Novel Memory Plasticity and Readout Layers for Continual Scene Detection

Ang Li
Columbia University
New York
{al4263}@columbia.edu

Abstract

In this paper, we introduce novel memory plasticity mechanisms called hybrid memory plasticity that can combine any sequences of multiplicative and additive plasticities, alongside innovative readout layers for effective memory storage retrieval, for addressing tasks that require temporal integration and classification. Our approach is applied to a continual scene detection task, where each scene comprises multiple frames requiring both memory integration and accurate classification. We have shown that our new biologically plausible neural network achieves SoTA result on the task we defined. Furthermore, we conduct a detailed neural network interpretability analysis to elucidate the mechanisms underlying the proposed memory plasticities. This framework provides valuable insights into biologically inspired architectures and their applications in continual learning, offering a deeper understanding of neuroscience principles.

1 Introduction

Understanding and recognizing sequences of patterns in dynamic environments is a fundamental capability of the human brain. Many everyday tasks, such as determining whether a sequence of events has been encountered before, rely on the ability to integrate temporal information and classify familiarity. This capability is rooted in synaptic plasticity—the brain's dynamic ability to strengthen or weaken synaptic connections in response to neural activity. Through this remarkable adaptability, humans can efficiently learn from experience, recall past events, and generalize across similar situations.

In recent years, theoretical neuroscience have sought to replicate these capabilities from a computational perspective. Synaptic plasticity has been studied extensively, with various forms of plasticity demonstrating unique roles in memory, learning, and decision-making. Among these, additive plasticity, such as Hebbian and anti-Hebbian mechanisms, has shown impressive performance in pattern recognition and classification tasks by dynamically adjusting synaptic weights to capture input-output relationships. For instance, Tyulmankov et al. (2022) demonstrated how anti-Hebbian plasticity could be used for continual familiarity detection tasks, with just a single linear readout layer.

Complementing additive approaches, multiplicative plasticity has emerged as a promising mechanism for integrating temporal information. Unlike traditional methods that rely on recurrent architectures, multiplicative plasticity enables the efficient encoding of sequential patterns through weight modulations driven by activity-dependent scaling. Aitken and Mihalas (2023) introduced a neural model leveraging multiplicative plasticity, showcasing its ability to integrate temporal dependencies without requiring explicit recurrent structures.

While additive plasticity is effective for precise classification and multiplicative plasticity excels at temporal integration, combining these mechanisms is not as straightforward as it might seem. Their interaction introduces challenges in balancing memory integration and feature discrimination,

necessitating the development of advanced readout architectures to fully harness their complementary strengths.

In this work, we propose a novel framework that combines multiplicative and additive plasticity for the continual scene detection task. To address the complexities of integrating these plasticity mechanisms, we design sophisticated readout architectures that enhance the network's ability to effectively retrieve relevant information in the plasticity matrix. Our contributions are as follows:

1. Introduced various forms of hybrid plasticity that combines any sequences of multiplicative and additive mechanisms for temporal integration and classification.
2. Developed advanced readout methods for accurate retrieval in this hybrid plasticity matrix.
3. Demonstrated the effectiveness of our framework on the continual scene detection task, achieving SoTA performance in retrieval accuracy and maximal retrieval length.
4. Conducted an interpretability analysis to uncover the underlying mechanisms of this novel memory plasticity.

2 Problem Formulation

We propose a problem called continual scene detection, which requires the integration of temporal information, classification, and retrieval to achieve high performance. The task challenges a model to differentiate between familiar and novel scenes in a dynamic, evolving sequence. Specifically:

A **frame** $\boldsymbol{f}$ is defined as a vector in $\mathbb{R}^n$, where each component is sampled from a Bernoulli distribution with probability $p = 0.5$, resulting in possible outcomes from the set $\{-1, 1\}$.

A **scene** s consists of $n = 4$ frames. The first frame $\boldsymbol{f_1}$ is generated randomly, and the subsequent frames are derived from $\boldsymbol{f_1}$ with a variation rate of 0.1. Specifically, this means that 10% of the components of $\boldsymbol{f_1}$ are randomly flipped to create each subsequent frame, introducing controlled variability.

