nature neuroscience

SUPPLEMENTARY INFORMATION

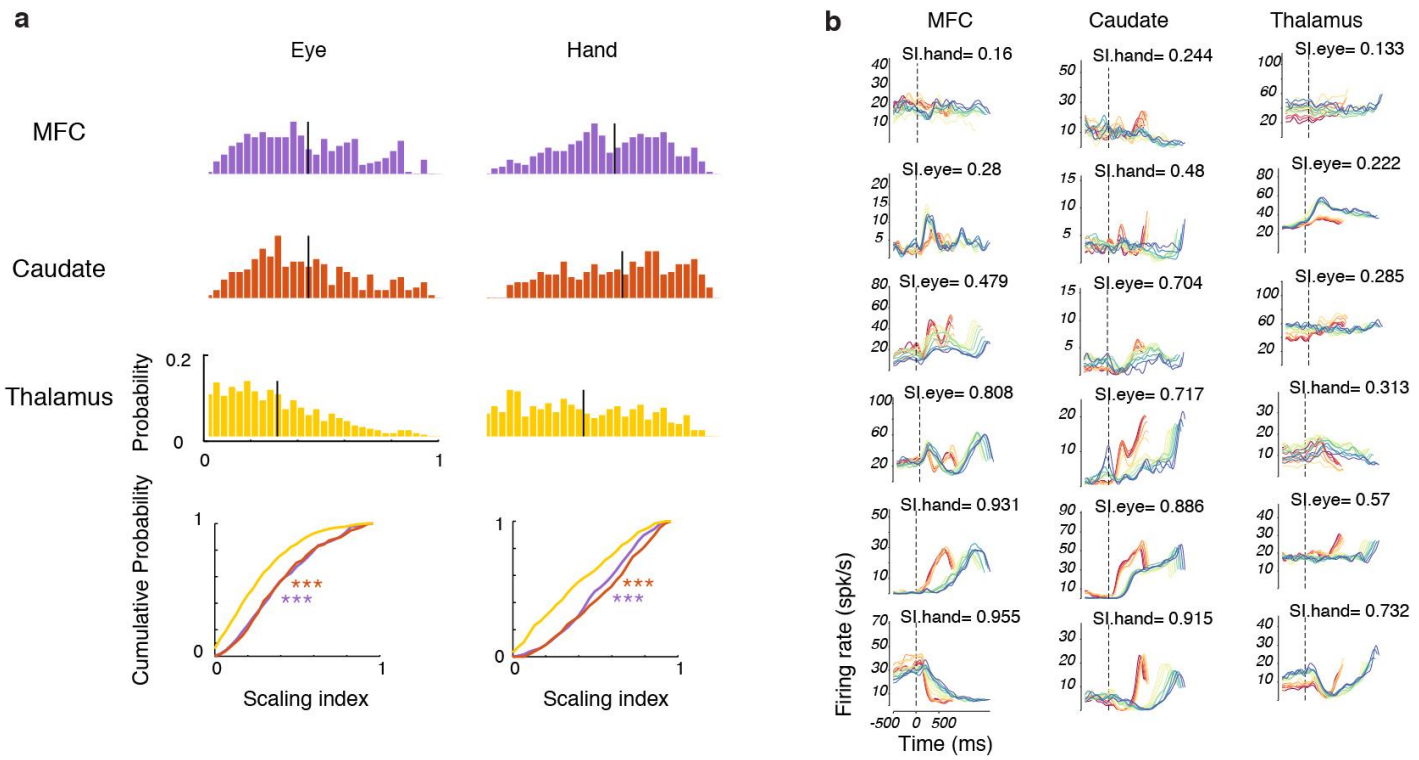
https://doi.org/10.1038/s41593-017-0028-6

In the format provided by the authors and unedited.

Flexible timing by temporal scaling of cortical responses

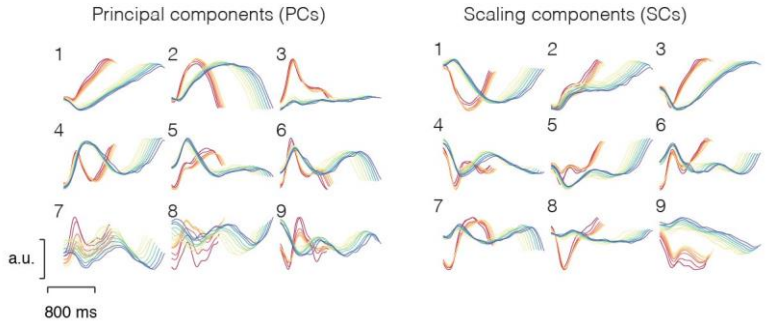
Jing Wang[1,5], Devika Narain[1,2,3,4], Eghbal A. Hosseini[2,5] and Mehrdad Jazayeri[1,2,5]*

