PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Encoding time in neural dynamic regimes with distinct computational tradeoffs

Shanglin Zhou[1]*, **Sotiris C. Masmanidis**[1,2], **Dean V. Buonomano**[1,3]*

1 Department of Neurobiology, University of California, Los Angeles, California, United States of America, **2** California Nanosystems Institute, University of California, Los Angeles, California, United States of America, **3** Department of Psychology, University of California, Los Angeles, California, United States of America

* zhoushanglin@gmail.com (SZ); dbuono@ucla.edu (DVB)

Abstract

Converging evidence suggests the brain encodes time in dynamic patterns of neural activity, including neural sequences, ramping activity, and complex dynamics. Most temporal tasks, however, require more than just encoding time, and can have distinct computational requirements including the need to exhibit temporal scaling, generalize to novel contexts, or robustness to noise. It is not known how neural circuits can encode time and satisfy distinct computational requirements, nor is it known whether similar patterns of neural activity at the population level can exhibit dramatically different computational or generalization properties. To begin to answer these questions, we trained RNNs on two timing tasks based on behavioral studies. The tasks had different input structures but required producing identically timed output patterns. Using a novel framework we quantified whether RNNs encoded two intervals using either of three different timing strategies: scaling, absolute, or stimulus-specific dynamics. We found that similar neural dynamic patterns at the level of single intervals, could exhibit fundamentally different properties, including, generalization, the connectivity structure of the trained networks, and the contribution of excitatory and inhibitory neurons. Critically, depending on the task structure RNNs were better suited for generalization or robustness to noise. Further analysis revealed different connection patterns underlying the different regimes. Our results predict that apparently similar neural dynamic patterns at the population level (e.g., neural sequences) can exhibit fundamentally different computational properties in regards to their ability to generalize to novel stimuli and their robustness to noise—and that these differences are associated with differences in network connectivity and distinct contributions of excitatory and inhibitory neurons. We also predict that the task structure used in different experimental studies accounts for some of the experimentally observed variability in how networks encode time.

Author summary

The ability to tell time and anticipate when external events will occur are among the most fundamental computations the brain performs. Converging evidence suggests the brain

OPEN ACCESS

Citation: Zhou S, Masmanidis SC, Buonomano DV (2022) Encoding time in neural dynamic regimes with distinct computational tradeoffs. PLoS Comput Biol 18(3): e1009271. https://doi.org/10.1371/journal.pcbi.1009271

Editor: Boris S. Gutkin, École Normale Supérieure, College de France, CNRS, FRANCE

Received: July 2, 2021

Accepted: February 8, 2022

Published: March 3, 2022

Data Availability Statement: All data are available in the main text or supplementary materials. Codes used for the simulations in this paper are available at (https://github.com/ShanglinZhou/RNN_2Intervals).

Funding: D.V.B. was supported by a NSF grant RI:2008741 and a NIH grant NS116589. S.C.M. was supported by a NIH grant NS125877. S.Z. was supported by the Marion Bowen Neurobiology Postdoctoral Grant Program at UCLA. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

encodes time through changing patterns of neural activity. Different temporal tasks, however, have distinct computational requirements, such as the need to flexibly scale temporal patterns or generalize to novel inputs. To understand how networks can encode time and satisfy different computational requirements we trained recurrent neural networks (RNNs) on two timing tasks that have previously been used in behavioral studies. Both tasks required producing identically timed output patterns. Using a novel framework to quantify how networks encode different intervals, we found that similar patterns of neural activity—neural sequences—were associated with fundamentally different underlying mechanisms, including the connectivity patterns of the RNNs. Critically, depending on the task the RNNs were trained on, they were better suited for generalization or robustness to noise. Our results predict that similar patterns of neural activity can be produced by distinct RNN configurations, which in turn have fundamentally different computational tradeoffs. Our results also predict that differences in task structure account for some of the experimentally observed variability in how networks encode time.

Introduction

The ability to predict when external events will occur, and to detect temporal regularities in the environment, are among the most fundamental computations the brain performs [1–5]. Thus, the brain must have a rich repertoire of mechanisms to tell time and perform temporal computations. Indeed, converging experimental and computational evidence indicates that a wide range of different brain areas encode time through dynamically changing patterns of neural activity [1,6–10]. These patterns can take the form of monotonic ramping of the firing rates of neurons, or so-called population clocks that can take the form of neural sequences or complex patterns of neural activity [1,11].

Experimental and computational analyses of the different neural encoding schemes for the representation of time have focused primarily on the discrimination and production of isolated intervals or durations. However, the computational requirements for processing temporal information go far beyond merely requiring a timer to discriminate or produce a single duration or interval. Some forms of temporal processing require the ability to smoothly scale a time-varying motor pattern. For example, the ability to play a song on the piano at different tempos, or catch a ball thrown at different speeds, requires that the underlying patterns of neural activity unfold at different speeds [12–15]. Indeed, some tasks in animal studies explicitly require animals to exhibit temporal scaling: depending on context cues or training blocks animals must temporally scale their motor response [14,16–18]. In contrast, other timing tasks are categorical in nature, for example in the language domain phrasal boundaries are based in part on a categorical boundary of the pause between phonemes—e.g., *great eyes* x *gray ties* [19,20], similarly, in the motor domain, the distinction between a double-click and two single clicks of a computer mouse is categorical. Furthermore, in both the human and animal literature standard temporal bisection tasks require subjects to make a two-alternative forced-choice categorical judgment regarding whether a stimulus was short or long [21,22].

It remains unclear if different computational requirements, such as the need to exhibit temporal scaling or categorical timing, rely on similar or fundamentally different underlying neural mechanisms to encode time. Consider a task in which an animal has to produce two intervals—e.g., in response to two different sensory cues. Generally speaking, three encoding schemes could allow the same network to produce these two different intervals: absolute timing, temporal scaling, and stimulus-specific timing. Under *absolute* timing the neurons would respond at the same moments in time during both the production of short and long intervals but additional neurons would be active during the long interval; in a *temporal scaling* scheme neurons encode the same relative time during both short and long intervals; and in a stimulus-specific code, there would be unrelated patterns for each interval (e.g., entirely different neural sequences for the short and long interval). These different schemes possess specific computational tradeoffs regarding their suitability for temporal scaling versus categorical timing.

To date, a large diversity of neural signatures for the encoding of time—including scaling, absolute timing, and stimulus-specific timing—have been observed during tasks that require animals to discriminate or produce multiple intervals [14,16–18,23–30]. Here we propose that some of this diversity is driven by task structure, and examine whether task structure influences the way recurrent neural networks may encode time. To address this hypothesis we trained RNNs on two tasks with identical output motor requirements and characterized how the networks encode time and generalize to novel stimuli. Our results establish that subtle differences in task structure lead to neural dynamic regimes that are better suited for temporal scaling or categorical timing.

Results

To begin to understand how task structure might shape how time is encoded in neural networks, we trained recurrent neural network models (RNNs) on one of two tasks inspired by previous experimental studies [14,18,23]. The RNNs were based on firing rate units with distinct populations of excitatory (80%) and inhibitory (20%) units. We will refer to the tasks as the 2-Context (Fig 1**A**) and 2-Stimulus (Fig 1**B**) tasks—critically, the timed motor outputs were identical in both tasks, requiring the production of either a short or long response. In the 2-Context task [e.g., 14,18], the *Go* cue (500 ms) indicated the onset of the trial (t = 0), and the analog level of a continuous context input signaled whether a trial is short or long. In the 2-Stimulus task, the short and long interval trials were cued by two distinct transient inputs [23]. In both cases, the short and long intervals consisted of a ramp-up of the output unit starting at the interval midpoint—a function that approximates the behavioral response rate of animals trained to correctly time their movements [23].

Performance was quantified by the ratio of correctly timed trials (see Methods) and the error between the actual output and the target. RNNs trained on both tasks learned to produce the same appropriately timed motor output (Fig 1**C–1F**), although the RNNs trained on the 2-Context task required fewer training trials to reach the same performance level (n = 20 simulations, two-sample two-sided t-test, $t_{38} = 9.75$, $P < 0.0001$).

Generalization to novel intervals

Having shown that RNNs can produce the same temporal output patterns when trained on two similar tasks, we next asked a key question: are there significant functional differences between how the RNNs trained on the different tasks perform in response to novel input conditions? To answer this question we examined generalization to untrained input conditions. To test the generalization in the 2-Context task we varied the amplitude of the context cue between the range of the trained values (0.75 = short; 0.25 = long). Interestingly the network exhibited fairly smooth generalization—i.e., in response to intermediate context levels it produced intermediate motor intervals (Fig 2**A**)—a finding consistent with previous computational studies [12,14]. To test generalization in the 2-Stimulus task we mixed the ratio of activation of the two stimulus cues—during training [1, 0] corresponded to short and [0, 1] to long, during testing an intermediary 50/50 mixed input corresponded to [0.5 0.5]. In contrast to the 2-Context task, the RNNs trained on the 2-Stimulus task did not generalize, but the

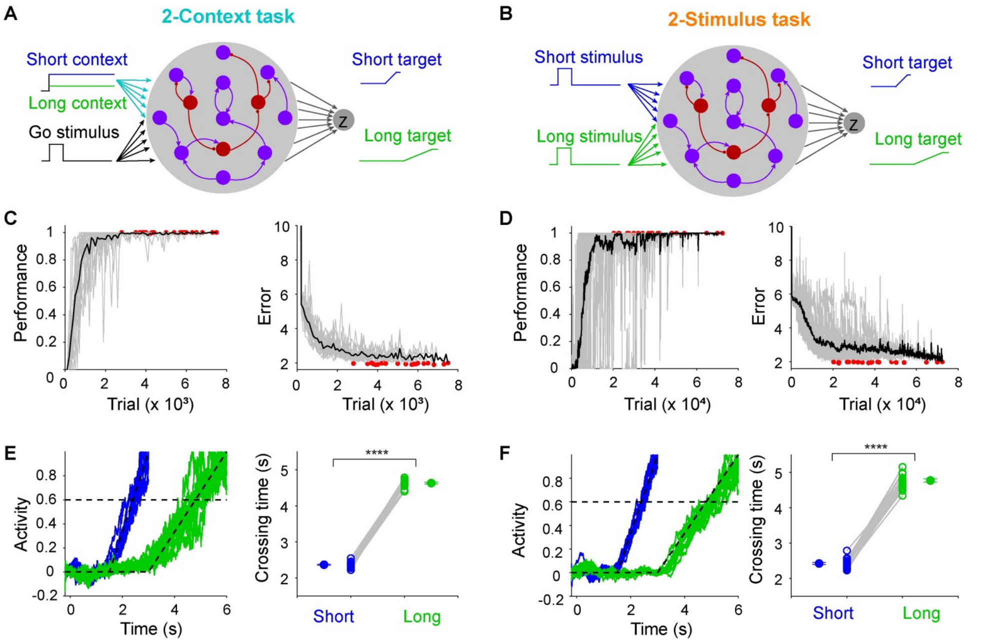

Fig 1. RNNs were trained on one of two timing tasks, both of which required producing the same timed output patterns. (**A**) Schematic of the 2-Context task. Each RNN was composed of 200 units—80% excitatory units (purple) and 20% inhibitory units (dark red)—and received a *go* and a *context* input. The context level signals the interval length to be produced: high = short (3 s, blue), low = long (6 s, green). (**B**) Schematic of the 2-Stimulus task. The same RNN was used in both tasks, except that the short- and long-interval was cued by two different inputs that were transiently activated. (**C**) Learning curve for the performance of 20 RNNs trained on the 2-Context task. Percentage of trials in which the timing of the output unit met criteria (left) and the error between the output and target (right). Gray traces represent results of each RNN, red dots denote the end of training for a given RNN, and the black trace represents the mean performance. (**D**) Same as in (**C**) but for the 2-Stimulus task. (**E**) Output traces across ten short (blue) and long (green) trials from an RNN trained on 2-Context task (left). Mean crossing times for long interval is significantly higher than that for short interval (right, $n = 20$ simulations, paired t test, $t_{19} = 77.70$, $P < 0.0001$). Dashed lines denote the targets and threshold. (**F**) Same as (**E**) but for 2-Stimulus task ($n = 20$ simulations, paired t test, $t_{19} = 45.79$, $P < 0.0001$).

https://doi.org/10.1371/journal.pcbi.1009271.g001

RNNs did not exhibit catastrophic degradation or behave randomly. Rather, the RNNs expressed categorical timing: the output intervals clustered near the short or long intervals (Fig 2B), essentially exhibiting a winner-take-all behavior.

To quantify these generalization patterns we measured the slope of a sigmoid fit between input levels and output intervals, as well as the correlation between them (Fig 2C and 2D, see Methods). The slope of the sigmoid was significantly lower in the 2-Context fits—indicating a quasi-linear relationship between context input level and produced intervals. The sigmoid slope was significantly higher in the 2-Stimulus task, consistent with the prototypical sigmoidal signature of categorical discrimination (Fig 2D, left panel). Similarly, the Pearson correlation coefficients further supported the observation that the input-interval relationship was much more linear in the 2-Context task compared to the 2-Stimulus task (Fig 2D, right panel).

