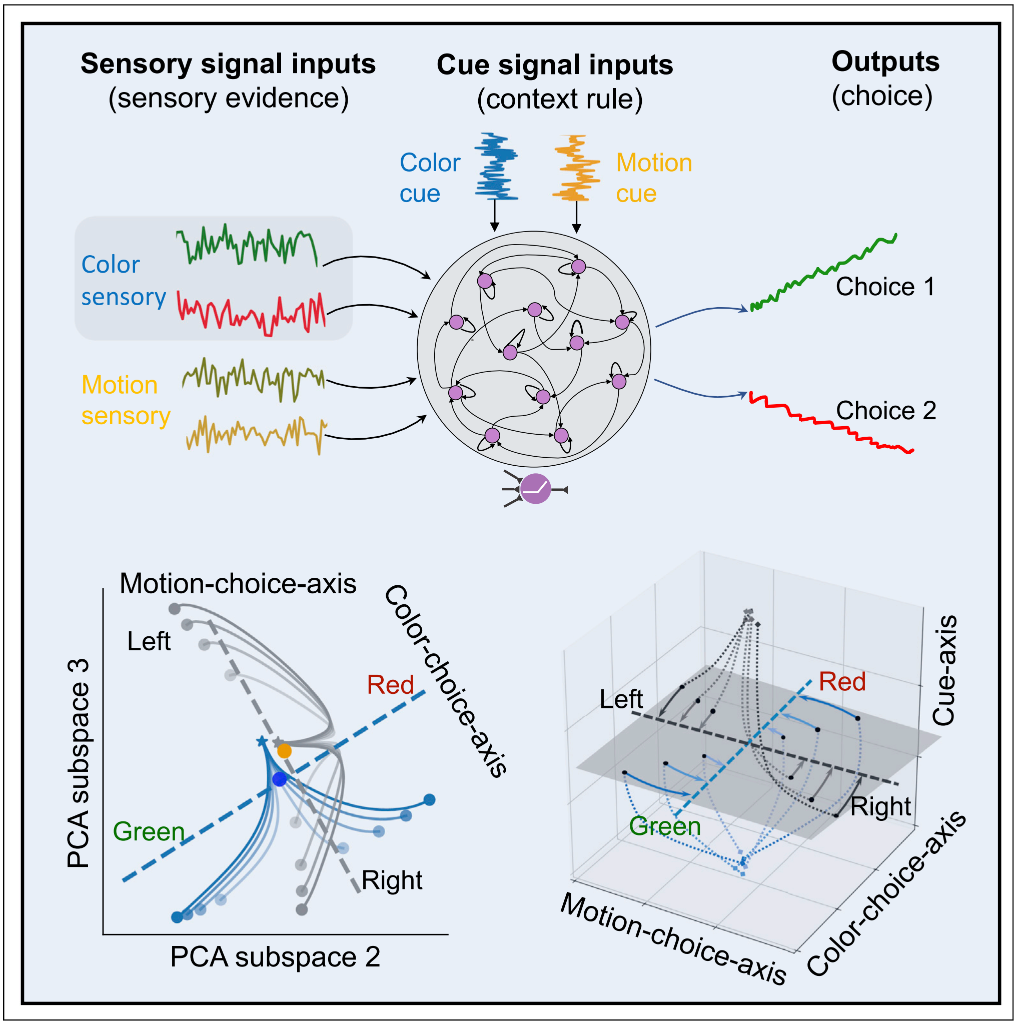
iScience

CellPress
OPEN ACCESS

Article

A geometric framework for understanding dynamic information integration in context-dependent computation



Xiaohan Zhang, Shenquan Liu, Zhe Sage Chen

mashqliu@scut.edu.cn (S.L.)
zhe.chen@nyulangone.org (Z.S.C.)

Highlights

Units with mixed selectivity emerged in context-dependent computation

Neural sequences emerged in the trained RNN during cue delay

Task-specific neural trajectories distinguished in low-dimensional subspaces

Sensory integration formed dynamic fixed points and line attractors

Zhang et al., iScience *24*, 102919
August 20, 2021 © 2021 The Author(s).
https://doi.org/10.1016/j.isci.2021.102919

CellPress
OPEN ACCESS

Article

A geometric framework for understanding dynamic information integration in context-dependent computation

Xiaohan Zhang,[1] Shenquan Liu,[1,*] and Zhe Sage Chen[2,3,*]

SUMMARY

The prefrontal cortex (PFC) plays a prominent role in performing flexible cognitive functions and working memory, yet the underlying computational principle remains poorly understood. Here, we trained a rate-based recurrent neural network (RNN) to explore how the context rules are encoded, maintained across seconds-long mnemonic delay, and subsequently used in a context-dependent decision-making task. The trained networks replicated key experimentally observed features in the PFC of rodent and monkey experiments, such as mixed selectivity, neuronal sequential activity, and rotation dynamics. To uncover the high-dimensional neural dynamical system, we further proposed a geometric framework to quantify and visualize population coding and sensory integration in a temporally defined manner. We employed dynamic epoch-wise principal component analysis (PCA) to define multiple task-specific subspaces and task-related axes, and computed the angles between task-related axes and these subspaces. In low-dimensional neural representations, the trained RNN first encoded the context cues in a cue-specific subspace, and then maintained the cue information with a stable low-activity state persisting during the delay epoch, and further formed line attractors for sensor integration through low-dimensional neural trajectories to guide decision-making. We demonstrated via intensive computer simulations that the geometric manifolds encoding the context information were robust to varying degrees of weight perturbation in both space and time. Overall, our analysis framework provides clear geometric interpretations and quantification of information coding, maintenance, and integration, yielding new insight into the computational mechanisms of context-dependent computation.

INTRODUCTION

Cognitive flexibility is an important characteristic that enables animals or humans to selectively switch between sensory inputs to generate appropriate behavioral responses (Diamond, 2013; Scott, 1962; Miyake and Friedman, 2012). This important process has been associated with various goal-directed behaviors, including multi-tasking and decision-making (Thea, 2012; Dajani and Uddin, 2015; Le et al., 2018; Pezzulo et al., 2014). Impaired cognitive flexibility has been observed among individuals with mental illnesses, such as schizophrenia, (Woodward et al., 2012; Maud et al., 2012) and those at risk for mental disorders (Murphy et al., 2012; Chamberlain et al., 2007; Vaghi et al., 2017). Therefore, identifying the computational principle underlying cognitive flexibility may improve our understanding of brain dysfunction. The prefrontal cortex (PFC) is known to contribute to cognitive flexibility, serving as the main storage of temporary working memory (WM) to represent and maintain contextual information (Baddeley, 2003; Miller, 2000; Todd et al., 2009). Neurophysiological recordings have shown that single PFC cells respond selectively to different task-related parameters (White and Wise, 1999; Eiselt and Nieder, 2016; Hyman et al., 2013; Machens et al., 2010; Rigotti et al., 2013) and the activity of PFC pyramidal neurons can maintain WM to perform context-dependent computation (Wallis et al., 2001). However, due to the heterogeneity and diversity of single-neuron responses, it remains challenging to understand how task-modulated single-neuron activities integrate task-related information to guide subsequent decision-making. To address this knowledge gap, researchers relied on population coding to understand the maintenance and manipulation of context information in decision-making tasks (Meyers et al., 2008; Cichy et al., 2014; King and Dehaene, 2014;

[1] School of Mathematics, South China University of Technology, Guangzhou, China

[2] Department of Psychiatry, Department of Neuroscience and Physiology, Neuroscience Institute, New York University Grossman School of Medicine, New York City, NY, USA

[3] Lead contact

*Correspondence: mashqliu@scut.edu.cn (S.L.), zhe.chen@nyulangone.org (Z.S.C.)

https://doi.org/10.1016/j.isci.2021.102919

Lundqvist et al., 2016). In the literature, several computational theories and analysis methods have been proposed (Wu et al., 2020; Mante et al., 2013). However, how the sensory information is integrated via dynamic population coding in a context-dependent manner remains poorly understood. Meanwhile, an intuitive and interpretable dynamical systems framework for context-dependent WM and decision making is lacking.

Recurrent neural networks (RNNs) have been widely used for modeling a wide range of neural circuits, such as the PFC and parietal cortex, in performing various cognitive tasks (Rajan et al., 2016; Hennequin et al., 2014; Sussillo et al., 2015). However, computational mechanisms of RNNs in performing those tasks remain elusive because of the black box modeling. In this paper, we trained an RNN to perform a delayed context-dependent task (Figure 1A) and proposed a geometric analysis framework to understand dynamic population coding and information integration. We found that the trained RNN captured critical physiological properties consistent with reported experimental data. Additionally, the trained RNN showed some emergent features of neuronal activity observed in the PFC, such as the mixed selectivity and sequential activity. Based on dimensionality reduction of population responses, we defined task epoch-specific subspaces and dynamic attractors during the sensory integration epoch, and showed that the context-configured network state is temporally tuned to regulate sensory integration to guide decision-making. Together, our analysis framework not only helps uncover the computational mechanisms of encoding and maintenance of context information in decision-making but also helps illustrate information integration based on interpretable geometric concepts.

RESULTS

Trained RNN for performing a delayed context-dependent integration task

We trained the RNN to perform a delayed context-dependent WM or decision-making task (Figure 1B). At each trial, the network received two types of noisy inputs: sensory stimulus and cue stimulus. The sensory input units encoded the momentary motion and color evidence toward two target directions. The cue input units encoded the contextual signal, instructing the network to discriminate a specific type of sensory input. The choice output units encoded the response direction. All units had non-negative and non-saturating firing rates to mimic the properties of biological neurons (Priebe and Ferster, 2008; Abbott and Chance, 2005).

Upon successful convergence of RNN training (Figures 1C and 1D), psychometric tests showed that the trained RNN captured critical physiological properties consistent with experimental findings (Figure 1E). For example, the trained network achieved better performance with higher color coherence stimulus in the color context, but not in the motion context; and vice versa (Figure 1E). Units in the trained RNN showed diverse firing rate profiles at different task epochs (Figure 1F). Furthermore, we analyzed the impact of the proportion of zero recurrent weights on the task performance (Figure S1A). The trained RNN exhibited a strong self-connection (Figure S1B). Additionally, recurrent weight perturbation analyses were also used to assess the stability of the trained RNN (Figure S2).

Single Unit Responses

Mixed selectivity

Mixed selectivity of PFC neurons is important for implementing complex cognitive functions, manifesting itself as an 'adaptive coding' strategy (Duncan, 2001). We found that many units of the trained RNN exhibited mixed selectivity for task-related variables (Figure 2A). A unit was said to be selective to a task-related variable if it responded differently to the values of the parameters characterizing that variable. We classified the units based on their responses to different task parameters, and found four distinct types of mixed-selective units from the trained RNN. (1) Some units (about 11.6%) exhibited mixed selectivity to task rules (both color context cue and motion context cue), such as unit 3. (2) Some units (about 11.3%) exhibited mixed selectivity to both color sensory stimuli (red and green), such as unit 8. (3) Some units (about 21.8%) exhibited mixed selectivity to both directions of coherent motion, such as unit 12. (4) Some units (about 21.6%) exhibited selectivity to both task rules and sensory stimuli. For example, unit 5 and unit 9 responded to both the color cue stimuli and color sensory stimuli. Unit 4 and unit 13 responded to both the motion cue stimuli and motion sensory stimuli.

