Supplementary Information for

Understanding the computation of time using neural network models

Zedong Bi, Changsong Zhou

Changsong Zhou
E-mail: cszhou@hkbu.edu.hk

This PDF file includes:
Supplementary text
Figs. S1 to S11
References for SI reference citations
S1. Method

S1.1. Network details. We adopted a discrete-time formulation of network dynamics, in which
\[ x_t = W^{rec} r_{t-1} + W^{in} u_t + W^{in,att} u^{att}_t - \theta^{att} + b + \sqrt{2\sigma^2} N(0, 1), \]  
where \( x_t, r_t \) and \( u_t \) are respectively the synaptic current, firing rate and network input at time step \( t \), \( b \) is the background input, \( W^{rec} \) is the recurrent weight, \( W^{in} \) is the input weight, and \( \sigma^{rec} \) is the strength of recurrent noise. We supposed \( r_t = f(x_t) \), with \( f(\cdot) \) being the softplus current-rate transfer function, i.e.
\[ f(x) = \log(1 + \exp(x)). \]

Input \( u_t \) is also noisy,
\[ u_t = u_{signal} + \sqrt{2\sigma^2} N(0, 1), \]
with \( \sigma_{in} \) being the strength of input noise. \( W^{in,att} \), \( u^{att}_t \) and \( \theta^{att} \) are the quantities related to the input units modulated by top-down attention. They are only valid when studying the effect of anticipatory attention in non-timing tasks (Fig. 6g-i). The model does not have these quantities in the other tasks. \( W^{in,att} \) is the weight from the attention-modulated units to the recurrent network, \( u^{att}_t \) is the input current to the attention-modulated units, and \( \theta^{att} \) is the firing threshold of these units.

The firing threshold is
\[ \theta^{att} = [\theta^{att}_0 - W^{fb,att} r_t] +, \]
with \( W^{fb,att} \) being positive feedback weight, so that \( \theta^{att} \) decreases with feedback current until zero, starting from \( \theta^{att}_0 = 1.5 \). Eq. S4 models the disinhibitory effect of feedback connections (1). Similar to \( u_t, u^{att}_t \) is also noisy, with the noise strength \( \sigma^{att}_{in} \) (eq. S3).

Some previous studies started with a continuous-time formulation, and obtained the discrete-time version using Euler method (omitting the attention-modulated units):
\[ x_t = (1 - \alpha)x_{t-1} + \alpha(W^{rec} r_{t-1} + W^{in} u_t + b + \sqrt{2\alpha^{-1}\sigma^{rec}} N(0, 1)), \]
with \( \alpha = \Delta t/\tau \) being the ratio of time step length \( \Delta t \) and membrane time constant \( \tau \). In our study, we effectively set \( \alpha = 1 \), similarly as the scheme used in Ref. (2, 3). We also set \( \Delta t = 20 \) ms. The output of the network is supposed to be
\[ z = W^{out} r + b^{out}, \]
with the dimension of \( z \) depending on tasks.

We set \( \sigma_{in} = 0.01, \sigma^{rec} = 0.05 \) when training the network. After training, when plotting the neuronal activities in the perception epoch (Fig. 2b-d), we kept \( \sigma_{in} = 0.01, \sigma^{rec} = 0.05 \) so that the neuronal temporal profiles under different durations of perception epoch did not fully overlap. When doing the other analysis, we turned off the noises by default.

S1.2. Task details.

S1.2.1. Timing tasks. Interval production task (IP). The network received from 2 input units: from one came the two pulses that defined the time interval, and from the other came the Go cue. The interval between the beginning of the simulation and the onset of the first pulse was
\[ T_{start} \sim U(60 \text{ ms}, 500 \text{ ms}), \]
where \( U(t_1, t_2) \) is a uniform distribution between \( t_1 \) and \( t_2 \). The interval between the offset of the first pulse and the onset of the second pulse was
\[ T \sim U(400 \text{ ms}, 1400 \text{ ms}). \]

Note that we set the range of \( T \) to be \([400 \text{ ms}, 1400 \text{ ms}] \) during training, but after training, we only investigated the performance of the network when \( T \in [600 \text{ ms}, 1200 \text{ ms}] \). The reason is that there were boundary effects if, after training, \( T \) took a value close to 400 ms or 1400 ms: if \( T \) was close to 400 ms, then the time interval produced by the network was biased to be larger than \( T \); whereas if \( T \) was close to 1400 ms, then the produced interval was biased to be smaller than \( T \). Such biases were weak if \( T \) took a middle value (Fig. S1e).

The interval between the offset of the second pulse and the onset of the Go cue (i.e., the delay period) was
\[ T_{delay} \sim U(600 \text{ ms}, 1600 \text{ ms}). \]

All input pulses (including the two pulses that defined the time interval, and the Go cue) lasted for 60 ms, and had strength 1. Input units stayed at 0 when there were no pulses.

The target output was a scalar. It stayed at zero from the beginning, jumped to 1 at time \( T \) after the offset of the Go cue, and kept at 1 until the end of the simulation at 300ms afterwards.

Interval comparison task (IC). The network received two successive long-lasting stimuli respectively from two input units.

The first stimulus, which came from the first unit, started at time \( T_{start} \) after the beginning of the simulation, and lasted for
At the 1140 ms after the presentation of the two stimuli, a two-dimensional one-hot vector lasting for 60 ms was input from a whether or not the spatial coordinate of the first pulse was larger than that of the second pulse.

Spatial reproduction task (SR). The network received from two channels of stimuli lasting for 600 ms, 1600 ms. The target outputs were two scalars that coded spatial locations. This line contained 32 units, whose preferred directions were uniformly spaced from -6 to 25. For unit $i$ with preferred location $y_i$, its activity in a pulse with location $x$ was

$$A_{in}(t) \exp[-\frac{1}{2}(|y_i - x|^2)],$$

where $A_{in}(t) = 1$ during the presentation of the pulse and $A_{in}(t) = 0$ at the other time. In our simulation, the spatial locations of the stimuli were uniformly drawn from 0 to 19. The second and third channels were both scalar inputs. The pulse from the second channel defined the time interval to be remembered together with the pulse from the first channel. The pulse from the third channel acted as Go cue. $T_{start}$, $T$ and $T_{delay}$ were distributed similarly as in IP (eqs. S7-S9).

The target output was a line with 32 units, which represented response location using similar tuning curves as the ones used for the input line (eq. S11):

$$\hat{z}_i = A_{out}(t) \exp[-\frac{1}{2}(|y_i - x|^2)],$$

where the amplitude $A_{out}(t)$ stayed at zero from the beginning, jumped to 1 at time $T$ after the offset of the Go cue, and stayed at 1 until the end of the simulation at 300 ms afterwards.

Timed decision making task (t-DM). The network received from three channels of scalar inputs. From the first two channels came the stimuli whose strengths were to be compared with each other, and from the last channel came the Go cue pulse. Starting from the beginning of simulation, the first two channels were set to 0 for duration $T_{start}$, and then jumped to $A_1$ and $A_2$ respectively; after $T$ time, these two channels were set to 0 again. The Go cue pulse came at time $T_{delay}$ after the offset of the first two channels. Here,

$$A_1 = \gamma + c, \quad A_2 = \gamma - c,$$

where $\gamma$ was the average strength of these two stimuli and was distributed as $\gamma \sim U(0.8, 1.2)$, and $c$ measured the strength difference of these two stimuli, and was distributed as

$$c \sim U([-0.08, -0.04, -0.02, -0.01, 0.01, 0.02, 0.04, 0.08]),$$

where $U(\{a_1, a_2, \ldots, a_n\})$ denotes a discrete uniform distribution over the set $\{a_1, a_2, \ldots, a_n\}$. $T_{start}$, $T$ and $T_{delay}$ were distributed similarly as in interval production task (eqs. S7-S9).

