PNAS RESEARCH ARTICLE | NEUROSCIENCE OPEN ACCESS

Dynamic tuning of neural stability for cognitive control

Muyuan Xu[a,1], Takayuki Hosokawa[b], Ken-Ichiro Tsutsui[c], and Kazuyuki Aihara[a]

Edited by Peter Strick, University of Pittsburgh Brain Institute, Pittsburgh, PA; received May 12, 2024; accepted September 29, 2024

The brain is thought to execute cognitive control by actively maintaining and flexibly updating patterns of neural activity that represent goals and rules. However, while actively maintaining patterns of activity requires robustness against noise and distractors, updating the activity requires sensitivity to task-relevant inputs. How these conflicting demands can be reconciled in a single neural system remains unclear. Here, we study the prefrontal cortex of monkeys maintaining a covert rule and integrating sensory inputs toward a choice. Following the onset of neural responses, sensory integration evolves with a 70 ms delay. Using a stability analysis and a recurrent neural network model trained to perform the task, we show that this delay enables a transient, system-level destabilization, opening a temporal window to selectively incorporate new information. This mechanism allows robustness and sensitivity to coexist in a neural system and hierarchically updates patterns of neural activity, providing a general framework for cognitive control. Furthermore, it reveals a learned, explicit rule representation, suggesting a reconciliation between the symbolic and connectionist approaches for building intelligent machines.

cognitive control | prefrontal cortex | dynamical systems | neural network model

Significance

A key challenge for systems neuroscience is to understand the coexistence of robustness and sensitivity in neural networks. In particular, a neural system must be robust against perturbations to its internal dynamics and initial states and yet sensitive to relevant changes in the environment. By analyzing neural dynamics in monkeys' prefrontal cortex and using computational modeling, we demonstrate that these conflicting demands can be reconciled in an ongoing process, where the system hierarchically updates neural representations of task-relevant information (meaning that some representations are updated while others are maintained). Our study provides fundamental insights into the neural mechanism of cognitive control and a basis for building higher cognitive functions in artificial neural network models.

Humans display coordinated, purposeful behavior in spite of a large degree of freedom in the neural system. This ability for cognitive control is believed to stem from the active maintenance and updating of patterns of neural activity in the prefrontal cortex (PFC), which represent goals and rules (1). However, active maintenance and updating have long been noted to involve fundamentally conflicting demands: To actively maintain patterns of activity, a neural system must resist noise and distractors (robustness). However, to update the activity and incorporate new information, it must be sensitive to task-relevant inputs. Traditional theories (2, 3) based on stable equilibrium states (classical attractors) can only account for the maintenance. To solve this problem, it has been assumed that sensory inputs are selectively gated by an additional mechanism (4–7) and only update neural activity when the gate is open. However, such a mechanism cannot account for the conflict between task-relevant inputs and intrinsic noise and is not robust in the sense that updating depends solely on external driving forces.

An alternative way to tackle this problem is to consider nonequilibrium, transient states (8). In such a state, a neural system's stability may change dynamically, allowing the conflicting demands to be reconciled in a single process. To test whether the brain employs such a mechanism, we analyze the neural dynamics in monkeys' PFC during a group reversal task (9, 10). Following stimulus onset, there is a short delay between the initial neural responses and sensory integration. This delay cannot be explained by slow synaptic transmission. However, it may reflect the time required for the neural system's stability to change. To explore this idea, we follow a dynamical systems approach (11–13) and recent studies that have demonstrated that flexible neural computations can be supported by reconfiguring recurrent dynamics based on a static contextual input (14–20). Using an appropriately trained recurrent neural network (RNN) model, we find a mechanism that dynamically tunes a neural system's stability, providing a general solution to the problem of the coexistence of robustness and sensitivity, and hierarchically updates neural representations of task-relevant information.

Results

On each trial, monkeys fixated on a central spot with their hands touching a key. A visual cue indicated a subsequent reward (juice) or punishment (saline, Fig. 1*A*). The cue stimuli were divided into two functional categories (groups), where each category was associated with one outcome (Fig. 1*B*). Following a delay period (0.75 to 1.5 s), the monkeys were required to release the key and make either acquisition (on a juice trial) or avoidance (on a saline trial) reactions. Between sessions (each session consists of 48 to 96 trials), the associative rule was reversed without explicit instructions. Therefore, the monkeys had to maintain the rule and integrate sensory inputs to infer the outcome contingency. After

Author affiliations: [a]International Research Center for Neurointelligence, The University of Tokyo, Bunkyo-ku, Tokyo 113-0033, Japan; [b]Department of Orthoptics, Faculty of Rehabilitation, Kawasaki University of Medical Welfare, Kurashiki, Okayama 701-0193, Japan; and [c]Laboratory of Systems Neuroscience, Graduate School of Life Sciences, Tohoku University, Sendai, Miyagi 980-8577, Japan

Author contributions: M.X. and K.A. designed research; M.X., T.H., and K.-I.T. performed research; M.X. analyzed data; and M.X. and K.A. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: muyuan@ircn.jp.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2409487121/-/DCSupplemental.

Published November 25, 2024.