We continuously generate a stream of scenes over time. At a given time point $t = R$, a decision is made to either generate a new scene or reuse a previous one to create a **familiar scene**. If a familiar scene is chosen, it is created by randomly permuting the scene from time $t = T - R$. However, if a familiar scene is generated at time $t = T$, it will not be generated again at $t = T + R$ to prevent repetition beyond a certain point.

The **objective** is to continuously determine whether a presented scene is **familiar** (previously seen with some variation) or **novel** (newly generated).

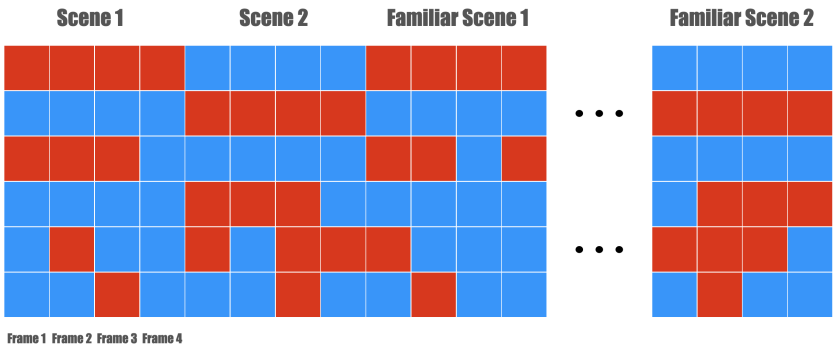


Figure 1: Caption of the graph

3 Novel Memory Plasticity

Memory networks require mechanisms to store and update information over time. We propose three distinct types of plasticity mechanisms that can be incorporated into a fully connected neural network

layer: additive plasticity, multiplicative plasticity, and a stacked approach. These mechanisms determine how the network's weights change in response to incoming stimuli.

3.1 Basic Architecture

The Memory Plasticity Layer consists of a standard weight matrix $\boldsymbol{W_1}$ and a dynamic plasticity matrix $\boldsymbol{P_1}(t)$ that changes over time. The layer receives input vector $\boldsymbol{x}(t)$ at each timestep t and produces hidden layer activations $\boldsymbol{h_1}(t)$ through one of three plasticity mechanisms.

3.2 Plasticity Mechanisms

3.2.1 Multiplicative Plasticity (M)

In multiplicative plasticity, the plastic component modulates the standard weights through element-wise multiplication:

$$\boldsymbol{h_1}(t) = \phi\left(\left(\boldsymbol{W_1} \cdot \boldsymbol{P_1}(t)\right)\boldsymbol{x}(t) + \boldsymbol{W_1}\boldsymbol{x}(t) + \boldsymbol{b_1}\right) \tag{1}$$

This can be interpreted as the plasticity matrix scaling the effectiveness of each synapse while preserving its sign.

3.2.2 Additive Plasticity (A)

Additive plasticity directly adds the plastic component to the standard weights:

$$\boldsymbol{h_1}(t) = \phi\left(\left(\boldsymbol{W_1} + \boldsymbol{P_1}(t)\right)\boldsymbol{x}(t) + \boldsymbol{b_1}\right) \tag{2}$$

This mechanism allows the plasticity to modify both the magnitude and sign of the effective weights.

3.2.3 Stacked Plasticity (Stack)

The stacked approach combines both mechanisms by splitting the hidden layer into two parts:

- Upper half: Uses multiplicative plasticity
- Lower half: Uses additive plasticity

This approach allows the network to leverage the benefits of both plasticity types.

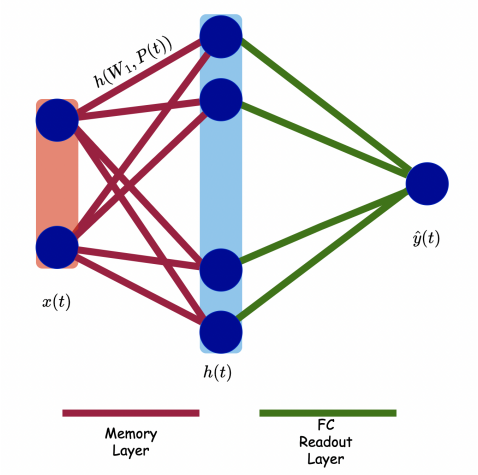
3.3 Plasticity Matrix Update Rule

The plasticity matrix $\boldsymbol{P_1}(t)$ is updated continuously during training and inference using the following rule:

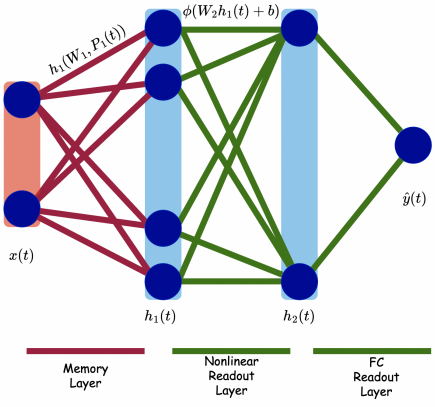
$$\boldsymbol{P_1}(t) = \lambda \boldsymbol{P_1}(t-1) + \boldsymbol{\eta} \odot \left(\boldsymbol{h_1}(t)\boldsymbol{x}(t)^T\right) \tag{3}$$

where:

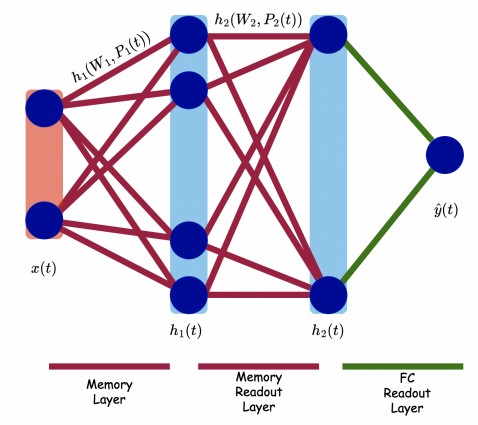
- λ is the decay factor controlling the retention of prior plasticity updates.
- $\odot$ represents element-wise multiplication.
- $\boldsymbol{\eta}$ determines the learning rates for plasticity and can take three forms:
  - **Scalar η:** A uniform scalar controlling the overall plasticity rate.
  - **Neuron-wise $\boldsymbol{\eta}$:** A vector matching the dimensions of $\boldsymbol{h_1}(t)$, where each element controls plasticity for a specific neuron.
  - **Synaptic-wise $\boldsymbol{\eta}$:** A matrix identical in size to $\boldsymbol{h_1}(t)\boldsymbol{x}(t)^T$, allowing per-synapse control of learning rates.



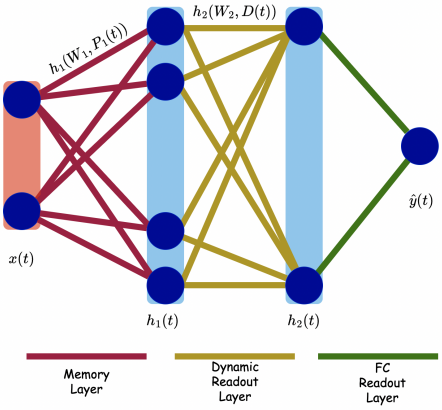
(a) Fully Connected Readout



(b) Nonlinear Readout



(c) Memory Based Readout



(d) Dynamic Read Out

Figure 2: The four types of readouts

4 Readout Layer

For the readout layer, we present four different versions: Fully Connected readout, Nonlinear readout, Memory Based readout, and Dynamic readout:

- **Fully Connected readout (FC):** This version is a plain fully connected layer, represented as follows:

$$\sigma\left(\boldsymbol{W_f h_1}(t) + \boldsymbol{b_f}\right) \tag{4}$$

- **Nonlinear readout (Nonl):** This version transforms $\boldsymbol{h_1}(t)$ with a nonlinear layer before a FC transformation:

$$\sigma\left(\boldsymbol{W_2 h_1}(t) + \boldsymbol{b_2}\right), \tag{5}$$

$$\sigma\left(\boldsymbol{W_f h_2}(t) + \boldsymbol{b_f}\right) \tag{6}$$

- **Memory Based readout (Memo):** This version leverages the benefit of additional layers of memory for improved performance. This transformation, identical to the one performed by the first memory layer, is denoted by $h_2 = h(\boldsymbol{W_2}, \boldsymbol{P_2}(t))$. Subsequently, we apply the same FC layer transformation:

$$\sigma\left(\boldsymbol{W_f h_2}(t) + \boldsymbol{b_f}\right) \tag{7}$$

- **Dynamic readout (Dyn):** This version is inspired by the observation that the decay rate λ in **Memo** is extremely low, which signifies minimal memory retention. **Dyn** addresses this issue by modifying the update rule for the plasticity matrix as follows:

$$\boldsymbol{P}(t) = \boldsymbol{\eta} \odot \left(\boldsymbol{h_1}(t)\boldsymbol{x}(t)^T\right) \tag{8}$$