In addition to the above accuracy measures, we also quantified the precision of timing across the different generalization conditions, as the standard deviation of the crossing time of each trial (Fig 2E). The precision for the 2-Context task was high (low standard deviation) for all the stimulus conditions. In contrast, in the middle range for the 2-Stimulus task precision was very low. This was mainly due to categorical timing, i.e., in some stimulus conditions, the motor output would randomly be attracted towards the short or long interval. Taken together, RNNs trained on the 2-Context task were far superior at generalizing to novel intervals in

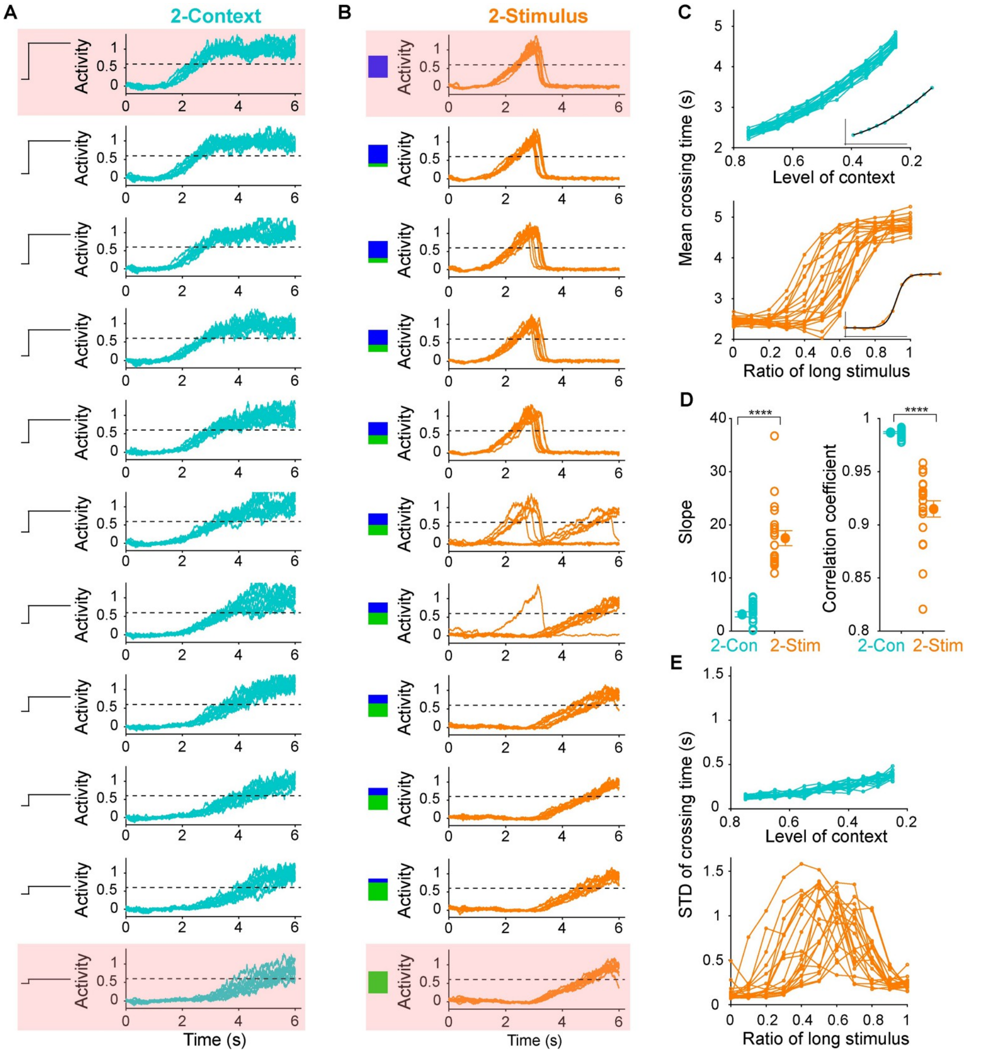

Fig 2. RNNs trained on the 2-Context task exhibited smooth generalization to novel intervals, while RNNs trained on the 2-Stimulus task exhibited categorical timing. (**A**) Output traces of an RNN trained on the 2-Context task across different context input levels. Dashed-black lines denote the output threshold used to quantify timing. Pink squares denote the trained conditions. (**B**) Similar to (**A**) but for the 2-Stimuls task. The blue and green squares represent the ratio of activation of the two input units. (**C**) Plots of the mean crossing time for each RNN across input conditions for the 2-Context (top) and 2-Stimulus (bottom) tasks. Insets, examples of the sigmoid-function fits for a single RNN (black). (**D**) Left, mean slope of the sigmoid fits for 2-Stimulus task is significantly higher than that for the 2-Context task ($n = 20$ simulations for each, two-sided t test, $t_{38} = 9.69$, $P < 0.0001$). Right, correlation coefficient between mean crossing times and input conditions for 2-Context task is significantly higher than that for the 2-Stimulus task ($n = 20$ simulations for each, two-sided t test on Fisher-transformed values, $t_{38} = 17.39$, $P < 0.0001$). The absolute correlation coefficient values are shown because in the 2-Context task the correlations are negative. (**E**) Standard deviations of the crossing times for each RNN in the 2-Context (top) and 2-Stimulus (bottom) tasks, as a function of input conditions.

https://doi.org/10.1371/journal.pcbi.1009271.g002

terms of both timing accuracy and precision, however, the RNNs trained on the 2-Stimulus task exhibited categorical timing.

By design, the key difference in the tasks is that in the 2-Context task there is a continuous input signaling the target interval throughout the task, whereas in the 2-Stimulus task two different input weight vectors signal the desired interval, and each of these inputs is only active for a brief period. To further determine whether the difference of the generalization patterns is robust to the input parameters, we manipulated the ‘similarity of the inputs corresponding to the short and long intervals in both tasks. Specifically, for the 2-Context task, different analog pairs of context level were used, ranging from (0.95, 0.05) to (0.55,0.45). For the 2-Stimulus task, we gradually increased the similarity by increasing the overlap ratio between the two inputs—proportions of the same elements in the two input weights (S1A and S1B Fig). In all five conditions, the generalization performance for the 2-Context task was better than that for the 2-Stimulus task (S1**C**–S1**E** Fig). While the tasks were designed to capture features of those used in behavioral experiments [14,18,23], in the 2-Context task the onset of the Go and Context stimuli redundantly signal trial onset (t = 0). Thus to understand the influence of the Go stimulus we also performed simulations without the Go stimulus in the 2-Context task (S2**A** Fig). As expected, omitting the Go stimulus left the generalization performance largely unchanged compared to the standard 2-Context task with Go stimulus, and still significantly better than that for the 2-Stimulus task (S2**B** Fig). Finally, to confirm that it is the presence of the continuous context input that plays a critical role in the differential generalization patterns, we performed “2-Context” simulations in which the short and long intervals were cued by a transient “context” stimulus rather than a persistent context input. Consistent with our expectations based on previous results [12,14,30], in the absence of a continuous context input the generalization was more consistent with categorical timing (S3 Fig)

Additional simulations confirmed that the difference of the generalization performance between the 2-Context task and 2-Stimulus were robust to the change of several hyperparameters including the initial gain (S4**A**–S4**C** Fig) and connection probability (S5**A**–S5**C** Fig) of the recurrent weights.

Potential dynamic regimes underlying the encoding of multiple intervals

Converging experimental and theoretical evidence indicates that a broad range of neural dynamic regimes encode time. But to date, these different regimes have not been contrasted in terms of their ability to encode multiple intervals and lead to generalization or categorical timing, or robustness to noise. Here we examine three broad potential strategies for the encoding of two intervals: scaling, absolute, and stimulus-specific codes. To illustrate these three strategies we consider how a network of neurons could encode both a short (3 s) and long (6 s) intervals (Fig 3)—note that while we use neural sequences to contrast the three encoding schemes, the same classification applies to other codes for time, including ramping activity. In a temporal scaling strategy (Fig 3**A**), the dynamics of each unit for the short interval is linearly scaled in time to produce the long interval (Fig 3**B**), which at the level of single units leads to two overlapping curves (Fig 3**C**). Similarly, when the neural trajectories of the entire population are projected into a low-dimensional space by principal component analysis the trajectories are also overlapping (Fig 3**D**). Under an absolute encoding strategy (Fig 3, middle panels) the temporal profile of each unit during the short interval does not change during the long interval. The long interval simply relies on recruiting additional neurons that have later temporal fields. Thus in PCA space, the curves for the short interval matched the first half of that for the long interval. In a stimulus-specific strategy (Fig 3, right panels), the temporal profile of each neuron is essentially uncorrelated during the short and long intervals. Thus in PCA space, the trajectories of the neural patterns of activity produced during the short and long intervals are distinct from one another.

Importantly, these encoding strategies are not necessarily mutually exclusive within a population of neurons. A network could use mixed encoding strategies in which different neurons are best described as scaling from one interval to another, while others encode absolute time. It is also possible that the dynamics of a given unit exhibit an absolute code early in a trial

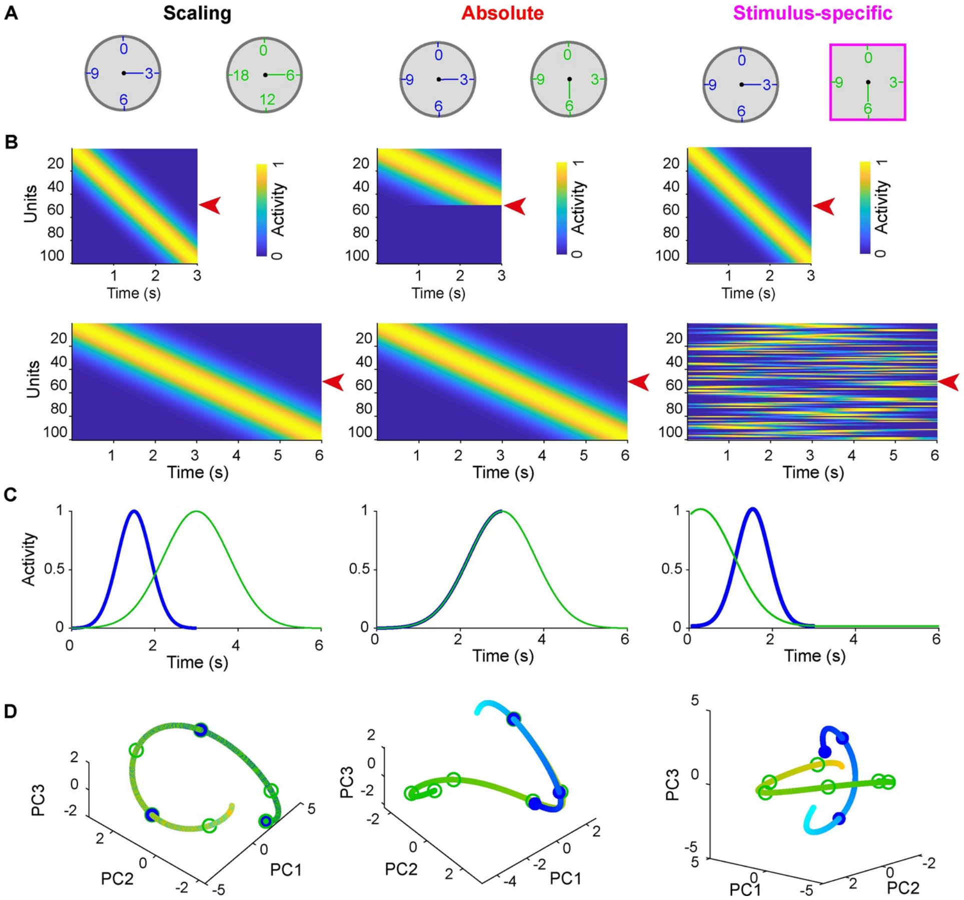

Fig 3. Three strategies for the encoding of two intervals by the same group of neurons. (**A**) Schematic of three potential strategies for timing two intervals: scaling, absolute, and stimulus-specific from left to right. (**B**) Prototypical dynamics for each of the encoding schemes for a population of units during production of the short (top) and long (bottom) intervals. (**C**) Activity traces of the units denoted by the red arrows in (**B**) for short (blue) and long (green) intervals. (**D**) Trajectories of three PCA components for short (cyan-blue) and long (yellow-green) interval for the corresponding population dynamics. The gradient colors (from the light to the dark) denote the flow of time. Circles denote the time points of the 1st, 2nd, 3rd, 4th, 5th, and 6th seconds.

https://doi.org/10.1371/journal.pcbi.1009271.g003

followed by scaling later in the trial. Note, however, that it would not make sense to consider a case in which a unit undergoes scaling early in a trial and then exhibits absolute timing.

We next describe how to quantify these three schemes both at the level of the neural population and of individual neurons in RNNs trained on either 2-Context or 2-Stimulus tasks.

Task structure differentially shapes the time encoding strategies at the population level

In order to visualize the internal dynamics of the RNNs we first plotted the normalized activity observed during the short and long intervals sorted according to the latency of peak activity for each unit during the short interval (Fig 4**A** and 4**B**, left panels), and sorted by the long intervals (Fig 4**A** and 4**B**, right panels). Interestingly, although the target output was a ramping pattern, relatively few RNN units appeared to be ramping. Rather, the global activity patterns in both tasks might be best conceptualized as neural sequences. Yet, while the self-sorted sequences appeared to be visually similar for both tasks, the cross-sorted sequences were dramatically different. Specifically, in the 2-Context task it appeared that neurons fired in the