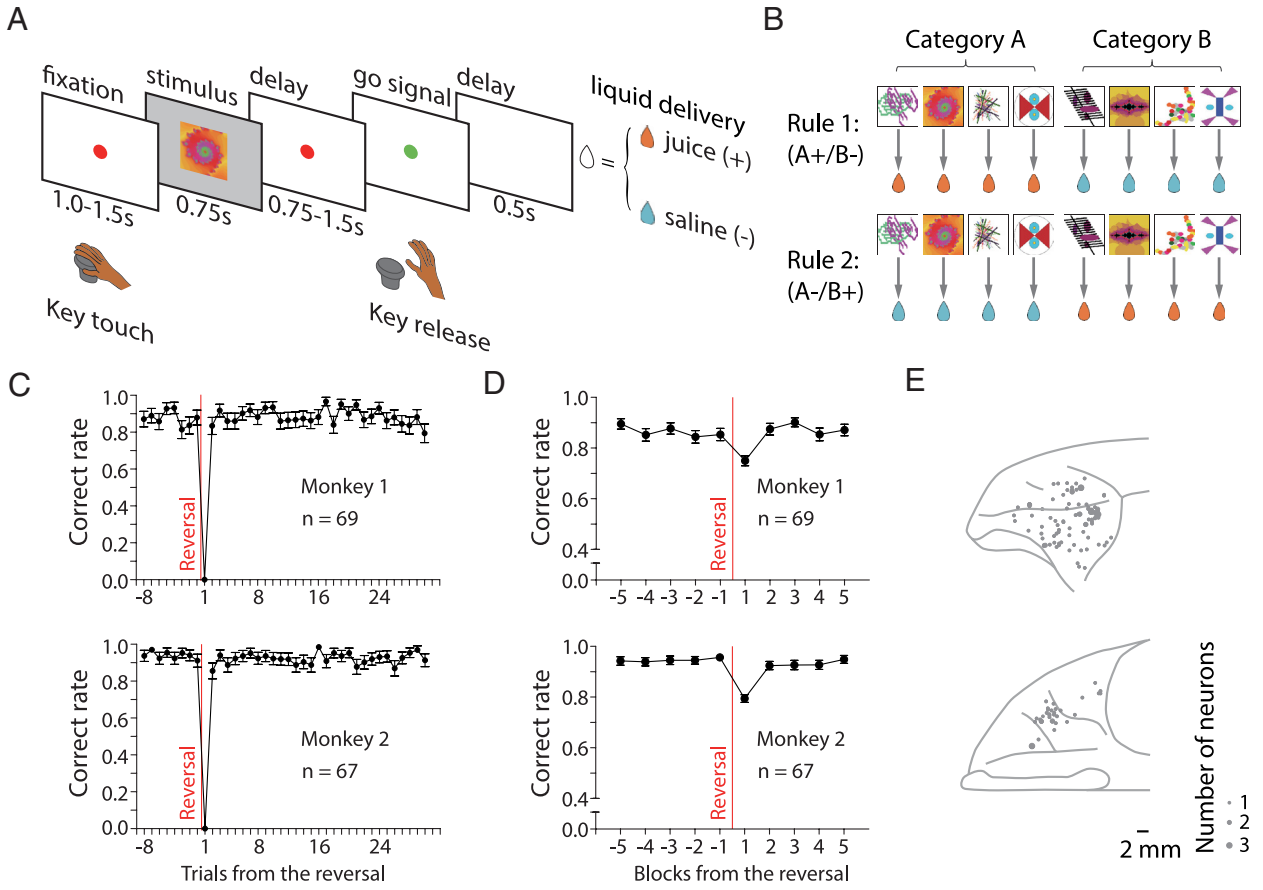

Fig. 1. Behavioral task, performance, and recording. (*A* and *B*) Group reversal task. Monkeys had to infer the outcome (the identity of the liquid delivered at the end of each trial) based on a visual cue displayed at the beginning of each trial. The eight cue stimuli were divided equally into two functional categories, so that four stimuli were associated with the reward (juice) and the other four were associated with the punishment (saline). Between sessions, the rule relating the cue stimuli to the outcomes was reversed without explicit instructions and this reversal was repeated throughout the experiment. (*C* and *D*) Average performance at the reversal. After training, the monkeys always made a mistake on the trial immediately after the reversal (which was always arranged to be a saline trial) and this failure served as an instruction for the change of the rule. (*C*) Average performance per trial. (*D*) Average performance per block (eight trials). (*E*) Recording sites. The size of each dot indicates the number of neurons recorded from that site. The *Upper* and *Lower* panels correspond to the lateral and *Bottom* views of the PFC, respectively.

training, the monkeys learned to adapt to this rule reversal by making a single mistake on the trial immediately after it (Fig. 1 *C* and *D* and *SI Appendix*, Fig. S1). While they performed the task, we recorded neural activity from the dorsal lateral PFC, ventral lateral PFC, and orbitofrontal cortex using single-unit recording methods (Fig. 1*E*).

The group reversal task requires the monkeys to infer both the outcome contingency (on a single trial) and the rule (across multiple trials, 21). Here, we focus on the former process and include only correctly performed trials in our analysis (*Materials and Methods*). In addition, we combine data from both monkeys and consider category, rather than stimulus identity, as a task variable, unless otherwise mentioned. This simplification results in four different trial types, or conditions, each corresponding to one specific combination of the rules and categories. As often reported, individual neurons demonstrate complex, heterogeneous activity (*SI Appendix*, Fig. S2*A*) and a mixed representation of task variables (22, 23, *SI Appendix*, Fig. S2*B*).

To explore the structural organization of neural activity at the population level, we apply principal component analysis (PCA) to the activity from 1.0 s before stimulus onset to 0.75 s after stimulus disappearance. The first two principal components (PCs) reveal strong rotational dynamics (24), which captures 48.1% variance. During the rotation, PFC population activity changes dynamically from encoding the rule to encoding the outcome contingency (Fig. 2*A*), suggesting an update of neural representations. However, the rotational dynamics, by itself, does not seem to distinguish the conditions. Therefore, we assume that it is independent of updating and remove it from the data. The remaining PCs reveal three main features of PFC population activity (Fig. 2*B*). First, as reported previously (14, 20), population activity for each rule occupies a different region of the state space. Second, before stimulus onset, neural trajectories in each of these regions (i.e., trajectories for the same rule but different categories) drift around a specific position (Fig. 2*B*, dashed squares), reminiscent of the dynamics near a stable equilibrium point. Third, after stimulus onset, the trajectories depart from the previous position but remain unseparated for some time before deflecting into different directions (Fig. 2*B*, arrows). Following stimulus disappearance, the trajectories settle down and each drifts around a new position (Fig. 2*B*, crosses).

To confirm these observations, we calculate the multidimensional distances (25) between the states for each pair of conditions (in the full-dimensional state space) and compare their temporal evolution with that of the mean population activity. In the precue period, the mean population activity builds up slowly. Stimulus onset increases the activity sharply at about 55 ms (initial neural responses), peaking at 135 ms and then gradually declining to the baseline level (Fig. 2*C*). The multidimensional distances can be distinguished as the within-rule-cross-category distances (distances between states for the same rules but different categories), cross-rule-cross-category distances (distances between states for different rules and different categories; note that these distances

correspond to the within-contingency distances in the group reversal task), and cross-rule-within-category distances (distances between states for the same categories but different rules), respectively (Fig. 2*D*). Before stimulus onset, the within-rule-cross-category distances have a vanishing value. By contrast, the cross-rule distances have a significant value, suggesting that the rules are encoded by population activity. Following stimulus onset, all distances increase, peak in 400 ms, and then gradually decrease to lower plateaus. The cross-contingency distances become considerably larger than the within-contingency distances, suggesting an encoding of outcome contingencies. On the other hand, the within-contingency distances remain significant, consistent with the rule-dependent organization of neural activity found by PCA (Fig. 2*B*).

The increase of the within-rule-cross-category distances, which indicates sensory integration, starts at 125 ms. Consistent with this observation, the instantaneous speed (25) of population activity traveling along each neural trajectory displays two peaks (Fig. 2*E*). The first peak corresponds approximately to the sharp increase of the mean population activity at stimulus onset. Following this peak, neural population activity slows down and begins to reaccelerate at the same time when the within-rule-cross-category distances start to increase. Together, these observations suggest a 70 ms delay between the onset of neural responses and sensory integration (*Materials and Methods*) and are further confirmed by using data from individual monkeys and by considering stimulus identity as the task variable (*SI Appendix*, Figs. S3 and S4, full-condition analysis).

To understand this delay, we pursue the idea that the population dynamics of neurons can be described by a dynamical system

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{I}, \qquad [1]$$

where $\boldsymbol{x}$ is a vector that contains the activation of neurons and $\boldsymbol{f}$ a function that describes the recurrent interactions. The external input $\boldsymbol{I}$ consists of a group of static contextual inputs coming from other populations of neurons, which collectively "dial" the system's behavior (14–20). The system is initially in a stable equilibrium point. At stimulus onset, its dynamics is reconfigured by setting some of the contextual inputs to different values, evoking transient neural responses. In addition, the system receives sensory inputs, which are modeled as perturbations to its intrinsic dynamics.

