Neural Networks 132 (2020) 375–393

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Learning to select actions shapes recurrent dynamics in the corticostriatal system

Christian D. Márton [a,b,*], Simon R. Schultz [a], Bruno B. Averbeck [b]

[a] Centre for Neurotechnology & Department of Bioengineering, Imperial College London, London, SW7 2AZ, UK
[b] Laboratory of Neuropsychology, Section on Learning and Decision Making, National Institute of Mental Health, National Institutes of Health, Bethesda, MD, USA

ARTICLE INFO

Article history:
Received 12 November 2019
Received in revised form 3 September 2020
Accepted 11 September 2020
Available online 19 September 2020

Keywords:
Learning
Recurrent neural network
Corticostriatal system
Dynamics
Reinforcement learning

ABSTRACT

Learning to select appropriate actions based on their values is fundamental to adaptive behavior. This form of learning is supported by fronto-striatal systems. The dorsal-lateral prefrontal cortex (dlPFC) and the dorsal striatum (dSTR), which are strongly interconnected, are key nodes in this circuitry. Substantial experimental evidence, including neurophysiological recordings, have shown that neurons in these structures represent key aspects of learning. The computational mechanisms that shape the neurophysiological responses, however, are not clear. To examine this, we developed a recurrent neural network (RNN) model of the dlPFC-dSTR circuit and trained it on an oculomotor sequence learning task. We compared the activity generated by the model to activity recorded from monkey dlPFC and dSTR in the same task. This network consisted of a striatal component which encoded action values, and a prefrontal component which selected appropriate actions. After training, this system was able to autonomously represent and update action values and select actions, thus being able to closely approximate the representational structure in corticostriatal recordings. We found that learning to select the correct actions drove action-sequence representations further apart in activity space, both in the model and in the neural data. The model revealed that learning proceeds by increasing the distance between sequence-specific representations. This makes it more likely that the model will select the appropriate action sequence as learning develops. Our model thus supports the hypothesis that learning in networks drives the neural representations of actions further apart, increasing the probability that the network generates correct actions as learning proceeds. Altogether, this study advances our understanding of how neural circuit dynamics are involved in neural computation, revealing how dynamics in the corticostriatal system support task learning.

1. Introduction

Human and nonhuman primates are capable of complex adaptive behavior. Adaptive behavior requires predicting the values of choices, executing actions on the basis of those predictions, and updating predictions following the rewarding or punishing outcomes of choices. Reinforcement learning (RL) is a formal, algorithmic framework useful for characterizing these behavioral processes. Experimental work suggests that RL maps onto fronto-striatal systems, dopaminergic interactions with those systems, and other structures including the amygdala and thalamus (Averbeck & Costa, 2017). Little is known, however, about how the RL formalism and the associated behaviors map onto mechanisms at the neural population level across these systems. How do neural population codes evolve with learning across these systems, and what are the underlying network mechanisms that give rise to these population codes?

Experimental work and modeling has implicated fronto-striatal systems in aspects of RL (Botvinick, 2012; Botvinick & Weinstein, 2014; Langdon et al., 2018; Lee et al., 2012; Niv, 2009, 2019; Wang et al., 2018). Several studies have suggested that the striatum codes action values (Amemori et al., 2011; Averbeck & Costa, 2017; Daw et al., 2011; Doya, 1910; Frank, 2005; Frank et al., 2004; Histed et al., 2009; Houk, 1995; Li & Daw, 2011; Nakahara et al., 2001; O'Doherty et al., 2004; Pasupathy & Miller, 2005; Samejima et al., 2005; Sarvestani et al., 2011; Seo et al., 2012; Suri & Schultz, 1998). These studies have further suggested that the phasic activity of dopamine, which codes reward prediction errors, drives updates of the striatal action value representations following reward feedback. Several areas in the PFC have also been implicated in action selection and decision making

* Corresponding author at: Laboratory of Neuropsychology, Section on Learning and Decision Making, National Institute of Mental Health, National Institutes of Health, Bethesda, MD, USA.
E-mail address: cdmarton@gmail.com (C.D. Márton).

https://doi.org/10.1016/j.neunet.2020.09.008

(Aitchison & Lengyel, 2017; Alexander & Brown, 2018; Averbeck et al., 2006; Balaguer et al., 2016; Botvinick, 2012; Botvinick & Weinstein, 2014; Cheong et al., 2006; Collins & Koechlin, 2012; Domenech & Koechlin, 2015; Friston, 2005; Histed et al., 2009; Koechlin & Summerfield, 2007; Lim & Goldman, 2013; Radulescu et al., 2019; Rich & Wallis, 2016; Sakai, 2008; Stokes et al., 2013; Tsujimoto et al., 2008; Verschure et al., 2014; Wallis et al., 2019; Wood & Grafman, 2003). These studies further suggest that PFC plans future actions and predicts future outcomes. While both striatum and PFC have been found to represent action value and choice signals, value signals were found to be stronger in dSTR than lPFC, while action related signals were stronger in lPFC (Averbeck et al., 2006; Pasupathy & Miller, 2005; Samejima et al., 2005; Seo et al., 2012).

Previous work in the motor system and in prefrontal cortex has shown that insight into the computational mechanisms that underlie complex tasks can be gained by treating neural populations as a dynamical system and studying how their trajectories evolve with time (Barak et al., 2013; Botvinick et al., 2019; Buonomano & Maass, 2009; Carnevale et al., 2015; Chaisangmongkon et al., 2017; Gallego et al., 2017, 2018; Hennequin et al., 2014; Mante et al., 2013; Musall et al., 2019; Rabinovich et al., 2008; Rajan et al., 2016; Remington et al., 2018; Richards et al., 2019; Shenoy et al., 2013; Sussillo & Abbott, 2009; Sussillo & Barak, 2013; Sussillo et al., 2015; Sutskever, 2013; Wang et al., 2018, 2017; Yang et al., 2018). In prefrontal cortex this work has helped shed light on how task execution is driven by dynamics around fixed and slow points in neural population space (Chaisangmongkon et al., 2017; Mante et al., 2013). These studies have examined representations and computational mechanisms in decision making tasks, where the values of choices have already been learned previously. In the present study, we have used a similar approach to study how representations *develop* as animals learn to make choices that deliver rewards.

In accordance with these findings, we built a joint recurrent network model of the corticostriatal system in which the striatal network represents RL-derived action values and the prefrontal cortex, via recurrent basal ganglia loops, selects appropriate actions based on this signal. We trained this system on a complex decision making task. We also obtained neural recordings from these two regions in two macaques trained on the same task.

We hypothesized that learning would drive specific structure in state space dynamics. We further hypothesized that a system designed to learn RL-derived action values and select appropriate actions based on them would be computationally similar to the fronto-striatal system in the brain. If so, the representational structure during learning in the network should be similar to that found in neural recordings. Moreover, the differing roles assigned to the striatal and prefrontal networks should suffice to induce a difference in representational structure across the two regions in a way that matches the asymmetries in action value and choice representation observed previously (Seo et al., 2012).

Investigating the change in representational structure with learning, we found that dynamic movement-sequence representations moved apart from each other in latent space with learning, in both the model and the neural data. We found that this process was driven by the evolution of potential surfaces in the networks such that movement-sequence specific gradient minima moved farther apart in activity space. This increase in distance, or in other words, the increase in the height of the gradient hill between sequence representations, makes it less likely that the wrong action is selected as learning proceeds.

2. Methods and materials

2.1. Neural data

The neural data employed here has been previously published in Seo et al. (2012), though not with the analysis that has been carried out here.

2.1.1. Subjects

Two adult male rhesus monkeys (Macaca mulatta) weighing 5.5–10 kg were used for recordings. All procedures and animal care were conducted in accordance with the Institute of Laboratory Animal Resources Guide for the Care and Use of Laboratory Animals. Experimental procedures for the first animal were in accordance with the United Kingdom Animals (Scientific Procedures) Act of 1986. Procedures for the second animal were in accordance with the National Institutes of Health Guide for the Care and Use of Laboratory Animals and were approved by the Animal Care and Use Committee of the National Institute of Mental Health (NIMH).

2.1.2. Task and stimuli

The two animals performed an oculomotor sequential decision-making task (Fig. 1A). A particular trial began when the animals acquired fixation on a green circle (Fixate). If the animal maintained fixation for 500 ms, the green target was replaced by a dynamic pixelating stimulus with a varied proportion of red and blue pixels and the target stimuli were presented (Stim On). The fixation circle stimulus was generated by randomly choosing the color of each pixel in the stimulus (n = 518 pixels) to be blue (or red) with a probability q. The color of a subset (10%) of the pixels was updated on each video refresh (60 Hz). Whenever a pixel was updated its color was always selected with the same probability q. The set of pixels that was updated was selected randomly on each refresh. In the original experiment we focused on differences between choices driven by reinforcement learning and choices driven by immediately available information, in alternating blocks of trials. In the present manuscript we only considered the learning blocks. The pixelating stimulus was relevant for the blocks in which choices were driven by immediately available information. Therefore, we will not consider this stimulus further.

The animal's task was to saccade to the correct target (Fig. 1A). The animal could make their decision at any time after the target stimuli appeared. After the animal made a saccade to the peripheral target, it had to maintain fixation for 300 ms to signal its decision (first Move + Hold). If the saccade was to the correct target, the target then turned green and the animal had to maintain fixation for an additional 250 ms (Fixate). After this fixation period, the green target was again replaced by a fixation stimulus and two new peripheral targets were presented (Stim On). In the case that the animal made a saccade to the wrong target, the target was extinguished and the animal was forced back to repeat the previous decision step. This was repeated until the animal made the correct choice. For every trial the animal's task was to correctly execute a sequence of three correct decisions for which the animal received either a juice reward (0.1 ml) or a food pellet reward (TestDiet 5TUL 45 mg). After that, a 2000 ms intertrial interval began. The animals always received a reward if they reached the end of the sequence of three correct decisions, even if errors were made along the way. If the animal made a mistake, it only had to repeat the previous decision, it was not forced back to the beginning of the sequence. The full task included both fixed and random conditions, as explained in detail in Seo et al. (2012).

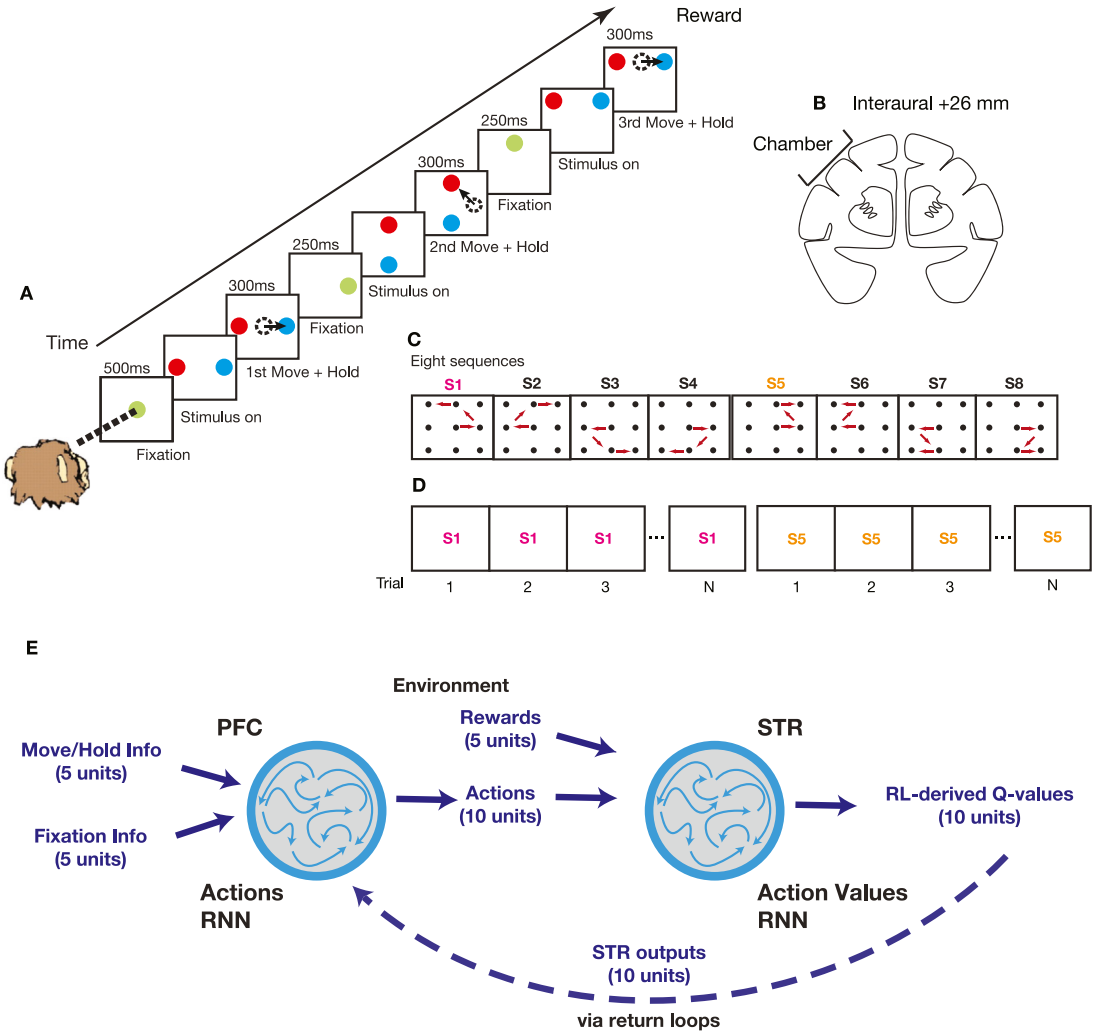

Fig. 1. Task and Model Overview. (A) Sequence of events in the task. (B) Coronal section showing approximate location of recording chamber. (C) The eight possible movement-sequences used in the task. (D) Trial structure. Trials were arranged into blocks of 8 repeats of the correct movement-sequence plus a variable number of error trials (i.e. sequence S1 followed by sequence S5). (E) Corticostriatal model, consisting of a prefrontal network and a striatal network of recurrently connected units. The prefrontal network selects actions based on inputs from the striatal network. The striatal network outputs action values based on inputs specifying actions performed and rewards received.

In the present study, however, only data from the fixed condition was used.

There were eight possible sequences in this task as every trial was composed of three binary decisions (Fig. 1C). The eight sequences were composed of ten different possible individual movements. Every movement occurred in at least two sequences. We also used several levels of color bias, q as defined above. On most recording days in the fixed sets we used $q \in (0.50, 0.55, 0.60, 0.65)$. The color bias was selected randomly for each movement and was not held constant within a trial. Choices on the 50% color bias condition were rewarded randomly. The sequences were highly overlearned. One animal had 103 total days of training, and the other 92 days before chambers were implanted. The first 5–10 days of this training were devoted to basic fixation and saccade training.