[1]McGovern Institute for Brain Research, Massachusetts Institute of Technology, Cambridge, MA, USA. [2]Department of Brain & Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA, USA. [3]Netherlands Institute for Neuroscience, Amsterdam, The Netherlands. [4]Department of Neuroscience, Erasmus Medical Center, Rotterdam, The Netherlands. Present address: [5]Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA, USA. Jing Wang and Devika Narain contributed equally to this research. *e-mail: mjaz@mit.edu



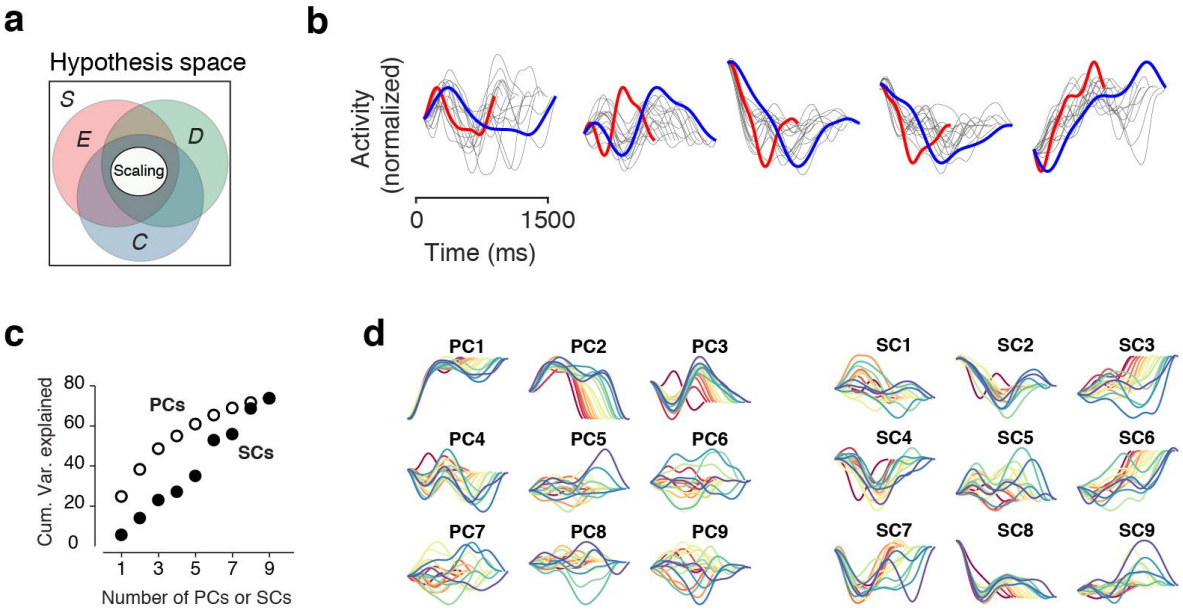
Supplementary Figure 1

Scaling index (SI) of single neurons in different brain areas.
(a) Histograms on the top show the normalized distribution of the scaling index for individual neurons in the MFC (n = 416 neurons from both animals), caudate (n = 278 neurons), and thalamus (n = 846 neurons) for the *Eye* (Left) and *Hand* conditions (Right). The bottom panel shows a comparison of the cumulative probability distribution of scaling index across the three areas. The thalamus shows a predominance of smaller scaling index value (Mann-Whitney-Wilcoxon test, one-tailed, $W(1260)$ = 310,733, z = 7.89, *** $P < .001$ in comparison with MFC, and $W(1122)$ = 189,163, z = 6.98, *** $P < .001$ in comparison with caudate) (b) Example PSTHs covering a range of SIs in the three brain areas. The SI value and effector condition for each neuron is indicated.