The intrinsic dynamics can be represented by a trajectory in the state space. The effect of perturbations depends on the local stability along this trajectory, which in turn describes the behavior of nearby trajectories. Specifically, a small perturbation to the trajectory $\boldsymbol{x} = \boldsymbol{x}(t)$, applied at a point $\boldsymbol{x}_0 = \boldsymbol{x}(t_0)$, causes a deviation $\Delta\boldsymbol{x} = \Delta\boldsymbol{x}(t)$, $t \geq t_0$, which evolves locally according to

$$\frac{d(\boldsymbol{x}+\Delta\boldsymbol{x})}{dt}\Big|_{\boldsymbol{x}=\boldsymbol{x}_0} = \boldsymbol{f}(\boldsymbol{x}_0+\Delta\boldsymbol{x}) + \boldsymbol{I} \approx \boldsymbol{f}(\boldsymbol{x}_0) + \boldsymbol{J}(\boldsymbol{x}_0)\Delta\boldsymbol{x} + \boldsymbol{I}, \qquad [2]$$

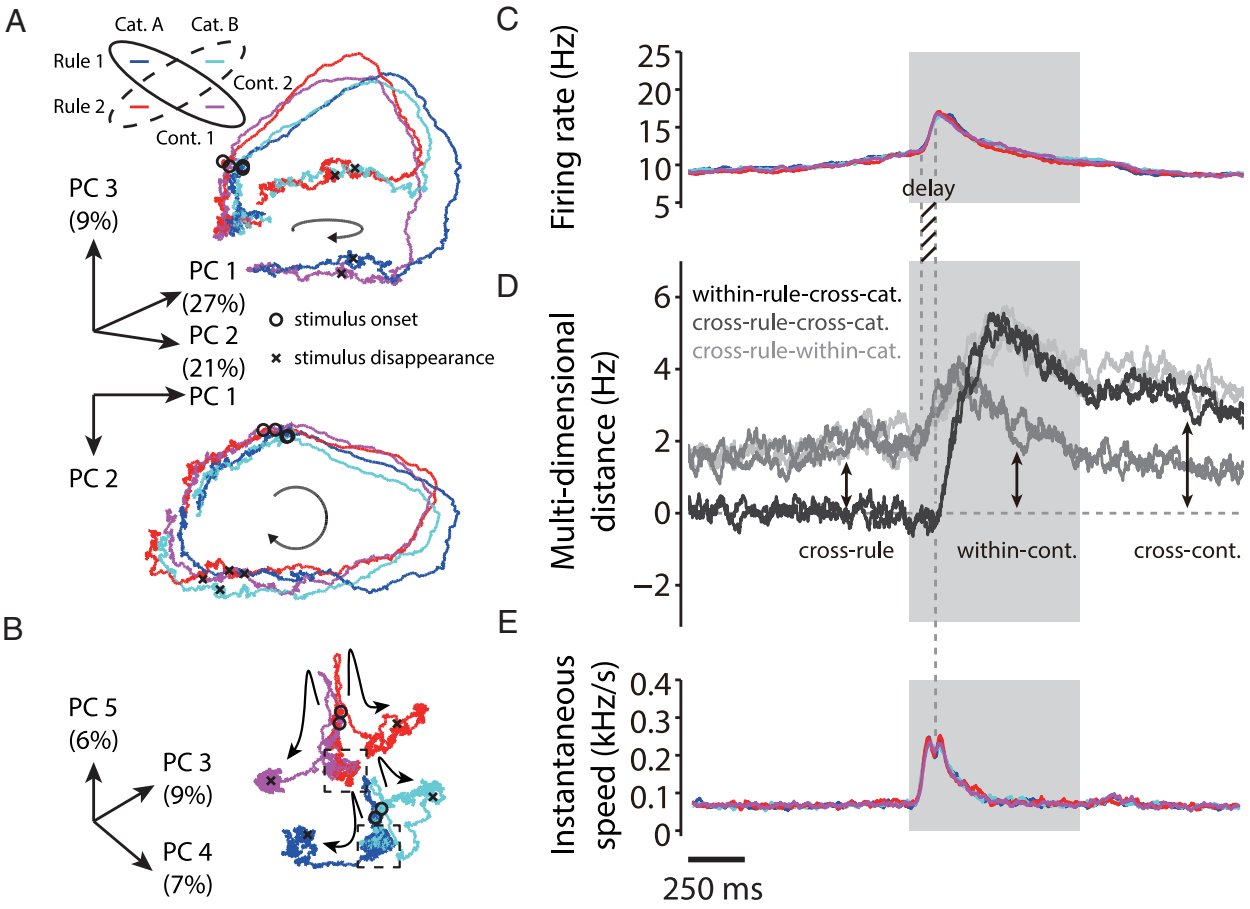

Fig. 2. Neural population dynamics in the group reversal task. (*A*) *Top*: The first three PCs. The arrow indicates the temporal evolution of the state. Percentage numbers show the fraction of variance explained by each PC. *Inset*: Conditions (color-coded). *Bottom*: The first two PCs demonstrate strong rotational dynamics. However, this rotational component does not seem to differ between conditions. (*B*) The 3rd through 5th PCs reveal the structural organization that underlies updating (see the main text). (*C–E*) Dynamic changes in neural population activity. Shaded areas indicate the cue period. Notably, there is a short delay between the initial neural responses and sensory integration (indicated by the increase of the within-rule-cross-category distances at stimulus onset). (*C*) Mean population activity. (*D*) Multidimensional distances between the states for each pair of conditions. (*E*) Instantaneous speeds of population activity traveling along the neural trajectories.

where $\boldsymbol{J}(\boldsymbol{x}_0) = \partial \boldsymbol{f}/\partial \boldsymbol{x}|_{\boldsymbol{x}=\boldsymbol{x}_0}$ is the Jacobian matrix evaluated at $\boldsymbol{x}_0$.

Compared with the unperturbed trajectory (Eq. **1**), we have

$$\frac{d\Delta \boldsymbol{x}}{dt} \approx \boldsymbol{J}(\boldsymbol{x}_0)\Delta \boldsymbol{x}. \quad [3]$$

Thus, the local stability depends on $\boldsymbol{J}(\boldsymbol{x}_0)$. In particular, deviations along eigenvectors associated with the positive eigenvalues of $\boldsymbol{J}(\boldsymbol{x}_0)$ are amplified, causing the perturbed trajectory to diverge from the unperturbed trajectory. Whereas deviations along eigenvectors associated with the negative eigenvalues are reduced, causing the perturbed trajectory to converge to the unperturbed one. The stability at an equilibrium point (26) can be understood as a special case, where the whole trajectory consists of a single point. Notably, the local stability does not depend on the external input $\boldsymbol{I}$. Therefore, when the contextual inputs are adjusted by stimulus onset, the neural system remains stable. However, because the system's dynamics is reconfigured, its state may move to a different region of the state space, causing the stability to change after a delay.

To test whether there could be such a change, we train an RNN model to perform a task analogous to the one performed by the monkeys (Fig. 3). On each trial, neurons in the network receive four contextual inputs. The first two inform the network of the current rule and are referred to as rule-dependent inputs. The other two inform the network of the current phase of the task (the precue period or the cue period) and are referred to as phase-dependent inputs. While the rule-dependent inputs are maintained throughout the trial, the phase-dependent inputs switch their values at a random time that corresponds to stimulus onset. A momentary sensory input follows this switching by a short, fixed delay (100 ms, compared to the time constant for each neuron, which is set to 20 ms). To simplify the training process, we again replace the cue stimuli with the categories. In addition, we assume that the rules are inferred independently and randomly choose the rule and category on each trial. The RNN is trained to make a choice by outputting either 1 or −1 at the end of the trial through a readout unit.

