Complementary roles for hippocampus and anterior cingulate in composing continuous choice

Justin M. Fine[1]*†, Assia Chericoni[1]*, Gabriela Delgado[2], Melissa C. Franch[1], Elizabeth A. Mickiewicz[1], Ana G. Chavez[1], Eleonora Bartoli[1], Danika Paulo[1], Nicole R. Provenza[1], Andrew Watrous[1], Seng Bum Michael Yoo[3], Sameer A. Sheth[1] and Benjamin Y. Hayden[1]

[1] Department of Neurosurgery, Baylor College of Medicine
[2] Department of Neuroscience, University of Minnesota
[3] Institute for Basic Science, Sungkyunkwan University

* Denotes equal contribution

†Corresponding author

Acknowledgements: We thank Paul Schrater, Xaq Pitkow, Jon Pearson, and Scott Linderman for invaluable discussions and insights on solving the controller problems. We thank Jeff Johnston for discussion on neural analysis. We also thank Joshua Adkinson, Raissa Mathura, and Victoria Pirtle for invaluable assistance. This project was supported by NIH grants R01 DA038615, R01 MH125377, U01 NS121472, and R01 MH129439, and by the McNair foundation.

Statement of Conflict: SAS is a consultant for Boston Scientific, Zimmer Biomet, Neuropace, Koh Young, Abbott; Co-founder, Motif Neurotech. None of the authors have any competing interests to declare.

ABSTRACT

Naturalistic, goal directed behavior often requires continuous actions directed at dynamically changing goals. In this context, the closest analogue to choice is a strategic reweighting of multiple goal-specific control policies in response to shifting environmental pressures. To understand the algorithmic and neural bases of choice in continuous contexts, we examined behavior and brain activity in humans performing a continuous prey-pursuit task. Using a newly developed control-theoretic decomposition of behavior, we find pursuit strategies are well described by a meta-controller dictating a mixture of lower-level controllers, each linked to specific pursuit goals. Examining hippocampus and anterior cingulate cortex (ACC) population dynamics during goal switches revealed distinct roles for the two regions in parameterizing continuous controller mixing and meta-control. Hippocampal ensemble dynamics encoded the controller blending dynamics, suggesting it implements a mixing of goal-specific control policies. In contrast, ACC ensemble activity exhibited value-dependent ramping activity before goal switches, linking it to a meta-control process that accumulates evidence for switching goals. Our results suggest that hippocampus and ACC play complementary roles corresponding to a generalizable mixture controller and meta-controller that dictates value dependent changes in controller mixing.

INTRODUCTION

Our understanding of how the brain implements reward-based decisions (including foraging) is largely based on studies of discrete choices over static options. Yet many of the decision problems organisms face in the real world are both continuous and interactive (Gordon et al., 2021; Maseli et al., 2023; Yoo et al., 2021). Consider a predator pursuing fleeing prey (Fabian et al., 2018; Fine et al., 2024; Yoo et al., 2020). At each moment, the predator can choose a trajectory from a continuous set of actions; across time, the predator must continuously adjust its actions to move closer to its goal, integrating feedback and predictions of future states. When there are multiple prey, the predator must also decide whether and when to adjust which goal is pursued (Cisek, 2012; Gomi and Kawato, 1993; Sridhar et al., 2021). Decision-makers may even blend their pursued goals, allowing them to hedge against the risk of needing large course corrections if they change their mind (Gallivan et al., 2016; Haith et al., 2015; Yang et al., 2022).

In continuous choice with multiple goals, including prey pursuit, the value landscape for choosing the best goal has multiple stable states because different control actions are required for distinct goals. Therefore, choosing the best goal at any moment requires an action controller that can solve a non-trivial and nonlinear goal optimization problem (Bertsekas, 1996; Sutton and Barto, 1998). While such nonlinear controllers are theoretically attainable, a new nonlinear controller would have to be learned for each new set of goals experienced, and could not be generalized outside the goals it was designed around (Theorodou et al, 2010). A solution to this problem is the usage of compositional control, which builds on modern control theory, artificial intelligence and robotics (Dvijotham and Todorov, 2012; Peng et al., 2019; Haarnoja et al., 2024). Optimal control theory has shown that the nonlinear control needed for solving continuous multi-goal decisions can be approximated via composition of individual linear controllers (Dvijotham and Todorov, 2012; Gomez et al., 2014; Pan et al., 2015; Peng et al., 2019; Wolpert and Kawato, 1998). In this framework, each individual controller is focused on resolving the best actions for a single goal, and the system's behavior reflects a blended weighting of multiple controllers. A key benefit of

compositional control is that a shared weighting mechanism can be generalized across environments with different goal-specific controllers (Gomez et al., 2014; Lake and Baroni, 2018; Matsuo et al., 2022; Todorov, 2009). A compositional controller also requires a separate meta-controller, which monitors both the environmental context and internal shifts of goal priorities to determine the need for changes in the blending of individual controllers. Understanding if and how the brain might implement a flexible, compositional control strategy is crucial, as it can elucidate the neural mechanisms enabling humans to arbitrate between competing goals in dynamic pursuit-like environments.

One implication of the blending hypothesis is that the brain should compute and explicitly represent the goal-specific policy weighting. Brain regions coding for a policy weighting should have preferential access to information needed for creating mappings of goals (e.g., goal distance) and goal specific control policies, and be able to compose them together (Kurth-Nelson et al., 2023; Whittington et al., 2022). We hypothesized this goal policy weighting would be present in the hippocampus, given its demonstrated role in encoding goal related information in cognitive maps (Behrens et al., 2018; Wikenheiser and Schoenenbaum, 2016). Indeed, a good deal of evidence links the hippocampus to encoding of goal specific information (McNaughton and Bannerman, 20254; Nyberg et al., 2022; Wikenheiser and Redish, 2015), including the coding of goal-directed actions (Zutshi et al., 2025), goal distance or goal heading (Chadwick et al., 2015; Howard et al., 2014; Sarel et al., 2017; Ormond and O'keefe, 2022), and place-like coding modulated by goals (Muhle-Karbe et al., 2023; Jercog et al., 2019). The hypothesis that the hippocampus represents the composition of controllers is further supported by previous theoretical and empirical work indicating the hippocampus forms maps and representations of novel objects or action policies through compositional coding (Barron et al., 2013; Bakkermans et al., 2025; Garvert et al., 2023; Kurth-nelson et al., 2023; Schwartenbeck et al., 2023; Whittington et al., 2020). This capacity for compositional coding and mapping of goal related information suggests that the hippocampus is ideally positioned to combine goal specific controllers during continuous choice.

The second implication of our hypothesis is the brain must exert meta-control over the policy blending strategies. The meta-controller must monitor and regulate lower-level control processes and shifts in goals by tracking changes in the task environment (Eppinger et al., 2021; Musslick et al., 2024; Yeung and Summerfield, 2012). A wealth of evidence links the anterior cingulate cortex (ACC) to monitoring and regulation processes across a variety of contexts (Alexander and Brown, 2011; Akam et al., 2021; Kolling et al., 2016; Sarafyazd and Jazayeri, 2019; Shenhav et al., 2013). It is also involved in high-level response strategy selection (Kennerley et al., 2006; Sheth et al., 2012; Shima and Tanji, 1998). During decisions, ACC often shows a clear ramping of activity leading up to choices and strategy switches, possibly reflecting an accumulation of evidence for computing decisions (Balewski et al., 2023; Blanchard et al., 2015; Hayden et al, 2011; Kane et al., 2022; Sarafyazd and Jazayeri, 2019; Voloh et al., 2023). Building on these findings, we hypothesize that ACC operates as a meta-controller in the context of continuous and multi-goal choice; we therefore predicted that switches in goals should be preceded by ACC ramping signals that terminate at the time of the switch.

To test these predictions, we examined neural activity in the human brain during performance of a joystick-controlled interactive prey-pursuit task (**Figure 1A**). We developed a novel mixture of controllers model that allowed us to infer subjects' moment-to-moment blending of different prey specific controllers. Examining goal-switching decisions, we found that hippocampus population dynamics closely mirrored the states of the policy mixing behavioral dynamics. ACC showed ramping activity that predicted these decisions before they occurred, with ramping dynamics that depend on the value of switching goals. These results demonstrate that hippocampus and ACC have complementary roles corresponding to implementation of continuous choice strategies and meta-control, respectively.

