

<https://doi.org/10.1038/s42003-024-07282-3>

Recurrent neural networks with transient trajectory explain working memory encoding mechanisms

Check for updates

Chenghao Liu^{1,5}, Shuncheng Jia^{1,2,5}, Hongxing Liu^{1,2,5}, Xuanle Zhao^{1,2}, Chengyu T. Li³, Bo Xu^{1,2}✉ & Tielin Zhang^{1,2,4}✉

Whether working memory (WM) is encoded by persistent activity using attractors or by dynamic activity using transient trajectories has been debated for decades in both experimental and modeling studies, and a consensus has not been reached. Even though many recurrent neural networks (RNNs) have been proposed to simulate WM, most networks are designed to match respective experimental observations and show either transient or persistent activities. Those few which consider networks with both activity patterns have not attempted to directly compare their memory capabilities. In this study, we build transient-trajectory-based RNNs (TRNNs) and compare them to vanilla RNNs with more persistent activities. The TRNN incorporates biologically plausible modifications, including self-inhibition, sparse connection and hierarchical topology. Besides activity patterns resembling animal recordings and retained versatility to variable encoding time, TRNNs show better performance in delayed choice and spatial memory reinforcement learning tasks. Therefore, this study provides evidence supporting the transient activity theory to explain the WM mechanism from the model designing point of view.

Working memory (WM) is an important cognitive ability that allows animals to hold limited key information to be used in the near future. It is essential for many other cognitive abilities, including learning, reasoning, and decision-making¹. One particularly interesting question about working memory is how the short-term memory content is maintained after the sensory stimulus ends. Early biological experiments showed persistent neuronal activities during delay periods^{2,3}, which have inspired recurrent neural networks (RNNs) that support context-dependent output⁴ to simulate attractor-type WM, attributing the memory maintenance to the attractor state of the network dynamics^{5,6}.

However, further studies found transient-trajectory type dynamics in delay periods of biological neurons, and the number of these transient neurons is surprisingly larger than that of the attractor-type neurons^{7–10}. Therefore, various alternative theories of WM maintenance have been proposed. Some proposed that population coding through stable subspace states is important for static WM maintenance with varying neuronal activities^{11,12}. Some other studies proposed that chaotic network activities could keep the memories of different stimuli separable over time^{13,14}. Another prominent theory proposed that oscillation and phase coding encode short-term memories^{15–17}. Besides these activity-based memory

encoding theories, activity-silent states through short-term synaptic plasticity were also hypothesized to be able to hold WM^{18–21}. Another line of research observed that transient activities of individual neurons could tile together to span the entire delay period^{22–25}, supported by some related theoretical studies indicating that the transient trajectory is a special form of heteroclinic orbits between saddle fixed points to encode stimuli and last beyond the stimulation duration^{26–28}.

At the same time, more RNNs were trained in working memory tasks and showed transient activities^{14,29–33}. Some studies observed transient activities in trained RNNs and thus they focused on studying how the memory is encoded. Some found it encoded with robust trajectories²⁹ while others found it encoded with amplitude of the transient oscillation³¹. Other studies focused on how to replicate the transient activity patterns from recordings. Barak et al. compared three types of models and showed both reservoir network and the partially trained RNN could match some features of the data¹⁴. Rajan et al. found that the neural activity pattern could take the form of a line attractor or transient trajectory, depending on the connection parameters of the circuit mechanism³². Orhan et al. showed that the change between these two patterns could be continuous in RNNs depending on the network parameters and tasks³³. However, whether keeping memory with

¹Institute of Automation, Chinese Academy of Sciences, Beijing, China. ²School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China.

³Lingang Laboratory, Shanghai, China. ⁴Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai, China.

⁵These authors contributed equally: Chenghao Liu, Shuncheng Jia, Hongxing Liu. ✉e-mail: xubo@ia.ac.cn; zhangtielin@ion.ac.cn

transient trajectory or persistent activity provides better performance in working memory tasks is still an open question, because few studies have simulated WM task to discuss their capabilities such as memory capacity and energy efficiency.

Here we based our tests and model design on a previous mouse working memory study, where neural activities were recorded from mice performing the olfactory delayed paired association task (ODPA). Brain-wide activity waves were found to encode transient-type WM patterns, where sequentially coactivated chains and loops of neurons were identified, and coupled spikes were hierarchically organized³⁴.

To build models to represent the transient trajectory encoding with adjustable configurations that control the activity patterns, we modified the vanilla RNN to enforce transient activities. The resulting transient RNN (TRNN) contains self-inhibiting transient neurons, sparse connection, and hierarchical topology matching the sensory-association-motor connectome. The self-inhibition modification is similar to the negative feedback design based on the spike-frequency adaptation (SFA) mechanism that was used to induce traveling waves in computational models^{35–37}. This mechanism implemented in the form of adaptive firing thresholds is also introduced to a spiking RNN to enhance its performance in sequence tasks by encoding the memory in the thresholds^{38–40}.

Unlike previous RNN modeling studies, we kept the vanilla RNN and the TRNN in separated groups in our analyses and combined the results that match the modeling data to the experimental recordings and the results that show performance advantages of transient trajectory encoding models. We first analyzed the activities patterns of the vanilla RNN and TRNN and found the TRNN matched the transient activity from animal recordings better qualitatively, while the vanilla RNN had more persistent activity. This makes them good representative models to compare the two working memory mechanism theories. Then we showed that even with transient trajectory encoding, TRNN is still versatile in terms of memory time. The activity dynamics were also analyzed in the form of low-dimensional trajectories to help explain how memory is encoded. Further analysis of the activities showed higher information richness and lower average squared activity of TRNN compared to a vanilla RNN using persistently firing neurons, suggesting a higher memory capacity and lower energy cost. Then we compared the size-matched networks in a range of simple WM tasks that simulated animal experiments, including delayed choice experiments and spatial memory experiments⁴¹, using reinforcement learning⁴², and got better performances with TRNNs. The closer activity patterns to animal recordings and better performances of TRNNs support transient trajectory as a working memory encoding mechanism.

Results

Reproduce transient trajectory in neural activity pattern with TRNNs

In the previous ODPA experiment (Fig. 1a), extracellular Neuropixel recordings were done in over 30 mouse brain regions (Fig. 1b). A total of 33,208 neurons were collected from 113 recording sessions³⁴. To study the transient trajectory observed in animal working memory experiments using computational models, we designed stimulated ODPA experiment as a task to train and test the models. Briefly, the ODPA task is composed of fixation period, a sample period, a delay period, a test period and a response period. The animal is rewarded for licking during the response period when the odor received during the sample period and the test period is a matching pair. Since the animal's action would not change the stimuli, the experiment was simulated with fixed sequences of input-output pairs as a supervised learning task (see "Methods" for more details about the animal experiment and the simulated versions of the task).

We started with vanilla RNNs that have been commonly used in working memory modeling studies. Examining the details of previous studies showed that different kinds of network architectures, such as reservoir network and vanilla RNN, had been used. The studies also used different network sizes, neuronal nonlinearity and optimizing methods (Supplementary Table 1)^{14,29,30,32,33,43,44}. We chose a vanilla RNN architecture

with ReLU nonlinearity and the backpropagation for optimization. To make the network more biologically plausible, we also initialized its recurrent connection weights according to a 4:1 composition ratio of excitatory and inhibitory neurons⁴⁵. Besides, we added regularization to the average neuronal activity during training to keep the neuronal activities stable. In practice, these two features were also important for the training of the models. When the neuronal activities were plotted, variability between models with different initializations could be observed (Fig. 1e, Supplementary Fig. 1b), as shown in the previous study³³.

To compare these to neuronal activities from animal experiments, we analyzed data shared with us by our collaborators collected in their recent work³⁴. The activities also have some variance between recording sessions (Fig. 1d, Supplementary Fig. 1a). The results from the animal experiments seem to have a clearer sequence of transient activities during the delay periods than those of the RNNs. To quantify this difference, we proposed the transient index (TI, Fig. 1c), which extended the previous sequential index³³ and contained three key components. The first component was the synchrony of the firing rate peaks of participating neurons measured as Shannon entropy (peak-firing-time entropy). The time of the firing rate peak was treated as a random variable and each neuron is a sample. The Shannon entropy measured how variable this peak time was and thus measured how synchronized the samples/neurons were. The second component, ridge-to-background ratio, described the persistency of the neuronal firing by measuring how sharp the firing rate peak was for each neuron²⁴. Neurons with sharper peaks were less persistent. The third component was the proportion of memory-related peak firing and measured the proportion of transient trajectory encoding neurons whose peak activities are in the delay period. This ensured that the firing rate peaks we examined were in the delay period and likely encoding working memories. Finally, three components were summed up to TI, which should be high for neuron activity peaks that tile and overlap the delay period. We found that RNNs indeed had much lower TIs than the animal neural networks (Fig. 1h).

To increase the TI and better replicate the transient dynamics of in vivo working memory encoding, we introduced a transient encoding modification to the basic model to get the TRNN (Fig. 1g). The transient encoding modification is composed of three parts. First, at the neuronal scale, it endowed the neuron with an additional self-inhibition input proportional to its past activity. Since connectivity constraint is another common way to modify RNNs, sparse connections were enforced at the network-connection scale⁴⁵. This was proposed because persistent firing in attractor models relied on strong recurrent connections^{5,6}. Third, the RNN was divided into three functional regions, sensory, association, and motor regions, congruent to the pathway found in natural neural networks³⁴. Inter-region connections were sparser than intra-region connections, based on the finding that there are more spike coupling events within brain regions than between³⁴. Compared to sparsity enforcement, self-inhibition targeted neurons with high activities more specifically and could be more effective at stopping persistent firing in neurons with slow dynamics. As expected, the activity dynamics of the TRNNs resemble that of the animal neural networks, and the TIs of the TRNNs are significantly higher than the RNNs (Fig. 1f, h). Since TRNN and RNN have the same number of learnable parameters but are different in activity patterns during memory encoding, in the following experiments, we could compare TRNNs to RNNs to study the advantages of the transient encoding mechanism.