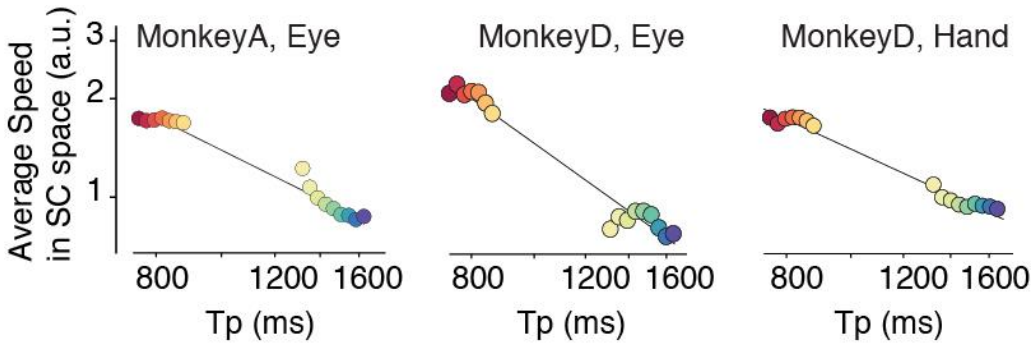
Supplementary Figure 2

The time course of PCs (left) and SCs (right) for MFC data.
First 9 PCs over the course of the production interval (abscissa) that explain 80% of variance in the MFC data (n = 281 neurons for Monkey A) in decreasing order of variance explained (left) for Short (warm colors) and Long (cool colors) intervals. First 9 SCs, obtained for the same data, in decreasing order of scaling (right, see Methods).



Supplementary Figure 3

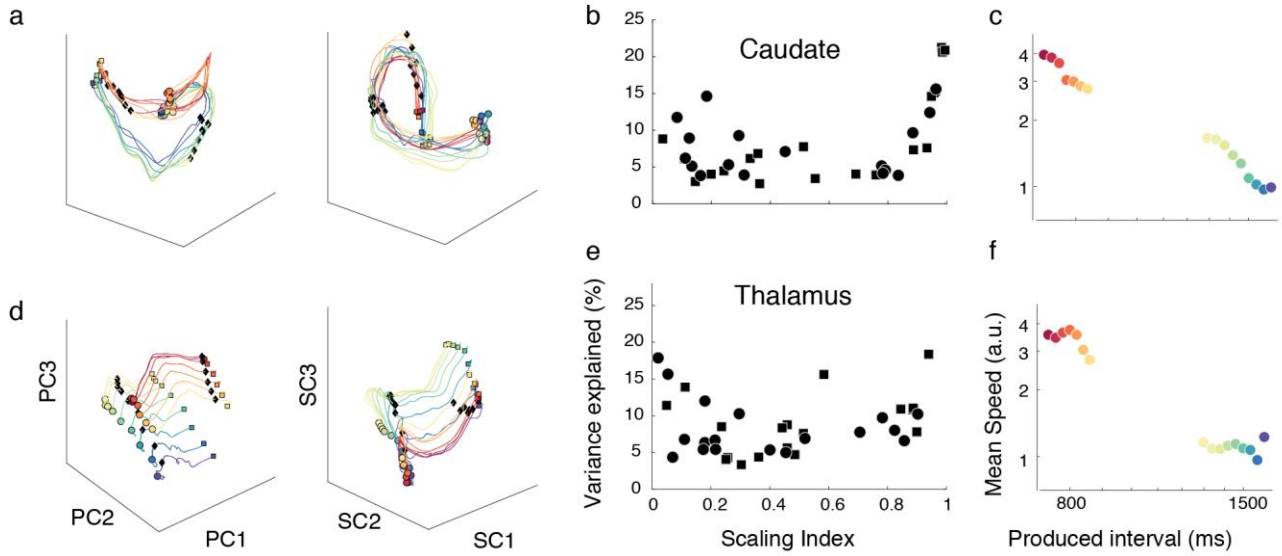
Analysis of scaling with surrogate data.
(a) Venn diagram showing the various constraints considered for non-scaling models. All surrogate data was generated from a Gaussian Process (GP) with the same level of temporal smoothness (white rectangle, *S*) as the data. We considered three additional constraints to make the surrogate data more similar to neural data without an explicit requirement for scaling. One constraint required responses for all production intervals to be at the same level at the time of Set and at the time of response. We refer to this constraint as endpoint matching (red circle, *E*). Another constraint required that the dimensionality of the surrogate data match the neural data, and additionally the variance explained by each principal component (PC) be matched. We refer to this constraint as dimensionality matching (green circle, *D*). Finally, we considered a constraint that required the collection of responses for different production intervals to have the same correlation (quantified as R^2) as expected from perfect scaling. We refer to this constraint as correlation matching (blue circle, *C*). We created surrogate data for each constraint and for various combination of constraints, and compared the scaling properties to the original data. Note that each constraint characterized a superset of the scaling hypothesis. **(b)** Example traces showing the procedure for generating the surrogate data (n = 281 surrogate units) in the *C+D+E+S* model for 5 randomly selected surrogate units aligned to the time of Set. We first sampled a *Short* trace (red) from a Gaussian process. The trace in blue corresponds to the perfectly scaled version of the red trace and is not a sample from the surrogate model. The surrogate data were generated using a constrained Gaussian Process (GP) prior as follows: the response for the shortest production intervals (red) was sampled from a GP with the same level of temporal smoothness as the neural data. The corresponding response with perfect scaling was generated by linear scaling (shown in blue). Note that the trace in blue is not a sample from the GP and is therefore, not part of the surrogate data. The gray traces correspond to the surrogate data. To generate the surrogate data, we drew samples from the Gaussian process that satisfied several criteria. First, the starting point as well as the ending point of every gray trace had to be perfectly matched to the starting point and ending point of the perfectly scaled blue trace. Second, across the population of surrogate data, the dimensionality had to match observed neural data. Finally, the correlation between every gray trace and the red trace was the same as the correlation between the red and blue trace. In this way, every sample of GP (gray traces) matched the smoothness, endpoints, dimensionality and correlation as the real data (i.e., *C+D+E+S* model). **(c)** Cumulative percentage variance explained by PCs and SCs for the surrogate data generated from the non-scaling *C+D+E+S* model. **(d)** The first 9 principal components of population activity (PC, left) and the corresponding 9 scaling components (right, SCs) plotted as the function of time from Set for the non-scaling *C+D+E+S* model. Note that PCs and SCs are based on the surrogate data (gray traces in panel b) – not the perfectly scaled data (blue traces in panel b).