The target outputs were two scalars $\hat{z}_0$ and $\hat{z}_1$. Both stayed at zero from the beginning. If $c > 0$, then $\hat{z}_0$ jumped to 1 at time $T$ after the offset of the Go cue, and stayed at 1 until the end of the simulation at 300 ms afterwards. Otherwise, $\hat{z}_1$ jumped to 1 at time $T$ after the offset of the Go cue.

S1.2.2. Non-timing tasks: default settings. Spatial reproduction task (SR). The network received pulses from two input channels. The first channel was a line that coded spatial locations. The second channel was a line that coded spatial locations. This line contained 32 units, whose preferred directions were uniformly spaced from -6 to 25. For unit $i$ with preferred location $y_i$, its activity in a pulse with location $x$ was

$$A_{in}(t) \exp[-\frac{1}{2}(|y_i - x|^2)],$$

where $A_{in}(t) = 1$ during the presentation of the pulse and $A_{in}(t) = 0$ at the other time. In our simulation, the spatial locations of the stimuli were uniformly drawn from 0 to 19. The second and third channels were both scalar inputs. The pulse from the second channel defined the time interval to be remembered together with the pulse from the first channel. The pulse from the third channel acted as Go cue. $T_{start}$, $T$ and $T_{delay}$ were distributed similarly as in IP (eqs. S7-S9).

The target output was a line with 32 units, which represented response location using similar tuning curves as the ones used for the input line (eq. S11):

$$\hat{z}_i = A_{out}(t) \exp[-\frac{1}{2}(|y_i - x|^2)],$$

where the amplitude $A_{out}(t)$ stayed at zero from the beginning, jumped to 1 at time $T$ after the offset of the Go cue, and stayed at 1 until the end of the simulation at 300 ms afterwards.

Timed decision making task (t-DM). The network received from three channels of scalar inputs. From the first two channels came the stimuli whose strengths were to be compared with each other, and from the last channel came the Go cue pulse. Starting from the beginning of simulation, the first two channels were set to 0 for duration $T_{start}$, and then jumped to $A_1$ and $A_2$ respectively; after $T$ time, these two channels were set to 0 again. The Go cue pulse came at time $T_{delay}$ after the offset of the first two channels. Here,

$$A_1 = \gamma + c, \quad A_2 = \gamma - c,$$

where $\gamma$ was the average strength of these two stimuli and was distributed as $\gamma \sim U(0.8, 1.2)$, and $c$ measured the strength difference of these two stimuli, and was distributed as

$$c \sim U([-0.08, -0.04, -0.02, -0.01, 0.01, 0.02, 0.04, 0.08]),$$

where $U(\{a_1, a_2, \ldots, a_n\})$ denotes a discrete uniform distribution over the set $\{a_1, a_2, \ldots, a_n\}$. $T_{start}$, $T$ and $T_{delay}$ were distributed similarly as in interval production task (eqs. S7-S9).

The target outputs were two scalars $\hat{z}_0$ and $\hat{z}_1$. Both stayed at zero from the beginning. If $c > 0$, then $\hat{z}_0$ jumped to 1 at time $T$ after the offset of the Go cue, and stayed at 1 until the end of the simulation at 300 ms afterwards. Otherwise, $\hat{z}_1$ jumped to 1 at time $T$ after the offset of the Go cue.
**S1.2.3. Non-timing tasks: studying the factors that influence the strength of temporal signal.** To study the effect of the overlap of sensory input to the strength of temporal signal in the delay epoch of SR, COMP and CD (Fig. 6d), we expanded the unit number in the line channels to 44 (default is 32), and broadened the standard deviation of the tuning curves (eq. S12) to 4 (default is 2). These units coded spatial locations in the range -12 to 31. In our simulation, the spatial locations of input stimuli were uniformly drawn from 0 to 19.

To study the effect of multi-tasking (Fig. 6e), we trained the network on t-SR and SR concurrently, or on t-DM and DM concurrently. The two tasks in each pair share the same input and output channels. We used a one-hot vector from another two-dimensional input channel to indicate which task should be performed (4). The network was to be able to perform either of the indicated task.

To study the effect of timing anticipation (Fig. 6f), we trained the network to perform SR, COMP, CD, DM and cue-DM, with the duration $T$ of the delay epoch (for SR, COMP and CD) or the stimuli-presentation epoch (for DM and cue-DM) was randomly between [800 ms, 1600 ms]. After training, we analyzed the simulation results when $T = 1200$ ms, and compared the results with the cases after training the network with $T$ fixed at 1200 ms. To study the effect of anticipatory attention (Fig. 6g-i), feedback was imposed on the second input channel of SR, COMP and CD, and was imposed on the third channel of cue-DM. This means that these input channels were modeled using the third term at the right-hand side in eq. S1, instead of the second term.

**S1.3. Training details.** Training was performed to minimize a cost function using back-propagation through time. Cost function was defined as

$$C = \sum_i m_i(z_i - \hat{z}_i)^2,$$

where $i$ is the index of output units, $z_i$ is the actual output defined by eq. S6, $\hat{z}_i$ is the target output, and $m_i$ is the mask. In all tasks, $m_i = 0$ before the onset of the first stimulus, and $m_i = 1$ afterwards; therefore, only the output after the onset of the first stimulus was constrained. When studying the effect of anticipatory attention in non-timing tasks (Fig. 6g-i), we added L2 regularization to feedback current $I^{fb}_i = W_i^{fb,att}r_t$ (see eq. S4), so that eq. S15 becomes

$$C = \sum_i m_i(z_i - \hat{z}_i)^2 + \beta_{fb} \sum_i (I^{fb}_i)^2,$$

with $\beta_{fb} = 10^{-4}$. This cost function was minimized using Adam optimizer at learning rate 0.0005, with batch size 64 in each training step. We trained 16 configurations to perform IP and IC tasks, and trained 30 configurations to perform t-SR and t-DM tasks. Different configurations were initialized using different random seeds.

Before training, recurrent self-connections ($W_{i,rec}^{i}$ in eq. S5) were initialized to 1, and other recurrent connections were initialized as independent Gaussian variables with mean 0 and standard deviation $0.3/\sqrt{N_{rec}}$, with $N_{rec} = 256$ being the number of recurrent units. This initialization strategy was used in Ref. (3). The identity self-connections prevent vanishing gradient during training (5), and the non-zero off-diagonal recurrent connections induce sequential activity in the network after training (3), so that the dynamics of the network becomes comparable to experimental observations (6–10). Output connections were initialized as independent Gaussian variable with mean 0 and standard deviation $1/\sqrt{N_{rec}}$. Input connections from the line input were initialized as variables drawn uniformly from $[-1/\sqrt{D_{channel}}, 1/\sqrt{D_{channel}}]$, with $D_{channel}$ being the standard deviation of the Gaussian tuning curve (eq. S11), which was 2 by default and 4 when studying the effect of input overlap in non-timing tasks. The input connections from the other channels were initialized as variables drawn uniformly from $[-1/\sqrt{D_{channel}}/\sqrt{D_{channel}}], with D_{channel} being the dimension of the input channel.

Every 200 training steps, we evaluated the performance of the network using a batch of size 512, and stopped training as soon as the performance of the network reached criterion (Fig. S1k-i). We introduced our criterion in t-SR and t-DM in details, the other tasks shared similar criterion:

In t-SR, a time interval was considered to be produced if: (1) the activities of all the 32 output units were below 0.2 before the offset of the Go cue, (2) one of them went above 0.5 at some point $t_p$ before $T + 300$ms after the offset time $t_{off}$ of the Go cue. The produced interval was $T_p = t_{off} - t_p$. Output location at time $t_p$ was read out using a population vector method (see the computer code in Ref. (4)). Training was stopped as soon as (1) time intervals were produced in over 95% simulation trials, (2) the relative error of the produced intervals $|T_p - T|/T < 0.025$, (3) the output locations were on average within 0.8 of the input locations.