The TRNN's resemblance to the experimental results was further analyzed. We firstly focused on odor-selective neurons that exhibited statistically-significant selection to specific odors (Fig. 2a, "Methods")³⁴. The proportion of stimulus selective neurons in the transient RNN reached a proportion of 38.2% (Fig. 2b), close to the statistical result of biological experiments (32.6%, Fig. 2c). Similar to the observation in the brain recording experiments, we found neurons that were selective to one of the odors at different phases during the delay periods (Supplementary Fig. 2a). We ranked all the neurons based on the time of their peak response differences in the two odors, plotted the differences in their firing rate

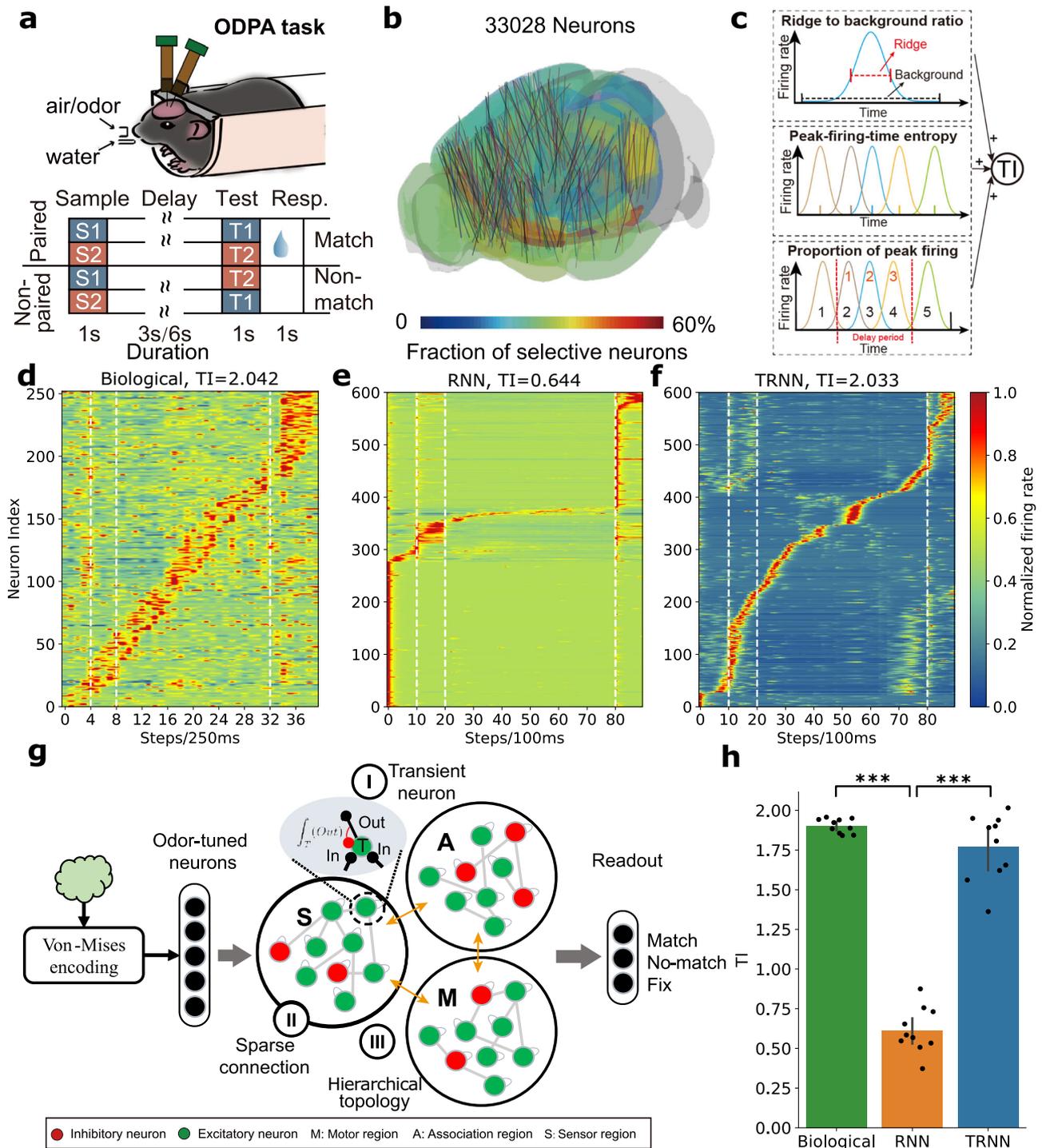
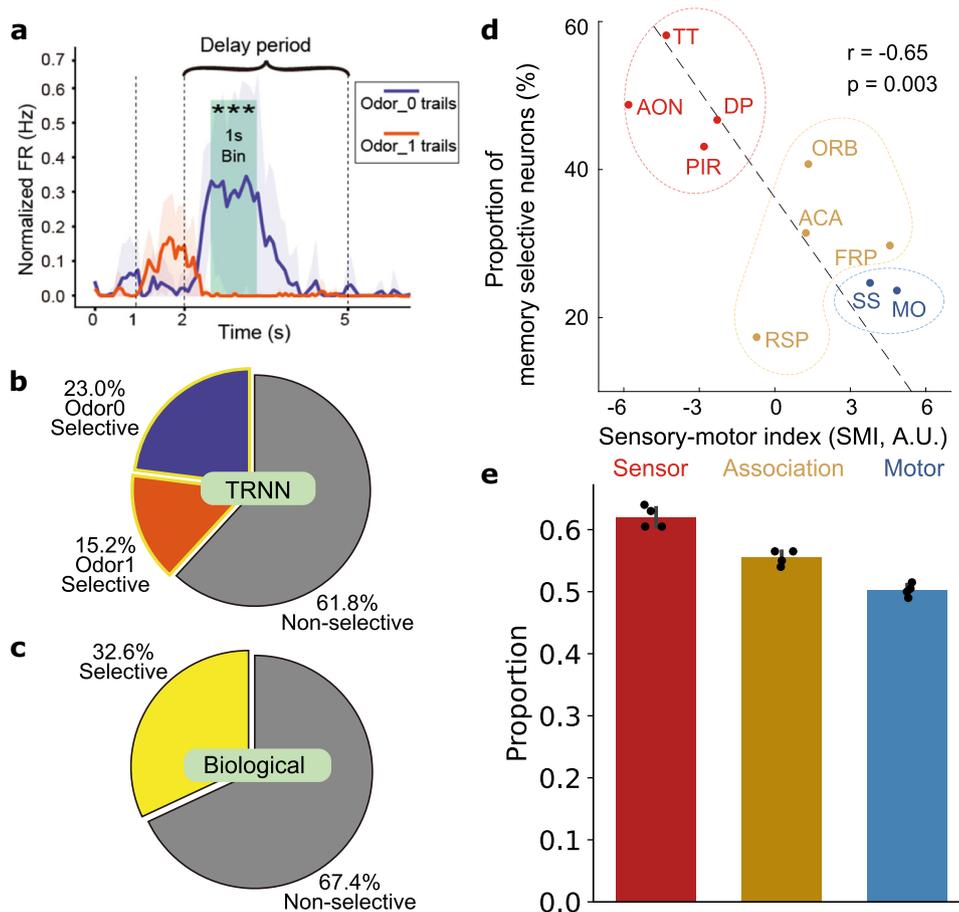


Fig. 1 | Transient activities examined in animal experiments and computational model. **a** This schematic diagram depicts the ODDPA task. **b** The visualization of Neuropixel probes in many brain regions to record large-scale neuronal activities. **a, b** are adapted from Huang et al. with permission³¹. **c** The transient index (TI) as the quantitative indicator for transient trajectory encoding is composed of three factors: the ridge-to-background ratio measures the temporal localization of the cell-assembly activity during delay; the peak-firing-time entropy measures the uniformity of the peak response times of the cell assembly in a given trial; the proportion of memory-related peak firing measures the number of neurons participating in encoding during the delay period. **d, e, f** Heatmap of activities of neurons during the ODDPA task from an example trial recorded in animal experiments (**d**), an RNN (**e**), and a TRNN (**f**). Only memory-selective neurons in the

animals were included. The neurons were ordered based on their activity peak times. The dashed vertical lines indicate the border between the four periods in the task. **g** The biology-inspired TRNN, including transient neurons with self-inhibition, sparse connections between neurons, and hierarchical topology between brain regions (orange arrows). **h** TIs of neural activities from animals and the two RNN models (mean \pm SEM). The biological group includes average TIs of correct trials in 14 sessions with over 85% correct trials. The TRNN or RNN groups include average TIs of 10 trained models each. The RNNs have significantly lower TIs than the animals (Student *t* test, $t = -30.00$, $p = 2.42e-19$) and the TRNNs (Student *t* test, $t = -14.34$, $p = 2.74e-11$). The difference between the animals and the TRNNs is not significant (Student *t* test, $t = -1.99$, $p = 0.059$).

Fig. 2 | Comparison of biological discovery and artificial modeling. **a** The activity of sample odor-selective neurons in response to stimulation of different odors (mean \pm SEM). The green shade labels the time bin highlighting the averaged activities significantly differ between trials with the two stimuli. **b** Composition of neurons in the TRNN. **c** Composition of neurons in animals doing the ODPa task. **d** Proportion of selective neurons in brain regions, including anterior olfactory nucleus (AON), taenia tecta (TT), dorsal peduncular area (DP), piriform area (PIR), orbital area (ORB), anterior cingulate area (ACA), frontal pole (FRP), retrosplenial area (RSP), somatosensory area (SS) and motor area (MO). **e** Proportion of selective neurons in different neuron groups in the TRNN models (mean \pm SEM).



(Supplementary Fig. 2b), and found that the selectivity also has a transient trajectory pattern. It was found in the biological experiment that the proportion of selective neurons has a negative correlation with the sensory-motor index, the ratio between the output strength to the motor cortex M1 and the input strength from the olfactory bulb, of the corresponding brain region (Fig. 2d). As a comparison, a similar trend was also identified in our simulated ODPa task using the TRNN model (Fig. 2e). As an extension to the biology experiments, we varied the magnitudes of differences between the two odor stimuli and tested the accuracies of the models. The performance plotted against the differences could be fitted with a logistic psychometric function with a log-likelihood of -18.25 (Supplementary Fig. 2c).