Supplementary Figure 4

Relationship between MFC population activity and behavior

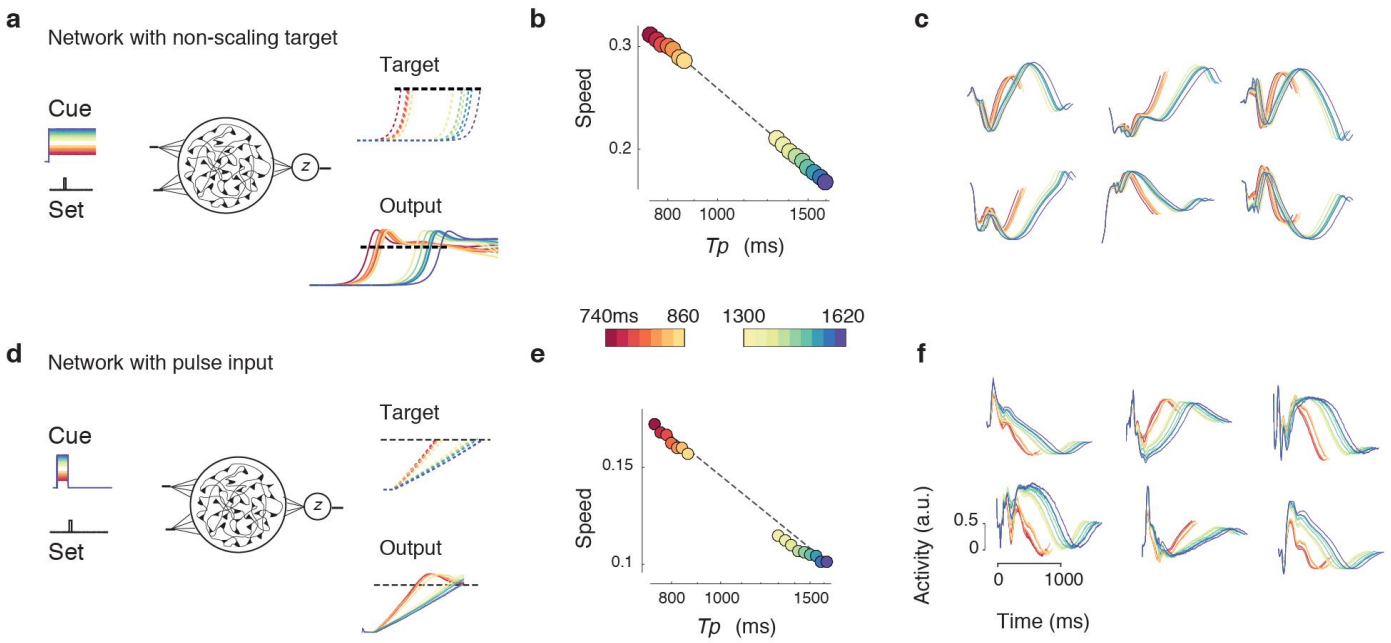
The speed of neural trajectory in MFC (n = 281 neurons for Monkey A, n = 135 neurons for Monkey D) within the scaling subspace spanned by the first 3 SCs predicted *Tp* across both *Short* and *Long* conditions on a trial-by-trial basis. Both ordinate and abscissa follow a logarithmic scale. The case of hand trials for Monkey A was shown in Fig. 5d. Here, all the other conditions are shown.