In the fixed condition employed here, the correct spatial sequence of eye movements remained fixed for blocks of eight correct trials (Fig. 1D). After eight trials were executed without any mistakes, the sequence switched pseudorandomly to a new one. Thus, the animal could draw on its memory to execute a particular sequence, except following a sequence switch.

2.1.3. Neural data analysis

The neural data analyzed comprised 365 units from dSTR and 479 units from lPFC (Fig. 1B, and as explained in more detail in Seo et al., 2012). Data from individual movements was not analyzed if animal failed to maintain fixation or did not saccade to one of the choice targets. We fitted an ANOVA model with a 200 ms sliding window applied in 25 ms steps aligned to movement onset, as done previously (Seo et al., 2012), and identified units that did not show a significant effect for the *sequence* factor at any point across the entire recording session. These units were excluded from the analysis, as were units that showed average firing rates below 1 Hz. For subsequent analysis, data was pooled across animals and recording sessions and averaged across runs.

To analyze neural population responses, we applied demixed principal component analysis (dPCA) (Brendel et al., 2011; Kobak et al., 2016) to the firing rate traces. As a dimensionality reduction technique, dPCA strives to find an encoding (or latent representation) which captures most of the variance in the data but also expresses the dependence of the representation on different task parameters such as stimuli or decisions. More specifically, it decomposes neural activity into different task parameters, in our case time ($\mathbf{X}_t$), sequence ($\mathbf{X}_s$), and certainty ($\mathbf{X}_c$), and any

combination of those ($\mathbf{X}_{sc}, \mathbf{X}_{st}, \mathbf{X}_{ct}$), as well as a noise term ($\mathbf{X}_{noise}$):

$$\mathbf{X} = \mathbf{X}_t + \mathbf{X}_s + \mathbf{X}_c + \mathbf{X}_{sc} + \mathbf{X}_{st} + \mathbf{X}_{ct} + \mathbf{X}_{noise} = \sum_{\phi} \mathbf{X}_{\phi} + \mathbf{X}_{noise}$$
$$L_{dPCA} = \sum_{\phi} \|\mathbf{X}_{\phi} - \mathbf{F}_{\phi}\mathbf{D}_{\phi}\mathbf{X}\|^2 \quad (1)$$

The various task parameters are summarized by $\mathbf{X}_{\phi}$. The decomposition into task parameters in dPCA is analogous to the variance decomposition in ANOVA, which some readers may be more familiar with. Once decomposed into separate task parameters, dPCA then finds separate encoder ($\mathbf{F}_{\phi}$) and decoder ($\mathbf{D}_{\phi}$) for every task parameter by minimizing the reconstruction loss (L_{dPCA}). This is achieved by performing reduced rank regression for every task parameter. Eventually, the rows of $\mathbf{D}_{\phi}$ contain the demixed principal components of $\mathbf{X}_{\phi}$.

In order to obtain task trajectories in the reduced dPCA space, the data ($\mathbf{X}$) was first smoothed with a Gaussian kernel and then projected into 3-dimensional latent space spanned by the first three demixed principal components of the *sequence*-subspace ($\mathbf{D}_s$). We obtained similar results when imaging in the *certainty*-subspace ($\mathbf{D}_C$) or the subspace obtained by the interaction between the *sequence* and the *certainty* ($\mathbf{D}_{sc}$) parameters.

Distance measures were obtained on the full datasets in full-dimensional neural space, not in the reduced subspace. Euclidean distance between all sequences was computed across all time points for each of the 8 trial repeats and averaged across all possible sequence combinations. For PFC, distances were computed between sequences *within* the two clusters (defined by whether a sequence ended in the upper or lower visual hemisphere (sequences S1, S2, S5, S6 and S3, S4, S7, S8, respectively; see Fig. 1C). As a measure of how compact the trajectories were, the Euclidian distance of every sequence to its centroid was computed across all time points for each of the 8 trial repeats and averaged across sequences. A sequence's centroid was defined as the mean across time of a trajectory in N-dimensional space with $N = 365$ for dSTR and $N = 479$ for lPFC.

2.2. Corticostriatal model

2.2.1. Overview

We trained a system of connected recurrent rate-networks (Fig. 1E) on the same decision-making task (Fig. 1A; Methods and Materials – Neural Data – Task and Stimuli) that the animals were trained on (Seo et al., 2012). The system was composed of two networks, a striatal and a prefrontal network (Fig. 1E). The striatal network (STR) received actions and rewards as inputs and produced updated value signals for each of the ten possible actions (Fig. 3 as outputs. The prefrontal network (PFC) received value signals as inputs, together with supplementary information indicating move/hold and fixation signals, and produced actions for the next step as outputs. The two networks were connected through their outputs only, but had separate weight matrices. This is explained in greater detail in the *Architecture* section. Training details can be found in the *Network system training* section.

We obtained the value signals by fitting a reinforcement learning (RL) algorithm (specifically Q-learning) to the behavioral choice data obtained from the two animals. More specifically, we obtained an action value signal that tracked the animal's perception of reward associated with each particular choice as indicated by his behavior. This is described in greater detail in the *Task Coding* section below.

The RL-derived value signals were then used together with the action and reward signals to train the system of networks in a supervised manner. Actions and rewards were obtained from the actual experimental runs, augmented with artificial runs (as described in the *Task Coding* section). Thus, during the training phase, the networks had to repeatedly produce blocks of movement-sequences, just like the experimental animals. Thereby, the striatal network learned to map changes in the action and reward signals (over the course of performing sequential movements with varying outcomes) to changes in the value signal. The prefrontal network learned to make use of changes in value signals (together with a few other supplementary inputs) to produce updated actions and generate movement-sequences. Altogether, the network system was taught to internalize the learning process that the experimental animals exhibited over the course of a block of movement-sequences. The system was taught to appropriately update value and action signals by providing full trials as a supervised training signal.

After training, the system was tested both by providing full (held-out) test trials as well as by running the system in autonomous mode. In the latter case, the system was left alone to produce a whole block of movement-sequence trials. Thereby, action-signals from the previous step were fed back as inputs to the striatal network, which produced an updated value signal that was, in turn, fed back as input to the prefrontal network, allowing it to produce the action for the next step. This is described in greater detail in the *Autonomous* section.

Finally, the model system was subjected to the same analysis as the neural data (as outline in the *Neural Data Analysis* section). The model system was then probed further to reveal how it produced decisions. This is described in greater detail in the *Model Analysis* section.

2.2.2. Architecture

We jointly trained a connected system of two recurrent neural networks to perform the movement-sequence task (see Methods and Materials). Single-unit dynamics in these networks are governed by:

$$\begin{aligned}
\tau\,\dot{\mathbf{x}}^s(t) &= -\mathbf{x}^s(t) + \mathbf{W}^s_{rr}\mathbf{r}^s(t) + \mathbf{W}^s_{ir}[\mathbf{u}_a(t), \mathbf{u}_r(t)] + \boldsymbol{\eta}(t) \\
\mathbf{r}^s(t) &= \mathbf{tanh}(\mathbf{x}^s) \\
\mathbf{u}_v &= \mathbf{W}^s_{ro}\mathbf{x}^s \\
\tau\,\dot{\mathbf{x}}^p(t) &= -\mathbf{x}^p(t) + \mathbf{W}^p_{rr}\mathbf{r}^p(t) + \mathbf{W}^p_{ir}[\mathbf{u}_v(t), \mathbf{u}_{ins}(t)] + \boldsymbol{\eta}(t) \\
\mathbf{r}^p(t) &= \mathbf{tanh}(\mathbf{x}^p) \\
\mathbf{u}_a(t_{+1}) &= \mathbf{W}^p_{ro}\mathbf{x}^p
\end{aligned} \quad (2)$$

where $x^s_i(t)$ and $x^p_i(t)$ are synaptic current variables of unit i at time t in the striatal and prefrontal network, respectively, and activity (firing rate variables r^s_i and r^p_i) is a nonlinear function of x ($r^s_i = \tanh(x^s_i)$ and $r^p_i = \tanh(x^p_i)$), $\mathbf{W}^s_{rr}$ and $\mathbf{W}^p_{rr}$ are the recurrent weight matrices, $\mathbf{W}^s_{ir}$ and $\mathbf{W}^p_{ir}$ are the input weight matrices, $\mathbf{W}^s_{ro}$ $\mathbf{W}^p_{ro}$ are the output weight matrices, and $[\mathbf{u}_a, \mathbf{u}_r]$ and $[\mathbf{u}_v, \mathbf{u}_{ins}]$ are the inputs of the striatal and prefrontal networks, respectively, and added noise $\eta_i(t)$.

The striatal and the prefrontal network had $N^s_{rr} = 1300$ and $N^p_{rr} = 1000$ units, respectively. The connectivity weight matrices $\mathbf{W}^s_{rr}$ and $\mathbf{W}^p_{rr}$ were initially drawn from the standard normal distribution and multiplied by a scaling factor of $\frac{g}{\sqrt{N^s_{rr}}}$ and $\frac{g}{\sqrt{N^p_{rr}}}$, respectively, with $g = 1.0$. The neural time constant was $\tau = 10$ ms. Each unit received an independent white noise input, η_i, with zero mean and $SD = 0.01$. Inputs were fed into the networks through the input weight matrices $\mathbf{W}^s_{ir}$ and $\mathbf{W}^p_{ir}$ which were initially drawn from the standard normal distribution and multiplied by a scaling factor of $\frac{1}{\sqrt{N^s_{ir}}}$ with $N^s_{ir} = 20$ and

$\frac{1}{\sqrt{N_{ir}^p}}$ with $N_{ir}^p = 510$ for the prefrontal network. Outputs were read out through the weight matrices $\mathbf{W}_{ro}^s$ and $\mathbf{W}_{ro}^p$, which were initially drawn from the standard normal distribution and multiplied by the same scaling factors as the input weight matrices. Otherwise, the model parameters were set in the same range as (Mante et al., 2013).

The striatal network received a 15-dimensional input vector, $\mathbf{u}_s = [\mathbf{u}_a, \mathbf{u}_r]$, composed of a 10-dimensional vector $\mathbf{u}_a$ specifying actions taken and a 5-dimensional vector $\mathbf{u}_r$ specifying the rewards received for those actions. Outputs for each of the networks were linearly read out from the synaptic currents of the recurrent circuit Eq. (2). The striatal network read-out was a 10-dimensional vector, $\mathbf{u}_v$, of action values derived from TD learning (as described further below).

The prefrontal network received a 20-dimensional input, $\mathbf{u}_p = [\mathbf{u}_v, \mathbf{u}_{ins}]$, composed of a 10-dimensional vector $\mathbf{u}_v$ of action value outputs from the striatal network, together with a 10-dimensional instruction vector $\mathbf{u}_{ins}$ specifying when to initiate fixation (Fixate) and when to move towards or hold the target (Move + Hold). The prefrontal network read-out was a 11-dimensional output vector, $\mathbf{u}_a = [\mathbf{u}_a, u_{vis}]$ composed of a 10-dimensional vector $\mathbf{u}_a$ of action outputs for the next step, and an additional unit $\mathbf{u}_{vis}$ coding for the visual hemifield (upper or lower) in which a particular sequence terminated.

2.2.3. Network system training

All synaptic weight matrices ($\mathbf{W}_{rr}$, $\mathbf{W}_{ir}$, and $\mathbf{W}_{ro}$) were updated with the gradient of the loss function Eq. (3), which was designed to minimize the square of the difference between network and target output:

$$l = 2\sum_{k=1}^{K}\sum_{t=0}^{T}\sum_{i=1}^{N_{rrs}}(\hat{y}_i - y_i)^2 + \sum_{k=1}^{K}\sum_{t=0}^{T}\sum_{i=1}^{N_{rrp}}(\hat{y}_i - y_i)^2 \tag{3}$$

The error was thus obtained by taking the difference between network output y and target output $\hat{y}$ and summing over all trials K in a batch (with $K = 10$), time points T and recurrent units N_{rr}. The total loss function Eq. (3) was obtained by combining the loss terms from the striatal and the prefrontal network while assigning double weight to the striatal loss term. The striatal and the prefrontal networks were jointly trained by obtaining the gradient of the combined loss function Eq. (3) through automatic differentiation with *autograd* (Maclaurin et al., 2017) and custom implementations of specific functions with GPU-based acceleration using *JAX* (Johnson et al., 2018). The network was trained with an initial learning rate of $\alpha = 0.001$ for 10 steps with 1000 iterations each, while the learning rate was decayed by $\frac{2}{3}$ at every step. After this training phase, all the synaptic weight matrices remained fixed.

2.2.4. Task coding

Actions ($\mathbf{u}_a$) and action values ($\mathbf{u}_v$) were coded as 10-D vectors in which every unit coded for one of the movement choice options (Fig. 1C; see Methods and Materials — Task and Stimuli). So, for instance, for the first binary choice option (Fig. 1C, S1-center) one unit coded the right movement choice and the other the left movement choice (as depicted in Fig. 2). Similarly, there was a pair of units for each of the following binary choice options (Figs. 2, 3). Thus, there were 10 units in total for the 10 possible movement directions (1C). Rewards ($\mathbf{u}_r$) and visual inputs which indicated target locations and fixation points ($\mathbf{u}_{ins}$) were coded as 5-D vectors with every unit coding for reward delivered at one of the 5 decision points (center, upwards, downwards, upper, lower; Fig. 1C). Actions and rewards were coded as brief transients. The reward signal interval was $\frac{2}{10}$th as long as the action signal interval. Rewards were delivered after the end of the action signal interval.

Value signals ($\mathbf{u}_v$) were derived by fitting a reinforcement learning (RL) algorithm to the choices of the experimental animals. In this algorithm, rewards drove the update of action values according to a temporal-difference reinforcement learning algorithm (Q-Learning) (Sutton & Barto, 2018):

$$\begin{aligned} Q(s_t, a_t) &\longleftarrow Q(s_t, a_t) + \alpha\,[r_{t+1} + \gamma\, max(Q(s_{t+1}, :)) - Q(s_t, a_t)] \\ \mathbf{p}_t &= softmax(\beta\, \mathbf{Q}_t) \\ \mathbf{Q}_{t+1} &= \tau\, \mathbf{Q}_t \end{aligned} \tag{4}$$

The learning rate parameter α, the discount factor γ and an additional inverse temperature parameter β were fitted to one of the training sessions of monkey 1 using *fminsearch* in Matlab, with the decay parameter set to $\tau = 0.95$. The values obtained for the parameters were $\alpha = 0.8100$, $\gamma = 0.2010$, and $\beta = 3.050$.