This change results in a dynamic readout of the current time step. Apart from this modification, everything else in **Dyn** mirrors **Memo**. The dynamic transition is denoted by $h_2 = h(\boldsymbol{W_2}, \boldsymbol{D}(t))$. Following this, the same FC layer transformation is applied:

$$\sigma\left(\boldsymbol{W_f}\boldsymbol{h_2}(t) + \boldsymbol{b_f}\right) \tag{9}$$

5 Meta-Learning and Curriculum Training

Meta-learning Tyulmankov et al. (2021) enables our memory network to optimize not only for specific tasks but also for mechanisms such as synaptic plasticity rules and network architectures. Specifically, Meta-learned plasticity rules, such as anti-Hebbian updates, improve memory retention and capacity. Specifically, we will try to meta learn the following parameters:

- Static weight matrices W and Bias vectors b
- Synaptic decay rate λ
- Plasticity rate η

Where those parameter helps us to do any type of continual familiar scene detection tasks by updating the plasticity matrix P.

5.1 Curriculum Training

Following Tyulmankov et al. (2022), we leverage a curriculum training algorithm that progressively introduces increasing levels of task difficulty, improving model convergence for tasks requiring longer memory retention. In the case of continual familiarity detection, the repeat interval R is incrementally increased during training until the network stabilizes.

Algorithm 1 Curriculum Training

Input: Initial repeat interval R_0, maximum iterations T, accuracy threshold θ

```
Initialize network parameters and R ← R_0
while training not converged do
    Train on dataset with current R for T iterations
    if accuracy ≥ θ then
        R ← R + 1
    else
        Break
    end if
end while
return Optimized network parameters
```

The maximum capacity R_{max} is defined as the largest value of R for which the familiarity detection accuracy remains above 99%.

6 Results

6.1 One Layer Memory Plasticity

To maintain an equitable comparison, we constrain the number of dynamic neurons to be equivalent across all models. In our investigation, we considered models with dynamic neuron counts of 50 and 100. For both, we have just one layer of memory plasticity and one layer of read out. Their respective performance metrics are detailed in the table below:

Analyzing the table, we make the following observation:

Max R													
Size \ Net	M						A						Stack
	FC	Nonl	Dyn		Memo		FC	Nonl	Dyn		Memo		FC
			M	A	M	A			M	A	M	A	
50	5	7	9	9	8	9	6	4	4	4	4	4	6
100	9	10	12	13	9	14	9	7	5	5	5	5	7

Table 1: Performance metrics for different dynamic neuron sizes

1. Across most types of readouts, **M** outperforms **A**.
2. **M** benefits from more intricate readouts and scaling. Conversely, the performance of **A** diminishes with more complex readouts.
3. The most noteworthy performance is achieved by the memory layer of **M**, combined with either **Dyn** or **Memo** readout layers.
4. Our methods show a 55.6% increasement in max retrieval rate compare to previous SoTA on average.
5. For scaling behavior: our method scales 66.7% faster than networks with only additive plasticity and 25% faster than those with only multiplicative plasticity.

A graph of performance is here:

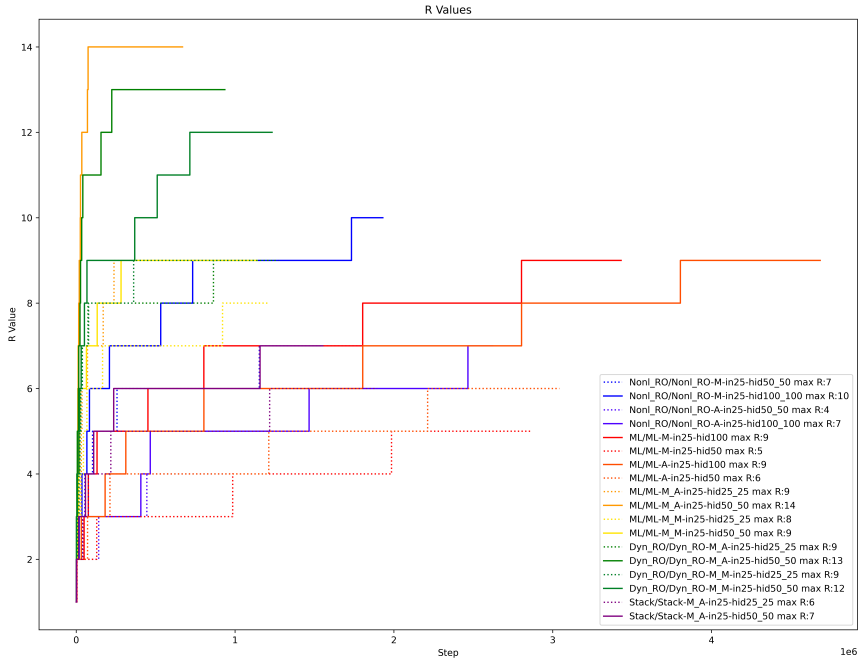


Figure 3: Performance across all networks and sizes

6.2 Multi Layer Memory plasticity

Here we study the effect of multi layer memory plasticity.

Network Name	neurons per layer	Rmax
MA	25, 25	7
MM	25, 25	6
MAA	17, 17, 17	2
MAM	17, 17, 17	5
MMA	17, 17, 17	4
MMM	17, 17, 17	3
MMMA	13, 13, 13, 13	2
MMMA	20, 15, 10, 5	4

Table 2: neurons = 50

Network Name	neurons per layer	Rmax
MA	60, 60	12
MAA	40, 40, 40	5
MAM	40, 40, 40	5
MMA	40, 40, 40	10
MMM	40, 40, 40	6
MMMA	30, 30, 30, 30	8
MMMA	40, 30, 20, 10	6

Table 3: neurons = 120

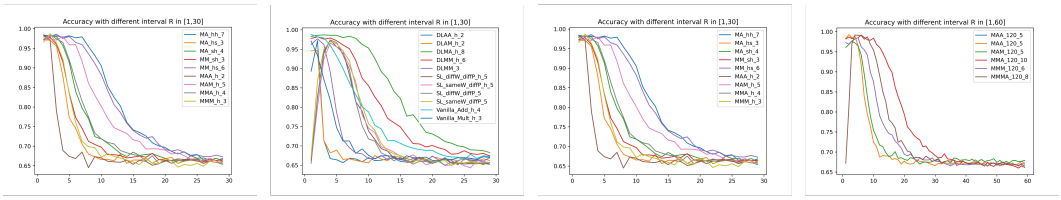


Figure 4: Accuracy rate with different R

We see that stacking the layer naively does not give us better results.

7 Interpreting the neural network

7.1 Memory Network

In this section, we address the possible reasons why multiplicative plasticity matrix has a better performance. From 5, we can see that a complex readout hurts the performance of A networks, whereas a complex readout helps the performance of M networks.

To figure out why the network works, we plot the hidden layer activities across time, the eta matrix, the weight matrix, the actual weight/plasticity across time. Specifically, we choose the **Memo** Network with **M** memory and **A** readout here67.

We plot the actual weight and the plasticity matrix with t from 74-77, where at 74,75 the network output 0 and 76,77 the network output 189

Notice that the **M** is dominated by several elements and rarely changes. So a hypothesis would be that most of the identification work were done by the dynamic readout, whereas **M** merely serves as a memory source.