Supplementary Figure 5

Analysis of scaling at the population level in the caudate (top) and thalamus (bottom).

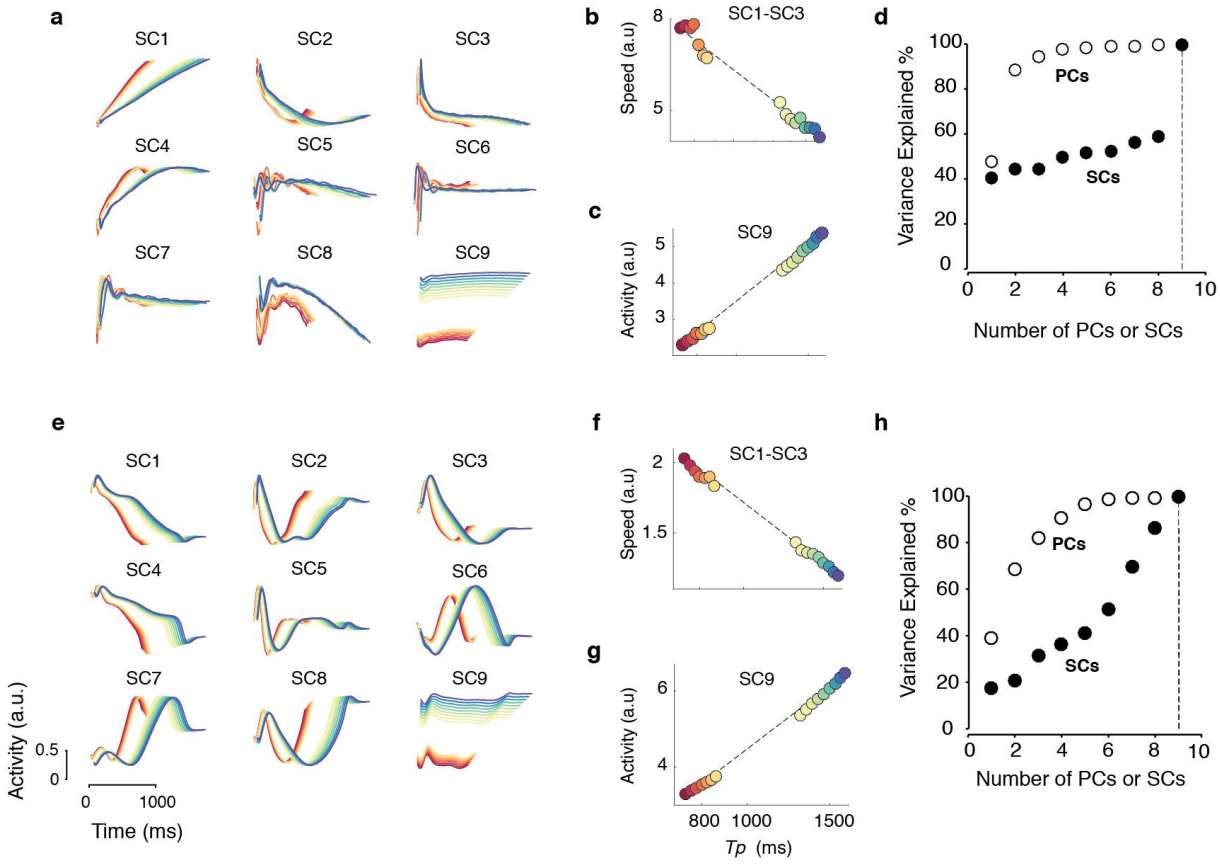
Left column: Population activity (n = 101 neurons for caudate and n = 481 neurons for thalamus, Moneky A) profiles projected onto the first 3 principal components (PCs). Activity profiles associated with different produced intervals for *Short* and *Long* conditions are plotted in different colors (same color scheme used throughout the paper). The state at 700 ms after Set is shown along the trajectories (diamond). Second column from the left: Population activity projected onto the first 3 scaling components (SCs). Activity spanned by the first 3 SCs overlap for different intervals in the caudate but not in the thalamus. Third column from left: Variance explained for individual SCs as a function of scaling index. Right column: The speed of neural trajectory within the scaling subspace spanned by the first 3 SCs. Both ordinate and abscissa follow a logarithmic scale.



