When and why does motor preparation arise in recurrent neural network models of motor control?

Marine Schimel[@,1], Ta-Chu Kao[2], and Guillaume Hennequin[1]

[1]Computational and Biological Learning Lab, Department of Engineering, University of Cambridge, Cambridge, U.K.
[2]Meta Reality Labs

[@] Corresponding author (mmcs3@cam.ac.uk)

Summary

During delayed ballistic reaches, motor areas consistently display movement-specific activity patterns prior to movement onset. It is unclear why these patterns arise: while they have been proposed to seed an initial neural state from which the movement unfolds, recent experiments have uncovered the presence and necessity of ongoing inputs during movement, which may lessen the need for careful initialization. Here, we modelled the motor cortex as an input-driven dynamical system, and we asked what the optimal way to control this system to perform fast delayed reaches is. We find that delay-period inputs consistently arise in an optimally controlled model of M1. By studying a variety of network architectures, we could dissect and predict the situations in which it is beneficial for a network to prepare. Finally, we show that optimal input-driven control of neural dynamics gives rise to multiple phases of preparation during reach sequences, providing a novel explanation for experimentally observed features of monkey M1 activity in double reaching.

During the production of ballistic movements, the motor cortex is thought to operate as a dynamical system whose state trajectories trace out the appropriate motor commands for downstream effectors (Shenoy et al., 2013; Miri et al., 2017; Russo et al., 2018). The extent to which these cortical dynamics are controlled by exogenous inputs before and/or during movement is the subject of ongoing study.

On the one hand, several experimental and modelling studies point to a potential role for exogenous inputs in motor preparation. First, cortical state trajectories are empirically well described by low-dimensional dynamics that evolve near-autonomously during movement (Churchland et al., 2012; Pandarinath et al., 2018; Schimel et al., 2022), such that there is a priori no reason to suspect that inputs are required for motor production. Rather, inputs would be required during preparation to bring the state of the cortical network into a suitable initial condition. This input-driven seeding process is corroborated by observations of movement-specific primary motor cortex (M1) activity arising well before movement initiation (Lara et al., 2018; Kaufman et al., 2014; Churchland et al., 2012; Meirhaeghe et al., 2023; Figure 1A), and associated models demonstrate the critical role of preparatory inputs therein (Sussillo et al., 2015; Hennequin et al., 2014; Kao et al., 2021).

On the other hand, recent studies have shown that the cortex receives critical input from the thalamus during movement production (Sauerbrei et al., 2020) , and that sensory feedback may also contribute significantly to the observed dynamics of M1 (Kalidindi et al., 2021). Moreover, most published network models of delayed reaches are able to perform the task just as well without preparatory inputs, i.e. with external inputs forcefully confined to the movement epoch – an illustratory example is shown in Figure 1B. Thus, the relative contributions of preparatory vs. movement-epoch inputs to motor cortex dynamics remain unclear.

In addition to the specific form that inputs to cortical dynamics might take, one may ask more broadly about the computational role of motor preparation. Motor preparation is known to benefit behaviour (e.g. by shortening reaction times and enabling more accurate execution Riehle and Requin, 1989; Churchland and Shenoy, 2007; Michaels et al., 2015) and may facilitate motor learning (Sheahan et al., 2016; Sun et al., 2022). However, from the perspective of cortical dynamics, preparation also introduces additional constraints. Specifically, the high density of M1 neurons projecting directly to the spinal cord (Dum and Strick, 1991) suggests that motor cortical outputs control lower-level effectors with little intermediate processing. For preparatory processes to avoid triggering premature movement, any pre-movement activity in the motor and dorsal premotor (PMd) cortices must carefully exclude those pyramidal tract neurons. While this can be achieved by constraining neural activity to evolve in a nullspace of the motor output (Kaufman et al., 2014), the question nevertheless arises: what advantage is there to having neural dynamics begin earlier in a constrained manner, rather than unfold freely just in time for movement production?

Here we sought a normative explanation for motor preparation at the level of motor cortex dynamics: we asked whether preparation arises in recurrent neural networks (RNNs) performing delayed reaching tasks, and what factors lead to more or less preparation. Such an explanation could not be obtained from previous network models of delayed reaches, as they typically assume from the get-go that the cortical network receives preparatory inputs during a fixed time window preceding the go cue (Sussillo et al., 2015; Kao et al., 2021). In this case, pre-movement activity is by design a critical determinant of the subsequent behaviour (Sussillo et al., 2015; Kao et al., 2021; Zimnik and Churchland, 2021). In this work, we removed this modelling assumption and studied models in which the correct behaviour could in principle be obtained without explicit motor preparation.

To study the role of motor preparation, and that of exogenous inputs in this process, we followed an optimal control approach (Harris and Wolpert, 1998; Todorov and Jordan, 2002; Yeo et al., 2016). We considered the dynamics of a recurrent neural network (RNN) model of M1 coupled to a model arm (Todorov and Li, 2003), and used a standard control cost functional to quantify and optimize performance in a delayed-reaching task. We used the iLQR algorithm (Li and Todorov, 2004) to find the spatiotemporal patterns of network inputs that minimize this cost functional, for any given network connectivity. Critically, these inputs could arise both before and during movement; thus, our framework allowed for principled selection amongst a continuum of motor strategies, going from purely autonomous motor generation following preparation, to purely input-driven unprepared dynamics.

We considered an inhibition-stabilized network – which was shown previously to capture prominent aspects of monkey M1 activity (Hennequin et al., 2014; Kao et al., 2021) – and found that optimal control of the model requires preparation, with optimal inputs arising well before movement begins. To understand what features of network connectivity lead to optimal preparatory control strategies, we first turned to low-dimensional models, which could be more easily dissected. We then generalized insights from those systems back to high-dimensional networks using tools from control theory, and found that preparation can be largely explained by two quantities summarizing the dynamical response properties of the network.

Finally, we studied the optimal control of movement *sequences*. Consistent with recent experimental findings (Zimnik and Churchland, 2021), we observed that optimal control of compound reaches leads to input-driven preparatory activity in a dedicated activity subspace prior to each movement chunk.

Overall, our results show that preparatory neural activity patterns arise from optimal control of reaching movements at the level of motor cortical circuits, thus providing a possible explanation for a number of observed experimental findings.

Model

A model of cortical dynamics for reaching movements

We considered a simple reaching task, in which the hand must move from a resting location to one of eight radially located targets in a 2D plane as fast as possible (Figure 1). The target had to be reached within 600 ms of a go cue that followed a delay period of varying (but known) duration. We modelled the trajectory of the hand via a two-jointed model arm (Li and Todorov, 2004; Kao et al., 2021), driven into motion by a pair of torques $\boldsymbol{m}(t)$ (Methods). We further assumed that these torques arose as a linear readout of the momentary firing rates $\boldsymbol{r}(t)$ of a population of M1 neurons,

$$\boldsymbol{m}(t) = \boldsymbol{C}\boldsymbol{r}(t) \tag{1}$$

where $\boldsymbol{C}$ was a randomly generated readout matrix. We modelled the dynamics of $N = 200$ M1 neurons using a standard rate equation,

$$\tau \frac{d\boldsymbol{x}(t)}{dt} = -\boldsymbol{x}(t) + \boldsymbol{W}\boldsymbol{r}(t) + \boldsymbol{h} + \boldsymbol{u}(t) \tag{2}$$

$$\boldsymbol{r}(t) = \phi\left[\boldsymbol{x}(t)\right], \tag{3}$$

where the momentary population firing rate vector $\boldsymbol{r}(t)$ was obtained by passing a vector of internal neuronal activations $\boldsymbol{x}(t)$ through a rectified linear function $\phi[\cdot]$, element-wise. In Equation 2, $\boldsymbol{h}$ is a constant input that establishes a baseline firing rate of 5 Hz on average, with a standard deviation of 5 Hz across neurons, $\boldsymbol{u}(t)$ is a task-dependent control input (see below), and $\boldsymbol{W}$ denotes the matrix of recurrent connection weights. Throughout most of this work, we considered inhibition-stabilized M1 dynamics (Hennequin et al., 2014; Methods), which have previously been shown to produce activity resembling that of M1 during reaching (Kao et al., 2021). Thus, our model can be viewed as a two-level controller, with the arm being controlled by M1, and M1 being controlled by external inputs. Note that each instantiation of our model corresponds to a set of $\boldsymbol{W}$, $\boldsymbol{C}$, and $\boldsymbol{h}$, none of which are specifically optimized for the task.

To prepare or not to prepare?

Previous experimental (Churchland et al., 2012; Shenoy et al., 2013) and modelling (Hennequin et al., 2014; Sussillo et al., 2015; Pandarinath et al., 2018) work suggests that fast ballistic movements rely on strong, near-autonomous internal dynamics in M1. Network-level models of ballistic control thus rely critically on a preparation phase during which the motor cortex is driven

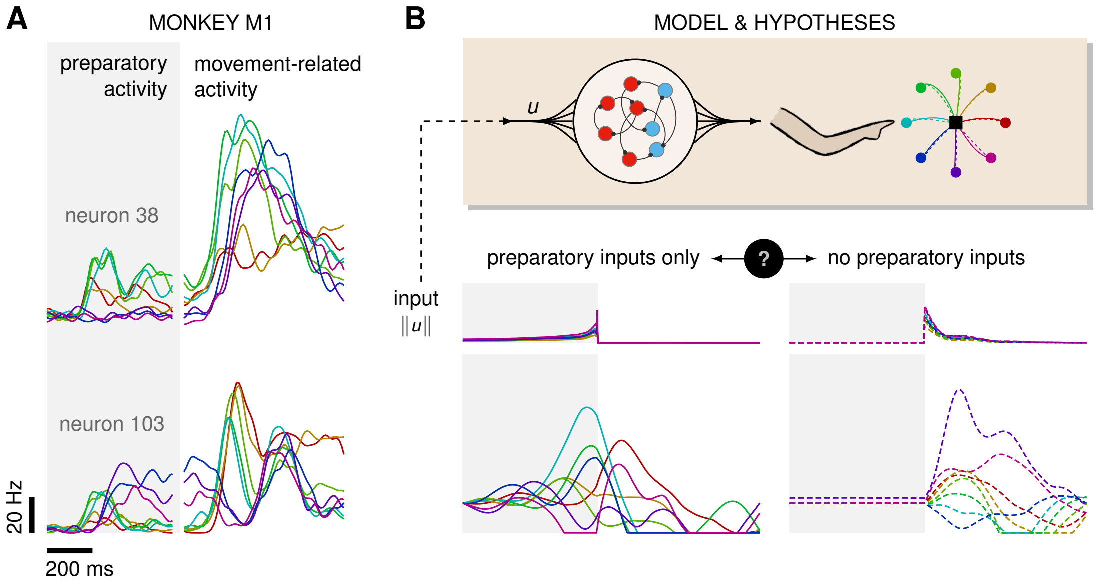

Figure 1: **Control is possible under different strategies.** **(A)** Trial-averaged firing rate of two representative monkey M1 neurons, across 8 different movements, separately aligned to target onset (left) and movement onset (right). Neural activity starts separating across movements well before the animal starts moving. **(B)** Top: a RNN model of M1 dynamics receives external inputs $\boldsymbol{u}(t)$ from a higher-level controller, and outputs control signals for a biophysical two-jointed arm model. Inputs are optimized for the correct production of eight center-out reaches to targets regularly positioned around a circle. Bottom: firing rate of a representative neuron in the RNN model for each reach, under two extreme control strategies. In the first strategy (left, solid lines), the external inputs $\boldsymbol{u}(t)$ are optimized whilst being temporally confined to the preparatory period. In the second strategy (right, dashed lines), they are optimized whilst confined to the movement period. Although slight differences in hand kinematics can be seen (compare corresponding solid and dashed hand trajectories), both control policies lead to successful reaches. These introductory simulations are shown for illustration purposes; the particular choice of network connectivity and the way the control inputs were found are described in the Results section.

into a movement-specific state that seeds its subsequent autonomous dynamics (Kao et al., 2021; Sussillo et al., 2015). However, somewhat paradoxically, the same recurent neural network models can also solve the task in a completely different regime, in which task-related inputs arise during movement only, with no preparatory inputs whatsoever. We illustrate this dichotomy in Figure 1. The same center-out reach can be produced with control inputs to M1 that arise either prior to movement only (full lines), or during movement only (dashed lines). In the latter case, no reach-specific preparatory activity is observed, making the model inconsistent with experimental findings. But what rationale is there in preparing for upcoming movements, then?

To address this question, we formulated delayed reaching as an optimal control problem, and asked what external inputs are required, and at what time, to drive the hand into the desired position with minimum control effort. Specifically, we sought inputs that were as weak as possible yet accurately drove the hand to the target within an alloted time window. We also penalized inputs that caused premature movement before the go cue.