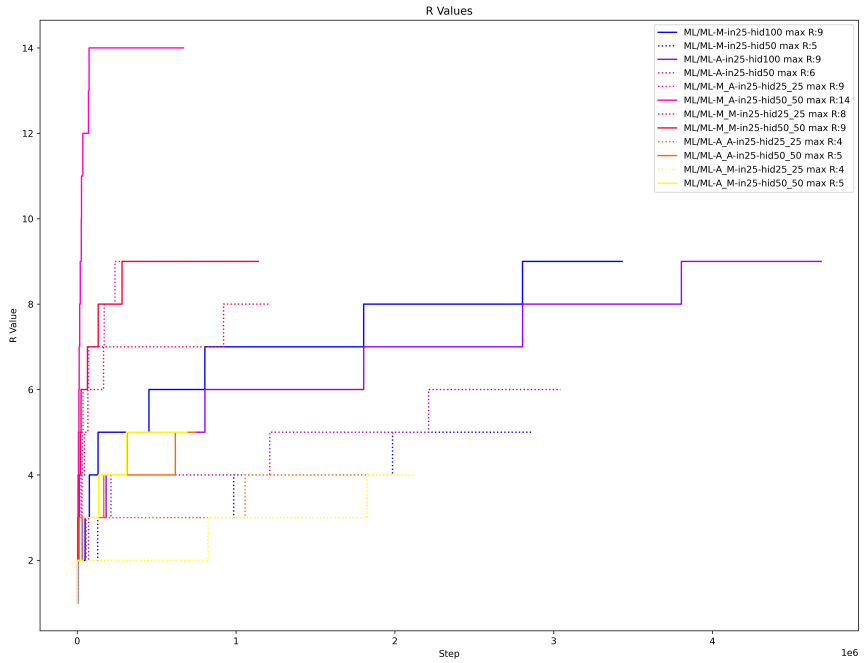


Figure 5: M vs A

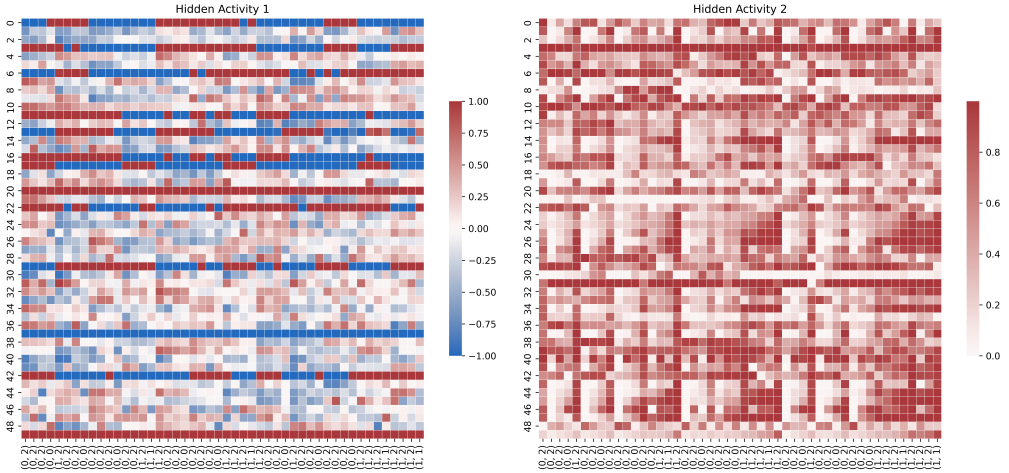
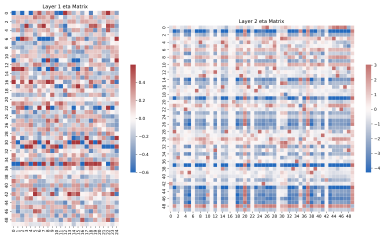


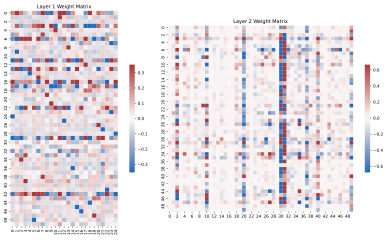
Figure 6: Hidden Layers Across Time

7.2 Plasticity Rate Analysis

As we can see in 4, some of the plasticity rate got decayed to 0, which means they are trying to forget rather than to memorize as much as possible, which is counter intuitive. However, taking a closer look, we find out that most of the diminishing rate are coming from additive plasticity matrix where multiplicative plasticity matrix with closing to 1 retention rate. This further confirms our hypothesis that multiplicative matrix is handling all the memory perservation where as the additive matrix is simply performing a readout role.

