Humans actively reconfigure neural task states

Harrison Ritz[1,✉], **Aditi Jha**[1,2], **Nathaniel D. Daw**[1,3], **and Jonathan D. Cohen**[1,3]

[1]Princeton Neuroscience Institute, Princeton University
[2]Department of Statistics, Stanford University
[3]Department of Psychology, Princeton University

The ability to switch between tasks is a core component of adaptive cognition, yet a mechanistic understanding of this capacity has remained elusive. In particular, there are longstanding debates over the extent to which task-switching is primarily influenced by interference from previous tasks or active preparation for upcoming tasks. We advance this debate by modeling the evolution of task representations in human electroencephalographic recordings as linear dynamical systems over a high-dimensional latent space. Using control theoretic analyses of dynamical systems fit to both brains and task-optimized neural networks, we reveal neural signatures of active preparation that reconcile the role of interference and control in task switching. These findings inform a core debate in cognitive control and offer a promising new paradigm for human neuroimaging analysis.

task switching | electroencephalography | state-space model | recurrent neural networks

Correspondence: ***hr0283 @princeton.edu***

Humans have a remarkable capacity to flexibly adapt how they perform tasks (1–4). This flexibility reflects a core feature of goal-directed cognition (5) and is a strong indicator of cognitive changes across the lifespan (6, 7) and in mental health (8, 9). Despite the centrality of cognitive flexibility in our mental life, we still have a limited mechanistic understanding of how people rapidly configure task processing to achieve their moment-to-moment goals.

Previous research has investigated task preparation primarily through the lens of performance costs associated with rapidly switching stimulus-response mappings ('task sets'; (2)). There is broad agreement that these switch costs depend both on the interference of the previous trial and preparation for the upcoming trial (2, 3). However, there are long-standing debates about the relative contributions of these two processes. The 'task set inertia' hypothesis argues that switch costs largely reflect interference from the previous trial, which passively decays over time (10, 11). The 'active reconfiguration' hypothesis argues that switch costs largely reflect active preparation for the upcoming task through cognitive control (1, 12). Previous experiments have attempted to adjudicate between these hypotheses using behavioral analyses or time-locked neuroimaging (13–18), but have faced challenges in adequately characterizing the underlying dynamics of task preparation.

In parallel, recent research in the neuroscience of *motor* control has explored how the brain prepares for upcoming motor actions. A major advance in this field has been the application of dynamical systems theory to explain how neural populations evolve toward action-specific brain states (19, 20). This framework offers a promising approach for studying the dynamics of *cognitive* control, aligning with recent simulations that use dynamical systems theory to model transitions between neurocognitive task states (21–23). Despite the opportunity for dynamical approaches to help adjudicate between classic theories, this approach remains underutilized for characterizing the neural dynamics of task preparation.

To bridge between dynamics-based theories and neural mechanisms, we fit large-scale dynamical systems to two human electroencephalographic (EEG) datasets, tracking the evolution of neural task states during preparation. Using control-theoretic analysis of these fitted dynamical systems, we tested the predictions of the Inertia and Reconfiguration hypotheses in both neural recordings and task-optimized recurrent neural networks (RNNs). We find that common dynamical motifs across biological and artificial neural systems support the active reconfiguration of neural task states, while reconciling these competing accounts.

Neural measures of task preparation

We first analyzed scalp EEG recordings made in a recent experiment by Hall-McMaster et al. (24), in which 30 participants performed a cued task-switching paradigm (Fig. 1A; Method 1 and 2). On each trial, participants were cued whether to perform the 'shape' or 'color' task. After a delay period, participants saw a colored shape and responded with a button press based on the cued dimension. Participants had poorer performance when they had to switch tasks, and the authors' EEG analyses revealed sensor-level encoding of task identity during the delay period. This task encoding was interpreted as a preparatory brain state that facilitated task-appropriate behavior (24–27).

It has previously been proposed that the strength of task encoding on switch versus repeat trials can adjudicate between task-switching theories (15, 18). However, the sensor-level analyses of these EEG data did not show significant differences in encoding strength (Fig. S1; Method 3). However, sensor-level encoding strength may be an insufficient test of theories that make specific predictions about the *dynamics* of preparatory task representations, which are covert and reflected only indirectly in measured signals. The Inertia hypothesis predicts the passive decay of the previous task representation, whereas the Reconfiguration hypothesis predicts strengthening of representations for the upcoming task. To test these predictions, we turned to a modeling approach that can characterize the rich temporal structure of the underlying neural system.

Inferring neural state-space dynamics

We quantified the evolution of preparatory task representations using a linear state space model (SSM). Unlike traditional neuroimaging analyses, SSMs account for the temporal structure of neural representations, here modeled as a linear dynamical system over a set of latent factors. Linear dynamical systems have long been used in computational neuroscience to model latent dynamics in multiunit neural recordings (28–30). Modeling activity at the level of latent factors can capture rich (non-Markovian) dynamics (31), reflecting the neural sources of the recorded electrode voltage. Despite the nonlinearity of neural activation functions (32), linear models often provide highly predictive models of neural dynamics (33–35). Critically, the linear SSM can be characterized using a suite of tools from dynamical systems theory and control theory (36, 37), providing an interpretable alternative to nonlinear SSMs, such as hidden Markov models (38–40) or RNNs (41, 42).

The SSM consisted of a set of latent factors (state vector $\mathbf{x}$; Fig. 1B; Method 5) that evolved according to an autoregressive process described by a recurrence matrix (A), inputs (with input vector $\mathbf{u}$ and input matrix B), and Gaussian 'process noise' ($\mathbf{w}_t \sim \mathcal{N}(0, W)$):

$$\begin{aligned} \mathbf{x}_t &= A\mathbf{x}_{t-1} + B\mathbf{u}_{t-1} + \mathbf{w}_t \\ \mathbf{x}_{t=1} &= B_0\mathbf{u}_0 + \mathbf{w}_1 \end{aligned} \tag{1}$$

We captured task switching dynamics by including both the previous and current task identity as inputs (see Method 6 for complete predictor list). To provide additional flexibility to the model, we expanded the inputs with a temporal spline basis (bottom of Fig. 1B; Method 6). To better capture task set inertia, the previous task identity ($B_0^{\text{prevTask}}\mathbf{u}_0^{\text{prevTask}}$) was included in the regressors for the initial condition (see Method 6).

To predict the EEG observations ($\mathbf{y}$; see below), factors were 'read out' through the observation matrix (C) and corrupted by 'observation noise' ($\mathbf{v}_t \sim \mathcal{N}(0, V)$):

$$\mathbf{y}_t = C\mathbf{x}_t + \mathbf{v}_t \tag{2}$$

We reduced the dimensionality of the EEG recordings using principal component analysis (PCA), retaining components that accounted for 99% of the variance (between 14-27). We fit the model to an epoch that included the cue period, delay period, and the beginning of the trial (before any response). We estimated the maximum a posteriori parameters of the SSM using expectation-maximization (Method 3, (43, 44)), developing an open-source package called `StateSpaceAnalysis.jl` (45).

EEG recordings are well-explained by a high-dimensional SSM

We first assessed the dimensionality of the latent space, choosing the number of factors based on cross-validated prediction. In systems neuroscience, latent variable models of multiunit neural recordings typically have fewer factors than observation dimensions (46, 47). Surprisingly, we found that an SSM

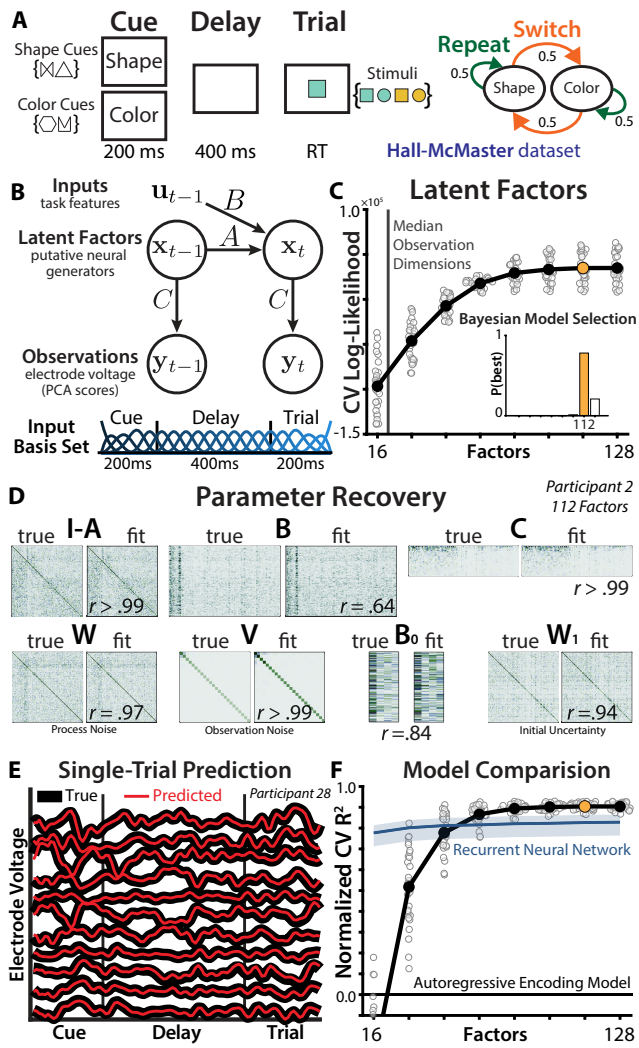

Fig. 1. State space analysis. **(A)** Left: Participants saw a task cue (two symbols mapped each task), waited through a delay period, then responded based on the cued dimension. Right: The tasks had an equal change of repeating or switching between trials. **(B)** Top: Schematic of the state space model. 'PCA': principal components analysis. Bottom: temporal spline basis set, spanning the epoch. **(C)** The log-likelihood from a held-out test set (y-axis), across different number of factors (x-axis). Dots indicate single subjects; log-likelihood is mean-centered within-participant to show relative differences. Inset: Bayesian model selection. **(D)** Parameter recovery on a synthetic dataset generated from a participants' estimated parameters. **(E)** Predictive accuracy in one example trial from the test set across a subset of the EEG electrodes. Thick black lines indicate EEG voltage, red lines indicate next-timestep predictions from a Kalman filter (112-factor model). **(F)** Cross-validated coefficient of determination for SSMs across factor size, relative to autoregressive encoding models (zero indicates equally-good fit). Blue line and shading indicates the 95% interval of the predictive accuracy of parameter-matched RNNs fit to each participant. Y-axis truncated for visualization.

with many more factors than the original number of EEG electrodes provided the best cross-validated fit (best model: 112 factors with 90% protected exceedance probability; Fig. 1C; (48)). This 'lifting' of the dimensionality may allow a linear model to better approximate the underlying nonlinear system (49).

To validate the reliability of this high-dimensional model, we evaluated the SSM on a synthetic dataset based on a participant's parameters (Fig. 1D; after factor alignment, see Method 6). We could accurately recover the ground-truth pa-

rameters, supporting the precision of the fitting procedure and the ability to differentiate effects between parameters.

The best-fitting model was highly accurate at predicting next-timestep EEG activity in held-out data (median R^2_{CV} = .98), even at the level of single trials (Fig. 1E). The model also made accurate predictions at longer horizons (median t+200 ms accuracy: R^2_{CV} = .43; Fig. S2A), and open-loop simulations closely reproduced the empirical power spectra (median R^2_{CV} = .98; Fig. S2B; (50)). Finally, model comparisons revealed that the best-fitting SSM performed better than both simpler autoregressive encoding models (i.e., without latent factors) and nonlinear recurrent neural networks (matched for parameters; Fig. 1F; Method 7). This suggests that these SSMs strike a good balance between the interpretability of linear models and the expressivity of nonlinear models.

Latent neural dynamics reveal stable task control

Having validated the predictive power of the SSM, we explored whether the brain transitions into task-specific neural attractors (points of local convergence) during preparation. While the Inertia and Reconfiguration hypotheses are agnostic to this property, task attractors are central to dynamic theories of task switching (21, 22), and to our knowledge remain untested.

To assess the evolution of neural task states towards an attractor, we factorized the SSM into separate subsystems for each task, a procedure known as 'additive state decomposition' ((51); Method 8):

$$\begin{aligned} \mathbf{x}_t^{\text{task}} &= A\mathbf{x}_{t-1}^{\text{task}} + B^{\text{task}}\mathbf{u}_{t-1}^{\text{task}} \\ \mathbf{x}_{t=1}^{\text{task}} &= B_0^{\text{prevTask}}\mathbf{u}_0^{\text{prevTask}} \end{aligned} \tag{3}$$

When we visualized participants' task trajectories using singular value decomposition (SVD; Method 8), we observed distinct trajectories for each task that appeared relatively consistent across switch and repeat trials (Fig. 2A). Thus, while the fitted systems were stable (resulting in a global attractor), time-varying inputs could produce complex trajectories.

We first verified that task states converged to similar locations from different initial conditions, measuring the Euclidean distances between states across switch and repeat conditions (Method 8). Brain states for the same task (e.g., red-magenta in Fig. 2A) became more similar over time (group-level t-test on distance-time correlations: d = -1.1, p < .001; Fig. 2B, black). By contrast, states for different tasks (e.g., red-cyan in Fig. 2A) diverged during the cue period and then maintained separation (Fig. 2B, pink). Both of these dynamics support convergence to a task-specific attractor state.

We next assessed whether task states decelerated as they settled into their putative attractors. Consistent with settling, task state velocity slowed over time (group-level t-test on velocity-time correlations: ds < -5.0, ps < .001 across switch and repeat trials). Moreover, task states decelerated more as they approached states that were reached after different initial conditions (i.e., the putative attractor location; group-level t-test

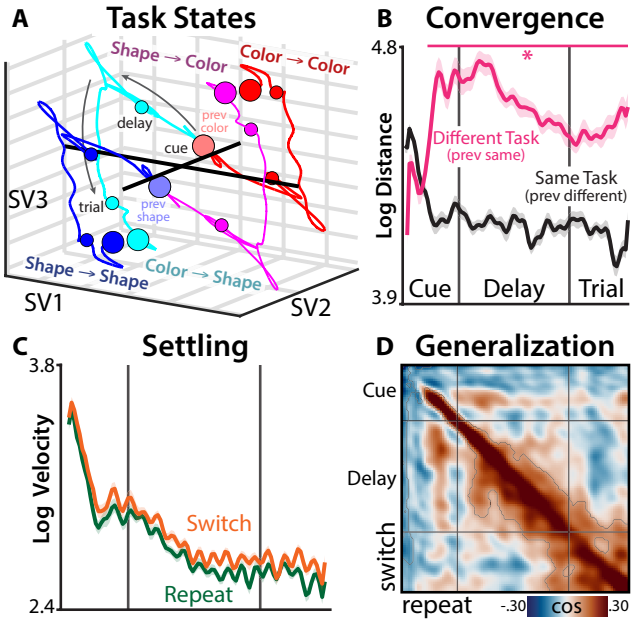

Fig. 2. Neural task attractors. **(A)** Simulated task states for each task, split by switch and repeat conditions. Task is contrast-coded, so tasks within the same switch condition are symmetrical. 'SV': singular vector. **(B)** Log Euclidean distances across switch and repeat conditions for the same task (black; e.g., cyan vs. blue in (A)) and different tasks (pink; e.g., cyan vs. red in (A)). **(C)** Log task state velocity, plotted for switch trials (orange) and repeat trials (green). First and last timepoint are excluded to accommodate edge effects in the temporal basis set. **(D)** Cosine similarity between task states across switch and repeat trials, at different temporal lags. Grid lines indicate the cue, delay, and trial periods. Asterisks and contours indicate p < .05 (TFCE-corrected; (52)).

on velocity-distance correlations: d = 0.97, p < .001, controlling for linear and quadratic effects of time).