Thus, we solved for spatio-temporal input trajectories that minimized a cost functional capturing the various task requirements. Our cost was composed of three terms: $\mathcal{J}_{\text{target}}$ penalizes deviations away from the target, with an "urgency" weight that increases quadratically with time, thus capturing the implicit incentive for animals to perform fast reaches in such experiments (which are normally conducted in sessions of fixed duration). $\mathcal{J}_{\text{null}}$ penalizes premature movement during preparation, as measured by any deviation in position, speed and acceleration of the hand. Finally, $\mathcal{J}_{\text{effort}}$ penalizes control effort in the form of input magnitude throughout the whole trial, thus promoting energy-efficient control solutions amongst a typically infinite set of possibilities (Kao et al., 2021; Sterling and Laughlin, 2015). Note that $\mathcal{J}_{\text{effort}}$ can be viewed as a standard regularization term, and must be included to ensure the control problem is well defined. The total objective thus had the following form :

$$\mathcal{J}\left[\boldsymbol{u}(t)\right] = \underbrace{\int_0^T \|\boldsymbol{\theta}(t) - \boldsymbol{\theta}^\star\|^2 \frac{t^2}{T^2}\frac{dt}{T}}_{\mathcal{J}_{\text{target}}}$$
$$+ \underbrace{\alpha_{\text{null}} \int_{-\Delta_{\text{prep}}}^{0} \left(\|\boldsymbol{\theta}(t) - \boldsymbol{\theta}_0\|^2 + \|\dot{\boldsymbol{\theta}}(t)\|^2 + \|\boldsymbol{m}(t)\|^2\right)\frac{dt}{T}}_{\mathcal{J}_{\text{null}}}$$
$$+ \underbrace{\alpha_{\text{effort}} \int_{-\Delta_{\text{prep}}}^{T} \|\boldsymbol{u}(t)\|^2 \frac{dt}{NT}}_{\mathcal{J}_{\text{effort}}}, \tag{4}$$

where $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$ denote the position and velocity of the

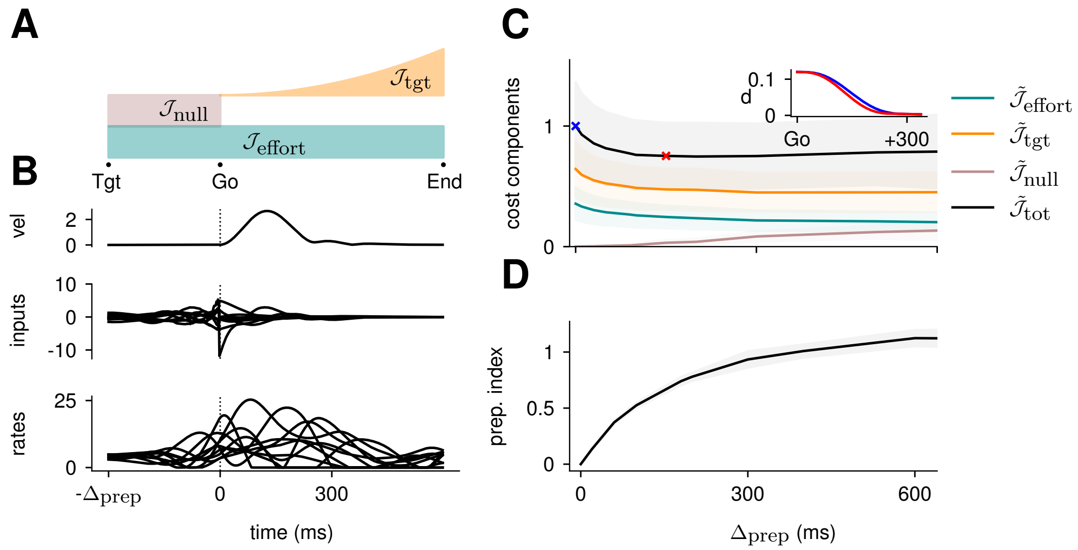

Figure 2: **Optimal control of the ISN network. (A)** Illustration of the different terms in the control cost function, designed to capture the different requirements of the task. "Tgt" marks the time of target onset, "Go" that of the go cue (known in advance) and "End" the end of the trial. **(B)** Time course of the hand velocity (top), optimal control inputs (middle; 10 example neurons), and firing rates (bottom, same neurons) during a delayed reach to one of the 8 targets shown in Figure 1A. Here, the delay period was set to $\Delta_{\text{prep}} = 300$ ms. Note that inputs arise well before the go cue, even though they have no direct effect on behaviour at that stage. **(C)** Dependence of the different terms of the cost function on preparation time. All costs are normalized by the total cost at $\Delta_{\text{prep}} = 0$ ms. The inset shows the time course of the hand's average distance to the relevant target when no preparation is allowed (blue) and when preparation is allowed (red). Although the target is eventually reached for all values of Δ_{prep}, the hand gets there faster with longer preparation times, causing a decrease in $\mathcal{J}_{\text{tgt}}$ – and therefore also in $\mathcal{J}_{\text{tot}}$. Another part of the decrease in $\mathcal{J}_{\text{tot}}$ is due to a progressively lower input energy cost $\mathcal{J}_{\text{effort}}$. On the other hand, the cost of staying still before the Go cue increases slightly with Δ_{prep}. **(D)** We define the preparation index as the ratio of the norms of the external inputs during preparation and during movement (see text). The preparation index measures how much the optimal strategy relies on the preparatory period. As more preparation time is allowed, this is used by the optimal controller and more inputs are given during preparation. For longer preparation times, this ratio increases sub-linearly, and eventually settles.

hand in angular space, Δ_{prep} was the duration of the delay period and T that of the movement period. As $\mathcal{J}_{\text{target}}$ and $\mathcal{J}_{\text{null}}$ depend on $\boldsymbol{u}(t)$ implicitly through Equations 1 and 2, $\mathcal{J}$ is a function of $\boldsymbol{u}$ only. Importantly, we allowed for inputs within a time window beginning Δ_{prep} ms before, and ending T ms after the go cue (set at $t = 0$). Therefore, both preparation-only and movement-only input strategies (cf. Figure 1) could potentially arise, as well as anything in between.

Here, we solved for the optimal control inputs using the iterative linear quadratic regulator algorithm (iLQR; Li and Todorov, 2004), an efficient trajectory optimization algorithm that is well-suited for handling the nonlinear nature of both the arm's and the network's dynamics. As our primary goal was to assess the role of preparation in a normative way, we did not study the putative circuit dynamics upstream of M1 that might lead to the computation of these optimal inputs.

We balanced the various components of our cost functional by choosing α_{null} and α_{effort} to qualitatively match the behavioural requirements of a typical reach-and-hold task. Specifically, we tuned them jointly so as to ensure (i) stillness during preparation, and (ii) reach duration of approximately ~ 400 ms, with the hand staying within 0.5cm of the target for ~ 200 ms after the end of the reach. We ensured that the main qualitative features of the solution, i.e. the results presented below, were robust to the choice of hyperparameter values within the fairly large range in which the above soft-constraints are satisfied (Supplementary Material S1).

Results

Preparation arises as an optimal control strategy

Using the above control framework, we assessed whether the optimal way of performing a delayed reach involves preparation. More concretely, does the optimal control strategy of the model described in Equation 2 involve any preparatory inputs during the delay period? For any single optimally performed reach, we found that network activity began changing well before the go cue (Figure 2B, bottom), and that this was driven by inputs that arose early (Figure 2B, middle). Thus, although preparatory network activity cancels in the readout (such that the hand remains still; Figure 2B, top) and therefore does not contribute directly to movement, it still forms an integral part of the optimal reach strategy. To quantify how much the optimal control strategy relied on inputs prior to movement, we defined the *preparation index* as the ratio of input magnitude during the delay period to that during the remainder of the trial:

$$\text{prep. index} = \sqrt{\frac{\int_{-\Delta_{\text{prep}}}^{0} \|\boldsymbol{u}(t)\|^2 dt}{\int_0^T \|\boldsymbol{u}(t)\|^2 dt}}. \tag{5}$$

We found that the preparation index rose sharply as we increased the delay period, and eventually plateaued at ~ 1.3 for delay periods longer than 300 ms (Figure 2C). Similarly, the total cost of the task was highest in the absence of preparation, and decreased until it also reached a plateau at $\Delta_{\text{prep}} \sim 300$ ms (Figure 2C, black). This appears somewhat counterintuitive, as having a larger Δ_{prep} means that both $\mathcal{J}_{\text{effort}}$ and $\mathcal{J}_{\text{null}}$ are accumulated over a longer period. To resolve this paradox, we examined each component of the cost function. We found that the overall decrease in cost with increasing preparation time was driven by a concurrent decrease in both $\mathcal{J}_{\text{tgt}}$ and $\mathcal{J}_{\text{effort}}$. The former effect was due to the model producing faster reaches (Figure 2C inset; hand position for a reach with (red) and without (blue) preparation) while the latter arose from the use of smaller control inputs when preparation was allowed. Together, these results suggest that the presence of a delay period changes the optimal control strategy for reaching, and increases performance in the task.

The results above show that delaying the reach beyond ~ 300 ms brings little benefit; in particular, all components of the cost stabilize past that point (Figure 2C). We thus wondered what features the optimally controlled dynamics would display as Δ_{prep} increased beyond 300 ms. Would the network defer preparation to a last minute surge, or prepare more gently over the entire preparatory window? Would the network produce the same neural activity patterns? We found that the optimal controller made very little use of any preparation time available up to 300 ms before the go cue: with longer preparation times, external input continued to arise just a couple of hundred milliseconds before movement initiation, and single neuron firing rates remained remarkably similar (Figure 3A). This was also seen in PCA projections of the firing rates, which traced out similar trajectories irrespective of the delay period (Figure 3B). We hypothesized that this behaviour is due to the network dynamics having a certain maximum characteristic timescale, such that inputs that arrive too early end up being "forgotten" – they increase $\mathcal{J}_{\text{effort}}$ and possibly $\mathcal{J}_{\text{null}}$ without having a chance to influence $\mathcal{J}_{\text{tgt}}$. We confirmed this by varying the characteristic time constant (τ in Equation 2). For a fixed Δ_{prep}, we found that for larger (resp. lower) values of τ, the optimal control inputs started rising earlier (resp. later) and thus occupied more (resp. less) of the alloted preparatory period (Supplementary Material S3).

Understanding optimal control in simplified models

Having established that the ISN model of M1 relies on preparatory inputs to solve the delayed reaching task, we next tried to understand *why* it does so. To further unravel the interplay between the structure of the network and the optimal control strategy, i.e. what aspects of the dynamics of the network warrant preparation, we turned to simpler, two-dimensional models of cortical dynamics. These 2D models are small enough to enable detailed analysis (Supplementary Material S3), yet rich enough to capture the two dominant dynamical phenomena that arise in ISN dynamics: nonnormal amplification (Murphy and Miller, 2009; Goldman, 2009; Hennequin et al., 2012) and oscillations (Brunel, 2000; Dayan and Abbott, 2001). Specifically, networks of E and I neurons have been shown to embed two main motifs of effective connectivity which are revealed by appropriate orthogonal changes of basis: (i) feedforward ("nonnormal") connectivity whereby a "source mode" of E-I imbalance feeds into a "sink mode" in which balance is restored, and (ii) anti-symmetric connectivity that causes the two populations to oscillate.

To study the impact of each of these prototypical connectivity motifs on movement preparation, we implemented them separately, i.e. as two small networks of two units each, with an overall connectivity scale parameter w which we varied (Figure 4A and D; Methods). As both nonnormal and oscillatory dynamics arise from linear algebraic properties of the connectivity matrix, we considered linear network dynamics for this analysis ($\phi(x) = x$ in Equation 3). Moreover, to preserve the existence of an output nullspace in which preparation could in principle occur without causing premature movement, we reduced the dimensionality of the motor readout from 2D (where there would be no room left for a nullspace) to 1D (leaving a 1D nullspace), and adapted the motor task so that the network now had to move the hand position along a single dimension (Figure 4B and E, top). Analogous to the previous arm model, we assumed that the hand's acceleration along this axis was directly given by the 1D network readout.

We found that optimal control of both dynamical motifs generally led to preparatory dynamics, with inputs arising before the go cue (Figure 4B and E, bottom). In the feedforward motif, the amount of preparatory inputs appeared to depend critically on the orientation of the readout. When the readout was aligned with the sink (brown) mode (Figure 4B, left), the controller prepared the network by moving its activity along the source (orange) mode (Figure 4C, left). This placed the network in a position from which it had a natural propensity to generate large activity transients along the readout dimension (c.f. flow field in Figure 4A); here, these transients were exploited to drive the fast upstroke in hand acceleration and throw the hand towards the target location. Note that this strategy reduces the amount of

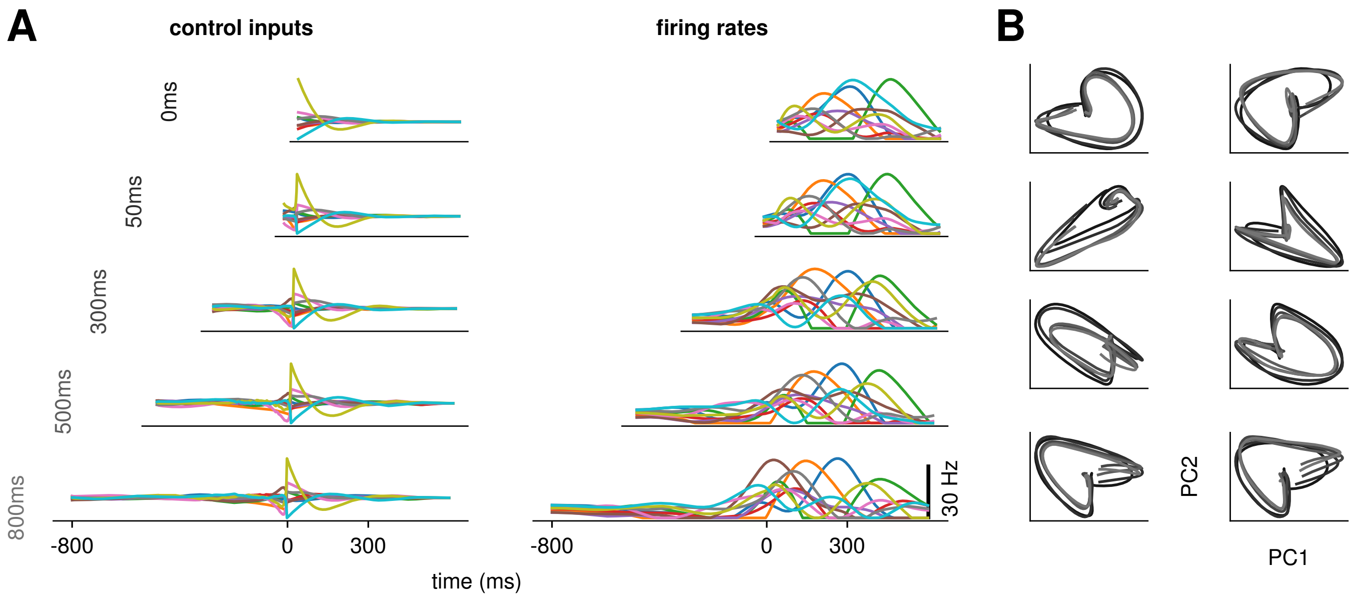

Figure 3: **Conservation of the optimal control strategy across delays** **(A)** Optimal control inputs to ten randomly chosen neurons in the model RNN (left) and their corresponding firing rates (right) for different preparation times Δ_{prep} (ranging from 0 to 800 ms; c.f. labels). **(B)** Projection of the movement-epoch population activity for each of the 8 reaches (panels) and each value of Δ_{prep} shown in A (darker to lighter colors). These population trajectories are broadly conserved across delay times, and become more similar for larger delays.

input the controller needs to deliver during the movement, because the network itself does most of the work. Nevertheless, in this case the network's own impulse response was not rich enough to accommodate the phase reversal required to subsequently slow the hand down and terminate the movement. Optimal control therefore also involved inputs during the movement epoch, leading to a preparatory index of ~ 0.54 (Figure 4G, dark blue).

When it was instead the source mode that was read out (Figure 4B, right), the only dimension along which the system could prepare without moving was the sink mode. Preparing this way is of no benefit, because the flow field along the sink mode has no component along the source (readout) mode. Thus, here the optimal strategy was to defer control to the movement epoch, during which the transient growth of network activity along the readout rested entirely on adequate control inputs. This led to a preparation index of ~ 0 (Figure 4G, pale green). Although the network did react with large activity excursions along the sink mode (Figure 4C, right), these were inconsequential for the movement. Importantly, of the two extreme readout configurations discussed above, the first one yielded a smaller overall optimal control cost (by a factor of ~ 1.5). Thus, at a meta-control level, ideal downstream effectors would read out the sink mode, not the source mode. Note that while increasing the connectivity strength initially led to more preparation (Figure 4H), a plateau was eventually reached for $w \geq 4$. Indeed, while stronger dynamics initially make preparation more beneficial, they also make it more difficult for preparatory activity to remain in the readout nullspace.