Although single-unit activity could be tuned to mixtures of multiple task-related variables, some other units were modulated primarily by only one of the task variables (what we will call 'pure-selectivity' units). For

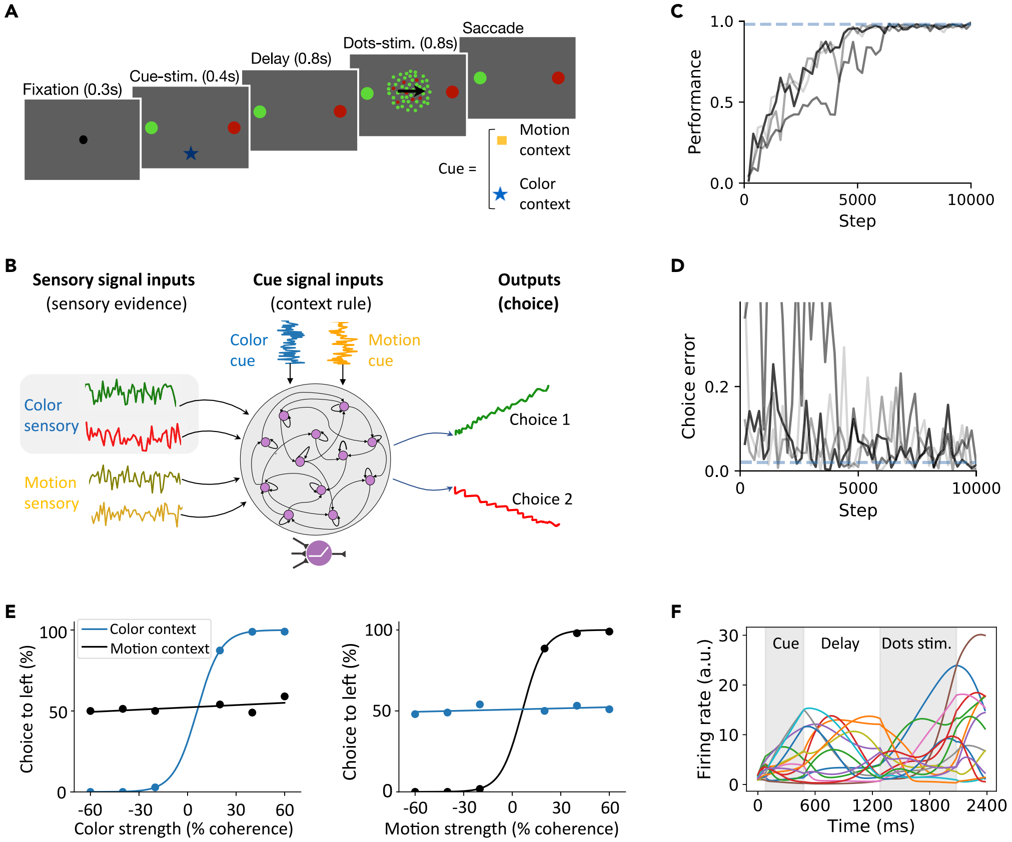


Figure 1. Trained RNN to perform the delayed context-dependent integration task

(A) Behavioral task description. A monkey was trained to discriminate, depending on the contextual cue, either the predominant color or predominant motion direction of randomly moving dots, and to further indicate its decision with a saccadic eye movement to a choice target. The cue stimulus onset determined the current context, which was characterized by different shapes and colors of the fixation point. The cue stimulus was followed by a fixed-delay epoch, and then followed by randomly moving dots stimuli. The monkey was rewarded for a saccade to the target matching the current context.

(B) Schematic of a fully connected, nonlinear RNN in context-dependent computation. The network received noisy inputs of two types: time-varying sensory stimulus and cue input. The stimulus inputs consisted of four task-relevant sensory information, each represented by an input unit that encoded the evidence for the direction of stimulus. The cue inputs consisted of two cue signals, which were represented by two input units (that indicated the current context and instructed the network to distinguish the type of stimulus). The two output channels encoded the response direction.

(C and D) RNN learning curve (C) and the performance curve (D). Training was completed once both quantities reached the convergence criterion (blue horizontal dashed lines).

(E) Psychometric curves in a delayed context-dependent integration task. The probability of a correct direction judgment is plotted as a function of color (*Left*) and motion (*Right*) coherence in color-context (blue) and motion-context (black) trials.

(F) The activities of representative units indicated by different colors. The first gray shading area indicates cue stimulus epoch and the second shading area indicates the presentation of random dots (i.e., integration of sensory stimulus epoch)

example, unit 1 was primarily selective to the color context cue, and unit 2 was selective to the motor cue. Some units only responded to sensory stimuli, such as unit 6, unit 7, unit 10, and unit 11 (Figure 2A). In the trained RNN, only a small subset of units showed activations to task variables (Figure 2B). An RNN unit was considered active if there was a time point where the instantaneous firing rate was greater than 5 Hz. Among those activated units, the number of units encoding the choice was much more than that of units encoding the context cue (Figure 2C). One possible explanation is that it was more difficult to integrate noisy sensory information than to distinguish the context information, so more units were recruited to process sensory information.

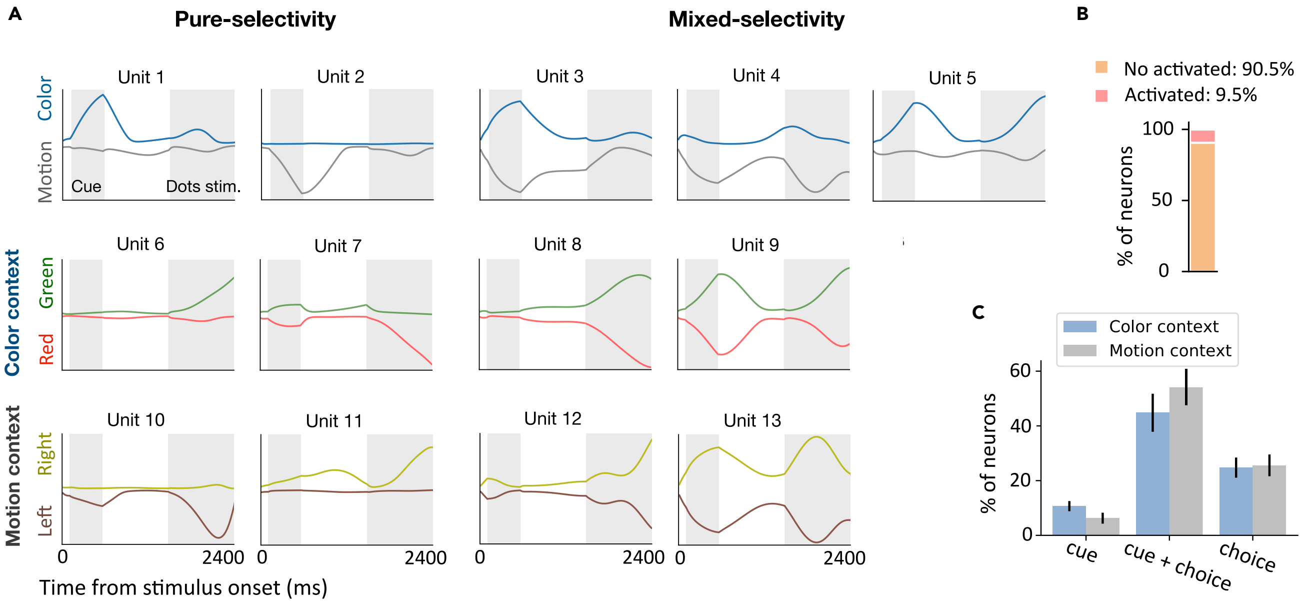


Figure 2. Single unitresponses

(A) Single unit responses under different task conditions, as indicated by different colors. Two shaded areas represent the period of contextual cue stimulus presentation and sensory stimulus presentation, respectively. *Left panel*: Responses of units showed pure selectivity to the task-related variable. Unit 1 preferred the color cue, unit 2 preferred the motion cue, unit 6 preferred green, unit 7 preferred red, unit 10 preferred the leftward direction, and unit 11 preferred the rightward direction. *Right panel*: Different task-related variables were mixed. Unit 3 showed mixed selectivity for the color cue and motion cue. Units 4 and unit 5 showed mixed selectivity for both the context cue and sensory stimulus. Unit 8 showed mixed selectivity for two task variables (green and red) for the color context. Unit 12 showed mixed selectivity for two task variables (left and right) for the motion context. Unit 9 showed mixed selectivity for three task features, such as the color context cue, green, and red sensory stimulus. Similarly, unit 13 showed mixed selectivity for the motion context cue, leftward direction, and rightward direction.

(B) The percentage of units in the trained RNN that were activated to perform a delay context-dependent decision-making task, averaging over all 16 trial conditions and 20 trained RNN models.

(C) Among all the activated units, the percentage of units that was selective to the context cue, sensory input, and their combinations in color and motion context, respectively. Error bar indicates SEM over 20 different training configurations.

We have mainly introduced four types of mixed selectivity units, with varying degrees of selective responses to different task-related variables. However, the diversity of mixed selectivity responses often resulted in response properties that were not easily interpretable. For example, in the color context, unit 9 simultaneously responded to the color cue stimulus and green signal, which also responded to the red signal. Specifically, units with such mixed selectivity behaved the same way in the same context, causing a difficulty of interpretation. This suggests that the activity of individual mixed selectivity units could not fully disambiguate the information; only when pooling information from multiple units, the ambiguity of information encoded by mixed selective units can be eliminated, supporting the necessity of population coding by ensembles of neurons (Rafael, 2015).

Population Response

We further studied the neural representation at the population level. The dynamics of population activity can be characterized through the high-dimensional state space $\mathbf{x}(t)\in\mathbb{R}^N$. The time-varying population activity can be visualized as a trajectory within the lower-dimensional subspace, and the distance between the points in the subspace reflects the population response difference.

Cue processing

To examine cue processing, we reported the dynamics of population activity throughout the cue stimulation and delay epochs. We performed principal component analysis (PCA) to identify a three-dimensional neural subspace, which was spanned by the first three principal components (PCs) that accounted for about 92% of neural activity variance (Figure 3A). During the cue presentation epoch, the network started at the same subspace and then evolved along different trajectories based on the contextual cues (Figure 3B).

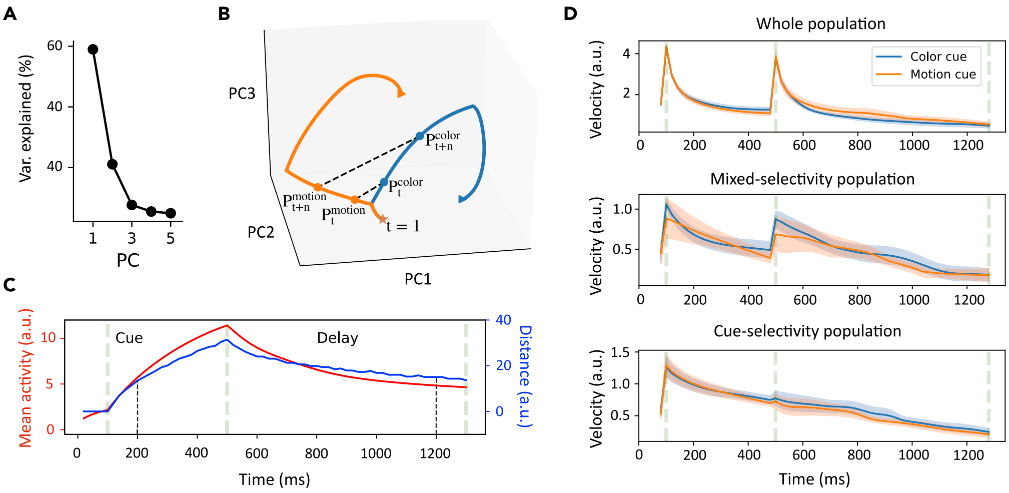


Figure 3. Neural population dynamics

(A) Ratio of explained variance of the first five PCs of neural subspace during the cue stimulus and delay epochs.

(B) Two different neural trajectories in a three-dimensional subspace during the cue stimulus and delay epochs. Orange and blue curves correspond to the motion and color contexts, respectively. The three-dimensional distance between two context-specific states at time *t* characterized the similarity of their population responses: $dist(P_t^{color}, P_t^{motion})$. In a context-specific state trajectory, we calculated the velocity, which quantify the change in trajectory position as a function of time. It is calculated by equation: $dist(P_t^{color}, P_{t+n}^{color})/n$, in which $dist(P_t^{color}, P_{t+n}^{color})$ represents the distance between states at time *t* and $t+n$.

(C) Distance between two context-specific trajectories in the cue stimulus and delay epochs as a function of time (blue curve). For comparison, the overall mean network activity (or network energy) is shown in a red curve, which is defined as the average firing rate of all activated units.

(D) The top panel plots the velocity of the temporal evolution of overall population state ($|\dot{x}|$) under color context (blue) and motion context (orange). Shaded area denotes SEM over 20 training configurations. The middle and bottom panels show the state velocity evolution of mixed-selectivity population and cue-selectivity population, respectively.

Further, the distance between two state trajectories increased (Figure 3C, blue curve), indicating that these two trajectories were continuously divergent during the cue stimulus epoch. Moreover, we defined the ''energy'' of the population activation state using the average activity of the total number of activated units (Figure 3C, gray curve).

We found that the distance curves between two cue-specific trajectories was temporally aligned with the increase of energy level during the cue stimulus and delay epochs. During the delay epoch, the network settled in a low-energy state. Moreover, the distance between two trajectories reached a peak value at the end of the cue stimulus presentation, then decreased during the delay epoch until reaching a plateau. However, the plateau value was greater than zero (a similar level as at time $t = 200$ ms), suggesting that the difference in cue-specific trajectories still existed to distinguish the context conditions.

We computed the ''velocity'' of population activity, which is defined as the change in trajectory position as the function of time (Stokes et al., 2013). In the figure illustration (Figure 3B), it can be calculated by equation $dist(P^t, P^{t+n})/n$, in which $dist(P^t, P^{t+n})$ represents the distance between states at time *t* and $t+\ n$ for a given context. Before the cue stimulus appeared, the overall population activity had a rapid acceleration to reach a peak (Figure 3D, top panel), and then the velocity of population activity gradually decayed to the pre-stimulus baseline level, suggesting that cue-specific trajectories were separated at a stable velocity in the late stage of cue-stimulus. During the delay epoch, the velocity of population activity jumped to a large value again and then dropped rapidly. The phenomenon was primarily contributed by mixed-selective units. Specifically, we plotted the respective velocities of population activity with pure-selectivity and mixed selectivity for comparison (Figure 3D). We observed that the velocity of the population that was only sensitive to the context cue decreased continuously throughout the cue stimulus and delay epochs. In contrast, for the mixed selectivity population, there was a jump point in the velocity curve at the beginning of the delay epoch. Therefore, the velocity is sensitive to change of epoch-wise population activity, and provides an informative measure of the population dynamics.