We also tested the prediction that neural states should become more stable as they approach an attractor (53). We quantified state stability by computing the cosine similarity between task states at different temporal lags (54). Consistent with stabilization, we found that generalization grew over the epoch (broadening similarity in Fig. 2C), with significant similarity growth for lags up to 224 ms (Fig. S3).

Finally, we confirmed that these neural task states were behaviorally relevant, finding that participants had faster reaction times on trials in which their EEG had higher cosine similarity with the sensor-projected task state (summed over timesteps: d = -0.55, p = .0054; Fig. S4; Method 8). Together, these findings support key tenets of dynamic theories of task-switching, shared by both the Inertia and Reconfiguration hypotheses: the brain transitions into stable, task-specific states that promote good performance.

Task dynamics support Reconfiguration

We next examined whether the fitted models could discriminate between the Inertia and Reconfiguration hypotheses. These hypotheses make opposing predictions for how strongly cued task representations should propagate through the brain on switch trials relative to repeat trials. Inertia predicts weaker task propagation on switch trials, due to interference from the previous task. Reconfiguration predicts stronger task propaga-

tion on switch trials, due to the deployment of cognitive control.

To model the propagation of task representations, we used 'Lyapunov functions,' standard tools from control theory for analyzing linear dynamical systems (36). Lyapunov functions have been used in neuroscience for asymptotic 'controllability' analyses of how input energy will propagate through a neural system (37, 55, 56). Here, we quantified the time-resolved energy of task inputs, using the recursive form of the Lyapunov equation (57):

$$\begin{aligned} X_t^{\text{task}} &= A X_{t-1}^{\text{task}} A^\top + (B^{\text{task}} \mathbf{u}_{t-1}^{\text{task}})(B^{\text{task}} \mathbf{u}_{t-1}^{\text{task}})^\top \\ X_{t=1}^{\text{task}} &= (B_0^{\text{prevTask}} \mathbf{u}_0^{\text{prevTask}})(B_0^{\text{prevTask}} \mathbf{u}_0^{\text{prevTask}})^\top \end{aligned} \quad \textbf{(4)}$$

Consistent with the Reconfiguration hypothesis, we found that the total task energy ($\sum_{t=1}^{T} \log \text{trace}(X_t^{\text{task}})$) was stronger when participants switched tasks than when they repeated tasks (Fig. 3A; group-level t-test: d = 1.2, $p = 1.9 \times 10^{-7}$).

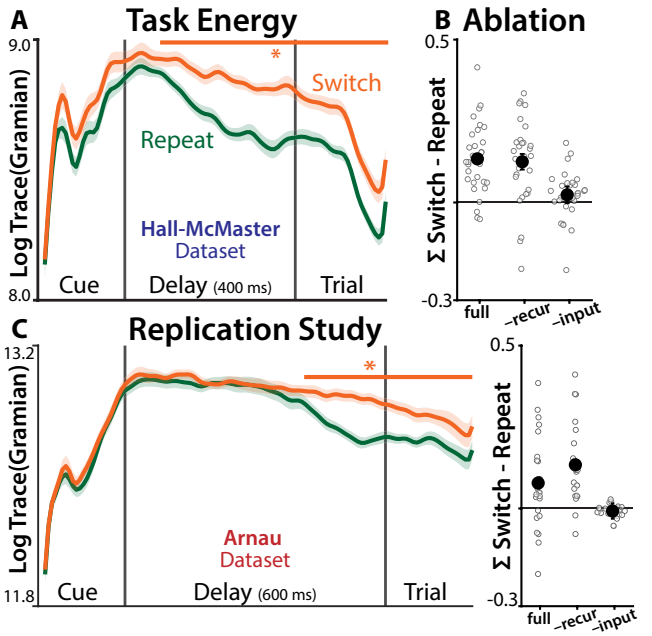

Fig. 3. Task Energy. **(A)** The log trace of the Task Gramian, separately for switch (orange) and repeat (green) conditions. Error bars indicate within-participant SEM; asterisks indicate $p < .05$ (TFCE). **(B)** Time-averaged contrast between task Gramians across switch and repeat conditions, computed under different model ablations. 'full': full model. '–recur': setting A and B_0 to zero. '–inputs': z-scoring $B^{\text{task}}\mathbf{u}_t^{\text{task}}$ at each timestep. Dots indicate participants, error bars indicate within-participant SEM. **(C)** Replication of (A) and (B) in an independent dataset.

While these differences in task energy are consistent with the Reconfiguration hypothesis, it is still possible that they emerge from the recurrent dynamics that are central to the Inertia account. To test this, we estimated task energy after ablating both the fitted recurrence (A) and the initial conditions (B_0^{prevTask}), components which together captured the passive decay of the previous task set. This ablation did not reduce the difference in task energy between switch conditions (Fig. 3B '-recur'; Fig. S5). To directly test the Reconfiguration hypothesis, we ablated time-varying task inputs (B^{task}). Here, we found that standardizing inputs at each timestep strongly reduced differences in task energy between switch conditions (Fig. 3B '-input'), further supporting the Reconfiguration account.

We confirmed the generalizability of these findings in an independent EEG dataset from Arnau et al. ((58); N=26, Method 1), which used shorter intertrial intervals (ITIs: 800 ms vs. 2200 ms; Method 2). Comparing short and long ITIs provides another way of testing for task set inertia: inertia should depend on the time from the previous task, unlike reconfiguration, which has been thought to depend only on the time since the cue (13). However, analysis of the Arnau dataset replicated the support for Reconfiguration. We found stronger, input-driven, task energy on switch trials (Fig. 3C; d = 0.68, p = .0068; between-study: d = 0.37, p = .17; BF_{null} = 1.69), and a replication of the other key results (Fig. S6).

These task state analyses provided consistent support for theories of neural task attractors in general, and the Reconfiguration hypothesis in particular. Time-varying inputs appear to move the brain into a stable task state, especially when switching between tasks, which we hypothesized reflects an underlying cognitive control process (22, 25).

Task switching in recurrent neural networks

To facilitate a stronger interpretation of the potential role for reconfiguration in task switching, we formalized the Inertia and Reconfiguration hypotheses using gated RNNs. These models provide full access to task representations in nonlinear dynamical systems trained to perform cognitive tasks.

While previous work has developed neural network models of inertia and reconfiguration, these have been limited by using hand-crafted structural mechanisms (21, 59–61), and have not adjudicated between these theories on the basis of their neural predictions. We addressed this by modeling task-switching hypotheses using different training curricula (22, 62), and then directly compared the networks' task-dependent dynamics to those observed in human EEG.

We trained RNNs on a 'context-dependent decision-making' task paralleling the EEG experiments (Method 9). Networks were trained on two kinds of trials. During 'Single Trials' they received a task cue, and then were presented with the task-relevant stimulus features after a delay period (Fig. 4A, upper; (63**?** , 64)). Whereas Single Trials had only one trial per sequence, in 'Double Trials' the networks performed two consecutive trials per sequence (Fig. 4A, lower), with an equal probability of switching and repeating tasks.

We used the amount of training on the Single and Double Trials to model the two task-switching hypotheses (Fig. 4B; Method 9). The Inertia hypothesis was captured by 'Single Networks,' which had 100% of their training on Single Trials, and were then tested on switch trials. We hypothesized that these networks would show task set inertia in the absence of learning how to actively switch between tasks. Reconfiguration was captured by 'Mixed Networks,' which were trained exclusively on Single Trials for the first 90% of training, followed by Double Trials during the last 10% of training. We hypothesized that this explicit experience with switching dur-

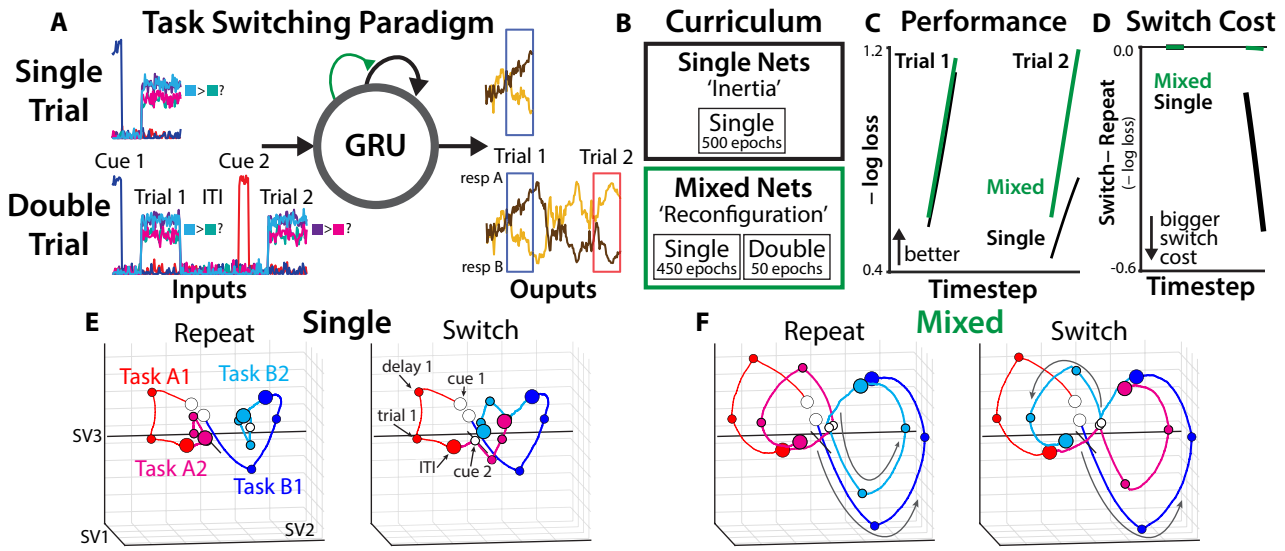

Fig. 4. Task dynamics in neural networks. **(A)** example of a 'single trial' (top) and a 'double trial' (bottom). **(B)** Single networks were only trained on single trials, whereas mixed networks had their final 10% of training on double trials. **(C)** Time-resolved loss over the course of double trial sequences, estimated after training. Mixed networks in green, single networks in black. Averaged over 512 networks; error bars are smaller than line width. **(D)** Difference in time-resolved loss for switch and repeat trials. **(E-F)** Hidden unit activity during double trials, projected to three dimensions with SVD. White dots are the initial timestep of each trial, colored dots are the final timestep of each trial. Shown here for the 'long-ITI' model (see Fig. 5, Fig. S11).

ing training would encourage the learning of active reconfiguration strategies.

Switching training produces flexible reconfiguration strategies

We first tested whether Single and Mixed Networks offer face-valid simulations of inertia and reconfiguration, training 512 RNNs with gated recurrent units (GRUs) under each curriculum (Method 9). To evaluate the networks' performance on Double Trials, we plotted the average loss at each timestep. We found that both networks had similar performance on the first trial of each pair, suggesting a similar capacity for task processing. (Fig. 4C). However, Mixed Networks showed much stronger performance on the second trial of each sequence. Mixed Networks' advantage on second trials was due in part to better switching than Single Networks, with a smaller difference in their loss between switch and repeat trials (Fig. 4D). This is consistent with Mixed Networks learning an adaptive reconfiguration strategy.

We assessed whether Mixed Networks' switching strategies were generalizable, as one would expect of flexible cognitive control. First, we tested how networks generalized to new trial sequences. Mixed Networks still performed better than Single Networks when tested on three trial sequences (Fig. S7), which neither had experienced during training. Next, we tested how networks generalized to new tasks. We trained another cohort of Mixed and Single networks on four tasks, with Mixed Networks learning to switch between only two of the four tasks (Method 9). After training, Mixed Networks had better switch-trial performance than Single Networks on the held-out task transitions (Fig. S8), consistent with learning a generalizable reconfiguration strategy.

We directly tested that the gating mechanisms were used for active reconfiguration, consistent with prior theories of cognitive control (65, 66). We found that ablating GRU 'update' gates during training substantially impaired their ability to learn from Double Trials (Fig. S9), consistent with a role for gating in their active reconfiguration.

Finally, we visualized Single and Mixed RNNs' hidden-unit activations using SVD (Fig. 4E-F; compare to Fig. 2A; Method 9). Whereas both networks exhibited similar first-trial dynamics, their second-trial dynamics sharply diverged. Single Networks appeared to move (unsuccessfully) between the two end states, whereas Mixed Networks exhibited robust preparatory dynamics that appeared to reproduce the first-trial trajectory.

Quantifying RNN reconfiguration using state space models

To compare the processing dynamics in Single and Mixed Networks against the human EEG data, we fit state-space models to RNNs' hidden unit activation during preparation for the second trial in each sequence (Method 10). As in the EEG analyses, SSMs with more latent factors than observation dimensions had the best cross-validation accuracy (next-step R^2_{CV} = .90 for 112 factors; Fig. S10), showcasing the ability of high-dimensional linear models to capture nonlinear systems.

Next, we compared the temporal generalization of task states between RNNs and EEG (Fig. 5A; Method 11). Consistent with the Reconfiguration hypothesis, we found that Mixed Networks had a strikingly similar generalization pattern to human EEG (cosine = [.52, .83], bootstrap 95% confidence interval; Method 11). Both Mixed Networks and EEG had robust generalization of task encoding across switch and repeat conditions. This was in stark contrast to Single Networks, which had poor correspondence to EEG (cosine = [-.39, .024]) and

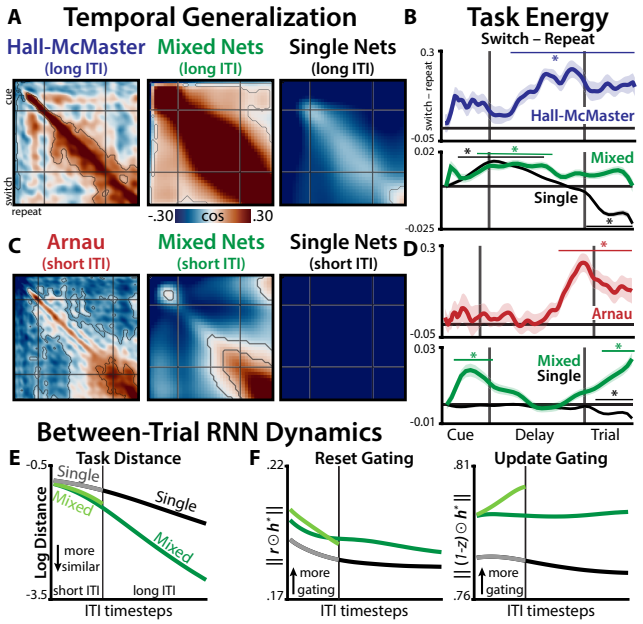

Fig. 5. RNN-EEG comparision. **(A)** Temporal generalization of task states in the first EEG dataset ('Hall-McMaster') and in RNN models with long intertrial intervals (ITIs). **(B)** Task energy contrasted between switch and repeat trials, plotted for the first EEG dataset (top) and long-ITI RNN models (bottom). **(C)** Temporal generalization and **(D)** task energy in the second EEG dataset ('Arnau') and in short-ITI RNNs; compare to (A) and (B). X-axis are not to scale (Hall-McMaster: 100 steps; Arnau: 125 steps; RNNs: 40 steps). **(E)** Euclidian distance between hidden units after each task. **(F)** Alignment between GRU gates (r, z) and a convergence target (e.g., following task A, $h_t^* = \hat{h}_{t+1}^{\text{taskB}}$; see Method 9). Contours and asterisks indicates $p < .05$ (TFCE).

negative alignment of task states across switch and repeat conditions.