In t-DM, a time interval was considered to be produced if: (1) the activities of both output units $z_0$ and $z_1$ were below 0.2 before the offset of the Go cue, (2) one of them went above 0.5 at some point $t_p$ before $T + 300$ms after the offset $t_{off}$ of the Go cue, whereas the other one stayed below 0.5. The produced interval was $T_p = t_{off} - t_p$. In the trials in which a time interval was produced, the decision was considered to be correct if: when $c > 0$ (or $c < 0$), $z_2$ (or $z_1$) went above 0.5 and $z_1$ (or $z_0$) kept below 0.5. Training was stopped as soon as (1) time intervals were produced in over 96% of simulation trials, (2) the relative error of the produced intervals $|T_p - T|/T < 0.025$, (3) the decision error rate was smaller than 0.02.

**S1.4. Data analysis.**

**S1.4.1. Types of neurons at the end of the delay epoch.** In IP or IC, we supposed $f_i(T)$ to be the activity of the ith neuron at the end of the delay epoch as a function of the duration $T$ of the perception (for IP) or stimulus (for IC) epoch. We picked neurons that can be strongly activated at the end of the delay epoch, namely the neurons whose $\max_{f_i([T_{min}, T_{max}])} f_i(T) > \theta_{sa}$, with $T_{min} = 600$ ms and $T_{max} = 1200$ ms respectively being the minimal and maximal values of $T$ in our simulation, and $\theta_{sa} = 2$. Our results are not sensitive to the value of $\theta_{sa}$. We classified $f_i(T)$ of the picked neurons into three types, namely monotonically increasing (MoI), monotonically decreasing (MoD), and non-monotonic (non-M) in the following way: We divided...
the range of $T$ (i.e., $[T_{min}, T_{max}]$) into four parts of the same length, and calculated the mean value of $f_i(T)$ in these four parts, say $f_i(\text{part } 1) = \frac{4}{T_{max} - T_{min}} \int_{T_{min}}^{T_{min} + (T_{max} - T_{min})/4} f_i(T) dT$, $f_i(\text{part } 2) = \frac{4}{T_{max} - T_{min}} \int_{T_{min} + (T_{max} - T_{min})/4}^{T_{min} + 2(T_{max} - T_{min})/4} f_i(T) dT$, etc. If $f_i(\text{part } 1) \leq f_i(\text{part } 2) \leq f_i(\text{part } 3) \leq f_i(\text{part } 4)$, then neuron $i$ belongs to MoI type; if $f_i(\text{part } 1) \geq f_i(\text{part } 2) \geq f_i(\text{part } 3) \geq f_i(\text{part } 4)$, then neuron $i$ belongs to MoD type; otherwise, neuron $i$ belongs to non-M type.

In t-SR, we supposed $g_i(T, x)$ to be the activity of the $i$th neuron at the end of the delay epoch as a function of $T$ at a given location $x$ of the first pulse. We picked neurons that can be strongly activated at the end of the delay epoch (i.e., the neurons whose $\max_{T,x} g_i(T, x) > \theta_o$). We then defined $f_i(T) = \max_x g_i(T, x)$, and classified neuron $i$ into MoI, MoD or non-M types according to the monotonicity of $f_i(T)$ in the similar way to the IP or IC case introduced above. Similarly, in t-DM, we classified neurons according to $f_i(T) = \max_x g_i(T, c)$, where $c$ is the half difference between the strengths of the presented stimuli (eq. S13).

**S1.4.2. Temporal scaling in the production epoch.** Analysis of temporal scaling was performed using similar technique to Ref. (2). Specifically, we calculated the $k$th scaling component $u_{SC,k}$ through the following equation:

$$u_{SC,k} = \arg \min_u \frac{\sum_{i} \sum_{T} (r_i^k(T; T)u - \text{Mean}_T(r_i^k(T; T)u))^2}{\sum_{i} \sum_{T} (r_i^k(T; T) - \text{Mean}(r_i^k(T; T)))^2},$$

where $r_i^k(T; T)$ is population activity at the scaled time when the duration of the perception epoch is $T$ (see below for details), the denominator is the total variance of the trajectories, and the numerator is the variance that cannot be explained by temporal scaling. To calculate the first scaling component $u_{SC,1}$, we set $r_i^1(T; T) = r^{PC}(T_p; T)$, with $0 \leq T \leq 1$, where $r^{PC}$ is the projection of the population activity in the subspace spanned by the first 9 principal components, and $T_p$ is the interval produced by the network in the production epoch; then we minimized $u$ in eq. S16. To calculate the second scaling component $u_{SC,2}$, we set $r_i^2(T; T) = r_i^1(T; T) - r_i^0(T; T)u_{SC,1}$, and then minimized $u$ in eq. S16 in the subspace orthogonal to $u_{SC,1}$. In this way, we calculated all the 9 scaling components one by one.

Scaling index (SI) of a subspace $U$ was defined as

$$SI = \frac{\sum_{i} \sum_{T} (r_i^k(T; T)U - \text{Mean}_T(r_i^k(T; T)U))^2}{\sum_{i} \sum_{T} (r_i^k(T; T) - \text{Mean}(r_i^k(T; T)))^2},$$

where $r_i^k(T; T)U$ is the projection of the scaled trajectory to the subspace $U$.

**S1.4.3. The geometry of coding combination.** During the perception epoch of t-SR, the network state is quantified by the time elapsed from the beginning of the epoch (temporal flow) and the spatial information of the first pulse. At the end of the delay epoch of t-SR, the network state is quantified by the time interval between the first two pulses and the spatial information of the first pulse. During the production epoch of t-SR, the network state is quantified by temporal flow, time interval and spatial information. Similar scenario also exists in t-DM, except that the non-temporal information is also the decision choice made by the network. In t-DM, the decision choice $d$ depends on the sign of the half difference $c$ between the strength of the presented two stimuli (eq. S13), we defined $r_i(\{d = 1, \{a\} = (r_i(\{c, \{a\}) \geq 0$ and $r_i(\{d = -1, \{a\} = (r_i(\{c, \{a\}) < 0$, where $\{a\}$ indicates the other parameters than decision choice, and used $r_i(\{d, \{a\} )$ to do the following analysis. Together, during the perception epoch and at the end of the delay epoch of t-SR and t-DM, two variables are coded in the network state; during the production epoch, three variables are coded in the network state. We used two measurements to quantify the geometry of the coding combination of multiple variables: (1) the angle between the first marginal principal components and (2) the distance in the coding space (11), introduced below.

Suppose the activity of the $i$th neuron $r_i(a, b)$ is a function of two variables $a$ and $b$, with the mean of $r_i(a, b)$ being subtracted so that $r_i(a, b)|_{a,b} = 0$. The marginal principal components (PCs) with respect to $a$ are the PCs of the dot set

$$\{\{r_i(a, b)\}_a\},$$

and the marginal PCs of $b$ are the PCs of $\{\{r_i(a, b)\}_b\}$. We quantified the coding orthogonality of $a$ and $b$ by calculating the angle between the first marginal PCs of $a$ and $b$. The portions of variance explained by $a$ and $b$ are respectively $p_a = \text{Var}_{i,a} (\{\{r_i(a, b)\}_a\})/\text{Var}_{i,a,b}$ and $p_b = \text{Var}_{i,b} (\{\{r_i(a, b)\}_b\})/\text{Var}_{i,a,b}$, with the total variance $\text{Var}_{i,a,b} = \text{Var}_{i,a,\{b\}}(\{r_i(a, b)\}_a)$. The portion of mixed variance between $a$ and $b$ is $p_{a+b} = 1 - p_a - p_b$.