TRNN maintains memory for a variable amount of time

With the memory encoded in stable attractor activities, the vanilla RNN should be able to keep memory for an arbitrary amount of time. On the other hand, since the memory-related neuronal representation changes dynamically, TRNN model might have trouble with variable delay periods. To test this hypothesis, we trained the TRNN models in ODPa tasks with a delay period between short (i.e., 3 s) and long (i.e., 6 s) time. The TRNNs were able to complete these tasks and got lower accuracy (mean \pm SEM over randomly initialized models, $92.19 \pm 0.13\%$ on average) in variable delay periods than in fixed ones ($96.25 \pm 0.26\%$). With variable delay periods, the transient firing peaks of the neurons could still be tiled to cover the entire delay period (Fig. 3a, Supplementary Fig. 3a). The only noticeable difference is that the temporal density of neurons with peaks in the second half of the delay period, which is also the variable part, is lower in networks trained with variable delay (Supplementary Fig. 3b). One question we would like to ask is whether the neurons have the same functions in trials of different lengths. To test this, we first divided the hidden layer neurons into different functional groups based on their peak activity time, including the baseline group

(peaking before the presentation of the first odor), sample responding group (peaking during the presentation of the first odor), memory encoding group (peaking during the delay period), and test responding group (peaking during the presentation of the second odor). We noticed that the functional group of the neurons varied with the sample odor (Supplementary Fig. 3c). One specific question is whether more neurons would be recruited to encode the memory for longer delays? We divided the memory encoding neurons in long trials with delay of 6 s into early ones and late ones, which exhibited firing peak in the first and last 3 s in the delay period, respectively. For these two groups of memory encoding neurons, their groups in the short trials with delay of 3 s responding to the same sample odor were analyzed (Fig. 3b). If the functions of the neurons are the same in the long and short trials, the two groups of memory encoding neurons should both belong to the memory encoding group in short trials. The early memory encoding neurons were indeed mostly from the memory encoding group. However, it was found that the late memory encoding neurons were not only from the memory encoding group in the short trials but also from the baseline and test responding groups. This suggests that more neurons are recruited for longer memory time, implying that the WM time is limited by neuron numbers and matching the short-term property of the WM.

Then we analyzed how working memory was maintained in trials with delay periods of different lengths. Firstly, the trajectories of neuronal activities before the end of the delay period were plotted, after dimension reduction with demixed principal component analysis (dPCA)⁴⁶ (Fig. 3c). The first stimulus-dependent component and the first time-dependent component were chosen to summarize the activity. As expected, when the sample odor was presented, the two types of odors were encoded in separate trajectories. During the delay period, the trajectories remained separated and kept evolving, regardless of the length of the delay period. This suggests that the memory is encoded dynamically in transient trajectories. Then we

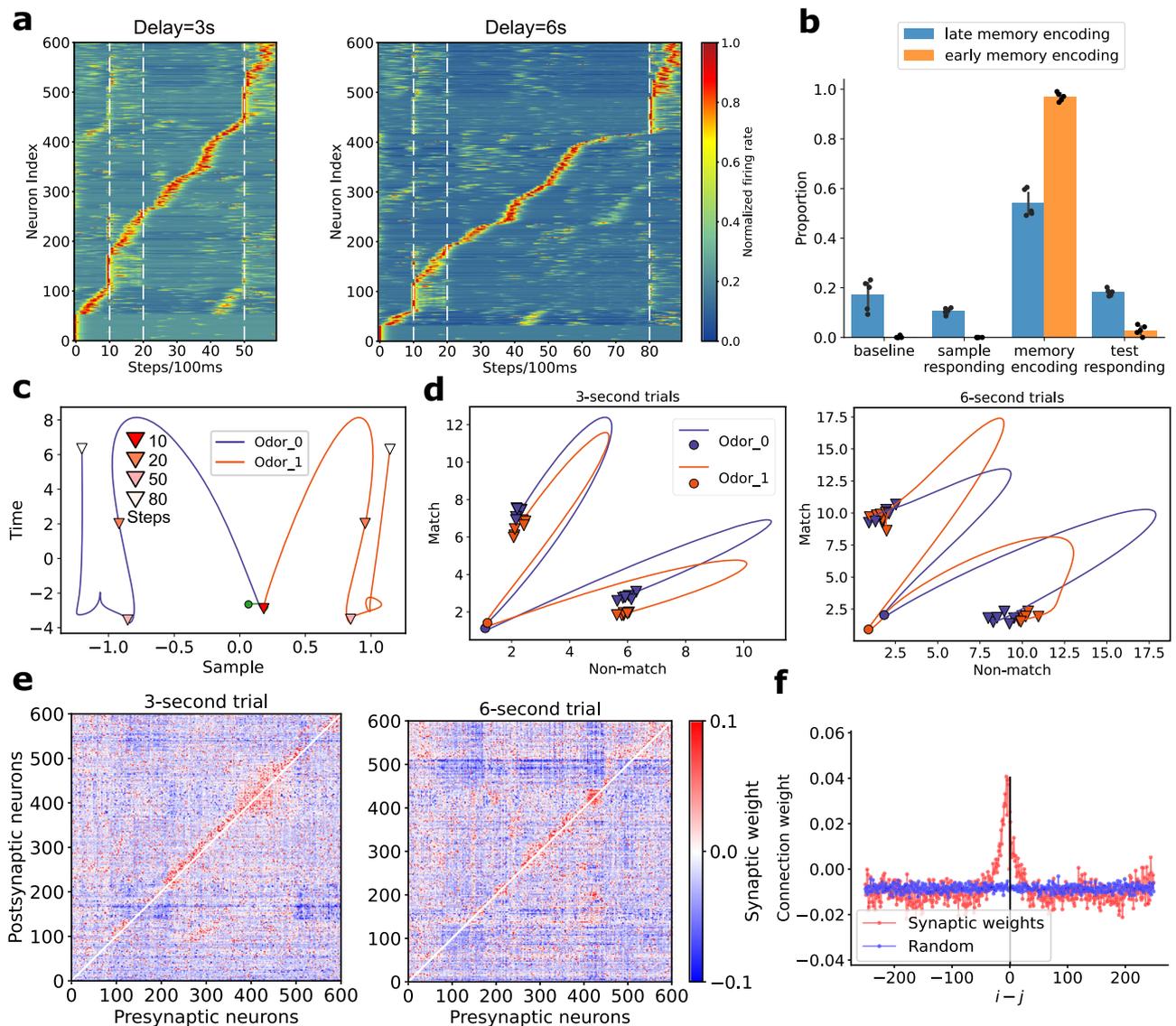


Fig. 3 | Analysis of TRNNs trained in tasks with variable delay periods. **a** Sample neuronal activity heat maps in trials with delay periods of 3 s (left) and 6 s (right). **b** Distribution of the neuronal classes in trials with 3-s delay periods for neurons classified as basic and extended memory encoding neurons in trials with 6-s delay periods. Error bars stand for the standard error of the means. **c** Values of the stimulus-dependent and time-dependent components after dPCA, showing trials with the two different sample odors. Triangles mark the boundaries between trial stages. The green circle marks the start of the trials. **d** Activities of match and non-match output neurons after the end of the delay periods of 3 s (left) and 6 s (right).

Circles mark the states at the end of the delay periods and the colors of them denote the sample odor types. The colors of the trajectories indicate the test odor types. **e** Synaptic weights between neurons in the hidden layer of the TRNN trained for variable delay period tasks. The weights are clipped between -0.1 and 0.1 which includes 93.98% of the data. The neurons are ordered based on their activity peak time in sample trials with different lengths of delay periods. **f** Connection strength between neuron i and neuron j averaged over pairs of neurons with the same rank-order distances $i-j$. The neurons are ordered as in Fig. 3 right.

looked at the activities of the output neurons to see how the memory was compared to the test odor, plotting the activities of the match and non-match output neurons (Fig. 3d). With the same sample odors, different test odors drove the activities in two directions, leading to different decisions. On the other hand, with different sample odors in memory, the same test odors also led to activities in different directions. This result suggests that consistent information could be decoded from the dynamic activities of the neural networks. The trajectories of all three output neurons in the entire trials show that the match and non-match neurons' activities remained close to each other and lower than the fix neuron's activity until the presentation of the test odor (Supplementary Fig. 3d). Lastly, the weight matrix of the recurrent connections between the hidden neurons was studied in Fig. 3e, where the neurons were kept in the same order as those in Fig. 3a. The feedforward connection from neurons early in the chain of activity to later

neurons is stronger than the feedback connection (Fig. 3f). This kind of asymmetry underlies the consecutive transient activity³² and traveling wave formation in head direction cells⁴⁷, which explains how the neural network maintains working memory in transient activities. Such asymmetry is also used in theoretical modeling of transient trajectories^{26,48}. These asymmetrical connections might also underlie the directional spike coupling observed in Huang et al.³⁴.

Information richness and energy cost vary with transient activity

After confirming that TRNNs could reproduce observations in animal experiments and maintain memory for a variable amount of time, we studied their capabilities as memory networks in the ODPa task, by considering their information richness and energy consumption (Fig. 4a). The three parts of the transient encoding modification were adjusted separately