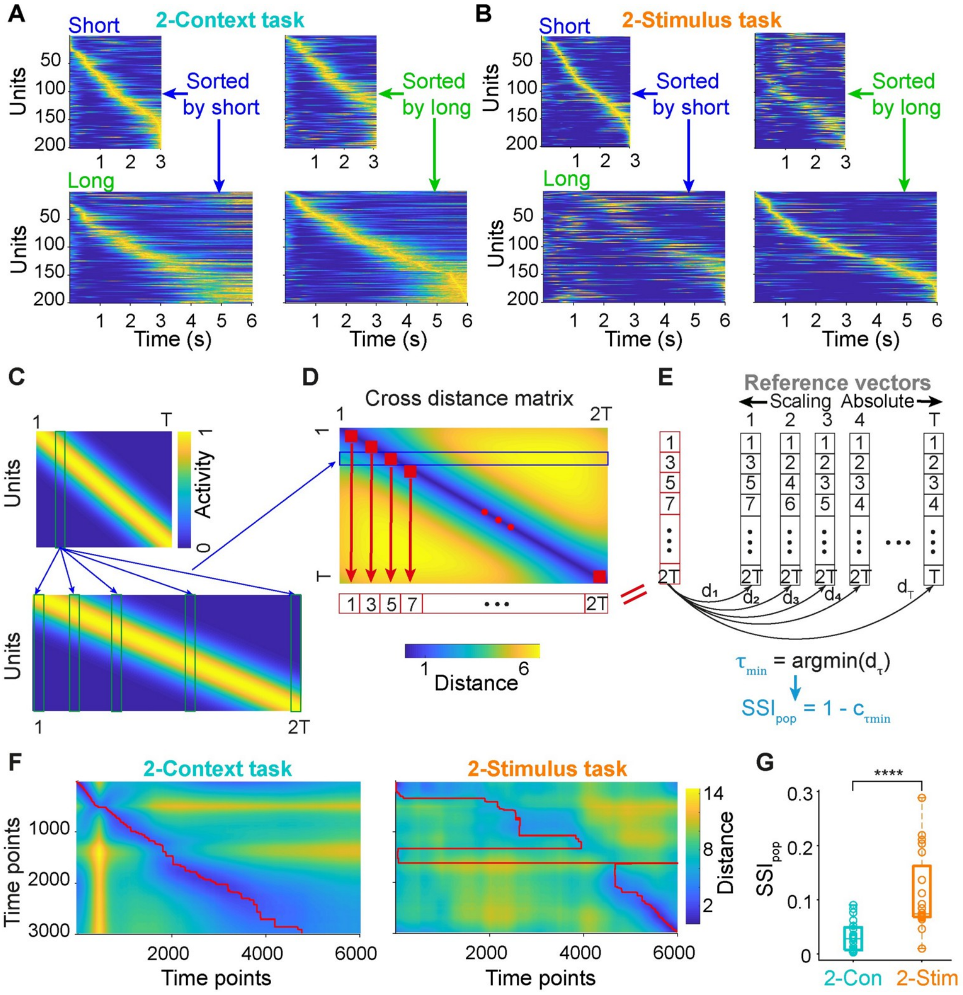

Fig 4. Distinct population dynamics in RNNs trained on the 2-Context and 2-Stimulus task. (**A**) Population activity for short (top) and long (bottom) intervals sorted according to the peak activity latency during short (left) and long (right) intervals for RNNs trained on the 2-Context task. (**B**) Same as **A** for the 2-Stimulus task. (**C**), (**D**), (**E**) Schematic of the calculation of the stimulus-specific index (SSI_{pop}). A prototypical neural sequence that undergoes pure temporal scaling from the short (top) to long (bottom) intervals is used as an example (**C**). The vectors of the pairwise time points from the short and long dynamics are used to calculate all pairwise Euclidean distances, and these pairwise distances comprise the cross-distance matrix (**D**), in which a row (e.g., blue rectangle) represents the distances between one column vector of short dynamics and all column vectors during the long dynamics. The minimal index vector (red vectors in (**D**) and (**E**)) represents the indices along the x-axis that corresponds to the minimum distances for each row of the cross-distance matrix (red squares). A series of reference vectors that vary from pure scaling to pure absolute timing (black vectors) are compared to the minimal index vector, and a value τ_{min} is defined as the τ at which the pairwise distance reaches the minimum. Finally, the correlation coefficient between the minimal index vector and the absolute-scaling reference vector at τ_{min} is used to calculate SSI_{pop}. (**F**) Cross distance matrices for an example simulation of the 2-Context (left) and 2-Stimulus tasks (right). Red lines denote the indices of the minimum values for each row. (**G**) SSI_{pop} for RNNs trained on the 2-Stimulus task is significantly higher than that for 2-Context task ($n = 20$ simulations for each, two-sided Wilcoxon rank-sum test $P < 0.0001$). Boxplot: central lines, median; bottom and top edges, lower and upper quartiles; bottom and top whiskers: extremes.

https://doi.org/10.1371/journal.pcbi.1009271.g004

same order for both the short and long intervals—suggestive of a scaling encoding strategy. However, in the 2-Stimulus task the cross-sorted PSTHgrams revealed a more complex relationship between the spatio-temporal patterns of activity during the short and long intervals—suggestive of a more stimulus-specific encoding strategy.

To quantify if the neural dynamics observed in the 2-Context and 2-Stimulus tasks were more consistent with a scaling, absolute, or stimulus-specific code, we first developed as stimulus-specific index (SSI_{pop}) based on previously described geometric approaches [12,30,31]. We started with the cross-Euclidean distance matrix between population dynamics for short and long intervals (see Methods), which compares the similarity of the activity across all time pairs during the short and long intervals (Fig 4C and 4D, example based on a case of perfect scaling of the entire population). We then extracted the index (time bin of the long interval) corresponding to the minimum value along each row of the cross-time distance matrix (red square in Fig 4D), which results in a vector of the time points in the long-interval that are closest to each of the time points in the short-interval: the minimal index vector (red row vector in Fig 4D and column vector in Fig 4E). This minimal index vector was then matched to all possible reference vectors representing perfect scaling codes to a perfect absolute code (black column vectors in Fig 4E) by computing the distances d_{τ} between each pair (Fig 4E). The reference vector with the minimum distance ($d_{\tau min}$) to the minimal index vector denoted the best absolute-scaling vector. The correlation ($c_{\tau min}$) between the best absolute-scaling vector and the minimal index vector determines how good the match is: 1.0 reflects perfect scaling, absolute timing, or a perfect mixture of absolute and scaling code. However, the correlation will be low or even negative in the case of a stimulus-specific code. Therefore, SSI_{pop} was defined by 1-$c_{\tau min}$ (Fig 4E), meaning that both perfect scaling and absolute timing would result in an $SSI_{pop} = 0$, and the stimulus-specific code would be proportional to SSI_{pop}.

We calculated SSI_{pop} for all 20 RNNs in both the 2-Context and 2-Stimulus tasks. SSI_{pop} was significantly higher during the neural dynamics of the 2-Stimulus task compared to the 2-Context task (Fig 4G), indicating that dynamics observed during the 2-Stimulus task reflected a stimulus-specific encoding strategy more so than the 2-Context task. However, consistent with the visual inspection of the dynamics and distance matrices (Fig 4A and 4F), it is clear that the 2-Stimulus task was not entirely accounted for by a stimulus-specific strategy, suggesting a mixed code. Thus we next examined the three encoding strategies from the perspective of the individual units in the network.

Task structure shapes timing encoding strategy at the level of single units

To understand whether the encoding of the short and long intervals was most consistent with a scaling, absolute, or stimulus-specific code at the level of single units, we used a previously described measure of absolute-versus-scaling index (ASI) [23], and incorporated a novel stimulus-specific index (SSI_{unit}) into the framework. Much as SSI_{pop} quantifies how different the dynamics of two neural populations are, SSI_{unit} quantifies how different the firing-rate profiles of a unit are during a short versus long trial (see Methods). More specifically, for a given unit, a high SSI_{unit} implies the temporal profiles during two trials are not related to each other through scaling, absolute timing, or a mixture of both with the absolute part followed by the scaling part. A low SSI_{unit} implies that the temporal profiles are related through scaling, absolute timing, or a mixture of both, thus justifying the use of the ASI to further quantify scaling versus absolute timing. To calculate the SSI_{unit} we first time-warped the temporal profile of a unit during the long interval into a series of reference absolute-scaling traces spanning from pure scaling to pure absolute timing with a mixture of both in between (Fig 5A). These reference traces were defined by a "breaking point" τ marking the transition from absolute timing to scaling ($\tau = 0$ reflects perfect scaling and $\tau = T_{short}$ reflect absolute timing). All reference traces were compared with the short dynamics by computing the Euclidean distance at each τ (d_{τ}). The reference trace with the minimum distance ($d_{\tau min}$) denoted the best match with the actual temporal profile of the unit. Finally, as with $SSI_{pop,}$ the SSI_{unit} was defined as 1.0 minus

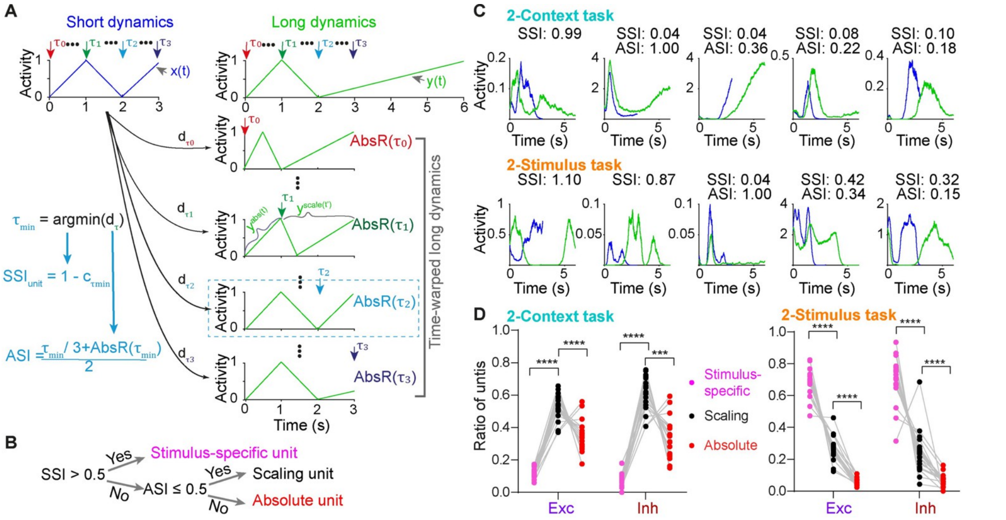

Fig 5. Different distribution of stimulus-specific, scaling, and absolute units between the 2-Context and 2-Stimulus tasks. (**A**) Schematic of the definitions of the stimulus-specific index (SSI_{unit}) and absolute vs. scaling index (ASI) at the single unit level. Consider a hypothetical firing rate profile of a unit during a short (blue, x(t)) and long (green, y(t)) trial. As described in Methods, a series of time-warped long dynamics are generated at breaking point τ_x: before τ_x the dynamics are the same during both the short and long intervals (absolute timing, $y^{abs}(t)$); after τ_x the dynamics is the scaled version of the corresponding original long dynamics ($> \tau_x$, scaling timing, $y^{scale}(t')$). Pairwise Euclidean distance between short dynamics and all time-warped long dynamics are computed at each τ_x. The point at which the distance is minimal defines τ_{min} and is used to compute the SSI_{unit} as in SSI_{pop}. To compute the ASI, a normalized measure of the distance before and after τ_{min} is calculated (AbsR) as described in Methods to quantify the weighting factor for the absolute part (before τ_{min}) and the scaling part (after τ_{min}). ASI is defined by τ_{min} and the weighting factor based on AbsR(τ_{min}). (**B**) For a given unit, the SSI_{unit} is computed first, and if the SSI_{unit} is higher than 0.5, it is classified as stimulus-specific unit. If the SSI_{unit} is lower than 0.5, its ASI is computed, and it is classified as scaling unit if its ASI is lower than 0.5, otherwise as an absolute unit. (**C**) Dynamics of five example unit traces for short (blue) and long (green) intervals for the 2-Context (top) and 2-Stimulus (bottom) tasks, the corresponding SSI and ASI values are shown on top. Notice that for a given unit, ASI is only computed only when its SSI_{unit} is lower than 0.5. (**D**) For the 2-Context task (left), most units are classified as scaling units—for both excitatory and inhibitory units (n = 20 simulations, two-way ANOVA with repeated measures, for the unit classification factor: $F_{(2,\,38)} = 114.4$ and $P < 0.0001$, posthoc Tukey tests $P < 0.0001$). For the 2-Stimulus task (right), stimulus-specific units are the most common ($n = 20$ simulations, two-way ANOVA with repeated measures, $F_{(2,\,38)} = 181.5$ and $P < 0.0001$, posthoc Tukey tests $P < 0.0001$).

https://doi.org/10.1371/journal.pcbi.1009271.g005

the correlation between the temporal profile during the short intervals and the reference trace at τ_{min} ($c_{\tau min}$). For a given unit with a low SSI_{unit} (≤ 0.5), we went on to calculate its ASI which is also based on τ_{min} (see Methods). With the SSI_{unit} and ASI in hand, we classified a given unit as either a stimulus-specific unit ($SSI_{unit} > 0.5$), a scaling unit ($SSI_{unit} \leq 0.5$, $ASI \leq 0.5$), or an absolute unit ($SSI \leq 0.5$, $ASI > 0.5$) (Fig 5B).

This approach allowed us to classify each unit of the network and contrast the distribution of temporal classifications between the 2-Context and 2-Stimulus tasks. These analyses revealed that RNNs exhibit a mixed encoding strategy, exhibiting a broad range of scaling, absolute, and stimulus-specific units (Fig 5C). However, there were highly significant differences in the distributions of temporal classes between the RNNs trained on the 2-Context and 2-Stimulus tasks (Fig 5D). The 2-Context RNNs were dominated by scaling units, while 2-Stimulus RNNs had more stimulus-specific units. The results partially explain why 2-Context RNNs were better at generalizing to novel intervals. Because our RNN structure obeyed Dale's law it was possible to contrast the encoding strategies of excitatory and inhibitory neurons. Interestingly the distribution of scaling, absolute, and stimulus-specific cells appeared similar between excitatory and inhibitory neurons (Fig 5D).

To establish a causal relationship between the distribution of temporal classes to the functional properties of the RNNs we selectively deleted units of different classes from the RNNs trained on both tasks (S6**A** Fig). We then investigated how the performance changed in response to these deletions. Performance and error across six deletion manipulations (stimulus-specific, scaling, and absolute temporal-classes for the excitatory and inhibitory populations) revealed inhibitory scaling units more severely impaired RNN function (S6**B** and S6**C** Fig) for the 2-Context task. In contrast, no single manipulation condition more severely affected both performance and error in the 2-Stimulus task (S6**D** and S6**E** Fig). Somewhat surprisingly these results reveal that in the case of the 2-Context task a single subtype of inhibitory neurons—those that were classified as scaling units—are the most critical for network dynamics and encoding time. Whereas in the 2-stimulus task the coding strategy can be considered to be truly mixed, in the sense that all temporal classes and excitatory-inhibitory neurons seem to contribute more or less equally to the underlying dynamics and the encoding of time.

Task structure differentially shapes the relationship between recurrent dynamics and input/output space

After quantifying how the different task structures shaped the encoding strategies, we sought to determine if the differences can be understood in terms of the relationship between RNN dynamics and the input/output subspaces. Generally, recurrent dynamics is driven by two sources: the interaction between the inputs and input weights, and between recurrent activity and recurrent weights. To start to understand how the inputs affected the recurrent dynamics and how the recurrent dynamics would lead to the output through the output weights, we first performed the principal component analysis on the concatenated dynamics of both intervals for each task (S7**A** and S7**B** Fig)—the first three PCs for the 2-Context task explained more variance than that for the 2-Stimulus task (88.15±0.75% vs 69.72±0.73%, S7**C** Fig). We then projected the recurrent dynamics into the low dimensional space spanned by the first three PCs (S7**A** and S7**B** Fig). Visually in PC space, the dynamics of the two intervals for 2-Context task orbited close to each other, while that for the 2-Stimulus task formed two distinct trajectories—consistent with our findings that 2-Context task tended to use an absolute-scaling strategy while 2-Stimulus, a stimulus-specific strategy. These observations were further established by plotting the dynamics in response to generalization conditions (Fig 2). In the 2-Context task the dynamics across different inputs smoothly transitioned to nearby trajectories, while in the 2-Stimulus task the trajectories clustered around the two trained (short and long) trajectories (S8 Fig).