In the case that the activity of the $i$th neuron $r_i(a, b, c)$ is a function of three variables, we also subtracted the mean of $r_i(a, b, c)$ so that $r_i(a, b, c)|_{a,b,c} = 0$. The marginal PCs of $a, b$ and $c$ are respectively the PCs of $\{\{r_i(a, b, c)\}_a\}, \{\{r_i(a, b, c)\}_b\}$, and $\{\{r_i(a, b, c)\}_c\}$. The portions of variance explained by these variables and their mixing were defined as (11):

$$p_a = \text{Var}_{i,a} (\{\{r_i(a, b, c)\}_a\})/\text{Var}_{i,a,b,c},$$

$$p_b = \text{Var}_{i,b} (\{\{r_i(a, b, c)\}_b\})/\text{Var}_{i,a,b,c},$$

$$p_c = \text{Var}_{i,c} (\{\{r_i(a, b, c)\}_c\})/\text{Var}_{i,a,b,c},$$

$$p_{a+b} = \text{Var}_{i,a,b} (\{\{r_i(a, b, c)\}_a - (r_i(a, b))_{a,c} - (r_i(a, b))_{b,c}\})/\text{Var}_{i,a,b,c},$$

$$p_{b+c} = \text{Var}_{i,b,c} (\{\{r_i(a, b, c)\}_b - (r_i(a, b))_{a,c} - (r_i(a, b))_{b,c}\})/\text{Var}_{i,a,b,c},$$

$$p_{a+c} = \text{Var}_{i,a,c} (\{\{r_i(a, b, c)\}_c - (r_i(a, b))_{a,c} - (r_i(a, b))_{b,c}\})/\text{Var}_{i,a,b,c},$$

where $\text{Var}_{i,a,b,c} = \text{Var}_{i,a,b,c} (\{r_i(a, b, c)\})$ is the total variance, "+" sign in the subscript indicates the mixing of several variables.

In Fig. 3, we used the network state trajectory after 400 ms (200 ms) of transient period of the perception (production) epoch to do the analysis.

Zedong Bi, Changsong Zhou
S1.4.4. Decoding. We studied two types of nearest-centroid decoders (12). Given a population state $f_0$, the decoded value $a_{d,1}$ read-out by Decoder 1 is

$$a_{d,1} = \arg \min_{a \in A} \| f_0 W^{\text{dec}} - f(a; b_{\text{train}})W^{\text{dec}} \|,$$  \[\text{S18}\]

where $A$ is the range of $a$, $f(a; b_{\text{train}})$ is the population state as a function of variable $a$ along an iso-$b$ line whose $b$ value is constantly $b_{\text{train}}$, and decoding weight $W^{\text{dec}}$ is the first PC of $f(a; b_{\text{train}})$. The decoded value $a_{d,2}$ read-out by Decoder 2 is

$$a_{d,2} = \arg \min_{a \in A} \| (f_0 - \langle f(a; b_{\text{test}}) \rangle a) W^{\text{dec}} - (f(a; b_{\text{train}}) - \langle f(a; b_{\text{train}}) \rangle a) W^{\text{dec}} \|,$$  \[\text{S19}\]

where $f(a; b_{\text{test}})$ is the iso-$b$ line that $f_0$ belongs to, and $\langle \cdot \rangle_a$ means averaging over $a$. From eq S19, both the mass centers of the two iso-$b$ lines $f(a; b_{\text{train}})$ and $f(a; b_{\text{test}})$ are translationally moved to the zero point before $f(a; b_{\text{train}})$ and $f(a; b_{\text{test}})$ are projected to the decoding space by $W^{\text{dec}}$.

S1.4.5. Correlation between decoding error, angle and mixed variance. In Fig. 4d, f, we computed the correlation between decoding error (DE), the angle (AG) between the first PCs of the decoded and generalized variables, and the mixed variance (MV) between the decoded and generalized variables. A subtle point here is that AG and MV may also be correlated (see Fig. 4c, e for the negative correlation between AG and MV in the production epoch of t-SR), therefore the Pearson’s correlation between DE and AG may be contributed by two pathways: (1) AG influences DE directly; (2) AG influences DE indirectly through MV, due to the correlation between AG and MV. Similar situation also exists for the correlation between DE and MV. To investigate the direct correlation and remove the indirect one, we iteratively took the following operation to reduce the correlation between AG and MV: removing a single data point (i.e., the AG and MV of a single training configuration) from the dataset, so that the absolute value of the correlation between AG and MV in the left dataset is minimal. We found that small correlation (with absolute value below 0.05) between AG and MV could usually be obtained after removing 2 or 3 data points from the whole dataset of 30 points (Figs. S7, S8). In this way, we got a dataset with small correlation between AG and MV, while at the same time, as large as possible. Pearson’s correlation were then calculated using the left dataset to draw Figs. 4d, f, S7, S8.

S1.4.6. Firing sequence and network structure. To plot Fig. 5a, b, we ordered the peak firing time of strongly active neurons (whose peak firing rates were larger than 2) in the studied epoch, and plotted weight connection as a function of the peak order difference between the post- and pre-synaptic neurons.

To plot Fig. 5c, d, we used a more elaborate method to illustrate the network structure underlying t-SR and t-DM. At time $t_0$ and non-time information $x_0$ (which may be spatial location or decision choice), we picked a set $N(t_0, x_0)$ of strongly active neurons whose firing rates at $t_0$ and $x_0$ were larger than a threshold 2 (our result is insensitive to this threshold). We then defined $T_{\text{peak},i}(t_0, x_0)$ to be the peak time of neuron $i$ near $t_0$ at $x_0$: if the activity $f_i(t_0, x_0)$ of neuron $i$ decreased (or increased) with time at time point $t_0$ and non-time information $x_0$, then $T_{\text{peak},i}(t_0, x_0)$ was the time point of the local maximum of $f_i(t_0, x_0)$ before (or after), but most nearest to, $t_0$. Iterating over all the possible values of $x_0$, we got all the strongly active neurons at time $t_0$: $N(t_0) = \bigcup_{x_0} N(t_0, x_0)$. For neuron $i$ in $N(t_0)$, we called its preferred non-time information $x_{\text{pref}}$ to be the value of $x_0$ that maximized its peak firing rate: $x_{\text{pref}} = \arg \max_{x_0} f_i(T_{\text{peak},i}(t_0, x_0), x_0)$. In this way, we classified all the neurons in $N(t_0)$ according to their non-time information preference: $N(t_0) = \bigcup_{x_0} N_{\text{pref},i}(t_0, x_0)$. To detect the behavior of the backpropagation weight from $i$ to each neuron of $N_{\text{ pref},i}(t_0, x_0)$ (except $i$ itself if $i \in N_{\text{pref},i}(t_0, x_0)$). In this way, we studied the recurrent weight $w_{\text{post} - \text{pre}, |x_{\text{post} - \text{pre}}|}$ as a function of the difference $x_{\text{post} - \text{pre}}$ between the orders of the big peak time of the post- and pre-synaptic neurons and the difference $|x_{\text{post} - \text{pre}}|$ of their preferred non-time information. Fig. 5c, d were plotted by averaging $w_{\text{ post} - \text{ pre}, |x_{\text{ post} - \text{ pre}}|}$ over $t_0$ and training configurations.