We obtained similar insights for oscillatory network dynamics (Figure 4D-F). A key difference however was that the flow field was rotationally symmetric such that no distinction could be made between "source" and "sink" units – indeed the optimal control strategy yielded the same results (up to a rotation of the state space) irrespective of which of the two units was driving the hand's acceleration (compare left and right panels in Figure 4D-F). Nevertheless, the optimal controller consistently moved the network's activity along the output-null axis during preparation, in such a way as to engage the network's own rotational flow immediately after the go cue (Figure 4F). This rotational flow drove a fast rise and decay of activity in the readout unit, thus providing the initial segment of the required hand acceleration. The hand was subsequently slowed down by modest movement-epoch control inputs which eventually receded, leading to a preparation index of ~ 0.58. Interestingly, the preparation index showed a decrease for very large w (Figure 4G), which did not reflect smaller preparatory inputs (Figure 4H) but rather reflected the larger inputs that were required during movement to cancel the fast oscillations naturally generated by the network.

The above results highlight how the optimal control strategy is shaped by the dynamical motifs present in the network. Crucially, we found that the optimal way to control the movement depends not only on the strength and flow of the internal network dynamics, but also on their interactions with the readout.

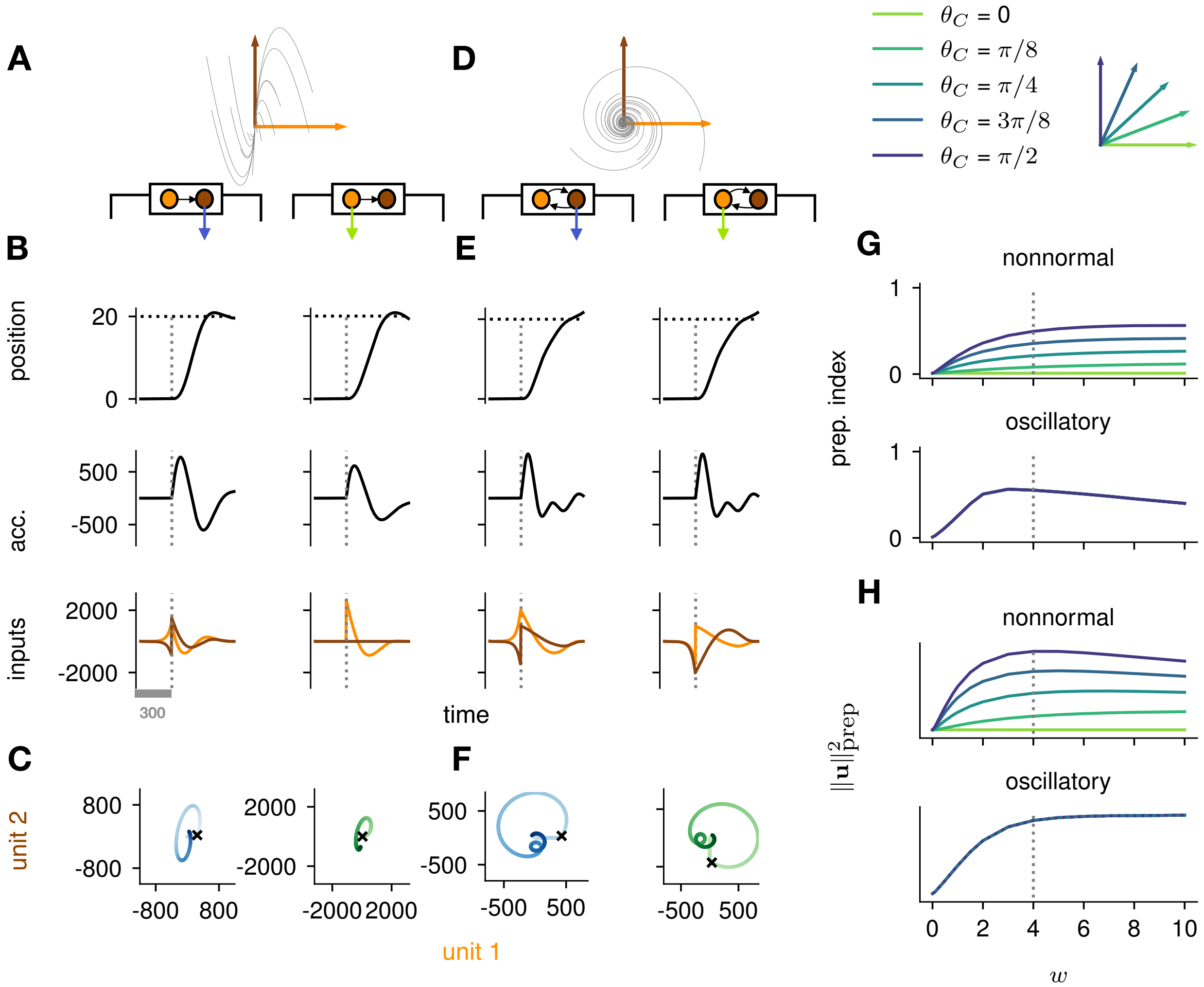

Figure 4: **Analysis of the interplay between the optimal control strategy and two canonical motifs of E-I network dynamics:** nonnormal transients driven by feedforward connectivity (A–C), and oscillations driven by anti-symmetric connectivity (D–F). **(A)** Activity flow field (10 example trajectories) of the nonnormal network, in which a "source" unit (orange) drives a "sink" unit (brown). We consider two opposite readout configurations, where it is either the sink (left) or the source (right) that drives the acceleration of the hand. **(B)** Temporal evolution of the hand position (top; the dashed horizontal line indicates the reach target), hand acceleration (middle) and optimal control inputs to the two units (bottom; colours matching panel A), under optimal control given each of the two readout configurations shown in A (left vs. right). The dashed vertical line marks the go cue, which follows a 300 ms delay period. While the task can be solved successfully in both cases, preparatory inputs are only useful when the sink is read out. **(C)** Network activity trajectories under optimal control. Each trajectory begins at the origin, and the end of the delay period is shown with a black cross. **(D-F)** Same as (A-C), for the oscillatory network. **(G-H)** Preparation index (top) and total amount of preparatory inputs (bottom) as a function of the scale parameter w of the network connectivity, for various readout configurations (colour-coded as shown in the top inset). The nonnormal network (top) prepares more when the readout is aligned to the most controllable mode, while the amount of preparation in the oscillatory network (bottom) is independent of the readout direction. The optimal strategy must balance the benefits from preparatory inputs which allow to exploit the intrinsic network dynamics, with the constraint to remain still. This is more difficult when the network dynamics are strong and pushing activity out of the readout-null subspace, explaining the decrease in preparation index for large values of w in the oscillatory network.

Control-theoretic properties predict the amount of preparation

Our investigation of preparation in a low-dimensional system allowed us to isolate the impact of core dynamical motifs, and highlighted how preparation depends on the geometry of the flow field, and its alignment to the readout. However, these intuitions remain somewhat qualitative, making them difficult to generalize to our high-dimensional ISN model.

To quantify the key criteria that appear important for preparation, we turned to tools from control theory. We reasoned that, for a network to be able to benefit from preparation and thus exhibit a large preparation index, there must be some advantage to using early inputs that do not immediately cause movement, relative

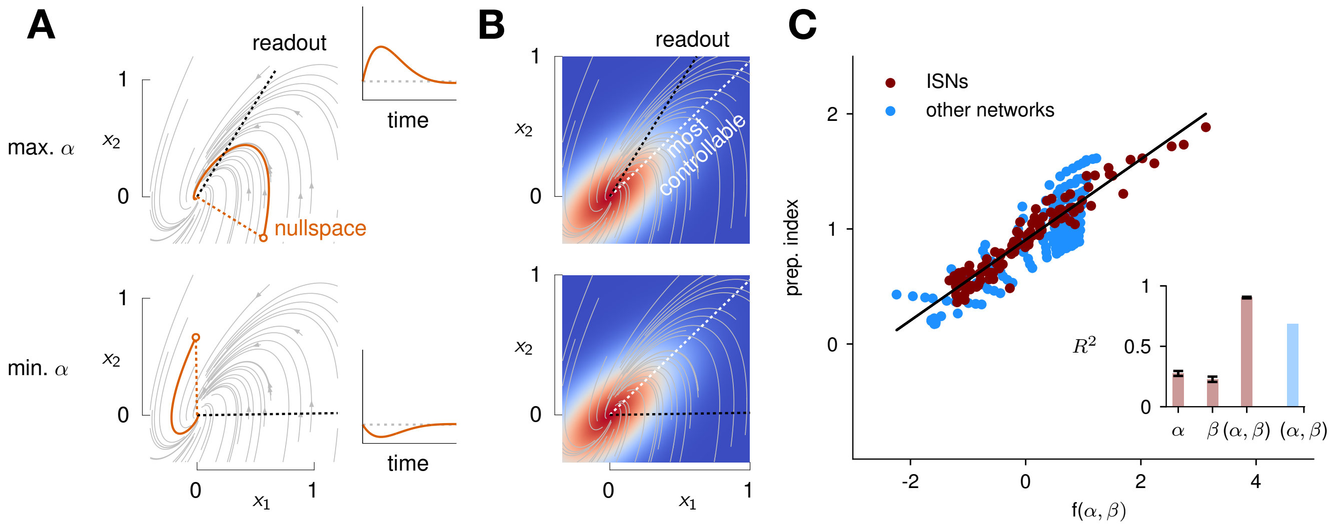

Figure 5: **Predicting the preparation index from the observability of the output nullspace (α) and the controllability of the readout (β, see details in text). (A)** Illustration of the observability of the output nullspace in a synthetic 2-dimensional system. The observability of a direction is characterized by how much activity (integrated squared norm) is generated along the readout by a unit-norm initial condition aligned with that direction. The top and bottom panels show the choices of readout directions (dotted black) for which the corresponding nullspace (dotted orange) is most (maximum α) and least (minimum α) observable, respectively. Trajectories initialized along the null direction are shown in solid orange, and their projections onto the readout are shown in the inset. **(B)** Illustration of the controllability of the readout in the same 2D system as in (A). To compute controllability, the distribution of activity patterns collected along randomly initialized trajectories is estimated (heatmap); the controllability of a given direction then corresponds to how much variance it captures in this distribution. Here, the network has a natural propensity to generate activity patterns aligned with the dashed white line ('most controllable' direction). The readout directions are repeated from panel A (dotted black). The largest (resp. smallest) value of β would by definition be obtained when the readout is most (resp. least) controllable. Note the tradeoff in this example: the choice of readout that maximizes α (top) does not lead to the smallest β. **(C)** The values of α and β accurately predict the preparation index ($R^2 = 0.93$) for a range of high-dimensional ISNs (maroon dots) with different connectivity strengths and characteristic timescales (Methods). The best fit (after z-scoring) is given by $f(\alpha, \beta) = (16.94 \pm 0.02)\alpha - (15.97 \pm 0.02)\beta$ (mean $\pm$ sem were evaluated by boostrapping). This confirms our hypothesis that optimal control relies more on preparation when α is large and β is small. Note that α and β alone only account for 34.8% and 30.4% of the variance in the preparation index, respectively (inset). Thus, α and β provide largely complementary information about the networks' ability to use inputs, and can be combined into a very good predictor of the preparation index. Importantly, even though this fit was obtained *using ISNs only*, it still captures 69% of preparation index variance across networks from other families (blue dots; Methods).

to using later inputs that do. We hypothesized that this advantage could be broken down into two criteria. First, there must exist activity patterns that are momentarily output-null (i.e. do not immediately cause movement) yet seed output-potent dynamics that subsequently move the arm. The necessity of this criterion was obvious in the 2D nonnormal network, which did not display any preparation when its nullspace was aligned with its "sink" mode. In the language of control theory, this criterion implies that the nullspace of the readout must be sufficiently "observable" – we captured this in a scalar quantity α (Methods; Kao and Hennequin, 2019; Skogestad and Postlethwaite, 2007). Second, there must be a sizeable cost to performing the movement in an entirely input-driven manner without relying on preparation. In other words, the network should be hard to steer along the readout direction, i.e. the readout must be of limited "controllability" – we captured this in another scalar quantity β (Methods).

We illustrate the meaning of these two metrics in Figure 5A and B for a 2-dimensional example network that combines nonnormality and oscillations. We show two extreme choices of readout direction (Figure 5A, dashed black): the one that maximizes α (top) and the one that minimizes it (bottom). In the first case, the readout nullspace (dashed orange) is very observable, i.e. trajectories that begin in the nullspace evolve to produce large transients along the readout (solid orange & inset). In the second case, the opposite is true. For each case, we also assessed the controllability of the readout (β). The controllability of a direction corresponds to how much variance activity trajectories exhibit along that direction, when they are randomly and isotropically initialized (Figure 5B). In other words, a very controllable direction is one along which network trajectories have a natural tendency to evolve.

We then assessed how well α and β could predict the preparation index of individual networks. In 2D networks, we found that a simple function that grows with α and decreases with β could accurately predict preparation across thousands of networks (Supplementary Material S3.2). To assess whether these insights carried over to high-dimensional networks, we generated a range of large ISNs with parametrically varied connectivity strengths and decay timescales (Methods). We then regressed the preparation index against the values of α and β computed for each of these networks (as controllability and observability are only defined for linear networks, we set $\phi(x) = x$ for this investigation). We found that a simple linear mapping, prep. index $= k_0 + k_\alpha \alpha + k_\beta \beta$, with parameters fitted to one half of the ISN networks, accurately predicted the preparation indices of the other half (Figure 5C; $R^2 = 0.93$, 5-fold cross-validated). Interestingly, we observed that although α and β (which are both functions of the network connectivity) were highly correlated across different networks, discarding either variable in our linear regression led to a significant drop in R^2 (Figure 5C, inset). Importantly, it was their difference that best predicted the preparation index ($k_\alpha > 0$ and $k_\beta < 0$), consistent with our hypothesis that the preparation index is a relative quantity which increases as the nullspace becomes more observable, but decreases as readout dimensions become more controllable.

We were able to confirm the generality of this predictive model by generating networks with other types of connectivity (oscillatory networks, and networks with unstructured random weights), which displayed dynamics very different from the ISNs (see Supplementary Material S4). Interestingly, despite the different distribution of α and β parameters in those networks, we could still capture a large fraction of the variance in their preparation index ($R^2 = 0.69$) using the linear fit obtained from the ISNs alone. This confirms that α and β can capture information about the networks' dynamics in a universal manner.

Note that we do not make any claims about the specific functional form of the relationship between α, β and the preparation index. Rather, we claim that there should be a broad trend for the preparation index to increase with α and decrease with β, and we acknowledge that this relationship could in general be nonlinear. Indeed, in 2D networks, we found that the preparation index was in fact better predicted by the ratio of α over β than by their difference (Supplementary Material S3.2).