To directly compare the relationship between the recurrent dynamics across time and the input/output weights, we projected the input weights—Input_{Go} and $\text{Input}_{\text{Context}}$ for the 2-Context task, $\text{Input}_{\text{Short}}$ and $\text{Input}_{\text{Long}}$ for the 2-Stimulus task—and the output weights into the same PC space. We then computed the pairwise angles between the projected input/output vectors and each segment vector of recurrent dynamics across time (see Methods) (S7**A** Fig) for both tasks. Interestingly, for the 2-Context task the dynamics of both intervals first evolved in the Input_{Go} input direction as revealed by the small angle for the first 2 segments. After that, both trajectories stayed in a plane almost orthogonal to the Go input till the end of the trial. The dynamics were almost orthogonal to the $\text{Input}_{\text{Context}}$ at the beginning (with angles close to 90 degrees) and then the angle decreased in the middle period and increased again to about 90 degrees at the later period. Finally, for output weights, the angle stayed close to 90 degrees at the beginning then it decreased to a low level till the end of the trial indicating that the dynamics followed the output weights directions in the later period of the trials to better generate the target ramp staring at the middle point of each trial (S7**D** Fig).

For the 2-Stimulus task, the dynamics of short and long intervals started to follow their corresponding input directions and then went to the opposite directions after input offset and stayed almost orthogonal thereafter. While for the output weights, the angle started at around 90 degrees and then decreased around the start point of the target ramp then it increased at the end of the trials to the opposite direction (S7**E** Fig).

Task structure differentially shapes the learned recurrent synaptic connectivity

Ultimately the task-specific differences in RNN dynamics must be attributed to differences in input structure and the recurrent weight matrix. Thus we next characterized the relationship between the recurrent weight matrices and performance. Since our RNNs respected Dale's law, we grouped weights into the four standard subtypes: all excitatory to excitatory unit connections (E→E), all excitatory to inhibitory unit connections (E→I), all inhibitory to excitatory unit connection (I→E), and all inhibitory to inhibitory unit connections (I→I). We then completely deleted each group of synapses and quantified the change in output performance (Fig 6**A**). Interestingly, deleting all E→E connections only slightly affected the performance and error for both tasks, while deleting all other three groups decreased the performance or increased the error. Deleting the I→E connections produced the largest change in error (Fig 6**B**). We next quantified the connection probability and mean weights of each group (Fig 6**C**). Consistent with the performance and error results, I→E connections exhibited the highest connection probability and mean weights for both tasks. Interestingly, to achieve similar output performance, the two tasks seemed to rely on different strategies in the structural level: 2-Context task favored higher connection probability, while 2-Stimulus task preferred higher mean weights (Fig 6**C**).

RNNs trained for the 2-Stimulus task are more robust to noise

We have seen that RNNs trained for the 2-Context task are better suited for generalization to novel intervals and this feature is related to the underlying dynamics being governed by a absolute-scaling encoding scheme. A question that emerges from these results is whether there is a computational tradeoff between the distinct dynamic regimes observed in both tasks? For example, while the RNNs trained on the 2-Context task exhibit better generalization, do they perform worse on any other measures? As a first step to address this question we analyzed the robustness of both tasks in response to noise. In the brain, of course, neural networks are continuously subject to extraneous noise, and thus robustness to noise imposes an important constraint on biologically functional dynamic regimes [32]

As above we first trained RNNs on either the 2-Context and 2-Stimulus tasks with the standard settings, namely noise level of 0.45 (σ in Eq 1), then we tested the networks by applying different values of σ. Example output traces for the 2-Stimulus task under all noise levels tested were less scattered than that for the 2-Context (Fig 7**A**). This was supported by the fact that the mean error for the 2-Stimulus task was lower than that for the 2-Context (Fig 7**B**). For both tasks, at high noise levels, there were some incorrect trials ($< 10\%$ and no significant difference between the two tasks) in which either the output never crossed the threshold during the trial or crossed the threshold outside of the acceptance windows. We then directly contrasted the temporal precision of the correct trials and found that the standard deviations for the 2-Stimulus task were lower than that for the 2-Context task (Fig 7**C**). Taken together, we conclude that the dynamic regimes underlying timing in the predominately stimulus-specific dynamics that emerged in the 2-Stimulus task provided a computational benefit in terms of robustness to

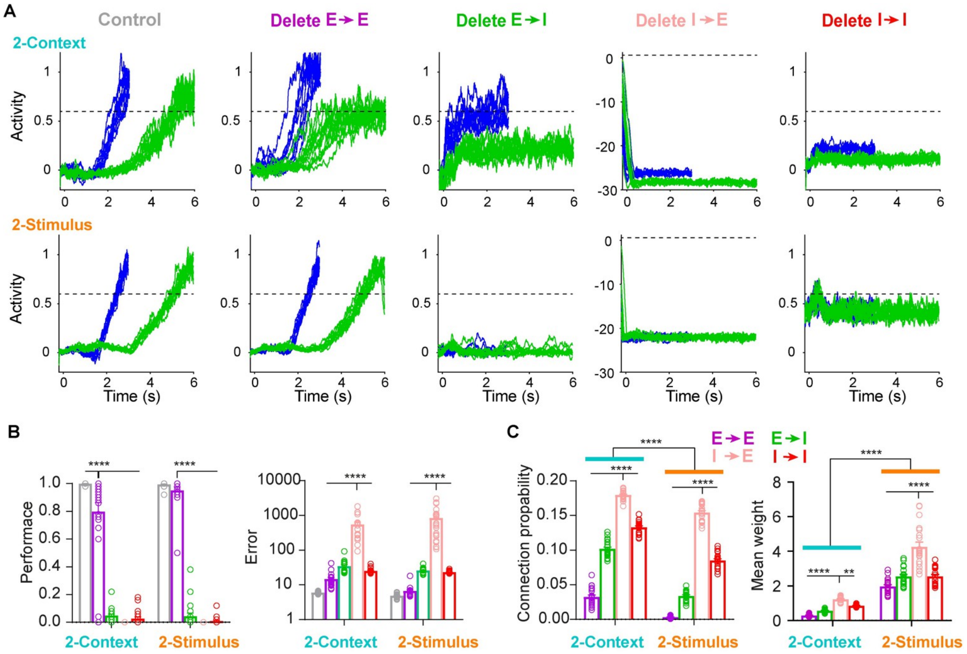

Fig 6. Differential connectivity patterns in RNNs trained on the 2-Context and 2-Stimulus tasks. (**A**) Example of the effects of deleting entire subgroups of synapses on performance in the 2-Context (top) and 2-Stimulus (bottom) tasks. From left to right, example output traces of the short (blue) and long (green) intervals for the control condition, and after deleting all excitatory unit to excitatory unit connections (Delete E→E), all excitatory unit to inhibitory unit connections (Delete E→I), all inhibitory to excitatory unit connections (Delete I→E), and all inhibitory unit to inhibitory unit connections (Delete I→I). **(B)** Mean performance (left) and error (right) of the outputs corresponding to the conditions in panel **A**. The performance for the Delete E→E condition is significantly lower than the control but significantly higher than the other conditions in 2-Context task. For the 2-Stimulus task performance for Delete E→E was not significantly worse than the control, but significantly higher than the other conditions (two-way ANOVA with mixed-effect design, $F_{4,152}$ = 823.9, $P < 0.0001$, posthoc Tukey tests $P < 0.0001$). The error for Delete I→E condition is significantly higher than the other conditions in both 2-Context and 2-Stimulus task (two-way ANOVA with mixed-effect design, $F_{4,152}$ = 39.8, $P < 0.0001$, posthoc Tukey tests $P < 0.0001$). **(C)** Left, connection probability in the 2-Context task was significantly higher than in the 2-Stimulus task(two-way ANOVA with mixed-effect design, $F_{1,38}$ = 338.3, $P < 0.0001$ for the task factor). Probability for the I→E connections is significantly higher than that for the other three conditions: E→E, E→I, I→I in both 2-Context and 2-Stimulus task ($F_{3,114}$ = 2884, $P < 0.0001$ for the connection factor, posthoc Tukey tests $P < 0.0001$). Right, the mean weight in the 2-Context task is significantly lower than that in the 2-Stimulus task (two-way ANOVA with mixed-effect design, $F_{1,38}$ = 219.1, $P < 0.0001$ for the task factor). Probability for the I→E connection is significantly higher than that for the other three conditions: E→E, E→I, I→I in both 2-Context and 2-Stimulus task ($F_{3,114}$ = 183.7, $P < 0.0001$ for the connection factor, posthoc Tukey tests). **** = $P < 0.0001$, and ** = P = 0.002.

https://doi.org/10.1371/journal.pcbi.1009271.g006

noise suggesting computational tradeoffs between different dynamic regimes for the encoding of time.

Similar to the generalization performance, the difference of the robustness to noise between the 2-Context and 2-Stimulus tasks was consistent across different input parameters (S1**F** Fig), initial gain (S4**D** Fig), and connection probability (S5**D** Fig) of the recurrent weights.

Discussion

Here we trained supervised RNNs on two simple temporal tasks that required the production of identical temporal output patterns based on previous behavioral results [14,18,23]: a ramping increase in output firing rate that peaked after either a short (3 s) or long (6 s) interval. The tasks differed only in how the short and long intervals were cued: either by a continuously presented context input (2-Context task) or by two distinct brief inputs (2-Stimulus task). In

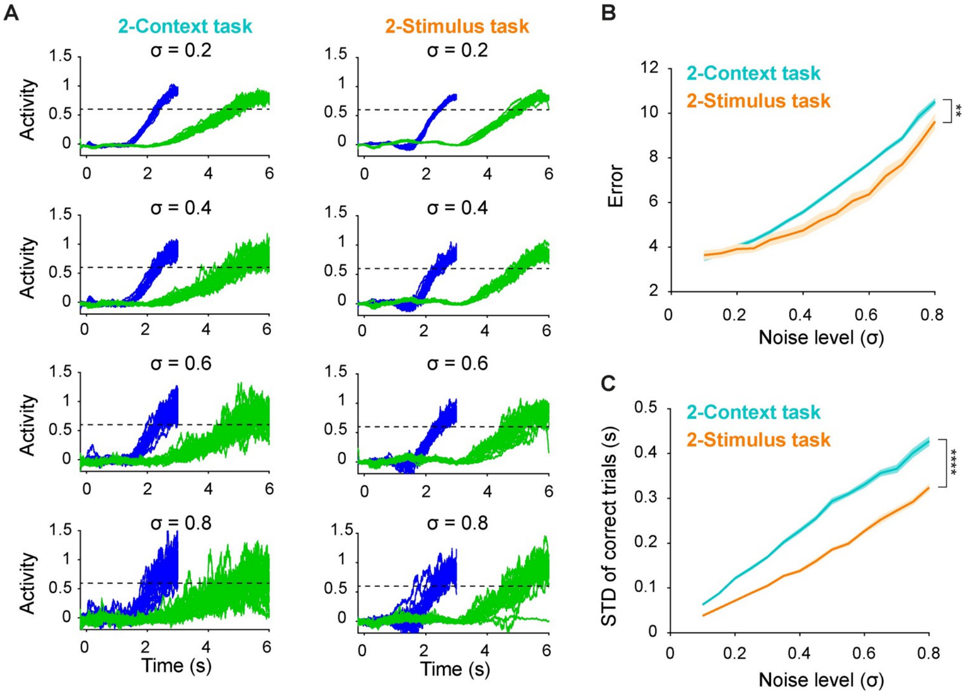

Fig 7. RNNs trained on the 2-Stimulus task were less sensitive to noise perturbations. (**A**) Output traces for short (blue) and long (green) intervals from an example RNN trained on the 2-Context (left) and 2-Stimulus (right) across different levels of noise (σ) during testing. (**B**) Mean error (across 50 trials) for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) (n = 20 simulations, two-way ANOVA with mixed-effect design, $F_{1,38} = 9.35$, $P = 0.004$). (**C**) Mean standard deviation of the time of threshold-crossing across all correct trials for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) ($F_{1,38} = 341$, $P < 0.0001$). Data are presented as mean ± SEM.

https://doi.org/10.1371/journal.pcbi.1009271.g007

principle the same dynamic regimes could have emerged and solved both tasks, yet, significantly different dynamic regimes emerged in the different tasks. Thus depending on the task RNNs encoded time in different ways, and exhibited fundamentally different computational properties, particularly regarding how the networks generalized to novel stimuli.

Neural dynamic regimes of population clocks

A converging body of experimental and computational evidence suggests that neural circuits encode time in spatiotemporal patterns of neural activity. Two experimentally relevant neural dynamics regimes by which neurons can encode time include ramping activity and population clocks. Ramping codes generally refer to monotonically increasing (or decreasing) firing rates throughout an interval [24,33–40]—in ramping codes firing rate often peaks at the time of the target interval, and in principle, a single neuron can encode time throughout the entire duration. Population clocks refer to time-varying patterns of activity in which time is encoded in the population activity of neurons, which generally exhibit nonmonotonic changes in firing rate, and importantly these dynamics are generated by the recurrent connectivity within a neural circuit [1,11,41,42]. Population clocks can include simple sparse neural sequences as well as complex spatiotemporal patterns in which a given neuron can exhibit multiple time fields [28,43–51].

In the current simulations, the target output patterns were a simple ramping pattern, yet most of the units in the RNNs were not well described as ramping units—even though it seems that this would be the simplest and most direct solution to solve the tasks. Rather, the neural dynamics observed in the RNNs studied here, are most consistent with the notion of population clocks in general and neural sequences in particular (Fig 4). These results are in line with other computational models in which neural sequences encode time [52–55]. The reason RNNs trained with supervised learning rules seem to converge to neural sequences rather than ramping activity are not well understood, but it has been recently proposed that neural sequences represent a fairly optimal encoding scheme for downstream neuron (the output unit in our case) to read out time [23].