S2. The relationship between the low dimensionality of the attractor in the delay epoch and the dominance of monotonotic neurons

We denote $M$ as the manifold of the population states at the end of the delay epoch at different durations $T$ of the perception epoch (Fig. 2e). The first principal component (PC) of $M$ explained about 90% of its variance (Fig. 2g, h), and the activities of most neurons changed monotonically with $T$ in $M$ (Fig. 2j). To understand the relationship between these two facts, let’s consider the extreme case that all neurons are linearly monotonic with $T$ in $M$, then $M$ is a line in the population-state space that can be parameterized as $[f_1(T), f_2(T), \ldots, f_N(T)]^T$, with $f_i(T)$ being the activity of the $i$th neuron at the end of the delay epoch when the duration of the perception epoch is $T$. In this case, PC1 of $M$, which explains 100% of the variance of $M$ because $M$ is a line, is the following vector with unit length:

$$\pm \frac{1}{\sqrt{\sum (f_i(T_{\text{max}}) - f_i(T_{\text{min}}))^2}} [f_1(T_{\text{max}}) - f_1(T_{\text{min}}), f_2(T_{\text{max}}) - f_2(T_{\text{min}}), \ldots, f_N(T_{\text{max}}) - f_N(T_{\text{min}})]^T,$$

where $T_{\text{min}} = 600\text{ms}$ and $T_{\text{max}} = 1200\text{ms}$ are respectively the minimal and maximal values of $T$ in our simulation, and the $\pm$ sign indicates that the direction of PC1 is undetermined. If neuron $i$ monotonically increases (or decreases) with $T$, then $f_i(T_{\text{max}}) - f_i(T_{\text{min}}) > 0$ (or $f_i(T_{\text{max}}) - f_i(T_{\text{min}}) < 0$). Apparently, if two neurons $i$ and $j$ have the same (or different) monotonicity, then their corresponding elements in PC1 have the same (different) signs. This is indeed what we found in our simulation (Fig. S2g, h).
S3. The geometric meaning of mixed variance

We denote the population state to be $r = \{r_1, r_2, \cdots, r_N\}$, where $r_i$ is the firing rate of the $i$th neuron, or in general, the activity projected on the $i$th basis vector, say, principal component. Suppose $r$ is parameterized by two variables $a$ and $b$, and we subtract the mean value of $r_i$ so that

$$E_{a,b}[r_i(a, b)] = 0,$$

where $E_{a,b}[\cdot]$ means the average over $a$ and $b$.

The total variance of $r$ is

$$v_{tot} = Var_{i,a,b}[r_i(a, b)]$$

$$= E_i[Var_{a,b}[r_i(a, b)]] + Var_{a,b}[E_i[r_i(a, b)]]$$

$$= E_i[Var_{a,b}[r_i(a, b)]]$$  \[S21\]

where $Var_{x}[\cdot]$ means the variance over variable $x$. The first equation is the definition of the total variance, the second equation is from the law of total variance, and the third equation is from eq. S20. Similarly, the variance explained by $a$ is

$$v_a = Var_{i,a}[E_b[r_i(a, b)]] = E_i[Var_{a}[E_b[r_i(a, b)]]],$$

[S22]

and the variance explained by $b$ is

$$v_b = Var_{i,b}[E_a[r_i(a, b)]] = E_i[Var_{b}[E_a[r_i(a, b)]]]$$

[S23]

Now let’s study a sufficient condition so that

$$v_{tot} = v_a + v_b,$$

[S24]

which means that the mixed variance

$$v_{mix} = v_{tot} - (v_a + v_b)$$

is zero.

From eqs. S21-S23, a sufficient condition to fulfill eq. S24 is

$$Var_{a,b}[r_i(a, b)] = Var_{a}[E_b[r_i(a, b)]] + Var_{b}[E_a[r_i(a, b)]] \quad \text{for every } i.$$

[S26]

According to the law of total variance,

$$Var_{a,b}[r_i(a, b)] = Var_{a}[E_b[r_i(a, b)]] + E_a[Var_{b}[r_i(a, b)].$$

[S27]

Therefore, to realize eq. S26, we can set

$$Var_{b}[E_a[r_i(a, b)]] = E_a[Var_{b}[r_i(a, b)]] \quad \text{for every } i.$$

[S28]

In other words

$$E_a[(E_a[r_i(a, b)] - E_a[Var_{b}[r_i(a, b)])^2] = E_a[Var_{b}[r_i(a, b)] - E_a[r_i(a, b)]^2]}$$

[S29]

Because $E_{a,b}[r_i(a, b)] = 0$, this equation gives

$$E_b[(E_a[r_i(a, b)])^2] = E_a[Var_{b}[r_i(a, b)]$$

[S30]

A sufficient condition to fulfill the equation above is

$$r_i(a, b) - E_b[r_i(a, b)] = f(b) \quad \text{for every } i,$$

[S31]

namely the value of $r_i(a, b) - E_b[r_i(a, b)]$ does not depend on $a$. This sufficient condition can be easily proved by substituting eq. S31 into eq. S30 and using the fact that $E_{a,b}[r_i(a, b)] = 0$. Now let’s try to understand the meaning of eq. S31. Consider four pairs of variables $(a_1, b_1), (a_2, b_1), (a_1, b_2)$ and $(a_2, b_2)$, we have

$$r_i(a_1, b_1) - E_b[r_i(a_1, b_1)] = f(b_1) = r_i(a_2, b_1) - E_b[r_i(a_2, b_1)] \quad \text{for every } i$$

[S32]

$$r_i(a_1, b_2) - E_b[r_i(a_1, b_2)] = f(b_2) = r_i(a_2, b_2) - E_b[r_i(a_2, b_2)] \quad \text{for every } i$$

[S33]

By subtracting eq. S32 from eq. S33, we have

$$r_i(a_1, b_1) - r_i(a_2, b_1) = r_i(a_2, b_1) - r_i(a_2, b_2) \quad \text{for every } i.$$  \[S34\]

This means that between the two iso-$b$ lines in which the values of $b$ are separately fixed at $b_1$ and $b_2$, the vector that connects the two points representing $a_1$ is equal to the vector that connects the two points representing $a_2$. In other words, these two iso-$b$ lines can be related by translational movement. By rewritten eq. S34 as $r_i(a_1, b_1) - r_i(a_2, b_1) = r_i(a_1, b_2) - r_i(a_2, b_2)$, we see that different iso-$a$ lines are also related by translational movement.

From the discussion above, translational relation between different iso-$a$ or iso-$b$ lines is a sufficient condition for zero mixed variance. How about the necessity? In other words, if we observe close-to-zero mixed variance in simulation, how will be the geometry of the iso-$a$ and iso-$b$ lines? We checked this point through simulation. In Fig. S6, we show the iso-space lines of several simulation examples, in the perception, delay and production epochs of t-SR task. We see that in examples with
small mixed variance, the iso-space lines of different spatial information tend to be parallel and of the same length; whereas in examples with large mixed variance, the iso-space lines may be non-parallel or of very different lengths. Additionally, if iso-a or iso-b lines are translationally related, then Decoder 2 (eq. S19) will have perfectly zero generalization error. We found that the generalization error of Decoder 2 is strongly positively correlated with mixed variance (Figs. 4f, S7, S8). These results imply that at least in the context of our simulation, mixed variance is a good index to quantify the translational relationship between different iso-a or iso-b lines, or in other words, the parallelogram-likeness of iso-a and iso-b grids (Fig. 3f, upper left).

The opposite extreme case that \( v_{mix} = v_{tot} \), which, from eq.S25, means \( v_a = v_b = 0 \). From eqs. S22, S23, this means that

\[
\text{Var}_a[E_b[r_i(a,b)]] = \text{Var}_a[E_a[r_i(a,b)]] = 0 \quad \text{for every } i.
\]