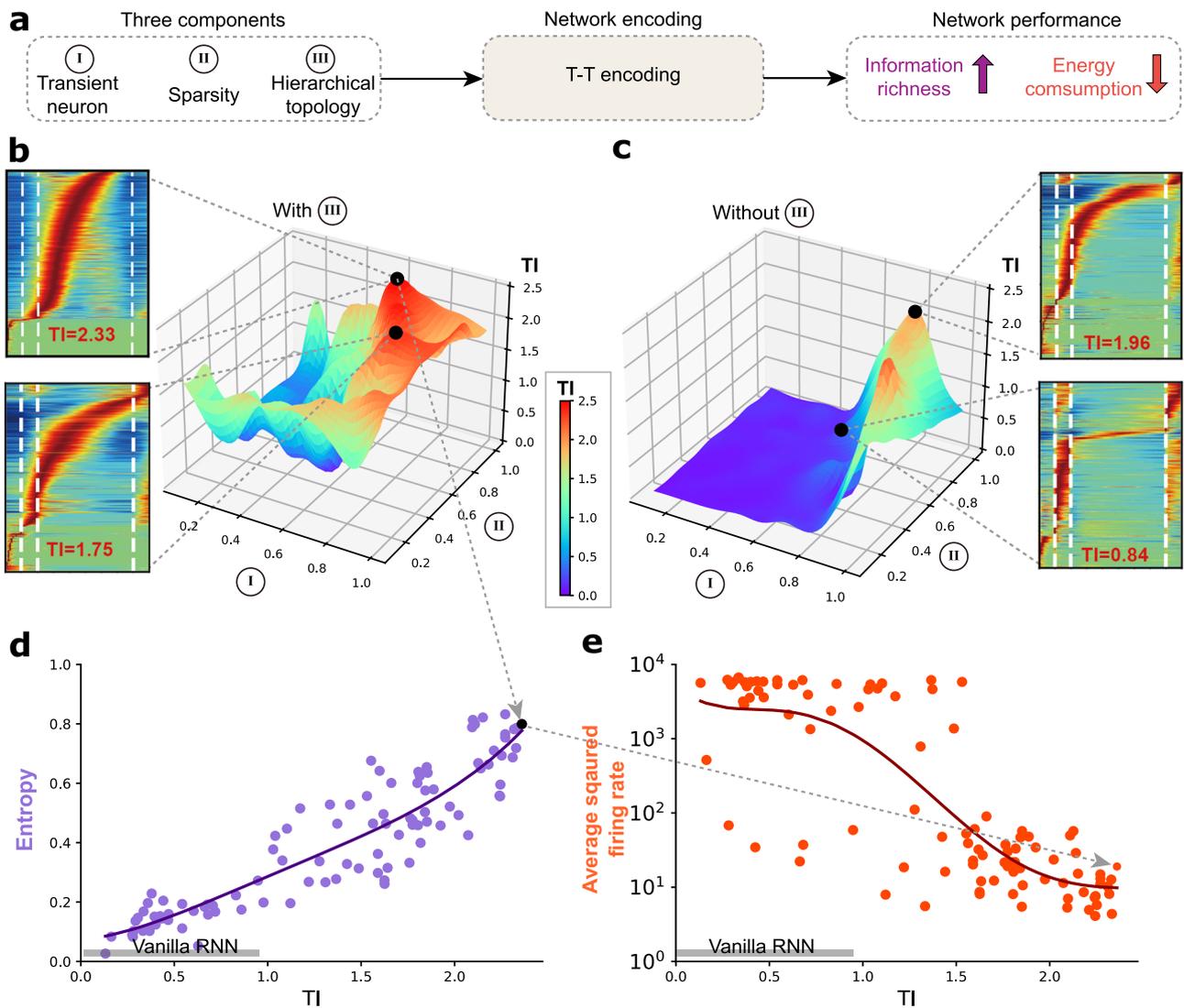


Fig. 4 | Three key components of the TRNN and their contributions. **a** The three components introduced in the TRNN for network encoding, including transient neuron (component I), network sparsity (component II), and hierarchical topology (component III). **b, c** Distribution of TI score performance for TRNNs using different proportions of transient neuron and network sparsity, with **(b)** and without **(c)** using additional hierarchical topology. Black dots in the manifold show some examples of transient trajectories with different TI scores, where transient neurons

using different inhibition strength is used to adjust the transient firing duration. **d, e** Distribution of entropy **(d, purple dots)** and average squared firing rate **(e, orange dots)** values for TRNN models achieving different TI values, given different configurations of three key components in **(b)**. Entropy represents the measurement of information richness. The energy is calculated as the average squared firing rate of all neurons. Note that the vanilla RNN always shows a small TI value and is marked with a gray bar.

with TI as an integrative measurement to quantify the transient degree. For transient neurons, we adjusted the inhibition strength of the neuronal feedback. The sparsity controlled the probability of synaptic connections, and the elimination of connection was random during initialization. Compared to full connection, the hierarchical topology reduced long-term projection connection numbers between neighboring brain regions. The effect of inhibition strength of neurons and sparsity of the network on TI was non-linear (Fig. 4b). In the case without hierarchical topology the TIs were much lower than that with hierarchical topology (Fig. 4c). To study how transient trajectory encoding affects the information richness and energy consumption of the network, we compared 100 networks of the same sizes with different TI values from Fig. 4b. For simplicity, we represented the information richness with the Shannon entropy of the activity as described in the TI. Similarly, since energy consumption was found to be positively correlated to neuronal firing rate^{49,50}, it was represented with the average squared firing rate of all neurons. These properties were plotted against the TI. We found that the entropy increased approximately linearly with the

incremental TI values (Fig. 4d) while the average squared firing rate decreased in a stepwise manner with the TI of the networks (Fig. 4e). These results suggest that compared to the vanilla RNNs with low TIs, the TRNNs with high TIs have higher information richness and lower energy consumption. Higher information richness implies that TRNN could encode more information in its activities and might have higher memory capacity. To test this, we compared their performances in reinforcement learning tasks with complex visual input and reward feedback, that are closer to animal experiments.

TRNN performs better in simulated WM tasks

Besides previously introduced standard classification task, we further tested the networks in reinforcement learning tasks, since they match the value-based learning paradigm used in animals experiments better, and they are also more versatile (e.g. they could be used to simulate spatial working memory experiments). Their performances in the reinforcement learning tasks were compared.

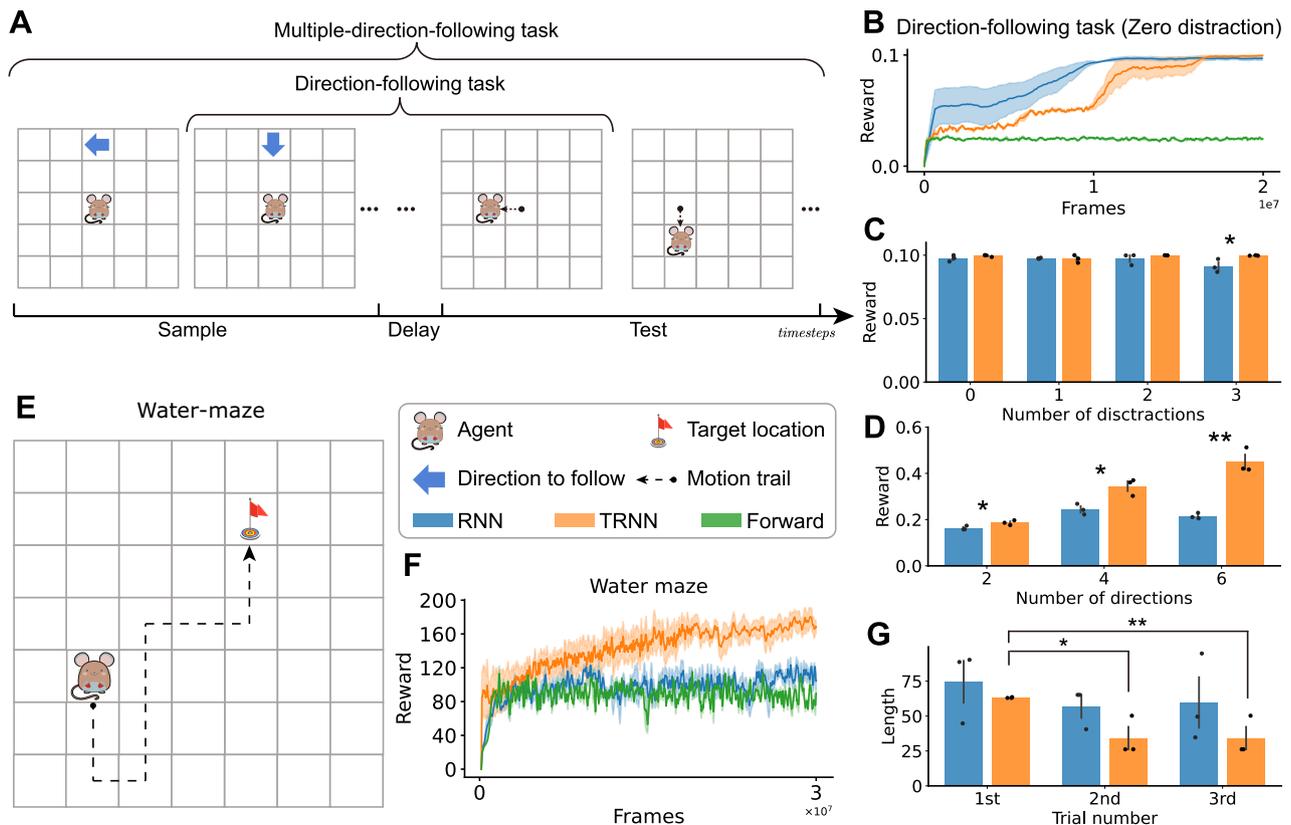


Fig. 5 | Performances comparison of TRNNs and vanilla RNNs in reinforcement learning tasks. **A** Diagram of the direction-following task. The agent is indicated with a mouse icon and has a vision of the entire arena. The blue arrow is the direction to follow. The red flag indicates the target location, which is invisible to the agent. The black arrow indicates the agent’s past movement. **B** Learning curve of the TRNN, vanilla RNN, and feedforward neural network in the direction-following task (mean ± SEM). **C** Rewards of the two models with different numbers of distraction symbols. The mean and the standard error of the mean of their rewards were plotted against the number of training frames. The differences between the TRNNs and RNNs are significant when three distractions (Student *t* test, $t = -2.7932$, $p = 0.0492$) are present. **D** Rewards of the trained RNNs in tasks with different numbers of directions. The difference between the TRNN and vanilla RNN was significant when the number of directions was 2 (Student *t* test, $t = -2.8974$, $p = 0.0442$), 4 (Student *t* test, $t = -3.9975$, $p = 0.0162$) and 6 (Student *t* test,

$t = -7.3029$, $p = 0.0019$). **E** Diagram showing one trial in the water maze task. **F** Learning curve of RNNs in the water maze task (mean ± SEM). **G** Relative time spent to find the platform in the first three trials. Each model was tested in 1000 episodes and the average time was taken. For TRNN, the time in the first trial is significantly higher than the second (Student *t* test, $t = 3.6143$, $p = 0.0225$) and the third (Student *t* test, $t = 6.0052$, $p = 0.0039$) trial. For RNN, the differences are not significant (Student *t* test. 1st vs 2nd, $t = 0.8082$, $p = 0.4643$; 1st vs 3rd, $t = 0.6341$, $p = 0.5604$). All these experiments are repeated three times with each model using different randomly initialized parameters for training. In the learning curves, the mean and the standard error of the mean of their rewards were plotted against the number of training frames. The learning curves were smoothed with the moving average method. The window size is 10 and the stride is 1. In the bar graphs, the average values were plotted with the standard error of the mean as the error bars.