The Q-learning algorithm thus yielded an action-value signal ($Q(s_t, a_t)$, (4)) for each particular state and action available at a particular point in time. This value signal reflects the animal's perception of the value of a particular action given the association with a reward. Upon delivery of a reward, the value signals increases, while upon omission, it decreases rapidly or stays flat (if no previous rewards had been delivered). Thus, if an animal makes a mistake and chooses a left movement instead of a right movement, it receives no reward, and the value signal reflects this omission (Fig. 2). Mistakes thus feature naturally in the value signal, and the error distribution of the experimental animals is reflected in the training dataset for the network system.

The full training dataset for the corticostriatal model system was composed of these Q-learning derived value signals together with the choices performed by the experimental animals and the rewards delivered during the experimental runs. The training set thus mimicked the trial structure of the original experiment. A subset of 25 blocks from the experimental runs was left out as a test set. Additionally, the training dataset was augmented with artificial data generated by randomly choosing one of the eight sequences for the current block and drawing action movement outcomes with the same error probabilities that the animals displayed in the real task (see Fig. 5D — Behavior). During training, a batch composed of 10 blocks was randomly chosen from across the entire training set at each step.

2.2.5. Autonomous mode

After training, we also tested the model's ability to autonomously produce movement-sequence blocks. This meant obtaining trial-by-trial value and action estimates. More specifically, we obtained an initial set of outputs ($\mathbf{u}_v$; Eq. (2)) from the striatal network by feeding in a vector of white noise inputs for $\mathbf{u}_a$ (with zero mean and $SD = 0.01$). Striatal outputs were then fed into the prefrontal network together with instructions $\mathbf{u}_{ins}$ specifying the particular movement stage as part of the prefrontal input vector $\mathbf{u}_p$. We obtained a vector of actions $\mathbf{u}_a(t_{+1})$ for the next step as outputs from the prefrontal network. Actions were decoded from among the choice options that were possible at a particular movement stage (after the first movement stage, certain choice options become unavailable, i.e. for sequences S1 and S5 in Fig. 1C the choice options in the lower visual hemifield are no longer possible at the second movement stage). Actions were decoded probabilistically from the result of passing action outputs through a softmax function ($\mathbf{a}_{choice} = softmax(\mathbf{u}_a)$).

The newly obtained action vector ($\mathbf{u}_a(t_{+1})$) was then concatenated with the previous action vector ($\mathbf{u}_a(t)$) and fed back into the dSTR network. If the action was correct, a reinforcement signal was delivered to the dSTR. If the action was incorrect, no

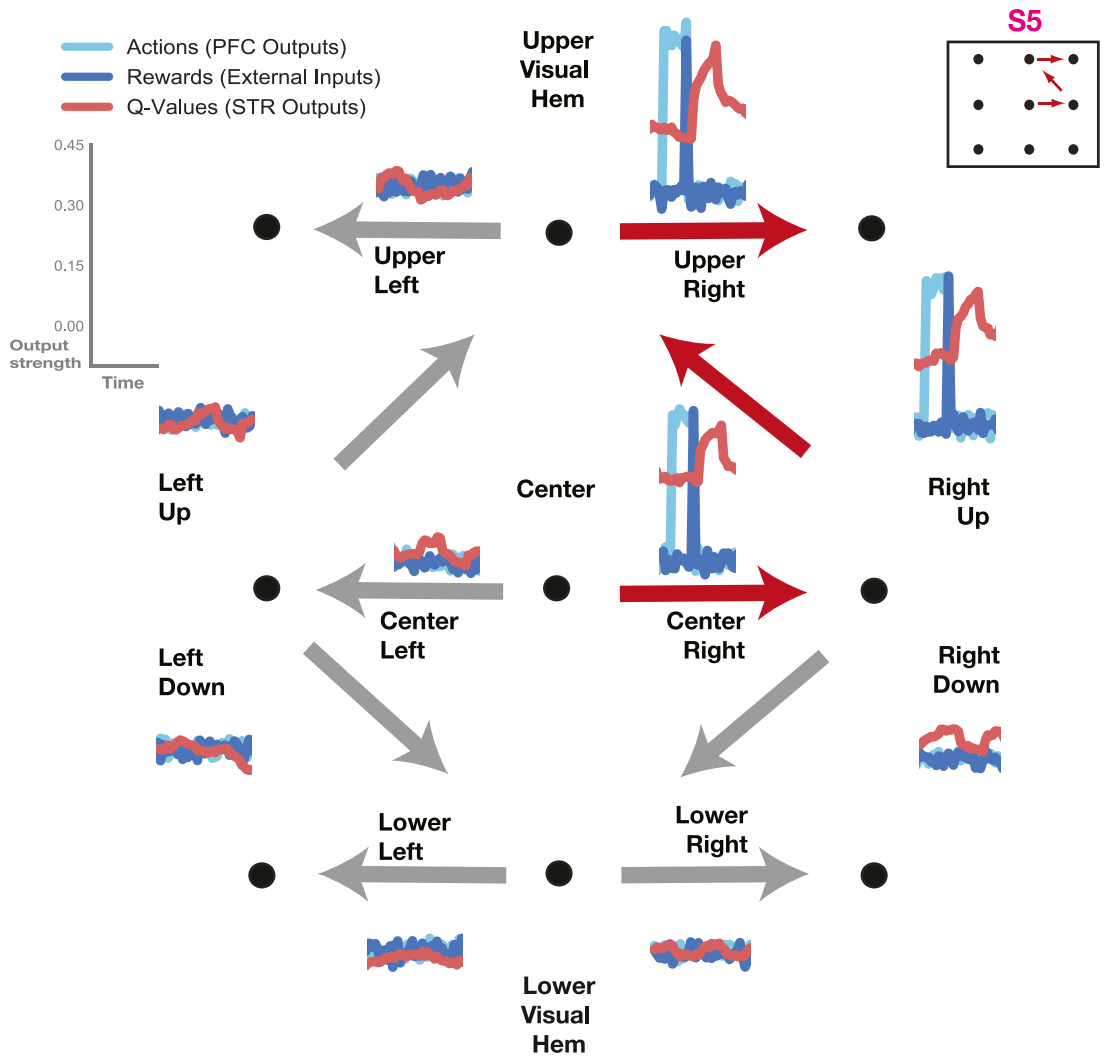

Fig. 2. Task coding. The correct movement sequence (S5) was signaled by the corresponding output units while units coding other movement directions stay flat. Actions (light blue) were indicated by pulses in the prefrontal network's output units corresponding to a particular direction. Rewards (dark blue, plotted on top) were delivered at the end of action pulses. Action values (red, plotted on top) were indicated by striatal output units corresponding to a particular direction; action values increased after reward delivery. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

reinforcement signal was delivered. Then, the signal for the next movement stage was fed into the dPFC network and a new set of dSTR outputs, for the next movement stage, was fed into the prefrontal network, etc. After the network system performed a complete block of movement-sequence trials, the action vector was reset to white noise and a new block could begin.

If the decoded action was wrong (in that it was not the correct action for the current movement-sequence block), the network system was forced to repeat this particular movement stage before proceeding (just like the animals in the original experiment). This was done by concatenating the set of wrong actions with the previous action vector, and feeding it back into the striatal network together with the reward vector $\mathbf{u}_r$ set to white noise. The striatal outputs obtained as such were then fed into the prefrontal network together with the instruction vector $\mathbf{u}_{ins}$ set to the same movement stage as before.

2.2.6. Model analysis

The neural population activity from the striatal and prefrontal networks of the corticostriatal model was imaged in the same way as the real neural recordings, by projecting activity into a 3-dimensional latent space spanned by the first three principal components of the *sequence*-subspace ($\mathbf{D}_s$) obtained through dPCA (Eq. (1)) (Brendel et al., 2011; Kobak et al., 2016).

Canonical correlation analysis (CCA) was used to compare model and neural population responses, as was done previously (Sussillo et al., 2015). First, both the model and the recorded neural data were averaged across trials, smoothed with a Gaussian kernel and reduced to 15 dimensions using dPCA. This ensured CCA was not performed along dimensions of high correlation but low variance. Then the first 10 canonical correlation coefficients were obtained using the entire duration of a chosen movement-sequence as a comparison window.

In order to image the potential surface (Figs. 7 and 9), we first projected neural population activity into 3-dimensional latent space and obtained a mesh of points ($\mathbf{X}^*$) around the sample trajectories. Then we projected this mesh of points back out into neural population space using the encoder matrix of the *stimulus*-subspace ($\mathbf{F}_s$; Eq. (1)). We started the networks off at each of these points by providing them as the initial vector of firing rates ($\mathbf{r}(\mathbf{t})$; Eq. (2)), and let the network system run through a whole block of movement-sequence trials. For illustration purposes, we imaged the potential surface at the same point for all trials in a

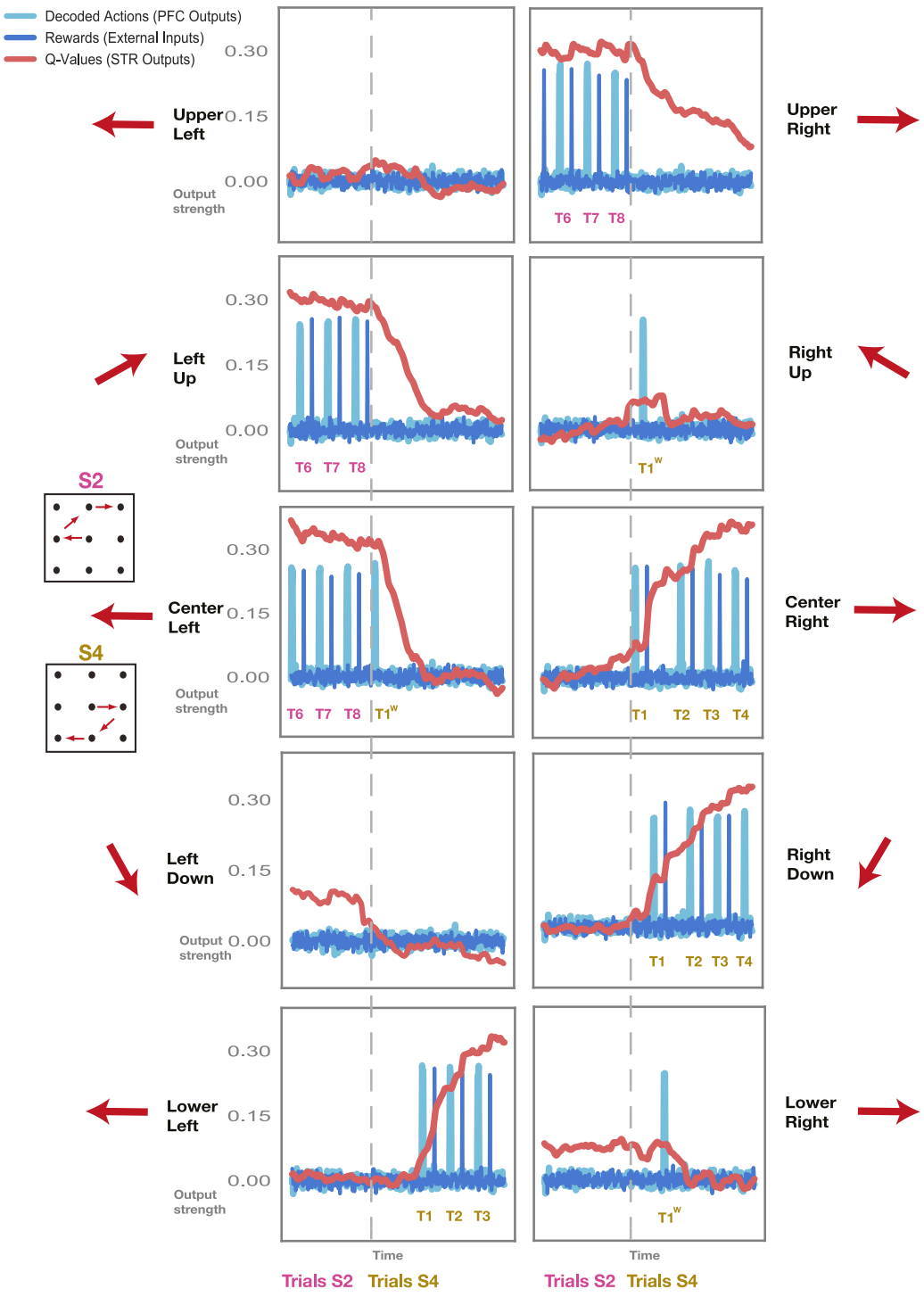

Fig. 3. Corticostriatal model: autonomous behavior. Sample transition between two blocks in which the correct sequence switches (from S2, in magenta, to S4, in light brown; block transition indicated by vertical gray dotted line). Decoded Actions (light blue) are plotted together with rewards (dark blue) and action values (red) for all output units. Action values spike after correctly executed, rewarded actions and increase with successive correct actions. Action values for wrong, unrewarded moves decay quickly. Trials with erroneous movements remain unrewarded and are repeated, just like in the original experiment. Supporting Figures: Fig. S1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

block. We obtained the potential surface at the chosen moment in time by calculating the magnitude of the gradient for the mesh of points. In order to ensure that the trough of the potential surface at the chosen moments in time really pointed to fixed point regions, we kept the input fixed and continued running the network system through 10 iterations (which corresponded to the length of the fixation period) to make sure the location of the trough remained fixed on the timescale of the network. If the location of the minimum remained fixed for the length of this period, we labeled this location as a fixed point. We then obtained the joint potential surface of two chosen sequences

by taking the point-wise minimum across the two sequences' manifolds.

In order to calculate the distance between the minima of the potential surfaces of two different sequences, we first confirmed the location of the fixed points in 3-dimensional latent space as described above. Then, we used Dijkstra's algorithm to obtain the minimal path length along the joint potential surface between two particular sequences' fixed points. We repeated this for all possible sequence combinations in the test set and averaged the result. In order to calculate the distance between sequences in latent space as well as the distance of a particular sequence to its centroid, we first projected the data into 10-dimensional sequence dPC-subspace (for computational reasons) and then proceeded the same way as in the neural data (see Methods and Materials – Neural Data – Data Analysis). We found no difference in these metrics when projecting into 10- or 20-dimensional dPC-space.