In other words, the mean value of \( r_i(a,b) \) over \( b \) (i.e., \( E_b[r_i(a,b)] \)) does not depend on \( a \), and the mean value of \( r_i(a,b) \) over \( a \) (i.e., \( E_a[r_i(a,b)] \)) does not depend on \( b \) either. This implies that different iso-a (and also iso-b) lines are strongly intertwined with each other, so that they have the same mean state value. A good example of this case is that every point in the 2-dimensional range of variables \([a_{min}, a_{max}] \otimes [b_{min}, b_{max}]\) (where \(a_{min}, a_{max}, b_{min}\) and \(b_{max}\) are the minimal and maximal values of \(a\) and \(b\) respectively) is mapped toward a random point in a state space \([r_{1, min}, r_{1, max}] \otimes [r_{2, min}, r_{2, max}] \otimes \cdots \otimes [r_{n, min}, r_{n, max}]\); in this case, every iso-a or iso-b dot set of states has the mean value located at the center of the state space \(\left(\frac{r_{1, min} + r_{1, max}}{2}, \frac{r_{2, min} + r_{2, max}}{2}, \cdots, \frac{r_{n, min} + r_{n, max}}{2}\right)\).
Fig. S1. Performance of the network after training. (a-e) Interval production (IP) task. (a) An example of the input and output of the network in IP. Red and blue lines: two input channels. Dashed black line: target output. Solid black line: actual output. (b) Probability distribution function (p.d.f) of self-connections (blue) and non-diagonal connections (red) of the recurrent network after training. (c) Three examples of the output in the production epoch of IP, when $T = 600$ ms (blue), 900 ms (red) and 1200 ms (yellow). Dashed line: target output. Solid line: actual output. The horizontal dashed black line indicates the threshold that the network is regarded to generate a movement in the production epoch when the output rises across this threshold. (d) Distribution of the scaling index of the output across training configurations in the production epoch of IP. (e) The difference between the produced time interval $T_p$ and the interval $T$ between the first two pulses in IP as a function of $T$. Error bar means standard deviation over 16 training configurations. During training, we set $T \in [400 \text{ ms}, 1400 \text{ ms}]$. This panel shows that if after training we set $T$ to be close to 400 ms, $T_p$ tends to be larger than $T$; whereas if we set $T$ to be close to 1400 ms, $T_p$ tends to be smaller than $T$. Therefore, by default, we set $T \in [600 \text{ ms}, 1200 \text{ ms}]$ for data analysis after training to reduce the bias of $T_p$. (f) Two examples of interval discrimination (IC) task. Upper: the case when the duration of the first stimulus is shorter than that of the second stimulus. Lower: the case when the duration of the first stimulus is longer than that of the second stimulus. Red and yellow lines: two input channels. Dashed black and pink lines: two channels of target output. Solid black and pink lines: two channels of actual output. (g) An example of timed spatial reproduction (t-SR) task. Left upper: the pulse with location information from the first input channel. Left lower: the pulses from the second (yellow) and third (blue) input channels. Right: actual output. (h) Two examples of timed decision making (t-DM) task. Upper: when the input from the first channel (red) is weaker than the input from the second channel (yellow), i.e., $c < 0$. Lower: when $c > 0$. (i-l) Performance of the network during training. (i) Performance of the network during the training of IP, quantified by the probability to successfully produce time interval (upper) and the relative error of the produced interval (lower). Gray lines indicate individual training configurations. Training stopped as soon as both quantities reach the criterion (horizontal dashed lines). (j) Performance of the network during the training of IC, quantified by the probability to successfully output a choice (upper) and the probability of choice error (lower). (k) Performance of the network during the training of t-SR, quantified by the probability to successfully produce time interval (upper), the relative error of the produced interval (middle) and the spatial error of the output. (l) Performance of the network during the training of t-DM, quantified by the probability to successfully produce time interval (upper), the relative error of the produced interval (middle) and the probability of choice error (lower).
Fig. S2. Interval production task. (a) Trajectory speed with time in the perception epoch, shaded belt indicating s.e.m. (standard error of mean). (b) Probability distribution function (p.d.f) of scaling indexes of the activities of single neurons in the production epoch, after counting neurons with the top 10% highest activity (upper panel), top 50% (middle panel) and all neurons (lower panel). (c) The scaling index and explained variance of principal components (PCs) in the production epoch. (d) We calculated the scaling components in the subspace spanned by the first nine principal components. Shown are the first (upper) and last (lower) scaling component of the production epoch of an example training configuration. Color of lines indicate to-be-produced interval $T$. (e) The mean activity of the last scaling component as a function of $T$, with the activities when $T = 600$ ms and $T = 1200$ ms are respectively normalized to be 0 and 1. (f) Scaling index (blue) and ratio of explained variance (orange) in the subspace spanned by the accumulated scaling components. This panel is in the same style as Fig. 2n, except that it analyzes the perception epoch of IP task. (g,h) These two panels explain the relationship between the low dimensionality of manifold $\mathcal{M}$ at the end of the delay epoch and the dominance of neurons monotonically tuned by $T$ (Section S2). (g) Histogram of the elements of PC1 of the manifold $\mathcal{M}$ at the end of the delay epoch at different $T$'s of an example training configuration. Note that the elements corresponding with monotonically decreasing (MoD) and monotonically increasing (MoI) neurons have different signs. (h) In 16 training configurations, for a given element in PC1 of $\mathcal{M}$, it has over 98% probability to have the same sign with most other elements corresponding with neurons of the same type, while have the opposite sign with most other elements corresponding with neurons of the opposite type. In panels c,e, error bars indicate s.e.m. over training configurations.
Fig. S3. Interval comparison tasks. (a-c) Stimulus1 epoch. (a) Population activity in the stimulus1 epoch in the subspace of the first three PCs. Colors indicate the duration $T$ of the epoch. Stars and circles respectively indicate the starting and ending points of the stimulus1 epoch. (b) Coefficient of determination ($R^2$) that quantifies the overlap of the firing profiles of individual neurons at different $T$s, in the same style as Fig. 2d in the main text. (c) Trajectory speed as a function of time in the stimulus1 epoch, shaded belt indicating s.e.m. (d-h) Delay epoch. (d) Trajectory speed in the delay epoch when $T = 600$ ms (blue) and 1200 ms (red), in the same style as Fig. 2f. (e) Ratio of explained variance of the first five PCs of manifold $M$ at the end of the delay epoch, in the same style as Fig. 2g. (f) The position of the state at the end of the delay epoch projected in the first PC of manifold $M$ as a function of $T$, in the same style as Fig. 2h. (g) The distance between two adjacent curves in the delay epoch as a function of time, in the same style as Fig. 2i. (h) The portions of monotonically decreasing (MoD), monotonically increasing (MoI), and non-monotonic (non-M) types of neurons at the end of the delay epoch, in the same style as Fig. 2k. (j-o) Stimulus2 epoch. (i) Population activity in the stimulus2 epoch in the subspace of the first three PCs. The meanings of color scheme, stars and circles are the same as panel a. Triangles indicate critical points. The duration of stimulus 2 is kept at 1200 ms. (j) Scaling index (blue) and ratio of explained variance (orange) in the subspace spanned by the accumulated scaling components, in the same style as Fig. 2n. In this panel and panels k-n, only the trajectories from the beginning of stimulus 2 to the critical points are studied. (k) Trajectory speed in the subspace of the first three scaling components, in the same style as Fig. 2o. (l) Probability distribution of the scaling indexes of single neurons, in the same style as Fig. S2b. (m) The scaling index and explained variance of principal components, in the same style as Fig. S2c. (n) Mean activity of the last scaling component as a function of $T$, in the same style as Fig. S2e. (o) Left panel: speed of the trajectory before (blue) and after (red) the critical point in the subspace of the first three scaling components (SCs). SCs are calculated using the trajectories before the critical points, the red line is plotted by projecting the trajectories after the critical points into the subspace of SCs calculated using those before critical points. Right panel: speed of the trajectory before (blue) and after (red) the critical point in the full population state space.