We first simulated a direction-following task, where the agent was shown an arrow, and after a short delay period, the agent was rewarded after moving one step towards the direction of the arrow (Fig. 5A). Both the TRNN and vanilla RNN were trained in the task, and they got very close to maximal rewards, much higher than feedforward neural networks that do not have memory (Fig. 5B). We found the neuronal activity of the vanilla RNN was not persistent at first glance (Supplementary Fig. 4c), confirming previous finding that the activity pattern of RNNs during WM tests could vary continuously between being persistent and sequential based on the task types³³. To quantify the transient pattern of the activities, we calculated their TI and found that TI of TRNNs were much higher (Supplementary Fig. 4d), suggesting that the assumption that the TRNN represents WM models encoding with more transient activities still held in reinforcement learning tasks.

Working memory is generally considered to be able to be maintained even in the presence of distractions⁵¹. Therefore, we designed a new direction-following task that contains a longer delay period when some additional symbols of distractions were presented. The reward of the TRNN was higher than that of the vanilla RNN with three distractions, which suggests that the TRNN is more robust to more distractions (Fig. 5C, Supplementary Fig. 4a).

It has been suggested that attractor models have difficulty in maintaining multiple items¹⁸, especially when the number of items is much bigger than the potential attractors^{52,53}. Our analysis above also suggests that transient-encoding networks have higher memory capacity than attractor-based networks. Since it has been shown that the memory capacity of RNNs varies with the network sizes and task paradigms⁵⁴, we tested two networks of the same size in the same task. In this task, we added more directions to the direction-following task, where the agents were expected to move multiple steps in those directions in order. The learning curves show that the TRNN is better at keeping multiple items in memory (Supplementary Fig. 4b). TRNNs performed better by achieving higher reward with more items to remember (Fig. 5D), indicating that the transient trajectory model has a much higher memory capacity. The rewards of the RNN were higher than from moving in one direction correctly and increased with the number of directions. This suggests that the RNN learned that the task was to move in the observed directions in order. Further analysis about their accuracy for each move in the sequence showed a slight primacy effect but not a recency effect, with better memory of the items that appeared earlier in the sequence than those appeared later, similar to findings in serial recall experiments⁵⁵ (Supplementary Fig. 4e).

Besides sequential information, we simulated a WM version of the Morris water maze experiment^{56,57} in the grid world environment (Fig. 5E), to test TRNN's spatial WM for navigation⁴¹. Specifically, the goal of the agent was to reach the target cell that was randomly assigned in each episode which had multiple trials. One trial composed of three main steps: the agent started from a random initial position; it found the target platform after a period of active searching; the agent reached the target and got reset to a random position. To facilitate training, the episodes had fixed time lengths, and the goal of the agent was to find the target as fast as possible. Learning curves showed that the total reward was increased during training in both models, whereby the TRNNs achieved remarkably higher rewards than vanilla RNNs (Fig. 5F). The vanilla RNN got similar rewards as the feedforward neural network, suggesting that the vanilla RNN is not fit for keeping spatial WM. To confirm this, the time consumption for agents to find the platform was compared. Similar to biological experimental results⁵⁶, the TRNN agents spent a longer time to locate the platform in the first trial but less time in the following two trials with the help of WM (Fig. 5G). As a comparison, the vanilla RNN agent spent similar time in different trials, indicating that they didn't use memory of the target location formed in the first few trials to find it faster in the following trials. This suggests that the TRNN has better spatial WM ability.

Discussion

In this paper, we showed that the structural modifications to a vanilla RNN are enough to reproduce the transient trajectory activity. Though transient, these trajectory activities could also keep memory for variable time. Compared to the vanilla RNN, TRNN has better performance in various reinforcement learning tasks that require working memory. These results support the transient trajectory theory as a mechanistic explanation of animal working memory.

There have been some ways to restrict the neural network to produce transient trajectory patterns in the literature. Goldman proposed a functionally feedforward network, which restricts the recurrent connection weight matrix to be a linear transformation of a feedforward connection weight matrix⁴⁸. This results in transient dynamics of neurons whose summed output could last. Since the linear network usually performs poorly in machine learning tasks, we did not take this approach. Bick and Rabinovich proposed a working memory model based on winner-less competition³⁸. Since all the inter-neuronal connections were inhibitory in that model, it was not used in this study. Instead, we chose some restrictions that resulted in transient activity of the neurons but not necessarily consecutive wave-like activity across the network. We expected that the training for memory tasks would make the network rely on consecutive activity across neurons to keep memory. This was based on the general concept that the RNN stores memory in cell activity. Since the network design prohibited lasting activity in individual neurons, the activities would dynamically span the entire delay period to keep the memory.

The self-inhibition connection was a simplification of the inhibitory connections in the brain, it sufficed to build artificial neural networks with the transient trajectory pattern. In this simplified implementation, the rate-coded neurons in our model could be interpreted as groups of excitatory and inhibitory neurons with recurrent connections of both signs. It is also possible that the transient activity feature is implemented with other mechanisms such as SFA in the brain^{35–40}.

In tasks with fixed length of memory time, the network only needs to establish a one-to-one association between its final state in the transient trajectory and the memory content. This is, however, not quite viable when the memory time varies. This challenges the transient trajectory theory as a mechanism of working memory in the nervous system. In our tests, however, TRNN models were found to generalize to variable memory time and the dynamic trajectories were analyzed. Unlike previous discovery²⁹, we found that the velocity of the trajectory in the phase plane did not decrease towards the end of the delay period, which suggests that the memory is maintained in a trajectory instead of a fixed point (Fig. 3c). With the decision being one of the major sources of variance in the activities, some principal

components should also represent the decision. As shown in Supplementary Fig. 3e, f, the decision-dependent component of the hidden neuronal activities has similar trajectories as the decision output neurons. This component is independent of the sample odor during the sample phase. However, when the test odor is presented, the component diverges based on the comparison between the two odors. This is shown clearer when both the decision-dependent component and the sample-dependent component are plotted in the same graph (Supplementary Fig. 3g, h).

It has been previously proposed that the asymmetric synaptic connection underlies traveling wave activities both in theoretical models⁴⁷ and numerical neural network models³². This type of asymmetric connection has been shown to be equivalent to the SFA or self-inhibition in a continuous attractor neural network (CANN)³⁷. Interestingly, in our models, both the self-inhibition and the asymmetric connection were present to support the working memory storage through the transient trajectory. The reason why the self-inhibition and the asymmetry were shown to have redundant function in the CANN could be that a distance-based recurrent connection has already been incorporated to form attractor states for memory keeping.

Since the firing rate model is used in our study, it was assumed that the memory is stored in the firing rates. However, the theories of WM only specify that the neuronal activity encodes the memory, but they do not exclude the possibility that the spike timing is involved⁵⁹. Therefore, whether spike timing coding is involved in WM should be considered before applying findings in this work. Ideally, spiking neural networks (SNN) could be used for simulation. Self-inhibition and connection modification should reduce persistency in SNN as well. However, they were not used in this study due to the lack of efficient training techniques for complex tasks.

Traditional computational models are usually specifically tailored to fit certain activity patterns, and it would be hard to compare the performance of the models on the same ground. Besides, traditional computational models are mostly limited to a specific task paradigm. In this study, we compared the memory capabilities of networks with different activity patterns using the transient encoding modifications. With basic properties of the networks, such as parameter number and structure, controlled, we were able to compare their capabilities as memory networks in various simulated working memory experiments. This was valuable because it would be very hard to manipulate the memory mechanism in live animals to evaluate the capabilities of different models. The simulation results could provide hypothetical answers to why one working memory model is adopted over the other.

We found the vanilla RNNs had low TIs and defined them as attractor models. The TRNNs had higher TIs on average and were defined as transient trajectory models. Although the boundary between them was not a clear cut. We analyzed their properties as a group by sampling more than one model in each group to avoid the bias issue. It was confirmed that, in the direction-following reinforcement learning task, the TRNNs have higher TIs. The water maze task does not have an apparent delay period and TI over the delay period does not exist. Thus, we assumed that the TRNNs represented the transient trajectory model and the vanilla RNNs represented the attractor model based on the earlier results.

In the multiple-direction-following task, we only observed primacy effect when multiple items are recalled. In free recall experiments, where items were freely recalled regardless of their order, both the primacy effect and recency effect were observed^{60–63}. Our setup is a type of serial recall experiment which showed no recency effect in a previous study⁵⁵. Specifically, the agents needed to recall the directions in the exact order as shown. Since they were not trained to fill the positions of forgotten directions with placeholder actions, if they skipped a direction, all following actions are likely wrong.

In the water maze experiment, the overall performance of the trained TRNN was much better than the vanilla RNN. The vanilla RNN and the feedforward neural network had similar performances. This suggests that vanilla RNNs have worse navigation ability possibly due to poorer memory of the entire map, failure to learn the task, or worse movement execution.

This could make the comparison of their memory abilities with the TRNNs less valid.

In summary, we introduced transient encoding modifications to RNNs to compare transient encoding of working memory with attractor encoding. Through a series simulation experiments and analyses, we found evidence that support transient encoding as a working memory mechanism, which is a supplement to animal experiments where direct manipulation of encoding mechanism is hard. In the future, this approach could be extended to combining working memory research with long-term memory research^{64–66} to study the information transfer between working memory and long-term memory and how items from observation and long-term memory interact within working memory.

Methods

Simulated working memory tasks

The ODPa task. The ODPa task was designed to test the decision-making response of head-fixed mice after being given two different odor stimuli. It starts with a fixation period of 1 s and a sample period of 1 s, during which one out of two odors is presented. After a delay period of 3 or 6 s, a test stimulus, one odor from another set of two odors, was presented for 1 s. At the end of the second stimulus, the animal could respond in the response window of 1 s. The animals are rewarded if they lick in response when the two stimuli matched. Note that each sample odor was assigned to match a different test odor before training arbitrarily.

