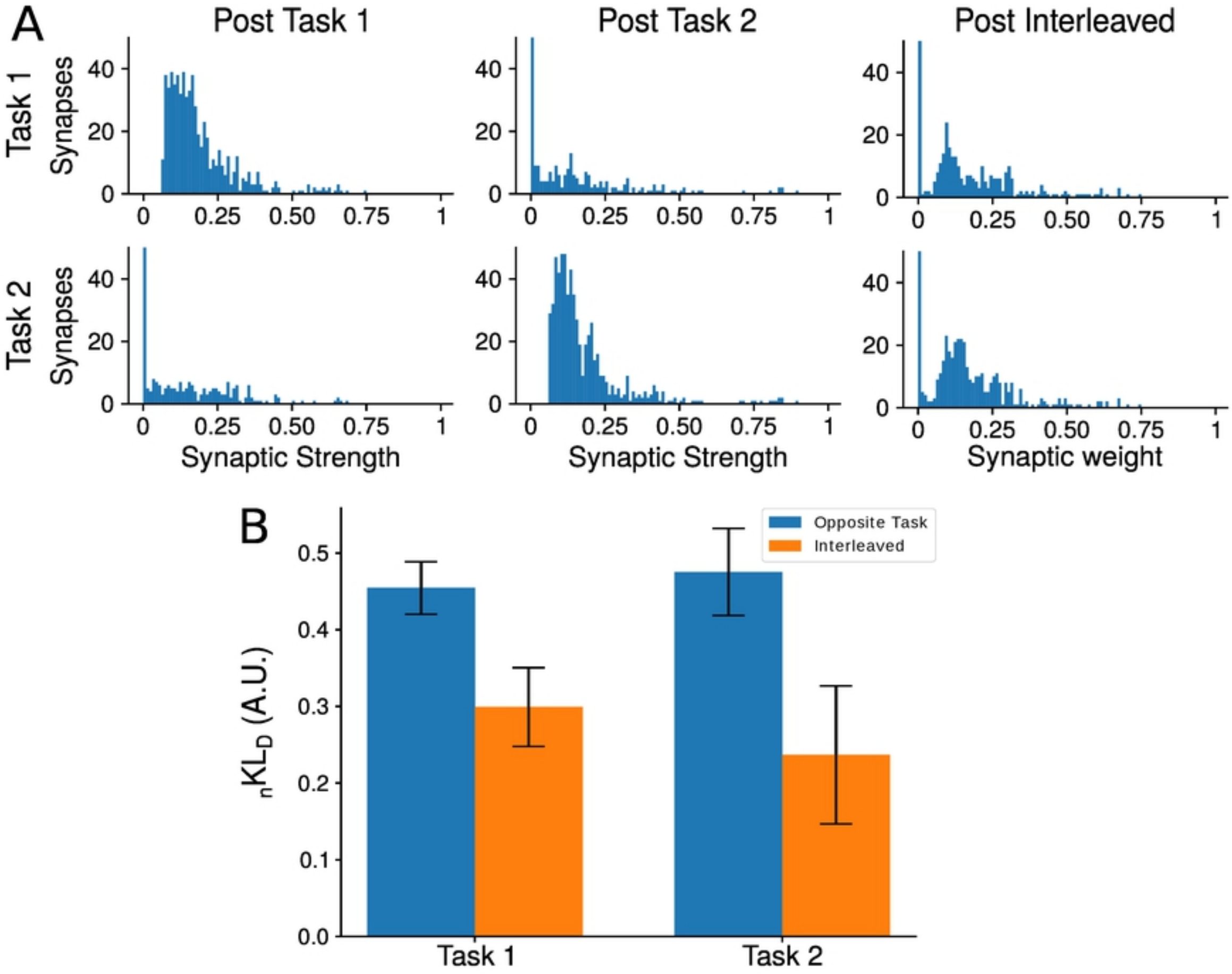

A
Post Task 1
Post Task 2
Post Interleaved
Task 1
Task 2
Synapses
Synaptic Strength
Synaptic weight
B
Opposite Task
Interleaved
$_nKL_D$ (A.U.)
Task 1
Task 2

Figure 6