After training, the model reproduces the main features of PFC population activity (Fig. 4 *A* and *B*). Its dynamics can be understood through the arrangement of equilibrium points and their stable and unstable manifolds in the state space, which in turn depends on the contextual inputs. Before stimulus onset, each rule is encoded by a stable equilibrium point. The system's state is attracted to the corresponding point and actively maintains the rule. At stimulus onset, the system's dynamics is reconfigured and the rule-coding stable equilibrium points are moved to a different region of the state space. Instead, a saddle point and two stable equilibrium points encoding the outcome contingencies appear in the current region. The saddle point has a one-dimensional unstable manifold, delimited by the two contingency-coding stable equilibrium points. Each sensory input has a component on one direction along this unstable manifold. Meanwhile, the stable manifold of the saddle point passes approximately through the position of the original rule-coding stable equilibrium point. Thus, the system's

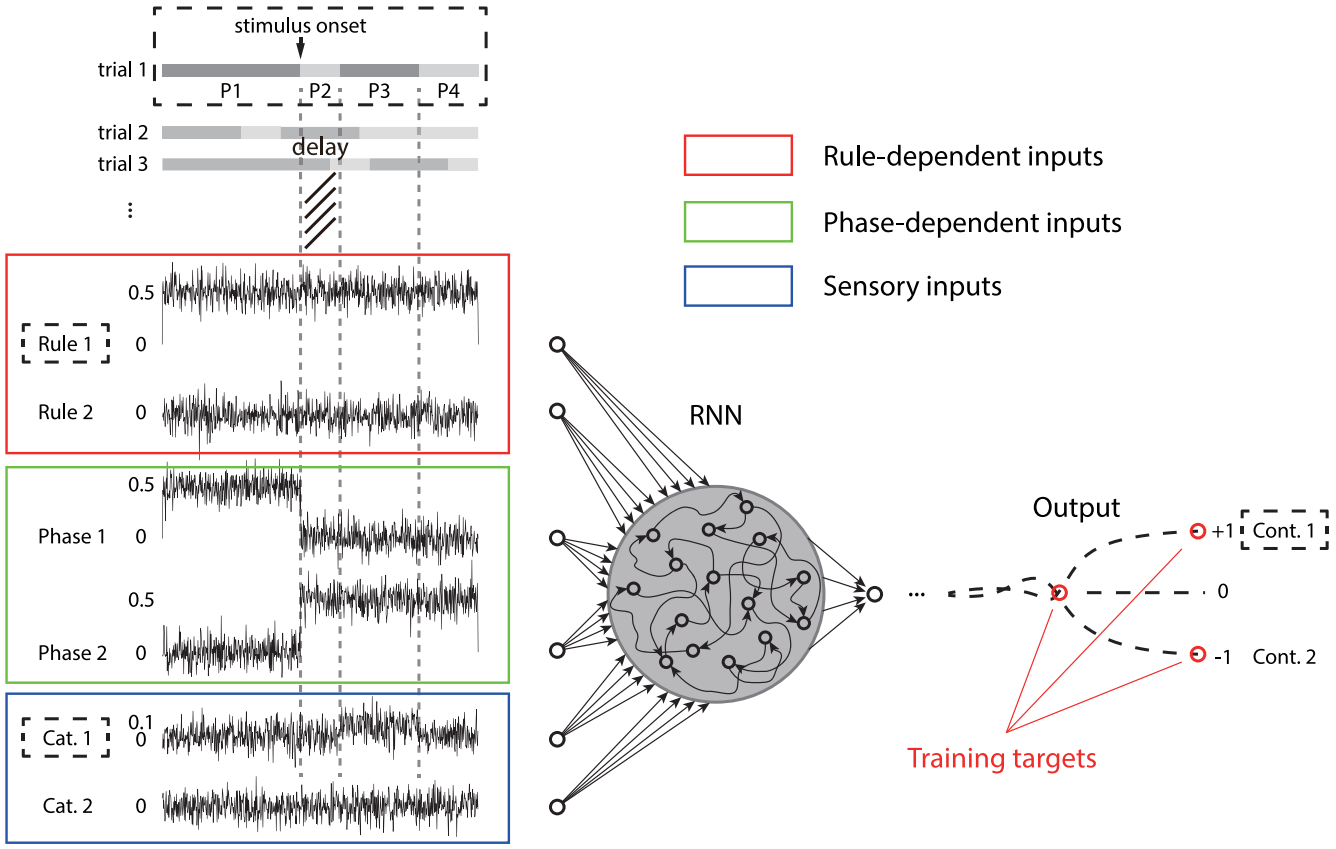

Fig. 3. Training protocol. The neural population is modeled by recurrently connected, nonlinear, rate-based neurons that receive four contextual inputs (rule-dependent and phase-dependent inputs) and two sensory inputs. Each trial is divided into four periods: the precue period (P1), delay period (P2), integration period (P3), and postintegration period (P4), respectively. The system's inputs are defined with respect to these four periods (*Materials and Methods*). The length of each trial and the duration of the delay and integration periods are fixed. By contrast, the duration of the precue period, the rule, and the category are determined randomly. The dashed rectangles show an example trial. To train the network, targets are defined at two time steps: the last time step of the delay period, where the target is set to zero, and the last time step of each trial, where the target is set to either 1 or −1, depending on the combination of the rule and category on that trial. The network is randomly initialized and trained with backpropagation.

state is released and approaches the saddle point along its stable manifold. Following a short period of time, which corresponds to the delay, the system's state is deflected by sensory inputs along the unstable manifold. After the deflection, the state is attracted to one of the contingency-coding stable equilibrium points.

The model dynamics deviates from PFC population activity in two aspects (comparing Fig. 4*A* with Fig. 2*B*): First, PFC trajectories fold back in the state space, which is not captured by the current model. This folding back may reflect the change in the mean population activity and might be accounted for by incorporating biologically plausible mechanisms, such as sparse coding (17, 18). Second, in the PFC, the states encoding the same outcome contingency appear on the same side of the states encoding the rules. However, in the model, they appear on different sides. This discrepancy suggests the existence of additional constraints on the structural organization of neural activity in the brain.

To explore the dynamic change of stability, we calculate, for a given rule, trajectories that correspond to the intrinsic dynamics of the system with noise and sensory inputs turned off. The initial state is taken to be the same state used during training or from a vicinity of the rule-coding stable equilibrium point. Each trajectory lies on one side of the stable manifold of the saddle point. Furthermore, we project the vector field that represents the local dynamics in the state space onto a plane determined by the three points that correspond to the rule-coding and contingency-coding stable equilibrium points (Fig. 4*B*). Far away from the trajectories, the vectors always point "inward" (toward the trajectories). After the reconfiguration of dynamics, there is a rapid change in the neighborhood of the trajectories, where the vectors change from pointing toward the trajectories (Fig. 4*B*, magenta square) to pointing away from them (Fig. 4*B*, red square). This change occurs in agreement with the onset of sensory inputs and implies a destabilization of the neural system. The trajectories deflect despite that sensory inputs are turned off (however, the deflection is delayed, as compared with the deflection of trajectories generated with sensory inputs). After the deflection, the vectors point again toward the trajectories, meaning that the stability is regained. Therefore, the system is destabilized in a temporal window.