We designed a digital version of the OPDA task to test the ANN models. It is a supervised learning task with sequential inputs and outputs. There are two major differences from the animal version. The first is that the response period is removed, and the model responds during the test period. Secondly, the odor input is replaced with a vector with amplitude A and direction θ . 24 neurons are used to encode the odor input using von Mises distribution. Each of the encoding neurons i has a preferred input direction θ_{pref}^i , which are evenly distributed between 0 and 2π . The output of neuron i at time step t could be expressed as the following equation.

$$u_t^i = Ae^{\kappa \cos(\theta - \theta_{pref}^i)} + \sqrt{2/a\sigma_{in}}N(0, 1), \quad (1)$$

where the amplitude A is set as $4/\exp(\kappa)$ when the odor stimulation is on and 0 when the stimulation is off. κ is the concentration coefficient and is set to 2. $N(0, 1)$ is the standard normal distribution used to represent input noise. The noise variance $\sigma_{in} = 0.1$, and $a = 0.1$. The inputs of $\theta = 0$ and $\theta = \pi$ are used to represent odor 0 and odor 1. Odor 0 and odor 1 are used in both the sample and test period and the test criterion is whether the two odors are the same. The model is expected to output the response during the test period. The stimulation was on during the sample period and the test period. The stimulation was off during the delay period and the response period.

We designed three binary output neurons for the ODPa task, and the expected activity is known for every input sequence. The fixation neuron signals inactivity and should be on until the start of the test period. The match/non-match neurons are only on during the test period when the sample and the test inputs are the same/different respectively.

Memory tasks in the grid world environment. The direction-following task and its variants and the 2D water maze task were all implemented in the grid world environment. The agent has a discrete action space with three actions, turning left/right by 90 degrees and moving forward. In both tasks, the observation is the visual input of the entire map. The direction following task is adapted from the Mortar Mayhem task from the Memory Gym library⁶⁷. The 2D water maze task was implemented using the Minigrid library⁶⁸.

In the direction-following task, the environment is a 5×5 grid. In the sample phase, a symbol is randomly chosen from up, down, left, and right arrows and a circle is shown. The symbol is displayed at the center of the grid for three timesteps. After the delay period of 16 timesteps, the agent is

required to act within six timesteps according to the symbol in memory for a reward of 0.1. If one of the arrows is shown, the agent should move to the next cell in the direction of the arrow. If the circle is shown, the agent should stay. In the version with distractions, 1 to 3 random distracting symbols are displayed during the delay period. In the multiple-direction version, 2, 4, or 6 random symbols are shown in the sample phase. In the test phase, the agent should take the accordant actions in the order as the symbols are shown. There are 6 timesteps for each action and 2 timestep gaps in between. To compare the TRNN and the vanilla RNN, 3 randomly initialized networks from each group were trained to solve the tasks.

The 2D water maze task is adapted from the design by Heess et al.⁵⁷. Instead of a circular arena, a 9×9 square grid world arena is used. There are landmarks positioned at the four corners of the arena. There is a hidden platform at a random cell which is only visible when the agent reaches it and its position changes across episodes. The agent has a vision of the entire arena. The agent starts at a random position and is only rewarded when it is on the platform. After 5 timesteps on the platform, the agent is placed in a random position and needs to find the platform again. The entire episode has 484 timesteps.

Design and training of the neural networks

Vanilla RNN. The vanilla RNN is a standard three-layer network. The hidden layer contains 600 artificial neurons with full recurrent connections. The W_{in} , W_r and W_{out} are connection weights between input-hidden, inner hidden, and hidden-output layers, respectively. The neurons are separated into excitatory and inhibitory groups (4:1 ratio). The output connections of the excitatory neurons are initialized by the Gamma distribution with a shape parameter of 0.1 and scale parameter of 1. The inhibitory output connections are initialized as the negative of a value from the Gamma distribution with a shape of 0.2 and scale of 1. The inhibitory neurons in the input and output layers are ignored. The network biases in hidden and output layers, b_r and b_{out} , are initialized to 0 before network learning. The input layer does not have a bias term. The artificial neuron in hidden layers can be defined as the following equation.

$$\tau \frac{dr_t}{dt} = -r_t + f(W_{in}u_t + W_r r_t + b_r + \sigma_r), \quad (2)$$

where u_t is the input from the input layer and r_t is the recurrent input. τ is the membrane time constant. Rectified linear unit (ReLU) is chosen as the activation function $f(\cdot)$. σ_r represents an independent Gaussian white noise with a mean of 0 and a standard variance of 0.1. We use the forward Euler method to numerically solve the equation and get the following equation.

$$r_t = (1 - \alpha)r_{t-1} + \alpha f(W_r r_{t-1} + W_{in}u_t + b_r + \sigma_r), \quad (3)$$

where $\alpha = \Delta t/\tau$ represents the time step size relative to the membrane time constant. In the ODPa task and reinforcement learning tasks, α is set as 0.6 and 0.98, respectively. The time step size $\Delta t = 100ms$. The network is trained by the Adam stochastic gradient descent with a batch size of 64. The learning rate is 0.001 and the loss function was defined as the cross-entropy error. The average squared neuronal activity is also added to the loss function with a factor of 0.01 to stabilize the network.

Transient RNN. The network architecture of the TRNN is the same as that of the vanilla RNN but additionally separates hidden layers into three sub-brain regions, i.e., the sensory, association, and motor brain regions. It is assumed that the inter-region connections are sparser than the intra-region connections. This hierarchical topology is implemented by applying an element-wise mask matrix W_m to the recurrent connection matrix W_r . Random inter-region connection elements in the mask are zeros and others are ones.

Compared to artificial neurons in the vanilla RNN in Eq. (1), the transient neuron has an additional self-inhibition term V to inhibit neural activities instantaneously. Originally proposed as a slow negative feedback

component in the traveling wave model³⁵, the transient neuron is defined by the following equations.

$$\tau_r \frac{dr_t}{dt} = -r_t + f(W_{in}u_t + W_r r_t + b_r + \sigma_r) - \gamma V_t \quad (4)$$

$$\tau_v \frac{dV_t}{dt} = -V_t + mr_t \quad (5)$$

where τ_r is the time constant of the transient neuron. ReLU is used as the activation function f . τ_v is the time constant of the dynamic inhibition term. The hyperparameters γ and m represent the scale of the influence of self-inhibition. In the ODPa task $\gamma = 2$, $m = 2$, in the direction following task $\gamma = 10$, $m = 2$, and in the water maze task $\gamma = 3$, $m = 2$. Similarly, with Euler method approximation, we get

$$r_t = (1 - \alpha_r)r_{t-1} + \alpha_r(f(W_r r_{t-1} + W_{in}u_t + b_r + \sigma_r) - \gamma V_{t-1}), \quad (6)$$

$$V_t = (1 - \alpha_v)V_{t-1} + \alpha_v(mr_{t-1}), \quad (7)$$

In ODPa task and the reinforcement learning tasks, α_r is set as 0.6 and 0.98 respectively and α_v is set as 1/10. A sparse connection is imposed by randomly initializing a proportion of the connection W_r as zero. They are still trainable after initialization. This proportion parameter was adjusted when studying the impact of sparsity on the TL. The connections between regions are initialized to be sparser than within the regions. The connection ratio is 0.9 within the sensory region and 1 within the other two regions. The connection ratio across regions is 0.8. A similar training method as with the RNN was used. Except for that an L2 regularization of the recurrent connection weights was added to the loss with a factor of 0.1 to maintain sparsity.

RNN models with convolution layers. For the tasks in the grid world environment, convolution layers were attached ahead of the RNN to extract features from the video input of 3 color channels (84 × 84 pixels for Memory Gym and 64 × 64 pixels for Minigrid). We used three convolution layers with 3, 32, and 64 filters respectively. The filters' sizes are 8 × 8, 4 × 4, and 3 × 3 respectively, and the strides are 4, 2, and 2 respectively. The output of the convolution layers is squeezed into 1D by concatenation before being fed into the RNN.

Proximal policy optimization. To train our models in these reinforcement learning tasks, we used the proximal policy optimization (PPO) algorithm due to its robustness (not sensitive to hyperparameter tuning)⁶⁹. As an on-policy algorithm, it suffers from sample complexity problems, where it needs to produce a lot of samples during the training. To alleviate the problem and speed up training, an asynchronous PPO algorithm was proposed with the sample factory architecture⁷⁰. We used their architecture and implementation to train our models within a reasonable amount of time. The goal of the PPO is to maximize the expected return of a policy π_θ , represented with $\mathcal{J}(\theta)$, which is equivalent to the Q-value under the policy, shown as the following equation.

$$\mathcal{J}(\theta) = \hat{E}_t[\hat{Q}_t]. \quad (8)$$

Denoting the parameters of the policy before the update as θ_{old} , it's been shown that the expected return of a policy π_θ could be linked to the policy before the update $\pi_{(\theta_{old})}$ with the following equation.

$$\mathcal{J}(\theta) = \mathcal{J}(\theta_{old}) + \hat{E}_t \left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \hat{A}_t \right]. \quad (9)$$

\hat{A}_t is the estimation of the advantage $A(s_t, a_t) = Q(s_t, a_t) - V(s_t)$ where V is the state value function. Let $r_t(\theta)$ denote the probability ratio $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$. The goal becomes maximizing a surrogate objective function.

$$\mathcal{L}(\theta) = \hat{E}_t[r_t(\theta)\hat{A}_t]. \quad (10)$$

To keep the updated policy from being too different from the current policy, PPO clipped $r_t(\theta)$ to be around 1 with a hyperparameter $\epsilon = 0.2$, then the surrogate objective becomes the following format.

$$\mathcal{L}^{CLIP}(\theta) = \hat{E}_t \left[\min \left(r_t(\theta)A_t, \text{clip} \left(r_t(\theta), \frac{1}{1+\epsilon}, 1+\epsilon \right) \hat{A}_t \right) \right]. \quad (11)$$

The advantage \hat{A}_t is estimated using generalized advantage estimation in a length-T trajectory based on state value functions, as shown in the following equations.

$$\hat{A}_t = \delta_t + (\gamma\lambda)\delta_{t+1} + \dots + (\gamma\lambda)^{T-t-1}\delta_{T-1}, \quad (12)$$

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t), \quad (13)$$

where neural network models are used to approximate both the policy π and the state value V . Thus a loss function that combines the policy surrogate and the mean squared error of state values is used, shown as follows:

$$\mathcal{L}^{CLIP+VF}(\theta) = \mathcal{L}^{CLIP}(\theta) - c_1 E_t \left[(V_\theta(s_t) - \hat{V}_t)^2 \right] + c_2 S[\pi_\theta](s_t), \quad (14)$$

where c_1 and c_2 are coefficients, $\hat{V}_t = \sum_{n=t}^{T-1} \gamma^n r_{n+1}$ is the sampled state value and S is an entropy of the policy. This combined target is maximized using stochastic gradient descent.