Modelling movement sequences

Having gained a better understanding of what features lead a network to prepare, we next set out to assess whether optimal control could also explain the neural preparatory processes underlying the generation of movement *sequences*. We revisited the experimental studies of Zimnik and Churchland (2021), where monkeys were trained to perform two consecutive reaches. Each trial started with the display of both targets, followed by an explicitly enforced delay period before the onset of the first reach. A distinction was made between "double" reaches in which a pause was enforced between reaches, and "compound" reaches in which no pause was required. This study concluded that, rather than the whole movement sequence unrolling from a single preparatory period, each reach was instead successively seeded by its own preparatory activity.

Here, we asked whether such an independent, successive preparation strategy would arise as an optimal control solution, in the same way that single-reach preparation did. Importantly, we could not answer this question by directly examining network inputs as we did for single reaches. Indeed, any network input observed before the second reach could be contributing either to the end of the first movement, or to the preparation of the next. In fact, the issue of teasing apart preparatory vs. movement-related activity patterns also arose in the analysis of the monkey data. To address this, Zimnik and Churchland (2021) exploited the fact that monkey M1 activity just before and during single reaches is segregated into two distinct subspaces. Thus, momentary activity patterns (during either single or double reaches) can be unambiguously labelled as preparatory or movement-related depending on which of the two subspaces they occupied. We performed a similar analysis (Methods) and verified that preparatory and movement activity patterns in the model were also well segregated in their respective subspaces in the single-reach task (Figure 6A-B). We then assessed the occupancy of the preparatory subspace during double reaching in the model, and took this measure as a signature of preparation.

To model optimal control of a double reach, we modified our cost functional to account for the presence of two consecutive targets (see Methods). We considered the same set of eight targets as in our single-reach task, and modelled all possible combinations of two targets (one example shown in Figure 6). We set the hyperparameters of the cost function such that both targets could be reached by the resulting optimal controller, in a way that matched important qualitative aspects of the monkeys' behaviour (in particular, such that both reaches were performed at similar velocities, with the second reach lasting slightly longer on average; Figure 6B-C; top).

We projected the network activity onto preparatory and movement subspaces identified using single and double reaches activity (Methods). For double reaches with a long (600ms) pause, the preparatory subspace was transiently occupied twice, with the two peaks occurring just before the onset of each movement in the sequence (Figure 6B; bottom).

Notably, the occupancy during the "compound" reach (without pause; Figure 6C) also started rising prior to the first movement before decaying very slightly and peaking again before the second reach, indicating two independent preparatory events. This is somewhat surprising, given that a movement sequence can also be

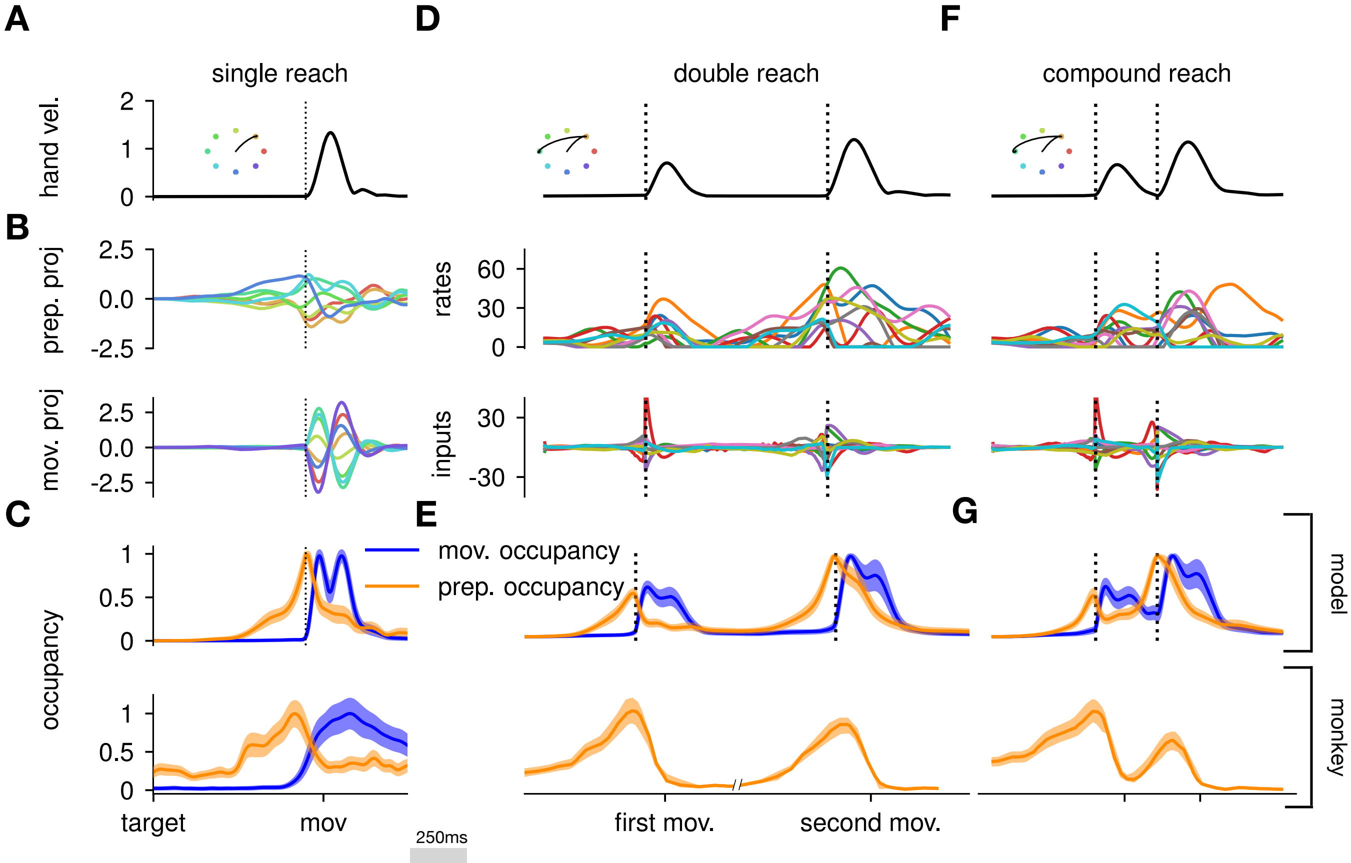

Figure 6: **The model executes a sequence of two reaches using an independent preparation strategy. (A)** Hand velocity during one of the reaches, with the corresponding hand trajectory shown in the inset. **(B-C)** We identified two 6-dimensional orthogonal subspaces, capturing 79% and 85% of total activity variance during single-reach preparation and movement respectively. **(B)** First principal component of the model activity for the 8 different reaches projected into the subspaces identified using preparatory (top) and movement-epoch (bottom) activity. **(C)** Occupancy (total variance captured across movements) of the orthogonalized preparatory and movement subspaces, in the model (top) and in monkey motor cortical activity (bottom; reproduced from Lara et al., 2018 for monkey Ax). We report mean ± s.e.m., where the error is computed by bootstrapping from the neural population as in Lara et al. (2018). We normalize each curve separately to have a maximum mean value of 1. To align the model and monkey temporally, we re-defined the model's 'movement onset' time to be 120 ms after the model's hand velocity crossed a threshold – this accounts for cortico-spinal delays and muscle inertia in the monkey. Consistent with Lara et al. (2018)'s monkey M1 recordings, preparatory subspace occupancy in the model peaks shortly before movement onset, rapidly dropping thereafter to give way to pronounced occupancy of the movement subspace. Conversely, there is little movement subspace occupancy during preparation. **(D)** Behavioural (top) and neural (middle) correlates of the delayed reach for one example of a double reach with an enforced pause of 0.6 s. The optimal strategy relies on preparatory inputs preceding each movement. **(E)** Same as (C), for double reaches. The onsets of the monkey's two reaches are separately aligned to the model's using the same convention as in (C). The preparatory subspace displays two clear peaks of occupancy. This double occupancy peak is also observed in monkey neural activity (bottom; reproduced from Zimnik and Churchland, 2021, with the first occupancy peak aligned to that of the model). **(F)** Same as (D), for compound reaches with no enforced pause in between. Even though the sequence could be viewed as a single long movement, the control strategy relies on two periods of preparation. Here, inputs before the second reach are used to reinject energy into the system after slowing down at the end of the first reach. **(G)** Even though no explicit delay period is enforced in-between reaches during the compound movement, the preparatory occupancy rises twice, before the first reach and once again before the second reach. This is similar to observations in neural data (bottom; reproduced from Zimnik and Churchland, 2021).

viewed as a single "compound" movement, for which we have shown previously a unique preparatory phase is sufficient (Figure 2). In our model, this behaviour can be understood to arise from the requirement that the hand stop briefly at the first target. To produce the second reach, the hand needs to accelerate again, which requires transient re-growth of activity in the network. Given that the network's dynamical repertoire exhibits limited timescales, this is most easily achieved by reinjecting inputs into the system.

In summary, our results suggest that the "independent" preparation strategy observed in monkeys is consistent with the optimal control of a two-reach sequence. While Zimnik and Churchland (2021) showed that RNNs trained on this task used this "independent" strategy, this was by design as the network was only cued for the second reach after the first one had started. In addition to replicating this proof of concept that it is possible to prepare whilst moving, our model also shows how and why independent preparation might arise as an optimal control solution.

Discussion

In this work, we proposed a model for the dynamics of motor cortex during a delayed reaching task in non-human primates. Unlike previous work, we treated M1 as an input-driven nonlinear dynamical system, with generic connectivity not specifically optimized for the task, but with external inputs assumed to be optimal for each reach. Motivated by a large body of evidence suggesting that preparation is useful before delayed reaches (Churchland et al., 2010; Lara et al., 2018; Afshar et al., 2011; Shenoy et al., 2013), but also evidence for thalamic inputs being necessary for accurate movement execution (Sauerbrei et al., 2020), we used this model to investigate whether and why neural circuits might rely on motor preparation during delayed reaching tasks. Interestingly, preparation arose as an optimal control strategy in our model, with the optimal solution to the task relying strongly on inputs prior to movement onset. Moreover, the benefits of preparation were dependent on the network connectivity, with preparation being more prevalent in networks whose rich internal dynamics can be advantageously seeded by early external inputs. We were able to quantify this intuition with a predictive model relating the dynamical response properties of a network to the amount of preparation it exhibits when controlled optimally. Finally, we found that prominent features of the monkeys' neural activity during sequential reaches arose naturally from optimal control assumptions. Specifically, optimally controlled networks relied on two phases of preparation when executing sequences of two reaches, corroborating recent experimental observations in monkey M1 (Zimnik and Churchland, 2021). Together, our results provide a normative explanation for the emergence of preparatory activity in both single and sequential reaching movements.

In recent years, task-optimized recurrent neural networks have become a very popular tool to model neural circuit dynamics. Classically, those models incorporate only those inputs that directly reflect task-related stimuli (e.g. motor target, go cue, etc). This requires assumptions about the form of the inputs, such as modelling them as simple step functions active during specific task epochs. However, as local neural circuits are part of a wider network of brain areas, a large portion of their inputs come from other brain areas at intermediate stages of the computation and may therefore not be directly tied to task stimuli. Thus, it is not always obvious what assumptions can reasonably be made about the inputs that drive the circuit's dynamics. Our optimization framework, which does not require us to make any specific assumptions about when and how inputs enter the network (although it does allow to incorporate prior information in the form of constraints), allows to bypass this problem and to implicitly model unobserved inputs from other areas. Importantly, our framework allows to ask questions – such as "why prepare" – that are difficult to phrase in standard input-driven RNN models. We note, however, that in the investigation we have presented here, the lack of imposed structure for the inputs also implied that the model could not make use of mechanisms known to contribute certain aspects of preparatory neural activity. For example, our model did not exhibit the usual visually-driven response to the target input, nor did it have to use the delay epoch to keep such a transient sensory input in memory (Guo et al., 2014; Li et al., 2015). Moreover, while we did not introduce any strong assumptions about the temporal structure of inputs, we did assume for simplicity that the optimal controller was aware of the duration of the delay period. While this made solving for the optimal control inputs easier, it made our task more akin to a self-initiated reach (Lara et al., 2018) than to a typical delayed reach with unpredictable, stochastic delay durations. We note that our current framework could be generalized to the stochastic setting by using model predictive control to optimize the control inputs (Rawlings et al., 2017). Although this would be considerably more expensive computationally, we hypothesize that explicitly modelling such uncertainty over the delay period would not only yield a better model of the typical delayed reaching task, but may also lead to preparatory activity patterns arising immediately after target onset and persisting until the go cue.

Dynamical systems have a longstanding history as models of neural populations (Dayan and Abbott, 2001). However, understanding how neural circuits can perform various computations remains a challenging question. Recently, there has been increased interest in trying to understand the role of inputs in shaping cortical dynamics. This question has been approached both from a data-driven perspective (Malonis et al., 2021; Soldado-Magraner et al., 2023), and in modelling work with e.g Driscoll et al. (2022) showing how a single network can perform different tasks by reorganizing its dynamics under the effect of external inputs and Dubreuil et al. (2021) relating network structure to the ability to process contextual inputs. To better understand how our motor system can generate flexible behaviours (Logiaco et al., 2021; Stroud et al., 2018), and to characterize learning on short timescales (Sohn et al., 2020; Heald et al., 2023), it is important to study how network dynamics can be modulated by external signals that allow rapid adaptation to new contexts without requiring extensive modifications of the network's connectivity. The optimal control approach we proposed here offers a way to systematically perform such evaluations, in a variety of tasks and under different assumptions regarding how inputs are allowed to impact the dynamics of the local circuit of interest. While our model's predictions will depend on e.g. the choice of connectivity or the design of the cost function, an exciting direction for future work will be to obtain those parameters in a data-driven manner, for instance using recently developed methods to infer dynamics from data (Pandarinath et al., 2018; Schimel et al., 2022), and advances in inverse reinforcement learning and differentiable control (Amos et al., 2018) to infer the cost function that behaviour optimizes.

Methods

Experimental model and subject details

In Figure 1, we showed data from two primate datasets that were made available to us by Mark Churchland, Matthew Kaufman and Krishna Shenoy. Details of animal care, surgery, electrophysiological recordings, and behavioral task have been reported previously in Churchland et al. (2012); Kaufman et al. (2014) (see in particular the details associated with the J and N "array" datasets). The subjects of this study, J and N, were two adult male macaque monkeys (Macaca mulatta). The animal protocols were approved by the Stanford University Institutional Animal Care and Use Committee. Both monkeys were trained to perform a delayed reaching task on a fronto-parallel screen. At the beginning of each trial, they fixated on the center of the screen for some time, after which a target appeared on the screen. After a variable delay period (0–1000 ms), a go cue appeared instructing the monkeys to reach toward the target. Recordings were made in the dorsal premotor cortex and in the primary motor cortex using a pair of implanted 96-electrode arrays. In Figure 6, we also reproduced data from Lara et al. (2018) and Zimnik and Churchland (2021). Details of animal care, surgery, electrophysiological recordings, and behavioral task for those data can be found in the Methods section of the respective papers.