RESULTS

Thirteen human participants performed a joystick-controlled prey-pursuit task (**Figure 1A-D,** Chericoni et al., 2025). On each trial, the participant used a joystick to move an avatar freely in a two-dimensional area to pursue fleeing prey (**Supplementary Video**). On each trial, there were two prey that evaded the participant. The prey were specified by different colors, indicating their value which changed along the dimensions of reward and speed (**Figure 1A-B**). Participants completed trials with three different combinations of prey value, ranging from no difference to a large difference in value. As the difference in prey value increased, the trials also took longer (as anticipated, because the larger value prey moved faster, increasing the difficulty, **Figure 1D**). Capture rate performance was high; the mean across-participant capture rate was 85.3% (78.4% - 92.1%; **Figure 1E**).

Inferring continuous choice strategies

In discrete decision-making tasks, assessing a participant's choice between options is trivial. In continuous decision contexts, inferring a participant's moment-to-moment decision strategy is quite a bit more challenging. We need to estimate how their control signal (in this case, avatar velocity) is driven by one or more goals, in this case, by either a single prey or mixture of prey. One difficulty with identifying these mixing strategies is that participants may use a variety of low-level control strategies for pursuing individual targets. For example, participants may opt to minimize positional error, velocity error, integral of position error, estimated future position error, or any combination of these. The control strategy is unknown to the experimenter. Therefore, we must estimate both the participant's per-target control strategy and the mixture of controllers.

The problems of estimating low-level controllers and target mixing are interrelated. To identify the target mixing at every point in time, we must estimate both the participant's target specific controller and the mixture weight that combines these target controllers (W_t; see **Methods** eq. 2-3). For the case of two possible pursuit targets, W_t encodes the continuum from fully pursuing one target ($W_t = 0$), to fully mixing

168 targets (W_t = 0.5), to fully pursuing the other target (W_t = 1). W_t determines how much
169 each target in our task contributes to the participant's momentary control signal. If
170 instead of estimating the controllers, however, we only used current position error
171 controllers in tandem with estimating W_t, this would provide incorrect estimates of W_t if
172 they are using a different control strategy (e.g., current and future position error; **Figure**
173 **1F**). To overcome this problem, we developed a model-based decomposition that
174 leverages optimal control and compositionality theory to estimate these controllers and
175 their mixtures (Todorov, 2009).

176 For each trial, we estimated W_t by modelling the participant's joystick control
177 using a second-order linear dynamic system (i.e., acceleration; see **Methods**). This
178 process involved optimizing separate controller gains for each target simultaneous with
179 optimizing W_t. We repeated this process for all types and combinations of controller
180 classes (see **Methods**), ranging from the usage of current position errors between
181 cursor and target, to integral error, or one step ahead future position error (see
182 **Methods** for a list of all classes). We used a Bayesian model averaging framework to
183 recover the per trial estimate of W_t. This involved weighting W_t for each controller class
184 combination by the posterior probability of that fit.

185 To test the validity of our approach, we performed model validation analysis of
186 ground-truth simulated W_t signals. We could recover W_t with high fidelity, and
187 independent of controller classes or W_t complexity, quantified by a linear regression of
188 the predicted and ground-truth W_t ($r^2 = 0.87 \pm 0.08$; **Table S1**). We also found W_t is
189 largely unrelated to subject speed ($r = 0.14 \pm 0.18$, $p = 0.38$), relative distance ($r = 0.11 \pm$
190 0.06, $p=0.53$) or relative speed ($r = 0.09 \pm 0.02$, $p=0.25$) of the two prey across
191 subjects. Importantly, these results indicate the controller mixing is identifiable and is a
192 unique construct that is not just a reflection of a singular state variable such as relative
193 distance.

194 We compared the fit of the blended controller model to participant behavior
195 against two other potential strategies: (1) all-or-none target selection (i.e., winner-take-
196 all) or (2) a blending strategy that mixes target input (e.g., relative position to subject)
197 and uses a single controller. Notably, the all-or-none strategy is nested within the

blending model, but the input blending model is distinct (**Figure 1G-I**, see proof in **S2**). Using simulations and ground-truth model recovery, we found a low confusion rate across models (correct classification for blending: 93.2%, all-or-none: 85.8%, and input mixing: 89.2%). We next evaluated fits of participant behavior with a Bayesian model comparison, with per trial and per subject model fits. The model posterior probabilities were used to adjudicate between the propensity of each strategy type. Model comparison indicated all subject's behavior was best captured by a blended controller strategy (84% $\pm$ 8%, posterior model probability across subject; **Figure 1I**).

We used the recovered W_t trajectories to understand participants' pursuit behavior (example in **Figure 1K**). To visualize W_t behavior, we plot their subject averaged distributions across different levels of reward difference between prey, including a small and large difference (**Figure 1J**). If we define the pursuit of the more rewarding targets as moments when $W_t>0.85$ and pursuit of the lower reward targets as moments when $W_t<0.15$, we see the expected behavior. Participants spent a larger portion of time pursuing the more rewarding target in the small ($t = 16.12$, $p < 0.0001$) and large ($t = 15.01$, $p < 0.0001$) reward difference trials. The majority of the time (55.4%), pursuit behavior was spent in a blended state ($0.15 < W_t < 0.85$). In summary, participants pursued the more rewarding target more often as the reward difference increased but generally hedged their actions between controllers, acting in accordance with a softmax weighted decision strategy.

Multiplexed states underlie goal switching

To determine which factors drove switches between targets, we first identified target switches (**Methods**), e.g., time series of switching from $W_t < 0.15$ to $W_t > 0.85$ (**Figure 1K**). We used a hierarchical Bayesian logistic model to determine factors that lead to changes in propensity for target switching (% change in odds of switching; **Figure 1L**). Differences in the prey value increased the likelihood of switching dramatically (85.9%). The speed at which the switched-to target was moving away from the participant also predicted switching (49.2%), as well as the elapsed trial length (19.7%). Switching behavior was also driven by a complex interaction between the

distance from the abandoned target and differences in prey value (34.2%). This pattern supports the idea that participants switched as the currently pursued prey got farther away, particularly with larger differences in prey value (**Figure 1L**), alongside other complex predictor interactions.