We then tested whether RNNs reproduced the EEG signature of stronger task energy in switch trials than repeat trials (see: Fig. 3). As with generalization, we found divergent patterns between Single and Mixed Networks (Fig. 5B). Mixed networks, similar to EEG, had stronger task energy on switch trials ($d = 0.15$, $p = .00081$; EEG cosine = [.46, .92]), again supporting Reconfiguration. Single Networks, unlike EEG, did not show stronger task energy on switch trials and had poor alignment with EEG (summed energy: $d = 0.037$, $p = .41$; EEG cosine = [-.40, .21]).

The temporal generalization profile in the Arnau dataset differed from the HallMcMaster dataset (Fig. 5C, left), with negative alignment between early and late task states. This may indicate that there is greater task set inertia in the Arnau dataset, potentially due to shorter ITIs. However, these differences may instead be due to similar reconfiguration mechanisms operating over different ITIs. To test these alternatives, we trained Mixed Networks with shortened ITIs (Method 9). Short-ITI networks closely matched the Arnau dataset on temporal generalization (EEG cosine = [.81, .95]; Fig. 5C) and switch-elevated task energy ($d = 0.15$, $p = .001$; EEG cosine = [.20, .84]; Fig. 5D). This suggests that participants have processes during the ITI that are similar in Mixed Networks, but missing from classic theories of cue-based reconfiguration.

We explored whether Mixed Networks' processing during the ITI was dominated by control or inertia by comparing their hidden-unit activity to Single Networks. We found that Mixed Networks more rapidly transitioned into task-neutral states after each trial (Fig. 5E; Fig. S11), which they achieved through elevated control (gating-in Task B states in the ITI after task A; Fig. 5F; Method 9). In sum, Mixed Networks actively configure task-neutral states during the ITI, offering a new explanation of ITI-dependent task preparation centered on control.

Discussion

Psychologists have debated for decades about the latent processes underlying cognitive flexibility (1, 3, 11). To advance this debate, we quantified people's latent task processing through high-dimensional dynamical systems fit to two EEG datasets. We found evidence for task-specific neural attractors, a core prediction of dynamic switching theories. Comparing switch and repeat trials, we found stronger task energy when participants switched tasks, consistent with theories of active reconfiguration. Supporting this interpretation, neural networks that were trained to switch between tasks reproduced these neural signatures of reconfiguration. These analyses uncovered new evidence of the reconfiguration of neural task states, while also refining classic theories of task switching.

Task-optimized networks offered a new explanation for how task switching is influenced by ITIs, a major difference between the two EEG studies. During the ITI, networks gated-in a neutral task state (Fig. 5), like a tennis player returning to the center of the court after a volley. Short ITIs didn't provide networks enough time to reach this neutral state. These simulations offer both a new formalization and a potential reconciliation of task-switching theories. As in the inertia hypothesis, inter-trial configuration will depend on task similarity (67). As in the reconfiguration hypothesis, these dynamics are produced through control. While we found a more expansive role for control than has been assumed in classic theories, including on repeat trials, this is consistent with the ubiquitous dynamics observed during within-task attentional control (68).

One limitation of the current work is that we estimated time-varying inputs to the SSM, including after the cue presentation. These estimated task inputs may reflect an underlying control process, consistent with the analogous dynamics we found in gated RNNs. Optimal feedback control, the gold-standard model of motor control (69, 70), offers a natural process model for control over dynamic neural systems. Notably, preparing for actions (71), for known tasks (72), and, here, for unknown tasks all appear to involve configuring the optimal initial conditions for dynamic processing. Optimal control theory may offer a prime candidate for unifying accounts of this goal-directed control over the neural state space (55, 73, 74).

For cognitive neuroscience more generally, there is high demand for better source reconstruction from MEG and EEG recordings. SSMs provide a powerful method for localizing latent dynamics (75–77), offering Bayesian estimates of source activity from lagged measurements. SSMs can also allow researchers to quantify the spread of information through effective networks (78, 79), offering a clearer view of the distributed neural systems that enable flexible cognition (80, 81).

ACKNOWLEDGEMENTS

We are extremely grateful to Sam Hall-McMaster and Stefan Arnau for making their datasets public and providing additional support. Thanks to Christopher Langdon for sharing PyTorch code for context-dependent decision-making tasks, Caroline Jahn for feedback on an early draft of the manuscript, and Jonathan Pillow for his state-space tutorial document. Thanks to the Cohen and Daw labs for their input throughout this project. This paper is dedicated to the late Mark Stokes.

HR was supported by the C.V. Starr Postdoctoral Fellowship. AJ was supported by the Google PhD fellowship. JDC was supported by a Vannevar Bush Faculty Fellowship administered by the Office of Naval Research.

Bibliography

1. Stephen Monsell. Task switching. *Trends Cogn. Sci.*, 7(3):134–140, March 2003.
2. Andrea Kiesel, Marco Steinhauser, Mike Wendt, Michael Falkenstein, Kerstin Jost, Andrea M Philipp, and Iring Koch. Control and interference in task switching—a review. *Psychol. Bull.*, 136(5):849–874, September 2010.
3. André Vandierendonck, Baptist Liefooghe, and Frederick Verbruggen. Task switching: interplay of reconfiguration and interference control. *Psychol. Bull.*, 136(4):601–626, July 2010.
4. Sebastian Musslick and Jonathan D Cohen. Rationalizing constraints on the capacity for cognitive control. *Trends Cogn. Sci.*, 0(0), July 2021.
5. Naomi P Friedman and Akira Miyake. Unity and diversity of executive functions: Individual differences as a window on cognitive structure. *Cortex*, 86:186–204, January 2017.
6. Nicholas J Cepeda, Arthur F Kramer, and Jessica C M Gonzalez de Sather. Changes in executive control across the life span: Examination of task-switching performance. *Dev. Psychol.*, 37(5):715–730, 2001.
7. Mark Steyvers, Guy E Hawkins, Frini Karayanidis, and Scott D Brown. A large-scale analysis of task switching practice effects across the lifespan. *Proc. Natl. Acad. Sci. U. S. A.*, August 2019.
8. Mark J Millan, Yves Agid, Martin Brüne, Edward T Bullmore, Cameron S Carter, Nicola S Clayton, Richard Connor, Sabrina Davis, Bill Deakin, Robert J DeRubeis, Bruno Dubois, Mark A Geyer, Guy M Goodwin, Philip Gorwood, Thérèse M Jay, Marian Joëls, Isabelle M Mansuy, Andreas Meyer-Lindenberg, Declan Murphy, Edmund Rolls, Bernd Saletu, Michael Spedding, John Sweeney, Miles Whittington, and Larry J Young. Cognitive dysfunction in psychiatric disorders: characteristics, causes and the quest for improved therapy. *Nat. Rev. Drug Discov.*, 11(2):141–168, February 2012.
9. Hannah R Snyder. Major depressive disorder is associated with broad impairments on neuropsychological measures of executive function: a meta-analysis and review. *Psychol. Bull.*, 139(1):81–132, January 2013.
10. A Allport, E A Styles, and S L Hsieh. Shifting intentional set - exploring the dynamic control of tasks. In *ATTENTION AND PERFORMANCE XV: CONSCIOUS AND NONCONSCIOUS INFORMATION PROCESSING*, pages 421–452. MIT Press, London, England, 1994.
11. Glenn Wylie and Alan Allport. Task switching and the measurement of "switch costs". *Psychol. Res.*, 63(3):212–233, August 2000.
12. Robert D Rogers and Stephen Monsell. Costs of a predictible switch between simple cognitive tasks. *J. Exp. Psychol. Gen.*, 124(2):207, 1995.
13. Nachshon Meiran. Reconfiguration of processing mode prior to task performance. *J. Exp. Psychol. Learn. Mem. Cogn.*, 22(6):1423–1442, November 1996.
14. Nick Yeung and Stephen Monsell. Switching between tasks of unequal familiarity: the role of stimulus-attribute and response-set selection. *J. Exp. Psychol. Hum. Percept. Perform.*, 29(2):455–469, April 2003.
15. Nick Yeung, Leigh E Nystrom, Jessica A Aronson, and Jonathan D Cohen. Between-task competition and cognitive control in task switching. *J. Neurosci.*, 26(5):1429–1438, February 2006.
16. Florian Schmitz and Andreas Voss. Decomposing task-switching costs with the diffusion model. *J. Exp. Psychol. Hum. Percept. Perform.*, 38(1):222–250, February 2012.
17. Lasse S Loose, David Wisniewski, Marco Rusconi, Thomas Goschke, and John-Dylan Haynes. Switch-independent task representations in frontal and parietal cortex. *J. Neurosci.*, 37(33):8033–8042, August 2017.
18. Lei Qiao, Lijie Zhang, Antao Chen, and Tobias Egner. Dynamic trial-by-trial recoding of task-set representations in the frontoparietal cortex mediates behavioral flexibility. *J. Neurosci.*, 37(45):11037–11050, November 2017.
19. Jonathan A Michaels, Benjamin Dann, and Hansjörg Scherberger. Neural population dynamics during reaching are better explained by a dynamical system than representational tuning. *PLoS Comput. Biol.*, 12(11):e1005175, November 2016.
20. Mark M Churchland and Krishna V Shenoy. Preparatory activity and the expansive null-space. *Nat. Rev. Neurosci.*, 25(4):213–236, April 2024.
21. Salva Ardid and Xiao-Jing Wang. A tweaking principle for executive control: neuronal circuit mechanism for rule-based task switching and conflict resolution. *J. Neurosci.*, 33(50):19504–19517, December 2013.
22. Sebastian Musslick, Seong Jun Jang, Michael Shvartsman, Amitai Shenhav, and Jonathan D Cohen. Constraints associated with cognitive control and the stability-flexibility dilemma. In *CogSci*. shenhavlab.org, 2018.
23. Paul I Jaffe, Russell A Poldrack, Robert J Schafer, and Patrick G Bissett. Modelling human behaviour in cognitive tasks with latent dynamical systems. *Nature Human Behaviour*, pages 1–15, January 2023.
24. Sam Hall-McMaster, Paul S Muhle-Karbe, Nicholas E Myers, and Mark G Stokes. Reward boosts neural coding of task rules to optimize cognitive flexibility. *J. Neurosci.*, 39(43):8549–8561, October 2019.
25. J D Cohen, K Dunbar, and J L McClelland. On the control of automatic processes: a parallel distributed processing account of the stroop effect. *Psychol. Rev.*, 97(3):332–361, July 1990.
26. J D Wallis, K C Anderson, and E K Miller. Single neurons in prefrontal cortex encode abstract rules. *Nature*, 411(6840):953–956, June 2001.
27. Alexandra Woolgar, Adam Hampshire, Russell Thompson, and John Duncan. Adaptive coding of task-relevant information in human frontoparietal cortex. *J. Neurosci.*, 31(41):14592–14599, October 2011.
28. E N Brown, L M Frank, D Tang, M C Quirk, and M A Wilson. A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *J. Neurosci.*, 18(18):7411–7425, September 1998.
29. Anne C Smith and Emery N Brown. Estimating a state-space model from point process observations. *Neural Comput.*, 15(5):965–991, May 2003.
30. Jakob H Macke, Lars Büsing, John P Cunningham, Byron M Yu Ece, Krishna V Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. In *Advances in neural information processing systems*, 2011.
31. Adrian Valente, Srdjan Ostojic, and Jonathan W Pillow. Probing the relationship between latent linear dynamical systems and low-rank recurrent neural network models. *Neural Comput.*, 34(9):1871–1892, August 2022.
32. Paul K LaFosse, Zhishang Zhou, Jonathan F O'Rawe, Nina G Friedman, Victoria M Scott, Yanting Deng, and Mark H Histed. Single-cell optogenetics reveals attenuation-by-suppression in visual cortical neurons. *bioRxivorg*, page 2023.09.13.557650, May 2024.
33. Erfan Nozari, Maxwell A Bertolero, Jennifer Stiso, Lorenzo Caciagli, Eli J Cornblath, Xiaosong He, Arun S Mahadevan, George J Pappas, and Dani S Bassett. Macroscopic resting-state brain dynamics are best described by linear models. *Nat. Biomed. Eng.*, pages 1–17, December 2023.
34. Joana Soldado-Magraner, Valerio Mante, and Maneesh Sahani. Inferring context-dependent computations through linear approximations of prefrontal cortex dynamics. *Sci. Adv.*, 10(51):eadl4743, December 2024.
35. Aditi Jha, Diksha Gupta, Carlos D Brody, and Jonathan W Pillow. Disentangling the roles of distinct cell classes with cell-type dynamical systems. *bioRxiv*, page 2024.07.08.602520, July 2024.
36. Robert L Williams and Douglas A Lawrence. *Linear state-space control systems*. John Wiley & Sons, Nashville, TN, January 2007.
37. E Tang and D S Bassett. Colloquium: Control of dynamics in brain networks. *Rev. Mod. Phys.*, 2018.
38. D Lehmann, H Ozaki, and I Pal. EEG alpha map series: brain micro-states by space-oriented adaptive segmentation. *Electroencephalogr. Clin. Neurophysiol.*, 67(3):271–288, September 1987.
39. Scott Linderman, Annika Nichols, David Blei, Manuel Zimmer, and Liam Paninski. Hierarchical recurrent state space models reveal discrete and continuous dynamics of neural activity in C. elegans. *bioRxiv*, page 621540, April 2019.
40. Chetan Gohil, Oliver Kohl, Rukuang Huang, Mats W J van Es, Oiwi Parker Jones, Laurence T Hunt, Andrew J Quinn, and Mark W Woolrich. Dynamic network analysis of electrophysiological task data. *Imaging Neuroscience*, 2:1–19, July 2024.
41. Florian Hess, Zahra Monfared, Manuel Brenner, and Daniel Durstewitz. Generalized teacher forcing for learning chaotic dynamics. In *International Conference on Machine Learning*, pages 13017–13049. PMLR, July 2023.
42. Matthijs Pals, A Erdem Sağtekin, Felix Pei, Manuel Gloeckler, and Jakob H Macke. Inferring stochastic low-rank recurrent neural networks from neural data. *arXiv [cs.LG]*, June 2024.
43. Zoubin Ghahramani, Geo Rey, and E Hinton. Parameter estimation for linear dynamical systems. Techinical Report, 1996.
44. Kevin P Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, London, England, 2023.
45. Harrison Ritz. harrisonritz/StateSpaceAnalysis.jl: v0.2.0, 2024.
46. Mark M Churchland, Byron M Yu, Maneesh Sahani, and Krishna V Shenoy. Techniques for extracting single-trial activity patterns from large-scale neural recordings. *Curr. Opin. Neurobiol.*, 17(5):609–618, October 2007.
47. Camden J MacDowell, Brandy A Briones, Michael J Lenzi, Morgan L Gustison, and Timothy J Buschman. Differences in the expression of cortex-wide neural dynamics are related to behavioral phenotype. *Curr. Biol.*, February 2024.
48. Klaas Enno Stephan, Will D Penny, Jean Daunizeau, Rosalyn J Moran, and Karl J Friston. Bayesian model selection for group studies. *Neuroimage*, 46(4):1004–1017, July 2009.
49. Steven L Brunton, Marko Budišić, Eurika Kaiser, and J Nathan Kutz. Modern koopman theory for dynamical systems. *arXiv [math.DS]*, February 2021.
50. Daniel Durstewitz, Georgia Koppe, and Max Ingo Thurm. Reconstructing computational system dynamics from neural data with recurrent neural networks. *Nat. Rev. Neurosci.*, pages 1–18, October 2023.
51. Olof Staffans. *Encyclopedia of mathematics and its applications: Well-posed linear systems series number 103*. Cambridge University Press, Cambridge, England, October 2009.
52. Stephen M Smith and Thomas E Nichols. Threshold-free cluster enhancement: addressing problems of smoothing, threshold dependence and localisation in cluster inference. *Neuroimage*, 44(1):83–98, January 2009.
53. Jake P Stroud, John Duncan, and Máté Lengyel. The computational foundations of dynamic coding in working memory. *Trends Cogn. Sci.*, 0(0), April 2024.
54. Fabrice Luyckx, Hamed Nili, Bernhard Spitzer, and Christopher Summerfield. Neural structure mapping in human probabilistic reward learning. *Elife*, 8, March 2019.
55. Ta-Chu Kao and Guillaume Hennequin. Neuroscience out of control: control-theoretic perspectives on neural circuit dynamics. *Curr. Opin. Neurobiol.*, 58:122–129, September 2019.
56. Clay B Holroyd. The controllosphere: The neural origin of cognitive effort. *Psychol. Rev.*, February 2024.
57. Corbin Klett, Matthew Abate, Yongeun Yoon, Samuel Coogan, and Eric Feron. Bounding the state covariance matrix for switched linear systems with noise. In *2020 American Control Conference (ACC)*, pages 2876–2881. IEEE, July 2020.
58. Stefan Arnau, Nathalie Liegel, and Edmund Wascher. Frontal midline theta power during the cue-target-interval reflects increased cognitive effort in rewarded task-switching. *Cortex*, 180:94–110, November 2024.
59. Sam J Gilbert and Tim Shallice. Task switching: a PDP model. *Cogn. Psychol.*, 44(3):297–337, May 2002.
60. Joshua W Brown, Jeremy R Reynolds, and Todd S Braver. A computational model of fractionated conflict-control mechanisms in task-switching. *Cogn. Psychol.*, 55(1):37–85, August 2007.