3. Results

We investigated dynamics during learning in the fronto-striatal system (Fig. 1). The task consisted of a sequence of three movements which the animal had to execute by saccading to the correct targets (Fig. 1A). While the animal executed these movement trajectories, recordings were obtained from dPCA and dlPFC (Fig. 1B). There were a total of 8 possible arrangements (or sequences) of 3 movements (Fig. 1C). These sequences, in turn, were arranged into blocks in which one particular sequence remained fixed for the duration of the block (Fig. 1D). Thus the animal was able to learn which sequence was correct in the current block using feedback about chosen movements. At the start of a new block, a new sequence was chosen randomly (see Methods and Materials for further details).

Two animals performed this task while we obtained neural recordings from lateral prefrontal cortex (lPFC) and dorsal striatum (dSTR) (Fig. 1B, see Methods and Materials and Seo et al., 2012 for further details). To investigate learning dynamics, we built a model of the fronto-striatal system. In this model, consistent with the neural data, the striatum represents action values, and the prefrontal cortex selects actions (Fig. 1E). Whether actions are selected via return loops through the thalamus, or in other structures downstream from the striatum is currently unclear. However, at least in some conditions in the in-vivo study, actions were selected in cortex before they were selected in the striatum (Seo et al., 2012). Therefore, we organized our model consistent with this. The prefrontal network received action value information from the striatum and selected actions based on this information. The prefrontal network received additional inputs signifying fixation and move/hold periods, which were presented to the animal as visual cues during the task. The network system was trained to produce movement-sequences arranged in a block of sequential trials, akin to the experimental animals. The training set consisted of actions performed by the two animals during the experiment and the rewards received during the experiment, plus a corresponding set of action values. These action values were generated by feeding actions and rewards through a temporal-difference reinforcement learning algorithm (Q-learning). All of this is explained in greater detail in Methods and Materials — Corticostriatal model. Altogether, the network system was taught to learn the association between actions, rewards, and value updates over the course of a block of movement-sequences.

After training, the fronto-striatal model learned to produce correct movement sequences (Fig. 2). In this example, sequence S5 (consisting of a rightwards movement, followed by upwards, and another rightwards movement) was the correct sequence for the current block. The prefrontal network units coding for these movements selected the correct action (light blue). Following selection of the correct action the network received a reward input (dark blue). The reward, in turn, made the value signal in the striatal network (red) increase for the rewarded movement. The output of the prefrontal and striatal network units coding for other movement directions remained flat. Altogether, the system learned to produce movement sequences with the correct action and action value output.

After training, the network system could be run autonomously (Fig. 3; see Methods and Materials – Corticostriatal Model – Autonomous mode). That is dlPFC selected actions based on the dSTR action value inputs and external inputs to dlPFC indicating the ordinal position of the current movement in the sequence. In the first step, a vector of white noise inputs and a vector of external inputs specifying the ordinal position of the movement were used to generate value signals in the striatal network. These value signals were input to the prefrontal network which selected the first movement. This movement was then fed back into the striatal network together with the corresponding reward outcome. These signals were used to generate the next value signal, and so on. In this manner, we generated several trials of the same movement sequence, which were strung together into blocks, as in the original experiment (Fig. 1D). During these blocks the system received rewards and ordinal information as the only inputs, and it reacted by adjusting value signals and actions accordingly. Thus, in brief, after training the network system was able to perform blocks of movement-sequences autonomously, just like the animals in the original experiment. In essence, the network has successfully internalized the learning dynamics that unfold over the course of a block.

We imaged the outputs of the network system as the correct sequence switched from one block to the next (Fig. 3). There were a total of 10 output units, one for each of the available movement directions (see Methods and Materials – Corticostriatal Model – Task Coding). Decoded actions (in light blue) from the outputs of the prefrontal network, action value outputs (in red) from the striatal network and rewards (in dark blue) delivered externally were all imaged together for these two sequential blocks. The correct sequence switched from one block to the next (sequence S2, magenta, was correct in the first block, and S4, light brown, was correct in the second; Fig. 3 inlays).

The correct sequence for the first block (S2, magenta) is composed of a left move in the center, followed by an upwards move on the left side and a rightwards move at the top (see Fig. 3 upper inlay on the left). Accordingly, the three output units at center left, left up and upper right were active during the first block (left of the gray dotted line), indicating the three sequential movements that made up the correct sequence (S2). The correct actions could be decoded probabilistically from the outputs of the prefrontal network; actions were signaled by sequential pulses (light blue) in the three output units that made up the correct sequence. Rewards were presented as short pulses at the end of an action (dark blue). The striatal network subsequently produced the corresponding action value signal (red). The value signal decayed over time, but recovered when correctly executed trials followed upon each other (as in the panel on the upper right, Fig. 3).

As the new block began (right of gray dotted line), the correct sequence changed (from S2 in magenta to S4 in light brown, Fig. 3 lower inlay on the left). The new sequence was composed of a center right move, followed by a downwards move on the right and a move to the left at the bottom, ending up in the lower visual hemifield. Accordingly, the three output units at center right, right down and lower left were now active (right of gray dotted line), indicating the three sequential movements that made up the new correct sequence (S4). The output units representing the

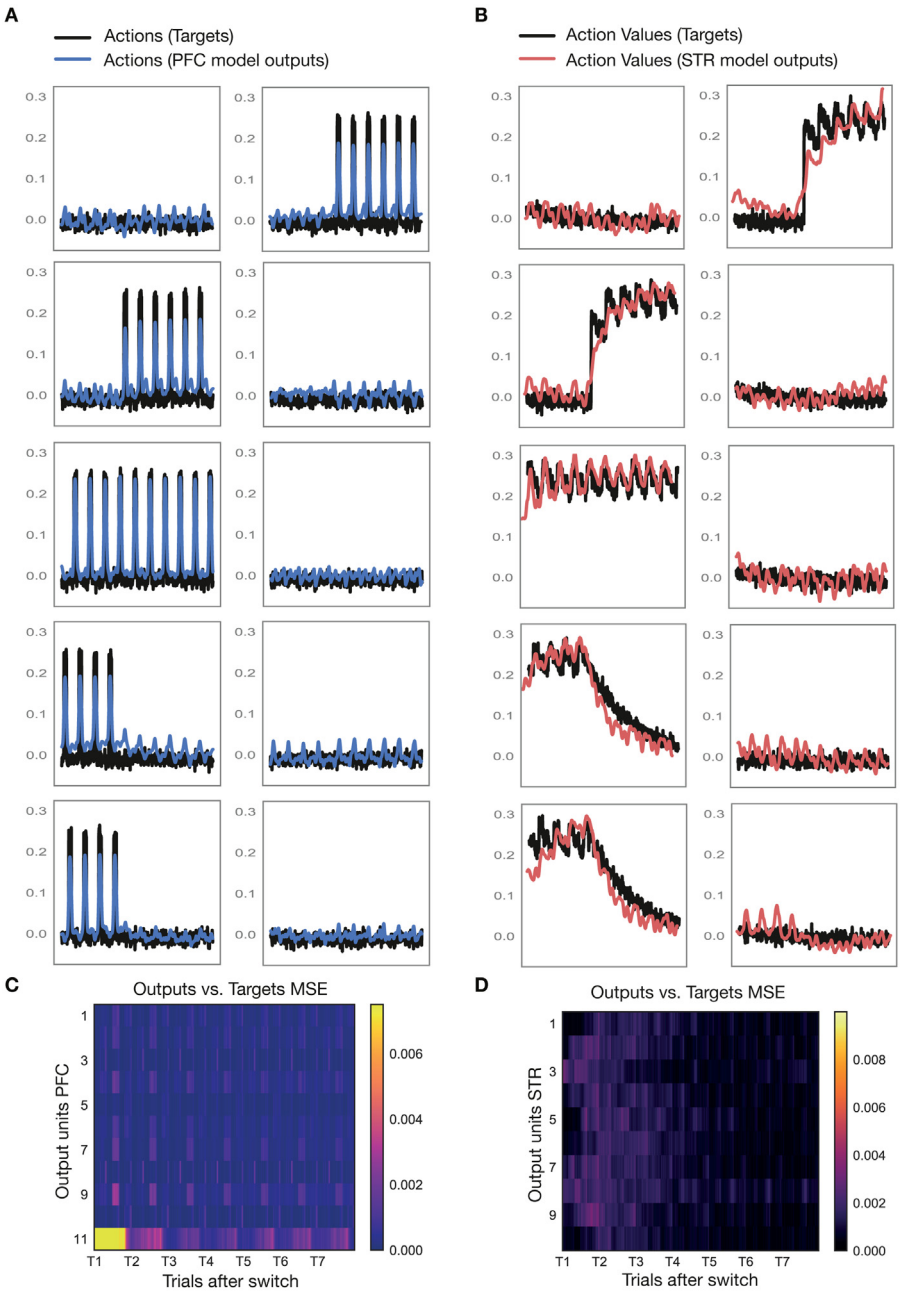

Fig. 4. Model Performance. A Outputs of prefrontal model network (blue) versus targets (black) for a sample test block. **B** Outputs of striatal model network (red) versus targets (black) for the same sample test block as in A. **C–D** Mean squared error (MSE) between outputs and targets averaged over 25 test blocks. MSE is depicted as a function of trials after the sequence switched for all output units in the prefrontal (C) and striatal network (D). Supporting Figures: Fig. S2. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

movements that had been correct during the previous sequence (center left, left up, upper right) were now inactive; the striatal network's value signal in these units decayed back towards zero.

Occasionally, the prefrontal network committed errors and the wrong movement was decoded. Errors occurred with the same frequency as in the behavioral data from the two monkeys carrying out this task (as the training set for the networks was derived from real behavior, see Methods and Materials – Corticostriatal model – Overview; Fig. 5D). Upon a block switch, errors could occur at all three movement stages. At the first movement stage, the PFC network produced the same movement which had been correct in the previous block (Fig. 3 center left panel, $T1^{w}$ in light brown). This move remained unrewarded. The striatal value signal decayed quickly after unrewarded moves, instead of spiking as it did after rewarded moves (e.g. center left, T8 in magenta). The network was forced to repeat the move before progressing further (just like the animals were in the original experiment; see Methods and Materials – Corticostriatal model – Autonomous mode). The striatal value signal for the opposite movement direction now increased, prompting the prefrontal network to produce the correct move at the second attempt (center right panel, T1 in light brown). The correct action was now rewarded and the striatal value signal for this move increased further (center right panel), while the value signal for the unrewarded action decayed further (center left panel). Further erroneous moves occurred at the second and third movement stages (right up and lower right

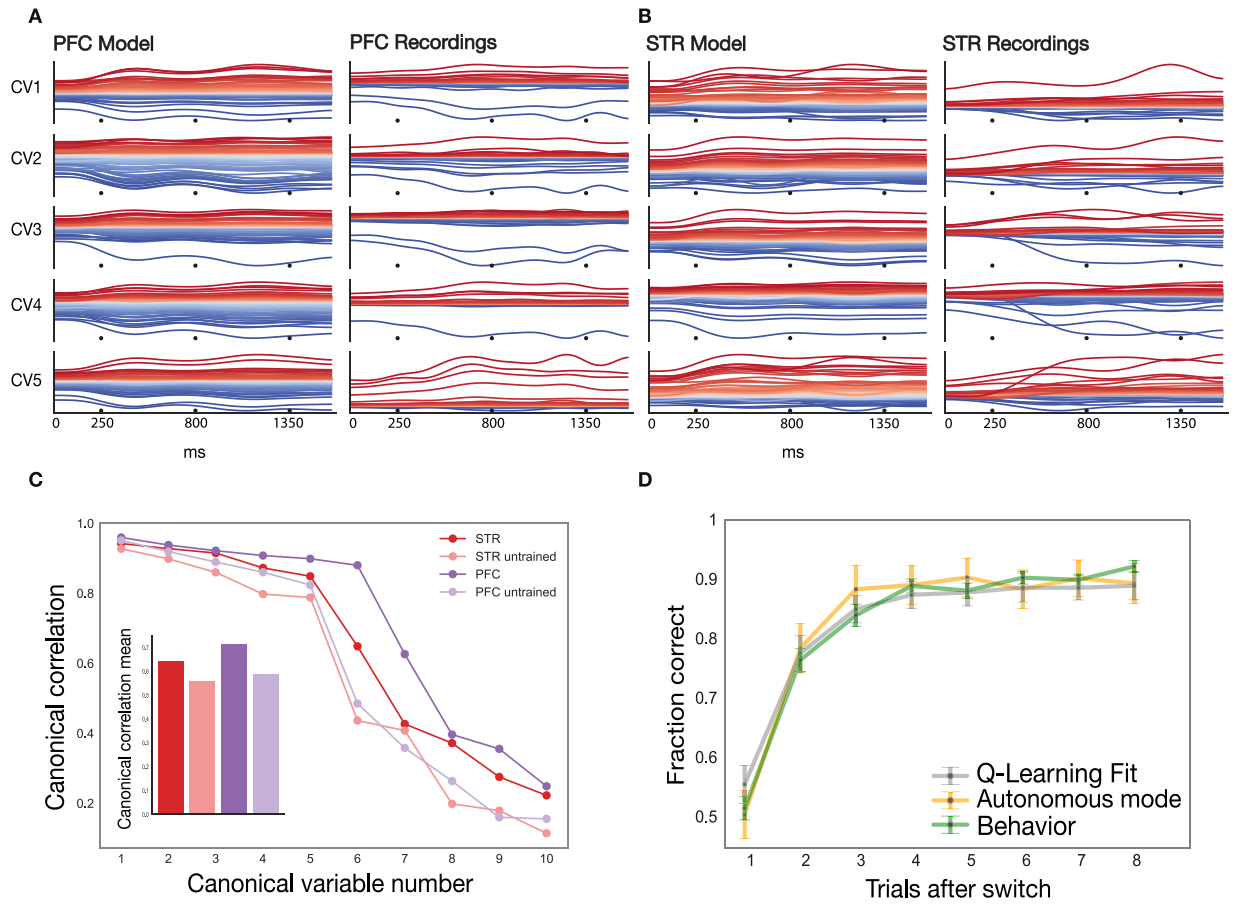

Fig. 5. Comparison of model and recorded data A CCA projections (canonical variables) for the *striatal* model network (left) and the neural data recorded from dSTR (right). These projections show the directions in state–space along which the data is maximally correlated with the model. Each row shows one canonical variable (CV 1–5). Traces are colored based on the mean value of the projection across the entire depicted duration of the trace. Traces show responses for the whole length one movement-sequence which is composed of three sequential movements (black dots mark movement onset). **B** CCA projections for the *prefrontal* model network (left) and the neural data recorded from PFC (right). **C** Summary of canonical correlations between model and neural data. CCA analysis provides a spectrum of correlation coefficients that can be used to assess model performance (see Methods and Materials). The canonical correlation coefficients are shown for the trained model (striatal network in dark red and prefrontal network in dark magenta), as well as for an untrained network with the same inputs which as a baseline (striatal network in light red, and prefrontal network in light magenta). **D** Fraction of correct decisions as a function of trials after the sequence switched, separately for the behavior of the two animals (solid line), the performance of the Q-learning algorithm (broken line; see Methods and Materials – Corticostriatal model – Coding), and the performance of the corticostriatal model network system (dotted line; see Methods and Materials – Corticostriatal model – Autonomous Mode). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

panels, respectively). These outputs remained unrewarded, and the network was again forced to repeat these moves. The correct moves were generated at the second attempt, and the network was allowed to progress to the next trial.