Supplementary Figure 6

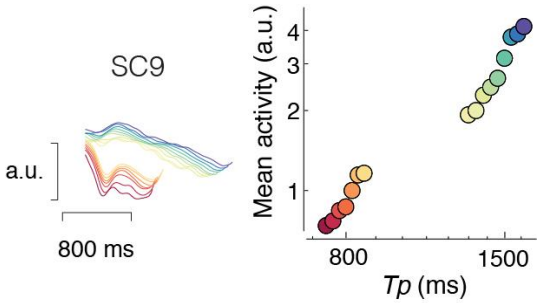
Alternative recurrent neural network models

(a) A recurrent neural network (RNN) trained to use an interval-dependent Cue input to produce time intervals flexibly. The network was trained using a non-scaling exponential objective with a fixed time constant. **(b)** The speed of dynamics measured within the space spanned by the first three PCs predicted *Tp* across both *Short* and *Long* conditions. Both ordinate and abscissa follow a logarithmic scale. **(c)** The response profiles of randomly selected units in the network (a) aligned to the time of Set. **(d)** A RNN trained to use a brief pulse as the interval-dependent Cue input to produce time intervals flexibly. The network was trained using a linear ramping objective like the network in the main text. **(e)** The speed of dynamics predicted *Tp* across both *Short* and *Long* conditions with the same format as b). **(f)** The response profiles of randomly selected units in network (d) aligned to the time of Set.



Supplementary Figure 7

Temporal scaling in the recurrent network mode
(a) The first 9 scaling components (SCs) of the population activity in the RNN with a scaling output function (Fig. 5). The early SCs correspond to the recurrent subspace, and the last SC represents the input subspace. **(b)** The average speed of population activity in the subspace spanned by the first 3 SCs is predictive of both within context and across context variations in *Tp*. Both ordinate and abscissa follow a logarithmic scale. **(c)** The average firing rate of the population activity projected onto the last SC is also predictive of *Tp*. **(d)** Cumulative percentage variance explained by PCs (white) and SCs (black). The dashed vertical line corresponds to the 9th component. **(e-h)** Same as a-d, for a network that was trained for a non-scaling exponential output objective function (Supplementary Fig. 6d).



Supplementary Figure 8

Nonscaling population activity in MFC
Left: The time course of the SC9 (the least scaling component) across conditions. Right: The average firing rate of population activity (n = 281 neurons for Monkey A) projected onto SC9 (left), also the putative input subspace, increases with produced intervals (*Tp*). This is consistent with the hypothesis that the average firing rate in the non-scaling subspace controls speed. Based on the recurrent network model, this subspace likely reflects the input drive to MFC. Both ordinate and abscissa are plotted in a logarithmic scale.

Supplementary Note

Surrogate data for testing the scaling property

Surrogate model 1. We required the firing rates to have heterogeneous and nonlinear profiles while mimicking the same temporal smoothness as our data. To address this constraint, for each surrogate neuron, we sampled the response profiles (i.e., analogous to firing rates) from a zero-mean multinomial Gaussian Process (GP) prior [1,2]. The covariance function between time points t_i and t_j, also known as the kernel function, $K(t_i, t_j)$, was formulated as follows:

$K(t_i, t_j) = \sigma^2 \exp(-\frac{|t_i - t_j|^2}{2\tau^2}), K(t_i, t_j) \in \mathbb{R}^{T \times T}$, with $\sigma^2 = 1$ and $\tau = 150$ ms.

These parameters were chosen to mimic the average smoothness observed in response profiles of neurons. This constraint allowed us to examine whether the scaling indices associated with different regions of interest could be attributed to the the smoothness of firing rate functions. The surrogate model 1 was not worthy of consideration as i did not exhibit any scaling. But the smoothness constraint was used in the subsequent more constrained models.

Surrogate model 2. The first model did not impose any structure among response profiles across the different produced intervals. In model 2, for each surrogate neuron, the starting point of the surrogate data was the same as the starting point of the real data and the endpoint of the surrogate data was the same as the endpoint of the real data, but the starting point and end point of the real data were not necessarily the same. To satisfy this constraint, for each surrogate neuron, we first generated a response profile corresponding to the shortest produced interval, which we will refer to as the primary profile. We then constrained the response profiles associated with all longer produced intervals to match the primary profile at both the starting and terminal points. To do so, we sampled the remaining profiles from a conditional GP where the endpoint values were constrained to be the same as the primary profile. This constraint can be applied using GP regression analysis [3], where we derived the expectation and covariance of $(g_2, \cdots, g_{tp-1})$ given the endpoints (g_1, g_{tp}). This model tested whether our scaling results could be attributed to a general level of similarity between smooth data with matched initial and terminal points.