Absolute, scaling, and stimulus-specific codes

We outlined three general temporal encoding strategies by which a population of neurons could solve temporal tasks that require producing multiple intervals (Fig 3)—such as the two tasks examined here. The scaling strategy is perhaps the most intuitive because it essentially exploits the same neural dynamics to produce both a short or long interval by altering the speed at which the dynamics unfold. Indeed, such scaling has been observed experimentally [14,16,23,26,38,56,57]. Neurons that exhibit absolute timing have also been experimentally observed, along with neurons that categorically detect the midpoint boundary between short and long intervals [14,23,26,27,56,57,58–61]. Stimulus-specific codes in which the same or different intervals can be encoded in different neural trajectories have also been described [17,47,58,62–64]. To date, however, these different encoding strategies have not been carefully analyzed or quantified. To this end, we described two general purpose quantitative measures—the ASI and SSI_{unit}—that can be applied across a wide range of single-unit data and used to classify neural responses.

These measures revealed a different distribution of unit types across the RNNs trained on the 2-Context and 2-Stimulus tasks (Fig 5). Specifically, over 50% of the units in the 2-Context RNNs were classified as scaling units, whereas in the 2-Stimulus RNNs over 50% were classified as stimulus-specific units—that is, their temporal profiles between the short and long interval were not consistent with either absolute or scaling coding strategies. This differential distribution is consistent with the intuition that because in the 2-Context task the context input is active during both the short and long intervals, and a stimulus-specific encoding strategy is more difficult to implement compared to the 2-Stimulus task—i.e., the input space of the 2-Context task is smaller. Put another way, in the 2-Stimulus task RNNs are likely to begin their trajectories at the beginning of each trial ($t = 0$) in more distant regions of neural state space than in the 2-Stimulus task.

The differential distribution of scaling, absolute, and stimulus-specific neurons accounts in part for the distinct computational features of both types of networks. Specifically, the classification of units into different temporal coding strategies allowed us to demonstrate that selectively deleting some classes impaired RNN performance more than others. Deleting a few inhibitory scaling units impaired RNN performance in the 2-Context task significantly more than deleting absolute or stimulus-selective units. In contrast in the 2-Stimulus task, all classes contributed to performance with an approximately equal weighting—reflecting a much more mixed encoding strategy [65,66].

Computational trade-offs between time-encoding dynamic regimes

The 2-Context and 2-Stimulus tasks required producing the same temporal output patterns but generated dramatically different behaviors when challenged with novel inputs. Of particular relevance was that in response to novel levels of activation of the inputs, the 2-Context RNN exhibited a smooth scaling of the temporal profile of the output. In this task, in response to the *go* stimulus, RNN generated a neural trajectory that resembled a neural sequence.

Depending on the analog value of the context input this trajectory unfolded at either a fast or slow speed to produce the short or long interval, respectively. Critically, in response to novel levels of activation of the tonic context input the velocity of the neural trajectory varied smoothly—thus generating smooth temporal scaling of the output pattern. This same property has been observed in numerous other models of timing [12,30,34,67–69]. Specifically, a single input or variable is able to modulate the velocity of the RNN dynamics in an approximately linear fashion.

In contrast to the temporal scaling behavior observed in the RNN trained on the 2-Context task, when the 2-Stimulus RNNs were tested with inputs they were not trained on (e.g., 50% Input 1 + 50% Input 2) they did not exhibit smooth generalization. Importantly, they also did not exhibit catastrophic degradation—i.e., the internal dynamics was robust to very different initial states. Rather they exhibited categorical timing—essentially a winner-take-all competition between two distinct trajectories. This result is consistent with the notion that RNNs can encode multiple neural trajectories in regimes that have been referred to as dynamic attractors [31,70], locally stable transient trajectories [71,72], or stable heteroclinic channels [73,74]. Here, two trajectories possess their own basins of attraction (or "rivers-of-attraction") which lead the activity of the network into one or the other of the two dynamic attractors.

Both temporal scaling and categorical timing are behaviorally relevant forms of timing. Specifically, some tasks require smoothly scaling the temporal output patterns, while others require categorically discriminating or producing one of distinct two intervals [12–15,21,22]. Thus, we have shown that the population clocks that emerge in RNNs can account for both temporal scaling and categorical timing and that it is possible to distinguish between both regimes based on the percentage of units that undergo scaling or stimulus-specific timing.

It is also relevant to note that RNNs learned to solve the 2-Context task in fewer training trials than the 2-Stimulus task. This may be because it is easier to adjust weights to generate a single trajectory at two different speeds than to generate largely distinct trajectories. Furthermore, during training the 2-Context task the RNN is always subject to tonic external input which in effect might facilitate learning by suppressing the potential emergence of chaotic regimes [75,76].

Experimental predictions

As is evident from the behavioral data, a wide range of distinct neural regimes, from ramping activity to a diverse range of neural population clocks, have been observed experimentally across different brain areas and behavioral tasks [for reviews see: 1,6,7,10]. Here we show that the same is true even in RNNs trained on two tasks that require the production of the same temporal output patterns. Our results thus suggest that much of the experimentally observed variability might be accounted for by relatively subtle differences in task structure. Furthermore, because most timing tasks used in laboratories tap into ecologically relevant behaviors, different tasks may encourage generalization patterns that best approximate their ecological relevance. These distinct generalization patterns will, in turn, result in time being encoded in different dynamic regimes—e.g., regimes that are well-suited for temporal scaling or categorical timing.

A number of strong experimental predictions emerge from our results. First, at the behavioral level, we predict that whether rodents are trained on the 2-Context or 2-Stimulus will lead to different generalization patterns to novel stimuli. For example, a single odor along with a tone context stimulus could be used for the 2-Context task, and two brief odors as the stimuli in the 2-Stimulus task. We predict that changing the loudness of the tone in the 2-Context task will scale the output pattern, but mixing the odors will result in categorical timing rather than the production of an intermediary interval. Second, we predict that neural recordings from animals trained on these tasks will exhibit specific neural dynamic signatures, i.e., in the 2-Context task more neurons will be categorized as scaling units compared to the 2-Stimulus task. Of course, one must take into account that results may be dependent on the brain areas being recorded. However, based on the current literature we expect this prediction to hold in those areas that have been implicated in timing across many tasks, including the striatum, supplementary/secondary motor areas, and prefrontal cortical areas.

Methods

Firing-rate RNN model

RNNs were based on firing-rate units that obeyed Dale's law (N = 200, 80/20% excitatory/inhibitory). RNN dynamics was described by the following equations:

$$\tau\frac{d\boldsymbol{x}}{dt} = -\boldsymbol{x} + \boldsymbol{W}^{rec} * \boldsymbol{r} + \boldsymbol{W}^{in} * \boldsymbol{I} + \sigma * \boldsymbol{N}(0,1) * \sqrt{2 * \tau} \quad (1)$$

$$o = \boldsymbol{W}^{out} * \boldsymbol{r} \quad (2)$$

$$\boldsymbol{r} = \min(ln(1 + e^{x}), 20) \quad (3)$$

where $\mathbf{x} \in \mathbb{R}^{N\times1}$ represents the input currents of RNN units, and firing rate vector $\mathbf{r}$ is obtained by applying a Softplus function constrained by an upper bound of 20. The time constant τ was equal to 100 ms for all units. $\mathbf{W^{in}} \in \mathbb{R}^{N\times2}$ and $\mathbf{I}$ are the input weights and external inputs, which are task-specific as described below. Each unit received independent Gaussian noise $\mathbf{N}$ (0,1) with the standard deviation of $\sigma\sqrt{2\tau}$. Unless otherwise specified, σ = 0.45. $\mathbf{W^{rec}} \in \mathbb{R}^{N\times N}$ is the recurrent weight matrix. Self-connections were absent in the network. The output (*o*) of the network is computed linearly from the output weights $\mathbf{W^{out}}$ and $\mathbf{r}$. RNNs were implemented and trained in Tensorflow starting from the code of Kim et al [77,78].

Training. Networks were trained using adaptive moment estimation stochastic gradient descent algorithm (Adam) to minimize the error between network output o and target z:

$$Error = \sqrt{\sum_{t=0}^{T}[o(t) - z(t)]^2} \quad (4)$$

where T is the total length of a given trial. The target and mask are task-dependent as described below. The learning rate was 0.01, and other TensorFlow default values were used.

Only recurrent weights $\mathbf{W^{rec}}$ and output weights $\mathbf{W^{out}}$ were trained. Unless otherwise specified, $\mathbf{W^{rec}}$ was initialized as a random sparse matrix with a connection probability of 0.2 from a normal distribution with zero mean and standard deviation (gain) of 1 and transformed to absolute values. To begin in an approximately balanced regime the inhibitory weights were multiplied by 4 for the initialization but not for training. To respect Dale's law during training a rectified linear operation was applied on $\mathbf{W^{rec}}$ to clip the weights down to zero and then excitation and inhibition were implemented by multiplying the clipped $\mathbf{W^{rec}}$ with a diagonal matrix of 1 and -1 representing excitatory and inhibitory units, respectively [78,79]. $\mathbf{W^{in}}$ was drawn from a standard normal distribution and was fixed during training.

During training, a discretization step of 20 ms was used. After training, RNNs were ported to Matlab using the trained parameters and a discretization step of 1ms was used to get the dynamics for analyses.

Parameters were updated every trial. After every 100 trials of training, the network was tested for 100 trials to compute the task performance (see below) and mean error. When task performance was higher than 97% and the mean error is lower than 2, the training was considered a success and stopped.

Interval tasks

2-Context task. Unless otherwise specified, inspired by the timing task used by previous works [14,18] in which context cues indicated the lengths of intervals, we designed a 2-Context two-interval task. In this task, the output of the RNN needs to generate either a short (3 s) or long (6 s) interval in each trial. For a given training trial with length *T*, two external inputs $\mathbf{I}^1{}_{\mathrm{go}}$ and $\mathbf{I}^2{}_{\mathrm{context}}$ were applied at $\mathrm{stim}_{\mathrm{onset}}$ after a baseline with random durations between 0.2 and 0.6 s. Specifically,

$$I_{go}(t) = \begin{cases} 1 & stim_{onset} < t \leq (stim_{onset} + 0.5) \\ 0 & otherwise \end{cases} \quad for\ both\ short\ and\ long\ trials,$$

$$I_{context}(t) = \begin{cases} 0.75\ or\ 0.25 & stim_{onset} < t \leq T,\ for\ short\ or\ long\ trials\ repectively \\ 0 & otherwise \end{cases}$$

The output targets were defined as:

$$z(t) = \begin{cases} \dfrac{t - stim_{onset} - 0.5 * Int_{target}}{0.5 * Int_{target}} & (stim_{onset} + 0.5 * Int_{target}) < t \leq (stim_{onset} + Int_{target}) \\ 1 & (stim_{onset} + Int_{target}) < t \leq (stim_{onset} + Int_{target} + 0.2) \\ 0 & otherwise \end{cases}$$

where the target intervals ($\mathrm{Int}_{\mathrm{target}}$) were 3 and 6 seconds for the short and long trials, respectively.

2-Stimulus task. Unless otherwise specified, the 2-Stimulus task was based on a two-interval odor discrimination task [23], which required the production of the identical output patterns as the 2-Context task. However, the short and long intervals were cued by two different inputs $\mathbf{I}_{\mathrm{short}}$ and $\mathbf{I}_{\mathrm{long}}$ which like the $\mathbf{I}_{\mathrm{go}}$ in the 2-Context task stepped up from 0 to 1 for a brief 0.5 s period.

Task performance. Response time for a given trial was defined as the time when the output crosses a threshold of 0.6. The correct trials were defined as those in which the output crossed the threshold within an acceptance window between $\mathrm{stim}_{\mathrm{onset}}$ + 0.5 $\mathrm{Int}_{\mathrm{target}}$ and $\mathrm{stim}_{\mathrm{onset}}$ + $\mathrm{Int}_{\mathrm{target}}$. Task performance was defined as the ratio of correct trials among all testing trials.

Unless otherwise specified, the "delay" epoch ($\mathrm{stim}_{\mathrm{onset}}$ to $\mathrm{stim}_{\mathrm{onset}}$ + $\mathrm{Int}_{\mathrm{target}}$) was used for analysis.

Generalization to novel inputs

To test how the RNN trained on the 2-Context task would generalize to novel intervals as in Fig 2, we first trained the RNN using the normal setting for the 2-Context task, namely $\mathbf{I}_{\mathrm{context}}$ of 0.75 and 0.25 for the short and long trials, respectively. Then we tested the trained RNNs by gradually varying the context level from 0.75 to 0.25 with steps of 0.05. Fifty trials of each level were obtained for analyses.

After training in the 2-Stimulus task generalization to novel inputs was tested by gradually varying the ratio of $\mathbf{I}_{\mathrm{short}}$ and $\mathbf{I}_{\mathrm{long}}$ with steps of 0.1 so that the sum of both inputs was always 1.

Correlation measure. To quantify changes in the temporal profile of the output units across different inputs during generalization tests we first computed the correlation coefficient between the mean response times (when the output crosses the threshold) and the generalization conditions for both the 2-Context and 2-Stimulus tasks (the absolute values of the correlations were used due to the negative correlation for the 2-Context task).

Sigmoid slope measure. To further quantify generalization to novel inputs in both tasks we also fitted the mean response times to the input conditions with a sigmoid function as follow:

$$y = b + \frac{a - b}{1 + e^{g*(m-x)}}$$

Standard nonlinear least square methods implemented in Matlab were used to optimize the fits. We then compared the slope *g* for both tasks. Higher *g* values reflect more categorical generalization.

Prototypical dynamical regimes for timing two intervals

To illustrate the possible neural dynamical strategies used for timing two intervals—scaling, absolute, and stimulus-specific, we generated three pairs of prototypical dynamics for the short (3 s) and long (6 s) intervals composed of 100 units with the time step of 0.001 s (Fig 3). In such settings, the dynamics for the short and long interval were represented as 100×3000 and 100×6000 matrices respectively, with the row being units and column being time points.

The dynamics for long interval were the same for all three strategies, which was described as:

$$x_i(t) = e^{\frac{-\left(t - \frac{i}{100}*6\right)^2}{2*0.8^2}} \quad for\ i = 1, 2, \ldots, 100$$

where dynamics of all units were Gaussian functions with the same variance but different means uniformly spanned the whole 6 s. The dynamics for the short interval were different for the three strategies and were defined as follows:

Scaling. The dynamics for the short interval in the scaling strategy was simply a matrix of uniform subsampling of the time dimension of the long dynamics.

Absolute. For the absolute strategy, the dynamics of the first 50 units for the short interval were the same as that for the long interval.

Stimulus-specific. For stimulus-specific example, we first uniformly subsampled the time dimension of the long dynamics matrix to 3 s. Then we randomized the order of the unit indices.