61. Seth A Herd, Randall C O'Reilly, Tom E Hazy, Christopher H Chatham, Angela M Brant, and Naomi P Friedman. A neural network model of individual differences in task switching abilities. *Neuropsychologia*, 62:375–389, September 2014.
62. J D Cohen, D Servan-Schreiber, and J L McClelland. A parallel distributed processing approach to automaticity. *Am. J. Psychol.*, 105(2):239–269, 1992.
63. Markus Siegel, Tobias H Donner, and Andreas K Engel. Spectral fingerprints of large-scale neuronal interactions. *Nat. Rev. Neurosci.*, 13(2):121–134, January 2012.
64. V Mante, D Sussillo, K V Shenoy, and W T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, 2013.
65. Todd S Braver and Jonathan D Cohen. Chapter 19 dopamine, cognitive control, and schizophrenia: the gating model. In *Progress in Brain Research*, volume 121 of *Progress in brain research*, pages 327–349. Elsevier, 1999.
66. Randall C O'Reilly and Michael J Frank. Making working memory work: a computational model of learning in the prefrontal cortex and basal ganglia. *Neural Comput.*, 18(2):283–328, February 2006.
67. Catherine M Arrington, Erik M Altmann, and Thomas H Carr. Tasks of a feather flock together: similarity effects in task switching. *Mem. Cognit.*, 31(5):781–789, July 2003.
68. Wery P M van den Wildenberg, Scott A Wylie, Birte U Forstmann, Borís Burle, Thierry Hasbroucq, and K Richard Ridderinkhof. To head or to heed? beyond the surface of selective action inhibition: a review. *Front. Hum. Neurosci.*, 4:222, December 2010.
69. Emanuel Todorov and Michael I Jordan. Optimal feedback control as a theory of motor coordination. *Nat. Neurosci.*, 5(11):1226–1235, November 2002.
70. Reza Shadmehr and John W Krakauer. A computational neuroanatomy for motor control. *Exp. Brain Res.*, 185(3):359–381, March 2008.
71. Mark M Churchland, John P Cunningham, Matthew T Kaufman, Stephen I Ryu, and Krishna V Shenoy. Cortical preparatory activity: representation of movement or first cog in a dynamical machine? *Neuron*, 68(3):387–400, November 2010.
72. Harrison Ritz and Amitai Shenhav. Humans reconfigure target and distractor processing to address distinct task demands. *Psychol. Rev.*, August 2023.
73. Vivek R Athalye, Jose M Carmena, and Rui M Costa. Neural reinforcement: re-entering and refining neural dynamics leading to desirable outcomes. *Curr. Opin. Neurobiol.*, 60:145–154, December 2019.
74. Harrison Ritz, Xiamin Leng, and Amitai Shenhav. Cognitive control as a multivariate optimization problem. *J. Cogn. Neurosci.*, 34(4):569–591, March 2022.
75. K J Friston, L Harrison, and W Penny. Dynamic causal modelling. *Neuroimage*, 19(4): 1273–1302, August 2003.
76. Andreas Galka, Okito Yamashita, Tohru Ozaki, Rolando Biscay, and Pedro Valdés-Sosa. A solution to the dynamical inverse problem of EEG generation using spatiotemporal kalman filtering. *Neuroimage*, 23(2):435–453, October 2004.
77. Elvira Pirondini, Behtash Babadi, Gabriel Obregon-Henao, Camilo Lamus, Wasim Q Malik, Matti S Hamalainen, and Patrick L Purdon. Computationally efficient algorithms for sparse, dynamic solutions to the EEG source localization problem. *IEEE Trans. Biomed. Eng.*, 65 (6):1359–1372, June 2018.
78. Gang Yan, Petra E Vértes, Emma K Towlson, Yee Lian Chew, Denise S Walker, William R Schafer, and Albert-László Barabási. Network control principles predict neuron function in the caenorhabditis elegans connectome. *Nature*, 550(7677):519–523, October 2017.
79. Matthew S Creamer, Andrew M Leifer, and Jonathan W Pillow. Bridging the gap between the connectome and whole-brain activity in*C. elegans*. *bioRxiv*, page 2024.09. 22.614271, September 2024.
80. P S Goldman-Rakic. Topography of cognition: parallel distributed networks in primate association cortex. *Annu. Rev. Neurosci.*, 11:137–156, 1988.
81. E K Miller and J D Cohen. An integrative theory of prefrontal cortex function. *Annu. Rev. Neurosci.*, 24:167–202, 2001.
82. Harrison Ritz and Amitai Shenhav. Orthogonal neural encoding of targets and distractors supports multivariate cognitive control. *Nature Human Behaviour*, pages 1–17, March 2024.
83. Alexander Walther, Hamed Nili, Naveed Ejaz, Arjen Alink, Nikolaus Kriegeskorte, and Jörn Diedrichsen. Reliability of dissimilarity measures for multi-voxel pattern analysis. *Neuroimage*, 137:188–200, August 2016.
84. e Holmes, Elizabeth, j Ward, Eric, and Kellie Wills. MARSS: Multivariate autoregressive state-space models for analyzing time-series data. *R J.*, 4(1):11, 2012.
85. Gavin Smith, João de Freitas, Tony Robinson, and Mahesan Niranjan. Speech modelling using subspace and EM techniques. *Advances in Neural Information Processing Systems*, 12, 1999.
86. Iris R Stone, Yotam Sagiv, Il Memming Park, and Jonathan W Pillow. Spectral learning of bernoulli linear dynamical systems models. *arXiv [stat.ML]*, March 2023.
87. W E Larimore. Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE, 1990.
88. Benedikt V Ehinger and Olaf Dimigen. Unfold: an integrated toolbox for overlap correction, non-linear modeling, and regression-based EEG analysis. *PeerJ*, 7(e7838):e7838, October 2019.
89. D R Cox and Nanny Wermuth. A comment on the coefficient of determination for binary responses. *Am. Stat.*, 46(1):1–4, February 1992.
90. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. *arXiv [cs.LG]*, December 2019.
91. G E Hinton and R R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, July 2006.
92. Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. November 2017.
93. J-R King and S Dehaene. Characterizing the dynamics of mental representations: the temporal generalization method. *Trends Cogn. Sci.*, 18(4):203–210, April 2014.
94. R E Kalman. Lectures on controllability and observability. In *Controllability and Observability*, pages 1–149. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
95. Daniel B Ehrlich and John D Murray. Geometry of neural computation unifies working memory and planning. *Proc. Natl. Acad. Sci. U. S. A.*, 119(37):e2115610119, September 2022.
96. Christopher Langdon and Tatiana A Engel. Latent circuit inference from heterogeneous neural responses during cognitive tasks. *Nat. Neurosci.*, pages 1–11, February 2025.
97. Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv [cs.CL]*, June 2014.
98. Robert Oostenveld, Pascal Fries, Eric Maris, and Jan-Mathijs Schoffelen. FieldTrip: Open source software for advanced analysis of MEG, EEG, and invasive electrophysiological data. *Comput. Intell. Neurosci.*, 2011(1):156869, January 2011.
99. Arnaud Delorme and Scott Makeig. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neurosci. Methods*, 134 (1):9–21, March 2004.
100. Fabio Crameri, Grace E Shephard, and Philip J Heron. The misuse of colour in science communication. *Nat. Commun.*, 11(1):5444, October 2020.

Methods

1. EEG Sample

A. Hall-McMaster Dataset. The first EEG dataset was originally published by Hall-McMaster and colleagues (24), and was made open-access at https://osf.io/kuzye/. Thirty participants (18-35 years old; 19 females), with normal or corrected-to-normal vision and no history of neurological or psychological disorders, underwent an EEG recording session during task performance. This study was approved by the Central University Research Ethics Committee at the University of Oxford, with all participants providing informed consent. See (24) for the full description.

B. Arnau Dataset. The second EEG dataset was originally published by Arnau and colleagues (58), and was made open-access at https://osf.io/ndgst/. This dataset consists of twenty-six participants (23 female, mean(SD) age 21.65(2.15) years, with normal or corrected-to-normal vision and no history of neurological or psychological disorders, and verified normal color vision. This study was The study was approved by the ethics committee of the Leibniz Research Centre for Working Environment and Human Factors, Dortmund, with all participants providing informed consent. See (58) for the full description.

2. EEG Task

A. Hall-McMaster dataset. Participants performed a cued task-switching experiment during the EEG recording. The major goal of the original study was to investigate the role of incentives on task encoding. This study focused on more traditional switching because this directly relates to the core theoretical questions about task set inertia and active reconfiguration.

In the original experiment, the full trial structure involved a cue for high vs low rewards for correct performance (800 ms, 400 ms delay), a cue for the shape vs color task (200 ms, 400 ms delay), a trial stimulus consisting of a square or circle colored in yellow or blue (min(RT, 1400 ms)), reward feedback (200 ms), and an inter-trial interval (1000-1400 ms). Participants' task was to respond to the trial stimulus according to the cued rule: if the task was 'shape', respond with a key press based on the shape; if the task was 'color', respond with a key press based on the color. Participants practiced the task until they reached a 70% accuracy criterion, and then performed 10 blocks of 65 trials in the main task. Conditions were balanced within each block. For the analyses, we included trials 1) that were not the first trial of a block, 2) where the current and the previous trial was accurate, 3) where the current RT was longer than 200 ms, and 4) that were not identified as containing artifacts during preprocessing. After preprocessing, we split up the experimental blocks into training and test sets, with Mean(SD) = 416(63) training trials and 47(9) testing trials. EEG data were recorded with 61 Ag/AgCl sintered electrodes (EasyCap; 10-10 layout (Fieldtrip template: elec1010)), a NeuroScan SynAmps RT amplifier, and Curry 7 acquisition software. See (24) for full description.

B. Arnau Dataset. Participants performed a cued task-switching experiment during the EEG recording. On each trial, participants saw a cue for the tilt vs color task for 200 ms, followed by 600 ms delay. In the trial phase, participants saw an array of gratings, and had to report the tilt or the color of a singleton, depending on the cued task. Participants had up to 1200 ms to respond, and then there was an inter-trial interval (800-1000 ms). As in Hall-McMaster, participants practiced the task before the main experiment, which consisted of eight experimental blocks with 256 trials each. This experiment also investigated the role of motivatoin on task switching, with a block-level incentive manipulation. However, this similarity is coincidental, due to motivation being an area of focus in recent high-quality experiments on cognitive control and task switching.

Here, EEG data were recorded using 128 passive Ag/AgCl electrodes (Easycap GmbH, Herrsching, Germany) with a NeurOne Tesla AC-amplifier (Bittium Biosignals Ltd., Kuopio, Finland). The EEG were preprocessed using EEGLab scripts modified from the original experiment in three ways. First, we resampled the data at 125 Hz (as in Hall-McMaster). Second, we used Blackman zero-phase FIR filtering, rather than Butterworth IIR filtering. We separately high-pass filtered at 0.1 Hz and low-pass filtered at 30 Hz, using the default EEGLab filter order for each frequency. Finally, we used more lenient inclusion criteria for the ICA auto-rejection procedure, as some participants had the majority of their components removed. We used the same performance-based inclusion criteria as in Hall-McMaster. After preprocessing, we split up the experimental blocks into training and test sets, with Mean(SD) = 676(147) training trials and 142(31) testing trials. See (58) for full details on the task and original preprocessing procedure.

3. Encoding Geometry Analysis

In the traditional multivariate analyses, we used Encoding Geometry Analysis (EGA; (82)) to quantify task encoding. At each timestep within the epoch, we fit a general linear model (GLM) to the across-trial voltage in the EEG electrodes. The regression model included an intercept, the current task, the previous task, cue identity (with separate regressor contrasting the cues within each task), cue repetitions vs alternations for repeat trials, the main effect of switch vs repeat, the current trial's reaction time, and the previous trial's reaction time. We fit this GLM separately to even and odd runs, and then performed multivariate spatial pre-whitening on the regression estimates (83). We tested the reliability of regression weights by correlating spatial patterns across even and odd folds, which is closely related to out-of-sample predictive accuracy (82).

4. Multiple comparisons corrections

To correct for multiple comparisons over time while accounting for temporal autocorrelation, we used threshold-free cluster enhancement through-out (TFCE; (52)) with a set of EEG-optimized parameters (H=2, C=1; from (52); 1,000 permutations for temporal generalization and 10,000 permutations for traces).