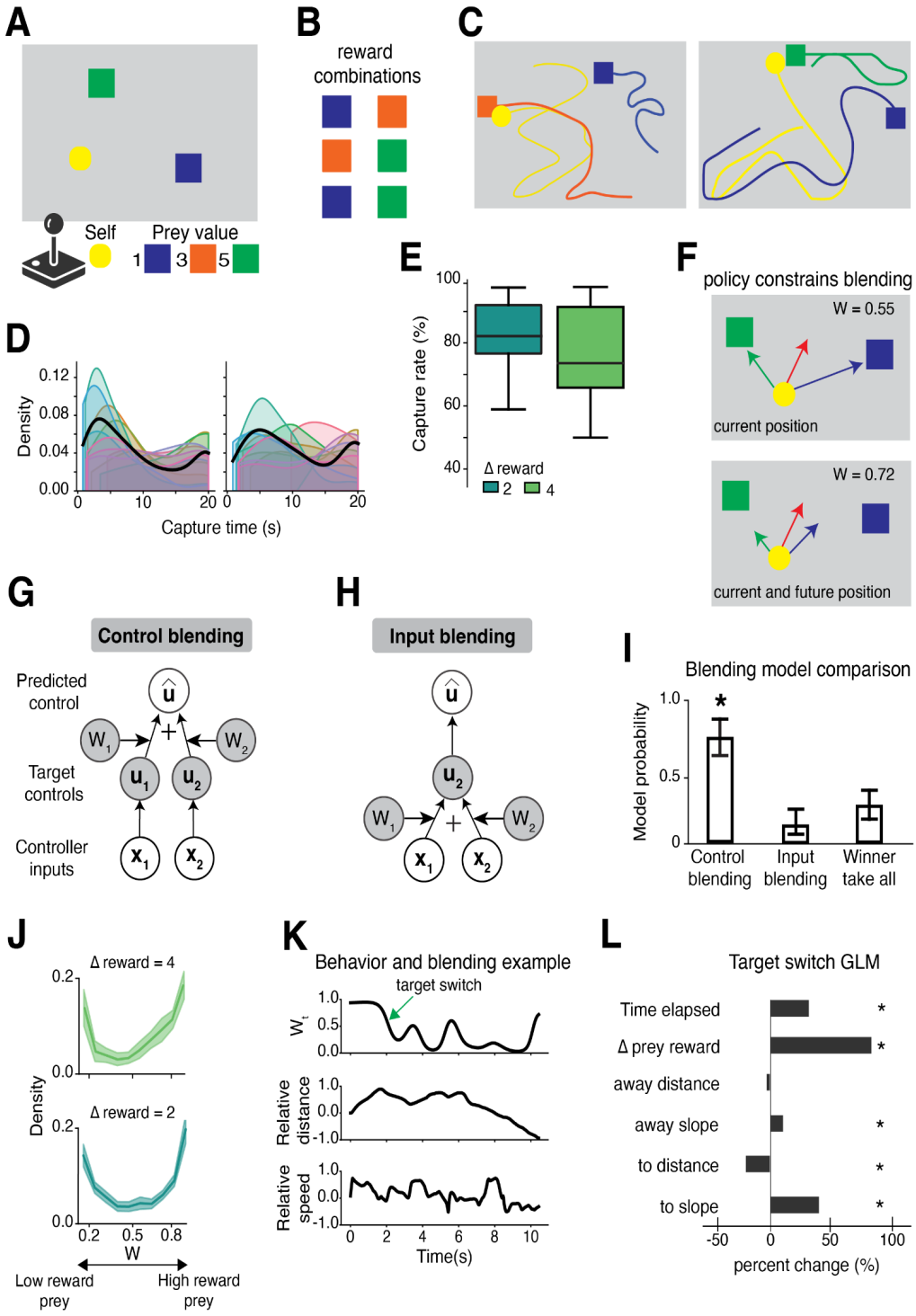

Figure 1. | Continuous pursuit task and control choice model. (**A**) Schematic of pursuit taks design. Prey were displayed as colored squares that denote their reward if caught and maximum speed. Participants' avatar was displayed as a yellow circle, with avatar velocity controlled by the joystick position. (**B**). Each row represents a different trial type where the prey reward differed (three in total). Participants also played trials in which prey of the same reward level were presented across all three unique prey types. (**C**). The left and right column show example movement traces from two different trial types performed by a participant. (**D**). Kernel smoothing density (per participant) of the time it took to capture a prey in trials with a small prey reward difference (left column) and large reward difference (right column). (**E**). Boxplot showing the capture rate median was higher for small prey reward difference (difference = 2) compared to large reward differences (difference = 4). This results from the larger difference including faster prey. (**F**). The top and bottom cartoons are designed to illustrate how the control blending variable W_t for a single time point is dependent on the type of low level control policies being deployed. The Blue and green arrows show theoretical control signals for the participant directed towards each potential prey. The top and bottom plots are examples of if the participant only used a current position control strategy or a current and future position control strategy, respectively. What is conveyed is that, even though both scenarios could have the same measured control signal (red arrow), changes to the underlying policy type are interrelated to the level of W_t, necessitating estimation of both. (**G**). Graphical model describing control blending model. Shaded circles are latent parameters (W) or variables (u_1,u_2 = target specific controls). Transparent circles are measured variables, including inputs (x_1,x_2) and participant control signals (u^). (**H**). Graphical model describing the input blending model. (**I**). Posterior bayesian model probability of models fit to subject data for control blending, input blending, and winner-take all models. The control blending was a significantly better fit across participants ($p < 0.001$ denoted by *). (**J**). Kernel smoothing density showing the distribution of controller blending (W_t) for small (bottom) and large (top) prey reward difference. The key explains that as W_t is greater than 0.5, they are putting more controls to pursuing the larger reward target and vice versa. (**K**). An exemplar set of time series from a single trial, showing the independence of fit W_t (top plot), relative prey distance to participant (middle plot), and relative prey to participant speed (bottom plot). The green arrow in the top plot of W_t shows how we found target switches as points where W_t crossed 0.5. (**L**). Coefficients from hierarchical GLM predicting % change in odds of switching prey (significant predictors denoted by * with posterior probability > 0.95).

Controller blending is differentially encoded in ACC and hippocampus neurons

To examine the neural basis of continuous choice and W_t, we recorded neurons in the hippocampus (n=314 neurons; **Figure 2A**) and anterior cingulate cortex (ACC, n=149 neurons; **Figure 2A**). We characterized neural tuning to W_t by fitting neuron spikes with a Poisson generalized additive model (**Methods; Figure 2B**) that uses a spline basis to estimate the tuning curves to W_t (**Figure 2C**), alongside auxiliary task parameters (e.g., relative prey value, relative prey distance, relative prey speed, avatar speed, trial time elapsed). We found neurons were tuned linearly to W_t in both

hippocampus (25.5%; **Figure 2D**) and ACC (15.8%; **Figure 2D**). We also found a substantial portion of neurons exhibiting nonlinear mixing of W_t with relative prey value in hippocampus (31.2%; **Figure 2D-E**) and ACC (46.1%; **Figure 2D-E**). Besides W_t, we also found neurons tuned to all the auxiliary variables (**Figure 2D**; binomial test was $p < 0.01$ for all variables in both regions).

An important question is whether the hippocampus and ACC differ in terms of their population tuning distribution to W_t. To compare how the hippocampus and ACC represent W_t, we used fisher information (see **Methods**) because it quantifies how well a neural tuning distribution can discriminate small changes in W_t. For example, if a population is chiefly concerned with the value of a pursuit strategy, the fisher information will be skewed toward certain W_t values (e.g., pursuing higher-value prey). Conversely, a more similar fisher information around low and high W_t indicates a more balanced representation of pursuing different prey. By comparing fisher information across low and high W_t values, we can pinpoint whether each brain region's tuning to W_t is more directed at coding for pursuit value (skewed fisher information) versus faithfully coding for W_t (similar low and high fisher information).

We first visualized the population tuning to W_t using a non-negative matrix factorization for dimensionality reduction (**Figure 2F**). Inspection of the hippocampus components suggests a similar population tuning to low ($W_t < 0.15$) and high W_t ($W_t > 0.85$; **Figure 2F**), while ACC components point to a coding skewed towards high value prey ($W_t > 0.85$; **Figure 2F**). We next quantify this by examining fisher information in separate permutations test for each of the low and high W_t levels, and then comparing the levels for differences and equivalences. Confirming visual intuition (**Figure 2F**), there was significant fisher information in the hippocampus for W_t (**Figure 2E**) linked to low (permutation test, $p < 0.005$) and high value prey (permutation test, $p < 0.005$), but the information for these two W_t levels was similar (equivalence test, both $p < 0.05$). Considering ACC (**Figure 2E**), we found evidence for W_t information at high prey levels (permutation test, $W_t > 0.85$, $p < 0.001$), and significant difference between these low and high prey reward levels (permutation test, $p < 0.005$). The hippocampal distribution of coding W_t similarly for low and high value prey is consistent with the idea that it

implements a policy blending code. In contrast, ACC's skewed representation towards high value prey suggests it may be involved in regulating the value of a pursuit blending strategy.