The stimulus-specific index at the population-level (SSI$_{\mathbf{pop}}$)

As in Fig 4, to quantify how well the short and long neural trajectories can be explained by the stimulus-specific strategy at the population level, we developed a novel stimulus-specific index in population-level (SSI$_{pop}$), which is largely based on establishing that the trajectories are not consistent with temporal scaling or absolute timing. We first obtained the mean population dynamics (Δt = 1 ms) for two intervals by averaging across 25 trials, which led to two matrices, $\mathbf{X_{short}}$ (200×3000) and $\mathbf{X_{long}}$ (200×6000). We computed the pairwise Euclidean distance between $\mathbf{X_{short}}$ and $\mathbf{X_{long}}$, which led to the distance matrix **D** (3000×6000). We then obtained the index of the minimum values across each row of **D,** which led to the minimal distance vector $\mathbf{I_{min}}$ (3000×1), which partially captures the relationship between the population dynamics for the short and long intervals.

Next $\mathbf{I_{min}}$ was contrasted with a reference matrix **R** (3000×3000) with τ indexing the column:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & . \; . \; . & 1 & 1 & 1 \\ 3 & 2 & 2 & 2 & . \; . \; . & 2 & 2 & 2 \\ 5 & 4 & 3 & 3 & . \; . \; . & 3 & 3 & 3 \\ 7 & 6 & 5 & 4 & . \; . \; . & 4 & 4 & 4 \\ . & . & . & . & . \; . \; . & . & . & . \\ . & . & . & . & . \; . \; . & . & . & . \\ . & . & . & . & . \; . \; . & . & . & . \\ 5998 & 5998 & 5998 & 5998 & . \; . \; . & 4499 & 2999 & 2999 \\ 6000 & 6000 & 6000 & 6000 & . \; . \; . & 6000 & 6000 & 3000 \end{bmatrix}$$

Specifically, a given column vector corresponding to τ in **R** is defined as:

$$[1, 2, 3, \ldots, \tau,\ \tau + \alpha, \tau + 2\alpha, \tau + 3\alpha, \ldots, \tau + (3000 - \tau)\alpha]$$

$$\alpha = \frac{6000 - \tau}{3000 - \tau},\ \tau = 1, 2, 3, \ldots, 2999$$

Each column vector (3000×1) in **R** represents one absolute-scaling reference profile spanning from pure scaling (τ = 1) to pure absolute (τ = 3000), with mixed profiles in between in which absolute timing transitions to scaling at τ with the scaling factor α varied to keep the length of each vector the same. We then computed the Euclidian distances between $\mathbf{I_{min}}$ and all the column vectors of **R** and extracted the vector with the minimum distance at τ_{min}, which indicates the best reference vector that can be used to explain the $\mathbf{I_{min}}$. Note that the construction of the **R** matrix accounts for units that fire throughout the entire trial—thus capturing the properties of a neuron that always fired at the end of the trial (e.g., a potential motor neuron). It is also possible to build **R** by fixing the scaling factor at 2 after each point τ, in which case the last element of each column in R would progressively change from 6000 to 3000. We have run analyses with this partial scaling approach as well with qualitatively similar results.

Finally, the SSI_{pop} was defined as:

$$SSI_{pop} = 1 - c(\tau_{min}).$$

where the $c(\tau_{\text{min}})$ is the correlation between the $\mathbf{I_{min}}$ and the reference vector at τ_{min}. For pure scaling dynamics for the two intervals as an example, $\mathbf{I_{min}}$ should be the main diagonal of distance matrix **D**, [1, 3, 5, 7, . . ., 6000], which makes $\tau_{\text{min}} = 1$, corresponding to the pure scaling reference vector. Consequently, the $c(\tau_{\text{min}})$ is 1 and SSI_{pop} is 0. That indicates that the pure scaling dynamics cannot be explained by stimulus-specific strategy but by absolute-scaling strategies, in which the dynamics of the short and long interval relate to each other in a way of absolute or scaling or mixed of both (see below for the absolute-scaling index at the single-unit level)

Stimulus-specific index and absolute-scaling index (ASI) for single units

We extended a previous description of an absolute vs. scaling index (ASI) for single units [23], by including a novel measure of the stimulus-specific profile: the stimulus-specific index at the single-cell level (SSI_{unit} as in Fig 5A). As described previously we searched for the best transformation of dynamics for the long interval (y(t)) to that for the short interval (x(t)), by concatenating an absolute portion of the long response (y^{abs}(t)) and a temporally scaled portion of the long response scaled by a factor α (y^{scale}(t')). More specifically, we searched for a breakpoint τ to divide y(t) into an absolute and scaled segment, that best matches x(t), as measured by the Euclidean distance (Dist(τ)). Specifically,

$$\alpha = (T_{long} - \tau)/(T_{short} - \tau)$$

$$Dist(\tau) = \sqrt{\sum_{t=0}^{\tau} (x(t) - y(t))^2 + \sum_{t=\tau}^{T_{short}} (x(t) - y(\tau + (t - \tau))^2}$$

$$\tau_{min} = argmin_{\tau}(Dist(\tau))$$

$$Corr(\tau_{min}) = \frac{\sum_{t=0}^{\tau_{min}} (x(t) - \bar{x})(y(t) - \bar{y}) + \sum_{t=\tau_{min}}^{T_{short}} (x(t) - \bar{x})[y(\tau_{min} + (t - \tau_{min})) - \bar{y}]}{\sqrt{\sum_{t=0}^{T_{short}} (x(t) - \bar{x})^2}\sqrt{\sum_{t=0}^{\tau_{min}} (y(t) - \bar{y})^2 + \sum_{t=\tau_{min}}^{T_{short}} (y(\tau_{min} + (t - \tau_{min})) - \bar{y})^2}}$$

$$SSI_{unit} = 1 - Corr(\tau_{min}).$$

$$W^{abs}(\tau_{min}) = 1/N_{1:\tau_{min}} \sum_{t=0}^{\tau_{min}} |[x(t) - x(0)][y(t) - y(0)]|$$

$$W^{scale}(\tau_{min}) = 1/N_{\tau_{min}:T_{short}} \sum_{t=\tau min}^{T_{short}} |[x(t) - x(\tau_{min})][y(\tau_{min} + (t - \tau_{min})) - y(\tau_{min})]|$$

$$AbsR(\tau_{min}) = \frac{W^{abs}(\tau_{min})}{W^{scale}(\tau_{min}) + W^{abs}(\tau_{min})}$$

$$ASI = (\frac{\tau_{min}}{T_{short}} + AbsR(\tau_{min}))/2$$

τ spans all possible breakpoints from 0 to T_{short} (for the short interval and T_{long} for the long interval). The segment before τ denotes the absolute period and the period after τ denotes the segment scaled by α for the long response. τ_{min} corresponds to the breakpoint with the minimal Euclidian distance Dist(τ_{min}). Different from previous work [23], we also computed the correlation coefficient between x(t) and transformed y(t), Corr (τ_{min}). Then the SSI_{unit} is defined as that 1 minus Corr(τ_{min}). In the following steps, the absolute and scaling weights are calculated between dynamics for the short interval and the time-warped dynamics for the long interval at τ_{min} with $N_{a:b}$ being the number of time points between a and b, and absolute ratio AbsR(τ_{min}) was also calculated. The absolute temporal factor corresponds to τ_{min} /T_{short}, and ASI was defined as the average of the absolute temporal factor and the AbsR (τ_{min}).

To classify each unit as a stimulus-specific, scaling, or absolute unit we first calculated SSI_{unit} for each unit. We then classified a unit as stimulus-specific if SSI_{unit} was > 0.5; if the SSI_{unit} was ≤ 0.5 then looked at its ASI and classified it as an absolute unit if ASI > 0.5, or as a scaling unit if ASI ≤ 0.5.

Unit-deletion and weight-deletion experiments

Based on the classification of units being stimulus-specific, scaling, or absolute, we ran deletion experiments to start to understand the causal role of each type of unit (S6 Fig). For a given unit to be deleted, we removed all the connections attached to that unit in connection matrix $\mathbf{W}^{\mathbf{rec}}$ as in Eq 1 and then ran the RNN with the rest parameters fixed. We tested various numbers of deleted units in each type. For a given condition, we randomly selected the deleted cells from the pool 10 times and repeated each deletion experiment for 20 trials for each interval. Then performance and error were averaged across all selections and trials.

To quantify how much each class of connection types—E→E, E→I, I→E and I→I connections—contributed to the recurrent dynamics and output performance, we performed synapse deletion experiments. Similar to the unit deletions, for a specific class of connections, we set all the weights of that group to be zeros while leaving the other weights unchanged. Performance and error were then computed for each condition (Fig 6).

Pairwise angle analysis

To understand the relationships between the RNNs trained on 2-Context and 2-Stimulus task and the input/output subspace (S7 Fig) defined by the inputs weights and output weights, we first performed principal component analysis (PCA) on the concatenated mean dynamics for the short and long intervals. We then projected the original dynamics into the first three PCs. We then binned the projected dynamics into segments of 250 ms. For a given segment, a vector was obtained by subtracting its start point from its end point. Finally, we computed the pairwise angles between all such segment vectors across time and projections of the input/output weight vectors in the same PC space.

Noise perturbation experiments

As in Fig 7, to test the robustness of the outputs of the RNN trained on the 2-Context and 2-Stimulus tasks, we first trained the two tasks with noise level $\sigma = 0.45$ as in Eq (1). We then tested the trained RNNs with various levels σ from 0.1 to 0.8 for 50 trials for each interval. We then compared the error between the outputs and targets for all trials and the standard deviation of the crossing times for the correct trials. Note that for all conditions tested, the incorrect trials were less than 10% for both tasks, and there was no significant difference for that between the two tasks.

Statistical analyses

Statistical analyses were carried out with standard functions in MATLAB (MathWorks) and Prism (GraphPad Software). The sample size, type of test, P values, and the F values for ANOVA are indicated in the figure legends. All data and error bars represent the mean and SEM except for the boxplot in Fig 4, where median and quartiles were presented. In all figures, the convention is *: $P < 0.05$, **: $P < 0.01$, ***: $P < 0.001$, ****: $P < 0.0001$.

Supporting information

S1 Fig. Generalization difference between the 2-Context and 2-Stimulus tasks are robust across different input parameters. (**A**) Training on the 2-Context task with different analog context level pairs to signal the short (blue) and long (green) intervals (top), produced similarly timed short (blue) and long (green) intervals (bottom). Dashed lines denote the threshold used to measure the crossing time. (**B**) Training on 2-Stimulus task across different levels of overlap between the two input weight vectors (overlap ratio), quantified by the angle between the two

weight vectors (top), and the corresponding learned output traces for short (blue) and long (green) intervals (bottom). (**C**) The mean (top) and standard deviations (bottom) of the crossing times across 10 simulations for the generalization experiments corresponding to the five conditions as in (**A**). (**D**) same as (**C**) but for 2-Stimulus task. (**E**) The sigmoid fit slopes of the generalization experiments in the five conditions of 2-Context task (cyan, as in **A**) were significantly lower than that for 2-Stimulus task (orange, as in **B**): two-way ANOVA, $F_{1,90} = 123.1$, $P < 0.0001$ (Left). The absolute correlation coefficients of the generalization experiments in the five conditions for 2-Context task (cyan, as in **A**) were significantly higher than that of the 2-Stimulus task (orange, as in **B**): two-way ANOVA on the Fisher-transformed data, $F_{1,90} = 374.2$, $P < 0.0001$ (right). (**F**) The mean error across all tested levels of the noise perturbation experiments in the five conditions for 2-Context task (cyan, as in **A**) is significantly higher than that for 2-Stimulus task (orange, as in **B**): two-way ANOVA, $F_{1,90} = 106.1$, $P < 0.0001$ (Left). Right, the same as the left but for standard deviations of the crossing times: two-way ANOVA, $F_{1,90} = 625.7$, $P < 0.0001$.
(TIF)

S2 Fig. Superior generalization in the RNNs trained on the 2-Context task is maintained in the absence of a Go stimulus. (**A**) Schematic of the 2-Context task without the Go stimulus. (**B**) Left, the sigmoid fit slopes in the generalization experiments for the 2-Context task without Go stimulus are not significantly different from the original 2-Context task, and still significantly lower than that for the standard 2-Stimulus task (one-way ANOVA with posthoc Tukey test, $F_{2,27} = 53.4$, ns: $P = 0.669$, ****: $P < 0.0001$). Right, the absolute correlation coefficients in the generalization experiments for the 2-Context task without Go stimulus not significantly different from the 2-Context task but significantly higher than that for the standard 2-Stimulus task (one-way ANOVA on the Fisher-transformed data with posthoc Tukey test, $F_{2,27} = 112.9$, ns: $P = 0.957$, ****: $P < 0.0001$).
(TIF)

S3 Fig. Generalization in the 2-Context task relies on continuous input. (**A**) Schematic of the standard 2-Cotnext task with persistent context input. (**B**) Schematic of a task in which the two intervals are signaled by the same brief input, but with different analog values. (**C**) Plots of the mean crossing time for each RNN across input conditions for the persistent (top) and transient (bottom) tasks. (**D**) Left, mean slope of the sigmoid fits for transient input task is significantly higher than that for the persistent 2-Context task (n = 10 simulations for each, two-sided t test, $t_{18} = 9.98$, $P < 0.0001$). Right, correlation coefficient between mean crossing times and input conditions for transient 2-Context is significantly lower than that for the persistent 2-Context task (n = 10 simulations for each, two-sided t test on Fisher-transformed values, $t_{18} = 7.52$, $P < 0.0001$). (**E**) Standard deviations of the crossing times for each RNN in the persistent 2-Context (top) and transient 2-Context (bottom) tasks, as a function of input conditions.
(TIF)

S4 Fig. Changing the initial gain of the recurrent weight matrix to 1.5 does not alter the generalization and robustness to noise differences between the 2-Context and 2-Stimulus tasks. (**A**) Plots of the mean crossing time for each RNN across input conditions for the 2-Context (top) and 2-Stimulus (bottom) tasks. Insets, examples of the sigmoid-function fits for a single RNN (black). (**B**) Left, mean slope of the sigmoid fits for 2-Stimulus task is significantly higher than that for the 2-Context task ($n = 20$ simulations for each, two-sided t test, $t_{18} = 6.91$, $P < 0.0001$). Right, correlation coefficient between mean crossing times and input conditions for 2-Context task is significantly higher than that for the 2-Stimulus task ($n = 20$ simulations for each, two-sided t test on Fisher-transformed values, $t_{18} = 16.56$, $P < 0.0001$). (**C**)