To analyze the system's performance in more detail, we plotted striatal and prefrontal outputs together with their target outputs for a sample block transition from the test set (Fig. 4A, B; see Methods and Materials – Corticostriatal model – Overview). In this case, actions from the test set (a portion of the original dataset which was not used in network training) were fed into the striatal network to obtain value signals which, in turn, were fed into the prefrontal network to obtain action outputs. Actions were not fed back into the striatal network this time; rather, the next action from the test set was picked as the next input to the striatal network. The network behaved similarly to the autonomous mode, with value signals increasing after successive correctly executed moves. Output traces generally follow target traces. To quantify this, we computed the mean squared error between targets and outputs over the duration of a block averaged over all blocks in the test set (Fig. 4C–D). The largest difference between outputs and targets occurred for the first trial when the correct sequence was unknown. Over the course of the block, as certainty increased, the error decreased for both prefrontal (Fig. 4C) and striatal networks (Fig. 4D). We also imaged the eigenvalue spectra of the two networks' weight matrices after training (Fig. S1). We found that variance was spread across many dimensions in the prefrontal network (Fig. S1A), while most of the variance was concentrated around 5 large eigenvalues in the striatal network (Fig. S1B).

We compared neural activity from the model with the recorded neural data using canonical correlation analysis (Fig. 5A–C; see Methods and Materials – Corticostriatal model – Model Analysis). We plotted the first five canonical variables for the neural responses from the model and the recorded data (Fig. 5A, B). The canonical variables are the directions in state–space along which model and recorded data are maximally correlated, and offer a way to assess the similarity between model and recorded population responses. Neural activity was reconstructed separately from each of the first five canonical variables. The model (left) and the neural data (right) shared many population-level response patterns. The model was able to pick up the slow oscillatory patterns observed in the neural data (Seo et al., 2012). We also plotted the

canonical correlation coefficient for the striatal and the prefrontal network across the first ten canonical variables (Fig. 5C). As a comparison, we obtained the correlation coefficients for an untrained model (a model that received the same inputs, but did not undergo supervised training). The first ten canonical correlation coefficients of all models were significant ($p < 1.0e-3$, MANOVA Wilks lambda statistic), meaning the first ten axes of the CCA captured most of the similarity between model and neural data. The canonical correlation coefficients were consistently higher for the trained than the untrained model, for both the striatal (STR) and the prefrontal (PFC) network. The first seven correlation coefficients of the trained striatal and prefrontal model networks were significantly higher than those of the untrained model networks ($p < 4.823e-11$, Z-statistic, Fisher's z-transformation). This shows that the trained model system is able to capture a lot of the detail in the neural dynamics. We also computed the average correlation coefficient across the first ten canonical dimensions (Fig. 5C inlay). The average correlation coefficients were overall higher for the trained networks (0.65 striatal and 0.71 prefrontal) than for the untrained networks (0.56 striatal and 0.59 prefrontal). We retrained the model 10 times with the same configuration and parameter settings, but starting with different randomly initiated weights (see Methods and Materials), and obtained a similar fit to recordings.

We also determined the behavior of the model system by measuring the fraction of correct decisions over the course of the movement-sequence block and compared it to the recorded behavior from the animals (Fig. 5D). The recorded behavioral data from the animals (solid line, as in Seo et al. (2012)) shows chance performance at the beginning of the block, and rapid, steady improvement over the course of the block. In order to determine the fraction of correct decisions of the model system, we let the networks produce movement-sequences autonomously (see Methods and Materials – Corticostriatal model – Autonomous mode). That is, we used the action outputs of the prefrontal network as inputs to the striatal network. In this way we obtained a distribution of activations for the next movement step in the prefrontal output units, from which we decoded the predicted action. The fraction of correct decisions of the autonomous model system averaged over 100 blocks (orange line) approximated that of the animals' behavior (green line). There was no significant difference between the outputs of the autonomous model and the behavior of the animals across the duration of the block. We also plotted the fraction of correct decisions obtained from the fit of the Q-learning algorithm (gray line), and it also showed no significant difference to the behavior of the animals (green line). This shows the model system was able to capture the animals' behavior in this task.

To study how neural representations evolved with learning, we imaged PFC neural population activity in 3-dimensional latent space using demixed principal component analysis (see Methods and Materials). Trajectories from neural recordings in dlPFC (Fig. 6A–D) are plotted alongside trajectories from the prefrontal network of the corticostriatal model (Fig. 6E–H). The plots capture representations for progressive trials over the course of a block (Fig. 1D), as the animals advance from 50% certainty at the start of the block (Fig. 6A&B) to close to 90% certainty by the fifth trial into the block (Fig. 6C&G).

Sequence representations from neural recordings in dlPFC (Fig. 6A–D) showed a separation by visual hemifield: sequences S1, S2, S5 and S6 which progressed along the upper visual hemifield (Fig. 1C) were clustered to the left while the remaining sequences which progressed along the lower visual hemifield were clustered to the right. This separation appeared to be present from the very start of a block. We also examined the evolution of representations across learning. As the animals learned in each block, they more frequently selected the correct option (Fig. 5D), which suggests they become more certain of their choices. We found that sequence trajectories within each particular cluster separated more from each other with increasing certainty (Fig. 6A–C). To capture this effect, we computed the Euclidean distance between sequences within clusters in neural population space (in the full-dimensional space of recordings, not in the reduced latent space; see Methods and Materials). We found this measure increased with certainty as learning progressed over the block (Fig. 6D). Sequence representations from the prefrontal network model (Fig. 6E–H) showed the same clustering by visual hemifield. We confirmed that there was a significant separation by visual hemifield (strongly colored traces belong to upper, lightly colored traces to lower hemifield) across the whole dataset ($p < 5.1e-5$, paired t-test, Bonferroni corrected), for the three certainty levels displayed in both the neural data and model (Fig. 6A–G). Task trajectories were somewhat more distinctly clustered by hemifield at the start in the neural data (Fig. 6A) than in the model (Fig. 6E). Overall, though, we also found Euclidian distance between sequences within clusters increased for the model trajectories (Fig. 6H). We confirmed that there was a significant increase in distance ($p < 1.0e-3$, paired t-test, Bonferroni corrected) between the sequence trajectories at the first two certainty levels (50%, 76%) and those at a high certainty level towards the end of the block (88%–91%), for both the neural data ((Fig. 6D) and the model (Fig. 8H).

To study what underlies increasing separation of movement-sequence representations with learning, we probed the prefrontal model network further (Fig. 7). We obtained the potential surface in the region around the latent sequence trajectories as learning progressed across the block (see Methods and Materials — Model Analysis). We obtained the joint surface across two particular movement-sequence trajectories in latent space by taking the point-wise minimum across the potential surfaces of the individual sequences, and imaged this common surface for increasing levels of certainty across a block (Fig. 7A–C). The potential surface shows how activity evolves at different points in neural latent space in the vicinity of sequence trajectories. Neural activity has a high propensity to be pushed away from locations where the magnitude of the gradient is high (yellow), and remain in locations where the magnitude is low (dark blue). We ascertained that the troughs of the potential surface pointed to energy minima (or fixed points) in neural activity (see Methods and Materials — Model Analysis).

Observing the evolution of joint potential surfaces with learning (Fig. 7A–C) for a particular point along the movement trajectory (magenta dot), one notices how energy minima for different regions become increasingly well separated. Along with this separation, the ridge in the joint potential surface between the two sequence minima heightened with increasing certainty during learning. As this happens, it becomes increasingly less likely to commit errors: the gradient along a particular side of the ridge drives activity more strongly towards a particular sequence's fixed point region, so that the chance to end up in the other minima decreases with learning. To quantify this effect, we determined the minimal path length between the various sequence's fixed point regions as learning progressed (see Methods and Materials — Model Analysis). We observed that minimal path length between fixed point regions increased with certainty during learning (Fig. 7D). We confirmed that there was a significant increase in distance ($p < 1.0e-3$, paired t-test, Bonferroni corrected) between the sequence trajectories at the first two certainty levels (50%, 76%) and those at a high certainty level towards the end of the block (88%–91%). Altogether, we established that the energy landscape in the network changes with learning such

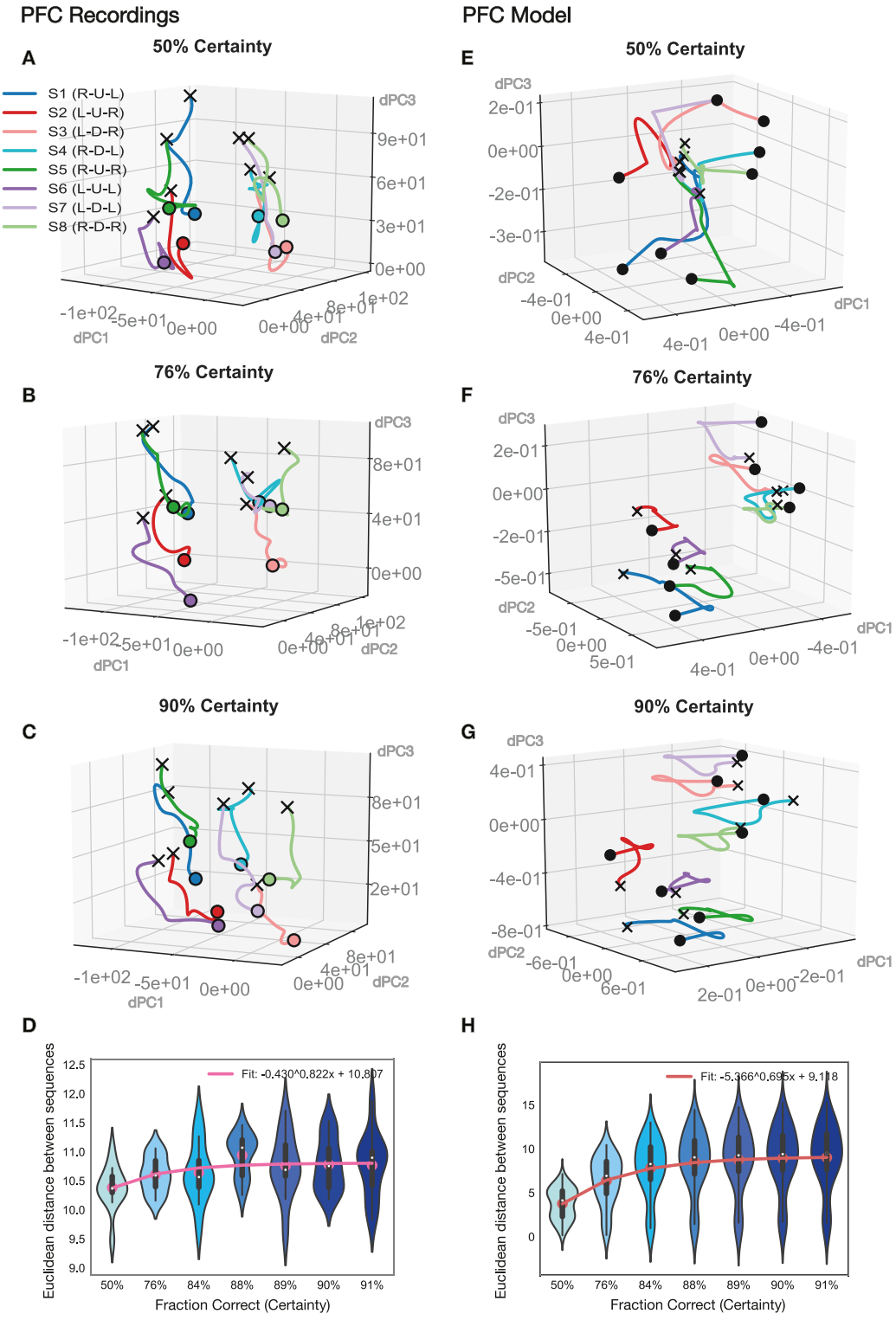

Fig. 6. Evolution of latent task coding with learning for PFC recordings and model network. A–D lPFC neural recordings. Latent task trajectories during the execution of the three-movement sequence, from the start of a trial (black cross) to the end (black dot), depicted in three-dimensional latent space (using the sequence subspace obtained through dPCA; see Methods and Materials – Neural Data – Data Analysis). Latent trajectories for all eight possible movement sequences (S1–S8) are plotted together, for increasing certainty (fraction correct) levels (A–C). **D** Euclidean distance between all sequence trajectories within the two clusters (brightly and lightly colored trajectories) as a function of increasing certainty (see Methods and Materials – Neural Data – Data Analysis). **E–H** PFC model network. Latent task trajectories during the execution of the three-movement sequence depicted in three-dimensional latent space (using the sequence subspace obtained through dPCA, as in A–D). Latent trajectories for all eight possible movement sequences (S1–S8) are plotted together, for increasing certainty levels (E–G). **H** Euclidean distance between all sequence trajectories within the two clusters (brightly and lightly colored trajectories) as a function of increasing certainty level (calculated as in D). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