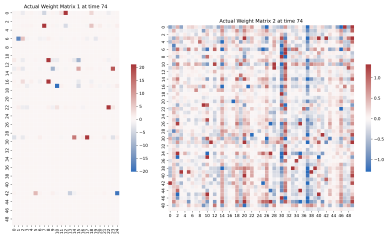


(a) eta matrix of **Memo M A** 50/50

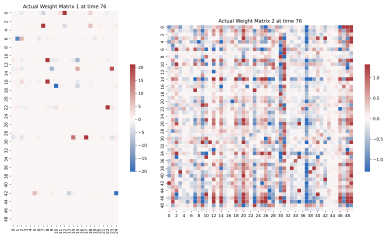


(b) weight matrix of **Memo M A** 50/50

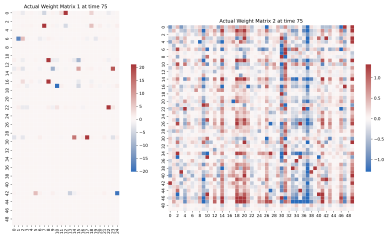
Figure 7: Dyn RO MA 50/50



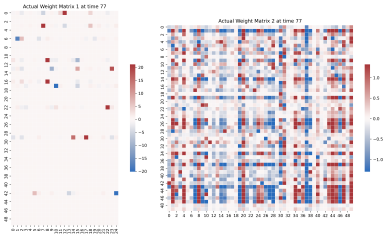
(a) T=74



(c) T=76

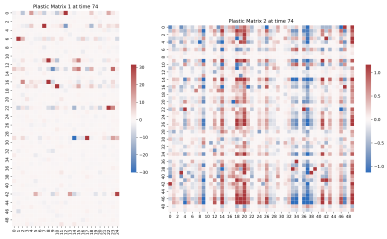


(b) T=75

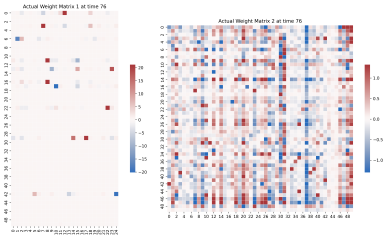


(d) T=77

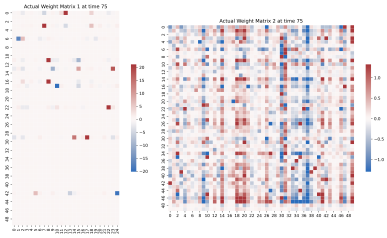
Figure 8: Actual weight



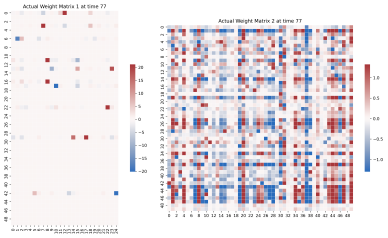
(a) T=74



(c) T=76



(b) T=75



(d) T=77

Figure 9: Plasticity

References

Aitken, K. and Mihalas, S. (2023). Neural population dynamics of computing with synaptic modulations. *Elife*, 12:e83035.

Network Name	**neurons per layer**	λ_1	λ_2	λ_3	λ_4	**Rmax**
1LA	50	0.97	x	x	x	4
1LM	50	0.95	x	x	x	3
MAhh	25, 25	0.99	0.19	x	x	7
MA	60, 60	0.99	0.23	x	x	12
MMhh	25, 25	0.99	0.0039	x	x	6
MMss	25, 25	0.99	0.16	x	x	3
AA	25, 25	0.84	0.79	x	x	2
AM	25, 25	0.81	0.86	x	x	2
MAA	17, 17, 17	0.88	0.71	0.51	x	2
MAA	40, 40, 40	0.96	0.74	0.33	x	5
MAM	17, 17, 17	0.98	0.25	0.97	x	5
MAM	40, 40, 40	0.96	0.6	0.46	x	5
MMA	17, 17, 17	0.98	0.97	0.09	x	4
MMA	40, 40, 40	0.99	0.99	0.14	x	10
MMM	17, 17, 17	0.93	0.96	0.26	x	3
MMM	40, 40, 40	0.97	0.97	0.271	x	6
MMMA	13, 13, 13, 13	0.86	0.86	0.40	0.90	2
MMMA	20, 15, 10, 5	0.97	0.82	0.18	0.98	4
MMMA	30, 30, 30, 30	0.99	0.89	0.98	0.16	8

Table 4: Plasticity rate analysis

Tyulmankov, D., Fang, C., Vadaparty, A., and Yang, G. R. (2021). Biological learning in key-value memory networks. *Advances in Neural Information Processing Systems*, 34:22247–22258.

Tyulmankov, D., Yang, G. R., and Abbott, L. (2022). Meta-learning synaptic plasticity and memory addressing for continual familiarity detection. *Neuron*, 110(3):544–557.