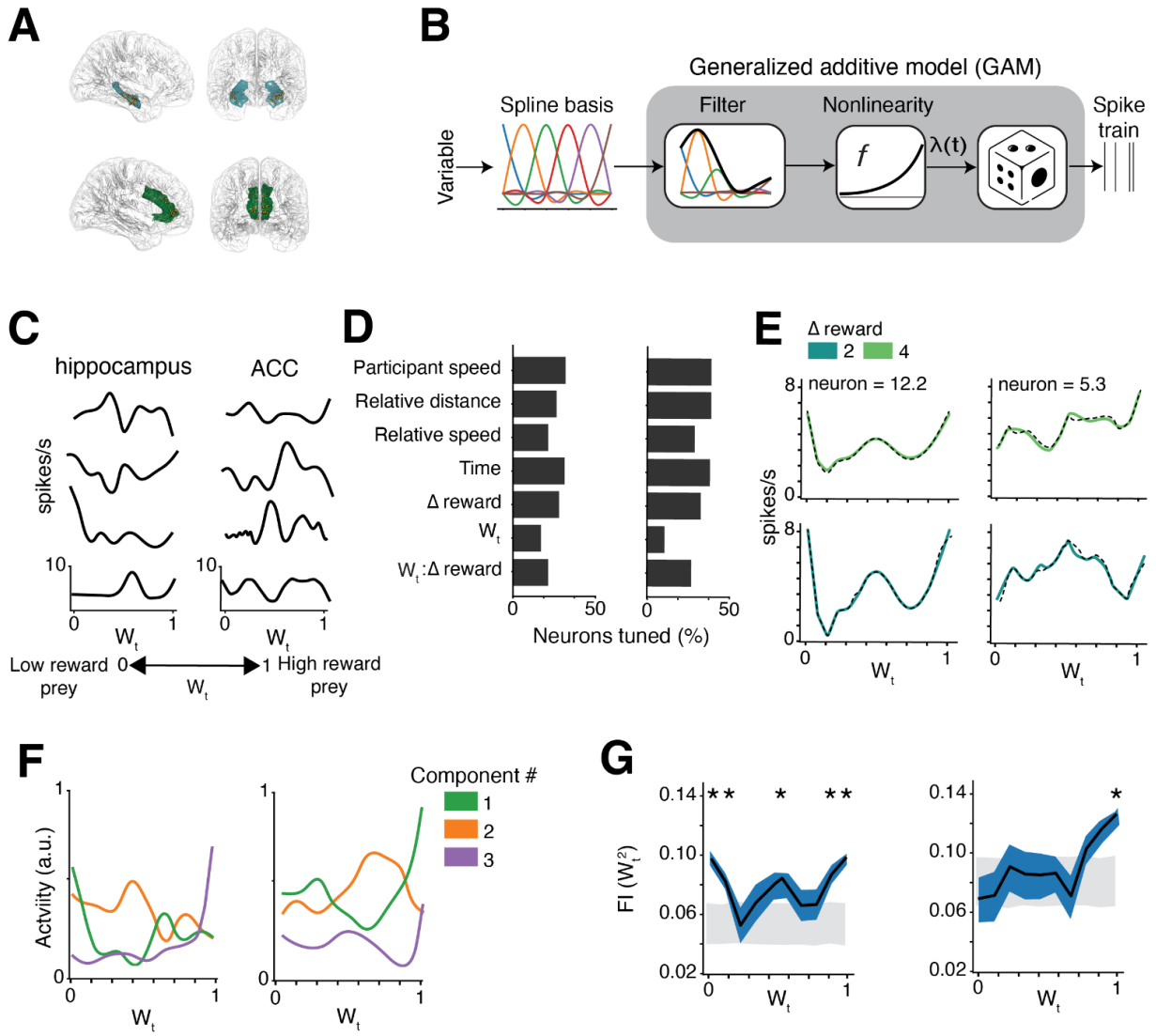

Figure 2. | Neurons in the hippocampus and ACC are differentially tuned to W_t. (**A**). Recording sites for the Hippocampus (top) and ACC (bottom) for 13 patients. (**B**). Diagram of Poisson generalized additive model (GAM) used to model tuning curves with spline bases. (**C**). Exemplar tuning curves to W_t for both hippocampus (left) and ACC (right). (**D**). The percent of neurons tuned to each variable for hippocampus (left) and ACC (right), estimated with the GAM. (**E**). Example neuron tuning curve and fit tuning from GAM (dashed line) that exhibited nonlinear mixed selectivity between W_t and Δ reward, for both hippocampus (left) and ACC (right). (**F**). Non-negative matrix factorization components showing low-dimensional tuning bases to W_t for hippocampus (left) and ACC (right). The units are normalized spikes/s. (**G**). Cross-validated Fisher information with mean (black line) and standard error (blue shading).

Permutation distribution with 95% confidence intervals are the gray shading. Significant information is shown per bin with an asterisk.

Hippocampal dynamics track controller blending

If a brain region encodes the controller W_t then neural population dynamics should exhibit a close parallel to behaviorally derived W_t dynamics (Khona and Fiete, 2022). In addition, if the neural coding of W_t is used to compose controllers, the neural population should convey a linear readout of W_t that is at least partially independent of different reward prey combinations. To assess whether neural dynamics encode the W_t dynamics, we aligned neural activity to target switches because the W_t trajectory during target switches follows a canonical sigmoidal-like pattern (**Figure 3A**). This sigmoidal W_t provides a template to assess whether neural population dynamics track W_t. To quantify the similarity in neural and W_t dynamics, we decomposed the neural populations aligned to target switches using demixed principal components analysis (dPCA; Kobak et al., 2016), and did so separately for ACC and hippocampus. The decomposition was designed to separate dynamics from two types of switches, switches from the low to the high-value target and vice versa (see **Methods**). We visualized the dynamics by projecting the population activity for each area onto the top dPCA component related to the controller W_t and projected held-out test trials on the top component to quantify linearly decodable information about W_t (see **Methods**).

We found the hippocampus had significant linearly decodable information about W_t (permutation test; $p < 0.01$) during the target switch. The top dPCA component for W_t explained 16.2% of the variance of the hippocampus population activity. Visual inspection of the hippocampal projection showed its neural dynamics closely mirrored the sigmoidal W_t trajectory predicted by behavior (**Figure 3C**). We also found a significant interaction of reward difference and W_t that accounted for 4.89% of population variance (**Figure 3B)**. This suggests hippocampus populations directly track how controllers are mixed with a generalizable (linear) code.

We also found linearly decodable W_t related information in ACC during target switches (permutation test; $p < 0.01$; **Figure 3D**). The top dPCA component for W_t explained 10% of the variance of the ACC population activity (**Figure 3B**). Visually

examining the ACC dynamics indicates W_t information only became available 150 ms after the switch onset, then hit a peak again close to the target switch offset (**Figure 3D**). These results are more consistent with ACC potentially driving the switch decision itself and monitoring the decision progress, rather than directly tracking W_t. Together, the combined hippocampus and ACC results indicate they engage in distinct processes that relate to shifting weighting strategies and versus implementing controller weighting, respectively.

The dPCA also indicated substantial condition-independent modes (CIM) in both brain areas. The CIMs explained 21% and 24% (**Figure 3B**) variance of population activity in hippocampus and ACC dynamics, respectively. Visually examining the CIMs in both hippocampus (**Figure 3E**) and ACC (**Figure 3F**) reveal dynamics that monotonically increase during the time-course of the target switch. While the origins of these CIM dynamics are unclear, their profiles indicate distinct profiles for tracking the progression of switch decisions between hippocampus and ACC.