5. Latent Linear Dynamical System

A. Generative Model. We developed a toolkit called StateSpaceAnalysis for fitting linear-Gaussian state space models (SSMs) to neuroimaging data. Our goal was to test how effectively SSMs could capture rich neuroimaging data, balancing the predictive power of machine learning approaches with the interpretability offered by linear-Gaussian assumptions. To provide the speed and precision needed for estimating the parameters of these large latent variable models, we developed this package using the open-source coding language Julia, available at `https://github.com/harrisonritz/StateSpaceAnalysis.jl` (45).

The generative model for the analysis is a partially observable autoregressive process: a linear dynamical system from which we can only make noisy measurements (or, equivalently, that only produces noisy emissions of its underlying state). This autoregressive process consists of a vector of latent factors ($\mathbf{x}$), and a discrete-time difference equation that describes how they evolve:

$$\begin{aligned} \mathbf{x}_t &= A\mathbf{x}_{t-1} + B\mathbf{u}_{t-1} + \mathbf{w}_t \\ \mathbf{x}_{t=1} &= B_0\mathbf{u}_0 + \mathbf{w}_1 \end{aligned}$$

Latent factors evolve according to their recurrent dynamics ($A\mathbf{x}$), the influence of known inputs ($B\mathbf{u}$), and process noise ($\mathbf{w}_t \sim \mathcal{N}(0, W)$). The initial conditions depend on the trial conditions ($B_0\mathbf{u}_0$) and initial uncertainty ($\mathbf{w}_1 \sim \mathcal{N}(0, W_1)$).

At each timestep, we get a noisy observation of these latent factors:

$$\mathbf{y}_t = C\mathbf{x}_t + \mathbf{v}_t$$

Observations are projections of the latent factors ($C\mathbf{x}$) corrupted by observation noise ($\mathbf{v}_t \sim \mathcal{N}(0, V)$). Note that process noise is carried forward in time through the recurrent dynamics, unlike observation noise.

B. Expectation-Maximization. A major benefit of SSMs are that they allow closed-form methods for Bayesian inference of the underlying latent state, and closed-form marginal likelihood of the observed data, $P(\mathbf{y}_{1:T} \mid \Theta)$ where Θ represents the model parameters. This is achieved using standard inferential tools from control theory: Kalman filtering (inferring $\mathbf{x}_t$ from past observations) and Rauch–Tung–Striebel (RTS) smoothing (inferring $\mathbf{x}_t$ using both past and future observations).

While the marginal likelihood from Kalman filtering would allow us to directly fit the parameters through techniques like maximum likelihood estimation, in practice it is usually much more efficient to use expectation-maximization (EM; (43, 44, 84)). EM maximizes the expected lower bound of the marginal posterior (often called the evidence lower bound or ELBO) by alternating between two steps. In the E-step, we estimate the latent variables using the RTS smoother. In the M-step, we find a point-estimate of the parameters that maximize the ELBO, given the estimated latent states. With priors on the parameters, this optimizes the mode of their posterior density. The Linear-Gaussian assumptions of this model considerably simplify this procedure by allowing us to work with the sufficient statistics of the estimated latent state (i.e., their mean and covariance).

B.1. Marginal Log Posterior.

$$\begin{aligned} \mathcal{L} = \mathbb{E}[\log\mathcal{N}(\mathbf{x}_1; B_0\mathbf{u}_0, W_1)] + \mathbb{E}[\log\mathcal{N}(\mathbf{x}_t; A\mathbf{x}_{t-1} + B\mathbf{u}_{t-1}, W)] + \mathbb{E}[\log\mathcal{N}(\mathbf{y}_t; C\mathbf{x}_t, V)] \\ + \log\mathcal{N}(A; 0, \Lambda_A) + \log\mathcal{N}(B; 0, \Lambda_B) + \log\mathcal{N}(C; 0, \Lambda_C) \end{aligned}$$

B.2. E-Step. Using the filter-smoother algorithm, we estimated the posterior mean ($\mathbf{m}_t$) and covariance (Σ_t) of the latent state across timesteps (t) and epochs (n), generating a set of sufficient statistics:

$$M_{(t_1,t_2)} = \sum_{n=1}^{N}\sum_{t=t_1}^{t2} \mathbf{m}_{n,t}\mathbf{m}_{n,t}^\top + \Sigma_t \qquad U_u = \sum_{n=1}^{N}\sum_{t=1}^{T-1} \mathbf{u}_{n,t}\mathbf{u}_{n,t}^\top$$

$$M_\Delta = \sum_{n=1}^{N}\sum_{t=2}^{T} \mathbf{m}_{n,t-1}\mathbf{m}_{n,t}^\top + \Sigma_{t-1,t} \qquad U_m = \sum_{n=1}^{N}\sum_{t=1}^{T-1} \mathbf{m}_{n,t}\mathbf{u}_{n,t}^\top$$

$$U_\Delta = \sum_{n=1}^{N}\sum_{t=1}^{T-1} \mathbf{m}_{n,t+1}\mathbf{u}_{n,t}^\top$$

$$Y_y = \sum_{n=1}^{N}\sum_{t=1}^{T} \mathbf{y}_{n,t}\mathbf{y}_{n,t}^\top \qquad U_{u_0} = \sum_{n=1}^{N} \mathbf{u}_{n,0}\mathbf{u}_{n,0}^\top$$

$$Y_\Delta = \sum_{n=1}^{N}\sum_{t=1}^{T} \mathbf{m}_{n,t}\mathbf{y}_{n,t}^\top \qquad U_{\Delta_0} = \sum_{n=1}^{N} \mathbf{m}_{n,1}\mathbf{u}_{n,0}^\top$$

Note that we can use the same smoothed covariance estimates (Σ) for all epochs, substantially speeding up the computation.

B.3. M-Step. We then use the computed sufficient statistics to update the parameters using a maximization procedure similar to ordinary least squares. Note that we include priors (Λ) on all of the dynamics matrices (scaled identity matrices).

For the dynamics terms:

$$\begin{bmatrix} A \\ B \end{bmatrix} \leftarrow \begin{bmatrix} M_{(1,T-1)} + \Lambda_A & U_m^\top \\ U_m & U_{(1,T-1)} + \Lambda_B \end{bmatrix}^{-1} \begin{bmatrix} M_\Delta \\ U_\Delta \end{bmatrix}$$

$$C \leftarrow (M_{(1,T)} + \Lambda_C)^{-1} Y_\Delta$$

$$B_0 \leftarrow (U_{u_0} + \Lambda_{B_0})^{-1} U_{\Delta_0}$$

For the covariance terms:

$$W \leftarrow \frac{1}{N(T-1) - D_x}(M_{(2,T)} + AM_{(1,T-1)}A^\top + BU_mB^\top - AM_\Delta - M_\Delta^\top A - BU_\Delta - U_\Delta^\top B + BU_mA^\top + AU_m^\top B^\top + A\Lambda_A A^\top + B\Lambda_B B^\top)$$

$$B \leftarrow \frac{1}{NT - D_y}(Y_y + CM_{(1,T)}C^\top - CY_\Delta - Y_\Delta^\top C + C\Lambda_C C^\top)$$

$$W_0 \leftarrow \frac{1}{N - D_x}(M_{(1,1)} + B_0U_{u_0}B_0^\top - B_0U_{\Delta_0} - U_{\Delta_0}^\top B_0 + B_0\Lambda_{B_0}B_0^\top)$$

With D_x and D_y indicates the numbers of factors and observation dimensions, respectively. Note that the estimator of W_0 is a little unusual, but is equivalent to (43) when the only input is an intercept term. Alternating between these E-steps and M-steps monotonically improves the ELBO towards a local optimum (i.e., depended on the initialization; see 'Subspace Identification' below). This toolkit is inspired by the `SSM` and `dynamax` python packages, but had several optimizations specifically tailored for linear-Gaussian systems (e.g., positive definite matrix classes).

C. Subspace Identification. We found that good initialization of the parameters was critical to the effectiveness and stability of the model-fitting procedure. We initialized the parameters using a procedure called subspace identification (SSID; (85, 86)). This procedure constructs a large delay embedding matrix which concatenates lagged future and past copies of the observations ($\mathbf{y}$) and inputs ($\mathbf{u}$). The 'horizon' of these lags was set to match the largest tested model (see below). We used the canonical variate analysis procedure (87), which uses SVD to find low-dimensional mappings between lagged inputs and outputs. We can recover the A and C matrices by processing different blocks of this SVD, and then estimate the rest of the parameters using fixed-gain Kalman filtering. We performed CVA using a modified version of the `ControlSytemIdentification.jl` Julia package (Method 13).

6. Fitting procedure

A. Preprocessing. We followed the same procedure for the Hall-McMaster and Arnau datasets. We split the data into training and testing folds, preprocessing the inputs and observations of each separately. We epoched the recordings around the preparatory period, including the cue, the delay period, and the first 200 ms of the trial (before any RTs had occurred, due to the short-RT trial rejection).

The SSM inputs included an intercept (i.e., effect of time), the current task, the previous task, the cue identity (with separate regressor contrasting the cues within each task), cue repetitions vs alternations for repeat trials, task switch vs task repeat, the upcoming trial's reaction time, and the previous trial's reaction time. For the initial conditions, we included an intercept, the previous task, the previous reaction time, and the upcoming reaction time. We convolved the predictors with a cubic B-spline basis set. Spline knots were placed to optimize coverage at every 5 timesteps, tiling the entire epoch (Hall-McMaster: 20 bases per predictor, Arnau: 25 bases per predictor). Each input was z-scored across trials to standardize and reduce collinearity.

We preprocessed the EEG electrodes using PCA, projecting the voltage timeseries into PCs accounting for 99% of the variance. We used PCA because the observations were degenerate due to spatial proximity, independent components analysis, and bad-channel interpolation. PCA also reduced the computational demands of this analyses, reducing the observation dimensionality and making the observation covariance closer to diagonal. We estimated the PCs using only the training data, and then used these components to project the test data.

B. SSID. To provide the best initialization across the latent dimension hyper-parameter, we estimated SSID for a horizon and latent dimensionality that matched the largest tested model (128 latent factors). We then truncated these systems for smaller models (as latent states from CVA are ordered by their singular values). We found that SSID was enhanced by reshaping the trial-wise data into a long timeseries (e.g., allowing for larger numbers of factors and low frequencies). During SSID, we also only included inputs in the initial timesteps of each epoch, reducing the collinearity between lagged inputs. Note that test data we removed before this process (which were on separate experimental blocks to minimize temporal proximity), and the EM procedure was fit within each epoch and with all inputs. In practice, we found that this procedure provided a good initial guess of the parameters without requiring multiple initializations, which was especially important when there were more factors than observation dimensions.

C. Expectation Maximization. As described above, the EM procedure cycled between estimating the latent states, and updating the parameters. We fit the model across eight different levels of latent dimensionality: (16, 32, 48, 64, 80, 96, 112, 128). We set the maximum number of iterations to 20,000, measuring the test-set log-likelihood every 100 iterations. We terminated the fitting procedure if either the total data likelihood in the training set stopped decreasing, or if the test log-likelihood stopped decreasing.

D. Parameter Recovery. We validated that the combination of model, data, and fitting procedure could produce identifiable parameter estimates when we know the generative model class. First, we used the SSM generative model to create a synthetic dataset based on a participant's parameters and inputs. We then performed on this synthetic dataset. Next, we aligned the estimated and recovered parameters, as the estimates from are only identifiable up to an invertible transformation. Intuitively, we could shuffle the ordering of the latent factors without changing the likelihood. Finally, we correlated the generating and recovered parameters (correlating the Cholesky factors for covariance matrices). While we had somewhat poorer recovery of input matrices (Fig. 1D), this strongly depended on the norm of the input matrix column that we were trying to recover (i.e., some predictors were weakly encoded, and these were difficult to recover), and also likely reflects the correlation between temporal bases.

7. Model Comparison

A. Bayesian Model Selection. We selected the number of latent dimensions for subsequent analyses using a standard Bayesian model selection procedure (48). This analysis estimates the expected probability of each model in the population. From these posterior mixture weights, we computed a 'protected exceedance probability', the probability that a model is the most popular within the tested set of models, relative to chance.

B. Sensor-Level Null Models. To compare to a set of simpler null hypotheses, we fit a set of four models directly to the electrode PCs (i.e., without a latent embedding beyond the PCA). First was an intercept-only model (for standard R^2). Second was an encoding analysis using the full temporal basis set (a generalized additive model similar to EEG methods like the unfold toolbox, (88)). Third was a vector autoregressive (VAR(1)) model, estimating the multivariate relationship between sensor PC responses on adjacent timesteps. Fourth was a model that incorporated both encoding and autoregression. Model four, which included both VAR and encoding, was the best-performing of these null models, so we used this as the benchmark in Figure 2D. We calculated the Cox-Snell R^2, which uses likelihoods instead of squared errors. This metric is equivalent for normal models with stationary residual covariance (89)), but in this case lets the fit metric account for the predicted covariance.

C. Neural-Optimized Recurrent Neural Networks. To compare to a more complex, non-linear model, we predicted the PC timeseries using recurrent neural networks (RNNs) implemented in PyTorch (90)). Each participant was fit with RNNs matched for the number of parameters at each level of SSM factor dimensionality. The inputs to this RNN were the previous PC scores and the same set of predictors as the SSM (formatted as constant inputs over each epoch, which fit better). These inputs were linearly projected into a set of hidden units, along with the previous hidden state, and then passed through a rectified linear unit (ReLU) activation function. The PC score on the next time was linearly decoded from the hidden state using the transpose of the

observation embedding layer (a 'tied' parameterization; (91)). We found similar performance when we fit both the encoding and decoding layers, as these required more parameters. We also fit an initial hidden state, which depended on the previous task.

$$
\begin{aligned}
h_t &= \text{ReLU}(x_{t-1}W_{ih}^T + h_{t-1}W_{hh}^T + b_{hh}) \\
h_{t=1} &= x_0 W_0^T \\
\hat{y} &= h_t W_{ih[y]}
\end{aligned}
$$

Note that for consistency with PyTorch, we use h_t to refer to the latent state (SSM: x_t), and x_t to refer to the inputs (SSM: u_t). $W_{ih[y]}$ refers to the columns of W_{ih} used to encode the previous observations.

We fit 32 random initializations of the RNN to each participant and factor size. Each fitting session involved 5000 epochs of updating the parameters by backpropagating through time the mean square error of the next-timestep prediction. Each batch contained all of the training data. Parameters were updated using the 'AdamW' learning rule (learning rate = .001, weight decay = .01; (92)). To evaluate each model's performance, we took the best test-set loss across all epochs and initializations.