Fig. S4. Timed spatial reproduction task. (a,b) Perception epoch. (a) Coefficient of determination ($R^2$) that quantifies the overlap of the firing profiles of individual neurons at different $T$s in the perception epoch, in the same style as Fig. 2d. (b) Trajectory speed as a function of time in the perception epoch, shaded belt indicating s.e.m. (c-f) Delay epoch. (c) Trajectory speed as a function of time in the delay epoch when $T = 600$ ms (blue) and 1200 ms (red), in the same style as Fig. 2f. (d) The manifold $\mathcal{M}$ at the end of the delay epoch are parameterized by both time interval $T$ between the first two pulses and the spatial location $x$ of the first pulse. We denote $\mathcal{M}(T; x_0)$ (or $\mathcal{M}(x; T_0)$) to be the set of dots in $\mathcal{M}$ at specific location $x_0$ (or time interval $T_0$). Left panel: the position of the state at the end of the delay epoch projected to the first PC of $\mathcal{M}(T; x_0)$ as a function of $T$, with the position when $T = 600$ ms (or 1200 ms) normalized to be 0 (or 1), in the same style as Fig. 2h. Gray curves: results from 16 training configurations, each at a randomly chosen $x_0$. Blue curve: mean value averaging over $x_0$ and training configurations. Right panel: the position of the state in the first PC of $\mathcal{M}(x; T_0)$. We see that in most training configurations, the position in $\mathcal{M}(x; T_0)$ encodes $x$ continuously and linearly, but big jump happens in some configurations. (g-k) Production epoch. (g) Scaling index (blue) and ratio of explained variance (orange) in the subspace spanned by the accumulated scaling components in the production epoch, averaging over spatial locations and training configurations, in the same style as Fig. 2n. (h) Trajectory speed in the subspace of the first three scaling components in production epoch, in the same style as Fig. 2o. (i) Probability distribution of the scaling indexes of single neurons, in the same style as Fig. S2b. (j) The scaling index and explained variance of principal components, similar to Fig. S2c. (k) Mean activity of the last scaling component, similar to Fig. S2e. Error bars representing s.e.m. are much smaller than the plot markers.
Fig. S5. Timed decision making task. (a-c) Perception epoch. (a) Left: Firing profiles of two example neurons in the perception epoch. Colors indicate \( c \) value, which is the half difference between the strength of the presented stimuli. Right: Trajectories in the subspace of the first two PCs. Stars and circles respectively indicate the starting and ending points of the perception epoch. (b) Coefficient of determination \( (R^2) \) that quantifies the overlap of the firing profiles of individual neurons at different \( T \)s in the perception epoch, in the same style as Fig. 2d. (c) Trajectory speed as a function of time in the perception epoch, shaded belt indicating s.e.m. (d-h) Delay epoch. (d) Trajectories in the subspace of the first three PCs. Stars and circles respectively indicate the starting and ending points of the delay epoch. Blackness of circles indicates \( T \) value as annotated. Curve color indicates \( c \) value as indicated in the color map of panel a, only \( c = -0.04, -0.01, 0.01, 0.04 \) cases are plotted. (e) Trajectory speed as a function of time in the delay epoch when \( T = 600 \) ms (blue) and 1200 ms (red), in the same style as Fig. 2f. (f) The position of the state in the first PC of \( M(T; d_0) \) as a function of \( T \), with the position when \( T = 600 \) ms (or 1200 ms) normalized to be 0 (or 1), in the same style as Fig. 2h. Here, \( M(T; d_0) \) represents the set of dots in manifold \( M \) at the end of the delay epoch at specific decision choice \( d_0 \). (g) The distance between two adjacent curves in the delay epoch as a function of time, in a similar style to Fig. 2i. Left panel: the two adjacent curves have the same \( c \) value, but slightly different \( T \) values. Right panel: the two adjacent curves have the same \( T \) value, but different \( c \) values. In the right panel, blue (orange) curve represents the case when their \( c \) values have the same (different) sign, so that they have the same (different) decision choice. We see that two trajectories representing the same (different) choice tend to get close to (far away from) each other, consistent with the scenario in panel d. (h) The portions of monotonically decreasing (MoD), monotonically increasing (MoI) and non-monotonic (non-M) types of neurons tuned by \( T \) at the end of the delay epoch, in the same style as Fig. 2k. (i-m) Production epoch. (i) Scaling index (blue) and ratio of explained variance (orange) in the subspace spanned by the accumulated scaling components, averaging over \( c \) values and training configurations, in the same style as Fig. 2n. (j) Trajectory speed in the subspace of the first three scaling components, in the same style as Fig. 2o. (k) Probability distribution of the scaling indexes of single neurons, in the same style as Fig. S2b. (l) The scaling index and explained variance of principal components, in the same style as Fig. S2c. (m) Mean activity of the last scaling component, in the same style as Fig. S2e. (n-s) The angle between first parameter-marginalized principal components and mixed variances in the perception (panels n,o), delay (panels p,q) and production epochs (panels r,s). These panels are in the same style as Fig. 3d, e, g-j, except that the non-spatial information is decision choice.
Fig. S6. Examples that illustrate the geometry of the coding combination of temporal and spatial information in t-SR. (a-e) Perception epoch. (a) Each dot represents the angle between F-PC1 and S-PC1 as well as their mixed variances in the perception epoch (after 400 ms of transient period) of t-SR in a training configuration. (b-e) Iso-space lines in the subspace spanned by F-PC1 and S-PC1, in the training configurations indicated in panel a. Stars indicate the points after 400 ms of transient period from the beginning of the perception epoch, and circles indicate the ending points of the perception epoch. Redness from light to strong indicates the spatial locations $x = 0, 2, 4, \ldots, 18$. (f-j) The same as panels a-e, except for showing the iso-space lines in the manifold $M$ at the end of the delay epoch, in the subspace spanned by the first time-interval PC (I-PC1) and S-PC1. Stars and circles indicate $T = 600$ ms and 1200 ms cases respectively. (k-o) The same as panels a-e, except that the iso-space lines in the production epoch are shown. Stars indicate the points after 200 ms of transient period from the beginning of the production epoch, and circles indicate the ending points of the production epoch.
Correlation decoding temporal flow across space in perception epoch