Arm model

To simulate reaching movements, we used the planar two-link arm model described in Li and Todorov (2004). The two links have lengths L_1 and L_2, masses M_1 and M_2, and moments of inertia I_1 and I_2 respectively. The lower arm's center of mass is located a distance D_2 from the elbow. By considering the geometry of the upper and lower limb, the position of the hand and elbow can be written as vectors $\boldsymbol{y}_\mathrm{h}(t)$ and $\boldsymbol{y}_\mathrm{e}$ given by

$$\begin{aligned} \mathbf{y}_\mathrm{h} &= \begin{pmatrix} L_1\cos\theta_1 + L_2\cos(\theta_1+\theta_2) \\ L_1\sin\theta_1 + L_2\sin(\theta_1+\theta_2)\end{pmatrix} \text{ and} \\ \boldsymbol{y}_\mathrm{e} &= \begin{pmatrix} L_1\cos\theta_1 \\ L_1\sin\theta_1 \end{pmatrix}. \end{aligned} \tag{6}$$

The joint angles $\boldsymbol{\theta} = (\theta_1;\theta_2)^T$ evolve dynamically according to the differential equation

$$\boldsymbol{m}(t) = \mathcal{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathcal{X}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) + \mathcal{B}\dot{\boldsymbol{\theta}}, \tag{7}$$

where $\boldsymbol{m}(t)$ is the momentary torque vector, $\mathcal{M}$ is the matrix of inertia, $\mathcal{X}$ accounts for the centripetal and Coriolis forces, and $\mathcal{B}$ is a damping matrix representing joint friction. These parameters are given by

$$\mathcal{M}(\boldsymbol{\theta}) = \begin{pmatrix} a_1 + 2a_2\cos\theta_2 & a_3 + a_2\cos\theta_2 \\ a_3 + a_2\cos\theta_2 & a_3 \end{pmatrix} \tag{8}$$

$$\mathcal{X}(\boldsymbol{\theta},\dot{\boldsymbol{\theta}}) = a_2\sin\theta_2 \begin{pmatrix} -\dot{\theta}_2(2\dot{\theta}_1 + \dot{\theta}_2) \\ \dot{\theta}_1^{\,2} \end{pmatrix} \tag{9}$$

$$\mathcal{B} = \begin{pmatrix} 0.05 & 0.025 \\ 0.025 & 0.05 \end{pmatrix} \tag{10}$$

with $a_1 = I_1 + I_2 + M_2L_1^2$, $a_2 = M_2L_1D_2$, and $a_3 = I_2$.

iLQR algorithm

Throughout this work, we used the iLQR algorithm (Li and Todorov, 2004) to find the locally optimal inputs that minimize our cost function. iLQR is a trajectory optimization algorithm that can handle nonlinear dynamics and non-quadratic costs. iLQR works in an iterative manner, by linearizing the dynamics and performing a quadratic approximation of the cost at each iteration, thus turning the control problem into a local linear-quadratic problem whose unique solution is found using LQR (Kalman et al., 1960). The LQR solver uses a highly efficient dynamic programming approach that exploits the sequential structure of the problem. Our implementation of iLQR followed from Li and Todorov (2004), with the difference that we performed regularization of the local curvature matrix as recommended by Tassa (2011).

Generation of the high-dimensional readouts and networks

Generation of inhibitory-stabilized networks Simulations in Figures 1, 3, 5 and 6 were conducted using inhibition-stabilized networks (ISN). Those were generated according to the procedure described in Hennequin et al. (2014) with minor adjustments. In brief, we initialized strongly connected chaotic networks with sparse and log-normally distributed excitatory weights, and stabilized them through progressive $\mathcal{H}_2$-optimal adjustments of the inhibitory weights until the spectral abscissa of the connectivity matrix fell below 0.8. This yielded strongly connected but stable networks with a strong degree of non-normality. When considering a larger range of ISNs (Figure 5), we independently varied both the variance of the distribution of initial excitatory weights and the spectral abscissa below which we stopped optimizing the inhibitory weights.

Generation of additional networks in Figure 5 To assess the generality of our findings in Figure 5, we additionally generated randomly connected networks by sampling each weight from a Gaussian distribution with $\sigma = R/\sqrt{N}$, where the spectral radius R was varied between 0 and 0.99. We also sampled skew-symmetric networks by drawing a random network $\boldsymbol{S}$ as above, and setting $\boldsymbol{W} = (\boldsymbol{S} - \boldsymbol{S}^T)/2$. We varied the radius R of the $\boldsymbol{S}$ matrices between 0 and 5. Moreover, we considered diagonally shifted skew-symmetric networks $\boldsymbol{W} = (\boldsymbol{S} - \boldsymbol{S}^T)/2 + \lambda \boldsymbol{I}$ where λ denotes the real part of all the eigenvalues and was varied between 0 and 0.8.

The elements of the readout matrix $\boldsymbol{C}$ mapping neural activity onto torques were drawn from a normal distribution with zero mean and standard deviation $\sigma_C = 0.05/\sqrt{N}$. This was chosen to ensure that firing rates of standard deviation on the order of 30Hz would be decoded into torques of standard deviation ~ 2 N/m, which is the natural variation required for the production of the reaches we considered.

Details of Figure 4

To more easily dissect the phenomena leading to the presence or absence of preparation, we turned to 2D linear networks in Figure 4. We modelled nonnormal networks with a connectivity $\boldsymbol{W} = \begin{bmatrix} 0 & 0 \\ w & 0 \end{bmatrix}$ and oscillatory networks with connectivity $\boldsymbol{W} = \begin{bmatrix} 0 & -w \\ w & 0 \end{bmatrix}$. The activity of the two units evolved as

$$\tau \dot{\boldsymbol{x}}(t) = -\boldsymbol{x}(t) + \boldsymbol{W}\boldsymbol{x}(t) + \boldsymbol{u}(t) \tag{11}$$

and directly influenced the acceleration of a one-dimensional output $y(t)$ according to

$$\ddot{y}(t) = \boldsymbol{C}_i \boldsymbol{x}(t) \tag{12}$$

where $\boldsymbol{C}_i = \begin{bmatrix} \cos\theta_C & \sin\theta_C \end{bmatrix}$ was a row matrix reading the activity of the network along an angle θ_C from the horizontal (first unit). Our setup aimed to mirror the reaching task studied in this work. We thus optimized inputs to minimize the following cost function :

$$\mathcal{J}[\boldsymbol{u}] =: \underbrace{\int_0^T \|y(t) - y^\star\|^2 \frac{t^2}{T^2} \frac{dt}{T}}_{\mathcal{J}_{\text{target}}} + \underbrace{\alpha_{\text{null}} \int_{-\Delta_{\text{prep}}}^0 \left(\|y(t)\|^2 + \|\dot{y}(t)\|^2 + \|\ddot{y}\|^2\right) \frac{dt}{T}}_{\mathcal{J}_{\text{null}}} + \underbrace{\alpha_{\text{effort}} \int_{-\Delta_{\text{prep}}}^T \|\boldsymbol{u}(t)\|^2 \frac{dt}{2T}}_{\mathcal{J}_{\text{effort}}}. \tag{13}$$

where $y^\star = 20$ was the target position.

Computing networks' controllability and observability to predict preparation in Figure 5

As part of our attempt to predict how much a network will prepare given its intrinsic properties only, we computed the prospective potency of the nullspace α, and the controllability of the readout β. For a stable linear dynamical system described by

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{14}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t), \tag{15}$$

the system's observability Gramian $\boldsymbol{Q}$ can be computed as the unique positive-definite solution of the Lyapunov equation

$$\boldsymbol{A}^T\boldsymbol{Q} + \boldsymbol{Q}\boldsymbol{A} + \boldsymbol{C}^T\boldsymbol{C} = 0. \tag{16}$$

The prospective potency of the nullspace $\boldsymbol{C}^\perp$ is then defined as

$$\alpha \triangleq \frac{\mathrm{Tr}(\boldsymbol{C}^\perp \boldsymbol{Q} \boldsymbol{C}^{\perp T})}{N-2} \tag{17}$$

Note that this measure α is invariant to the specific choice of basis for the nullspace $\boldsymbol{C}^\perp$. Similarly, to assess the controllability of the readout, we first computed the controllability Gramian of the system $\boldsymbol{P}$, which is the solution of

$$\boldsymbol{A}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{B}\boldsymbol{B}^T = 0, \tag{18}$$

with $\boldsymbol{B} = \boldsymbol{I}$ in our system. We then defined the controllability of the readout as

$$\beta \triangleq \frac{\mathrm{Tr}(\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^T)}{2}. \tag{19}$$

Details of Figure 6

Cost function We modelled sequences of reaches by modifying our cost functional to account for the presence of two targets, as

$$\begin{aligned}\mathcal{J}[\boldsymbol{u}] = & \underbrace{\int_0^{\Delta_{\text{move}}^{(1)}+\tau} \|\boldsymbol{\theta}(t) - \boldsymbol{\theta}_1^\star\|^2 \frac{t^2}{T^2} dt}_{\mathcal{J}_{\text{target}}^{(1)}} \\ & + \alpha_{\text{pause}} \underbrace{\int_{\Delta_{\text{move}}^{(1)}}^{\Delta_{\text{move}}^{(1)}+\tau} \|\dot{\boldsymbol{\theta}}(t)\|^2 dt}_{\mathcal{J}_{\text{pause}}} \qquad (20) \\ & + \underbrace{\int_{\Delta_{\text{move}}^{(1)}+\tau}^{T} \|\boldsymbol{\theta}(t) - \boldsymbol{\theta}_2^\star\|^2 \frac{(t-\Delta_{\text{move}}^{(1)}-\tau)^2}{T^2} dt}_{\mathcal{J}_{\text{target}}^{(2)}} \\ & + \underbrace{\alpha_{\text{null}} \int_{-\Delta_{\text{prep}}}^{0} \|\boldsymbol{\theta}(t) - \boldsymbol{\theta}_0\|^2 + \|\dot{\boldsymbol{\theta}}(t)\|^2 + \|\boldsymbol{m}(t)\|^2 dt}_{\mathcal{J}_{\text{null}}} \\ & + \underbrace{\alpha_{\text{effort}} \int_{-\Delta_{\text{prep}}}^{T} \|\boldsymbol{u}(t)\|^2 dt}_{\mathcal{J}_{\text{effort}}} \qquad (21)\end{aligned}$$

where τ describes how long the monkey's hands had to stay on the intermediate target before performing its second reach. We used $\tau = 600\,\text{ms}$ and $\alpha_{\text{pause}} = 100$ for the double reaches in which a pause was explicitely enforced during the experiment. For compound reaches, the experiment did not require monkeys to stop for any specific duration. However, to ensure that the hand stopped on the target in the model (as it does in experiments when monkeys touch the screen) rather than fly through it, we set $\tau = 6\,\text{ms}$ and $\alpha_{\text{pause}} = 100$ when modelling compound reaches.

Preparatory subspace analysis Lara et al. (2018) proposed an analysis to identify preparatory and movement-related subspaces. This analysis allows to assess when the neural activity enters those subspaces, independently of whether it is delay-period or post-go-cue activity.

The method identifies a set of preparatory dimensions and a set of movement dimensions, constrained to be orthogonal to one another, as in Elsayed et al. (2016). These are found in the following manner: the trial-averaged neural activity is split between preparatory and movement-related epochs, yielding two matrices of size $N \times MT$ where N is the number of neurons, T is the number of time bins and M is the number of reaches. One then optimizes the $W_{\text{prep}} \in \mathbb{R}^{N\times d_{\text{prep}}}$ and $W_{\text{mov}} \in \mathbb{R}^{N\times d_{\text{mov}}}$ (where d_{prep} and d_{mov} are the predefined dimensions of the two subspaces) such that the subspaces respectively capture most variance in the preparatory and movement activities, while being orthogonal to one another. This is achieved by maximizing the following objective :

$$\mathcal{C}(W_{\text{prep}}, W_{\text{mov}}) = \frac{1}{2}\left(\frac{\text{Tr}(W_{\text{prep}}^T C_{\text{prep}} W_{\text{prep}})}{Z_{\text{prep}}(d_{\text{prep}})} + \frac{\text{Tr}(W_{\text{mov}}^T C_{\text{mov}} W_{\text{mov}})}{Z_{\text{mov}}(d_{\text{mov}})}\right) \qquad (22)$$

where $C_{\text{prep/mov}}$ are the covariance matrices of the neural activity during the preparatory and movement epochs, respectively. The normalizing constant $Z_{\text{prep}}(d_{\text{prep}})$ denotes the maximum amount of variance in preparatory activity that can be captured by any subspace of dimension d_{prep} (this is found via SVD), and similarly for $Z_{\text{mov}}(d_{\text{mov}})$. The objective is maximized under the constraints $W_{\text{prep}}^T W_{\text{mov}} = 0$, $W_{\text{prep}}^T W_{\text{prep}} = I$ and $W_{\text{mov}}^T W_{\text{mov}} = I$. We set subspace dimensions $d_{\text{prep}} = d_{\text{mov}} = 6$, although our results were robust to this choice.

The occupancy of the preparatory subspace was defined as

$$\text{occupancy}^{\text{prep}}(t) = \sum_{k=1}^{d_{\text{prep}}} \text{var}_\theta(x_k^{\text{prep}}(t,\theta))$$

and that of the movement subspace was defined as

$$\text{occupancy}^{\text{mov}}(t) = \sum_{k=1}^{d_{\text{mov}}} \text{var}_\theta(x_k^{\text{mov}}(t,\theta)).$$

For single reaches, we defined preparatory epoch responses as the activity in a 300ms window before the end of the delay period, and movement epoch responses as the activity in a 300ms window starting 50ms after the go cue. We normalized all neural activity traces using the same procedure as Churchland et al. (2012); Elsayed et al. (2016). For double reaches, we followed Zimnik and Churchland (2021) and used neural activity traces from both single reaches and the first reach of double-reach sequences. Note that we did not include any activity from the second reaches in the sequence, or from compound reaches, when defining the subspaces.