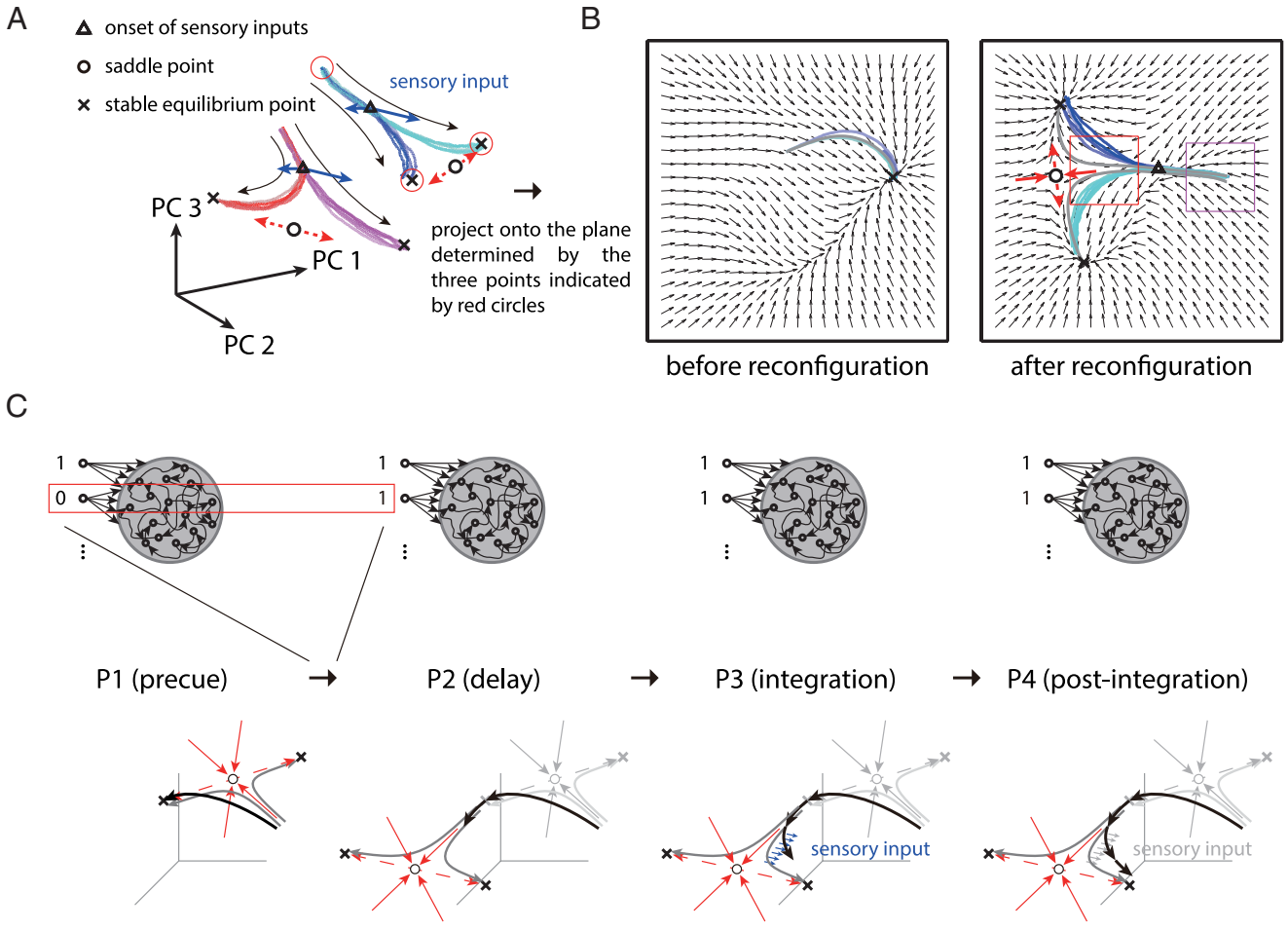

Fig. 4. Dynamic tuning of neural stability and a computational framework for cognitive control. (*A*) Transient neural dynamics for updating (generated after the reconfiguration). The same conventions as in Fig. 2*B*. Sensory inputs are shown at their onset. Equilibrium points (stable equilibrium points and saddle points) are computed separately by fixing the rule-dependent and phase-dependent inputs at corresponding values. Dashed red arrows indicate the local one-dimensional unstable manifold of the saddle point. (*B*) Dynamic tuning of neural stability. Vectors indicate the local dynamics in the state space and are normalized for the purpose of illustration. The gray trajectories correspond to the intrinsic dynamics calculated in the absence of noise and sensory inputs. Before the reconfiguration of dynamics at stimulus onset, the rule is encoded by a stable equilibrium point. After the reconfiguration, this stable equilibrium point is moved away and a saddle point and two contingency-coding stable equilibrium points appear in the current region. The saddle point is arranged in such a way that its stable manifold (indicated locally by the solid red arrows) passes approximately through the position of the original rule-coding stable equilibrium point and its unstable manifold connects to the two contingency-coding stable equilibrium points. The system's state approaches the saddle point along its stable manifold and is destabilized in a temporal window due to the repelling effect of the saddle point. (*C*) A general framework for cognitive control. Branching channels can be switched and "concatenated" from one to another by controlling a group of contextual inputs (shown by the binary values beside the RNN), creating neural representations with complex hierarchical (tree) structures. Arrows in red: the stable and unstable manifolds of a saddle point used for current computation. Arrows in light gray: the stable and unstable manifolds of a saddle point that has been moved away. Solid black arrows: example trajectories generated according to these dynamic structures.

Together, these results suggest that updating can be described using a branching channel, which "connects" the rule-coding stable equilibrium point to the two contingency-coding stable equilibrium points. This channel specifies the way to achieve each outcome and provides an explicit representation of the rule. Under appropriate conditions, all trajectories that enter this channel remain in it, ensuring the robustness against noise and distractors. On the other hand, sensitivity appears at the branching point, allowing task-relevant inputs to be integrated smoothly, driving the system's state into one of the branches. This mechanism of updating can be easily extended to a large number of inputs.

Discussion

We have described a mechanism of updating, which allows robustness and sensitivity to coexist in a neural system and explains the delay between initial neural responses to a stimulus and sensory integration. This mechanism depends on an internal rule representation: a branching channel that exists in the system's state space. Channels, or robust transient dynamics, have been well studied before in neural systems (8, 27). However, branching channels provide an explicit way to construct, through combination, neural representations with complex hierarchical (tree) structures. Indeed, a neural system may harbor a set of branching channels, where each channel can be switched and "concatenated" to the next by controlling a group of contextual inputs (Fig. 4*C*). When the system consists of multiple subsystems, a stimulus may trigger a cascade of updating. For example, in the current task, there may be a frontal area that monitors the failure, makes inferences about the rule (21), and, when necessary, adjusts the rule-dependent inputs to the areas we recorded. Cognitive control thus depends on a dynamic interplay among the states of the subsystems, the contextual inputs each subsystem receives, and the sensory inputs.

This framework concurs with the view that each cortical area functions as an adaptive processor (28). In particular, it suggests that, depending on whether sensory stimuli are processed under the same behavioral context, the difference in the responses they evoke may appear either from the beginning of the responses or with a certain delay. Previously, such a delay has been found in the frontal eye field (29) and the lateral intraparietal area (30, 31) and has been suggested to reflect the reset of a neural integrator (30). Our model also suggests a component in neural integration that is independent of sensory inputs. This component is caused by the repelling effect of the saddle point and may provide an explanation to the "urgency signal" (31) or "instability" (32) in neural integrators. Finally, the framework also concurs with the view that the brain can be compared to a railroad system, where the PFC serves as a switch operator and dynamically sets the patterns of "railroad tracks" according to task demands (1). In our model, the unstable manifold of the saddle point specifies the desired inputs and may thus be interpreted as an attentional template (33, 34). Only the sensory input that matches the template (35, 36) enters working memory (37).

From the perspective of building intelligent machines, our results suggest a reconciliation between the symbolic (38) and connectionist (39) approaches. The ability to construct structured representations opens the possibility to define structure-sensitive operations (40), a key concept in computer science (as opposed to processing only numerical data), as well as in the classical symbolic view of human cognition. On the other hand, because such structured representations can be learned from nonsymbolic vector representations, their semantic meanings might be grounded from bottom up (41): There is no need for an autonomous "symbolic level." The dynamical systems point of view may thus connect different levels of analysis (42–45) and provide insights for building higher cognitive functions, such as reasoning, problem-solving (46), and language processing (47), in connectionist models.

Materials and Methods

Behavioral Task and Recording. Animals were treated according to the NIH's Guide for the Care and Use of Laboratory Animals and the Tohoku University's Guidelines for Animal Care and Use. The project was approved by the Center for Laboratory Animal Research of Tohoku University.