8. Task State Analysis

A. Additive state decomposition. To isolate the dynamics of task representations, we leveraged a powerful property of linear systems: the superposition principle. SSMs can be factorized into an additive set of subsystems for each input, a procedure known as 'additive state decomposition' (51). We used this principle to model 'task subsystems' in which state dynamics are only caused by task-related inputs. We modeled task inputs for switch using the difference between 'current task' and 'previous task' predictors, and task inputs for repeat trials using their sum. This allowed us to further decompose the system into a switch-task subsystem and a repeat-task subsystem. Within each subsystem, the state evolves according to:

$$
\begin{aligned}
\mathbf{x}_t^{\text{task}} &= A\mathbf{x}_{t-1}^{\text{task}} + B^{\text{task}}\mathbf{u}_{t-1}^{\text{task}} \\
\mathbf{x}_{t=1}^{\text{task}} &= B_0^{\text{prevTask}}\mathbf{u}_0^{\text{prevTask}}
\end{aligned}
$$

Since we contrast-coded the tasks (+1/-1), the dynamics for each task are symmetrical around the origin. For the initial conditions of this task subsystem, we used the 'previous task' component of the initial conditions ($B_0^{\text{PrevTask}}\mathbf{u}_0^{\text{PrevTask}}$), which is also symmetrical between switch and repeat trials for the same task. To better equate task states across participants, we normalized the latent space by transforming the parameters by the inverse of the univariate state noise ($\texttt{Diagonal}(W)^{-1}$).

B. Spatiotemporal Visualization. Using the decomposition and symmetry principles afforded by the linear system, we compared the estimated latent trajectories of task representations between switch and repeat trials. We first visualized task states using group-level singular value decomposition (SVD). The goal of this visualization method was to provide an aggregate task state that accommodated the different realizations across participants (i.e., that aligned their latent states). In both of the spatial and temporal visualizations, the use of SVD makes the sign of these visualizations arbitrary.

To visualize the latent trajectories over time, we spatially concatenated participants' switch and repeat task trajectories into a wide 2D 'temporal' matrix ((switch timesteps+repeat timesteps)×(factors×participants)). We then used SVD to extract separate scores for the switch and repeat subsystems. Since we only have two tasks, tasks were coded as a contrast, which meant that the trajectories were symmetrical around an interpretable zero point. For illustration purposes, we plotted the trajectories for both tasks.

C. Temporal Generalization Analysis on Task State. We performed temporal generalization analysis (93), using a similarity-based approach rather than a decoding-based approach (54). To compute temporal generalization, we computed the cosine of the angle between task states for switch and repeat conditions at every timepoint and temporal lags.

To test whether the temporal generalization of task representations increases over the epoch, we extracted the off-main diagonals of the similarity matrix, which reflect task similarity at different temporal lags. We averaged the diagonals above and below the main diagonal, and then correlated them with a linear time vector (i.e., tested whether the similarity at a given lag increases or decreases over timesteps). After computing these correlations for each participant, we tested for a group-level difference from zero using TFCE.

D. Task Distance. To characterize how task states converged (or diverged), we measured the proximity of task states across switch and repeat subsystems. We computed the log Euclidean distance between either (1) when the current task was the same and the previous tasks were different, or (2) when the current task was different and the previous tasks were the same.

E. Task Velocity. Within the switch and repeat subsystems, we quantified the velocity of task trajectories. To determine whether velocity differed when states were more convergent, we computed the partial correlation between the same-task proximity and the velocity, controlling for linear and quadratic effects of timestep. For plotting and analysis, we trimmed the first two and last two timesteps, as these had extreme values due to edge artifacts in the temporal basis set (refer to the first and last bases in Fig. 1B).

F. Performance Alignment. To test whether the inferred task states mattered for good performance, we explored how trial-level brain states predicted upcoming reaction times. First, we projected each participant's task trajectory for switch and repeat subsystems back into the observation space ($C\mathbf{x}^{\text{task}}$). On every trial and timestep, we computed the cosine similarity between the task state and the brain response. We then put this similarity into a regression along with predictors accounting for task, switch trials, task $\times$ switch interaction, and the 2-norm of the neural response. We tested whether EEG-task alignment was significantly different from zero at the group level using TFCE.

G. Task Energy. To quantify the magnitude of task representations, we used Lyapunov functions. These are control theoretic tools used to quantify asymptotic system properties in methods like controllability analysis (37, 55, 94). Controllability quantifies the strength of the input-state coupling. For a time-invariant system, the controllability Gramian (X_∞) defines the asymptotic state covariance that is attributable to a (white noise) input:

$$\begin{aligned} X_\infty &= AX_\infty A^\top + BB^\top \\ &= \sum_{t=0}^{\infty} A^t BB^\top A^{t\top} \end{aligned}$$

X_∞ can be computed by solving the corresponding Lyapunov equation. To capture the influence of a particular set of time-varying inputs, we simulated the time-resolved trajectory of this Gramian using a recursive Lyapunov function (57):

$$\begin{aligned} X_t^{\text{task}} &= AX_{t-1}^{\text{task}} A^\top + (B^{\text{task}}\mathbf{u}_{t-1}^{\text{task}})(B^{\text{task}}\mathbf{u}_{t-1}^{\text{task}})^\top \\ X_{t=1}^{\text{task}} &= (B_0^{\text{prevTask}}\mathbf{u}_0^{\text{prevTask}})(B_0^{\text{prevTask}}\mathbf{u}_0^{\text{prevTask}})^\top \end{aligned}$$

This task-dependent Gramian measures how much the system changes due to both immediate inputs and the spread of inputs through recurrence. It is similar in spirit to the 2-norm of the task state (34), but additional takes into account the cumulative state change (not just the distance from the origin). We summarized task energy using a standard metric of 'average controllability' (37), taking the trace of X^{task} each timestep.

H. Model Ablation. We eliminated different components of the fitted model to assess their contributions to the task state trajectories. We assessed the role of system recurrence by setting the recurrence matrix (A) and initial conditions ($B_0 u_0^{\text{PreviousTask}}$) to zero. We assessed the role of input strength by z-scoring Bu_t^{task} at each timestep.

9. Task-Optimized RNNs

A. Task. We trained networks to perform an extension of the 'context-dependent decision-making' task (CDM; (63, 64, 95, 96), which was designed to be highly similar to the task that participants performed during EEG.

We trained networks on two kinds of sequences. On 'single trials', networks performed a classic CDM task. In each sequence, networks first recieved a task cue input (one-hot coded; 10 timesteps), followed by a delay period (20 timesteps). They then got two pairs of stimulus inputs (40 timesteps), having to judge which input in the task-relevant pair had higher amplitude. We trained the networks using a logistic loss function on the correct ($y_i = 1$) and incorrect ($y_i = 0$) response options ($-(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i))$).

On 'double trials', networks performed two consecutive trials (i.e., without resetting the state). They performed the first trial, had a short intertrial interval (20 timesteps for short-ITI models or 60 timesteps for long-ITI models) and then performed the second trial. Sequences within an epoch had a equal number of target repetitions, distractor repetitions, and task repetitions (64 conditions).

Each epoch contained 150 conditions sets (9600 sequences), each with normally distributed noise (cue d' = 1.0, stimulus d' = 0.15). We trained at relatively low SNRs to pressure the networks to find task-relevant states.

B. Network Architecture. We trained RNNs with gated recurrent units (GRUs; 108 hidden units; (97)) to perform task-switching. These networks used a standard single-layer RNN architecture: linearly encoded inputs (x_t), a recurrent hidden state with hyperbolic tangent activation function (h_t), and linear decoded outputs (y_t). The GRU component of these models learn a pair of 'gates', each of which encodes the hidden state and inputs, and outputs a sigmoid-constrained gating value that

is element-wise multiplied with hidden states. The 'reset gate' (r_t) is applied to the hidden state before it is combined with the inputs and passed through the activation function to produce the 'proposal state'. The update gate (z_t) mixes the proposal state and the previous hidden state. This was implemented through the standard GRU class in PyTorch:

$$
\begin{aligned}
r_t &= \sigma(W_{ir}x_t + b_{ir} + W_{hr}h_{t-1} + b_{hr}) \\
z_t &= \sigma(W_{iz}x_t + b_{iz} + W_{hz}h_{t-1} + b_{hz}) \\
n_t &= \tanh(W_{in}x_t + b_{in} + r_t \odot (W_{hn}h_{t-1} + b_{hn})) \\
h_t &= (1 - z_t) \odot n_t + z_t \odot h_{t-1} \\
y_t &= \sigma(W_{ho}h_t)
\end{aligned}
$$

Note that for consistency with PyTorch, we use h_t to refer to the latent state (SSM: x_t), and x_t to refer to the inputs (SSM: u_t). We trained the GRUs using backpropagation through time, using the AdamW learning rule (learning rate = .01, weight decay = .01).

C. Curricula. We trained two groups of GRUs on different curricula. 'Single networks' only trained on the single trials (500 epochs). 'Mixed networks' trained on single trials for 90% of their training (450 epochs), and then double trials for the final 10% (50 epochs). The rationale for this design was to teach both groups how to perform the task, but give the Mixed networks enough experience that they would learn a compensatory strategy for switching. Note that 450 epoch constituted over-training on the task, at a point where the 'test loss' (noise-free loss) was increasing (Fig. S9). We trained 512 networks in each group, using matched random seed to ensure that training was the same for the first 90% of epochs.

D. Performance Evaluation. Networks were tested on noise-free double trials to evaluate whether they had learned the underlying task structure and could switch between tasks. We measured the within-sequence performance through the per-timestep loss, averaged over conditions and log transformed. To test for temporal generalization, we evaluated networks on 'triple trials', which included a third trial that was absent from training.

E. Temporal Generalization. To test the generality of mixed networks' capacity to switch between task, we explored a curriculum for assessing abstract flexibility. In this curriculum, single trials involved learning four different tasks (mapped to two responses). In double trials, networks switched and repeated only two out of the four tasks. Single networks performed 1000 epochs of single trials, and mixed networks performed 900 single and 100 double. We then assessed both networks' switching ability on double trials containing the two tasks that had been held out of switching practice, comparing the timestep-resolved loss on switch and repeat trials.

F. Gate Ablations. To assess the role of the reset and update gates in GRU's performance, we trained a series of model ablations. We either fixed the reset gate to be open, fixed the update gate to be open, or trained a vanilla RNN. GRUs were trained with tanh activation function and learning rate = .01, whereas the RNN was trained with a ReLU activation function and learning rate = .001 (which improved its performance). Free parameters were matched across networks (108 hidden units for full model, 134 units for ablated models, 190 units for RNN). We trained these models on 500 epochs of the single trials, and then 500 epochs of the double trials.

G. Hidden Unit Visualization. We used a similar group SVD procedure to the EEG (see Method 8) to visualize hidden unit activation during double trials. Using trained networks, we first simulated a set of noisy trials (n=2048) for a single and mixed network (same random seed). Next, we concatenated the hidden unit activations across sequences and networks into a joint matrix $((\text{timesteps} \times \text{sequences}) \times (\text{ single-network hidden units} + \text{ mixed-network hidden units}))$ Performing SVD on this matrix provides embeddings that captures the common and distinct components of the hidden unit activation, and in practice it produces quite similar trajectories across the two networks. We then averaged the sequences within each task pair and projected these average sequences into the (weighted) right singular vectors corresponding to each network.

10. Network Distillation

We distilled the trained RNNs into a high-dimensional SSM, allowing us to more directly compare the task dynamics to human participants. The fitting procedure was closely matched between networks and EEG. We preprocessed training and test trials in a similar way (1536 trials for training, 512 for testing): performing PCA on the hidden unit trajectories (keeping components accounting for 99% of the variance) and constructing a temporal basis for the inputs (10 spline bases over the epoch). Network inputs were a bias, task on switch trials, task on repeat trials, and the main effect of switch. Inputs to initial conditions were a bias and the previous task. We analyzed the preparation for the second trial: cue period (10 timesteps), delay period (20 timesteps), and the initial trial period (10 timesteps).

We fit across five levels of latent dimensionality $(16, 40, 64, 88, 112)$, initializing the parameters with SSID at a horizon of 112. We ran for all 512 single networks and mixed networks. Networks were evaluated using the same methods as human participants.

11. Quantifying RNN dynamics

12. EEG-RNN comparison

To evaluate the similarity between (1) temporal generalization and (2) task energy across networks and humans, we first subsampled human task trajectories (100 or 125 timesteps) to match the duration of networks' task trajectories (40 timesteps). Next wek then averaged the each measure within humans and RNNs, and then computed the cosine similarity between these vectorized traces across groups ('congruence coefficient'). To produce a bootstrap estimate, we sampled with replacement 10,000 times from both the RNN and EEG groups, averaging and comparing each sample as above, and then computed the 95th percentiles of this bootstrap distribution.

A. ITI comparison. We compared Mixed and Single Networks that had learned under short ITIs (20 timesteps) and Mixed and Single networks that had learned under long ITIs (60 timesteps), using two different methods (N=512 for each combination of network type and ITI).

First, we computed the Euclidean distance between hidden unit activations during the ITI. We used the distance between the average states for each of the two possible previous tasks (i.e., the distances between the post-A states and the post-B states).

Second, we tested whether the networks were using gating to elicit a plausible convergence target state. We used the learned weights to reconstruct the update and reset gates for the post-A and post-B trajectories. For the convergence target, we used the next timestep of the other task. For example, the target during the ITI after task A was $h_t^* = \hat{h}_{t+1}^{\text{taskB}}$. This target has the effect of bringing the state to a point of convergence, and we found similar results using the terminal location of the trajectories. We unit-normalized this target to make this comparision more comperable between networks. To test whether these target states were aligned with the gating policy, we took the element-wise multiplication of the gates with the convergence target, and then computed the 2-norm at each timestep. We computed this for the epoch-averaged timeseries, but found similar results when we computed it per-trial and then averaged.

13. Software and Visualization

1. Programming Languages
   (a) Julia (v1.11.2)
   (b) Python (v3.10.13)
   (c) Matlab (R2023b)
2. Software Packages
   (a) PyTorch (v2.4.0; pytorch.org)
   (b) MatlabTFCE (github.com/markallenthornton/MatlabTFCE)
   (c) ControlSystemIdentification.jl (v2.10.2; github.com/baggepinnen/ControlSystemIdentification.jl)
   (d) FieldTrip (98)
   (e) EEGLAB (v2024.2; (99))
3. Visualization
   (a) HenriquesLab bioRxiv template (overleaf.com/latex/templates/henriqueslab-biorxiv-template/nyprsybwffws)
   (b) Scientific Color Maps (100)

The author(s) are pleased to acknowledge that the work reported on in this paper was substantially performed using the Princeton Research Computing resources at Princeton University which is a consortium of groups led by the Princeton Institute for Computational Science and Engineering (PICSciE) and Office of Information Technology's Research Computing.