Standard deviations of the crossing times for each RNN in the 2-Context (top) and 2-Stimulus (bottom) tasks, as a function of input conditions. (**D**) Left, mean error (across 50 trials) for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) (n = 10 simulations, two-way ANOVA with mixed-effect design, $F_{1,18} = 32.48$, $P < 0.0001$). Right, mean standard deviation of the time of threshold-crossing across all correct trials for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) ($F_{1,18} = 128.50$, $P < 0.0001$). Data are presented as mean ± SEM.
(TIF)

S5 Fig. Full initial connectivity of the weight matrix does not alter the generalization and robustness to noise differences between the 2-Context and 2-Stimulus tasks. (**A**) Plots of the mean crossing time for each RNN across input conditions for the 2-Context (top) and 2-Stimulus (bottom) tasks. Insets, examples of the sigmoid-function fits for a single RNN (black). (**B**) **Left**, mean slope of the sigmoid fits for 2-Stimulus task is significantly higher than that for the 2-Context task ($n = 20$ simulations for each, two-sided t test, $t_{18} = 4.35$, $P = 0.00039$). **Right**, correlation coefficient between mean crossing times and input conditions for 2-Context task is significantly higher than that for the 2-Stimulus task ($n = 20$ simulations for each, two-sided t test on Fisher-transformed values, $t_{18} = 6.48$, $P < 0.0001$). (**C**) Standard deviations of the crossing times for each RNN in the 2-Context (top) and 2-Stimulus (bottom) tasks, as a function of input conditions. (**D**) Left, mean error (across 50 trials) for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) (n = 10 simulations, two-way ANOVA with mixed-effect design, $F_{1,18} = 5.78$, $P = 0.027$). Right, mean standard deviation of the time of threshold-crossing across all correct trials for 2-Context task (cyan) is higher than that for 2-Stimulus task (orange) ($F_{1,18} = 86.03$, $P < 0.0001$). Data are presented as mean ± SEM.
(TIF)

S6 Fig. Differential functional effects of deleting specific classes of units. (**A**) Schematic of the deletion experiments. To delete a given unit denoted by the red arrow (bottom), all in and out weights of the recurrent weight matrix of that units were set to zero. (**B**) Performance of RNNs trained on the 2-Context task after progressively deleting units from specific temporal classes: stimulus-specific, scaling, and absolute temporal classes for both excitatory (left) and inhibitory (right) units. For each data point, units were randomly selected 10 times, and 10 test trials were obtained. A three-way ANOVA revealed highly significant effects of main temporal-class ($F_{2,619} = 31$, $P < 10^{-12}$) and Ex-Inh ($F_{2,619} = 390$, $P < 10^{-66}$) factors. Additionally, there was a highly significant interaction between temporal-class and Ex-Inh class ($F_{2,619} = 27$, $P < 10^{-10}$) and multi-comparison analyses showed that performance for inhibitory scaling cells was significantly lower than all other 5 deletion manipulations (P < 0.0001 for all comparisons). (**C**) Similar to (**B**) but for error. As in (**B**), there were highly significant main effects ($F_{2,619} = 34$, $P < 10^{-14}$, and $F_{2,619} = 118$, $P < 10^{-24}$, for temporal-class and Ex-Inh, respectively), as well as a significant interaction between temporal-class and Ex-Inh ($F_{2,619} = 46$, $P < 10^{-18}$). And again the inhibitory scaling cells increased the error more than all other deletion manipulations ($P < 0.0001$ for all comparisons). (**D-E**) There were no main effects of temporal-class or Ex-Inh that were consistently significant for both the performance and error measure. The interaction between temporal-class and Ex-Inh was either trending ($F_{2,619} = 2.5$, $P = 0.08$) or mildly significant ($F_{2,619} = 3.6$, $P = 0.027$) for the performance and error analyses, respectively. Data are presented as performance mean ± SEM across 20 RNNs. Notice that the performance of stimulus-specific units in (**D**) and (**E**) (magenta) are very similar to, and mostly obscured by the absolute traces (red). (**F**) Mean output traces across 20 simulations when deleting 6 excitatory (left) and inhibitory (right) units of the three types: stimulus-

specific, scaling, and absolute for 2-Context task. (**G**) Same as **F** but for 2-Stimulus task.
(TIF)

S7 Fig. Differential subspace dynamics for RNNs trained on 2-Context and 2-Stimulus tasks. (**A**) For the 2-Context task, recurrent unit dynamics for the short (blue) and long (green) intervals were projected into the first three PC spaces. Asterisks denote the onset of inputs (t = 0), arrows denote the corresponding weights vectors (Input_{Go}, black; $\text{Input}_{\text{Context}}$, cyan; and Output, red) projected onto the same PC space. Color dots denote the 250 ms intervals along each trajectory. Inset, schematic of angles between segments of the approximate RNN trajectory (orange) and the three weight vectors. These vectors were used to compute the pairwise angles to the Input_{Go}, $\text{Input}_{\text{Context}}$ and Output vectors. (**B**) Similar to **(A)** but for 2-Stimulus task, but here the two input vectors represented the $\text{Input}_{\text{Short}}$ (blue) and $\text{Input}_{\text{Long}}$ (green) weight vectors. (**C**) Same number of PCs explained more variance for 2-Context task than that for 2-Stimulus task (Two-way ANOVA, $F_{(1,\ 38)} = 255.6$ and $P < 0.0001$). (**D**) Average pairwise angles between segments of short (top)/long (bottom) dynamics and inputs/output vectors as in **(A)** for 2-Context task (20 simulations, data presents as Mean ± SEM). Shaded area donted the duration of the transient Input_{Go} **(E)** Same as in **(D)** but for 2-Stimulus task. The shaded area denotes the duration of the transient $\text{Input}_{\text{Short}}$ and $\text{Input}_{\text{Long}}$.
(TIF)

S8 Fig. PCA plots of the recurrent dynamics for generalization to novel intervals. (**A**) Recurrent dynamics corresponding to different context levels (denoted by the color) as in Fig 2 were projected into the first three PCs in 20 RNNS trained on 2-Context task. The arrows denoted the directions of Input_{Go} (black), $\text{Input}_{\text{Context}}$ (cyan), and Output (red) weights projected into the same PC space. (**B**) similar as in **(A)** but for the 2-Stimulus task.
(TIF)

Author Contributions

Conceptualization: Shanglin Zhou, Sotiris C. Masmanidis, Dean V. Buonomano.

Formal analysis: Shanglin Zhou, Dean V. Buonomano.

Funding acquisition: Sotiris C. Masmanidis, Dean V. Buonomano.

Methodology: Shanglin Zhou, Dean V. Buonomano.

Supervision: Sotiris C. Masmanidis, Dean V. Buonomano.

Visualization: Shanglin Zhou.

Writing – original draft: Shanglin Zhou, Sotiris C. Masmanidis, Dean V. Buonomano.

Writing – review & editing: Shanglin Zhou, Dean V. Buonomano.

References

1. Paton JJ, Buonomano DV. The Neural Basis of Timing: Distributed Mechanisms for Diverse Functions. Neuron. 2018; 98(4):687–705. https://doi.org/10.1016/j.neuron.2018.03.045. PMID: 29772201; PubMed Central PMCID: PMC5962026.
2. Buhusi CV, Meck WH. What makes us tick? Functional and neural mechanisms of interval timing. Nat Rev Neurosci. 2005; 6(10):755–65. https://doi.org/10.1038/nrn1764 PMID: 16163383.
3. Meck WH, Ivry RB. Editorial overview: Time in perception and action. Current Opinion in Behavioral Sciences. 2016; 8:vi–x. https://doi.org/10.1016/j.cobeha.2016.03.001.
4. Cannon JJ, Patel AD. How Beat Perception Co-opts Motor Neurophysiology. Trends in Cognitive Sciences. 2021; 25(2):137–50. https://doi.org/10.1016/j.tics.2020.11.002 PMID: 33353800

5. Coull JT, Nobre AC. Where and When to Pay Attention: The Neural Systems for Directing Attention to Spatial Locations and to Time Intervals as Revealed by Both PET and fMRI. The Journal of Neuroscience. 1998; 18(18):7426–35. https://doi.org/10.1523/JNEUROSCI.18-18-07426.1998 PMID: 9736662

6. Merchant H, Harrington DL, Meck WH. Neural Basis of the Perception and Estimation of Time. Annual Review of Neuroscience. 2013; 36(1):313–36. https://doi.org/10.1146/annurev-neuro-062012-170349 PMID: 23725000.

7. Issa JB, Tocker G, Hasselmo ME, Heys JG, Dombeck DA. Navigating Through Time: A Spatial Navigation Perspective on How the Brain May Encode Time. Annual Review of Neuroscience. 2020; 43(1): null. https://doi.org/10.1146/annurev-neuro-101419-011117 PMID: 31961765.

8. Coull JT, Cheng R-K, Meck WH. Neuroanatomical and Neurochemical Substrates of Timing. Neuropsychopharmacology. 2011; 36(1):3–25. https://doi.org/10.1038/npp.2010.113 PMID: 20668434

9. Buonomano DV, Karmarkar UR. How do we tell time? Neuroscientist. 2002; 8(1):42–51. https://doi.org/10.1177/107385840200800109 PMID: 11843098.

10. Fung BJ, Sutlief E, Hussain Shuler MG. Dopamine and the interdependency of time perception and reward. Neuroscience & Biobehavioral Reviews. 2021; 125:380–91. https://doi.org/10.1016/j.neubiorev.2021.02.030 PMID: 33652021

11. Buonomano DV, Laje R. Population clocks: motor timing with neural dynamics. Trends in Cognitive Sciences. 2010; 14(12):520–7. https://doi.org/10.1016/j.tics.2010.09.002 PMID: 20889368

12. Hardy NF, Goudar V, Romero-Sosa JL, Buonomano DV. A model of temporal scaling correctly predicts that motor timing improves with speed. Nature Communications. 2018; 9(1):4732. https://doi.org/10.1038/s41467-018-07161-6 PubMed Central PMCID: PMC6226482. PMID: 30413692

13. Slayton MA, Romero-Sosa JL, Shore K, Buonomano DV, Viskontas IV. Musical expertise generalizes to superior temporal scaling in a Morse code tapping task. PLOS ONE. 2020; 15(1):e0221000. https://doi.org/10.1371/journal.pone.0221000 PMID: 31905200

14. Wang J, Narain D, Hosseini EA, Jazayeri M. Flexible timing by temporal scaling of cortical responses. Nature Neuroscience. 2018; 21(1):102–10. https://doi.org/10.1038/s41593-017-0028-6 PMID: 29203897

15. Lerner Y, Honey CJ, Katkov M, Hasson U. Temporal scaling of neural responses to compressed and dilated natural speech. Journal of Neurophysiology. 2014; 111(12):2433–44. https://doi.org/10.1152/jn.00497.2013 PMID: 24647432

16. Mello GBM, Soares S, Paton JJ. A scalable population code for time in the striatum. Curr Biol. 2015; 9:1113–22. https://doi.org/10.1016/j.cub.2015.02.036 PMID: 25913405

17. Shimbo A, Izawa E-I, Fujisawa S. Scalable representation of time in the hippocampus. Science Advances. 2021; 7(6):eabd7013. https://doi.org/10.1126/sciadv.abd7013 PMID: 33536211

18. Kunimatsu J, Suzuki TW, Ohmae S, Tanaka M. Different contributions of preparatory activity in the basal ganglia and cerebellum for self-timing. Elife. 2018; 7. https://doi.org/10.7554/eLife.35676 PMID: 29963985; PubMed Central PMCID: PMC6050043.

19. Aasland WA, Baum SR. Temporal parameters as cues to phrasal boundaries: A comparison of processing by left- and right-hemisphere brain-damaged individuals. Brain and Language. 2003; 87 (3):385–99. https://doi.org/10.1016/s0093-934x(03)00138-x PMID: 14642541

20. Schwab S, Miller JL, Grosjean F, Mondini M. Effect of Speaking Rate on the Identification of Word Boundaries. Phonetica. 2008; 65(3):173–86. https://doi.org/10.1159/000144078 PMID: 18679044

21. Kim J, Ghim J-W, Lee JH, Jung MW. Neural Correlates of Interval Timing in Rodent Prefrontal Cortex. The Journal of Neuroscience. 2013; 33(34):13834–47. https://doi.org/10.1523/JNEUROSCI.1443-13.2013 PMID: 23966703

22. Allan LG, Gibbon J. Human bisection at the geometric mean. Learning and Motivation. 1991; 22(1–2):39–58.

23. Zhou S, Masmanidis SC, Buonomano DV. Neural Sequences as an Optimal Dynamical Regime for the Readout of Time. Neuron. 2020; 108(4):651–8.e5. https://doi.org/10.1016/j.neuron.2020.08.020 PMID: 32946745

24. Leon MI, Shadlen MN. Representation of time by neurons in the posterior parietal cortex of the macaque. Neuron. 2003; 38:317–27. https://doi.org/10.1016/s0896-6273(03)00185-5 PMID: 12718864

25. Gouvea TS, Monteiro T, Motiwala A, Soares S, Machens C, Paton JJ. Striatal dynamics explain duration judgments. Elife. 2015; 4. https://doi.org/10.7554/eLife.11386 PMID: 26641377; PubMed Central PMCID: PMC4721960.

26. Emmons EB, De Corte BJ, Kim Y, Parker KL, Matell MS, Narayanan NS. Rodent Medial Frontal Control of Temporal Processing in the Dorsomedial Striatum. The Journal of Neuroscience. 2017; 37 (36):8718–33. https://doi.org/10.1523/JNEUROSCI.1376-17.2017 PMID: 28821670

27. Merchant H, Zarco W, Pérez O, Prado L, Bartolo R. Measuring time with different neural chronometers during a synchronization-continuation task. Proceedings of the National Academy of Sciences. 2011; 108(49):19784–9. https://doi.org/10.1073/pnas.1112933108 PMID: 22106292

28. Crowe DA, Zarco W, Bartolo R, Merchant H. Dynamic Representation of the Temporal and Sequential Structure of Rhythmic Movements in the Primate Medial Premotor Cortex. The Journal of Neuroscience. 2014; 34(36):11972–83. https://doi.org/10.1523/JNEUROSCI.2177-14.2014 PMID: 25186744

29. Kraus BJ, Robinson RJ, White JA, Eichenbaum H, Hasselmo ME. Hippocampal "Time Cells": Time versus Path Integration. Neuron. 2013; 78(6):1090–101. https://doi.org/10.1016/j.neuron.2013.04.015 PMID: 23707613; PubMed Central PMCID: PMC3913731.