The details of the behavioral task have been described elsewhere (9, 10). Briefly, two male Japanese monkeys (Macaca fuscata) were trained to adapt their behavior to a repeated stimulus–outcome reversal. Each monkey sat on a chair in front of a screen. When a red fixation spot appeared on the screen, the monkey was required to touch a key attached to the chair and fix on the spot. After a precue period that varied between 1.0 s and 1.5 s or was fixed to 1.25 s, a visual cue was displayed at the center of the screen for a fixed period of 0.75 s. A delay period that varied between 0.75 s and 1.5 s or was fixed to 1.25 s followed the cue offset. The fixation spot then turned green as an instruction for the monkey to release the key. Following a second delay period of 0.5 s, or 0 s in some training sessions, a small quantity of liquid was delivered through a double-spout device in front of the monkey's mouth. Depending on the cue and the rule that related the cue stimuli to the outcomes, the liquid could be either juice, which served as a reward, or saline, which served as a punishment. A trial was counted as "correctly performed" if the monkey licked the spout on a juice trial or did not lick the spout on a saline trial in the time window from 0.2 s before to 0.5 s after the liquid delivery. On each trial, the eye movement of the monkey was monitored and the same trial was repeated immediately when the monkey failed to fix at the center of the screen or released the key early. Therefore, the monkey learned to complete the trial even when the predicted outcome was saline.

The rule was reversed covertly between sessions. Each session consisted of 6 to 12 blocks and each block consisted of eight trials. The visual cue was arranged in a randomized way such that each cue stimulus was used once per block. Before the recording, the monkeys were trained with three different stimulus sets (one of these stimulus sets is shown in Fig. 1*B* in the main text). The duration of the precue period and the first delay period varied during the training sessions and the first 3 mo of recording sessions and then became fixed. To guide the single-unit recording, a stereotaxic MRI scan was taken for each monkey. The activity of a neuron was first isolated while the monkey performed the group reversal task with a randomly selected stimulus set and then recorded using the same stimulus set for 4 to 6 sessions. In total, we recorded the extracellular activity of 83 neurons from the dorsal lateral PFC (DLPFC), 88 neurons from the ventral lateral PFC (VLPFC), and 54 neurons from the orbitofrontal cortex (OFC).

Single-Neuron Activity. Data analysis and simulations are performed using Matlab 2013.

We align the data at stimulus onset and estimate the instantaneous firing rate of each neuron using a 50 ms sliding window. In most analyses, category is considered as the task variable and there are $N_{condition} = 4$ conditions. Namely, rule 1 (A+/B−) and category A, rule 1 (A+/B−) and category B, rule 2 (A−/B+) and category A, and rule 2 (A−/B+) and category B. In the full-condition analysis, stimulus identity is considered as the task variable and there are $N_{condition} = 16$ conditions (i.e., the combinations of two rules and eight cue stimuli). In total, there are $N_{neuron} = 225$ neurons and $T = 2{,}450$ time steps (from 1.0 s before stimulus onset to 0.75 s after stimulus disappearance), where each time step corresponds to 1 ms.

To test how the single-neuron activity is modulated by each task variable, we calculate a time-independent regression coefficient for each neuron using multivariable linear regression (14). To this end, we first subtract the across-trial mean from the instantaneous firing rate and divide the result by the corresponding SD. We then use the z-scored firing rate in the following linear regression:

$$\boldsymbol{r}_{i,t}(k) = \beta_{i,t}(1)\boldsymbol{rule}(k) + \beta_{i,t}(2)\boldsymbol{category}(k) + \beta_{i,t}(3)\boldsymbol{contingency}(k) + \beta_{i,t}(4), \quad [4]$$

where k is the trial number and $\boldsymbol{r}_{i,t}(k)$ is the z-scored firing rate of neuron i at time t. $\boldsymbol{rule}(k)$, $\boldsymbol{category}(k)$, and $\boldsymbol{contingency}(k)$ denote the task variables and each takes a binary value. After the regression, the coefficients $\beta_{i,t}(1)$, $\beta_{i,t}(2)$,

and $\beta_{i,t}(3)$ are pooled into three N_{neuron}-by-T matrices, each corresponding to one task variable. We select the column that has the maximal norm in the cue period from each matrix. The time-independent regression coefficients are defined as the entries of that column.

Neural Population Activity. Neural population activity is constructed for each condition by pooling the mean across-trial firing rates into an N_{neuron}-by-T matrix. Each column of the matrix denotes neural population activity at one time step and can be represented by a single point in the N_{neuron}-dimensional state space. The temporal evolution of population activity traces a neural trajectory. To explore the structural organization of neural population activity, we concatenate the matrices and apply principal component analysis (PCA) to the resulting N_{neuron}-by-($N_{condition} \times T$) matrix.

Multidimensional Distance. The multidimensional distance between the states for a pair of conditions is calculated by subtracting the median of a null distribution from the corresponding Euclidean distance in the N_{neuron}-dimensional state space (25). To estimate the null distribution, we randomly shuffle the condition of each trial and calculate the Euclidean distance and repeat this procedure for 1,000 times.

Instantaneous Speed of Population Activity. The instantaneous speed of population activity at time step t is estimated as the norm of $D\left(\boldsymbol{p}_{\mathbf{t-n}}, \boldsymbol{p}_{\mathbf{t+n}}\right)/(2n)$, where $\boldsymbol{p}_{t-n}$ and $\boldsymbol{p}_{t+n}$ denote the population activity and D is the Euclidean distance. $n = 20$ ms is the half-width of a sliding window. Note that, although the absolute value of the estimation depends on n, the temporal profile of the estimation is independent of n.

Duration of the Delay. The within-rule-cross-category distance fluctuates around the line of zero before and shortly after stimulus onset. The beginning of sensory integration is estimated by the last time the distance crosses this line from below. To determine the time of crossing, denoted as t_{cross1}, we first measure the time when the distance starts to increase just before the crossing, which is denoted as t_{rise}. In a 50 ms sliding window, we ask whether the successive distance values over time are correlated, using a Spearman rank-order test (48). Starting from a window centered at $t_0 = 175$ ms (well after the crossing), we move the window backward in 1 ms steps until the test becomes nonsignificant ($P > 0.01$). The time for the earliest positive correlation is defined as t_{rise}. Next, we use a 10 ms sliding window and move it forward in 1 ms steps. t_{cross1} is defined as the earliest time when the mean distance value in this window becomes positive.

The onset of neural responses is estimated in a similar way. The mean population activity increases slowly before and shortly after stimulus onset. We first detrend the activity by extrapolating a linear fit generated based on the activity between −300 ms and stimulus onset. We then use a similar procedure to determine the time when the activity is increased sharply by stimulus onset, denoted as t_{cross2}.

Using these methods, we obtain two estimates for t_{cross1} (129 ms and 135 ms) and four estimates for t_{cross2} (35 ms, 63 ms, 45 ms, and 36 ms). However, for some conditions, may be underestimated due to a drift in the baseline level (i.e., imperfect detrending). Therefore, for a conservative estimate, we use the following values instead: For the onset of neural responses, we use $t_{response} = \text{mean}\left(t_{cross2}\right) + \text{std}\left(t_{cross2}\right) = 57.7$ ms. For the onset of sensory integration, we use $t_{integration} = \text{mean}\left(t_{cross1}\right) + \text{std}\left(t_{cross1}\right) = 127.8$ ms. The delay thus has a duration of 70 ms.