Surrogate model 3. The first two constraints ensured that the smoothness of the surrogate data and its starting/ending points corresponded to the recorded neuronal profiles. As a third constraint, we required the surrogate data to have the same dimensionality as the physiological data. To satisfy this constraint, we projected the surrogate data onto its principal components (PCs) and adjusted the variance according to the variance explained by PCs in neural data. This can be achieved by multiplying the data in the PC space by a diagonal matrix that scales each axis according to the desired variance. This model tests whether our scaling results could be attributed to data generated from a low-dimensional process with the same smoothness and matching initial and terminal points.

Surrogate model 4. As a final constraint, we required the surrogate data generated for different produced intervals to exhibit the same coefficient of determination (R^2) as the surrogate data generated from perfect scaling. To satisfy this constraint, for each surrogate neuron, we first sampled one instance from the GP for the shortest produced interval; i.e., the primary profile for that neuron. We then stretched the primary profile temporally to generate a set of perfectly scaled response profiles for all other produced intervals. We computed R^2 as a measure of similarity between perfectly scaled responses. We then created samples from GP for other produced intervals that were matched in starting/ending points and had the same R^2 as the perfectly scaled data. As a final step, we applied the same strategy as in surrogate model 3 to match the dimensionality of the surrogate and neural data. This model enabled us to test whether our scaling results could be attributed to the dimensionality of the data, given the same smoothness and similarity in response profiles achieved by matching temporal correlation as well as initial and terminal points.

We will refer to these models by the constraints they impose as follows:

Surrogate model 1 (S)	Smoothness constrained (*S*)
Surrogate model 2 (*E*+*S*)	Smoothness constrained (S) Endpoints constrained (*E*)
Surrogate model 3 (*D*+*E*+*S*)	Smoothness constrained (*S*) Endpoints constrained (*E*) Dimensionality unconstrained (*D*)
Surrogate model 4 (*C*+*D*+*E*+*S*)	Smoothness constrained (*S*)

	Endpoints constrained (*E*) Dimensionality constrained (*D*) Correlation constrained (*C*)

Supplementary Note Table 1. Surrogate models

The same analyses used to assess the neural responses were used to asses scaling in data generated from the surrogate models. This included (1) reducing the dimensionality to the first 9 PCs (which captured 80% of total variance of the physiological data), (2) computing the scaling index for each PC, (3) assessing the relationship between variance explained and scaling index for random projections of activity in the state space, (4) repeating the step for 200 times to obtain a distribution of scaling indices, and (5) comparing the distribution among different surrogate models and regions of interest.

Training of the recurrent neural network model

Multiple networks were trained with inputs that were presented randomly across trials as specified above using backpropagation-through-time [5]. We computed model parameters by minimizing a squared loss function between the network output $z_s(t)$ and a target function $f_s(t)$ The objective least-squares function used during training was :

$$H(\theta) = {}^{1}\!\Big/\!_{|TS|} \sum_{S} \sum_{t} [z_s(t) - f_s(t)]^2$$

The target function $f_s(t)$ was computed for a subset of the trials and was only defined from the onset of the *Set* pulse till the length of the desired interval. For the duration of the *Set* pulse, the target function assumed a value of zero to prevent rapid transient switches in activity before the production stage.

Once training was complete, we generated a new set of randomized test trials, chose a threshold corresponding to the desired output ($z_s = 1$) and computed the times at which the network output crossed that threshold after *Set*. The time of threshold-crossing relative to Set was taken as the network's produced interval (Tp). Network responses were also tested for intervals that interpolated within and extrapolated beyond the trained range. All networks

exhibited good interpolation performance. Networks that exhibited good extrapolation performance within 150 ms beyond trained ranges were used for further analysis (90% of tested networks). In the main text, we report interpolated intervals that matched the monkeys' response variability for both intervals.

We trained and tested multiple networks with different input profiles and different training objectives to check the consistency of the solution (Supplementary Fig. 6). In these networks, the scalar nature of the objective function was controlled by two parameters, α and A, to create a continuum of objective functions between a ramp (linear scaling) and a delta function (non-scaling). For example, with parameters $A = 3$ and $\alpha = 2.8$, the above function approximates a scaled linear ramp of desired duration (t_s).

$$f_s(t) = A(e^{t/\alpha \cdot t_s} - 1)$$

We tested additional networks with explicit non-scaling objectives and also with Cue inputs that did not remain constant throughout the trial. We have reported the results of two such variations in Supplementary Fig. 6, one in which the objective does not scale with interval duration (Supplementary Fig. 6a), and one in which the Cue input was a transient pulse (Supplementary Fig. 6d).