Since there is no sensory input during the delay epoch, the cue information must be stably stored as WM to guide the subsequent decision-making. A related question is how the model performance will be affected subject to weight perturbation. Motivated by the local optogenetic manipulation in animal experiments (Gray et al., 2017), we conducted weight perturbation by up- or down-scaling the recurrent connection weights with different scale values. Figure S2 illustrates a global perturbation, in which the weights of trained RNN are scaled by the same constant across all task epochs. In contrast, Figure S3 illustrates a local perturbation, in which the weights were only scaled locally in time during the delay epoch. We found that a small degree of perturbation did not affect the RNN's ability to perform tasks. Additionally, the RNN performance was more robust with local perturbation than with global perturbation.

Sequential activity

Neural sequences are emergent properties of RNNs in many cognitive tasks, which has been thought to be a common feature of population activity during a wide range of behaviors (Fiete et al., 2010; Rajan et al., 2016; Orhan and Ma, 2019; Rajakumar et al., 2021). We found that our trained RNN generated emergent sequential activity during the delay epoch (Figure 4F). To quantify the sequentiality of neural activity, we calculated the sequentiality index (SI). Briefly, the SI is defined as the sum of the entropy of the peak response time distribution of the recurrent neurons and the mean log ridge-to-background ratio of the neurons, where the ridge-to-background ratio for a given neuron is defined as the mean activity of the neuron inside a small window around its peak response time divided by its mean activity outside this window (Orhan and Ma, 2019). By virtue of random sampling and Monte Carlo statistics, we found that the trained network has a higher SI ($p< 0.017$, two-sample Kolmogorov-Smirnov test) than the untrained network (Figures 4A and 4B), suggesting that stronger sequential activity emerged from a trained network (Figure 4F).

Next, we investigated the computational mechanism that produces neural sequential activity. Similar to Rajan et al. (2016), we ordered the peak firing time of recurrent units and computed the mean and standard deviations of the recurrent weights W_{ij}^{rec}. This mean statistic was plotted as a function of the order difference ($i - j$) between the i-th and j-th units (Figures 4C and 4D). Interestingly, the connection weight of trained RNN showed an asymmetric peak, that is, the connection in 'forward' direction (*i.e.*, from earlier-peaking to later-peaking units) was strengthened more than those in 'backward' direction (*i.e.*, from later-peaking to earlier-peaking units) (Figure 4D). Therefore, this asymmetrical peak weakened the connections between temporally distant units, while strengthening the connections between temporally close units. However, this asymmetric structure was absent in the untrained network (Figure 4C), resulting in the loss of sequential activation structure (Figures 4E and S4). Put together, the asymmetric weight profile in the trained RNN could prolong responses in later-peaking units, producing the emergent sequential activity.

Rotation dynamics

The oscillatory nature of dynamical system is sometimes characterized by the so-called rotation dynamics, in which the sequential activity of neural populations can be approximated follow a rotating trajectory through their state space. The rational dynamics was initially reported in the motor cortex (Churchland et al., 2012). In recent year, researchers have uncovered rotational dynamics more broadly in other cortical areas, such as the auditory cortex (Libby and Buschman, 2021), and PFC (Aoi et al., 2020). The jPCA is a dimensionality reduction technique that finds an oscillatory structure in the data (Churchland et al., 2012). We performed jPCA on population responses in different contexts and visualized the two-dimensional projections of responses in the jPCA space (Figure 5). Interestingly, the trained RNN showed a strong rotation dynamic similar to those observed (Sussillo et al., 2015). Moreover, our result supported the findings of Lebedev et al., 2019, which showed that the rotation dynamics is essentially a "by-product" of sequential activation of population activity.

Definition of task epoch-specific subspaces and axes

To probe how neural population activity dynamically encoded task-related variables, we analyzed the population responses during six different task periods, including four single task epochs (cue stimulus epoch, delay epoch, integration of sensory stimulus epoch, and response epoch) and two cross-epoch periods (Figure 6). As expected, the correlation between population firing during these six periods varied (Figure S6).

We first performed epoch-wise PCA on the population response at single task epochs to generate the corresponding state subspaces (*i.e.*, Cue-subspace, Delay-subspace, Integ-subspace, and Resp-subspace).

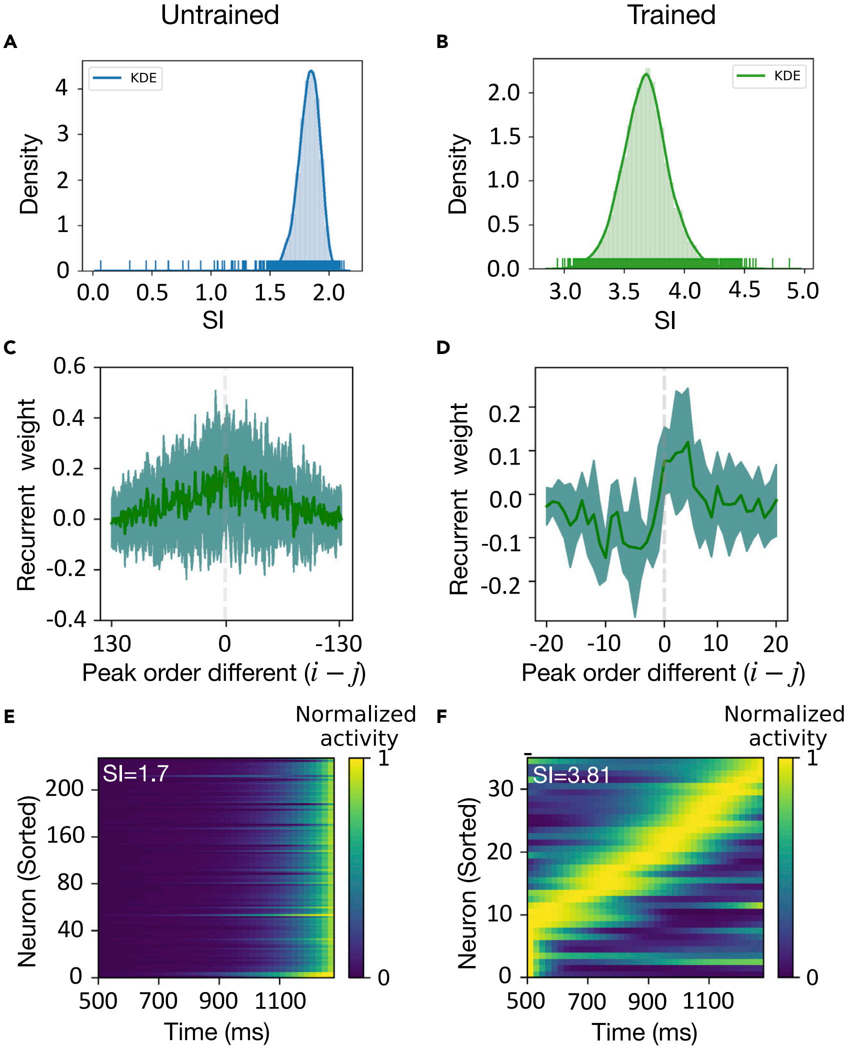


Figure 4. Sequential neural representation

(A and B) The Monte Carlo distribution of sequentiality index (SI), which was computed by 10,000 samples from repeated random sampling. SI was computed from the untrained (A) and trained (B) networks.

(C and D) Units were sorted according to their peak time in untrained (C) and trained (D) networks. The recurrent weights (W_{ij}) were plotted as a function of the peak order difference between pre- and post-synaptic units during the cue-delay epoch. A positive ($i-j$) value represents a connection from a pre-synaptic to a post-synaptic unit. A negative ($i-j$) value represents a connection from a post-to a pre-synaptic unit. Both mean (solid lines) and standard deviation (shaded regions) statistics were computed from multiple trained networks.

(E and F) Heat maps of normalized unit activity normalized by the peak response (per row) and sorted by the peak time. The activity did not appear ordered in an untrained network (E), whereas sequential activity emerged in the trained network (F).

To characterize the evolution of trajectory, we further defined four task-related axes (Table 1): the axis of color context cue (C-cue-axis), the axis of motion context cue (M-cue-axis), the axis of color choice (C-choice-axis), the axis of motion choice (M-choice-axis). The definitions of these task epoch-specific subspaces and axes provide a geometric framework for population response analyses. Next, we projected these four task-related axes onto the corresponding state subspaces and examined the neural trajectory in a subspace-specific manner.

First, we performed PCA on the population response during the cue stimulus epoch, and the first two principal components (cue-PCs) explained 93.1% of data variance (Figure S6A). This indicates that trajectories

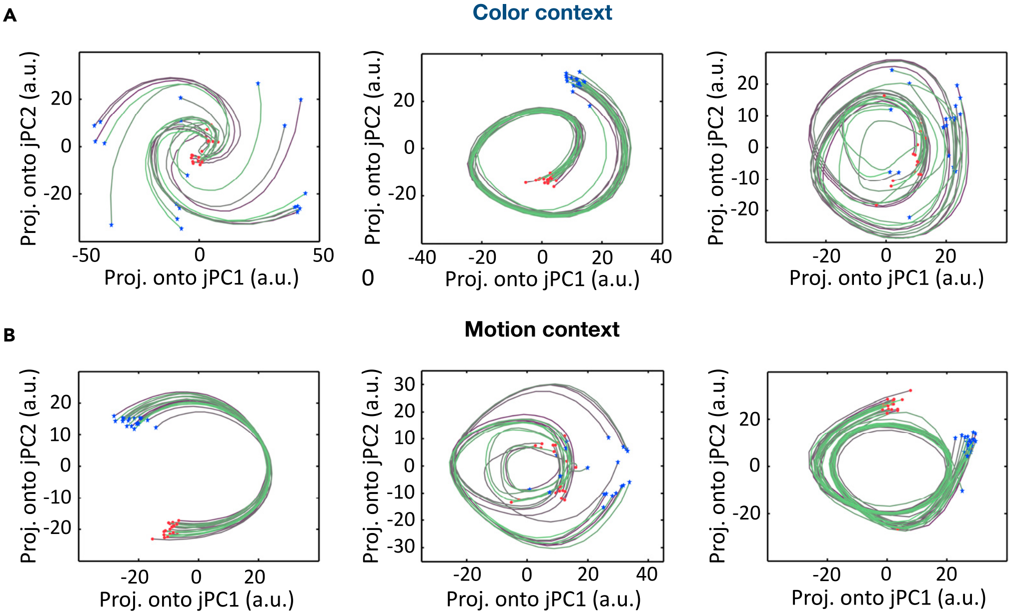


Figure 5. jPCA projectionsof the population response during the delay Epoch

(A) Three examples of two-dimensional population rotational dynamics in the color context, which corresponded to different training configurations. Each trace represents one computer simulation trial (blue star: trial start point; red circles: trial endpoint).

(B) Three examples of two-dimensional population rotational dynamics in the motion context.

in the two-dimensional subspace (denoted as Cue-subspace spanned by cue-PC1 and cue-PC2) captured the majority of variance of population responses. We had a similar finding in the Cue-subspace as shown in Figure 3B: The network state started from the same location and then produced different trajectories according to different cue stimulus conditions (Figure 7A). For a given context, the neural state evolved along a stereotypical trajectory across all sensory stimuli.

Next, we calculated the angle between Cue-subspace and four predefined axes (*i.e.*, C-cue-axis, M-cue-axis, C-choice-axis, and M-choice-axis), respectively. Geometrically, if the angle between an axis and a plane is greater than 70°, they are considered nearly orthogonal; if it is less than 20°, they are considered nearly parallel or overlapping. We found that the angle between C-cue-axis and Cue-subspace, as well as the angle between M-cue-axis and Cue-subspace, were both less than 20° (Figure 7B). This indicates that the space constructed by the C-cue-axis and M-cue-axis was overlapping with Cue-subspace. However, the angle between the C-choice-axis and Cue-subspace was around 70°-90°, as well as the angle between the M-choice-axis and Cue-subspace was both around 70°-90°. Therefore, both the C-choice-axis and M-choice-axis were nearly orthogonal with Cue-subspace. Based on these observations, we only projected C-cue-axis and M-cue-axis onto Cue-subspace (Figure 7A). We found that in the color context, all trajectories moved along the C-cue-axis but were insensitive to the M-cue-axis. Similarly, in the motion context, all trajectories moved along the M-cue-axis but were insensitive to the C-cue-axis. Furthermore, the angle between M-cue-axis and C-cue-axis mostly centered around 75°-90° (Figure S6B), suggesting that these two contextual cues were represented in two almost orthogonal subspaces.