The RNN Model. The RNN model consists of $N = 100$ recurrently connected, nonlinear, rate-based neurons. Its dynamics is described by the following equations:

$$\tau \frac{\boldsymbol{dx}}{\boldsymbol{dt}} = -\boldsymbol{x} + \boldsymbol{W}_{rec}\boldsymbol{r} + \boldsymbol{W}_{in}\boldsymbol{u} + \boldsymbol{c}_{\boldsymbol{x}} + \rho_{\boldsymbol{x}}, \quad [5]$$

$$\boldsymbol{r} = \tanh(\boldsymbol{x}), \quad [6]$$

$$z = \boldsymbol{W}_{out}\boldsymbol{r} + c_z, \quad [7]$$

where $\boldsymbol{x}$ is an N-dimensional vector that contains the activation of neurons. $\boldsymbol{r}$ is a vector that represents the firing rates and is obtained by an element-wise application of the nonlinear hyperbolic tangent function to $\boldsymbol{x}$. $\tau = 20$ ms is the time constant of the neurons.

The recurrent interactions between neurons are described by a matrix $\boldsymbol{W}_{rec}$. $\boldsymbol{u} = [u_{rule1}(t)u_{rule2}(t)u_{phase1}(t)u_{phase2}(t)u_{sensory1}(t)u_{sensory2}(t)]^T$ denotes the system's input, where $u_{rule1}(t)$ and $u_{rule2}(t)$ are the rule-dependent inputs, $u_{phase1}(t)$ and $u_{phase2}(t)$ are the phase-dependent inputs, and $u_{sensory1}(t)$ and $u_{sensory2}(t)$ are the sensory inputs. The external inputs contribute to the activation of neurons through an N-by-six matrix $\boldsymbol{W}_{in}$. In addition, each neuron receives a constant offset current $\boldsymbol{c}_{\boldsymbol{x}}$ and intrinsic noise $\boldsymbol{\rho}_{\boldsymbol{x}}$. At each time step, $\boldsymbol{\rho}_{\boldsymbol{x}}$ is drawn from a normal distribution with zero mean and a SD $3.1623 \cdot \sqrt{\Delta t} = 0.1$. The RNN's choice is readout through a one-by-N matrix $\boldsymbol{W}_{out}$ and an offset c_z.

We simulate each trial for $T = 800$ ms. The differential equations are integrated using the Euler–Maruyama method with $\Delta t = 1$ms. Each trial is divided into four periods (see Fig. 3 in the main text). The first period is the precue period. The duration of this period changes randomly from 200 ms to 500 ms. For simplicity, we use a finite step size of 75 ms. The second and third periods are the delay and integration periods. The duration of these periods are fixed to 100 ms and 200 ms, respectively. The fourth period is the postintegration period and its duration changes according to that of the first period.

Each input consists of a time-dependent offset and random noise. By default, the offset is set to zero. A rule is represented by setting the offset for the corresponding rule-dependent input to $M_{rule} = 0.5$ and a category is represented by setting the offset for the corresponding sensory input to $M_{sensory} = 0.1$, during the integration period. The offset for the phase-dependent input $u_{phase1}(t)$ is set to $M_{phase} = 0.5$ during the precue period. The offset for $u_{phase2}(t)$ is set to M_{phase} during the delay, integration, and postintegration periods. Finally, the noise is drawn at each time step from a normal distribution with zero mean and a SD $3.1626\sqrt{\Delta t} = 0.1$.

The RNN is initialized following a standard procedure (14, 49). The entries of W_{rec} and W_{out} are drawn from a normal distribution with zero mean and variance $1/N$. The entries of W_{in} are drawn from a normal distribution with zero mean and variance 1/6. The initial state of the RNN, denoted by a vector x_0, and the constant offsets c_x and c_z are drawn from a uniform distribution on the interval $[-1, 1]$.

Following the initialization, the RNN is trained with a supervised method known as the Hessian-Free optimization (50). The target $\hat{z}(t)$ is defined at two time steps: At the last time step of the delay period (i.e., the time step immediately before sensory integration), the target is set to zero. At the last time step of each trial, the target is set to either 1 or −1, depending on the combination of the rule and category on that trial. The error function is defined as

$$\frac{1}{2S}\sum_{s=1}^{S}\sum_{t=t_1,t_2}\left(z_s(t)-\hat{z}_s(t)\right)^2, \quad [8]$$

where s is the trial number. t_1 and $t_2 = T$ denote the time steps at which the target is defined. The Hessian-Free optimization uses backpropagation through time (51) to minimize the error with respect to the model parameters, including W_{rec}, W_{in}, W_{out}, x_0, c_x, and c_z. The RNN is trained with $S = 1{,}000$ independently generated trials.

Analysis of Model Dynamics. We align the simulated data at stimulus onset and apply PCA to the activity from −200 ms to 300 ms. Equilibrium points are found by fixing the contextual inputs at corresponding values and minimizing the squared norm of the r.h.s. of Eq. **5** (14, 26). In addition, we perform linear analysis around the saddle point to identify the unstable manifold locally.

In our simulation, we do not assume any specific structure in sensory inputs and initialize their directions in the state space (represented by the columns of W_{in}) randomly. After training, the two input directions form an angle of 88.2°. To test whether they have components on the one-dimensional unstable manifold, we calculate the angles between the input directions and the unstable manifold and compare those angles with a null distribution. Each input direction forms a similar angle with the unstable manifold (80.1° and 80.2°, respectively). The null distribution is estimated as follows: First, we fix two orthogonal N-dimensional vectors and calculate the internal bisector of the angle formed by these vectors. Next, we randomly choose a third vector from the orthogonal complementary space of the internal bisector and calculate the angles between the third and the first two vectors. Because of the symmetry, the third vector always forms identical angles with the first two. Finally, we repeat this procedure for 10^5 times. The

comparison with the null distribution reveals a significant overlap between the input directions and the unstable manifold ($P = 0.015$).

Data, Materials, and Software Availability. Data analyzed in this study are available from the corresponding author upon reasonable request. The code used to train and analyze the RNN model is available on GitHub (52).

ACKNOWLEDGMENTS. We thank Charles Yokoyama and Motomasa Komuro for providing valuable comments on the manuscript. This work was supported by International Research Center for Neurointelligence at The University of Tokyo Institutes for Advanced Study (M.X. and K.A.), Institute for AI and Beyond at The University of Tokyo (M.X. and K.A.), Japan Society for Promotion of Science Grant Nos. 26115501 (T.H.), 26750377 (T.H.), 24243067 (K.–I.T.), 24223004 (K.–I.T.), 20H00104 (K.–I.T.), 23H00073 (K.–I.T.), JP20H05921 (K.A.), Japan Science and Technology Agency Moonshot Research and Development Grant Nos. JPMJMS2292 (K.–I.T.), JPMJMS2021 (K.A.), Japan Agency for Medical Research and Development Grant No. JP23dm0307009 (K.A.), and Council for Science, Technology, and Innovation, Cross-ministerial Strategic Innovation Promotion Program (SIP), the 3rd period of SIP, Grant Nos. JPJ012207 and JPJ012425 (K.A.).