14. Supplementary Figures

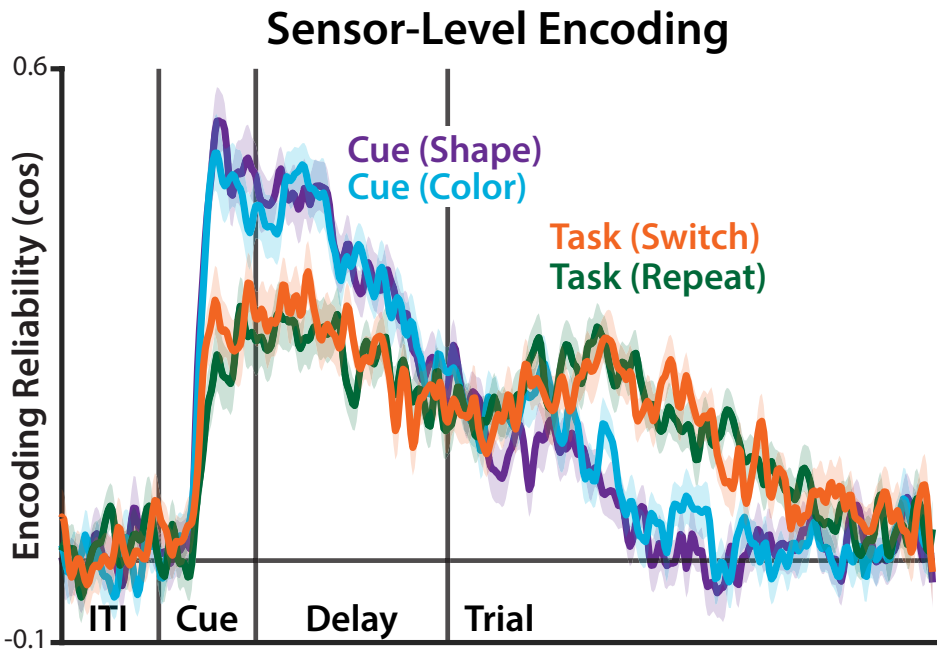

Fig. S1. Sensor-level encoding. Cross-validated encoding reliability of cues for each task (pink and cyan) and tasks for each switch condition (purple and blue). Error bars reflect within-participant standard error. Differences in task encoding strength was not significantly different between switch and repeat conditions.

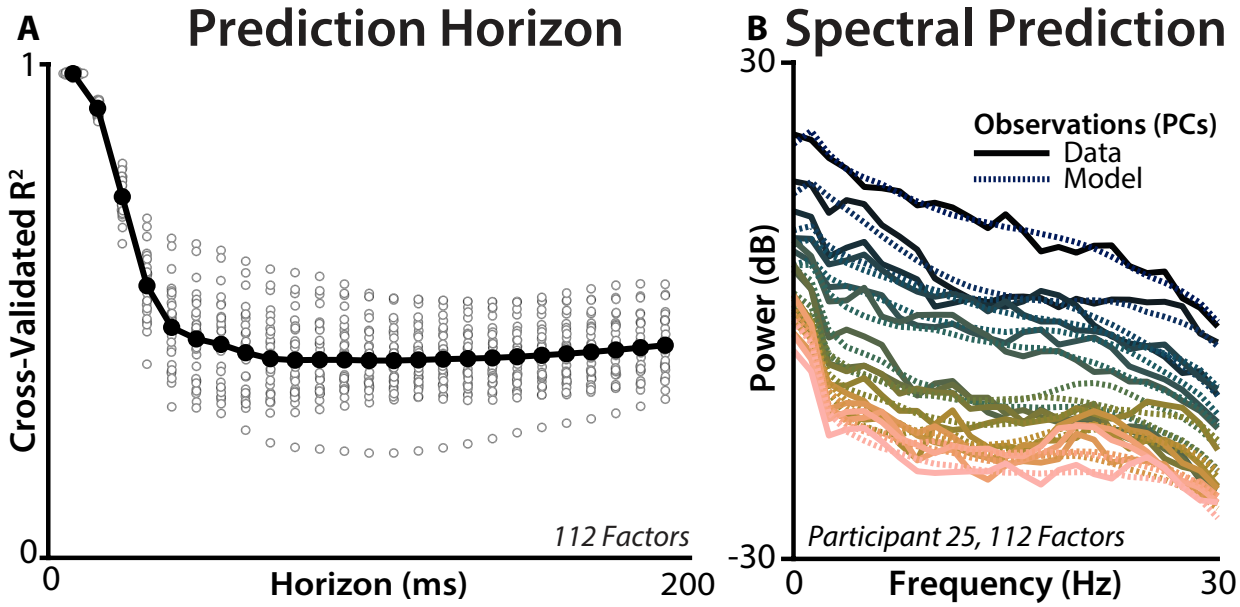

Fig. S2. Extended predictions. **A)** Predictive accuracy of the model when forecasting at different timesteps into the future. White dots reflect individual participants. **B)** Overlap in the power spectrum for test-set observations (solid lines) and model simulations (dashed lines), shown for participant 25. Power spectra are plotted for each observation dimension (i.e., principal component), averaged over epochs using EEGLab's `spectopo` function. Note that these predictions are not filtered (i.e., these are feedforward simulations, averaged over 100 repetitions). Across participants, there was high overlap in the power spectra (median R^2_{CV} = .98; 99% of the noise ceiling, which was estimated by the similarity between training set spectra and test set spectra). There was still high overlap after mean-centering and linearly detrending the spectra within each channel (median R^2_{CV} = .82; 92% of noise ceiling)

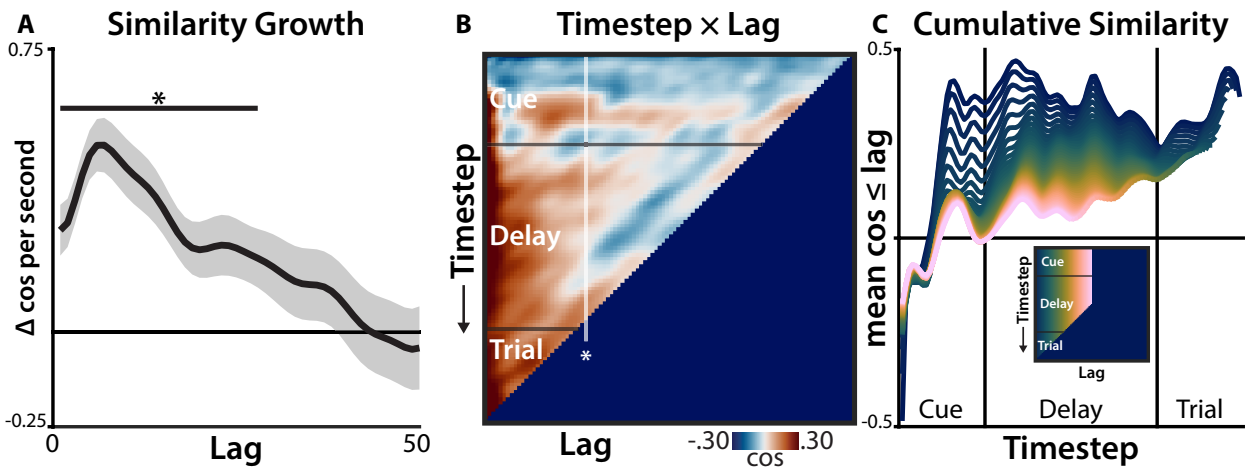

Fig. S3. Change in generalization over the epoch. **A)** Average cosine similarity between task patterns across switch and repeat conditions, plotted at every timestep (y-axis) and time lag (x-axis;compare to Fig. 2C). Vertical white line reflect significance cut-off from (B). **B)** Change in similarity over the epoch (y-axis; linear regression estimate), as a function of lag (x-axis). Similarity change is significantly greater than zero for lags up 31 timesteps (224 ms; TFCE-corrected). Error bars reflect SEM. Corresponds to columns of (A). **C)** At each timestep (x-axis), the cumulative similarity up to each lag (line color; see inset for legend). Corresponds to rows of (A).

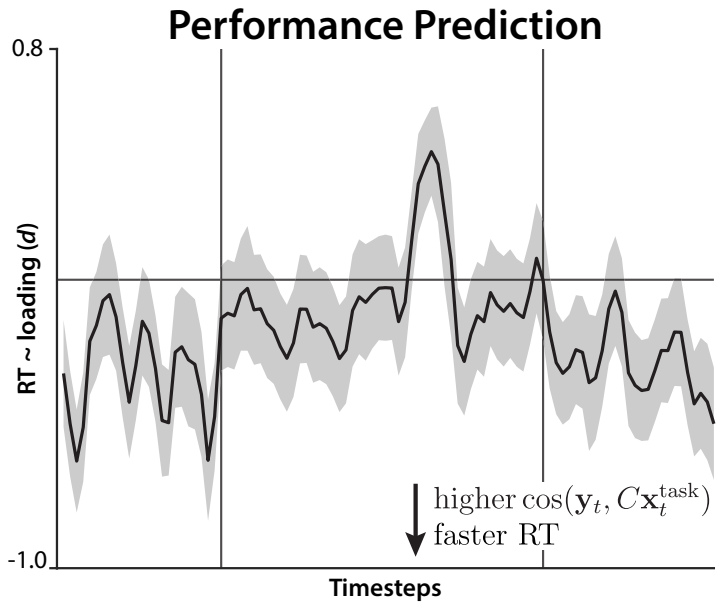

Fig. S4. Relevance of task states for good performance. Group-level effect size (y-axis; Cohen's d) of the individual regression weights mapping 'task loading' on to log reaction time. Task loading is the cosine similarity between EEG activity and the estimated task state. Error bars reflect SEM. No individual timestep is significantly different from zero (using TFCE), however the sum of the regression weights over time is significant ($d = -0.55$, t-test $p = .0054$), with stronger alignment predicting shorter reaction times.

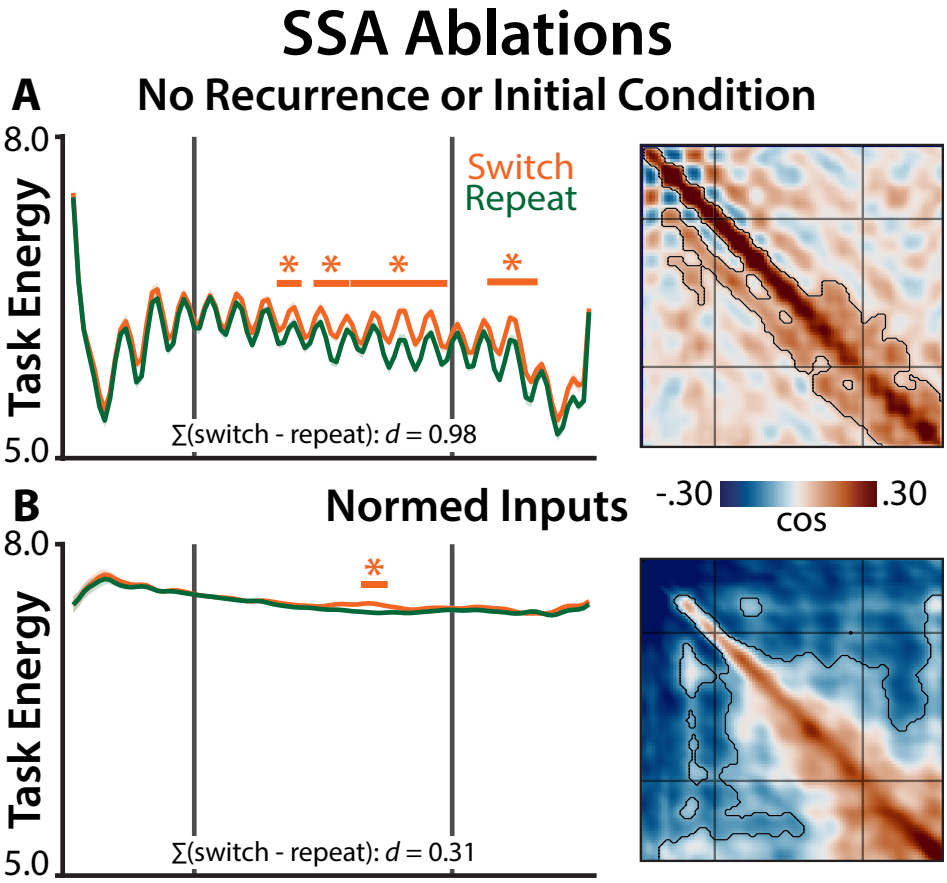

Fig. S5. Ablation. Task energy (left) and temporal generalization (right) for ablated models (generating task states after reduce the influence of fitted parameters). **A)** Removing recurrence (setting A and B_0 to 0) largely preserved differences in task energy across switch conditions, but dramatically changed the pattern of temporal generalization. Note that apparent oscillations are due to the locations of the spline bases. **B)** Removing input magnitude (z-scoring $B^{\text{task}}\mathbf{u}^{\text{task}}_{t-1}$) eliminated the task strength effect, but largely preserved the pattern of temporal generalization (albeit with stronger negative generalization than positive generalization, consistent with interference).

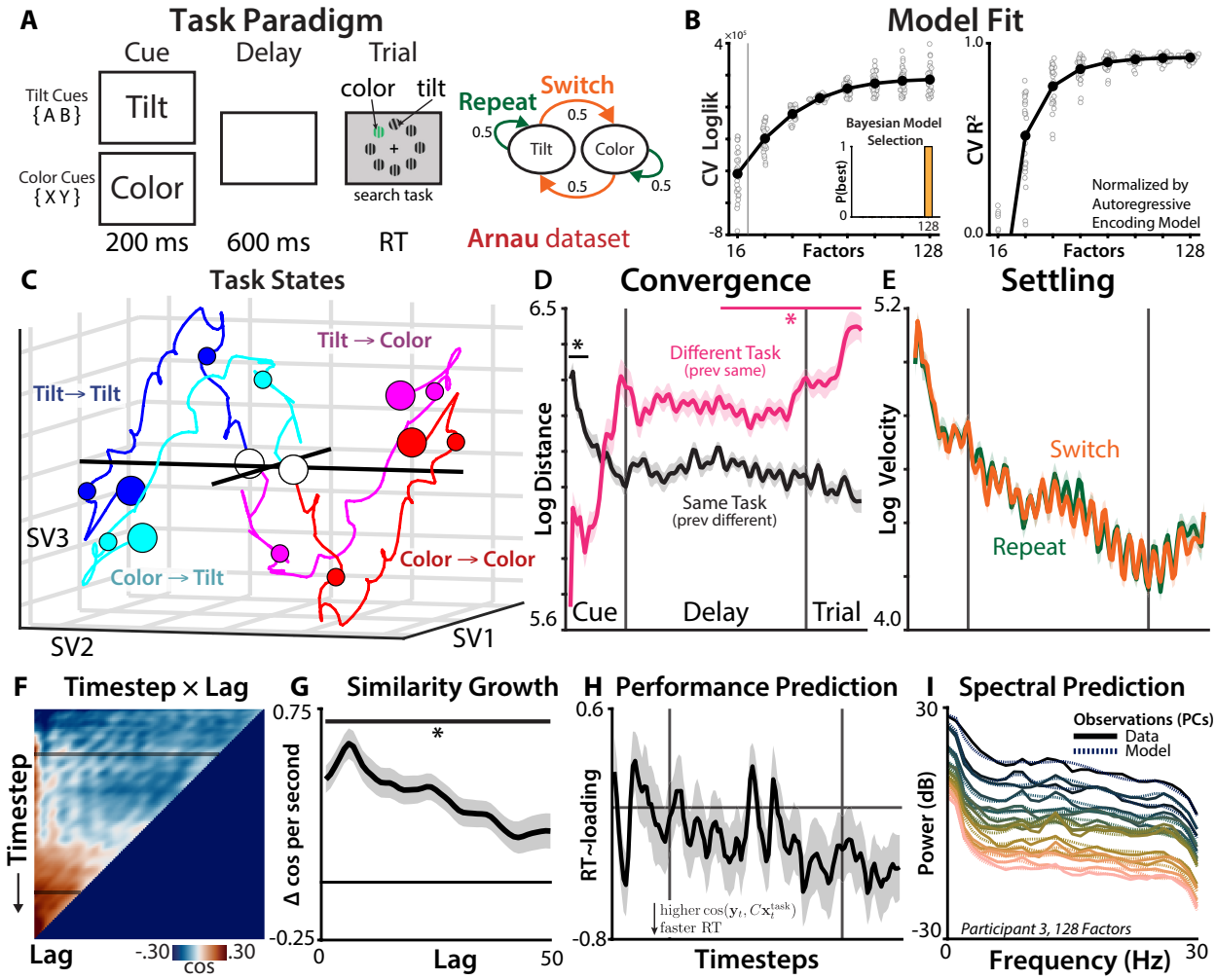

Fig. S6. Replication Dataset ('Arnau'). **(A)** As in Hall-McMaster, participants saw a task cue (two symbols mapped each task), waited through a delay period, then performed the task. Search task required participants to find the colored grating or the tilted grating. Left: The tasks had an equal change of repeating or switching between trials. **(B)** As in Hall-McMaster, models with more factors than observation dimensions provided the best fit. Left, the 128-factor model had the highest CV loglikelihood. Right, the best fitting model had high predictive accuracy ($R^2 = 92\%$, relative to an autoregressive null model). Compare to Fig. 1C,F. **(C)** Low-dimensional projection of the simulated task trajectory. Compare to Fig. 2A. **(D)** Log Euclidean distance between task states for the same task (black) and different tasks (pink). Compare to Fig. 2B. **(E)** Log velocity of task states on switch and repeat trials. Compare to Fig. 2C. **(F)** Generalization of task states at different timepoints (y-axis) and lags (x-axis). Compare to Fig. S3A. **(G)** Linear increase in similarity over time a different lags (columns of (F). Compare to Fig. S3B. **(H)** Group-level effect size (y-axis; Cohen's d) of the individual regression weights mapping 'task loading' (cosine between EEG observation and projected task state) on to log reaction time. Note that weights are not significantly different from zero (summed over time d = -0.31, left-tailed p = 0.06), however they are significant using a sign-rank test (p = 0.0095) and during the trial event (p = 0.022). Compare to Fig. S4. **(I)** Prediction of spectral power for test-set observations from open-loop model simulations (centering and detrending within each channel: $R^2 = 0.88$, which is 91% of the noise ceiling). Compare to Fig. S2. Astericks and contours indicate $p < .05$ (TFCE-corrected).