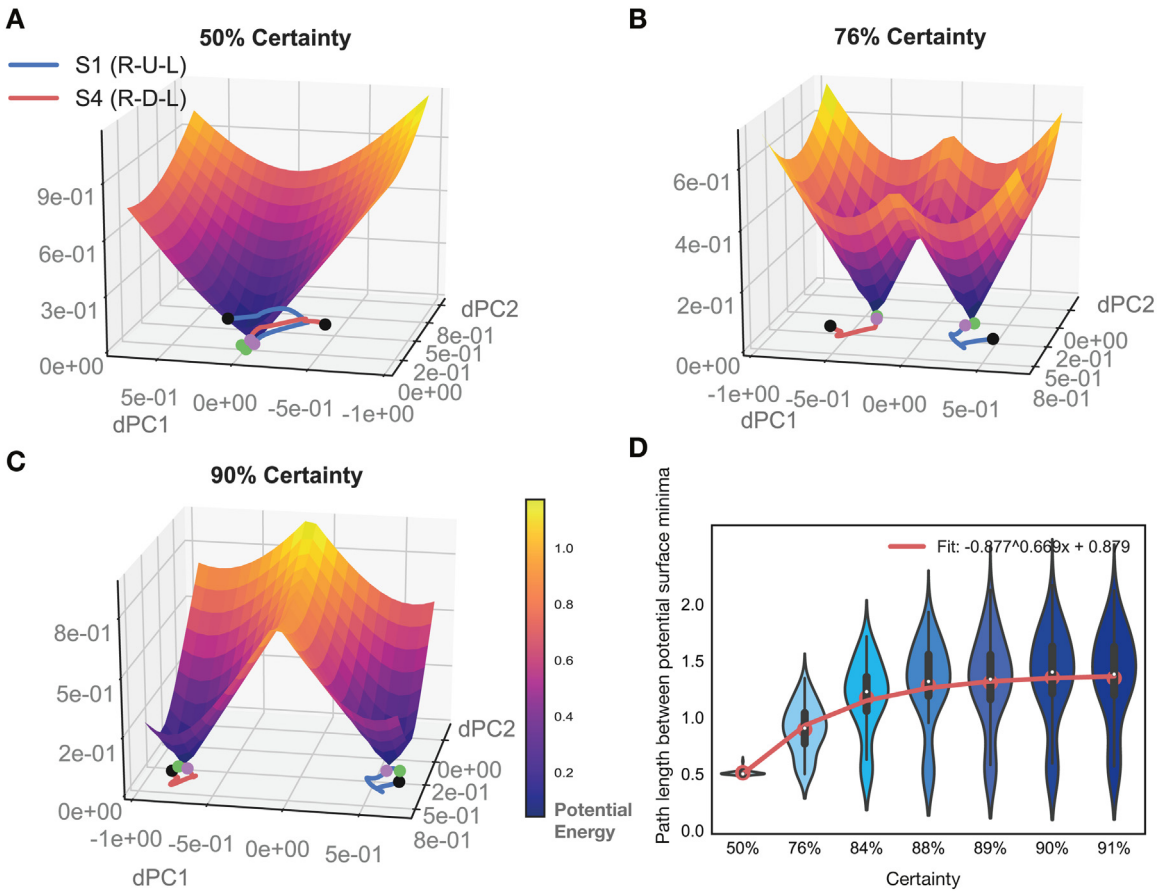

Fig. 7. Evolution of potential surface with learning in PFC model network. A–C Latent trajectories (obtained through dPCA) for two different movement sequences are depicted in two dimensional latent space (S1 in blue and S4 in red, on the bottom). Solid dots depict the beginning (green) and end (black) of the three-movement sequence. The surface depicts the potential energy of the network in the two dimensional space in which the two sequence trajectories sit. To obtain the potential surface, the network was iterated one step forward with inputs held fixed to a particular chosen timepoint (magenta dot; see Methods and Materials – Corticostriatal Model – Model Analysis). Latent movement-sequence trajectories and potential surface are depicted for increasing certainty (fraction correct) levels in A–C. **D** Minimum path length between the gradient minima of all sequence pairs in the test set. Path length was calculated along the joint gradient surface by using Dijkstra's algorithm (see Methods and Materials – Corticostriatal Model – Model Analysis). Results are depicted for increasing certainty levels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

that fixed point regions for different movement-sequences are pushed farther apart from each other, underlying the increase in behavioral accuracy.

We also examined how neural representations evolved with learning in the striatum (Fig. 8). Trajectories from neural recordings in dSTR (Fig. 8A–D) are plotted alongside trajectories from the striatal model network (Fig. 8E–H). Sequence representations in the dSTR (Fig. 8A–D) did not display any particular clustering by visual hemifield, unlike representations in dlPFC (Fig. 6A–D). We confirmed that there was no significant separation by visual hemifield (upper depicted by strong traces, lower by light traces). Sequence representations were instead scattered around in latent space. As learning progressed, sequence representations spread further apart from each other (Fig. 8A–C). We computed the Euclidean distance between all sequences in neural population space and found it to be increasing with learning (Fig. 8D). Trajectories from the striatal model network (Fig. 8E–H) were also scattered around latent space, similar the neural recordings (Fig. 8A–D). Like in the recordings, sequence representations in the model spread further apart from each other as learning progressed. To quantify this effect, we again computed the Euclidean distance between sequences (see Methods and Materials) and found it to be increasing with learning (Fig. 8H), as in the neural data (Fig. 8D). We confirmed that there was a significant increase in distance ($p < 1.0e-3$, paired t-test, Bonferroni corrected) between the sequence trajectories at the first two certainty levels (50%, 76%) and those at a high certainty level towards the end of the block (88%–91%), for both the neural data (Fig. 8D) and the model (Fig. 8H).

To examine this effect in further detail, We computed the potential surface for various points as learning progressed (Fig. 9), as done previously for the PFC model. In the STR model network we also observed energy minima for different sequences becoming increasingly well separated as learning progressed (Fig. 9A–C). Along with this separation, the ridge in the joint potential surface between different sequence minima heightened with increasing certainty during learning. To quantify this effect, we determined the minimal path length between the various sequence's fixed point regions as learning progressed, as before. We observed that minimal path length between fixed point regions increased with certainty during learning in the STR model network (Fig. 9D). We confirmed that there was a significant increase in distance ($p < 1.0e-3$, paired t-test, Bonferroni corrected) between the sequence trajectories at the first two certainty levels (50%, 76%) and those at a high certainty level towards the end of the block (88%–91%). Altogether, we established that the energy landscape

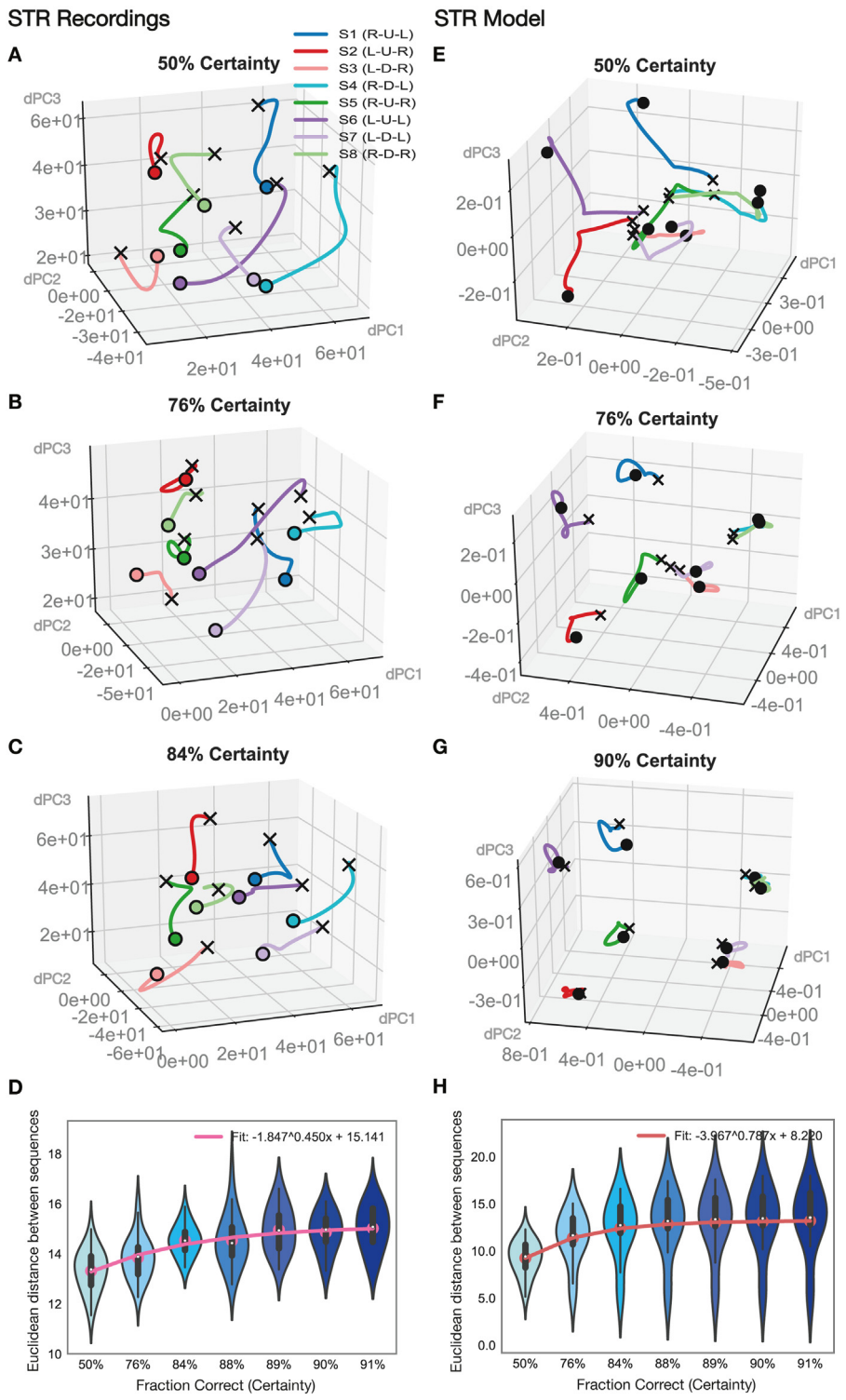

Fig. 8. Evolution of latent task coding with learning for STR recordings and model network. A–D dSTR recordings. Latent task trajectories during the execution of the three-movement sequence, from the start of a trial (black cross) to the end (black dot), depicted in three-dimensional latent space (using the sequence subspace obtained through dPCA; see Methods and Materials – Neural Data – Data Analysis). Latent trajectories for all eight possible movement sequences (S1–S8) are plotted all together for increasing certainty (fraction correct) levels (A–C). **D** Euclidean distance between all sequence trajectories within the two clusters (brightly and lightly colored trajectories) as a function of increasing certainty level (see Methods and Materials – Neural Data – Data Analysis). **E–H** STR model network. Latent trajectories for all eight possible movement sequences (S1–S8) are plotted together in dPCA-derived latent space (using the sequence subspace obtained through dPCA, as in A–D) for increasing certainty (fraction correct) levels (E–G). **H** Euclidean distance between all sequence trajectories within the two clusters (brightly and lightly colored trajectories) as a function of increasing certainty level. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

in the network changes with learning such that fixed point regions for different movement-sequences are pushed farther apart from each other.

We also examined how the shape of a particular movement-sequence representation in latent space changes with learning in

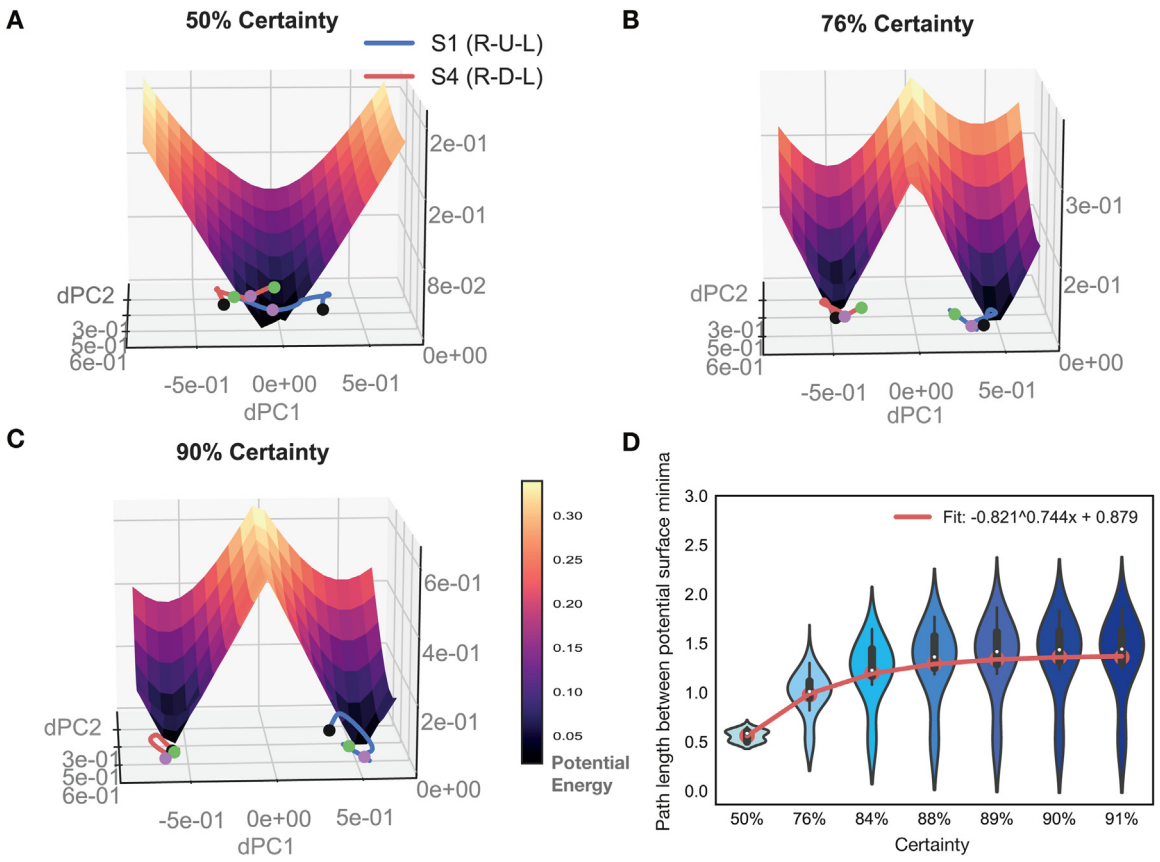

Fig. 9. Evolution of gradient landscape with learning in STR model network. A–C Latent trajectories (obtained through dPCA) for two different movement sequences (S1 in blue, S4 in red) are depicted in two dimensional latent space (bottom). Solid dots depict the beginning (green) and end (black) of the three-movement sequence. The surface depicts the gradient of the network in the two dimensional space in which the two sequence trajectories sit. To obtain the gradient manifold, the network was iterated one step forward with inputs held fixed to a particular chosen timepoint (magenta dot; see Methods and Materials – Corticostriatal Model – Model Analysis). Latent movement-sequence trajectories and gradient surfaces are depicted for increasing certainty (fraction correct) levels (50%, 76% and 90% certainty in A–C). **D** Minimum path length between the gradient minima of all sequence pairs in the test set. Path length was calculated along the joint gradient surface by using Dijkstra's algorithm (see Methods and Materials – Corticostriatal Model – Model Analysis). Results are depicted for increasing certainty levels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

the two regions (Fig. 10). We observed that trajectories became more compact with learning (from lighter to darker blue) as could be seen in neural recordings from dSTR (Fig. 10A) and in the striatal model network (Fig. 10B). To quantify this effect, we computed the Euclidean distance of a particular sequence to its centroid for increasing certainty levels (see Methods and Materials) and found this measure to be decreasing with learning in both striatal recordings (Fig. 10C) and in the striatal model (Fig. 10D). The decreasing trend in prefrontal recordings was not significant (Fig. 10E), while our PFC model network showed a more clear decreasing trend (Fig. 10F).