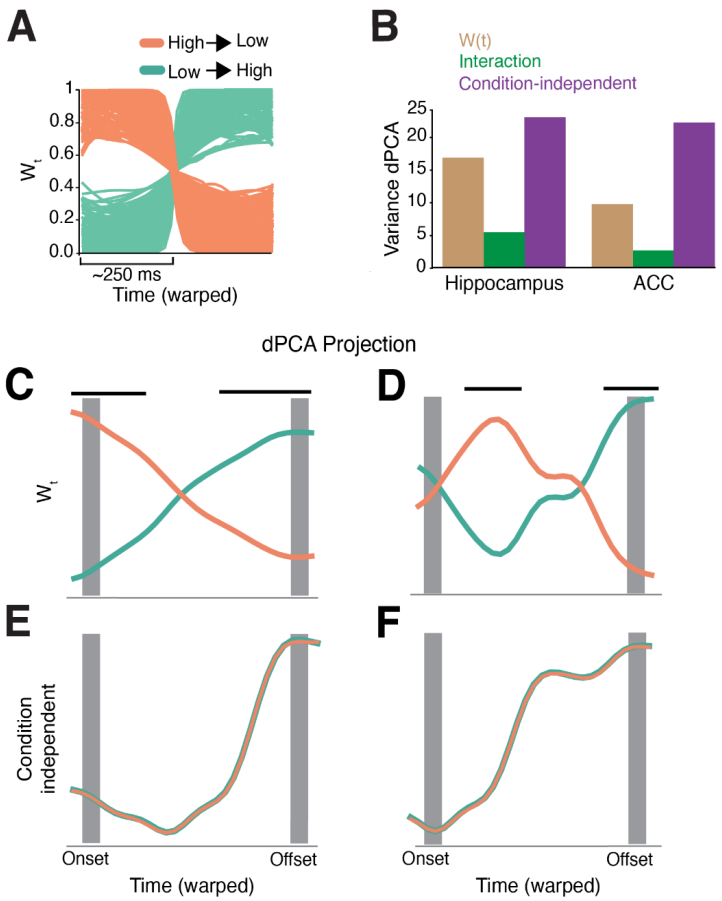

Figure 3.| Low-dimensional controller blending dynamics during goal switch. (A). Controller blending dynamics extracted during target switches (**Methods**) for a single participant. Orange traces are switches from a high to low value prey, and green traces are switches from low to high value prey. (**B).** dPCA percent variance of each task factor for each area. (**C**) Hippocampus projection onto top dPCA component for blending weight factor. Gray shaded bars mark the median timing of the onset and offset of the target switch. Black lines show time points of significant decoding. (**D).** ACC projection onto top dPCA component for blending weight factor. Same format as (**C**) (**E).** Hippocampus projection onto top dPCA component for condition independent factor. (**F).** ACC projection onto top dPCA component for the condition independent factor.

Neural ramping predicts goal switching

Previous studies examining discrete decisions indicate ACC population ramping signals are predictive of the onset of decisions (Balewski et al., 2023; Sarafyazd and Jazayeri, 2019). Building on these findings, we hypothesized that ACC may also play a role in driving target switch decisions exemplified by ramping signals coinciding with the target switch onset (**Figure 4A**), while no such ramping should appear during time

windows without switches. To compute the ramping, we adopted an approach applied previously in perceptual decision making that uses a linear support vector machine (SVM; Steinemann et al., 2024). We built separate SVMs for hippocampus and ACC activity time-locked to the onset of target switch decisions. The SVMs were trained to predict whether a target switch will happen within the next 100 ms (**Methods**), with separate SVMs for switches going low to high value targets and vice versa.

Examining time-resolved decoding in the hippocampus indicated target switches could be decoded in both switch directions (permutation test $p < 0.05$; **Figure 4B**). Importantly, however, this information only became significantly decodable after the onset of the target switch (>200 ms after switch onset; **Figure 4B**). Target switch decoding was also significant in the ACC (permutation test $p < 0.05$; **Figure 4C**). We found decoding accuracy in ACC starts to ramp up around 100 ms before the switch onset and peaks around the switch onset, providing support for the hypothesis that ACC ramping predicts target switches (**Figure 4C**). However, these target switches were only decodable in ACC when going from a high to low value target. The differences in timing of decoding between areas were confirmed by examining the time lag of peak decoding between areas (**Methods**). We found they were different using a permutation test ($p < 0.05$), with ACC peak ramp decoding occurring approximately 195 ms (± 27 ms) earlier than hippocampus.

Previous studies of the neural decision-making processes often find ramping profiles at the single neuron level (e.g., Gold and Shadlen, 2007; Kiani et al., 2014; Luo et al., 2024). Therefore, we asked whether the population ramping profiles we found above were also evident in clusters of single neurons. We used a method that combines k-means clustering with linear dynamic systems, which allows us to cluster based on the dynamics (**Methods**). The number of clusters were optimized (silhouette score) separately for hippocampus and ACC, and only clusters exhibiting significant selectivity for whether a target switch occurred were retained using a permutation test.

This process yielded two clusters for the hippocampus (both permutation tests, $p < 0.01$; **Figure 4D-E**). Visually examining their temporal properties, we find that cluster one (C1) recapitulates what we found in population decoding (**Figure 4D**). Specifically,

this cluster of neurons reflects selectivity to target switches after the onset rather than ramping up to the decision point (time = 0). Examining cluster two (C2) suggests it closely tracked controller W_t rather than ramping, as well (**Figure 4D**). Examining ACC clusters (all permutation tests, $p < 0.01$; **Figure 4F-G**), we found cluster one (C1) neurons exhibit a ramp up and down time-locked with the target switch onset (**Figure 4F**). The other two clusters (C2 and C3; **Figure 4F**) exhibited selectivity to target switches but neither exhibited the same ramping profile as cluster one. Together, these results indicate that ACC exhibits a ramping that signals target switch decisions, implicating it in driving decisions to change targets and reweight control policies. The specificity of the ACC ramping to different switch conditions further indicates this decision process is sensitive to the value of the target switch, rather than being a generic decision signal.

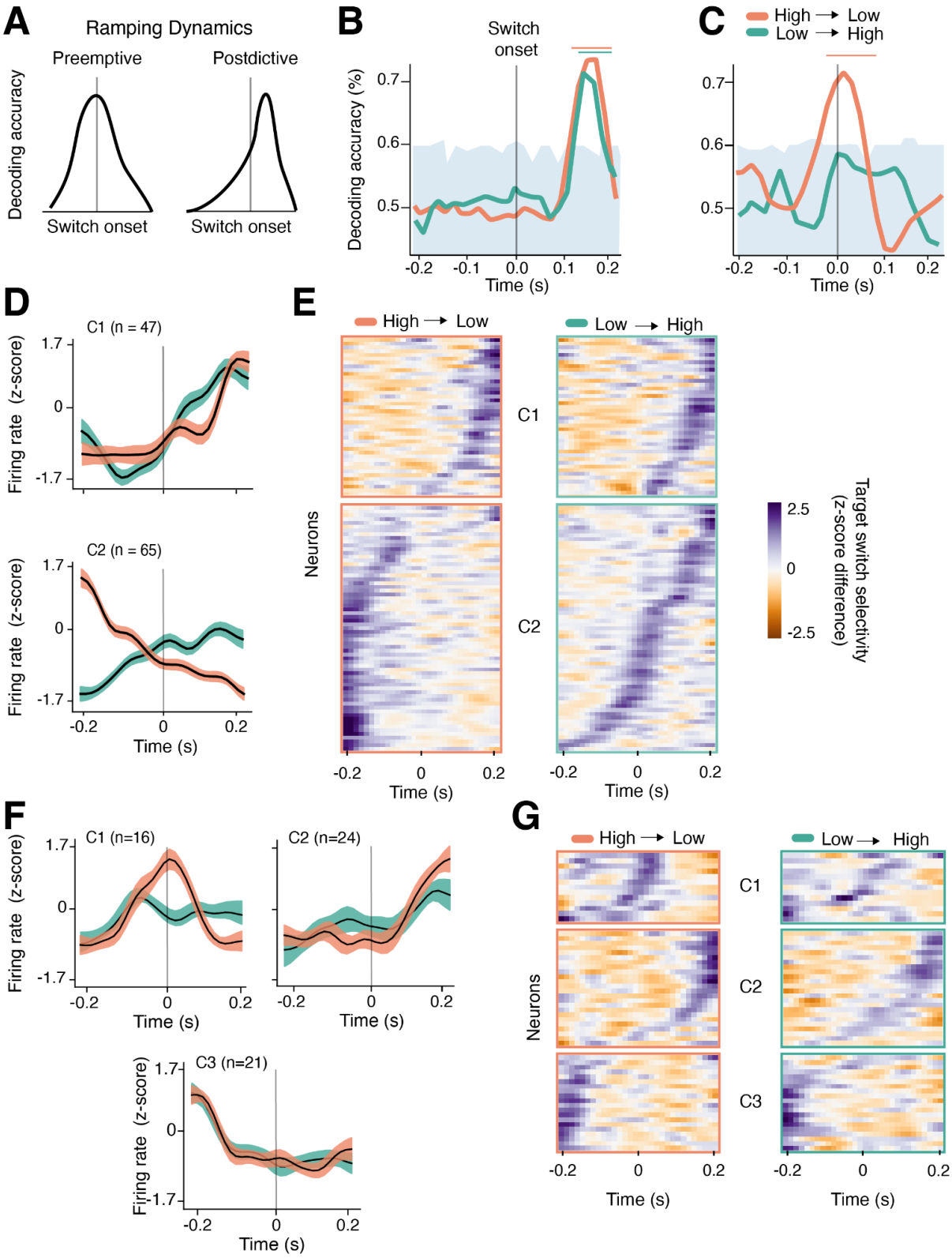
A
Ramping Dynamics
Preemptive
Postdictive
Decoding accuracy
Switch onset
Switch onset
B
Switch onset
Decoding accuracy (%)
0.7
0.6
0.5
-0.2
-0.1
0.0
0.1
0.2
Time (s)
C
High → Low
Low → High
0.7
0.6
0.5
-0.2
-0.1
0.0
0.1
0.2
Time (s)
D
C1 (n = 47)
Firing rate (z-score)
1.7
0
-1.7
C2 (n = 65)
Firing rate (z-score)
1.7
0
-1.7
-0.2
0
0.2
Time (s)
E
High → Low
Low → High
C1
C2
Neurons
-0.2
0
0.2
Time (s)
-0.2
0
0.2
Time (s)
2.5
0
-2.5
Target switch selectivity
(z-score difference)
F
C1 (n=16)
Firing rate (z-score)
1.7
0
-1.7
-0.2
0
0.2
Time (s)
C2 (n=24)
-0.2
0
0.2
Time (s)
C3 (n=21)
Firing rate (z-score)
1.7
0
-1.7
-0.2
0
0.2
Time (s)
G
High → Low
Low → High
C1
C2
C3
Neurons
-0.2
0
0.2
Time (s)
-0.2
0
0.2
Time (s)