Cue information maintenance

We further explored the question: What is the relationship between neural trajectories across task epochs? The answer to this question can help us understand how the cue information generated in the cue stimulus epoch are preserved during the delay epoch. Specifically, we performed PCA during a 1000 ms time window starting at 100 ms after the cue stimulus presentation and ending at 200 ms before the end of delay epoch. We examined the state trajectory in the two-dimensional space (denoted as Cue-Delay-subspace) spanned by the first two principal components (cue-delay-PC1, cue-delay-PC2), which explained 75% variance (Figure S6D). Meanwhile, we projected four predefined axes onto Cue-Delay-subspace (Figure 7C), and then calculated the angle between the axes and Cue-Delay-subspace (Figure 7D).

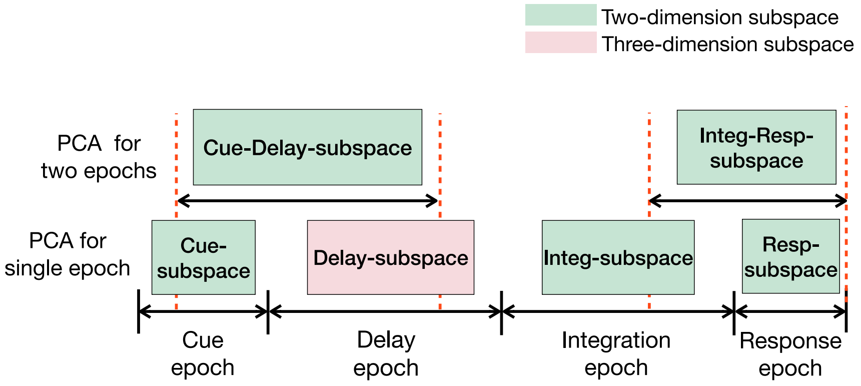


Figure 6. Illustration of distinct subspaces generated in different task Epochs

Six subspaces are obtained by performing epoch-wise PCA on population response activities during six different periods, including four single task epochs (Cue epoch, Delay epoch, Integration epoch, and Response epoch) and two cross-epochs (Cue-Delay epoch and Integration-Response epoch).

Figure 7C describes the evolution of the population dynamics in relation to cue stimulus and delay epochs. To depict the transition of population dynamics more clearly, we further divided the 1000-ms time window into two intervals. The first 300-ms interval was within the cue stimulus period, starting from the 100 ms after the cue stimulus onset until the end of cue stimulus. During this interval, the separation between two context-specific trajectories (blue for the color context, gray for the motion context) diverged along their corresponding context cue axes, and reached a maximum at the end of this interval. In the follow-up 700-ms window within the delay epoch, the separation between two context-specific trajectories remained converged along their corresponding context cue axes. Therefore, the separation of context information generated in the cue stimulus epoch was preserved in the delay epoch, even if this separability weakened during the delay epoch. Moreover, for the given context, the projection of trajectories in the C-cue-axis at $t = 100$ ms was almost the same as that at $t = 1000$ ms. This correspondence was consistent with the correspondence in the energy of population activity (Figure 3C, gray curve).

Cue-dependent processing of sensory stimulus

Furthermore, we studied how population activity responded appropriately to sensory stimulus according to the current task context. Similarly, we applied PCA to the neural activity during the sensory stimulus epoch. The first three principal components (integ-PCs) explained 81% cross-trial variance (Figure 8A), which was caused by the strength and direction of the color evidence, the strength and direction of the motion evidence, and context information (color or motion). As shown by the evolution of the population dynamics in the three-dimensional subspace (Figure 8B), each neural trajectory corresponded to a specific task condition (the blue and gray curve sets correspond to the neural trajectories in the color and motion contexts, respectively), indicating that the activity trajectories in this subspace captured the relationship between the task-related variables.

To identify the Integ-subspace, we projected the trajectories into three possible subspaces: integ-PC1 and integ-PC2, integ-PC1 and integ-PC3, and integ-PC2, and integ-PC3. Among these projections of trajectories and choice axes, we selected the subspace defined by integ-PC2 and integ-PC3 that was most geometrically meaningful in a sense that the C-choice-axis and M-choice-axis is parallel to chosen Integ-subspace. We further restricted our analysis to the two-dimensional Integ-subspace (spanned by integ-PC2 and integ-PC3). We computed the angle between four task-related axes and Integ-subspace (Figure 8C). The angle between the C-cue-axis and Integ-subspace, as well as the angle between the M-cue-axis and Integ-subspace, both centered around 70°. This means that both the C-cue-axis and M-cue-axis were orthogonal with Integ-subspace. Moreover, the C-choice-axis and M-choice-axis were nearly orthogonal (Figure S8A). Further, the angle between the C-choice-axis and Integ-subspace, as well as the angle between the M-choice-axis and Integ-subspace, were both less than 30°. This implies that the subspace spanned by the C-choice-axis and M-choice-axis was overlapping with Integ-subspace; and the projections of the C-choice-axis and M-choice-axis onto the Integ-subspace were still orthogonal (Figure 8D). Next, we investigated the mechanism of selection and integration through examining the

Table 1. The definition of geometry concepts in five neural subspaces. The matrix X varies according to different subspace.

Figure	Subspace (in two dimensions)	Geometric notion	Definition and interpretation
Figure 7A	Cue-subspace	C-cue-axis M-cue-axis	PCA on matrix $X_{cue,color}$, the first PC is defined as the C-cue-axis. PCA on matrix $X_{cue,motion}$, the first PC is defined as the M-cue-axis.
		cue-PC1 cue-PC2	PCA on population responses during the cue stimulus epoch. The first three PCs are: cue-PC1, cue-PC2, cue-PC3.
Figure 7C	Cue-Delay-subspace	cue-delay-PC1 cue-delay-PC2	PCA on population responses across two task epochs (Figure 6). The first three PCs are: cue-delay-PC1, cue-delay-PC2, cue-delay-PC3
Figure 8D	Integ-subspace	C-choice-axis M-choice-axis	PCA on matrix $X_{integ,color}$, the first PC is defined as the C-choice-axis. PCA on matrix $X_{integ,motion}$, the first PC is defined as the M-choice-axis
		integ-PC2 integ-PC3	PCA on population responses during the sensory stimulus epoch. The first three PCs are: integ-PC1, integ-PC2, integ-PC3.
Figure 9B	Resp-subspace	resp-PC2 resp-PC3	PCA on population responses during the response epoch (Figure 6). The first three PCs are: resp-PC1, resp-PC2, resp-PC3.
Figure 9D	Integ-Resp-subspace	integ-resp-PC2 integ-resp-PC3	PCA on population responses across two task epochs (Figure 6). The first three PCs are: integ-resp-PC1, integ-resp-PC2, integ-resp-PC3.

population responses in Integ-subspace in both two- (Figure 8D) and three-dimensional subspaces (Figure 8B). Several observations are in order.

First, the integration of sensory stimuli corresponded to an evolution of the neural trajectory during the presentation of stimulus signal. All trajectories started from the same starting point in three-dimensional subspace (Figure 8B), which corresponded to the initial pattern of population responses during the delay epoch. When the stimulus started, the trajectories quickly moved away from their initial state. In Integ-subspace, there was a distinct gradual evolution of the neural trajectory along the C-choice-axis and M-choice-axis (Figure 8D). Specifically, in the color (motion) context trials, the neural trajectory moved along two opposite directions, which corresponded to the two different visual targets, namely, green (left) and red (right).

Second, population responses varied according to different sensory inputs. Specifically, the neural trajectories corresponding to the color context were very different from those corresponding to the motion context, implying that sensory signals were separable at a context-dependent manner. In Figure 8D, the blue and gray curves represented neural trajectories in the color and motion contexts, respectively. In the color context, the patterns of population responses also varied with respect to different strengths and directions of the color stimulus. Therefore, neural trajectories captured multi-dimensional task-related variables, such as the context information, strength, and direction of sensory stimulus. Moreover, instead of following a straight line along the C-choice-axis or M-choice-axis, the neural trajectory formed an arc within the corresponding choice axis. The distance between the projection point of each arc onto the C-choice-axis and the color-center point (dark blue circle) reflected the strength of the corresponding color evidence while the position (two sides of the color-center point) of the projection point of each arc onto the color axis reflected the direction of the target (toward green or red). Once the stimulus signal disappeared, the network stopped to integrate the sensory evidence, yet the integrated evidence continued to be preserved along the C-choice-axis (Figures 9A and 9B). Similar discussions were also held in the motion context.

Third, population responses in the color and motion contexts occupied two different parts of subspace, which corresponded to two orthogonal-but-interesected subspaces. According to the above discussion, the neural trajectory in the color context was guided by the C-cue-axis and C-choice-axis. Therefore, the

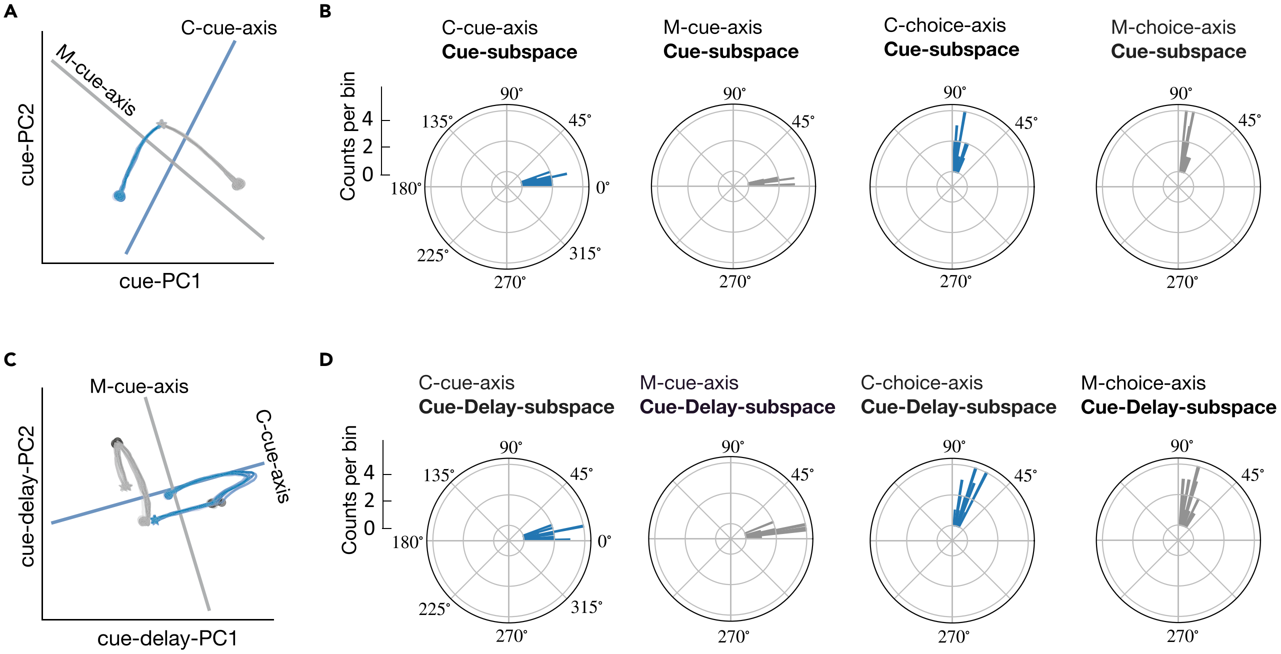


Figure 7. Visualization of neural trajectory in task subspaces

(A) Neural trajectory in the Cue-subspace during the cue stimulus epoch. Blue and gray curves correspond to the color and motion contexts, respectively. Stars and circle indicate the start and endpoints of the cue stimulus epoch, respectively. Solid lines represent the projections of the C-cue-axis (blue) and M-cue-axis (gray) in this subspace.

(B) Polar histograms of the angles between four task-related axes and Cue-subspace during the cue stimulus epoch over 20 training configurations.

(C) Neural trajectory in the combined Cue-Delay-subspace. Stars denote the neural state at 100 ms after cue stimulus onset, circles denote the beginning points of the cue-delay epoch, and dark circles denote the state at 200 ms before the end of delay.

(D) Polar histograms of the angles between four task-related axes and Cue-Delay-subspace.

population activity during the color context occupied a subspace (*i.e.*, Color-subspace spanned by the C-cue-axis and C-choice-axis). In a similar fashion, population activity in the motion context occupied the Motion-subspace spanned by the M-cue-axis and M-choice-axis. We found that the projection of all neural trajectories onto an irrelevant choice axis was almost the same at each moment (Figure 8D). This indicates that the Color-subspace and Motion-subspace were mutually orthogonal: the sensory information in the color context could not be captured by the M-choice-axis, and the sensory information in the motor context could not be captured by the C-choice-axis. Moreover, we calculated the angle between the C-cue-axis and the integ-PC1, as well as the angle between the M-cue-axis and the integ-PC1. Noting that the range of this angle was 0°～180°, which helped us detect whether the projection directions of the two cue axes on integ-PC1 were opposite. We found at each trial, one angle was greater than 90° and the other was less than 90° (Figure 8E). That is, the projection directions of the C-cue-axis and M-cue-axis on integ-PC1 were always opposite, suggesting that Color-subspace and Motion-subspace were orthogonal but intersected.