In the sample factory architecture, the training process is delegated to three parallel components, including the rollout workers, policy workers, and learners. The rollout workers are responsible only for environmental simulation. The rollout worker calculates observations x_t , rewards r_t , and feeds them to the policy workers. Policy workers sample actions based on the current policy and update hidden states h_t of the agents. The length-T trajectories become available to the learners once completed. The learners continuously process batches of trajectories and update the neural network parameters asynchronously, meaning that the policy workers don't wait for the policy updates. In our experiments, one learner, one policy worker, and 8–32 rollout workers were used, and each rollout worker hosted 12 environment instances.

Analysis of the neural networks

Transient index. The transient index is used to measure how well the activity pattern of neuronal activities during the delay period matches the transient trajectory theory. Three aspects of the activity pattern are considered, including the uniformity, selectivity, and specificity of the neuronal activities. These are measured by Shannon entropy, ridge-to-background ratio³³, and proportion of memory-related peak firing respectively.

Shannon entropy. It is calculated to represent the information capacity of all neurons in networks using different encoding methods. For each time window t in a delay period, the probability of neurons attending encoding peak firing of a trajectory is $p(t) = n_t/N$. n_t is the number of neurons with their peak activity in this time window and N is the total number of neurons. Then, the Shannon entropy is calculated using the following equation,

$$H(x) = - \sum_{t=1}^L p(t) \log_2 p(t), \quad (15)$$

where L represents the number of split time windows (each is 100 ms) for raw delay period. A pseudo-count of 0.1 was added to each bin before calculating the entropy. Shannon entropy also represents the uniformity of the peak firing time, where a higher entropy usually indicates that neurons fire more uniformly during the delay period. When combined with the other properties, the entropy is normalized using the following equation,

$$E(x) = \frac{H(x)}{\log_2 L}. \quad (16)$$

Ridge-to-background ratio. It is used to describe the selectivity of the neural activity, i.e., how concentrated the firing activity of neurons is around the peak. The ridge was defined as the mean activity in the $\Delta t = 1$ timestep surrounding the peak value, and the background was defined as the mean activity in all the timesteps. Thus the ratio is defined as the following equation:

$$R(x) = \frac{1}{N} \sum_i \frac{\sum_{t=t_{ip}-\Delta t}^{t_{ip}+\Delta t} r_{it}}{\sum_{t=0}^T r_{it}}, \quad (17)$$

where r_{it} is the firing rate of neuron i at time step t . T is the trial length and t_{ip} is the peak firing time of neuron i .

Proportion of memory-related peak firing. It measures how specific the peak firing of the neurons is to the delay period when the memory is maintained. It is calculated using the following equation.

$$P(x) = \frac{N_a}{N}, \quad (18)$$

where N_a is the number of neurons whose firing peak lies in the delay period. Neurons firing at the peak during the delay period could help to form the transient trajectory and transmit odor information. A higher proportion indicates more neurons are involved in the formation of the transient trajectory. Then the TI indicator is defined as the sum of these three sub-indicators after normalization to the value range of zero to one, shown as the following equation.

$$TI(x) = E(x) + R(x) + P(x). \quad (19)$$

Memory-selective neurons

To identify memory-selective neurons, the trials are each divided into one second time bins. Wilcoxon rank-sum test⁷¹ is done by comparing the mean firing rates of the trials from different stimuli in each time bin. If the mean firing rates are significantly different in any time bin between two stimuli, the neuron is considered selective to the stimuli with higher responses. One neuron might be selective to multiple stimuli depending on which stimulus is compared against. However, memory-selective neurons are defined as those that have statistically significant firing rates for one specific stimulus. In contrast, those without selectivity or multiple selective stimuli are defined as non-selective neurons. Based on this definition, memory-selective neurons in TRNNs after learning an ODPA task could also be identified accordingly.

Psychometric function

We fitted the TRNN model's performance data in the OPDA task to a logistic psychometric function, shown as follows:

$$P(\theta) = \frac{1}{1 + e^{\frac{\theta-m}{s}}}. \quad (20)$$

The maximum likelihood estimation is used to determine the parameters m and s in the psychometric function. Powell algorithm is used to

calculate the maximum likelihood and the corresponding parameters m and s . A total number of $n = 64$ trials is tested for each θ . With r_i correct trials for θ_i found in the tests, the log-likelihood function is defined as the following equation.

$$l = \sum_i [r_i \log P(\theta_i) + (n - r_i) \log P(\theta_i)]. \quad (21)$$

Demixed principal component analysis

We used dPCA to reduce the dimension of the population activities. This algorithm is described in detail by Kobak et al.⁴⁶. In brief, the activity matrix of the neurons, X , is decomposed into task variable dependent terms by taking averages over the task variables. In our case, we first decomposed the matrix into sample-odor-independent, sample-odor-dependent, and noise terms, shown as the following equation.

$$X = X_t + X_{st} + X_{noise} = \sum_{\phi} X_{\phi} + X_{noise}. \quad (22)$$

For additional discussion, we also decomposed the matrix into sample-odor-dependent, decision-dependent, time dependent, and noise terms, shown as the following equation.

$$X = X_t + X_{st} + X_{dt} + X_{noise} = \sum_{\phi} X_{\phi} + X_{noise}. \quad (23)$$

Each term is then approximated using separate decoder and encoder matrices, D_{ϕ} and F_{ϕ} , by minimizing the following loss function.

$$L_{dPCA} = \sum_{\phi} \left\| X_{\phi} - F_{\phi} D_{\phi} X \right\|^2. \quad (24)$$

The decoder matrix D_{ϕ} is composed of the demixed principal components similar to PCA. The difference is that the demixed principal components reconstruct the task variable-specific activities. These components are ordered by the amount of variance explained. In the sample response and delay period analysis, the principal components specific to the sample-odor-dependent term D_{st} was used. For the discussion about the decision representation, the sample-odor-dependent term D_{st} and the decision-dependent term D_{dt} were used.

Statistics and reproducibility

For the TI comparison in the ODPA task, Student t test was used to analyze the statistical significance. The animal data had averaged TIs in 14 recording sessions, and the modeling data had TIs of 10 models trained with different initializations. In the reinforcement learning tasks, all data was tested with Student t test. Three models trained with different randomly initialized parameters were included in each group.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

The source data used for the graphs in the figure is provided in the supplementary data file. The data could also be reproduced using the modeling code. All other data are available from the corresponding author on reasonable request.

Code availability

The code used in this study is available at <https://github.com/jiashuncheng/Transient>.

Received: 13 June 2024; Accepted: 15 November 2024;
Published online: 28 January 2025