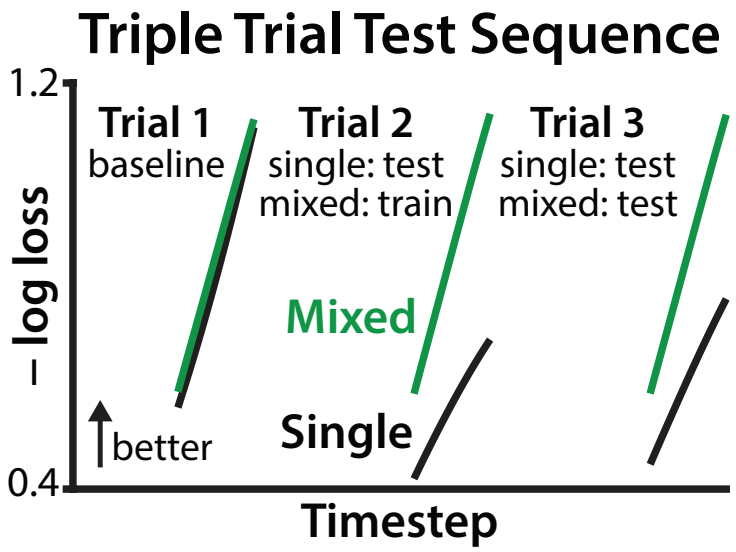

Fig. S7. Sequence generalization. Average test loss across timesteps within a 'triple trial'. Networks were trained on single trials (single networks) or single and double trials (mixed networks), and then tested on this three-trial sequence. SEM smaller than line thickness. Shown for short-ITI model.

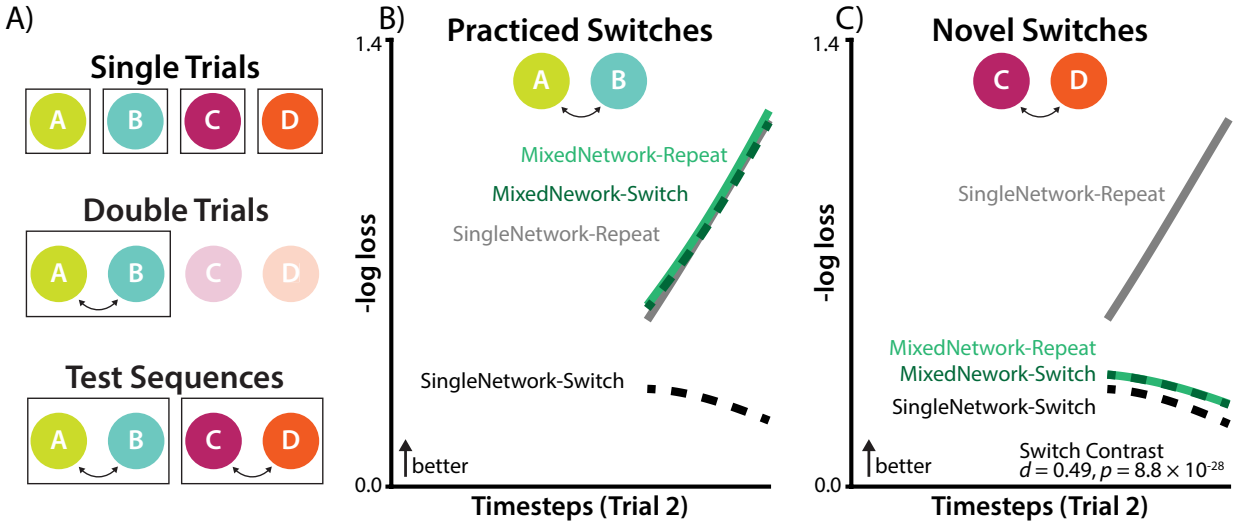

Fig. S8. Task generalization. **A)** We trained GRUs on a modified version of the protocol. In single trials, networks learned four different tasks. In double trials, networks practiced switching between two of the tasks. In test (shown in A and B), we examined switching performance between the practiced and novel task pairs. As in the main experiment, single networks performed only single trials (1000 epochs), and mixed networks were trained on 90% single trials and 10% double trials. **B)** The mixed network shows great performance on task transitions that it had practiced, it (as good at switching as the single network is at repeating). This is similar to the results Figure 4F (shown fore each condition, rather than as contrasts). **C)** On novel task transitions, the mixed network still shows lower switch costs and better performance on switch trials. Note that it has worse performance on repeat trials, likely due to these networks having less experience with C and D tasks (and/or catastrophic interference). Shown for short-ITI model.

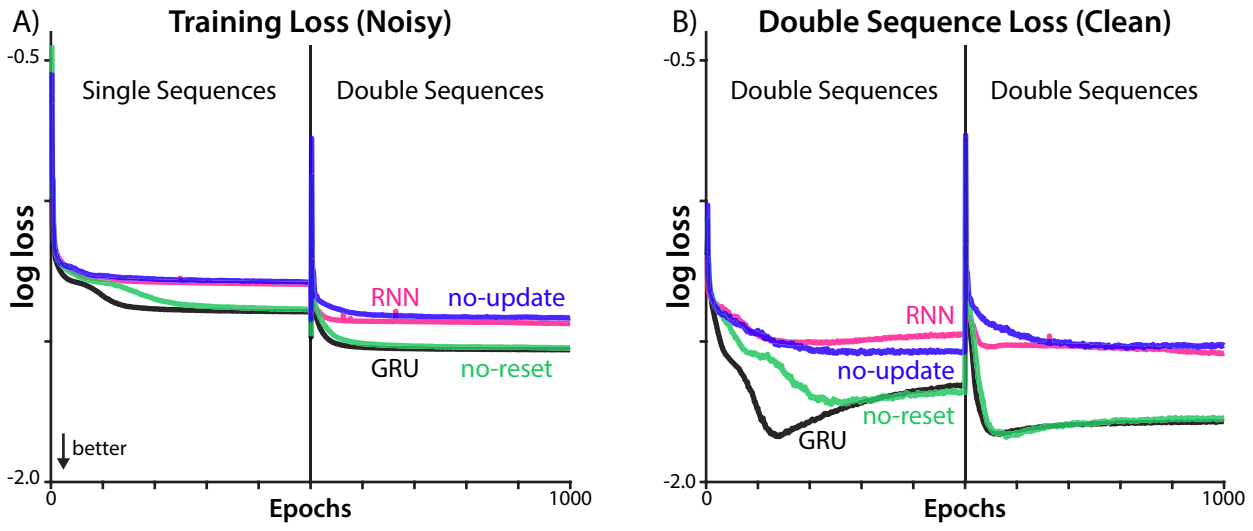

Fig. S9. Gate Ablation. We trained models with all gates ('GRU', black), fixed-open reset gates ('no-reset', green), fixed-open update gates ('no-update', blue), and classic Elman RNNs ('RNN', pink). All models are matched for the number of parameters (see Methods). The first epoch is removed to improve visualization. Note that we plot loss (instead of -loss) to follow convention. **A)** We trained networks on single trials for 500 epochs, and then double trials for 500 epochs. During this training, noise was added to inputs to encourage the learning of a robust policy. During learning, GRU and no-reset models had similar learning trajectories, with better performance than no-update or RNN models. **B)** Network loss was also tested on double trials at every epoch, without input noise. Here too, the GRU and no-reset models had performance gains from experience with the double trials. Note that GRU test loss during the single trials is increasing, suggesting that this is in an over-trained regime. Shown for short-ITI model.

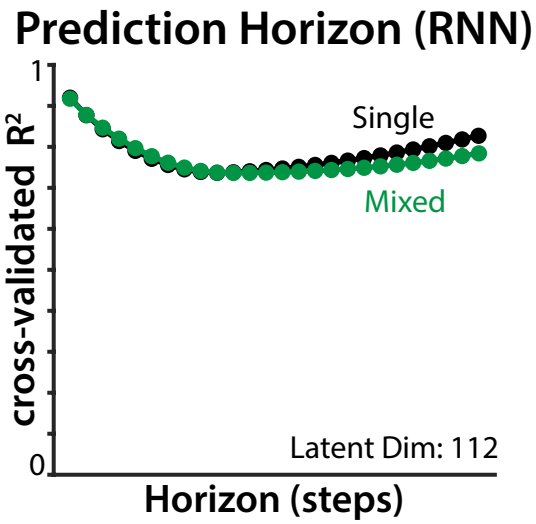

Fig. S10. Prediction horizon for s fit to RNNs. had similarly good predictive accuracy when fit to either single or mixed networks had. This high predictive accuracy was present both at the next timestep (training objective) and at distant temporal lags (25 timesteps is more than half of the epoch). Shown for short-ITI model.

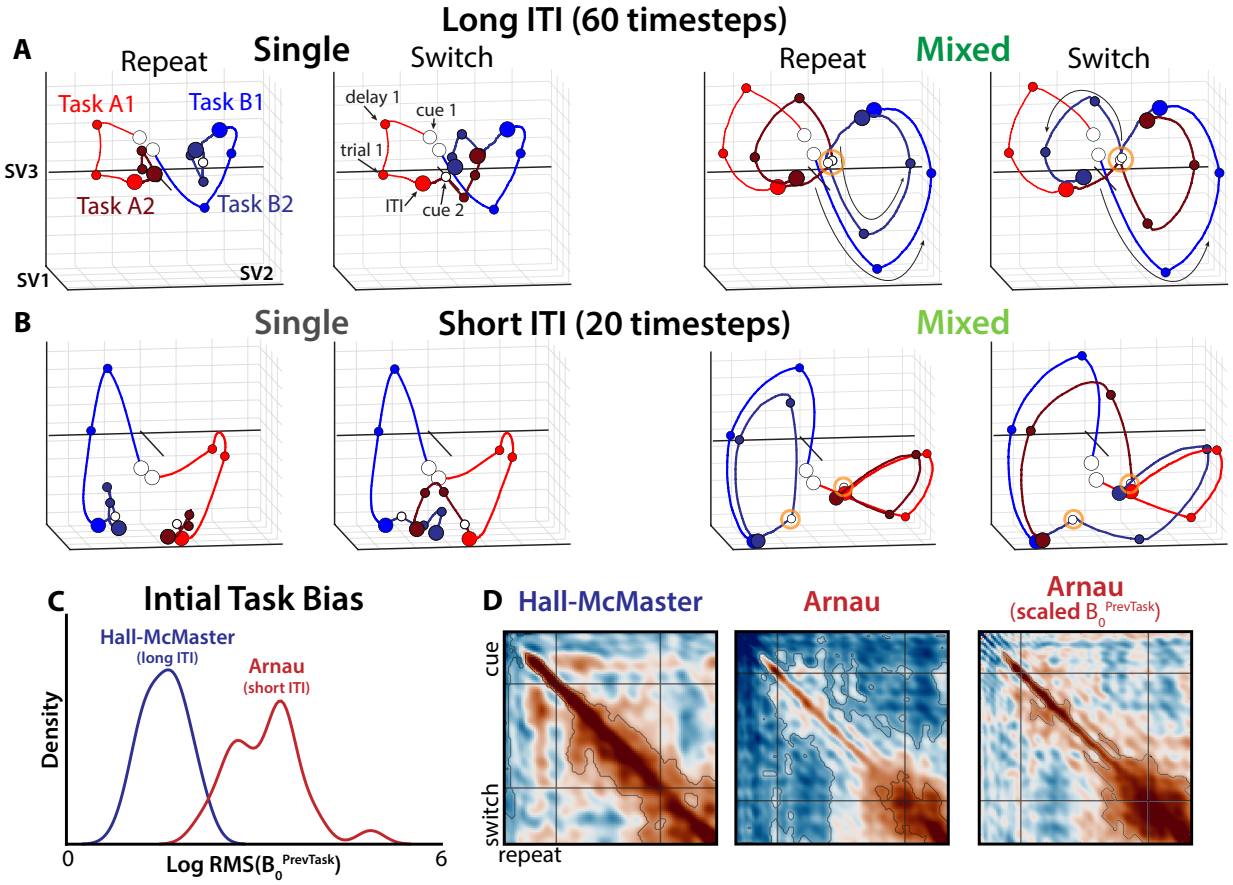

Fig. S11. Comparison between RNNs trained with short and long ITIs. **(A)** Low-dimensional projection of an RRN hidden state for long-ITI networks (same as in Fig. 4). For mixed networks (right two panels), states were close to each other at the time of the second cue. **(B)** RNN hidden state projection for short-ITI networks. Here, states are close to the previous task at the time of the cue. **(C)** Participants in the Hall-McMaster dataset had weaker encoding of 'previous task' in the their initial conditions than participants in the Arnau dataset (both shown at 128 factors). **(D)** Temporal generalization plots for the datasets. The first two plots are from the HallMcMaster and Arnau datasets, shown for reference (same as in Fig. 5). The third panel is from the Arnau dataset, where the magnitude of the initial conditions was rescaled by the ratio between the studies. Rescaling the initial conditions in the Arnau dataset recovers a similar temporal generalization profile as the Hall-McMaster dataset.