Task response

We applied PCA to the population activity during the response epoch, and found that the first three principal components (resp-PCs) explained 96% data variance (Figure S10A). The three-dimensional trajectories under different trial conditions occupied different positions in the Resp-subspace (Figure 9A). Similarly, we calculated the angle between the four task-related axes and the Resp-subspace (spanned by resp-PC2, resp-PC3). The result showed that: (1) The angle between C-cue-axis and Resp-subspace, as well as the angle between C-cue-axis and Resp-subspace, were both around 70°-90°. (2) The angle between the C-choice-axis and Resp-subspace, as well as the angle between the M-choice-axis and Resp-subspace, both centered around 30° (Figure S10B). Therefore, the near orthogonality between C-choice-axis and M-choice-axis could be maintained in Resp-subspace (Figure 9B). (3) The projection

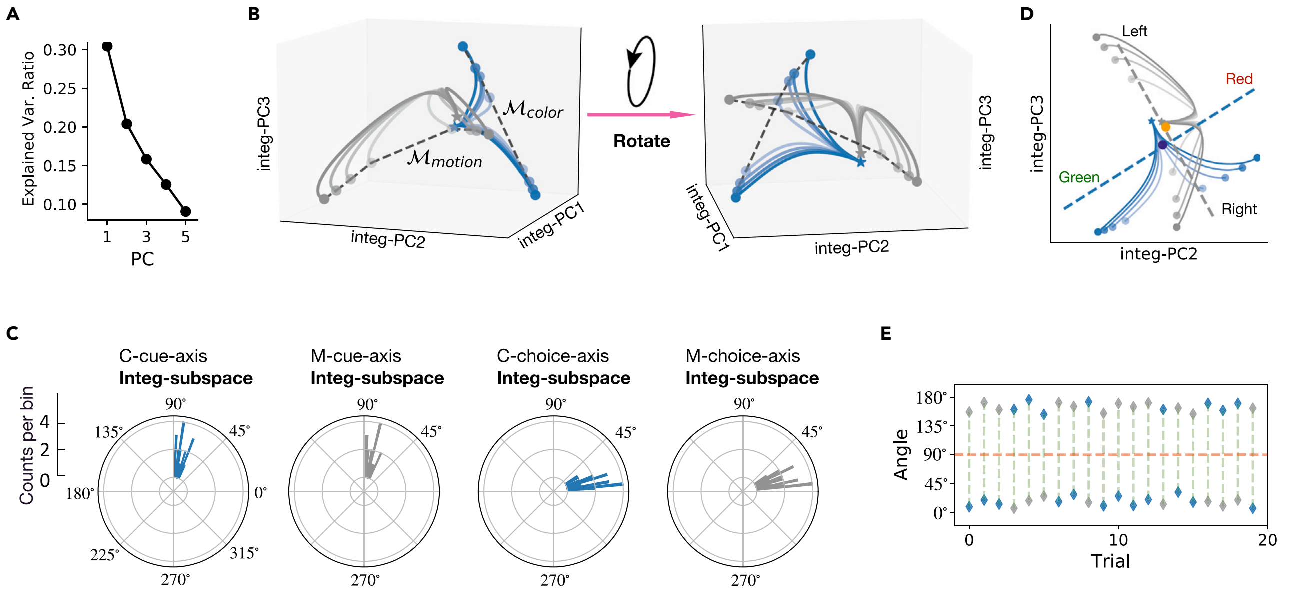


Figure 8. Population dynamics during the sensory stimulus Epoch

(A) Ratio of explained variance of the first five PCs in Integ-subspace.

(B) Neural trajectories in the three-dimensional subspace spanned by integ-PC1, integ-PC2, and integ-PC3. The blue and gray curve sets correspond to the neural trajectories in the color and motion contexts, respectively. Stars and circles indicate the start points and endpoints, respectively. The black dash curves connecting the endpoints mark manifold $\mathcal{M}_{color}$ and $\mathcal{M}_{motion}$, respectively.

(C) Polar histograms of the angles between four task-related axes and Integ-subspace.

(D) Neural trajectories in the two-dimensional subspace spanned by integ-PC2 and integ-PC3. The blue dashed line represents the projection of the C-cue-axis and the dark blue solid circle denotes the center point of the projection. The gray dashed line represents the projection of the M-cue-axis and the orange solid circle denotes the center point of the projection.

(E) The angles between integ-PC1 and the C-cue-axis (blue) or M-cue-axis (gray) in 20 trials.

of the neural trajectory on the corresponding choice axis was almost the same at each moment, suggesting that there was no sensory evidence integration during the response epoch (Figure 9B).

To examine how the neural trajectory evolved from the sensory stimulus epoch to the response epoch, we reapplied PCA to the population activity during a combined-epoch 600-ms period: starting at 500 ms after the presentation of the sensory stimulus and ending at the moment of decision. The first three PCs (integ-resp-PCs) explain 92% cross-trial variance (Figure S11B). The neural trajectory in the two-dimensional subspace (denoted as Integ-Resp-subspace, Figure 9D) had similar characteristics to that during the sensory stimulus period: First, the angle between the C-choice-axis and Integ-Resp-subspace, as well as the angle between the M-choice-axis and Integ-Resp-subspace, both centered around 30° (Figure S11A). Second, the projection directions of the C-cue-axis and M-cue-axis on integ-resp-PC1 were always opposite (Figure S11C), thus the integ-resp-PC1 fully captured the context cue information. Third, the projection of all neural trajectories on an irrelevant choice axis was the same at each moment (Figure 9D). Therefore, the population activity during this extended period also occupied two orthogonal-but-intersected subspaces.

Finally, we investigated the impact of weight perturbation on low-dimensional neural trajectories. Specifically, we perturbed the recurrent connection weights and examined the neural activity trajectory in the corresponding three-dimensional subspace. The range of local and global perturbation was restricted, and we examined the neural trajectories during the delay epoch and sensory stimulus epoch, respectively. Under a small weight perturbation, the relationship between task-related variables, such as the strength and direction of the stimulus, remained relatively robust (Figures S7 and S9). When the level of perturbation increased, the task-related information became lost gradually.

A geometric interpretation of context-dependent integration

Based upon our geometric framework and subspace analyses, we propose a geometric interpretation for context-dependent computation during WM and decision making. To further illustrate the geometric

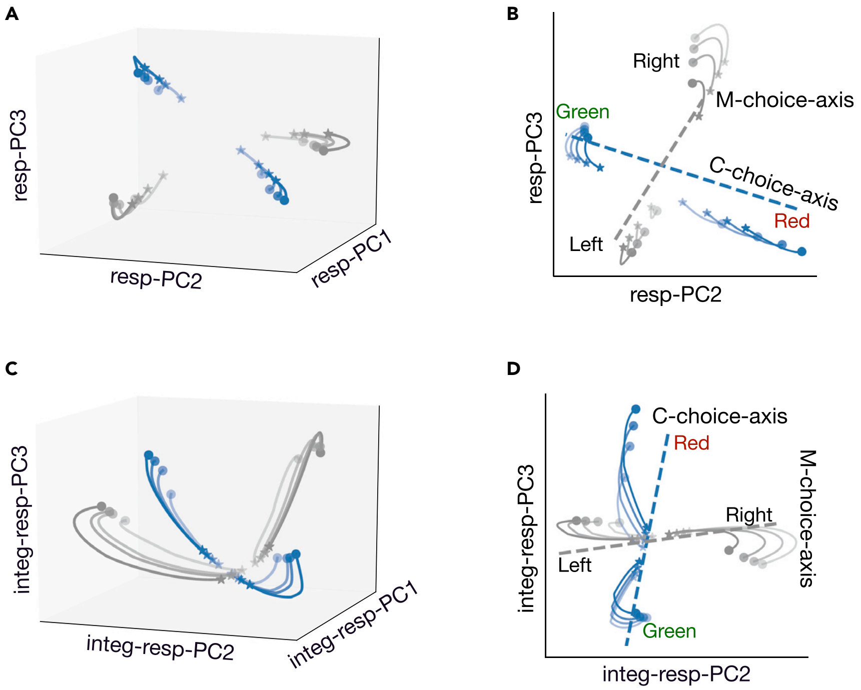


Figure 9. Visualization of neural trajectory in the response subspace

(A) Neural trajectories in the three-dimensional subspace spanned by resp-PC1, resp-PC2, and resp-PC3 during the response epoch.

(B) Same as panel A, except in two-dimensional subspace.

(C) Neural trajectories in the three-dimensional subspace spanned by integ-resp-PC2, integ-resp-PC2, and integ-resp-PC3. Each trajectory started from 500 ms after the beginning of sensory stimulus until the action response.

(D) Same as panel C, except in two-dimensional subspace.

notion, we defined dynamic attractors (such as fixed points) in the high-dimensional RNN dynamics (Sussillo and Barak, 2013).

Specifically, we used numerical optimization to minimize $q(\mathbf{x}) = \frac{1}{2}||\dot{\mathbf{x}}||^2$ to identify the fixed points and slow points. Each dot in Figure 10A represents a slow point with $q(\mathbf{x})<0.01$, and each cross represents a fixed point with $q(\mathbf{x})<0.0001$. These fixed points were further arranged along a line for a specific context (blue crosses: color context; black crosses: motion context), forming a line attractor (Figures 10A and 10B).

We have showed the end of neural trajectories during the sensory integration epoch were aligned with manifold $\mathcal{M}$ (Figure 8B); and the projection of manifold $\mathcal{M}$ in the Integ-subspace was parallel to the corresponding choice axis (Figure 8D), suggesting that the color and motion information could be captured by manifold $\mathcal{M}_{color}$ and manifold $\mathcal{M}_{motion}$, respectively. Interestingly, the line attractors were aligned near the manifold, suggesting that the integration of sensory evidence could be explained by the arrangement of fixed points: During the sensory integration epoch, the population dynamics drove context-specific trajectories to back to their attractors associated with the correct choice. That is, the population activity was attracted toward the corresponding manifold $\mathcal{M}$ of slow dynamics at the end of the sensory stimuli (Figure 8B). By projecting these slow points and fixed points onto Resp-subspace, the population trajectories evolved around the corresponding fixed points (Figure 10C). Additionally, the sensory information integrated during the sensory stimuli epoch was maintained during the response epoch.

Moreover, our analysis within the proposed geometric framework has shown that Color-subspace and Motion-subspace were orthogonal-but-intersected. Therefore, the color contextual cue triggered a dynamical process so that the relevant color evidence was integrated while ignoring the irrelevant motion evidence from the same trials because of their mutual orthogonality in the subspace. The opposite pattern was also evident in the motion context.

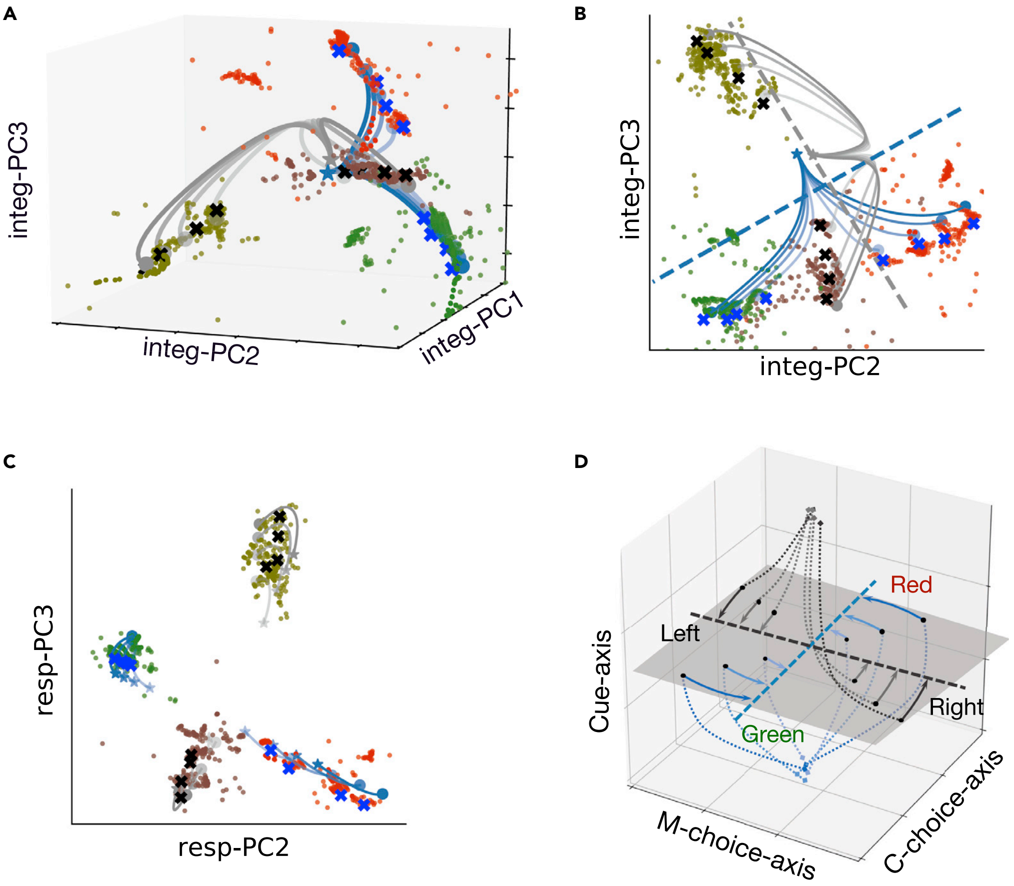


Figure 10. Visualization of dynamic attractors in neural subspace

(A) Neural trajectories in three-dimensional Integ-subspace for different sensory stimuli eventually converged to different attractors. Blue and black crosses correspond to the color and motion contexts, respectively. Dots denote slow points and different colors correspond to different trial stimuli (red dots as red stimuli, green dots as green stimuli, olive dots as leftward direction stimuli, and brown dots as rightward direction stimuli).