Parameter table

<table>
<tr><th>symbol</th><th>fig. 1</th><th>fig. 2</th><th>fig. 3</th><th>fig. 5</th><th>fig. 4</th><th>fig. 6</th><th>unit</th><th>description</th></tr>
<tr><td>L_1</td><td colspan="4">30</td><td>-</td><td>30</td><td>cm</td><td>length of the upper arm in model</td></tr>
<tr><td>L_2</td><td colspan="4">30</td><td>-</td><td>30</td><td>cm</td><td>length of the forearm in model</td></tr>
<tr><td>I_1</td><td colspan="4">0.025</td><td>-</td><td>0.025</td><td>$kg.m^{-2}$</td><td>inertia of upper arm</td></tr>
<tr><td>I_2</td><td colspan="4">0.045</td><td>-</td><td>0.045</td><td>$kg.m^{-2}$</td><td>inertia of foream</td></tr>
<tr><td>M_1</td><td colspan="4">1.4</td><td>-</td><td>1.4</td><td>kg</td><td>mass of upper arm</td></tr>
<tr><td>M_2</td><td colspan="4">1.0</td><td>-</td><td>1.0</td><td>kg</td><td>mass of forearm</td></tr>
<tr><td>D_2</td><td colspan="4">16</td><td>-</td><td>16</td><td>cm</td><td>elbow to lower arm center of mass distance</td></tr>
<tr><td>r</td><td colspan="4">12</td><td>20</td><td>12</td><td>cm</td><td>radius of the target reach</td></tr>
<tr><td>$\mu_{\mathbf{h}}$</td><td colspan="4">20</td><td>0</td><td>-</td><td>mV</td><td>mean baseline firing rate</td></tr>
<tr><td>$\sigma_{\mathbf{h}}$</td><td colspan="4">5</td><td>0</td><td>-</td><td>mV</td><td>s.t.d of the baseline firing rate</td></tr>
<tr><td>α_{effort}</td><td colspan="4">5E-7</td><td>1E-5</td><td>5E-7</td><td>-</td><td>coef. of input cost</td></tr>
<tr><td>α_{null}</td><td colspan="4">1</td><td>1</td><td>10</td><td>-</td><td>coef. of cost of moving during the delay</td></tr>
<tr><td>α_{pause}</td><td colspan="5">-</td><td>100</td><td>-</td><td>coef. of cost of moving between reaches</td></tr>
<tr><td>τ</td><td colspan="6">150</td><td>ms</td><td>single-neuron time constant</td></tr>
<tr><td>$\Delta^{(1)}_{\text{move}}$</td><td colspan="5">-</td><td>300</td><td>ms</td><td>duration of the first reach</td></tr>
<tr><td>Δ_{prep}</td><td>500</td><td>300</td><td>-</td><td colspan="2">300</td><td>500</td><td>ms</td><td>delay period time</td></tr>
<tr><td>T</td><td>1100</td><td>900</td><td>-</td><td colspan="2">900</td><td>2000 — 1406</td><td>ms</td><td>total movement duration</td></tr>
<tr><td>N</td><td colspan="4">200</td><td>-</td><td>200</td><td>-</td><td>number of neurons</td></tr>
<tr><td>p_{con}</td><td colspan="4">0.2</td><td>-</td><td>0.2</td><td>-</td><td>connection probability (E neurons)</td></tr>
<tr><td>p_{E}</td><td colspan="4">80</td><td>-</td><td>80</td><td>-</td><td>percentage of E neurons</td></tr>
<tr><td>p_{I}</td><td colspan="4">20</td><td>-</td><td>20</td><td>-</td><td>percentage of I neurons</td></tr>
</table>

Table 1: Parameters used for the various simulations.

Acknowledgments

We are grateful to Matthew T. Kaufman and Mark M. Churchland for sharing data for the monkey experiments. We thank Kristopher Jensen, David Liu, Javier Antorán, and Rory Byrne for helpful comments on the manuscript. M.S. was funded by an EPSRC DTP studentship, and part of this work was performed using resources provided by the Cambridge Tier-2 system operated by the University of Cambridge Research Computing Service (http://www.hpc.cam.ac.uk) funded by EPSRC Tier-2 capital grant EP/P020259/1. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising from this submission.

Competing interests

T-C.K. is currently a research scientist at Meta Reality Labs, but only contributed to this work while studying at the University of Cambridge.

References

Afshar, A., Santhanam, G., Byron, M. Y., Ryu, S. I., Sahani, M., and Shenoy, K. V. (2011). Single-trial neural correlates of arm movement preparation. *Neuron*, 71(3):555–564.

Amos, B., Jimenez, I., Sacks, J., Boots, B., and Kolter, J. Z. (2018). Differentiable mpc for end-to-end planning and control. In *Advances in Neural Information Processing Systems*, pages 8289–8300.

Brunel, N. (2000). Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *J. Comput. Neurosci.*, 8:183–208.

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Foster, J. D., Nuyujukian, P., Ryu, S. I., and Shenoy, K. V. (2012). Neural population dynamics during reaching. *Nature*, 487(7405):51–56.

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Ryu, S. I., and Shenoy, K. V. (2010). Cortical preparatory activity: representation of movement or first cog in a dynamical machine? *Neuron*, 68(3):387–400.

Churchland, M. M. and Shenoy, K. V. (2007). Delay of movement caused by disruption of cortical preparatory activity. *Journal of neurophysiology*, 97(1):348–359.

Dayan, P. and Abbott, L. F. (2001). *Theoretical neuroscience*. Cambridge, MA: MIT Press.

Driscoll, L., Shenoy, K., and Sussillo, D. (2022). Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *bioRxiv*.

Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F., and Ostojic, S. (2021). The role of population structure in computations through neural dynamics. *bioRxiv*, pages 2020–07.

Dum, R. P. and Strick, P. L. (1991). The origin of corticospinal projections from the premotor areas in the frontal lobe. *Journal of Neuroscience*, 11(3):667–689.

Elsayed, G. F., Lara, A. H., Kaufman, M. T., Churchland, M. M., and Cunningham, J. P. (2016). Reorganization between preparatory and movement population responses in motor cortex. *Nature communications*, 7(1):1–15.

Goldman, M. S. (2009). Memory without feedback in a neural network. *Neuron*, 61:621–634.

Guo, Z. V., Li, N., Huber, D., Ophir, E., Gutnisky, D., Ting, J. T., Feng, G., and Svoboda, K. (2014). Flow of cortical activity underlying a tactile decision in mice. *Neuron*, 81:179–194.

Harris, C. M. and Wolpert, D. M. (1998). Signal-dependent noise determines motor planning. *Nature*, 394(6695):780–784.

Heald, J., Wolpert, D., and Lengyel, M. (2023). The computational and neural bases of context-dependent learning. *Ann. Rev. Neurosci.*, qq:1–27.

Hennequin, G., Vogels, T. P., and Gerstner, W. (2012). Non-normal amplification in random balanced neuronal networks. *Physical Review E*, 86(1):011909.

Hennequin, G., Vogels, T. P., and Gerstner, W. (2014). Optimal control of transient dynamics in balanced networks supports generation of complex movements. *Neuron*, 82(6):1394–1406.

Kalidindi, H. T., Cross, K. P., Lillicrap, T. P., Omrani, M., Falotico, E., Sabes, P. N., and Scott, S. H. (2021). Rotational dynamics in motor cortex are consistent with a feedback controller. *Elife*, 10:e67256.

Kalman, R. E. et al. (1960). Contributions to the theory of optimal control. *Bol. soc. mat. mexicana*, 5(2):102–119.

Kao, T.-C. and Hennequin, G. (2019). Neuroscience out of control: control-theoretic perspectives on neural circuit dynamics. *Cur. Op. Neurobiol.*, 58:122–129.

Kao, T.-C., Sadabadi, M. S., and Hennequin, G. (2021). Optimal anticipatory control as a theory of motor preparation: a thalamo-cortical circuit model. *Neuron*, 109.

Kaufman, M. T., Churchland, M. M., Ryu, S. I., and Shenoy, K. V. (2014). Cortical activity in the null space: permitting preparation without movement. *Nature neuroscience*, 17(3):440–448.

Lara, A. H., Elsayed, G. F., Zimnik, A. J., Cunningham, J. P., and Churchland, M. M. (2018). Conservation of preparatory neural events in monkey motor cortex regardless of how movement is initiated. *Elife*, 7:e31826.

Li, N., Chen, T.-W., Guo, Z. V., Gerfen, C. R., and Svoboda, K. (2015). A motor cortex circuit for motor planning and movement. *Nature*, 519:51–56.

Li, W. and Todorov, E. (2004). Iterative linear quadratic regulator design for nonlinear biological movement systems. In *ICINCO (1)*, pages 222–229.

Logiaco, L., Abbott, L., and Escola, S. (2021). Thalamic control of cortical dynamics in a model of flexible motor sequencing. *Cell Reports*, 35(9):109090.

Malonis, P. J., Hatsopoulos, N. G., MacLean, J. N., and Kaufman, M. T. (2021). M1 dynamics share similar inputs for initiating and correcting movement. *bioRxiv*.

Meirhaeghe, N., Riehle, A., and Brochier, T. (2023). Parallel movement planning is achieved via an optimal preparatory state in motor cortex. *Cell Reports*, 42(2):112136.

Michaels, J. A., Dann, B., Intveld, R. W., and Scherberger, H. (2015). Predicting reaction time from the neural state space of the premotor and parietal grasping network. *Journal of Neuroscience*, 35(32):11415–11432.

Miri, A., Warriner, C. L., Seely, J. S., Elsayed, G. F., Cunningham, J. P., Churchland, M. M., and Jessell, T. M. (2017). Behaviorally selective engagement of short-latency effector pathways by motor cortex. *Neuron*, 95:683–696.

Murphy, B. K. and Miller, K. D. (2009). Balanced amplification: a new mechanism of selective amplification of neural activity patterns. *Neuron*, 61:635–648.

Pandarinath, C., O'Shea, D. J., Collins, J., Jozefowicz, R., Stavisky, S. D., Kao, J. C., Trautmann, E. M., Kaufman, M. T., Ryu, S. I., Hochberg, L. R., et al. (2018). Inferring single-trial neural population dynamics using sequential auto-encoders. *Nature methods*, 15(10):805–815.

Rawlings, J. B., Mayne, D. Q., and Diehl, M. (2017). *Model predictive control: theory, computation, and design*, volume 2. Nob Hill Publishing Madison, WI.

Riehle, A. and Requin, J. (1989). Monkey primary motor and premotor cortex: single-cell activity related to prior information about direction and extent of an intended movement. *Journal of neurophysiology*, 61(3):534–549.

Russo, A. A., Bittner, S. R., Perkins, S. M., Seely, J. S., London, B. M., Lara, A. H., Miri, A., Marshall, N. J., Kohn, A., Jessell, T. M., et al. (2018). Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97(4):953–966.

Sauerbrei, B. A., Guo, J.-Z., Cohen, J. D., Mischiati, M., Guo, W., Kabra, M., Verma, N., Mensh, B., Branson, K., and Hantman, A. W. (2020). Cortical pattern generation during dexterous movement is input-driven. *Nature*, 577(7790):386–391.

Schimel, M., Kao, T.-C., Jensen, K. T., and Hennequin, G. (2022). iLQR-VAE : control-based learning of input-driven dynamics with applications to neural data. In *International Conference on Learning Representations*.

Sheahan, H. R., Franklin, D. W., and Wolpert, D. M. (2016). Motor planning, not execution, separates motor memories. *Neuron*, 92(4):773–779.

Shenoy, K. V., Sahani, M., and Churchland, M. M. (2013). Cortical control of arm movements: a dynamical systems perspective. *Annual review of neuroscience*, 36.

Skogestad, S. and Postlethwaite, I. (2007). *Multivariable feedback control: analysis and design*, volume 2. Wiley New York.

Sohn, H., Meirhaeghe, N., Rajalingham, R., and Jazayeri, M. (2020). A network perspective on sensorimotor learning. *Trends in Neurosciences*.

Soldado-Magraner, J., Mante, V., and Sahani, M. (2023). Inferring context-dependent computations through linear approximations of prefrontal cortex dynamics. *bioRxiv*.

Sterling, P. and Laughlin, S. (2015). *Principles of neural design*. MIT press.

Stroud, J. P., Porter, M. A., Hennequin, G., and Vogels, T. P. (2018). Motor primitives in space and time via targeted gain modulation in cortical networks. *Nature neuroscience*, 21(12):1774–1783.

Sun, X., O'Shea, D. J., Golub, M. D., Trautmann, E. M., Vyas, S., Ryu, S. I., and Shenoy, K. V. (2022). Cortical preparatory activity indexes learned motor memories. *Nature*, pages 1–6.

Sussillo, D., Churchland, M. M., Kaufman, M. T., and Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature neuroscience*, 18(7):1025–1033.

Tassa, Y. (2011). *Theory and Implementation of Biomimetic Motor Controllers*. Hebrew University of Jerusalem.

Todorov, E. and Jordan, M. I. (2002). Optimal feedback control as a theory of motor coordination. *Nature neuroscience*, 5(11):1226–1235.

Todorov, E. and Li, W. (2003). Optimal control methods suitable for biomechanical systems. In *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (IEEE Cat. No. 03CH37439)*, volume 2, pages 1758–1761. IEEE.

Yeo, S.-H., Franklin, D. W., and Wolpert, D. M. (2016). When optimal feedback control is not enough: Feedforward strategies are required for optimal control with active sensing. *PLoS computational biology*, 12(12):e1005190.

Zimnik, A. J. and Churchland, M. M. (2021). Independent generation of sequence elements by motor cortex. *Nature neuroscience*, 24(3):412–424.

When and why does motor preparation arise in recurrent neural network models of motor control?

Marine Schimel[@1], **Ta-Chu Kao**[2], **and Guillaume Hennequin**[1]

[1]Computational and Biological Learning Lab, Department of Engineering, University of Cambridge, Cambridge, U.K.
[2]Meta Reality Labs

[@] Corresponding author (mmcs3@cam.ac.uk)

SUPPLEMENTARY MATERIAL

Contents

S1 Choice of the hyperparameters of the model

Our cost function for the delayed single-reaching task was composed of 3 components. The relative weighings of the different terms in our cost, which are hyperparameters of the model, affect the way in which the task is solved. To ensure robustness of our results to hyperparameter changes, we scanned the space of α_{null} and α_{effort} (as the solution is invariant to scaling of the cost, only those relative weighings matter), and evaluated the solutions found across this hyperparameter space for a delayed reach of 300 ms.

Our evaluation was based on multiple criteria. We considered the target to have been successfully reached if the mean distance to the target in the last 200 ms of the movement was lower than 5 mm (for a reach radius of 12 cm). We considered that the requirement to stay still during the delay period was satisfied if the mean torques during preparation were smaller than 0.02 N/m. We computed the preparation index and total cost as described in Equation (5) and Equation (4). We moreover computed the total input energy per neuron as $\frac{1}{N}\int_{-\Delta_{prep}}^{T} \|\boldsymbol{u}\|^2 dt$, and the maximum velocity as $\max_t \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2}$. These various quantities are shown for a range of hyperparameters in Figure S1, with the choice of hyperparameters used throughout our simulations marked with a red star. This shows that the behaviour of the model is consistent across a range of hyperparameter settings around the one we used.