1. E. K. Miller, J. D. Cohen, An integrative theory of prefrontal cortex function. *Annu. Rev. Neurosci.* **24**, 167–202 (2001).
2. J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.* **79**, 2554–2558 (1982).
3. D. J. Amit, *Modeling Brain Function: The World of Attractor Neural Networks* (Cambridge University Press, 1992).
4. J. D. Cohen, T. S. Braver, R. O'Reilly, A computational approach to prefrontal cortex, cognitive control and schizophrenia: Recent developments and current challenges. *Philos. Trans. R. Soc. Lond. B Biol. Sci.* **351**, 1515–1527 (1996).
5. S. Hochreiter, J. Schmidhuber, Long short-term memory. *Neural. Comput.* **9**, 1735–1780 (1997).
6. T. S. Braver, J. D. Cohen, "On the control of control: The role of dopamine in regulating prefrontal function and working memory" in *Control of Cognitive Processes: Attention and Performance XVIII*, S. Monsell, J. Driver, Eds. (MIT Press, 2000), pp. 713–737.
7. M. J. Frank, B. Loughry, R. C. O'Reilly, Interactions between frontal cortex and basal ganglia in working memory: A computational model. *Cogn. Affect. Behav. Neurosci.* **1**, 137–160 (2001).
8. M. Rabinovich, R. Huerta, G. Laurent, Transient dynamics for neural processing. *Science* **321**, 48–50 (2008).
9. K. I. Tsutsui, T. Hosokawa, M. Yamada, T. Iijima, Representation of functional category in the monkey prefrontal cortex and its rule-dependent use for behavioral selection. *J. Neurosci.* **36**, 3038–3048 (2016).
10. T. Hosokawa *et al.*, Behavioral evidence for the use of functional categories during group reversal task performance in monkeys. *Sci. Rep.* **8**, 15878 (2018).
11. E. E. Fetz, Are movement parameters recognizably coded in the activity of single neurons? *Behav. Brain Sci.* **15**, 679–690 (1992).
12. D. V. Buonomano, W. Maass, State-dependent computations: Spatiotemporal processing in cortical networks. *Nat. Rev. Neurosci.* **10**, 113–125 (2009).
13. K. V. Shenoy, M. Sahani, M. M. Churchland, Cortical control of arm movements: A dynamical systems perspective. *Annu. Rev. Neurosci.* **36**, 337–359 (2013).
14. V. Mante, D. Sussillo, K. V. Shenoy, W. T. Newsome, Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature* **503**, 78–84 (2013).
15. G. Hennequin, T. P. Vogels, W. Gerstner, Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron* **82**, 1394–1406 (2014).
16. D. Sussillo, M. M. Churchland, M. T. Kaufman, K. V. Shenoy, A neural network that finds a naturalistic solution for the production of muscle activity. *Nat. Neurosci.* **18**, 1025–1033 (2015).
17. H. F. Song, G. R. Yang, X. J. Wang, Training excitatory-inhibitory recurrent neural networks for cognitive tasks: A simple and flexible framework. *PLoS Comput. Biol.* **12**, e1004792 (2016).
18. W. Chaisangmongkon, S. K. Swaminathan, D. J. Freedman, X. J. Wang, Computing by robust transience: How the fronto-parietal network performs sequential, category-based decisions. *Neuron* **93**, 1504–1517 (2017).
19. J. Wang, D. Narain, E. A. Hosseini, M. Jazayeri, Flexible timing by temporal scaling of cortical responses. *Nat. Neurosci.* **21**, 102–110 (2018).
20. E. D. Remington, D. Narain, E. A. Hosseini, M. Jazayeri, Flexible sensorimotor computations through rapid reconfiguration of cortical dynamics. *Neuron* **98**, 1005–1019 (2018).
21. M. Sarafyazd, M. Jazayeri, Hierarchical reasoning by neural circuits in the frontal cortex. *Science* **364**, eaav8911 (2019).
22. M. Rigotti *et al.*, The importance of mixed selectivity in complex cognitive tasks. *Nature* **497**, 585–590 (2013).
23. D. Raposo, M. T. Kaufman, A. K. Churchland, A category-free neural population supports evolving demands during decision-making. *Nat. Neurosci.* **17**, 1784–1792 (2014).
24. M. M. Churchland *et al.*, Neural population dynamics during reaching. *Nature* **487**, 51 (2012).
25. M. G. Stokes *et al.*, Dynamic coding for cognitive control in prefrontal cortex. *Neuron* **78**, 364–375 (2013).
26. D. Sussillo, O. Barak, Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural. Comput.* **25**, 626–649 (2013).
27. R. Laje, D. V. Buonomano, Robust timing and motor patterns by taming chaos in recurrent neural networks. *Nat. Neurosci.* **16**, 925–933 (2013).
28. C. D. Gilbert, M. Sigman, Brain states: Top-down influences in sensory processing. *Neuron* **54**, 677–696 (2007).
29. J. I. Gold, M. N. Shadlen, Representation of a perceptual decision in developing oculomotor commands. *Nature* **404**, 390–394 (2000).
30. J. D. Roitman, M. N. Shadlen, Response of neurons in the lateral intraparietal area during a combined visual discrimination reaction time task. *J. Neurosci.* **22**, 9475–9489 (2002).
31. A. K. Churchland, R. Kiani, M. N. Shadlen, Decision-making with multiple alternatives. *Nat. Neurosci.* **11**, 693–702 (2008).
32. B. W. Brunton, M. M. Botvinick, C. D. Brody, Rats and humans can optimally accumulate evidence for decision-making. *Science* **340**, 95–98 (2013).
33. J. Duncan, G. W. Humphreys, Visual search and stimulus similarity. *Psychol. Rev.* **96**, 433–458 (1989).
34. R. Desimone, J. Duncan, Neural mechanisms of selective visual attention. *Annu. Rev. Neurosci.* **18**, 193–222 (1995).
35. D. M. MacKay, "Towards an information-flow model of human behaviour" in *Systems Research for Behavioral Science*, W. Buckley, Ed. (Routledge, 2017), pp. 359–369.
36. D. Mumford, On the computational architecture of the neocortex: II The role of cortico-cortical loops. *Biol. Cybern.* **66**, 241–251 (1992).
37. A. Baddeley, *Working Memory* (Oxford University Press, 1986).
38. A. Newell, H. A. Simon, Computer science as empirical inquiry: Symbols and search. *Commun. ACM* **19**, 113–126 (1976).
39. D. E. Rumelhart, J. L. McClelland; PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations* (MIT Press, 1986).
40. J. A. Fodor, Z. W. Pylyshyn, Connectionism and cognitive architecture: A critical analysis. *Cognition* **28**, 3–71 (1988).
41. S. Harnad, The symbol grounding problem. *Phys. D* **42**, 335–346 (1990).
42. D. Marr, *Vision* (Freeman, San Francisco, 1982).
43. D. Broadbent, A question of levels: Comment on McClelland and Rumelhart. *J. Exp. Psychol. Gen.* **114**, 189–192 (1985).
44. D. E. Rumelhart, J. L. McClelland, Levels indeed! A response to Broadbent. *J. Exp. Psychol. Gen.* **114**, 193–197 (1985).
45. S. Pinker, A. Prince, On language and connectionism: Analysis of a parallel distributed processing model of language acquisition. *Cognition* **28**, 73–193 (1988).
46. A. Newell, H. A. Simon, *Human Problem Solving (Vol. 104, No. 9)* (Prentice-hall, Englewood Cliffs, NJ, 1972).
47. N. Chomsky, *Aspects of the Theory of Syntax* (MIT Press, Cambridge, MA, 1965).
48. T. Sato, J. D. Schall, Pre-excitatory pause in frontal eye field responses. *Exp. Brain Res.* **139**, 53–58 (2001).
49. K. Rajan, L. F. Abbott, Eigenvalue spectra of random matrices for neural networks. *Phys. Rev. Lett.* **97**, 188104 (2006).
50. J. Martens, I. Sutskever, "Learning recurrent neural networks with hessian-free optimization" in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (2011), pp. 1033–1040.
51. P. J. Werbos, Backpropagation through time: What it does and how to do it. *Proc. IEEE* **78**, 1550–1560 (1990).
52. M. Xu, T. Hosokawa, K.–I. Tsutsui, K. Aihara, Dynamic-tuning-of-neural-stability-RNN. GitHub. https://github.com/muyuan-xu/Dynamic-tuning-of-neural-stability-RNN. Deposited 12 November 2024.