(B) The projection of slow points and fixed points onto Integ-subspace.

(C) The projection of slow points and fixed points onto the Resp-subspace.

(D) A schematic of neural trajectory through a three-dimensional subspace during the transition from evidence of sensory integration to choice making. Three axes include two choice axes and one context cue axis. Dotted lines represent neural trajectories during the sensory stimulus epoch, and solid lines reflect neural trajectories during a 75-ms window following the saccade onset.

Overall, we found clear geometrical structures in the task epoch-specific subspaces, which can provide (1) a new perspective to reexamine experimental data; (2) a quantitative framework to test computational hypotheses or mechanisms of context-dependent computation (3) an abstraction of theory (e.g., dynamic line attractor); for WM or decision making that supports experimental findings.

DISCUSSION

Context-dependent computation is a key hallmark for achieving cognitive flexibility. However, the computational principles underlying context-dependent WM or decision-making remains incompletely understood. We trained an RNN to perform a delayed context-dependent decision-making task, and proposed a geometric framework that helps uncover population dynamics of the trained RNN. Importantly, the trained RNN produced some emergent neurophysiological features at both single unit and population levels. The PCA and weight perturbation analysis further revealed neural representations of context-specific dynamic population coding and information integration. In low-dimensional neural subspaces, the RNN encoded the context information through the separation of neural trajectories and maintained the context information during the delay epoch. Finally, sensory integration during the decision-making period can be viewed by an evolving neural trajectory that occupies in two orthogonal-but-intersected subspaces.

Artificial RNNs are generally considered as black-box models and the underlying dynamical mechanisms are poorly understood. To unravel the black box, Sussillo and Barak (2013) proposed a dynamical systems framework to uncover the computational mechanisms of RNNs. In a similar manner, we used reverse-engineering and subspace analyses to uncover the mechanisms of population coding in the trained RNN. At the single unit level, many units exhibited mixed selectivity to different task-related variables. At the population level, sequential activation ('neural sequences') and rotational dynamics emerged from the trained RNN during the delay period.

In experimental data, neural sequential activity often emerged in temporally structured behaviors, which has been observed in the many brain regions, such as the mouse prefrontal or parietal cortices (Harvey et al., 2012; Schmitt et al., 2017). Moreover, several lines of RNN modeling work have been reported and our results of neural sequences and SI were also consistent with previous modeling (Orhan and Ma, 2019; Bi and Zhou, 2020). Notably, the experimental results reported by Mante et al. (2013) did not show sequential activity. One possible reason may be due to the fact that there was no temporal relationship between the contextual cue and the subsequent signal stimulus. Dimensionality reduction-based subspace analyses have proven useful to uncover temporal dynamics of RNNs (Kobak et al., 2016; Kao, 2019), and intuitive geometric notions can further reveal the orthgonality of task variables in the neural subspace (Bi and Zhou, 2020).

Our analyses demonstrated that the context cue triggered the separation of neural trajectories in a low-dimensional subspace, and the dynamic system was in a high-activity state. During the delay epoch, the population activity decayed to a stable low-activity state while maintaining the context separation. During the stimulus signal presentation, we found two main features of population responses: (1) The population responses exhibited different patterns to varying sensory inputs. (2) The population response in the color and motion contexts occupied two orthogonal-but-intersected subspaces. These features of population responses can be summarized schematically in Figure 10D, which provide fundamental constraints on the mechanisms of context-dependent computation in flexible cognitive tasks.

RNNs for understanding computational mechanisms of brain functions

Due to its powerful computational capabilities, RNNs exhibit complex dynamics similar to experimental findings despite their oversimplification and abstraction of biological brain operation (nonlinear, feedback, and distributed computation) (Wolfgang et al., 2002; Sussillo and Abbott, 2009; Buonomano and Maass, 2009; Mante et al., 2013). In an analogue to animal behavioral training, RNNs can perform a wide range of cognitive tasks with supervised learning and labeled examples, including WM (Barak et al, 2010, 2013; Rajan et al., 2016), motor control (Laje and Buonomano, 2013; Hennequin et al., 2014), and decision-making (Song et al., 2016; Sussillo et al., 2015; Thomas, 2017; Yang et al., 2019; Aoi et al., 2020). The activity and network connectivity of the trained RNN can be accessed with specific perturbation strategies to help reveal underlying computational mechanisms. For example, the trained RNN can discover features of structural and functional connectivity that support robust transient activities bin a match-to-category task (Chaisangmongkon et al., 2017). Orhan and Ma (2019) identified the circuit-related and task-related factor that generates the sequential or persistent activity by training the RNN to perform various WM tasks. Goudar and Buonomano (2018) demonstrated that time-varying sensory and motor patterns can be stored as neural trajectories within the RNN, helping us understand the time-warping codes in the brain. RNNs have also been incorporated with more biological features, such as Dale's principle, which serve as a valuable platform for generating or testing new hypotheses (Song et al., 2016; Xue et al., 2021; Rajakumar et al., 2021).

In many brain regions, information is encoded by the activity patterns of the neuronal population. A dynamical system view of population activity has become increasingly prevalent in neuroscience (Churchland et al., 2012; Sauerbrei et al., 2020; Shreya et al., 2020). With the help of dimensionality reduction, the population activity over time corresponds to neural trajectories in a low-dimensional subspace (Churchland et al., 2012; Kobak et al., 2016). Although we have focused our RNN modeling on a cognitive task, our geometric framework and subspace analysis can be applied to investigate other brain areas or brain functions, such as the premotor cortex or primary motor cortex in various motor tasks (Elsayed et al., 2016; Kao, 2019; Sussillo et al., 2015) and the striatum (Zhou et al., 2020).

Relation to existing working memory theories

In context-dependent computational tasks, in order to decide which type of subsequent sensory inputs to be integrated, the information about the context rule needs to be preserved as WM across different task epochs. In the literature, various theories and computational models for WM have been developed. The classic models suggest that the PFC support WM via the stable, persistent activity of stimulus-specific neurons (Wang, 2001; Miller and Cohen, 2001; Miller, 2000), which bridges the gap between the memory representation of the context cue and sensory stimulus epochs. However, some experimental observations show that neural population coding during WM is dynamic (Barak et al., 2010; Spaak et al., 2017). Classical and new theoretical models in WM have been reviewed and tested in light of recent experimental findings (Lundqvist et al., 2016; Miller et al., 2018). Similar to other modeling efforts (Orhan and Ma, 2019; Xue et al., 2021), the single-unit response in our computer simulations showed strong temporal dynamics rather than persistent activity during the delay epoch. Therefore, our model supports the finding of experimental studies that information is often stored in dynamic population codes during WM. However, the WM is probably more complex than a simple theory that can explain everything, and neural representations (e.g., persistent vs. oscillatory dynamics) may highly depend on the specific task and recording area/technique. Moreover, unlike previous studies that suggested a dissociation between the stimulus-driven response and subsequent delay activity in the PFC (Barak et al., 2010; Meyers et al., 2008), our results showed the opposite phenomenon. In fact, the context information encoded by population activity during the cue stimulus epoch was maintained by low energy activity (Figure 3C) and sequential representation (Figure 4F) during the delay epoch.

In the trained RNN, population response is highly dynamic during both the cue stimulus and the delay epoch (Figure 3C). One possible explanation for the dynamic activity during the early part of the delay epoch is that it reflects the transformation from a transient context cue input into a stable WM representation, and this transformation is first contained in a high-energy dynamic trajectory in the state space, and then the system returns to a low-energy state. In the synaptic theory of WM (Mongillo et al., 2008), memories can be maintained as a pattern of synaptic weights, and neural activity changes synaptic efficacy, leaving a synaptic memory trace via short-term synaptic plasticity. This WM model suggests that the previous cue stimuli may be recovered from the network architecture, allowing for an energy-saving mode of short-term memory, rather than relying on the maintenance of high-energy persistent activity. Based on this synaptic theory of WM, Stokes (2015) have predicted that WM representations should be stationary and have low energy. Our RNN results are also consistent with this prediction.

Limitations of the study

There are several limitations of our work. First, neurons in the mammalian cerebral cortex have diverse cell types and follow Dale's principle—that is, they have excitatory and inhibitory effects on their postsynaptic neurons. However, such biological constraints were not considered in our current rate-based RNN model. Second, we trained the RNN using a standard gradient-based back-propagation algorithm, whereas synaptic plasticity in the brain is known to use Hebbian plasticity or spike-timing dependent plasticity (STDP). A combination of unsupervised and reinforcement learning algorithms to train RNNs would be more biologically plausible in modeling neuronal population dynamics. Third, finding line attractors in high-dimensional RNN dynamics relies on numerical methods and remains challenging. Development of efficient subspace methods for identifying dynamical attractors would be the subject of our future research direction.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - ○ Lead contact
  - ○ Materials availability
  - ○ Data and code availability

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2021.102919.

ACKNOWLEDGMENTS

This work was supported by a grant (No. 11172103) from the National Natural Science Foundation of China, China (S.L.).

AUTHOR CONTRIBUTIONS

X.Z., S.L. and Z.S.C. designed the experiment. X.Z. performed all experiments and analyses. X.Z. and Z.S.C. wrote the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: March 15, 2021
Revised: May 25, 2021
Accepted: July 27, 2021
Published: August 20, 2021

REFERENCES

Abbott, L.F., and Chance, F.S. (2005). Drivers and modulators from push-pull and balanced synaptic input. Prog. Brain Res. *149*, 147–155.

Aoi, M.C., Mante, V., and Pillow, J.W. (2020). Prefrontal cortex exhibits multidimensional dynamic encoding during decision-making. Nat. Neurosci. *23*, 1410–1420.

Baddeley, A. (2003). Working memory: looking back and looking forward. Nat. Rev. Neurosci. *4*, 829–838.

Barak, O., Tsodyks, M., and Romo, R. (2010). Neuronal population coding of parametric working memory. J. Neurosci. *30*, 9424–9430.

Barak, O., Sussillo, D., Romo, R., Tsodyks, M., and Abbott, L.F. (2013). From fixed points to chaos: three models of delayed discrimination. Prog. Neurobiol. *103*, 214–222.

Bi, Z., and Zhou, C. (2020). Understanding the computation of time using neural network models. Proc. Natl. Acad. Sci. U S A *117*, 10530–10540.

Buonomano, D.V., and Maass, W. (2009). State-dependent computations: spatiotemporal processing in cortical networks. Nat. Rev. Neurosci. *10*, 113–125.

Chaisangmongkon, W., Swaminathan, S.K., Freedman, D.J., and Wang, X.-J. (2017). Computing by robust transience: how the fronto-parietal network performs sequential, category-based decisions. Neuron *93*, 1504–1517.

Chamberlain, S.R., Fineberg, N.A., Menzies, L.A., Blackwell, A.D., Bullmore, E.T., Robbins, T.W., and Sahakian, B.J. (2007). Impaired cognitive flexibility and motor inhibition in unaffected first-degree relatives of patients with obsessive-compulsive disorder. Am. J. Psychiatry *164*, 335–338.

Churchland, M.M., Cunningham, J.P., Kaufman, M.T., Foster, J.D., Nuyujukian, P., Ryu, S.I., and Shenoy, K.V. (2012). Neural population dynamics during reaching. Nature *487*, 51–56.

Cichy, R.M., Pantazis, D., and Oliva, A. (2014). Resolving human object recognition in space and time. Nat. Neurosci. *17*, 455–462.

Dajani, D.R., and Uddin, L.Q. (2015). Demystifying cognitive flexibility: implications for clinical and developmental neuroscience. Trends Neurosci. *38*, 571–578.

Diamond, A. (2013). Executive functions. Annu. Rev. Psychol. *64*, 135–168.

Duncan, J. (2001). An adaptive coding model of neural function in prefrontal cortex. Nat Rev Neurosci *2*, 820–829.

Eiselt, A.-K., and Nieder, A. (2016). Single-cell coding of sensory, spatial and numerical magnitudes in primate prefrontal, premotor and cingulate motor cortices. Exp. Brain Res. *234*, 241–254.

Elsayed, G.F., Lara, A.H., Kaufman, M.T., Churchland, M.M., and Cunningham, J.P. (2016). Reorganization between preparatory and movement population responses in motor cortex. Nat. Commun. *7*, 13239.

Fiete, I.R., Senn, W., Wang, C.Z.H., and Hahnloser, R.H.R. (2010). Spike-time-dependent plasticity and heterosynaptic competition organize networks to produce long scale-free sequences of neural activity. Neuron *65*, 563–576.