We further compared latent representations for correct and wrong movements in the prefrontal model network (Fig. S2). We imaged representations of the first movement during the first trial in the block (when error rate is highest) in 3-dimensional dPCA-derived latent space for a few different sample sequences. We found that when the wrong movement was executed (i.e. S1 Error, dotted blue line), the representation moved away from that of the correct movement (right move, R in bold, for sequence S1, solid blue line) and closer to that of sequences which shared the same executed movement (left move, L in bold, for sequence S2, solid red line, and sequence S3, solid rose line). Similarly, the movement representation for the wrong move in sequence 8 (S8, dotted light green line) moved away from the correct move trajectory (solid light green line) and closer to the movement representation of sequence 6 (solid magenta line) which shared the same movement direction.

4. Discussion

The results reveal how learning shapes neurophysiological responses in the corticostriatal system in the brain. Examining data from recordings in the dlPFC-dSTR circuit during an oculomotor sequence learning task, we found that learning increased the distance between action-sequence representations in activity space. To examine this further, we built a model of the corticostriatal system composed of a striatal network encoding action values and a prefrontal network selecting actions. This model was able to autonomously perform the task, matching the animals' behavioral accuracy and closely approximating the representational structure present in neural recordings. Our model revealed that learning shapes the gradient landscape such that fixed point regions corresponding to different action sequences are pushed farther apart from each other. This makes it more likely the network generates the correct action. This offers testable predictions: when task learning is poor, representations should be more clustered together in activity space, making accurate decision making and decoding difficult, and vice versa. Altogether, this work shows how learning is expressed at the network level and suggests network level disruptions may lead to improper task learning.

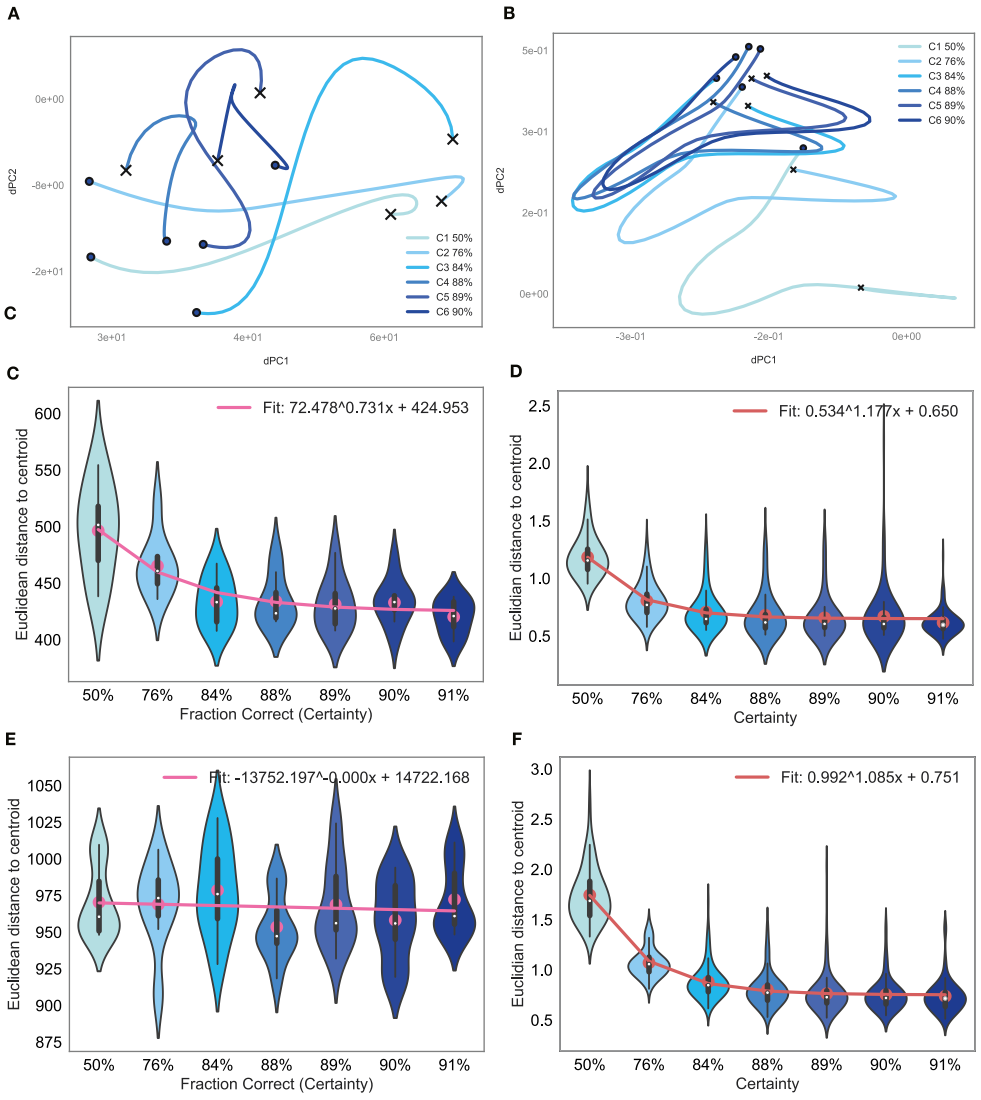

Fig. 10. Shaping of latent movement sequence representations with learning. A–B Latent trajectory for a single movement sequence (obtained through dPCA) depicted in two dimensional latent space for increasing certainty levels (from low, in light blue, to high in dark blue). Neural recordings from dSTR (A) and STR model network (B). **C–F** Euclidean distance between a sequence's centroid (multi-dimensional mean across time) and each point in time for all sequences in the test set (averaged across time points), as a function of increasing certainty (fraction correct). Neural recordings from dSTR (C) and STR model network (D). Neural recordings from lPFC (E) and PFC model networks (F). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Essentially, our network system has learned to express the *process of learning* in network dynamics. This was achieved by guiding the networks through sequential movement-sequence trials arranged in blocks, analogous to the experience of the experimental animals. During this training phase, action and value network weights were continuously updated throughout the block based on a supervisory feedback signal. Following standard network training approaches, this feedback signal was backpropagated in time so that changes in actions and values upon rewards throughout the block were incorporated into each network's respective weight matrix. Being shared across time, the weight matrix thus effectively internalized the *process of learning* over the course of the block. During the test phase, however, weights remained fixed. The network system's ability to respond to changes in external signals (rewards, task cues) appropriately (by increasing or decreasing value signals upon delivery or absence of rewards, respectively) during autonomous testing (completing blocks of movement-sequence trials on their own) attests to the fact that the system has properly internalized (learned) the dynamics that underlie the process of learning over the course of a block.

Our corticostriatal model system respects neuroscientific evidence that implicates the striatum in action value representation (Averbeck & Costa, 2017; Pasupathy & Miller, 2005; Samejima et al., 2005) and the PFC in action selection (Averbeck et al., 2006). We have not explicitly modeled the direct/indirect pathways of the striatum (Gerfen et al., 1990; Vicente et al., 2016), however, as our data originally comes from a behaving monkey experiment, so that we cannot be sure to which pathway the neurons that we recorded from belong. Also, our dataset is likely composed mostly of medium spiny projection neurons, as these make up about 95% of striatal neurons, and we recorded without pre-selection. We did not explicitly model the details of different neuronal types (e.g. projection neurons, interneurons) here, though this is a promising avenue for future research. Our model system's behavioral accuracy and internal task representations are similar to those exhibited by the experimental animals. The methods used to train this system, however, are not

biologically validated, similar to previous approaches (Chaisangmongkon et al., 2017; Mante et al., 2013; Yamins et al., 2014; Yang et al., 2018). We used a supervised training paradigm to teach the movement-sequence task to the system. Alternatively, one may have used a reinforcement learning (RL) paradigm, rather than the gradient of an error signal, to train the network system (Song et al., 2017). This would have impacted the way network weights are updated during training. Here, however, we were interested in the dynamics underlying the learning of sequential actions, which unfold over a longer timescale (the length of the block) than the changes induced by moment-to-moment updates to the network weights. Thus, findings are likely not impacted by different training protocols, and there is no particular benefit to using the RL paradigm for training here. If one wished to study moment-to-moment changes in the network weights due to learning, then using the RL paradigm would be more appropriate. One could also implement this system with spiking networks (Nicola & Clopath, 2017), though there is no reason to believe this would change the dynamics we observed here in any particular way (the energy cost to the system may be lower, however).

Our model and results largely agree with previous findings from the corticostriatal system (Pasupathy & Miller, 2005; Samejima et al., 2005; Seo et al., 2012). Our model system learned to choose actions with the help of an action network (analogous to prefrontal cortex) and RL-derived action values represented in a value network (analogous to the striatum). We found that task representations in neural recordings from dlPFC were organized by visual hemifield, unlike dSTR representations, which showed no particular organizational structure; we also observed this in our model. Both, dlPFC and dSTR latent task trajectories were found to spread apart with learning over the course of the block, in the neural data and the model, analogous to the effect of learning observed in neural recordings from these regions (Seo et al., 2012). Our model system naturally committed occasional errors in its decisions, as the full dataset it was trained on contained occasional error trials. The effect of errors (absence of reward) was incorporated into the value signal through the use of the Q-learning algorithm. After training, the model exhibited the same error distribution over the course of a block as the experimental animals. We observed that when the network committed errors, latent trajectories of wrong movements were represented closer to the opposite movement choice in activity space (where the trajectory would have been had the opposite movement been correct). This effect was previously observed in closely related neural data, in an experiment during which we recorded only from dlPFC (Averbeck & Lee, 2007), and also in related work (Averbeck et al., 2002).

There are previous studies that observed accentuated responses to the first and last movement of an action sequence in lPFC and dSTR neurons of behaving monkeys and rodents (Barnes et al., 2005; Fujii & Graybiel, 2003; Jin & Costa, 2010; Jog, 1999). We have not consistently observed the start–stop activity in our neural datasets (Averbeck & Lee, 2007; Averbeck et al., 2006; Seo et al., 2012), and we believe it has to do with several differences in the underlying task. Specifically, in our task, the movements were not explicitly cued, and had to be executed from memory. We also used a smaller set of sequences than the studies which observed start–stop signals (only 8) and the changed frequently among them over the course of the recording period (approx. every 10 trials). Additionally, the animals had to repeatedly learn which sequence was correct. Therefore, the cognitive demands of the tasks were quite different. One could have taught the networks to produce somewhat modified responses to the first and last movement in a trial sequence (by varying the magnitude of the required output signal during the training phase). However, this would not have affected our main observation which is the change in dynamics during the period between the first and last movement, where the process of learning is expressed. In fact, the outputs of the two networks vary slightly in magnitude due to the inherent noise in the system, and a significant increase in distance between sequence trajectories is observed between the first and last trial of a block despite this variation.

We also found that striatal representations became more compact with learning, both in recordings and in our model. This happened at the same time as sequence-specific representations spread further apart from each other in activity space. This effect could be partially driven by changes in the mean and variance of the neural population firing rate with learning, as well as by changes in higher order statistics. A decrease in responses among top-down signals as rewards become more predictable fits well within a predictive coding framework (Keller & Mrsic-Flogel, 2018; Summerfield & de Lange, 2014). Previously it was found that the Fano factor, a measure of variability (variance of spike count divided by its mean), decreased in prefrontal neurons with learning (Qi & Constantinidis, 2012). Also, changes in firing patterns within the neural population with learning – such as changes in synchronous firing (Baeg et al., 2007) – might be responsible for the changes in latent representation observed here.

Representations in recordings from dlPFC were not found to become significantly more compact with learning, unlike in dSTR. This could be a consequence of less units in dlPFC encoding a learning signal (Seo et al., 2012). In our model we did not see a difference in how compact the representations became with learning in the two networks. This could be amended by additional processing steps which transform the output of the striatal network before it reaches the prefrontal network, which were not included in our model.

In our choice of the noise level, we were guided by previous approaches (Chaisangmongkon et al., 2017; Mante et al., 2013; Yamins et al., 2014; Yang et al., 2018), which suggested an appropriate noise level for network training to perform well. We have experimented with various noise levels (ranging from independent noise with a st. dev. of 0.001 to 0.02) and we found the networks consistently exhibited the same behavior. If the noise level is increased much further, though, it interferes with the training process and eventually impedes learning (by producing large oscillatory dynamics). Overall, the results are robust to small variations in noise, but once the signal gets drowned out by noise the networks become impossible to train. We have not systematically studied the effect of noise on learning here, however, which is an important avenue for future research (e.g. how different types of (correlated) noise might enhance or hinder the learning process).

Our findings are in line with previous work showing that activity in a large population of neurons is confined to a lower dimensional manifold (Chaisangmongkon et al., 2017; Gallego et al., 2018; Mante et al., 2013; Remington et al., 2018; Shenoy et al., 2013; Wang et al., 2018). We found striatal and prefrontal responses encoded the sequence task on a low-dimensional manifold. Previous work was confined to investigating dynamics on this manifold once the task was acquired (and the manifold fixed). Here, we investigated how this manifold was shaped during task acquisition, and found that learning is expressed in dynamics that act upon that manifold. These dynamics re-shaped the manifold in such a way that it became less likely for the network to commit errors.

5. Conclusion

We used a model of the corticostriatal system together with recordings from dlPFC-dSTR circuit in macaques during an oculomotor sequence learning task to investigate how learning shapes neural responses at the population level. The corticostriatal model was able to autonomously perform the task and to approximate neural task representations. Probing the model, we found that learning shapes latent representations such that it becomes less likely to commit errors as learning increases; the potential surface is reshaped with learning such that fixed point regions representing different action choices move farther apart from each other. All in all, this work reveal how neural circuit dynamics in the corticostriatal system drive task learning.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by a Wellcome Trust, United Kingdom NIH-fellowship (C.D.M., S.S. & B.B.A.) and by NIH, United States grant ZIA MH002928-01 (B.B.A.).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.neunet.2020.09.008.