Figure 4. | Ramp decoding subspaces signal changes of mind. (A). Graphical depiction of our predicted ramping dynamics expected from choice accumulation, which should preempt the choice and peak around switch onset (left), versus ramping that is postdictive and describes target switches after completion. (**B**). decoding accuracy for hippocampus time locked to switch onsets, with bars overlaid denoting significant decoding time-points ($p < 0.05$) with a permutation test (1000 permutes), and shaded error bar showing the 5th to 95th percentile of the null distribution. Different colored lines represent decoding accuracy for the different switch conditions. (**C**)**.** decoding accuracy for Hippocampus, with the same format as (**C**). (**D**). Mean cluster sorted firing rate acquired from hippocampus time locked to switch onset. (**E**). Hippocampus z-scored cluster selectivity to target versus no-target switch trials, separated for high to low value (left column) and low to high value target switches (right column). (**F).** Mean cluster sorted firing rate acquired from ACC time locked to switch onset. (**G).** ACC z-scored cluster selectivity to target versus no-target switch trials, separated for high to low value (left column) and low to high value target switches (right column).

DISCUSSION

Decision-making in continuous contexts poses multiple interrelated control problems (Merel et al., 2019; Yoo et al., 2021; Burge and Bonnen, 2025). We must combine dynamic feedback and predictions of future target states to decide which goals to pursue and how to pursue them. Here, we identified the algorithmic and neural basis of these processes by examining human behavior in a multi-prey pursuit task. We developed a control-theoretic model that we used to identify goal pursuit decisions that underlie moment-to-moment actions. We find that control is parsimoniously explained as continuously composing low-level, goal-specific control policies. A key benefit to using compositional control is that it provides an efficient neural mechanism for generalizing decision strategies across novel environments (Driscoll et al., 2024; Johnston et al., 2024; Sandbrink et al, 2024); the challenging problem of learning specific controllers for each combination of rewards thereby becomes simplified to a compositional solution. For example, when novel combinations of familiar prey are encountered during pursuit, there is no need to learn a new set of controllers, as existing ones can be readily combined. More strikingly, this compositional approach supporting pursuit parallels modern solutions for training AI agents and robots to learn generalizable complex and skilled behavior (Haarnoja et al., 2018; Matsuo et al., 2022).

Our modeling approach opens the door to understanding a wide range of naturalistic behaviors not well captured by standard approaches (Pearson et al., 2014; Yoo et al., 2021) such as cooperative hunting or any situation in which continuous behavior involves the choice between multiple, competing goals. Previous studies have used unsupervised and clustering methods to identify behavioral motifs in naturalistic behavior (Anderson and Perona, 2014; Calhoun et al., 2019; Voloh et al., 2023; Weinreb et al., 2024; Wiltschko et al., 2015). While these studies have been useful for behavioral parcellation, their focus has been on overt and discrete behavior states (e.g., turning). Nonetheless, these approaches remain largely agnostic to the neural implementation of an agent's internal cognitive states, such as the hierarchical choice to switch between states and which mixture of goals they are pursuing. While unsupervised methods may aid in finding clusters of overt behavior, we find that

uncovering the moment-by-moment subject choice requires positing task- and theory-driven constraints on how the subjects might solve the problem (Kwon et al., 2020). Such a finding indicates that understanding ethological, high-dimensional behavior must go beyond simply categorizing discrete behavioral motifs.

Using our modeling, we found neural results that align with and extend the theory that the hippocampus implements a cognitive map (Behrens et al., 2018; O'keefe and Nadel, 1979; Behrens). The cognitive map theory posits that the hippocampus creates structured representations of the world and does so by composing and binding individual basis elements (Kurth-Nelson et al., 2023; Whittington et al., 2022). Previous studies have focused on hippocampal compositions over visual scene construction (Schwartenbeck et al., 2023), spatial information (Hoydal et al., 2019), or novel concept compounds (Barron et al., 2013). We extend these findings by showing the hippocampus also encodes compositions that bind goal-specific control policies. Specifically, we found that low-dimensional (dPCA) hippocampal dynamics follow a general and compositional code that can be used to combine different prey-specific controllers. Thus, these results point to a role of the hippocampus in forming both maps of the world and goal-directed actions (Zuthsi et al., 2025).

Based on the known involvement of the ACC in cognitive flexibility and decision-making, we reasoned here that it serves the crucial role of driving switching of goals and policy reweighting during continuous choice. In support of this, we found ACC population ramping activity predicted an impending target switch. This pattern aligns with a large literature demonstrating that ACC accumulates evidence for strategic shifts in decision-making or rule application (Hayden et al., 2011; Johnston et al., 2007; Kennerly et al., 2006; Kolling et al., 2012; Sarafyazd and Jazayeri, 2019; Shenhav et al., 2016). However, our results also show this ramping switch signal is dependent on the value of switching goals, a finding in line with previous studies and theories linking ACC to integrating cost-benefit information into decisions (Amiez et al., 2007; Friedman et al., 2015; Shenhav et al., 2016). A possible explanation of our result is that abandoning a high value versus low value prey represents decisions with different values and costs, thus eliciting different ramping signals. The implication is ACC is not merely signaling

movement transitions but is integrating decision variables. This integration likely involves weighing the cost and benefits of switching targets, aligning with foraging models (Stephens & Krebs, 1986; Hayden et al., 2011; Kane et al., 2022). Such cost-benefit decisions could also include computing the expected value of persistence and control for a current goal (Blanchard et al., 2015; Kolling et al., 2012 ; Shenhav et al., 2016). Our work extends this role of ACC as a value driven meta-controller to the realm of continuous choice.

More broadly, our results are consistent with a network where hippocampus and ACC play complementary roles in ethological decisions like prey pursuit and foraging. ACC has been a key consideration in previous studies of foraging because it integrates information from regions coding for reward and environment information and is associated with choosing alternative options (Kolling et al., 2016; Tervo et al., 2014). This indicates ACC is concerned with representing decision control signals that are related to the value of choices. Consistent with this, we found ACC neural tuning to policy blending was skewed towards high value prey. Such a higher-order control signal is useful for driving changes in low-level control blending but may be distinct from the source implementation of online goal blending (Fine et al., 2024). Our results point to the hippocampus as a node responsible for directly implementing the continuous blending. In this complementary network, ACC signals changes to controls while hippocampus represents the control blending in an unbiased manner.

REFERENCES

Akam, T., Rodrigues-Vaz, I., Marcelo, I., Zhang, X., Pereira, M., Oliveira, R. F., Dayan, P., & Costa, R. M. (2021). The anterior cingulate cortex predicts future states to mediate model-based action selection. *Neuron*, *109*(1), 149-163.e7.