Goudar, V., and Buonomano, D.V. (2018). Encoding sensory and motor patterns as time-invariant trajectories in recurrent neural networks. Elife *7*, e31134.

Gray, D.T., Smith, A.C., Burke, S.N., Gazzaley, A., and Barnes, C.A. (2017). Attentional updating and monitoring and affective shifting are impacted independently by aging in macaque monkeys. Behav. Brain Res. *322*, 329–338.

Harvey, C.D., Coen, P., and Tank, D.W. (2012). Choice-specific sequences in parietal cortex during a virtual-navigation decision task. Nature *484*, 62–68.

Hennequin, G., Vogels, T.P., and Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. Neuron *82*, 1394–1406.

Higham, N.J. (1986). Computing the polar decomposition with applications. SIAM J. Sci. Stat. Comput. *7*, 1160–1174.

Hyman, J.M., Whitman, J., Emberly, E., Woodward, T.S., and Seamans, J.K. (2013). Action and outcome activity state patterns in the anterior cingulate cortex. Cereb. Cortex *23*, 1257–1268.

Kao, J.C. (2019). Considerations in using recurrent neural networks to probe neural dynamics. J. Neurophysiol. *122*, 2504–2521.

King, J.R., and Dehaene, S. (2014). Characterizing the dynamics of mental representations: the temporal generalization method. Trends Cogn. Sci. *18*, 203–210.

Kingma, D.P., and Ba, J.L. (2015). Adam: a method for stochastic optimization. Proc. Int. Conf. Learn. Representations (Iclr).

Kobak, D., Brendel, W., Constantinidis, C., Feierstein, C.E., Kepecs, A., Mainen, Z.F., Qi, X.-L., Romo, R., and Machens, C.K. (2016). Demixed principal component analysis of neural population data. Elife *5*, e10989.

Laje, R., and Buonomano, D.V. (2013). Robust timing and motor patterns by taming chaos in recurrent neural networks. Nat. Neurosci. *16*, 925–933.

Le, X., Caroline, B., Laetitia, H.-B., and Peter, S. (2018). Regulation of striatal cells and goal-directed behavior by cerebellar outputs. Nat. Commun. *9*, 1–14.

Lebedev, M.A., Ossadtchi, A., Mill, N.A., Urpì, N.A., Cervera, M.R., and Nicolelis, M.A.L. (2019). Analysis of neuronal ensemble activity reveals the pitfalls and shortcomings of rotation dynamics. Sci. Rep. *9*, 18978.

Libby, A., and Buschman, T.J. (2021). Rotational dynamics reduce interference between sensory and memory representations. Nat. Neurosci. *24*, 715–726.

Lundqvist, M., Rose, J., Herman, P., Brincat, S.L., Buschman, T.J., and Miller, E.K. (2016). Gamma and beta bursts underlie working memory. Neuron *90*, 152–164.

Machens, C.K., Romo, R., and Brody, C.D. (2010). Functional, but not anatomical, separation of 'what' and 'when' in prefrontal cortex. J. Neurosci. *30*, 350–360.

Mante, V., Sussillo, D., Shenoy, K.V., and Newsome, W.T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. Nature *7*, 78–84.

Maud, C.-L., Anick, C., Karyne, A., Jean-Pierre, R., and Guy, B. (2012). Theory of mind and context processing in schizophrenia: the role of cognitive flexibility. Psychiatry Res. *200*, 184–192.

Meyers, E.M., Freedman, D.J., Kreiman, G., Miller, E.K., and Poggio, T. (2008). Dynamic population coding of category information in inferior temporal and prefrontal cortex. J. Neurophysiol. *100*, 1407–1419.

Miller, E.K. (2000). The prefrontal cortex and cognitive control. Nat. Rev. Neurosci. *1*, 59–65.

Miller, E.K., and Cohen, J.D. (2001). An integrative theory of prefrontal cortex function. Annu. Rev. Neurosci. *24*, 167–202.

Miller, E.K., Lundqvist, M., and Basto, A.M. (2018). Working memory 2.0. Neuron *100*, 463–475.

Miyake, A., and Friedman, N.P. (2012). The nature and organization of individual differences in executive functions: four general conclusions. Curr. Dir. Psychol. Sci. *21*, 8–14.

Mongillo, G., Barak, O., and Tsodyks, M. (2008). Synaptic theory of working memory. Science *319*, 1543–1546.

Murphy, F.C., Michael, A., and Sahakian, B.J. (2012). Emotion modulates cognitive flexibility in patients with major depression. Psychol. Med. Lond. *42*, 1373–1382.

Nemati, N., Zakizadeh, B., Tojari, F., and Afshardous, M. (2014). The comparison of general health in athletic and nonathletic elderly. Adv. Environ. Biol. *8*, 1074–1076.

Orhan, A.E., and Ma, W.J. (2019). A diverse range of factors affect the nature of neural representations underlying short-term memory. Nat. Neurosci. *22*, 275–283.

Pezzulo, G., van der Meer, M.A., Lansink, C.S., and Pennartz, C.M. (2014). Internally generated sequences in learning and executing goal-directed behavior. Trends Cogn. Sci. *18*, 647–657.

Priebe, N.J., and Ferster, D. (2008). Inhibition, spike threshold, and stimulus selectivity in primary visual cortex. Neuron *57*, 482–497.

Rafael, Y. (2015). From the neuron doctrine to neural networks. Nat. Rev. Neurosci. *16*, 487–497.

Rajakumar, A., Rinzel, J., and Chen, S.Z. (2021). Stimulus-driven and spontaneous dynamics in excitatory-inhibitory recurrent neural networks for sequence representation. Neural Comput. *33* in press.

Rajan, K., Harvey, C.D., and Tank, D.W. (2016). Recurrent network models of sequence generation and memory. Neuron *90*, 128–142.

Rigotti, M., Barak, O., Warden, M.R., Wang, X.-J., Daw, N.D., Miller, E.K., and Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. Nature *497*, 585–590.

Sauerbrei, B.A., Guo, J.Z., Cohen, J.D., Mischiati, M., Guo, W., Kabra, M., Verma, N., Mensh, B., Branson, K., and Hantman, A.W. (2020). Cortical pattern generation during dexterous movement is input-driven. Nature *577*, 386–391.

Schmitt, L.I., Wimmer, R.D., Nakajima, M., Happ, M., Mofakham, S., and Halassa, M.M. (2017). Thalamic amplification of cortical connectivity sustains attentional control. Nature *545*, 219–223.

Scott, W.A. (1962). Cognitive complexity and cognitive flexibility. Sociometry *25*, 405–414.

Shreya, S., Sridevi, S.V., and Dahleh, M. (2020). Performance limitations in sensorimotor control: trade-offs between neural computation and accuracy in tracking fast movements. Neural Comput. *32*, 865–886.

Song, H.F., Yang, G.R., and Wang, X.-J. (2016). Training excitatory-inhibitory recurrent neural networks for cognitive tasks: a simple and flexible framework. PLoS Comput. Biol. *12*, e1004792.

Spaak, E., Watanabe, K., Funahashi, S., and Stokes, M.G. (2017). Stable and dynamic coding for working memory in primate prefrontal cortex. J. Neurosci. *37*, 6503–6516.

Stokes, M.G. (2015). Activity-silent working memory in prefrontal cortex: a dynamic coding framework. Trends Cogn. Sci. *19*, 394–405.

Stokes, M.G., Kusunoki, M., Sigala, N., Nili, H., Gaffan, D., and Duncan, J. (2013). Dynamic coding for cognitive control in prefrontal cortex. Neuron *78*, 364–375.

Sussillo, D., and Abbott, L.F. (2009). Generating coherent patterns of activity from chaotic neural networks. Neuron *63*, 544–557.

Sussillo, D., and Barak, O. (2013). Opening the blackbox: low-dimensional dynamics in high-dimensional recurrent neural networks. Neural Comput. *25*, 626–649.

Sussillo, D., Churchland, M.M., Kaufman, M.T., and Shenoy, K.V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. Nat. Neurosci. *18*, 1025–1033.

Thea, I. (2012). Exploring the nature of cognitive flexibility. New Ideas Psychol. *30*, 190–200.

Thomas, M. (2017). Biologically plausible learning in recurrent neural networks reproduces neural dynamics observed during cognitive tasks. Elife *6*, e20899.

Todd, B., Jessica, P., Hannah, L., and Deanna, B. (2009). Flexible neural mechanisms of cognitive control with human prefrontal cortex. Proc. Natl. Acad. Sci. U S A *106*, 7351–7356.

Vaghi, M.M., Velrtes, P.E., Kitzbichler, M.G., Apergis-Schoute, A.M., van der Flier, F.E., Fineberg, N.A., Sule, A., Zaman, R., Voon, V., Kundu, P., et al. (2017). Specific frontostriatal circuits for impaired cognitive flexibility and goal-directed planning in obsessive-compulsive disorder: evidence from resting-state functional connectivity. Biol. Psychiatry *81*, 708–717.

Wallis, J., Anderson, K., and Miller, E. (2001). Single neurons in prefrontal cortex encode abstract rules. Nature *411*, 953–956.

Wang, X. (2001). Synaptic reverberation underlying mnemonic persistent activity. Trends Neurosci. *24*, 455–463.

Wang, J., Narain, D., Hosseini, E.A., and Jazayeri, M. (2018). Flexible timing by temporal scaling of cortical responses. Nat. Neurosci. *21*, 102–110.

White, I.M., and Wise, S.P. (1999). Rule-dependent neuronal activity in the prefrontal cortex. Exp. Brain Res. *126*, 315–335.

Wolfgang, M., Thomas, N., and Henry, M. (2002). Real-time computing without stable states: a new framework for neural computation based on perturbations. Neural Comput. *14*, 2531–2560.

Woodward, N.D., Karbasforoushan, H., and Heckers, S. (2012). Thalamocortical dysconnectivity in schizophrenia. Am. J. Psychiatry *169*, 1092–1099.

Wu, Z., Litwin-Kumar, A., Shamash, P., Taylor, A., Axel, R., and Shadlen, M.N. (2020). Context-dependent decision making in a premotor circuit. Neuron *106*, 316–328.

Xue, X., Halassa, M.M., and Chen, Z.S. (2021). Spiking Recurrent Neural Networks Represent Task-Relevant Neural Sequences in Rule-dependent Computation. www.biorxiv.org/content/10.1101/2021.01.21.427464v1.

Yang, G.R., Joglekar, M.R., Song, H.F., Newsome, W.T., and Wang, X.-J. (2019). Task representations in neural networks trained to perform many cognitive tasks. Nat. Neurosci. *22*, 297–306.

Zhou, S., Masmanidis, S.C., and Buonomano, D.V. (2020). Neural sequences as an optimal dynamical regime for the readout of time. Neuron *108*, 651–658.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Software and algorithms		
Python	Open source	www.python.org
Sequentiality index	Orhan and Ma (2019)	https://github.com/eminorhan/recurrent-memory
Fixed point analysis	Sussillo and Barak (2013)	https://github.com/elipollock/EMPJ
Network structure	Yang et al. (2019)	https://github.com/gyyang/multitask
Dimensionality reduction and sequence visualization	Bi and Zhou (2020)	https://github.com/zedongbi/IntervalTiming
Deposited code	This study	https://github.com/Xhan-Zhang/contextInteg

RESOURCE AVAILABILITY

Lead contact

Further information and requests for data should be directed to and will be fulfilled by the Lead Contact, Zhe S. Chen (zhe.chen@nyulangone.org).

Materials availability

The study did not generate new reagents.