30. Remington ED, Narain D, Hosseini EA, Jazayeri M. Flexible Sensorimotor Computations through Rapid Reconfiguration of Cortical Dynamics. Neuron. 2018; 98(5):1005–19.e5. https://doi.org/10.1016/j.neuron.2018.05.020 PMID: 29879384; PubMed Central PMCID: PMC6009852.

31. Goudar V, Buonomano DV. Encoding sensory and motor patterns as time-invariant trajectories in recurrent neural networks. eLife. 2018; 7:e31134. https://doi.org/10.7554/eLife.31134 PMID: 29537963; PubMed Central PMCID: PMC5851701.

32. Huk AC, Hart E. Parsing signal and noise in the brain. Science. 2019; 364(6437):236–7. https://doi.org/10.1126/science.aax1512 PMID: 31000652

33. Liu Y, Tiganj Z, Hasselmo ME, Howard MW. A neural microcircuit model for a scalable scale-invariant representation of time. Hippocampus. 2019; 29(3):260–74. https://doi.org/10.1002/hipo.22994 PMID: 30421473

34. Shankar KH, Howard MW. A Scale-Invariant Internal Representation of Time. Neural Computation. 2012; 24(1):134–93. https://doi.org/10.1162/NECO_a_00212 PMID: 21919782

35. Narayanan NS. Ramping activity is a cortical mechanism of temporal control of action. Current Opinion in Behavioral Sciences. 2016; 8:226–30. https://doi.org/10.1016/j.cobeha.2016.02.017 PMID: 27171843

36. Durstewitz D. Self-organizing neural integrator predicts interval times through climbing activity. J Neurosci. 2003; 23(12):5342–53. https://doi.org/10.1523/JNEUROSCI.23-12-05342.2003 PMID: 12832560.

37. Simen P, Balci F, de Souza L, Cohen JD, Holmes P. A model of interval timing by neural integration. J Neurosci. 2011; 31(25):9238–53. Epub 2011/06/24. https://doi.org/10.1523/JNEUROSCI.3121-10.2011 PMID: 21697374; PubMed Central PMCID: PMC3142662.

38. Jazayeri M, Shadlen Michael N. A Neural Mechanism for Sensing and Reproducing a Time Interval. Current Biology. 2015; 25(20):2599–609. https://doi.org/10.1016/j.cub.2015.08.038 PMID: 26455307

39. Simen P, Vlasov K, Papadakis S. Scale (In)Variance in a Unified Diffusion Model of Decision Making and Timing. Psychological Review. 2016; 123(2):151–81. https://doi.org/10.1037/rev0000014 WOS:264619571900002.

40. Finkelstein A, Fontolan L, Economo MN, Li N, Romani S, Svoboda K. Attractor dynamics gate cortical information flow during decision-making. Nat Neurosci. 2021; 24(6):843–50. https://doi.org/10.1038/s41593-021-00840-6 PMID: 33875892.

41. Buonomano DV, Mauk MD. Neural network model of the cerebellum: temporal discrimination and the timing of motor responses. Neural Comput. 1994; 6:38–55.

42. Medina JF, Garcia KS, Nores WL, Taylor NM, Mauk MD. Timing mechanisms in the cerebellum: testing predictions of a large-scale computer simulation. J Neurosci. 2000; 20(14):5516–25. https://doi.org/10.1523/JNEUROSCI.20-14-05516.2000 PMID: 10884335.

43. Lynch Galen F, Okubo Tatsuo S, Hanuschkin A, Hahnloser Richard HR, Fee Michale S. Rhythmic Continuous-Time Coding in the Songbird Analog of Vocal Motor Cortex. Neuron. 2016; 90(4):877–92. https://doi.org/10.1016/j.neuron.2016.04.021 PMID: 27196977

44. Hahnloser RHR, Kozhevnikov AA, Fee MS. An ultra-sparse code underlies the generation of neural sequence in a songbird. Nature. 2002; 419:65–70. https://doi.org/10.1038/nature00974 PMID: 12214232

45. Long MA, Fee MS. Using temperature to analyse temporal dynamics in the songbird motor pathway. Nature. 2008; 456(7219):189–94. https://doi.org/10.1038/nature07448 PMID: 19005546

46. Picardo Michel A, Merel J, Katlowitz Kalman A, Vallentin D, Okobi Daniel E, Benezra Sam E, et al. Population-Level Representation of a Temporal Sequence Underlying Song Production in the Zebra Finch. Neuron. 2016; 90(4):866–76. https://doi.org/10.1016/j.neuron.2016.02.016 PMID: 27196976

47. Pastalkova E, Itskov V, Amarasingham A, Buzsaki G. Internally Generated Cell Assembly Sequences in the Rat Hippocampus. Science. 2008; 321(5894):1322–7. https://doi.org/10.1126/science.1159775 PMID: 18772431; PubMed Central PMCID: PMC2570043.

48. Carnevale F, de Lafuente V, Romo R, Barak O, Parga N. Dynamic Control of Response Criterion in Premotor Cortex during Perceptual Detection under Temporal Uncertainty. Neuron. 2015; 86(4):1067–77. https://doi.org/10.1016/j.neuron.2015.04.014 PubMed Central PMCID: PMC5077164. PMID: 25959731

49. Bakhurin KI, Goudar V, Shobe JL, Claar LD, Buonomano DV, Masmanidis SC. Differential Encoding of Time by Prefrontal and Striatal Network Dynamics. The Journal of Neuroscience. 2017; 37(4):854–70. https://doi.org/10.1523/JNEUROSCI.1789-16.2016 PMID: 28123021; PubMed Central PMCID: PMC5296780.

50. Stokes MG, Kusunoki M, Sigala N, Nili H, Gaffan D, Duncan J. Dynamic Coding for Cognitive Control in Prefrontal Cortex. Neuron. 2013; 78(2):364–75. https://doi.org/10.1016/j.neuron.2013.01.039 PMID: 23562541; PubMed Central PMCID: PMC3898895.

51. MacDonald Christopher J, Lepage Kyle Q, Eden Uri T, Eichenbaum H. Hippocampal "Time Cells" Bridge the Gap in Memory for Discontiguous Events. Neuron. 2011; 71(4):737–49. https://doi.org/10.1016/j.neuron.2011.07.012 PMID: 21867888

52. Cone I, Shouval HZ. Learning precise spatiotemporal sequences via biophysically realistic learning rules in a modular, spiking network. eLife. 2021; 10:e63751. https://doi.org/10.7554/eLife.63751 PMID: 33734085

53. Maes A, Barahona M, Clopath C. Learning spatiotemporal signals using a recurrent spiking network that discretizes time. PLOS Computational Biology. 2020; 16(1):e1007606. https://doi.org/10.1371/journal.pcbi.1007606 PMID: 31961853

54. Hardy NF, Buonomano DV. Encoding Time in Feedforward Trajectories of a Recurrent Neural Network Model. Neural Comput. 2018; 30(2):378–96. https://doi.org/10.1162/neco_a_01041 PMID: 29162002; PubMed Central PMCID: PMC5873300.

55. Tupikov Y, Jin DZ. Addition of new neurons and the emergence of a local neural circuit for precise timing. PLOS Computational Biology. 2021; 17(3):e1008824. https://doi.org/10.1371/journal.pcbi.1008824 PMID: 33730085

56. Gámez J, Mendoza G, Prado L, Betancourt A, Merchant H. The amplitude in periodic neural state trajectories underlies the tempo of rhythmic tapping. PLOS Biology. 2019; 17(4):e3000054. https://doi.org/10.1371/journal.pbio.3000054 PMID: 30958818

57. Merchant H, Pérez O, Bartolo R, Méndez JC, Mendoza G, Gámez J, et al. Sensorimotor neural dynamics during isochronous tapping in the medial premotor cortex of the macaque. European Journal of Neuroscience. 2015; 41(5):586–602. https://doi.org/10.1111/ejn.12811 PMID: 25728178

58. Taxidis J, Pnevmatikakis EA, Dorian CC, Mylavarapu AL, Arora JS, Samadian KD, et al. Differential Emergence and Stability of Sensory and Temporal Representations in Context-Specific Hippocampal Sequences. Neuron. 2020. https://doi.org/10.1016/j.neuron.2020.08.028 PMID: 32949502

59. Murakami M, Vicente MI, Costa GM, Mainen ZF. Neural antecedents of self-initiated actions in secondary motor cortex. Nat Neurosci. 2014; 17(11):1574–82. https://doi.org/10.1038/nn.3826 PMID: 25262496

60. Jin DZ, Fujii N, Graybiel AM. Neural representation of time in cortico-basal ganglia circuits. Proc Natl Acad Sci U S A. 2009; 106(45):19156–61. https://doi.org/10.1073/pnas.0909881106 PMID: 19850874; PubMed Central PMCID: PMC2776432.

61. Mendoza G, Méndez JC, Pérez O, Prado L, Merchant H. Neural basis for categorical boundaries in the primate pre-SMA during relative categorization of time intervals. Nature Communications. 2018; 9 (1):1098. https://doi.org/10.1038/s41467-018-03482-8 PMID: 29545587; PubMed Central PMCID: PMC5854627.

62. Duysens J, Schaafsma SJ, Orban GA. Cortical Off Response Tuning for Stimulus Duration. Vision Res. 1996; 36(20):3243–51. https://doi.org/10.1016/0042-6989(96)00040-5 PMID: 8944284

63. Chubykin AA, Roach EB, Bear MF, Shuler MGH. A Cholinergic Mechanism for Reward Timing within Primary Visual Cortex. Neuron. 2013; 77(4):723–35. https://doi.org/10.1016/j.neuron.2012.12.039 PMID: 23439124; PubMed Central PMCID: PMC3597441.

64. Merchant H, Pérez O, Zarco W, Gámez J. Interval Tuning in the Primate Medial Premotor Cortex as a General Timing Mechanism. The Journal of Neuroscience. 2013; 33(21):9082–96. https://doi.org/10.1523/JNEUROSCI.5513-12.2013 PMID: 23699519

65. Buonomano DV, Maass W. State-dependent Computations: Spatiotemporal Processing in Cortical Networks. Nat Rev Neurosci. 2009; 10:113–25. https://doi.org/10.1038/nrn2558 PMID: 19145235; PubMed Central PMCID: PMC2676350.

66. Rigotti M, Barak O, Warden MR, Wang X-J, Daw ND, Miller EK, et al. The importance of mixed selectivity in complex cognitive tasks. Nature. 2013; 497(7451):585–90. https://doi.org/10.1038/nature12160 PMID: 23685452

67. DePasquale B, Cueva CJ, Rajan K, Escola GS, Abbott LF. full-FORCE: A target-based method for training recurrent networks. PLOS ONE. 2018; 13(2):e0191527. https://doi.org/10.1371/journal.pone.0191527 PMID: 29415041

68. Murray JM, Escola GS. Learning multiple variable-speed sequences in striatum via cortical tutoring. eLife. 2017; 6:e26084. https://doi.org/10.7554/eLife.26084 PMID: 28481200

69. Jazayeri M, Afraz A. Navigating the Neural Space in Search of the Neural Code. Neuron. 2017; 93 (5):1003–14. https://doi.org/10.1016/j.neuron.2017.02.019 PMID: 28279349

70. Laje R, Buonomano DV. Robust timing and motor patterns by taming chaos in recurrent neural networks. Nat Neurosci. 2013; 16(7):925–33. https://doi.org/10.1038/nn.3405 PMID: 23708144; PubMed Central PMCID: PMC3753043.

71. Monteforte M, Wolf F. Dynamic Flux Tubes Form Reservoirs of Stability in Neuronal Circuits. Physical Review X. 2012; 2(4):041007.

72. Chaisangmongkon W, Swaminathan SK, Freedman DJ, Wang XJ. Computing by Robust Transience: How the Fronto-Parietal Network Performs Sequential, Category-Based Decisions. Neuron. 2017; 93 (6):1504–17 e4. https://doi.org/10.1016/j.neuron.2017.03.002 PMID: 28334612; PubMed Central PMCID: PMC5586485.

73. Rabinovich MI, Simmons AN, Varona P. Dynamical bridge between brain and mind. Trends Cogn Sci. 2015; 19(8):453–61. Epub 2015/07/08. https://doi.org/10.1016/j.tics.2015.06.005 PMID: 26149511.

74. Afraimovich V, Zhigulin V, Rabinovich M. On the origin of reproducible sequential activity in neural circuits. Chaos: An Interdisciplinary Journal of Nonlinear Science. 2004; 14:1123–9. https://doi.org/10.1063/1.1819625 PMID: 15568926

75. Rajan K, Abbott LF, Sompolinsky H. Stimulus-dependent suppression of chaos in recurrent neural networks. Physical Rev E. 2010; 82:011903(5). https://doi.org/10.1103/PhysRevE.82.011903 PMID: 20866644

76. Vogels TP, Rajan K, Abbott LF. Neural network dynamics. Annu Rev Neurosci. 2005; 28:357–76. https://doi.org/10.1146/annurev.neuro.28.061604.135637 PMID: 16022600.

77. Kim R, Sejnowski TJ. Strong inhibitory signaling underlies stable temporal dynamics and working memory in spiking neural networks. Nature Neuroscience. 2021; 24(1):129–39. https://doi.org/10.1038/s41593-020-00753-w PMID: 33288909

78. Kim R, Li Y, Sejnowski TJ. Simple framework for constructing functional spiking recurrent neural networks. Proc Natl Acad Sci U S A. 2019; 116(45):22811–20. https://doi.org/10.1073/pnas.1905926116 PMID: 31636215; PubMed Central PMCID: PMC6842655.

79. Song HF, Yang GR, Wang XJ. Training Excitatory-Inhibitory Recurrent Neural Networks for Cognitive Tasks: A Simple and Flexible Framework. PLoS Comput Biol. 2016; 12(2):e1004792. https://doi.org/10.1371/journal.pcbi.1004792 PMID: 26928718; PubMed Central PMCID: PMC4771709.