Similar to previous work [6], we analyzed the rate of change of the network state to uncover its underlying dynamics. We computed $\boldsymbol{q}(x) = \frac{1}{2} \mid \boldsymbol{F}(x) \mid^2$ to find the fixed and slow points of the system (corresponding to tolerances of $q = 1e-18$ and $1e-5$ respectively). We analyzed the local dynamics around fixed points by performing an eigendecomposition of the corresponding linearized system. Fixed points associated with no positive eigenvalues were classified as stable fixed points. The unstable fixed points (F_{saddle}) that we detected in some networks were usually associated with one unstable mode and often helped mediate the transition of trajectories after *Set* towards $F_{terminal}$.

We projected activities of neurons on each trial onto the first three principal components of the total activity covariance matrix (across all trials) to obtain state space trajectories (Fig. 5c). Fixed points were projected onto the same basis for visualization purposes. For various networks, the mean squared speed of network trajectories during interval production over the first three PCs was used to relate dynamics to *Tp* (Fig. 5, Supplementary Fig. 6). We additionally computed the

scaling components (SCs) for the activity of various networks (Supplementary Fig. 7) The speed of the trajectory within the scaling subspace was calculated based on projection onto the first 3 SCs (Supplementary Fig. 7).

Two inhibitory-neuron model

Building on previous work [4], we engineered a two-neuron model that captured the key intuition behind speed control in the interval timing task. The network consists of two mutually inhibitory neurons (*u*, and *v*), which receive a common input representing the contextual cue, denoted as θ (Fig. 6a). The time-varying firing rates of the model neurons are governed by a pair of nonlinear differential equations as follows:

$$\begin{cases} \tau\dot{u} = -u + f(W_{u\theta}\theta - W_{uv}v) \\ \tau\dot{v} = -v + g(W_{v\theta}\theta - W_{vu}u) \end{cases} \cdots(*)$$

Model structure and parameters were explained in the table below.

W_{ij}	$W_{u\theta} = W_{v\theta} = 6$ $W_{uv} = W_{vu} = 6$	Magnitude of synaptic weight from input *j* to unit *i*
θ	$\theta = [0.6, 0.8]$	Balanced input to *u* and *v*
τ	100 ms	Time constant of each unit, $\Delta t = 10ms$ (time step)
$f(x), g(x)$	$\frac{1}{1+\exp(-x)}$	Activation function for *u* and *v* respectively

Supplementary Note Table 2. Structure and parameters for the two-inhibitory neuron model.

The nullcline for each variable is defined as the set of all the states where the derivative of the variable vanishes. For example, solving for $\dot{u} = 0$ provides a solution of the kind $u = f(\theta, v)$, which represents the nullcline for *u* (Fig. 6c). The intersection of the nullclines pertaining to each variable represent a point where variables do not change, i.e., fixed points (FPs). For a range of synaptic weights (W_{ij}) and input levels (θ), the system has two stable FPs that are separated by an unstable one. Increasing θ systematically shifts the position of the two nullclines (Fig. 6c). It also reduces the energy gradient between the FPs (Fig. 6b) causing the system to slow down.

We numerically solved the coupled equations ($*$) to compute the system's response dynamics. The energy (E) of the system was defined as the path integral over a vector field:

$\vec{V} = [\dot{u}, \dot{v}] = -\nabla E(u, v)$. This was computed numerically by integrating the speed along a deterministic trajectory from the unstable FP to the stable ones using the equations above ($*$) to numerically simulate the relaxation of the activity towards the final FP.

Supplementary Software

Matlab codes for generating the dynamics of two-inhibitory neurons are available at https://github.com/wangjing0/two-neuron-model.

References

1. Rasmussen, C. E. & Williams, C. K. I. *Gaussian Processes for Machine Learning*. (University Press Group Limited, 2006).
2. Cunningham, J. P., Gilja, V., Ryu, S. I. & Shenoy, K. V. Methods for estimating neural firing rates, and their application to brain-machine interfaces. *Neural Netw.* **22,** 1235–1246 (2009).
3. Puntanen, S. & Styan, G. P. H. Schur complements in statistics and probability. in *The Schur Complement and Its Applications* (ed. Zhang, F.) 163–226 (Springer US, 2005). doi:10.1007/0-387-24273-2_7
4. Machens, C. K., Romo, R. & Brody, C. D. Flexible control of mutual inhibition: a neural model of two-interval discrimination. *Science* **307,** 1121–1124 (2005).
5. Werbos, P. J. Backpropagation through time: what it does and how to do it. *Proc. IEEE* **78,** 1550–1560 (1990).
6. Sussillo, D. & Barak, O. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Comput.* **25,** 626–649 (2013).