Data and code availability

All code has been deposited and is publicly available on GitHub (https://github.com/Xhan-Zhang/contextInteg)

METHOD DETAILS

Network structure

We constructed an RNN network of $N = 256$ fully interconnected neurons described by a standard firing-rate model. The continuous-time dynamics of the RNN are governed by the following equations

$$\begin{aligned} \tau \dot{\mathbf{x}} &= -\mathbf{x} + \mathbf{W}^{rec}\mathbf{r} + \mathbf{W}^{in}\mathbf{u} + \mathbf{b} + \sqrt{2\tau\sigma_{rec}^2}\xi, \\ \mathbf{r} &= f(\mathbf{x}) \end{aligned} \qquad \text{(Equation 1)}$$

where $\mathbf{x}$, $\mathbf{r}$, and $\mathbf{u}$ represent the synaptic current, firing rate, and network input, respectively; $\tau =$ 20 ms is time constant, which mimics the synaptic dynamic on the basis of NMDA receptors; $\mathbf{b}$ is the background input; ξ are independent zero-mean Gaussian white noise scaled by $\sigma_{rec} = 0.05$, which represent the intrinsic noise. The firing rate $\mathbf{r}$ is related to the corresponding current $\mathbf{x}$ by a Softplus transfer function $f(x) = \log(1 + \exp(x))$, which maps every input currents to a positive firing rate. The output of the network is a linear mapping between the firing rate and a readout synaptic weight:

$$\mathbf{z} = \mathbf{W}^{out}\mathbf{r} + \mathbf{b}^{out} \qquad \text{(Equation 2)}$$

where $\mathbf{W}^{in}$, $\mathbf{W}^{rec}$, and $\mathbf{W}^{out}$ are the input weight, recurrent weight, and output weight, respectively. We used Euler's method to discretize the continuous-time equation and derive the discrete-time version

$$\begin{aligned} \tau \dot{\mathbf{x}}_t &= (1-\alpha)\mathbf{x}_{t-1} + \alpha\left(\mathbf{W}^{rec}\mathbf{r}_{t-1} + \mathbf{W}^{in}\mathbf{u}_t + \mathbf{b} + \sqrt{2\alpha^{-1}\sigma_{rec}^2}\,\xi\right) \\ \mathbf{r}_t &= f(\mathbf{x}_t) \end{aligned} \qquad \text{(Equation 3)}$$

where $\xi \sim \mathcal{N}(0,1)$ was drawn from a standard normal distribution; $\alpha = \Delta t/\tau$, and Δt is time step. In our study, we set $\Delta t = 20$ ms. Similar to the previous computational models (Wang et al., 2018; Orhan and Ma, 2019), we used $\alpha = 1$. Furthermore, we assumed that the RNN received two types of noisy input: rule-specific input $\mathbf{u}_{rule}$ and stimulus-specific input $\mathbf{u}_{stim}$:

$$\begin{aligned}\mathbf{u} &= (\mathbf{u}_{rule}, \mathbf{u}_{stim}) + \mathbf{u}_{noise} \\ \mathbf{u}_{noise} &\sim \sqrt{2\sigma_{in}^2}\mathcal{N}(0,1)\end{aligned} \quad \text{(Equation 4)}$$

where σ_{in} denotes the standard deviation of input noise. We set $\sigma_{rec} = 0.05$, $\sigma_{in} = 0.01$ in our computer simulations.

Task description

The task is schematized in Figure 1A, which is inspired and modified from a context-dependent task performed by macaque monkeys (Mante et al., 2013). The network received pulses from three input channels. The first channel consisted of the context cue stimulus, which contained two units encoding different context information. The other two channels consisted of two different sensory stimuli modalities. One channel represented the color sensory stimulus, which contained two units encoding the color stimulus strength $\gamma_{color,1}$, $\gamma_{color,2}$. The other channel represented the motion sensory stimulus, which encoded the motion stimulus strength $\gamma_{motion,1}$, $\gamma_{motion,2}$. The stimulus strengths were determined by the coherence for the color modality and motion modality (c_{color}, c_{motion}), and set as follows:

$$\gamma_{color,1} = \overline{\gamma} + c, \quad \gamma_{color,2} = \overline{\gamma} - c \quad \text{(Equation 5)}$$

A similar equation held for the motion modality. $\overline{\gamma}$ denotes the average strength of the two color stimuli, which was drawn from a uniform distribution $\overline{\gamma} \sim \mathcal{U}(0.8, 1.2)$ (where $\mathcal{U}(a, b)$ represents a uniform distribution between a and b). Coherence c measured the strength difference of these two stimuli, which was uniformly distributed as

$$c \sim \mathcal{U}(-0.08, -0.04, -0.02, -0.01, 0.01, 0.02, 0.04, 0.08) \quad \text{(Equation 6)}$$

The task consisted of distinct epochs. For each trial, a fixation epoch was present before the stimulus presentation. It was followed by the context cue stimulus epoch that lasted $T_{stim1} = 400$ ms. After a delay epoch (with duration of $T_{delay} = 800$ ms), the stimulus signal was presented in the second stimulus epoch with a duration of $T_{stim2} = 800$ ms. Finally, the network responded in the Go epoch with an interval of T_{resp}.

RNN training

If the relevant evidence points towards choice 1, the output channel 1 (composed of two output units) was activated, otherwise the output channel 2 (composed of two output units) was activated. The cost function L is the mean squared error (MSE) between the network output ($\mathbf{z}$) and target outputs ($\hat{\mathbf{z}}$):

$$L = \frac{1}{N_{out}} \sum_{i=1}^{N_{out}} \left(\mathbf{z}_i - \hat{\mathbf{z}}_i\right)^2 \quad \text{(Equation 7)}$$

We optimized the weights $\{\mathbf{W}^{in}, \mathbf{W}^{rec}, \mathbf{W}^{out}\}$ using the well-established Adam algorithm (Kingma and Ba, 2015), with default configuration of hyperparameters. The learning rate is 0.0005, and the exponential decay rate for the first and second moment estimates are 0.9 and 0.999, respectively. The off-diagonal connections of recurrent weight matrix $\mathbf{W}^{rec}$ were initialized as independent Gaussian variables with mean 0 and standard deviations $0.3/\sqrt{N}$, and diagonal connections were initialized to 1. The initial input connection weights were uniformly drawn from -0.5 to 0.5. The output connection weights $\mathbf{W}^{out}$ were initialized from an independent Gaussian random distribution with mean 0 and standard deviation $0.4/\sqrt{N}$.

Definition of task-related axes

We first grouped neural population activity into the matrix $\mathbf{X} \in \mathbb{R}^{N \times CT}$, where $N = 256$ denotes the number of RNN units, $C = 1 \times 8$ denotes the number of the conditions (8 different sensory stimulus conditions in the given context), and T denotes the time step. Spike activity was binned by a 20-ms window. Different task epochs correspond to different $\mathbf{X}$-matrices.

C-cue-axis and M-cue-axis. During the cue stimulus epoch, for the given color context, we obtained the matrix $\mathbf{X}_{cue,color} \in \mathbb{R}^{N \times CT}$, where $N = 256$, $T = 400/20$. We further performed PCA on the matrix $\mathbf{X}_{cue,color}$. The first PC explained 91% of data variance (Figure S6C, green bar), so we defined it as the C-cue-axis. Similarly, for the given motion context, we performed PCA on the matrix $\mathbf{X}_{cue,motion} \in R^{N \times CT}$ and the ratio of explained variance of the first five PCs is also shown in Figure S6C (grey bar). The first PC explained 92% of data variance. Therefore, we defined the first PC dimension as the M-cue-axis(Table 1).

C-choice-axis and M-choice-axis. The population activity matrices in the given context were $\mathbf{X}_{integ,color} \in \mathbb{R}^{N \times CT}$ and $\mathbf{X}_{integ,motion} \in \mathbb{R}^{N \times CT}$, where $T = 800/20$. We performed PCA on $\mathbf{X}_{integ,color} \in \mathbb{R}^{N \times CT}$ and $\mathbf{X}_{integ,motion} \in \mathbb{R}^{N \times CT}$, respectively. The ratio of explained variance of the first five PCs in the specific context is shown in Figure S8B. In a similar fashion, we defined C-choice-axis and M-choice-axis for the color and motion contexts, respectively.

QUANTIFICATION AND STATISTICAL ANALYSIS

Computation of sequentiality index (SI)

We computed the SI to quantify the sequential activation of the population response of excitatory neurons during the delay epoch. The SI is defined as the sum of the entropy of the peak response time distribution of the recurrent neurons and the mean log ridge-to-background ratio of the neurons, where the ridge-to-background ratio for a given neuron is defined as the mean activity of the neuron inside a small window around its peak response time divided by its mean activity outside this window (Orhan and Ma, 2019).

To compare the SI statistics of trained RNN with untrained RNN, we performed additional Monte Carlo control analyses. Briefly, given each randomly initialized weights, we simulate the activity of 256 neurons from the RNN using the same number of trials. We then averaged, sorted the population activity, and computed the SI statistic. We repeated the procedure 10,000 times and used that to compute the Monte Carlo p-value.

Finding rotation dynamics via jPCA

The jPCA method has been used to reveal rotational dynamics of neuronal population responses (Churchland et al., 2012). We assumed that the data were modeled as a linear time-invariant continuous dynamical system of the form: $\dot{\mathbf{x}} = \mathbf{M}\mathbf{x}$, where the linear transformation matrix $\mathbf{M}$ was constrained to be skew-symmetric (*i.e.*, $\mathbf{M}^\top = -\mathbf{M}$). The jPCA algorithm projects high-dimensional data $\mathbf{x}(t)$ onto the eigenvectors of the $\mathbf{M}$ matrix, and these eigenvectors arise in complex conjugate pairs. Given a pair of eigenvectors $\{v_k, \overline{v}_k\}$, the k-th jPCA projection plane axes are defined as $u_{k,1} = v_k + \overline{v}_k$ and $u_{k,2} = j(v_k - \overline{v}_k)$ (where $j = \sqrt{-1}$). The solution to the above continuous-time differential equation is given by $\mathbf{x}(t) = e^{\mathbf{M}t}\mathbf{x}(0)$, where the family $\{e^{\mathbf{M}t}\}$ is often referred to as the semi-group generated by the linear operator $\mathbf{M}$. Since $\mathbf{M}$ is skew-symmetric, $e^{\mathbf{M}}$ is orthogonal; therefore, it can describe the rotation of the initial condition $\mathbf{x}(0)$ over time. It is emphasized that the jPCA method is based on a linear approximation of the nonlinear dynamics described by the RNN, so that the potential rotation or oscillatory dynamics can be captured.

Applying eigenvalue decomposition to the real skew-symmetric matrix $\mathbf{M}$, so that $\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{-1}$, where $\mathbf{\Lambda}$ is a diagonal matrix whose entries $\{\lambda_i\}_{i=1}^N$ are a set of (zero or purely imaginary) eigenvalues. Upon time discretization (assuming $dt = 1$), we obtained the discrete analog of dynamic equation $\mathbf{x}(t+1) = (\mathbf{I} + \mathbf{M})\mathbf{x}(t)$. Alternatively, we directly solved a discrete dynamical system of the vector autoregressive (VAR) process form $\mathbf{x}(t+1) = \mathbf{Q}\mathbf{x}(t)$over the space of orthogonal $\mathbf{Q}$ matrices. Mathematically, we have previously shown that this is equivalent to solving the following constrained optimization problem (Nemati et al., 2014):

$$\mathbf{Q}^* = \underset{\mathbf{Q}}{\operatorname{argmin}} \| |\mathbf{A} - \mathbf{Q}| \|_F^2, \text{subject to } \mathbf{Q}\mathbf{Q}^\top = \mathbf{I}, \qquad \text{(Equation 8)}$$

where $\| \bullet \|_F$ denotes the matrix Frobenius norm, and $\mathbf{A} = (\mathbf{X}_{t+1}\mathbf{X}_t^\top)(\mathbf{X}_t\mathbf{X}_t^\top)^{-1}$ represents the least square solution to the unconstrained problem $\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t)$. The solution to the above constrained optimization is given by the orthogonal matrix factor of Polar Decomposition of matrix $\mathbf{A}$, namely $\mathbf{A} = \mathbf{Q}\mathbf{P}$ (Higham, 1986).

Finding fixed points and line attractor

We focused on finding the fixed points and slow points of dynamical system to explore the mechanism by which the RNN performed the context-dependent task. The computational task was to find some points that satisfy $\dot{\mathbf{x}}(t) = 0$ for all t, that is, we needed to solve a first-order differential equation

$$-\mathbf{x} + \mathbf{W}^{rec}\mathbf{r} + \mathbf{W}^{in}\mathbf{u} + \mathbf{b} + \sqrt{2\tau\sigma_{rec}^2}\xi = 0 \qquad \text{(Equation 9)}$$

for a constant input $\mathbf{u}$ and with a transfer function $\mathbf{r} = f(\mathbf{x})$. However, the nonlinear function $f(\mathbf{x})$ makes it difficult to find an analytical solution to the differential equation. Therefore, we identified the fixed points and slow points through numerical optimization. According to the algorithm described in Sussillo and Barak (2013), we solved the optimization problem as follows:

$$\min_{\mathbf{x}} q(\mathbf{x}) \quad \text{(Equation 10)}$$

where $q(\mathbf{x}) = \frac{1}{2}\|\dot{\mathbf{x}}\|^2$. To identify the local minimum, we defined the RNN state $\mathbf{x}$ as a slow point if $q(\mathbf{x}) < 0.01$ and $\mathbf{x}$ is a fixed point if $q(\mathbf{x}) < 0.0001$. These fixed points and slow points were calculated during the sensory stimulus epoch. The initial conditions for optimization were points in the neighborhood of $\mathbf{x}(t)$, which was the start state of RNN system trajectories. For each fixed point, we repeated the optimization procedure 300 times, and the initial conditions at each time were sampled from the neighborhood of $\mathbf{x}(t)$. Based on these 300 candidate fixed points, we chose stable fixed points as attractors, characterized by slow dynamics. We determined a fixed point being stable if neural states empirically converged to attractors. Here, we only considered stable fixed points (represented by cross symbols in Figure 10). For a given context, there were 8 different trial conditions, and the population activity on each trial condition was characterized by a state trajectory (Figure 8C). In total, we identified 16 attractors and each state trajectory eventually converged to its fixed-point attractor (Figure 10A). The initial conditions for slow point optimization were sampled from a normalized random Gaussian matrix. We repeated the optimization procedure 56 times for each trial condition and obtained $56 \times 16 = 896$ slow points.