In Figure S2, we illustrate the output of the model for several hyperparameter settings. One can notice that for very small values of α_{effort} the reach is successful, but executed with larger

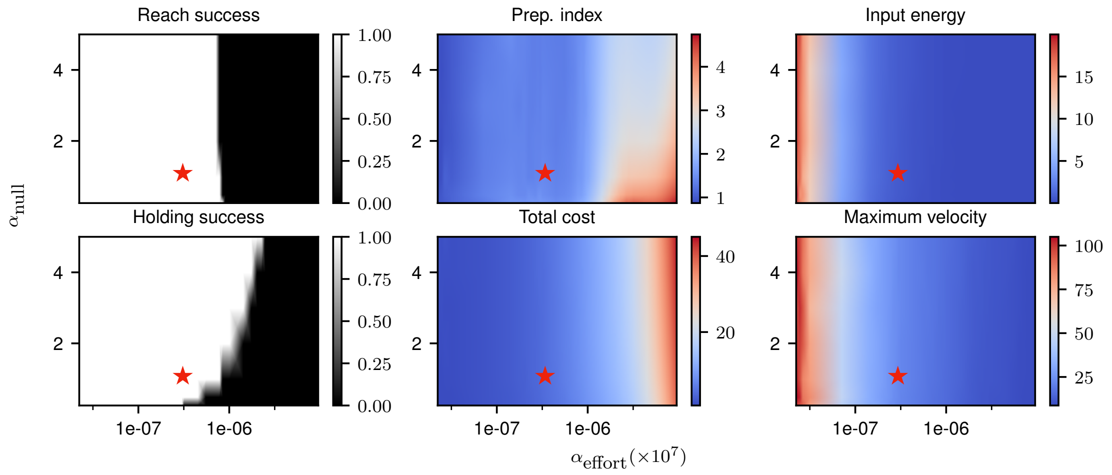

Figure S1: Correlates of the behaviour and control strategy across a wide range of hyperparameters. The "reach success" and "holding success" are set to 1 if the success criterion (see text) is satisfied and 0 otherwise. The task is executed successfully over a wide range of hyperparameters. The red star denotes the set of hyperparameters used in the main text simulations. This configuration was chosen to lie in a region in which the task can be successfully solved, with the performance being robust to small changes in the hyperparameters.

torques and velocity than is necessary – e.g the red and yellow reaches are equally successful but the red one is much faster – which comes at the cost of larger inputs. We chose the set of hyperparameters for our simulations such as to lie in an intermediate regime in which the task is solved successfully, but without requiring more inputs than necessary.

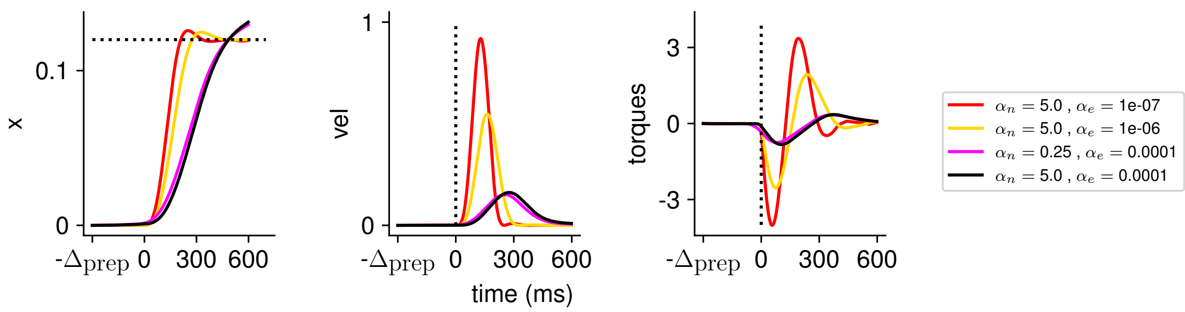

Figure S2: Illustration of the behaviour for several hyperparameter settings. (Left) Hand position along the horizontal axis, with the dotted line denoting the position of the target. (Middle) Temporal profile of the hand velocity. (Right) Temporal profile of the torques driving the hand.

S2 Investigation of the effect of the network decay timescale

Figure 3 highlighted that preparatory inputs tend to consistently arise late during the delay period. We hypothesized that this may be a reflection of the intrinsic tendency of the network dynamics to decay, such that inputs given too early may be "lost". To test this, we changed the characteristic timescale of the dynamics *during preparation only*, leading to the following dynamics:

$$\tau_t \frac{d\boldsymbol{x}(t)}{dt} = -\boldsymbol{x}(t) + \boldsymbol{W}\phi\left[\boldsymbol{x}(t)\right] + \boldsymbol{h} + \frac{\tau_t}{\tau_{\text{mov}}}\boldsymbol{u}(t) \text{ where } \begin{cases} \tau_t = \tau_{\text{prep}} \text{ if t } \leq 0 \\ \tau_t = \tau_{\text{mov}} \text{ if t } \geq 0 \end{cases} \tag{S1}$$

with $\tau_{\text{mov}} = 150\text{ms}$. This allowed us to evaluate whether having dynamics decaying more slowly during preparation led to inputs starting earlier. Note that we also rescaled the inputs during preparation by $\frac{\tau_{\text{prep}}}{\tau_{\text{mov}}}$, to ensure that the effective cost of the inputs was not affected by the timescale change.

As shown in Figure S3, inputs started rising earlier when the network's decay timescale was longer. This was consistent with the hypothesis that the length of the window of preparation that the optimal controller uses depends on the network's intrinsic timescale.

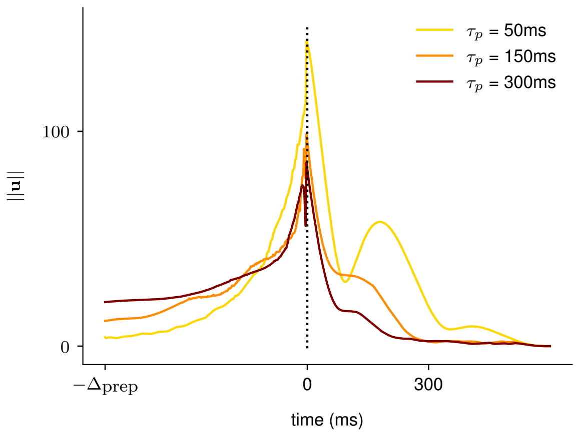

Figure S3: **Illustration of the effect of the characteristic neuronal timescale on the temporal distribution of the inputs.** We modified the characteristic neuronal timescale of the ISN during preparation only, and assessed how that changed the temporal distribution of inputs for 3 different timescales ($\tau_{\text{prep}} = 50\text{ms}$, $\tau_{\text{prep}} = 150\text{ms}$, $\tau_{\text{prep}} = 300\text{ms}$, top to bottom). As hypothesized, inputs started rising earlier during the preparation window when the decay timescale of the network was longer.

S3 Additional results in the 2D system

Our visualization of the behaviour of 2D networks in Figure 4 allowed us to identify features of the dynamics that were well-suited to predicting preparation. Below, we compute α and β numerically and analytically in 2D oscillatory and nonnormal networks, to gain insights into how these quantities vary with the networks' dynamics. We then show how preparation can be predicted highly accurately across a large number of 2D systems, using only those quantities to summarize the network dynamics.

S3.1 Controllability and observability computations

In Figure S4, we computed α and β numerically, as a function of the connectivity strength and the choice of readout, for the nonnormal and the oscillatory motifs shown in Figure 4. This highlights the very different behaviours of the two networks, which are to some extent also reflected in higher-dimensional models. In particular, we find a strong effect of the alignment between the readout and the network dynamics in nonnormal networks, while α and β are independent of θ_C in oscillatory networks. Interestingly, we see that β is constant across all oscillatory networks, while α increases with w.

As the reduced 2D model is more amenable to mathematical analysis than its high-dimensional counterpart, we can gain further insights into the origin of these differences by computing $\alpha(w, \theta_C)$ and $\beta(w, \theta_C)$ analytically.

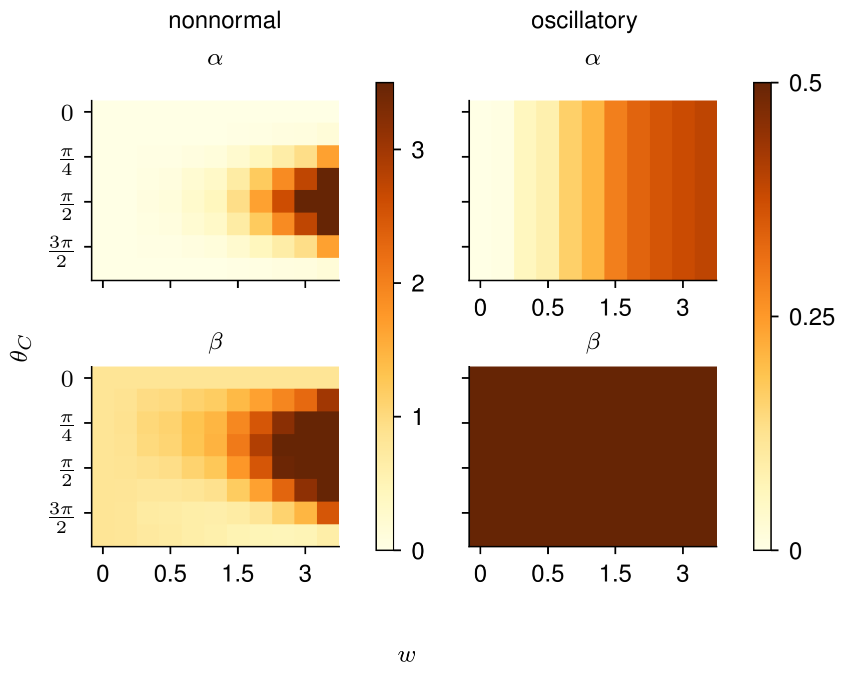

Figure S4: Illustration of α and β as a function of θ_C and w in the 2D networks.

Recall that the observability Gramian Q of a linear input-driven dynamical system satisfies

$$\boldsymbol{A}^T\boldsymbol{Q} + \boldsymbol{Q}\boldsymbol{A} + \boldsymbol{C}^T\boldsymbol{C} = 0 \tag{S2}$$

and the controllability Gramian satisfies

$$\boldsymbol{A}\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{B}\boldsymbol{B}^T = 0, \tag{S3}$$

and that we defined $\mathrm{Tr}(\boldsymbol{C}^{\perp}\boldsymbol{Q}\boldsymbol{C}^{\perp^T})$ and $\beta = \mathrm{Tr}(\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^T)$ where $\boldsymbol{C}^{\perp}$ denotes the nullspace of the readout matrix. Below, we compute these quantities for the 2D oscillatory and nonnormal networks, with $\boldsymbol{B} = \boldsymbol{I}$ and $\boldsymbol{C}$ a unit-norm vector whose direction we parametrize via a quantity θ_C. Note that we ignore the effect of dt and τ in the mathematical analysis, as those quantities can straightforwardly be included in the final result via a rescaling of w and $\boldsymbol{B}$.

Oscillatory network In the case of $\boldsymbol{A} = -\boldsymbol{I} + \boldsymbol{S}$ where $\boldsymbol{S}$ is a skew-symmetric network (i.e $\boldsymbol{S}^T = -\boldsymbol{S}$), Equation (S3) is solved by $\boldsymbol{P} = \boldsymbol{I}/2$ independently of the value of $\boldsymbol{S}$. This explains why $\beta = \mathrm{Tr}(\boldsymbol{C}\boldsymbol{P}\boldsymbol{C}^T) = \frac{1}{2}\|\boldsymbol{C}\|^2$ is independent of both the connectivity strength w and the orientation of the readout θ_C for skew-symmetric networks (see Figure S4; bottom right). Practically, this means that skew-symmetric networks are equally controllable in all directions: when driven by random inputs, these networks display isotropic activity of equal variance along all directions. Moreover, as w controls the oscillation frequency of the network, but does not change the decay timescale of the eigenmodes, the amount of variance generated by a random stimulation is independent of w. Interestingly, we can see in Figure S4 (top right) that α displays a different behaviour, and increases with w. As highlighted above, skew-symmetric systems are rotationally symmetric. Without loss of generality, we can thus define our 1D vector to read out the first unit, i.e $\boldsymbol{C} = \begin{bmatrix}1 & 0\end{bmatrix}$.

The observability Gramian must satisfy

$$\boldsymbol{A}^T\boldsymbol{Q} + \boldsymbol{Q}\boldsymbol{A} + \boldsymbol{C}^T\boldsymbol{C} = 0 \implies \begin{bmatrix}-1 & w \\ -w & -1\end{bmatrix}\boldsymbol{Q} + \boldsymbol{Q}\begin{bmatrix}-1 & -w \\ w & -1\end{bmatrix} = \begin{bmatrix}-1 & 0 \\ 0 & 0\end{bmatrix}. \tag{S4}$$

This can be found in closed-form by solving the 2D system of equations, yielding

$$\boldsymbol{Q} = \begin{bmatrix}\frac{1}{4} + \frac{1}{4(1+w^2)} & -\frac{w}{4(1+w^2)} \\ -\frac{w}{4(1+w^2)} & \frac{w^2}{4(1+w^2)}\end{bmatrix}. \tag{S5}$$

From there, we obtain $\alpha = \mathrm{Tr}(\boldsymbol{C}^{\perp}\boldsymbol{Q}\boldsymbol{C}^{\perp T}) = \frac{w^2}{4(1+w^2)}$. As found empirically, this quantity will initially increase before plateauing towards 1/4 as w becomes large.

One might wonder why observability displays such a dependence on the oscillatory frequency of the network, even though the network is rotationally symmetric, and w does not affect the decay timescale. As highlighted in Equation (S2), controllability and observability Gramian would be identical for a skew-symmetric system if $\boldsymbol{C} = \boldsymbol{I}$. However, a feature of the systems we consider is the existence of a nullspace, i.e the fact that the readout $\boldsymbol{C}$ only targets a subset of dimensions across the whole space (implying that $\boldsymbol{C}^T\boldsymbol{C}$ is a low-rank matrix). Intuitively, the reason why α increases with w while β is constant in skew-symmetric networks can be understood as follows: α is computing how much *readout activity* a vector initialized in the nullspace of $\boldsymbol{C}$ will generate, while β is computing the amount of energy that will be generated *across all directions* by a vector initialized in the readout space. Thus, assuming once again $\boldsymbol{C} = \begin{bmatrix}1 & 0\end{bmatrix}$ and $\boldsymbol{C}^{\perp} = \begin{bmatrix}0 & 1\end{bmatrix}$, the activity of vectors initialized along $\boldsymbol{C}$ and $\boldsymbol{C}^{\perp}$ respectively and evolving autonomously from there is given by $\boldsymbol{v}_C(t) = \begin{bmatrix}e^{-t}\cos(wt) & e^{-t}\sin(wt)\end{bmatrix}$ and $\boldsymbol{v}_{C^{\perp}}(t) = \begin{bmatrix}e^{-t}\sin(wt) & -e^{-t}\cos(wt).\end{bmatrix}$

From there, we can compute $\beta = \int_0^{\infty} \|\boldsymbol{v}_C(t)\|^2 dt = \int_0^{\infty} e^{-2t} dt = \frac{1}{2}$. Thus, as found above, only the decay timescale of the envelope (fixed to 1 here) affects the value of β.