Alexander, W. H., & Brown, J. W. (2011). Medial prefrontal cortex as an action-outcome predictor. *Nat. Neurosci.*, *14*(10), 1338–1344.

Amiez, C., Joseph, J. P., & Procyk, E. (2006). Reward encoding in the monkey anterior cingulate cortex. *Cereb. Cortex*, *16*(7), 1040–1055.

Anderson, D. J., & Perona, P. (2014). Toward a science of computational ethology. *Neuron*, *84*(1), 18–31.

Bakermans, J. J. W., Warren, J., Whittington, J. C. R., & Behrens, T. E. J. (2025). Constructing future behavior in the hippocampal formation through composition and replay. *Nat. Neurosci.*, 1–12.

Balewski, Z. Z., Elston, T. W., Knudsen, E. B., & Wallis, J. D. (2023). Value dynamics affect choice preparation during decision-making. *Nat. Neurosci.*, *26*(9), 1575–1583.

Balzani, E., Lakshminarasimhan, K. J., Angelaki, D., & Savin, C. (2020). Efficient estimation of neural tuning during naturalistic behavior. *Neural Inf Process Syst*, *33*, 12604–12614.

Barron, H. C., Dolan, R. J., & Behrens, T. E. J. (2013). Online evaluation of novel choices by simultaneous representation of multiple memories. *Nat. Neurosci.*, *16*(10), 1492–1498.

Behrens, T. E. J., Muller, T. H., Whittington, J. C. R., Mark, S., Baram, A. B., Stachenfeld, K. L., & Kurth-Nelson, Z. (2018). What is a cognitive map? Organizing knowledge for flexible behavior. *Neuron*, *100*(2), 490–509.

Bertsekas, D. P. (1996). *Neuro-Dynamic Programming*. Athena Scientific.

Blanchard, T. C., Strait, C. E., & Hayden, B. Y. (2015). Ramping ensemble activity in dorsal anterior cingulate neurons during persistent commitment to a decision. *J. Neurophysiol.*, *114*(4), 2439–2449.

Burge, J., & Bonnen, K. (2025). Continuous psychophysics: Past, present, future. *Trends Cogn. Sci.*

Calhoun, A. J., Pillow, J. W., & Murthy, M. (2019). Unsupervised identification of the internal states that shape natural behavior. *Nat. Neurosci.*, *22*(12), 2040–2049.

Cervera, R. L., Wang, M. Z., & Hayden, B. Y. (2020). Systems neuroscience of curiosity. Current Opinion in Behavioral Sciences, 35, 48-55.

Chadwick, M. J., Jolly, A. E. J., Amos, D. P., Hassabis, D., & Spiers, H. J. (2015). A goal direction signal in the human entorhinal/subicular region. *Curr. Biol.*, *25*(1), 87–92.

Chericoni, A., Fine, J. M., Chavez, A. G., Franch, M., Mickiewicz, E., Mathura, R., ... & Hayden, B. Y. (2025). Independent Continuous Tracking of Multiple Agents in the Human Hippocampus. bioRxiv, 2025-03.

Cisek, P. (2012). Making decisions through a distributed consensus. *Curr. Opin. Neurobiol.*, *22*(6), 927–936.

Driscoll, L. N., Shenoy, K., & Sussillo, D. (2024). Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. *Nat. Neurosci.*, *27*(7), 1349–1363.

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

Dvijotham, K., & Todorov, E. (2013). Linearly Solvable Optimal Control. In *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control* (pp. 119–141). John Wiley & Sons, Inc.

Ebitz, R. B., Sleezer, B. J., Jedema, H. P., Bradberry, C. W., & Hayden, B. Y. (2019). Tonic exploration governs both flexibility and lapses. PLoS computational biology, 15(11), e1007475.

Eisenreich, B. R., Hayden, B. Y., & Zimmermann, J. (2019). Macaques are risk-averse in a freely moving foraging task. Scientific reports, 9(1), 15091.

Eppinger, B., Goschke, T., & Musslick, S. (2021). Meta-control: From psychology to computational neuroscience. *Cogn. Affect. Behav. Neurosci.*, *21*(3), 447–452.

Fabian, S. T., Sumner, M. E., Wardill, T. J., Rossoni, S., & Gonzalez-Bellido, P. T. (2018). Interception by two predatory fly species is explained by a proportional navigation feedback controller. *J. R. Soc. Interface*, *15*(147), 20180466.

Fine, J. M., Yoo, S.-B. M., & Hayden, B. Y. (2024). Control over a mixture of policies determines change of mind topology during continuous choice. *bioRxivorg*.

Friedman, A., Homma, D., Gibb, L. G., Amemori, K.-I., Rubin, S. J., Hood, A. S., Riad, M. H., & Graybiel, A. M. (2015). A corticostriatal path targeting striosomes controls decision-making under conflict. *Cell*, *161*(6), 1320–1333.

Gallivan, J. P., Logan, L., Wolpert, D. M., & Flanagan, J. R. (2016). Parallel specification of competing sensorimotor control policies for alternative action options. *Nat. Neurosci.*, *19*(2), 320–326.

Garvert, M. M., Saanum, T., Schulz, E., Schuck, N. W., & Doeller, C. F. (2023). Hippocampal spatio-predictive cognitive maps adaptively guide reward generalization. *Nat. Neurosci.*

Gold, J. I., & Shadlen, M. N. (2007). The neural basis of decision making. *Annu. Rev. Neurosci.*, *30*(1), 535–574.

Gómez, V., Kappen, H. J., Peters, J., & Neumann, G. (2014). Policy search for path integral control. In *Machine Learning and Knowledge Discovery in Databases* (pp. 482–497). Springer Berlin Heidelberg.

Gomi, H., & Kawato, M. (1993). Recognition of manipulated objects by motor learning with modular architecture networks. *Neural Netw.*, *6*(4), 485–497.

Gordon, J., Maselli, A., Lancia, G. L., Thiery, T., Cisek, P., & Pezzulo, G. (2021). The road towards understanding embodied decisions. *Neurosci. Biobehav. Rev.*, *131*, 722–736.

Haarnoja, T., Moran, B., Lever, G., Huang, S. H., Tirumala, D., Humplik, J., Wulfmeier, M., Tunyasuvunakool, S., Siegel, N. Y., Hafner, R., Bloesch, M., Hartikainen, K., Byravan, A., Hasenclever, L., Tassa, Y., Sadeghi, F., Batchelor, N., Casarini, F., Saliceti, S., … Heess, N. (2024). Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Sci. Robot.*, *9*(89), eadi8022.

Haarnoja, T., Pong, V., Zhou, A., Dalal, M., Abbeel, P., & Levine, S. (2018, May). Composable deep reinforcement learning for robotic manipulation. *2018 IEEE International Conference on Robotics and Automation (ICRA)*.

Haith, A. M., Huberdeau, D. M., & Krakauer, J. W. (2015). Hedging your bets: Intermediate movements as optimal behavior in the context of an incomplete decision. *PLoS Comput. Biol.*, *11*(3), e1004171.

Hayden, B. Y., Pearson, J. M., & Platt, M. L. (2011). Neuronal basis of sequential foraging decisions in a patchy environment. *Nat. Neurosci.*, *14*(7), 933–939.

Hayden, B. Y. (2019). Why has evolution not selected for perfect self-control?. Philosophical Transactions of the Royal Society B, 374(1766), 20180139.

Howard, L. R., Javadi, A. H., Yu, Y., Mill, R. D., Morrison, L. C., Knight, R., Loftus, M. M., Staskute, L., & Spiers, H. J. (2014). The hippocampus and entorhinal cortex encode the path and Euclidean distances to goals during navigation. *Curr. Biol.*, *24*(12), 1331–1340.

Høydal, Ø. A., Skytøen, E. R., Andersson, S. O., Moser, M.-B., & Moser, E. I. (2019). Object-vector coding in the medial entorhinal cortex. *Nature*, *568*(7752), 400–404.

Jercog, P. E., Ahmadian, Y., Woodruff, C., Deb-Sen, R., Abbott, L. F., & Kandel, E. R. (2019). Heading direction with respect to a reference point modulates place-cell activity. *Nat. Commun.*, *10*(1), 2333.