Correlation decoding space across temporal flow in perception epoch

Correlation decoding time interval across space in delay epoch

Correlation decoding space across time interval in delay epoch

Correlation decoding temporal flow across space in production epoch

Correlation decoding space across temporal flow in production epoch

Correlation decoding time interval across space in production epoch

Correlation decoding space across time interval in production epoch

Fig. S7. Decoding generalizability in t-SR. (a-b) Perception epoch. (a) Upper: decoding error as a function of $|t_{train} - t_{test}|$, after Decoder 1 (solid line) or Decoder 2 (dashed line) is trained to read the time elapsed from the beginning of the perception epoch (i.e., temporal flow) using the state trajectory at spatial location $x_{t_{train}}$, and then tested at spatial location $x_{t_{test}}$, in the same style as Fig. 4g. Horizontal dashed line indicates chance level, supposing the decoder works by random guess. Lower: The correlations between the angle (AG) between the first temporal-flow PC and the first spatial PC, the mixed variance (MV) between temporal flow and spatial information, the error of Decoder 1 (DE1) and the error of Decoder 2 (DE2), in the same style as Fig. 4d, f. Note that the correlation between AG and MV is approximately zero, see Section S1.4.5 for this point. (b) Upper: Decoding error as a function of $|t_{train} - t_{test}|$, after Decoder 1 (solid line) or Decoder 2 (dashed line) is trained to read the spatial location at time $t_{train}$ after the beginning of the perception epoch, and then tested at time $t_{test}$. Lower: Correlations between AG, MV, DE1 and DE2. (c-d) Delay epoch. (c) Similar to panel a, except for decoding time interval across spatial information using the state in manifold $\mathcal{M}$ at the end of the delay epoch. (d) Decoding spatial information across time interval using the states in manifold $\mathcal{M}$ at the end of the delay epoch. (e-h) Production epoch. (e) Decoding temporal flow across spatial information in the production epoch. The decoder was trained using $r(t; x_{t_{train}}, T_0)$ and tested using $r(t; x_{t_{test}}, T_0)$, where $r(t; x_{t_{train}}, T_0)$ represents the population activity as a function of $t$ at specific spatial information $x_0$ and time interval $T_0$. $T_0 = 1200$ ms in this panel and panels f. (f) Decoding space across temporal flow in the production epoch. The decoder was trained using $r(x; t_{t_{train}}, T_0)$ and tested using $r(x; t_{t_{test}}, T_0)$, where $r(x; t_{t_{train}}, T_0)$ represents the population activity as a function of spatial information $x$ at specific time point $t_0$ and time interval $T_0$. (g) Decoding temporal flow across time interval in the production epoch. The decoder was trained using $r(t; x_{t_{train}}, x_0)$ and tested using $r(t; x_{t_{test}}, x_0)$. The results are averaged over $x_0 \in [0, 20]$. Upper left: The decoded value $t_{dec}$, as a function of the time $t$ elapsed from the beginning of the production epoch, after Decoder 1 (solid line) or Decoder 2 (dashed line) was trained to read $t$ at $T = 1200$ ms, and then tested at $T = 600$ ms (blue), 900 ms (red) and 1200ms (yellow). The dashed line indicates perfect temporal scaling. Upper right: Decoding error as a function of $T$, after a decoder is trained to read scaled temporal flow $t/T$ at $T = 1200$ ms (indicated by the vertical dashed line), and then tested at $T = T_1$. Lower: correlations. (h) Decoding space across time interval in the production epoch. The decoder was trained using $\langle r(x; t_{t_{train}}, t_0) \rangle_{t_0}$ and tested using $\langle r(x; t_{t_{test}}, t_0) \rangle_{t_0}$, where $\langle \cdot \rangle_{t_0}$ means averaging over temporal flow $t_0$. Zedong Bi, Changsong Zhou
Fig. S8. Decoding generalizability in t-DM. All panels are in the same style as Fig. S7, except that the non-temporal information in t-DM is the decision choice. Note that in some panels (lower panels of b, d, h), the correlation between DE2 and AG as well as the correlation between DE2 and MV are absent. The reason is that in these cases, the decoding error is perfectly zero in all training configurations, so the correlation is undefined.
Fig. S9. Sequential activity and network structure. (a) The neuronal activity (with maximum normalized to 1) in the production epoch of IP task in an example training configuration, sorted according to peak time. (b, c) The same as panel a, but for the stimulus1 (panel b) or stimulus2 (panel c) epoch of IC. (d) Mean (solid line) and s.d. (shaded belt) of the recurrent weights as a function of the peak order difference between post- and pre-synaptic neurons in the production epoch of IP. (e, f) The same as panel d, but for the stimulus1 (panel e) or stimulus2 (panel f) epoch of IC. (g) Recurrent weight as a function of the difference $|x_1 - x_2|$ between the preferred spatial locations of post- and pre-synaptic neurons and their peak order difference in the production epoch of t-SR. (h) Recurrent weight as a function of peak order difference in the sequence of neurons with the same (blue) or different (orange) preferred decision choices in the production epoch of t-DM. Shaded belt indicates s.e.m.
Fig. S10. Coding geometry and network structure in the absence of timing task requirement. (a,b) The angle and mixed variance between the subspaces coding temporal flow (F), time interval (I) and spatial information (S) in the delay epoch of t-SR, in the same style as Fig. 3i, j. (c,d) Similar to panel a,b, except for the delay epoch of t-DM, where the non-temporal information is decision choice (D). (e) The angle between the first temporal-flow PC and the first spatial (in t-SR, SR, COMP and CD) or decision-choice (in t-DM, DM and cue-DM) PC. Whisker plots: center line, median; box, 25th to 75th percentiles; whiskers, ±1.5×the interquartile range. In t-SR and t-DM, the perception epoch is studied; in SR, COMP and CD, the delay epoch is studied; in DM and cue-DM, the stimulus-presentation epoch is studied. Asterisk indicates significant (p<0.05) larger than 45° (t test). The horizontal dotted line indicates 45°, the vertical dotted line separates the spatial task group (t-SR, SR, COMP and CD) from the decision-making task group (t-DM, DM and cue-DM). The two horizontal dashed line indicate the median values of t-SR and t-DM (which respectively are the only timing task in each group) separately. (f) Mixed ratio ρ in several tasks, where ρ = \( \frac{\min(v_{\text{time}}, v_{\text{non-time}})}{\min(v_{\text{time}}, v_{\text{non-time}})} \), where \( v_{\text{min}} \) is the mixed variance, \( v_{\text{time}} \) and \( v_{\text{non-time}} \) are the variance explained by temporal and non-temporal information separately. (g) Recurrent weight as a function of the difference \(|x_1 - x_2|\) between the preferred spatial locations of post- and pre-synaptic neurons and their peak order difference in the delay epoch of SR. (h) The same as panel g, except for COMP. (i) The same as panel g, except for CD. (j) Recurrent weight as a function of peak order difference in the sequence of neurons with the same (blue) or different (orange) preferred decision choices during the presentation of the stimuli in cue-DM. Shaded belt indicates s.e.m. (k) The same as panel j, except for DM.
Fig. S11. Dynamics of the network when trained to produce long time intervals. (a-b) Perception epoch. (a) Population activity in the perception epoch in the subspace of the first three PCs. Colors indicate the time interval $T$. Stars and circles respectively indicate the starting and ending points of the perception epoch. (b) Coefficient of determination ($R^2$) that quantifies the overlap of the firing profiles of individual neurons at different $T$'s in the perception epoch, in the same style as Fig. 2d. (c-g) Delay epoch (c) Trajectory speed in the delay epoch when $T = 1200$ ms (blue) and 2400 ms (red), in the same style as Fig. 2f. (d) Ratio of explained variance of the first five PCs of manifold $M$ at the end of the delay epoch, in the same style as Fig. 2g. (e) The position of the state at the end of the delay epoch projected in the first PC of manifold $M$ as a function of $T$, in the same style as Fig. 2h. (f) The distance between two adjacent curves in the delay epoch as a function of time, in the same style as Fig. 2i. (g) The portions of monotonically decreasing (MoD), monotonically increasing (MoI) and non-monotonic (non-M) types of neurons tuned by $T$ at the end of the delay epoch, in the same style as Fig. 2k. (h-l) Production epoch. (h) Scaling index (blue) and ratio of explained variance (orange) in the subspace spanned by the accumulated scaling components, in the same style as Fig. 2n. (i) Trajectory speed in the subspace of the first three scaling components, in the same style as Fig. 2o. (j) Probability distribution of the scaling indexes of single neurons, in the same style as Fig. S2b. (k) The scaling index and explained variance of principal components, in the same style as Fig. S2c. (l) Mean activity of the last scaling component as a function of $T$, in the same style as Fig. S2e.
References

1. R Hattori, KV Kuchibhotla, RC Froemke, T Komiyama, Functions and dysfunctions of neocortical inhibitory neuron subtypes. Nat. Neurosci. 20, 1199–1208 (2017).
2. J Wang, D Narain, EA Hosseini, M Jazayeri, Flexible timing by temporal scaling of cortical responses. Nat. Neurosci. 21, 102–110 (2018).
3. AE Orhan, WJ Ma, A diverse range of factors affect the nature of neural representations underlying short-term memory. Nat. Neurosci. 22, 275–283 (2019).
4. GR Yang, MR Joglekar, HF Song, WT Newsome, XJ Wang, Task representations in neural networks trained to perform many cognitive tasks. Nat. Neurosci. 22, 297–306 (2019).
5. QV Le, N Jaitly, GE Hinton, A simple way to initialize recurrent networks of rectified linear units. arXiv https://arxiv.org/abs/1504.00941 (2015).
6. A Goel, DV Buonomano, Temporal interval learning in cortical cultures is encoded in intrinsic network dynamics. Neuron 91, 1–8 (2016).
7. CJ MacDonald, KQ Lepage, UT Eden, H Eichenbaum, Hippocampal ‘time cells’ bridge the gap in memory for discontiguous events. Neuron 71, 737–749 (2011).
8. RHR Hahnloser, AA Kozhevnikov, MS Fee, An ultra-sparse code underlies the generation of neural sequence in a songbird. Nature 419, 65–70 (2002).
9. E Pastalkova, V Itskov, A Amarasingham, G Buzsáki, Internally generated cell assembly sequences in the rat hippocampus. Science 321, 1322–1327 (2008).
10. CD Harvey, P Coen, DW Tank, Choice-specific sequences in parietal cortex during a virtual-navigation decision task. Nature 484, 62–68 (2012).
11. D Kobak, et al., Demixed principal component analysis of neural population data. eLife 5, e10989 (2016).
12. JD Murray, et al., Stable population coding for working memory coexists with heterogeneous neural dynamics in prefrontal cortex. Proc. Natl. Acad. Sci. USA 114, 394–399 (2017).