Importantly, α will instead have a dependence on w arising from the fact that it depends on the size of the activity *projected into the readout*, as

$$\beta = \int_0^{\infty} \|\boldsymbol{v}_{C^{\perp}}(t)^T\boldsymbol{C}\|^2 dt = \int_0^{\infty} e^{-2t}\sin^2(wt)dt \tag{S6}$$

$$= \frac{1}{2}\int_0^{\infty} e^{-2t}(1-\cos(2wt))dt \tag{S7}$$

$$= \frac{1}{2}\int_0^{\infty} e^{-2t} - \Re e^{-2(1-iw)t} dt \tag{S8}$$

$$= \frac{1}{4} - \frac{1}{4(1+w^2)} \tag{S9}$$

$$= \frac{w^2}{4(1+w^2)}. \tag{S10}$$

The dependence of this quantity on w can be understood by the fact that activity patterns initialized in the readout nullspace benefit from the existence of rotational dynamics, which allows them to be read-out before the activity decays completely.

Nonnormal network In the nonnormal network, we have $\boldsymbol{A} = -\boldsymbol{I} + \boldsymbol{W} = \begin{bmatrix}-1 & 0 \\ w & -1\end{bmatrix}$. The nonnormal 2D system, unlike its oscillatory counterpart, does not have rotational symmetry. Thus, to remain general, we will consider $\boldsymbol{C}(\theta_C) = \begin{bmatrix}\cos\theta_C & \sin\theta_C\end{bmatrix}$, and $\boldsymbol{C}^{\perp}(\theta_C) = \begin{bmatrix}-\sin\theta_C & \cos\theta_C\end{bmatrix}$. Solving Equation (S2) for $\boldsymbol{B} = \boldsymbol{I}$ leads to an expression for the controllability Gramian of the nonnormal system as

$$\boldsymbol{P}(w) = \begin{bmatrix}\frac{1}{2} & \frac{w}{4} \\ \frac{w}{4} & \frac{1}{2}+\frac{w^2}{4}\end{bmatrix}. \tag{S11}$$

Similarly, computation of the observability Gramian leads to

$$\boldsymbol{Q}(w,\theta_C) = \begin{bmatrix}\frac{w^2\sin^2\theta_C}{4} + \frac{\cos\theta_C\sin\theta_C w}{2} + \frac{\cos^2\theta_C}{2} & \frac{w\sin^2\theta_C}{4} + \frac{\cos\theta_C\sin\theta_C}{2} \\ \frac{w\sin^2\theta_C}{4} + \frac{\cos\theta_C\sin\theta_C}{2} & \frac{\sin^2\theta_C}{2}\end{bmatrix}. \tag{S12}$$

We can then compute

$$\alpha(\theta_C, w) = \boldsymbol{C}^{\perp T}\boldsymbol{Q}\boldsymbol{C}^{\perp} = \frac{w^2}{4}\sin^4\theta_C \tag{S13}$$

and

$$\beta(\theta_C, w) = \boldsymbol{C}^T \boldsymbol{P} \boldsymbol{C}^{\perp} = \frac{(w \sin\theta_C + \cos\theta_C)^2 - \cos^2\theta_C + 2}{4}. \tag{S14}$$

This highlights the dependence of α and β on θ_C, which can also be seen in Figure S4 (left). Interestingly, these expressions also make evident the supralinear scaling of α and β with w in nonnormal networks. Note however that we never investigate preparation in the very large w regime, as the simulation of such networks with discretized dynamics is prone to numerical issues.

S3.2 Predicting preparation in 2D networks

To assess how well preparation could be predicted from the control-theoretic properties α and β (c.f. main text) of 2D networks, we generated 20000 networks with weight matrix

$$\boldsymbol{W}(a, \omega, w_{\text{ff}}) = \begin{bmatrix} a & \frac{1}{2}\left(w_{\text{ff}} + \sqrt{w_{\text{ff}}^2 + 4\omega^2}\right) \\ \frac{1}{2}\left(w_{\text{ff}} - \sqrt{w_{\text{ff}}^2 + 4\omega^2}\right) & a \end{bmatrix} \tag{S15}$$

where $a \sim \mathcal{U}(0, 0.8)$, $\omega \sim \mathcal{U}(0, 4)$, and $w_{\text{ff}} \sim \mathcal{U}(0, 4)$. Equation (S15) implies that $\boldsymbol{W}$ has a pair of complex-conjugate eigenvalues $a \pm i\omega$, and also embeds a feedforward coupling of strength w_{ff} from the second to the first dimension. For each network configuration, we computed the corresponding values of α and β. To confirm our intuition that the preparation index should increase with α and decrease with β, we first attempted to fit prep. index $= c_0 + c_1 \frac{\alpha}{\beta}$. Interestingly, we found that while this quantity was positively correlated with the preparation index across networks, a substantial fraction of variance remained unexplained (test $R^2 = 0.16$). Labelling the preparation index by the rotational frequency of the network highlighted that a substantial fraction of the variance across networks came from this timescale of oscillations (Figure S5, left). Indeed, a regression model of the for prep. index $= c_0 + c_1 \omega \frac{\alpha}{\beta}$ captured 80% of the variance in preparation index, yielding an accurate fit across networks with only two free parameters (Figure S5, right).

We stress that the predictive power of these simple fits is remarkable given that the preparation index comes out of a complex process of optimization over control inputs. Thus, the control-theoretic quantities α and β appear to appropriately summarize the benefits of preparation for individual networks.

The fact that the preparation index also grows with ω can be understood by considering the alignment between the activity trajectories which the network can autonomously generate and those that are required for solving the motor task. Indeed, a network that is intrinsically unable to generate outputs with the right oscillatory timescale would have to rely on movement-related inputs, i.e. would have a low preparation index. As observed here, the network's characteristic frequency has a big impact in 2D networks, consistent with ω determining the *only* oscillatory pattern that the network can generate on its own. For high-dimensional networks, however, we did not have to incorporate such a measure of compatibility between task requirements and network dynamics (c.f. Figure 5). We speculate that this is due to averaging effects. Indeed, larger networks possess a wide range of intrinsic oscillatory timescales, and the readout matrix - which here was not aligned to the network's dynamics in any specific way - is expected to read out a little bit of all frequencies, including task-appropriate ones.

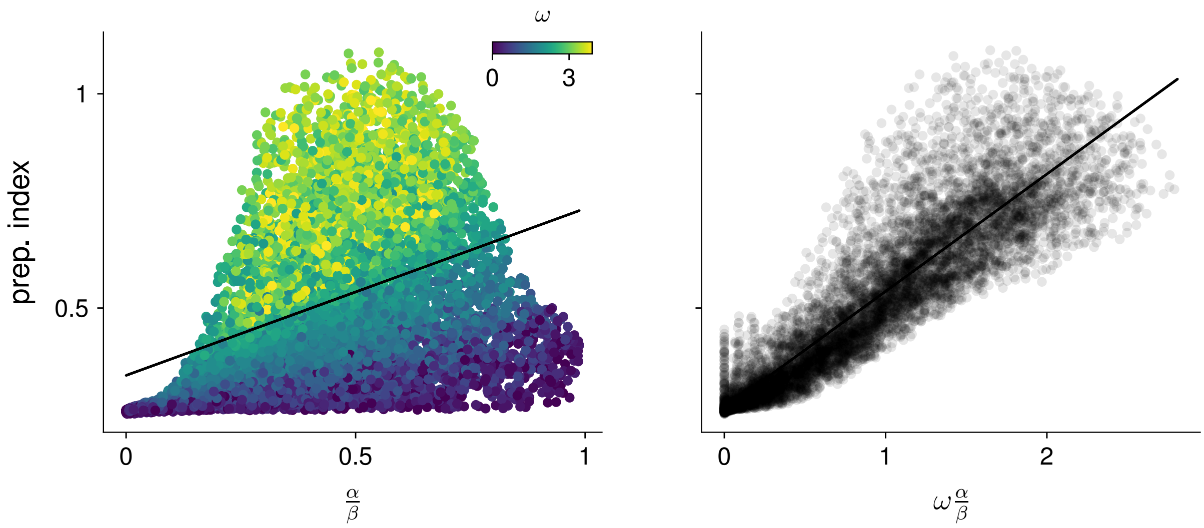

Figure S5: **Predicting the preparation index from characteristic network quantities.** We evaluated how well the preparation index could be predicted as a linear function of $\frac{\alpha}{\beta}$ (left). A substantial amount of residual variance appeared to arise from variability in the oscillation frequency ω (color). Accounting for this frequency by regressing the preparation index against $\omega\frac{\alpha}{\beta}$ gave a better fit (right).

S4 Comparison across networks

Our main investigation was largely focused on on behaviour of inhibition-stabilized networks, which are believed to constitute good models of M1. We however found that the expression we derived to obtain a network's preparation index from its control-theoretic properties generalized across to other types of networks. Below, we detail the other network families we considered, and show how their dynamics qualitatively differ from the ISN, although their preparation can be predicted using the same quantities.

We modelled three additional classes of networks: randomly connected networks with either (i) unstructured or (ii) skew-symmetric connectivities, (iii) a surrogate network obtained by applying a similarity transformation to the ISN that preserved its eigenvalue spectrum but eliminated any "nonnormality" (i.e, we found $\mathcal{T}$ such that $\tilde{\boldsymbol{A}} = \mathcal{T}^{-1}\boldsymbol{A}\mathcal{T}$ where $\tilde{\boldsymbol{A}}$ was a diagonal matrix with the same eigenvalues as $\boldsymbol{A}$). Note that we did not apply the transformation to the readout or input matrices, such that the transfer function of the system was changed by our transformation. This was voluntary, as we were interested in the effect that transforming the dynamics would have on the input-output response. These networks were chosen for the diversity of dynamical motifs they exhibit: combinations of rapidly and slowly decaying modes, oscillations, and transient dynamics. Moreover, each of these network families could be sampled from in a straightforward manner, allowing to compute results across many instantiations of each network type. We again used random readout matrices not specifically adjusted to the dynamics of the network nor to the motor task. To get an intuition for how different networks solve the task, we generated one network from each family and qualitatively compared their inputs and internal activations when performing the same delayed reach (Figure S6A). We first considered an unconnected network, i.e. a network whose recurrent weights were all 0. Unsurprisingly, this network had no use for a preparation phase. Indeed, there is no benefit to giving early inputs as the network is unable to amplify them. More surprisingly, a random network with a much stronger connectivity – as can be seen in its eigenvalue spectrum forming a small ball of radius close to 1 (Figure S6A(top)) – also displayed very little preparation. The strong, visually apparent similarity between the inputs to the random and unconnected

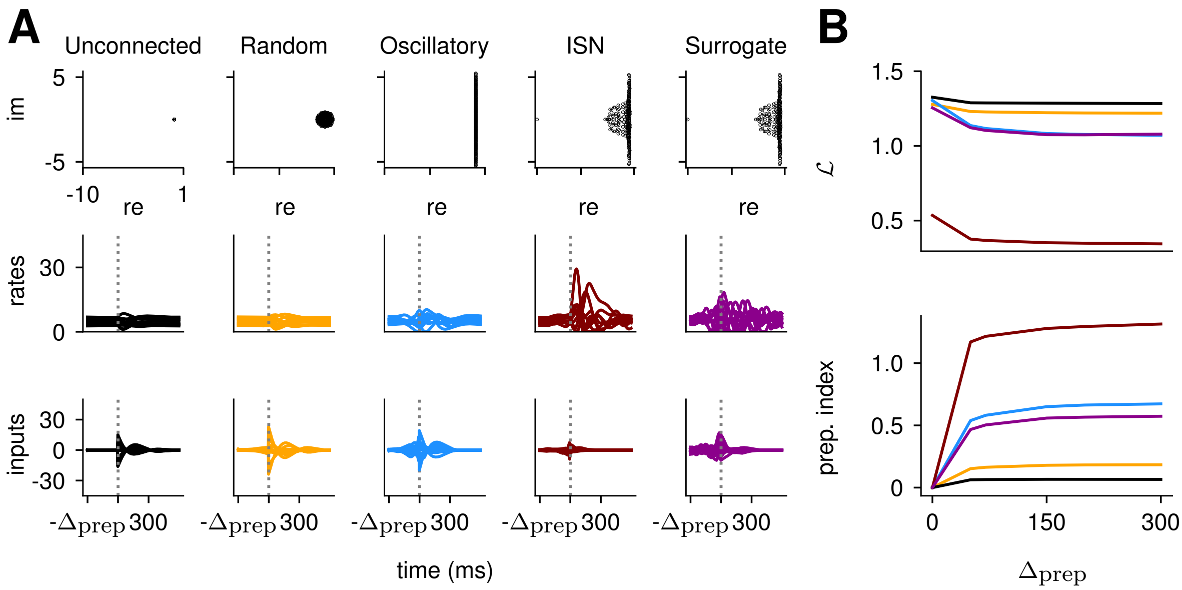

Figure S6: **Preparation arises across a range of network architectures** : neural correlates of the reach are shown for 5 different networks (A), alongside the loss and prep. index as a function of Δ_{prep}. **(A)** Eigenvalue spectrum (top), internal network activations (middle) and inputs (bottom) for different network types. The unconnected network does not rely on preparatory inputs at all. The random network with weights draw from $\mathcal{N}(0, 0.95/\sqrt{N})$ uses very little delay period inputs while the skew-symmetric network with $\mathcal{N}(0, 4/\sqrt{N})$ shows a substantial amount of inputs during the delay period. The inhibition-stabilized network can be seen to rely most on preparation, more so than the similarity transformed ISN. **(B)** Loss (top) and preparation index (bottom) as a function of delay period length for the different networks. The unconnected and random networks can be seen to benefit very little from longer preparation times. Indeed, even as Δ_{prep} increases, their amount of preparatory inputs remains very close to 0. On the other hand, the skew-symmetric network and the ISN use preparatory inputs (bottom), which allow them to have a lower loss for larger values of Δ_{prep}. Interestingly, the surrogate ISN prepares considerably less than the full ISN.

networks suggests that the optimal way of controlling the random network relies largely on ignoring its internal dynamics and solving the task almost entirely in an input-driven regime. The example skew-symmetric network, which had imaginary eigenvalues only (ranging between -5.5 and 5.5), displayed considerably more preparation, but still relied on strong inputs during the movement phase that resembled those of the unconnected and random networks. Finally, the ISN relied much more on preparation; the small inputs it receives are strongly amplified into large activity patterns owing to its strong, nonnormal recurrent connectivity. Interestingly however, the similarity transformed ISN lost much of that ability to amplify inputs, instead displaying dynamics resembling that of the skew-symmetric network. This highlights the effect of the ISN's nonnormal dynamics in shaping the network's activity and optimal inputs.

Next, we assessed more directly how beneficial preparation was for the different networks. We evaluated how the total loss and preparation index evolved as a function of the delay period length (Figure S6B). As expected, the control of networks that relied on preparation (skew-symmetric and ISN) benefitted more from longer delays. The ISN has markedly lower control cost and higher preparation index than other networks, reflecting the fact that even weak (thus energetically cheap) inputs were sufficient to produce internal activity and thus output torques of the desired magnitude (c.f. Figure S6A, right).

The above results give a sense of the range of possible dynamics that different types of networks display. Interestingly, despite these differences, we showed in Figure 5 that the preparation index could be predicted with a simple formula across all networks.