References

- Baddeley, A. Working memory. *Science* **255**, 556–559 (1992).
- Fuster, J. M. & Alexander, G. E. Neuron activity related to short-term memory. *Science* **173**, 652–654 (1971).
- Funahashi, S., Bruce, C. J. & Goldman-Rakic, P. S. Mnemonic coding of visual space in the monkey's dorsolateral prefrontal cortex. *J. Neurophysiol.* **61**, 331–349 (1989).
- Xie, Y., Liu, Y. H., Constantinidis, C. & Zhou, X. Neural mechanisms of working memory accuracy revealed by recurrent neural networks. *Front. Syst. Neurosci.* **16**, 760864 (2022).
- Compte, A. Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. *Cereb. Cortex* **10**, 910–923 (2000).
- Wimmer, K., Nykamp, D. Q., Constantinidis, C. & Compte, A. Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory. *Nat. Neurosci.* **17**, 431–439 (2014).
- Shafi, M. et al. Variability in neuronal activity in primate cortex during working memory tasks. *Neuroscience* **146**, 1082–1108 (2007).
- Stokes, M. G. et al. Dynamic coding for cognitive control in prefrontal cortex. *Neuron* **78**, 364–375 (2013).
- Meyers, E. M., Freedman, D. J., Kreiman, G., Miller, E. K. & Poggio, T. Dynamic population coding of category information in inferior temporal and prefrontal cortex. *J. Neurophysiol.* **100**, 1407–1419 (2008).
- Brody, C. D. Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. *Cereb. Cortex* **13**, 1196–1207 (2003).
- Murray, J. D. et al. Stable population coding for working memory coexists with heterogeneous neural dynamics in prefrontal cortex. *Proc. Natl. Acad. Sci.* **114**, 394–399 (2017).
- Druckmann, S. & Chklovskii, D. B. Neuronal circuits underlying persistent representations despite time varying activity. *Curr. Biol.* **22**, 2095–2103 (2012).
- Maass, W., Natschläger, T. & Markram, H. Real-time computing without stable states: a new framework for neural computation based on perturbations. *Neural Comput.* **14**, 2531–2560 (2002).
- Barak, O., Sussillo, D., Romo, R., Tsodyks, M. & Abbott, L. F. From fixed points to chaos: three models of delayed discrimination. *Prog. Neurobiol.* **103**, 214–222 (2013).
- Lisman, J. E. & Idiart, M. A. P. Storage of 7 ± 2 short-term memories in oscillatory subcycles. *Science* <https://doi.org/10.1126/science.7878473> (1995).
- Kopell, N., Whittington, M. A. & Kramer, M. A. Neuronal assembly dynamics in the beta1 frequency range permits short-term memory. *Proc. Natl. Acad. Sci.* **108**, 3779–3784 (2011).
- Champion, K. P., Gozel, O., Lankow, B. S., Ermentrout, G. B. & Goldman, M. S. An oscillatory mechanism for multi-level storage in short-term memory. *Commun. Biol.* **6**, 1–9 (2023).
- Lundqvist, M., Herman, P. & Miller, E. K. Working memory: delay activity, yes! persistent activity? Maybe not. *J. Neurosci.* **38**, 7013–7019 (2018).
- Mongillo, G., Barak, O. & Tsodyks, M. Synaptic theory of working memory. *Science* **319**, 1543–1546 (2008).
- Masse, N. Y., Yang, G. R., Song, H. F., Wang, X.-J. & Freedman, D. J. Circuit mechanisms for the maintenance and manipulation of information in working memory. *Nat. Neurosci.* **22**, 1159–1167 (2019).
- Kozachkov, L. et al. Robust and brain-like working memory through short-term synaptic plasticity. *PLoS Comput. Biol.* **18**, e1010776 (2022).
- Baeg, E. H. et al. Dynamics of population code for working memory in the prefrontal cortex. *Neuron* **40**, 177–188 (2003).
- Fujisawa, S., Amarasingham, A., Harrison, M. T. & Buzsáki, G. Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nat. Neurosci.* **11**, 823–833 (2008).
- Harvey, C. D., Coen, P. & Tank, D. W. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature* **484**, 62–68 (2012).
- Zhu, J. et al. Transient delay-period activity of agranular insular cortex controls working memory maintenance in learning novel tasks. *Neuron* **105**, 934–946.e5 (2020).
- Rabinovich, M. et al. Dynamical encoding by networks of competing neuron groups: winnerless competition. *Phys. Rev. Lett.* **87**, 068102 (2001).
- Afraimovich, V. S., Rabinovich, M. I. & Varona, P. Heteroclinic contours in neural ensembles and the winnerless competition principle. *Int. J. Bifurc. Chaos* **14**, 1195–1208 (2004).
- Afraimovich, V. S., Zhigulin, V. P. & Rabinovich, M. I. On the origin of reproducible sequential activity in neural circuits. *Chaos Interdiscip. J. Nonlinear Sci.* **14**, 1123–1129 (2004).
- Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J. & Wang, X.-J. Computing by robust transience: how the fronto-parietal network performs sequential, category-based decisions. *Neuron* **93**, 1504–1517.e4 (2017).
- Cueva, C. J. et al. Low-dimensional dynamics for working memory and time encoding. *Proc. Natl. Acad. Sci.* **117**, 23021–23032 (2020).
- Ichikawa, K. & Kaneko, K. Short-term memory by transient oscillatory dynamics in recurrent neural networks. *Phys. Rev. Res.* **3**, 033193 (2021).
- Rajan, K., Harvey, C. D. & Tank, D. W. Recurrent network models of sequence generation and memory. *Neuron* **90**, 128–142 (2016).
- Orhan, A. E. & Ma, W. J. A diverse range of factors affect the nature of neural representations underlying short-term memory. *Nat. Neurosci.* **22**, 275–283 (2019).
- Huang, E. et al. *Hierarchical replay of multi-regional sequential spiking associated with working memory*. 2023.10.08.561458 Preprint at <https://doi.org/10.1101/2023.10.08.561458> (2023).
- Pinto, D. J. & Ermentrout, G. B. Spatially structured activity in synaptically coupled neuronal networks: I. Traveling fronts and pulses. *SIAM J. Appl. Math.* **62**, 206–225 (2001).
- Bressloff, P. C. Spatiotemporal dynamics of continuum neural fields. *J. Phys. Math. Theor.* **45**, 033001 (2011).
- Mi, Y., Fung, C. C. A., Wong, K. Y. M. & Wu, S. Spike Frequency Adaptation Implements Anticipative Tracking in Continuous Attractor Neural Networks. in *Advances in Neural Information Processing Systems* (eds. Ghahramani, Z. et al.) 27 (Curran Associates, Inc., 2014).
- Bellec, G., Salaj, D., Subramoney, A., Legenstein, R. & Maass, W. Long short-term memory and learning-to-learn in networks of spiking neurons. in *Proceedings of the 32nd International Conference on Neural Information Processing Systems* 795–805 (Curran Associates Inc., Red Hook, NY, USA, 2018).
- Bellec, G. et al. A solution to the learning dilemma for recurrent networks of spiking neurons. *Nat. Commun.* **11**, 3625 (2020).
- Salaj, D. et al. Spike frequency adaptation supports network computations on temporally dispersed information. *eLife* **10**, e65459 (2021).
- Dudchenko, P. A. An overview of the tasks used to test working memory in rodents. *Neurosci. Biobehav. Rev.* **28**, 699–709 (2004).
- Niv, Y. Reinforcement learning in the brain. *J. Math. Psychol.* **53**, 139–154 (2009).
- Nachstedt, T. & Tetzlaff, C. Working memory requires a combination of transient and attractor-dominated dynamics to process unreliably timed inputs. *Sci. Rep.* **7**, 2473 (2017).
- Turner, E., Dabholkar, K. & Barak, O. Charting and navigating the space of solutions for recurrent neural networks. in *Proceedings of the 35th International Conference on Neural Information Processing Systems* 25320–25333 (Curran Associates Inc., Red Hook, NY, USA, 2021).
- Song, H. F., Yang, G. R. & Wang, X.-J. Training excitatory-inhibitory recurrent neural networks for cognitive tasks: a simple and flexible framework. *PLOS Comput. Biol.* **12**, e1004792 (2016).

46. Kobak, D. et al. Demixed principal component analysis of neural population data. *eLife* **5**, e10989 (2016).
47. Zhang, K. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *J. Neurosci.* **16**, 2112–2126 (1996).
48. Goldman, M. S. Memory without feedback in a neural network. *Neuron* **61**, 621–634 (2009).
49. Yi, G. & Grill, W. M. Average firing rate rather than temporal pattern determines metabolic cost of activity in thalamocortical relay neurons. *Sci. Rep.* **9**, 6940 (2019).
50. Levy, W. B. & Calvert, V. G. Communication consumes 35 times more energy than computation in the human cortex, but both costs are needed to predict synapse number. *Proc. Natl. Acad. Sci.* **118**, e2008173118 (2021).
51. Baddeley, A. Working memory: theories, models, and controversies. *Annu. Rev. Psychol.* **63**, 1–29 (2012).
52. Edin, F. et al. Mechanism for top-down control of working memory capacity. *Proc. Natl. Acad. Sci.* **106**, 6802–6807 (2009).
53. Almeida, R., Barbosa, J. & Compte, A. Neural circuit basis of visuo-spatial working memory precision: a computational and behavioral study. *J. Neurophysiol.* **114**, 1806–1818 (2015).
54. Xie, Y. et al. *Natural constraints explain working memory capacity limitations in sensory-cognitive models*. 2023.03.30.534982 Preprint at <https://doi.org/10.1101/2023.03.30.534982> (2023).
55. Drewnowski, A. & Murdock, B. B. The role of auditory features in memory span for words. *J. Exp. Psychol. Hum. Learn.* **6**, 319–332 (1980).
56. Steele, R. J. & Morris, R. G. M. Delay-dependent impairment of a matching-to-place task with chronic and intrahippocampal infusion of the NMDA-antagonist D-AP5. *Hippocampus* **9**, 118–136 (1999).
57. Heess, N., Hunt, J. J., Lillicrap, T. P. & Silver, D. *Memory-based control with recurrent neural networks*. Preprint at <https://doi.org/10.48550/arXiv.1512.04455> (2015).
58. Bick, C. & Rabinovich, M. I. Dynamical origin of the effective storage capacity in the brain's working memory. *Phys. Rev. Lett.* **103**, 218101 (2009).
59. Brette, R. Philosophy of the spike: rate-based vs. spike-based theories of the brain. *Front. Syst. Neurosci.* **9**, 151 (2015).
60. Oberauer, K. et al. Benchmarks for models of short-term and working memory. *Psychol. Bull.* **144**, 885–958 (2018).
61. Howard, M. W. & Kahana, M. J. Contextual variability and serial position effects in free recall. *J. Exp. Psychol. Learn. Mem. Cogn.* **25**, 923–941 (1999).
62. Bjork, R. A. & Whitten, W. B. Recency-sensitive retrieval processes in long-term free recall. *Cognit. Psychol.* **6**, 173–189 (1974).
63. Murdock, B. B. Jr. The serial position effect of free recall. *J. Exp. Psychol.* **64**, 482–488 (1962).
64. Tomé, D. F. et al. *Dynamic and selective engrams emerge with memory consolidation*. <https://doi.org/10.1101/2022.03.13.484167> (2022).
65. Luboinski, J. & Tetzlaff, C. Organization and priming of long-term memory representations with two-phase plasticity. *Cogn. Comput.* **15**, 1211–1230 (2023).
66. Zenke, F., Agnes, E. J. & Gerstner, W. Diverse synaptic plasticity mechanisms orchestrated to form and retrieve memories in spiking neural networks. *Nat. Commun.* **6**, 6922 (2015).
67. Pleines, M., Pallasch, M., Zimmer, F. & Preuss, M. Memory gym: partially observable challenges to memory-based agents. in *The Eleventh International Conference on Learning Representations* (2023).
68. Chevalier-Boisvert, M. et al. Minigrad & Miniworld: modular & customizable reinforcement learning environments for goal-oriented tasks. In *Advances in Neural Information Processing Systems 36* (NIPS, 2024).
69. Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. *Proximal Policy Optimization Algorithms*. Preprint at <http://arxiv.org/abs/1707.06347> (2017).
70. Petrenko, A., Huang, Z., Kumar, T., Sukhatme, G. & Koltun, V. Sample factory: egocentric 3D control from pixels at 100,000 FPS with asynchronous reinforcement learning. in *Proceedings of the 37th International Conference on Machine Learning* 119, 7652–7662 (JMLR.org, 2020).
71. Mann, H. B. & Whitney, D. R. On a test of whether one of two random variables is stochastically larger than the other. *Ann. Math. Stat.* **18**, 50–60 (1947).

Acknowledgements

We'd like to thank Dr. Xiaoxing Zhang for the discussion about the interpretation of the animal experimental results. T.Z. and B.X. disclose support for research and publication of this work from the Strategic Priority Research Program of Chinese Academy of Sciences [Grant No. XDB1010302], the Beijing Nova Program [Grant No. 20230484369], the Youth Innovation Promotion Association CAS, and CAS Project for Young Scientists in Basic Research [YSBR-116]. C.T.L. discloses support for research of this work from the Shanghai Municipal Science and Technology Major Project [Grant No. 2021SHZDZX]. C.L. discloses support for research of this work from the Youth Innovation Promotion Association of the Chinese Academy of Sciences, and the National Postdoctoral Researcher Funding Program [Grant No. GZC20232998].

Author contributions

T.Z. and B.X. conceived the work. C.L., S.J., H.L., and X.Z. carried out experiments. C.L., S.J., H.L., X.Z., C.T.L., and T.Z. contributed to the writing of the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains Supplementary Material available at <https://doi.org/10.1038/s42003-024-07282-3>.

Correspondence and requests for materials should be addressed to Bo Xu or Tielin Zhang.

Peer review information *Communications Biology* thanks the anonymous reviewers for their contribution to the peer review of this work. Primary Handling Editor: Jasmine Pan. A peer review file is available.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025