References

Aitchison, L., & Lengyel, M. (2017). With or without you: Predictive coding and Bayesian inference in the brain. *Current Opinion in Neurobiology*, *46*, 219–227.
Alexander, W. H., & Brown, J. W. (2018). Frontal cortex function as derived from hierarchical predictive coding. *Scientific Reports*, *8*(1), 1–11.
Amemori, K.-i., Graybiel, A. M., & Gibb, L. G. (2011). Shifting responsibly: The importance of striatal modularity to reinforcement learning in uncertain environments. *Frontiers in Human Neuroscience*, *5*(47), 1–20.
Averbeck, B. B., Chafee, M. V., Crowe, D. A., & Georgopoulos, A. (2002). Parallel processing of serial movements in prefrontal cortex. *Proceedings of the National Academy of Sciences of the United States of America*, *99*(20), 1–6.
Averbeck, B. B., & Costa, V. D. (2017). Motivational neural circuits underlying reinforcement learning. *Nature Neuroscience*, *20*(4), 505–512.
Averbeck, B. B., & Lee, D. (2007). Prefrontal neural correlates of memory for sequences. *Journal of Neuroscience*, *27*(9), 2204–2211.
Averbeck, B. B., Sohn, J.-W., & Lee, D. (2006). Activity in prefrontal cortex during dynamic selection of action sequences. *Nature Neuroscience*, *9*(2), 276–282.
Baeg, E. H., Kim, Y. B., Kim, J., Ghim, J. W., Kim, J. J., & Jung, M. W. (2007). Learning-induced enduring changes in functional connectivity among prefrontal cortical neurons. *Journal of Neuroscience*, *27*(4), 909–918.
Balaguer, J., Spiers, H., Hassabis, D., & Summerfield, C. (2016). Neural mechanisms of hierarchical planning in a virtual subway network. *Neuron*, *90*(4), 893–903.
Barak, O., Sussillo, D., Romo, R., Tsodyks, M., & Abbott, L. F. (2013). From fixed points to chaos: Three models of delayed discrimination. *Progress in Neurobiology*, *103*, 214–222.
Barnes, T. D., Kubota, Y., Hu, D., Jin, D. Z., & Graybiel, A. M. (2005). Activity of striatal neurons reflects dynamic encoding and recoding of procedural memories. *Nature*, *437*(7062), 1158–1161.
Botvinick, M. M. (2012). Hierarchical reinforcement learning and decision making. *Current Opinion in Neurobiology*, *22*(6), 956–962.
Botvinick, M., Ritter, S., Wang, J. X., Kurth-Nelson, Z., Blundell, C., & Hassabis, D. (2019). Reinforcement learning, fast and slow. *Trends in Cognitive Sciences*, *23*(5), 408–422.
Botvinick, M., & Weinstein, A. (2014). Model-based hierarchical reinforcement learning and human action control. *Philosophical Transactions of the Royal Society, Series B (Biological Sciences)*, *369*(1655), 20130480.
Brendel, W., Romo, R., & Machens, C. K. (2011). Demixed principal component analysis. *Advances in Neural Information Processing Systems*, 2654–2662.
Buonomano, D. V., & Maass, W. (2009). State-dependent computations: Spatiotemporal processing in cortical networks. *Nature Reviews Neuroscience*, *10*(2), 113–125.
Carnevale, F., de Lafuente, V., Romo, R., Barak, O., & Parga, N. (2015). Dynamic control of response criterion in premotor cortex during perceptual detection under temporal uncertainty. *Neuron*, *86*(4), 1067–1077.
Chaisangmongkon, W., Swaminathan, S. K., Freedman, D. J., & Wang, X.-J. (2017). Computing by robust transience: How the fronto- parietal network performs sequential, category- based decisions. *Neuron*, *93*(6), 1504–1517.e4.
Cheong, I., Huang, X., Bettegowda, C., Diaz, L. A., Kinzler, K. W., Zhou, S., & Vogelstein, B. (2006). A bacterial protein enhances the release and efficacy of liposomal cancer drugs. *Science*, *314*(5803), 1308–1311.
Collins, A., & Koechlin, E. (2012). Reasoning, learning, and creativity: Frontal lobe function and human decision-making. *PLoS Biology*, *10*(3), Article e1001293.
Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., & Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, *69*(6), 1204–1215.
Domenech, P., & Koechlin, E. (2015). Executive control and decision-making in the prefrontal cortex. *Current Opinion in Behavioral Sciences*, *1*, 101–106.
Doya, K. (1910). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current Opinion in Neurobiology*, *10*, 732–739.
Frank, M. J. (2005). Dynamic dopamine modulation in the basal ganglia: A neurocomputational account of cognitive deficits in medicated and nonmedicated parkinsonism. *Journal of Cognitive Neuroscience*, *17*(1), 51–72.
Frank, M. J., Seeberger, L. C., & OReilly, R. C. (2004). By carrot or by stick: Cognitive reinforcement learning in parkinsonism. *Science*, *306*(1), 71–100.
Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society, Series B (Biological Sciences)*, *360*(1456), 815–836.
Fujii, N., & Graybiel, A. M. (2003). Representation of action sequence boundaries by macaque prefrontal cortical neurons. *Science*, *301*(5637), 1246–1249.
Gallego, J. A., Perich, M. G., Miller, L. E., & Solla, S. A. (2017). Neural manifolds for the control of movement. *Neuron*, *94*(5), 978–984.
Gallego, J. A., Perich, M. G., Naufel, S. N., Ethier, C., Solla, S. A., & Miller, L. E. (2018). Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature Communications*, *9*(1), 1–13.
Gerfen, C., Engber, T., Mahan, L., Susel, Z., Chase, T., Monsma, F., & Sibley, D. (1990). D1 and D2 dopamine receptor-regulated gene expression of striatonigral and striatopallidal neurons. *Science*, *250*(4986), 1429–1432.
Hennequin, G., Vogels, T. P., & Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, *82*(6), 1394–1406.
Histed, M. H., Pasupathy, A., & Miller, E. K. (2009). Learning substrates in the primate prefrontal cortex and striatum: Sustained activity related to successful actions. *Neuron*, *63*(2), 244–253.
Houk, J. C. (1995). A model of how the basal ganglia generate and use neural signals that predict reinforcement. In *Models of information processing in the basal ganglia* (pp. 249–274). MIT Press.
Jin, X., & Costa, R. M. (2010). Start/stop signals emerge in nigrostriatal circuits during sequence learning. *Nature*, *466*(7305), 457–462.
Jog, M. S. (1999). Building neural representations of habits. *Science*, *286*(5445), 1745–1749.
Johnson, M., Townsend, J., Hawkins, P., & Maclaurin, D. (2018). JAX: Autograd and XLA. Retrieved from https://github.com/google/jax.
Keller, G. B., & Mrsic-Flogel, T. D. (2018). Predictive processing: A canonical cortical computation. *Neuron*, *100*(2), 424–435.
Kobak, D., Brendel, W., Constantinidis, C., Feierstein, C. E., Kepecs, A., Mainen, Z. F., Qi, X.-L., Romo, R., Uchida, N., & Machens, C. K. (2016). Demixed principal component analysis of neural population data. *eLife*, *e10989*, 1–36.
Koechlin, E., & Summerfield, C. (2007). An information theoretical approach to prefrontal executive function. *Trends in Cognitive Sciences*, *11*(6), 229–235.
Langdon, A. J., Sharpe, M. J., Schoenbaum, G., & Niv, Y. (2018). ScienceDirect Model-based predictions for dopamine. *Current Opinion in Neurobiology*, *49*, 1–7.
Lee, D., Seo, H., & Jung, M. W. (2012). Neural basis of reinforcement learning and decision making. *Annual Review of Neuroscience*, *35*(1), 287–308.
Li, J., & Daw, N. D. (2011). Signals in human striatum are appropriate for policy update rather than value prediction. *Journal of Neuroscience*, *31*(14), 5504–5511.
Lim, S., & Goldman, M. S. (2013). Balanced cortical microcircuitry for maintaining information in working memory. *Nature Neuroscience*, *16*(9), 1306–1314.
Maclaurin, D., Duvenaud, D., Johnson, M., & Townsend, J. (2017). Autograd. Retrieved from https://github.com/HIPS/autograd.
Mante, V., Sussillo, D., Shenoy, K. V., & Newsome, W. T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, *503*(7474), 78–84.
Musall, S., Urai, A. E., Sussillo, D., & Churchland, A. K. (2019). ScienceDirect Harnessing behavioral diversity to understand neural computations for cognition. *Current Opinion in Neurobiology*, *58*, 229–238.
Nakahara, H., Doya, K., & Hikosaka, O. (2001). Parallel cortico-basal ganglia mechanisms for acquisition and execution of visuomotor sequences - A computational approach. *Journal of Cognitive Neuroscience*, *13*(5), 626–647.
Nicola, W., & Clopath, C. (2017). Supervised learning in spiking neural networks with FORCE training. *Nature Communications*, *8*(1), 1–15.

Niv, Y. (2009). Reinforcement learning in the brain. *Journal of Mathematical Psychology*, *53*(3), 139–154.
Niv, Y. (2019). Learning task-state representations. *Nature Neuroscience*, *22*(10), 1–10.
O'Doherty, J. P., Dayan, P., Schultz, J., Deichmann, R., Friston, K., & Dolan, R. J. (2004). Dissociable roles of ventral and dorsal striatum in instrumental conditioning. *Science*, *304*, 452–454.
Pasupathy, A., & Miller, E. K. (2005). Different time courses of learning-related activity in the prefrontal cortex and striatum. *Nature*, *433*, 873–876.
Qi, X.-L., & Constantinidis, C. (2012). Variability of prefrontal neuronal discharges before and after training in a working memory task. *PLoS One*, *7*(7), e41053–8.
Rabinovich, M., Huerta, R., & Laurent, G. (2008). Transient dynamics for neural processing. *Science*, *321*(5885), 47–48.
Radulescu, A., Niv, Y., & Ballard, I. (2019). Holistic reinforcement learning: The role of structure and attention. *Trends in Cognitive Sciences*, *23*(4), 1–15.
Rajan, K., Harvey, C. D., & Tank, D. W. (2016). Recurrent network models of sequence generation and memory. *Neuron*, *90*(1), 128–142.
Remington, E. D., Narain, D., Hosseini, E. A., & Jazayeri, M. (2018). Flexible sensorimotor computations through rapid reconfiguration of cortical dynamics. *Neuron*, *98*(5), 1005–1019.e5.
Rich, E. L., & Wallis, J. D. (2016). Decoding subjective decisions from orbitofrontal cortex. *Nature Neuroscience*, *19*(7), 973–980.
Richards, B. A., Lillicrap, T. P., Beaudoin, P., Bengio, Y., Bogacz, R., Christensen, A., Clopath, C., Costa, R. P., Berker, A., Ganguli, S., Gillon, C. J., Hafner, D., Kepecs, A., Kriegeskorte, N., Latham, P., Lindsay, G. W., Miller, K. D., Naud, R., Pack, C. C., ... Kording, K. P. (2019). A deep learning framework for neuroscience. *Nature Neuroscience*, *22*(11), 1–10.
Sakai, K. (2008). Task set and prefrontal cortex. *Annual Review of Neuroscience*, *31*(1), 219–245.
Samejima, K., Ueda, Y., Doya, K., & Kimura, M. (2005). Representation of action-specific reward values in the striatum. *Science*, *310*(5752), 1333–1337.
Sarvestani, I. K., Lindahl, M., Hellgren-Kotaleski, J., & Ekeberg, O. (2011). The arbitration–extension hypothesis: A hierarchical interpretation of the functional organization of the basal ganglia. *Frontiers in Systems Neuroscience*, *5*(13), 1–12.
Seo, M., Lee, E., & Averbeck, B. B. (2012). Action selection and action value in frontal-striatal circuits. *Neuron*, *74*(5), 947–960.
Shenoy, K. V., Sahani, M., & Churchland, M. M. (2013). Cortical control of arm movements: A dynamical systems perspective. *Annual Review of Neuroscience*, *36*(1), 337–359.
Song, H. F., Yang, G. R., & Wang, X.-J. (2017). Reward-based training of recurrent neural networks for cognitive and value-based tasks. *eLife*, *6*(e21492), 1–24.
Stokes, M. G., Kusunoki, M., Sigala, N., Nili, H., Gaffan, D., & Duncan, J. (2013). Dynamic coding for cognitive control in prefrontal cortex. *Neuron*, *78*(2), 364–375.
Summerfield, C., & de Lange, F. P. (2014). Expectation in perceptual decision making: Neural and computational mechanisms. *Nature Publishing Group*, *15*(11), 1–12.
Suri, R. E., & Schultz, W. (1998). Learning of sequential movements by neural network model with dopamine-like reinforcement signal. *Experimental Brain Research*, *121*, 350–354.
Sussillo, D., & Abbott, L. F. (2009). Generating coherent patterns of activity from chaotic neural networks. *Neuron*, *63*(4), 544–557.
Sussillo, D., & Barak, O. (2013). Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Computation*, 1–24.
Sussillo, D., Churchland, M. M., Kaufman, M. T., & Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, *18*(7), 1025–1033.
Sutskever, I. (2013). *Training recurrent neural networks* (Ph.D. thesis), (pp. 1–101).
Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.
Tsujimoto, S., Genovesio, A., & Wise, S. P. (2008). Transient neuronal correlations underlying goal selection and maintenance in prefrontal cortex. *Cerebral Cortex (New York, N.Y.: 1991)*, *18*(12), 2748–2761.
Verschure, P. F. M. J., Pennartz, C. M. A., & Pezzulo, G. (2014). The why, what, where, when and how of goal-directed choice: Neuronal and computational principles. *Philosophical Transactions of the Royal Society, Series B (Biological Sciences)*, *369*(1655), 20130483.
Vicente, A. M., Galvão-Ferreira, P., Tecuapetla, F., & Costa, R. M. (2016). Direct and indirect dorsolateral striatum pathways reinforce different action strategies. *Current Biology*, *26*(7), R267–R269.
Wallis, J. D., Anderson, K. C., & Miller, E. K. (2019). Single neurons in prefrontal cortex encode abstract rules. *Nature*, *411*, 953–956.
Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Hassabis, D., & Botvinick, M. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature Neuroscience*, *21*(6), 1–14.
Wang, J., Narain, D., Hosseini, E. A., & Jazayeri, M. (2017). Flexible timing by temporal scaling of cortical responses. *Nature Neuroscience*, *21*(1), 1–13.
Wood, J. N., & Grafman, J. (2003). Human prefrontal cortex: Processing and representational perspectives. *Nature Reviews Neuroscience*, *4*(2), 139–147.
Yamins, D. L. K., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(23), 8619–8624.
Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T., & Wang, X.-J. (2018). Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*, *22*(2), 1–16.