Johnston, K., Levin, H. M., Koval, M. J., & Everling, S. (2007). Top-down control-signal dynamics in anterior cingulate and prefrontal cortex neurons following task switching. *Neuron*, *53*(3), 453–462.

Johnston, W. J., Fine, J. M., Yoo, S. B. M., Ebitz, R. B., & Hayden, B. Y. (2024). Semi-orthogonal subspaces for value mediate a binding and generalization trade-off. *Nat. Neurosci.*, *27*(11), 2218–2230.

Kane, G. A., James, M. H., Shenhav, A., Daw, N. D., Cohen, J. D., & Aston-Jones, G. (2022). Rat anterior cingulate cortex continuously signals decision variables in a patch foraging task. *J. Neurosci.*, *42*(29), 5730–5744.

Kennerley, S. W., Walton, M. E., Behrens, T. E. J., Buckley, M. J., & Rushworth, M. F. S. (2006). Optimal decision making and the anterior cingulate cortex. *Nat. Neurosci.*, *9*(7), 940–947.

Khona, M., & Fiete, I. R. (2022). Attractor and integrator networks in the brain. *Nat. Rev. Neurosci.*, *23*(12), 744–766.

Kiani, R., Cueva, C. J., Reppas, J. B., & Newsome, W. T. (2014). Dynamics of neural population responses in prefrontal cortex indicate changes of mind on single trials. *Curr. Biol.*, *24*(13), 1542–1547.

Kobak, D., Brendel, W., Constantinidis, C., Feierstein, C. E., Kepecs, A., Mainen, Z. F., Qi, X.-L., Romo, R., Uchida, N., & Machens, C. K. (2016). Demixed principal component analysis of neural population data. *Elife*, *5*, e10989.

Kolling, N., Behrens, T. E. J., Mars, R. B., & Rushworth, M. F. S. (2012). Neural mechanisms of foraging. *Science*, *336*(6077), 95–98.

Kolling, N., Behrens, T., Wittmann, M. K., & Rushworth, M. (2016). Multiple signals in anterior cingulate cortex. *Curr. Opin. Neurobiol.*, *37*, 36–43.

Kriegeskorte, N., & Wei, X.-X. (2021). Neural tuning and representational geometry. *Nat. Rev. Neurosci.*, *22*(11), 703–718.

Kurth-Nelson, Z., Behrens, T., Wayne, G., Miller, K., Luettgau, L., Dolan, R., Liu, Y., & Schwartenbeck, P. (2023). Replay and compositional computation. *Neuron*, *111*(4), 454–469.

Kwon, M., Daptardar, S., Schrater, P., & Pitkow, X. (2020). Inverse Rational Control with Partially Observable Continuous Nonlinear Dynamics. *Adv. Neural Inf. Process. Syst.*, *33*, 7898–7909.

Lake, B., & Baroni, M. (2017). Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. *ICML*, *80*, 2879–2888.

Luo, T. Z., Kim, T. D., Gupta, D., Bondy, A. G., Kopec, C. D., Elliot, V. A., DePasquale, B., & Brody, C. D. (2023). Transitions in dynamical regime and neural mode underlie perceptual decision-making. *bioRxivorg*.

Maselli, A., Gordon, J., Eluchans, M., Lancia, G. L., Thiery, T., Moretti, R., Cisek, P., & Pezzulo, G. (2023). Beyond simple laboratory studies: Developing sophisticated models to study rich behavior. *Phys. Life Rev.*, *46*, 220–244.

Matsuo, Y., LeCun, Y., Sahani, M., Precup, D., Silver, D., Sugiyama, M., Uchibe, E., & Morimoto, J. (2022). Deep learning, reinforcement learning, and world models. *Neural Netw.*, *152*, 267–275.

McNaughton, N., & Bannerman, D. (2024). The homogenous hippocampus: How hippocampal cells process available and potential goals. *Prog. Neurobiol.*, *240*(102653), 102653.

Merel, J., Botvinick, M., & Wayne, G. (2019). Hierarchical motor control in mammals and machines. *Nat. Commun.*, *10*(1), 5489.

Muhle-Karbe, P. S., Sheahan, H., Pezzulo, G., Spiers, H. J., Chien, S., Schuck, N. W., & Summerfield, C. (2023). Goal-seeking compresses neural codes for space in the human hippocampus and orbitofrontal cortex. *Neuron*, *111*(23), 3885-3899.e6.

Murphy, K. P. (2022). *Probabilistic machine learning: An introduction*. MIT Press.

Musslick, S., Cohen, J. D., & Goschke, T. (2024). Meta-control. In *Reference Module in Neuroscience and Biobehavioral Psychology* (pp. 1–17). Elsevier.

Nyberg, N., Duvelle, É., Barry, C., & Spiers, H. J. (2022). Spatial goal coding in the hippocampal formation. *Neuron*, *110*(3), 394–422.

Ormond, J., & O'Keefe, J. (2022). Hippocampal place cells have goal-oriented vector fields during navigation. *Nature*, *607*(7920), 741–746.

Pan, Y., Theodorou, E. A., & Kontitsis, M. (2015). Sample efficient path integral control under uncertainty. *Neural Inf Process Syst*, *abs/1509.01846*.

Passingham, R. E. (1979). *The hippocampus as a cognitive mapl J. O'Keefe & L. Nadel, Oxford University Press, Oxford (1978). 570 pp.,£ 25.00*.

Pearson, J. M., Watson, K. K., & Platt, M. L. (2014). *Decision making: The neuroethological turn*. *82*(5), 950–965.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., & Duchesnay. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, *12*, 2825–2830.

Peng, X. B., Chang, M., Zhang, G., Abbeel, P., & Levine, S. (2019). MCP: Learning composable hierarchical control with multiplicative compositional policies. *Neural Inf Process Syst*, *abs/1905.09808*.

Sandbrink, K., Bauer, J. P., Proca, A. M., Saxe, A. M., Summerfield, C., & Hummos, A. (2024). Flexible task abstractions emerge in linear networks with fast and bounded units. *arXiv [Cs.LG]*.

Sarafyazd, M., & Jazayeri, M. (2019). Hierarchical reasoning by neural circuits in the frontal cortex. *Science*, *364*(6441), eaav8911.

Sarel, A., Finkelstein, A., Las, L., & Ulanovsky, N. (2017). Vectorial representation of spatial goals in the hippocampus of bats. *Science*, *355*(6321), 176–180.

Schwartenbeck, P., Baram, A., Liu, Y., Mark, S., Muller, T., Dolan, R., Botvinick, M., Kurth-Nelson, Z., & Behrens, T. (2023). Generative replay underlies compositional inference in the hippocampal-prefrontal circuit. *Cell*, *186*(22), 4885-4897.e14.

Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with python. *SciPy*, *7*(1), 92–96.

Shenhav, A., Botvinick, M. M., & Cohen, J. D. (2013). The expected value of control: An integrative theory of anterior cingulate cortex function. *Neuron*, *79*(2), 217–240.

Shenhav, A., Cohen, J. D., & Botvinick, M. M. (2016). Dorsal anterior cingulate cortex and the value of control. *Nat. Neurosci.*, *19*(10), 1286–1291.

Sheth, S. A., Mian, M. K., Patel, S. R., Asaad, W. F., Williams, Z. M., Dougherty, D. D., Bush, G., & Eskandar, E. N. (2012). Human dorsal anterior cingulate cortex neurons mediate ongoing behavioural adaptation. *Nature*, *488*(7410), 218–221.

Shima, K., & Tanji, J. (1998). Role for cingulate motor area cells in voluntary movement selection based on reward. *Science*, *282*(5392), 1335–1338.

Sridhar, V. H., Li, L., Gorbonos, D., Nagy, M., Schell, B. R., Sorochkin, T., Gov, N. S., & Couzin, I. D. (2021). The geometry of decision-making in individuals and collectives. *Proc. Natl. Acad. Sci. U. S. A.*, *118*(50).

Steinemann, N., Stine, G. M., Trautmann, E., Zylberberg, A., Wolpert, D. M., & Shadlen, M. N. (2024). Direct observation of the neural computations underlying a single decision. *Elife*, *12*.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. cambridge.org.

Tervo, D. G. R., Proskurin, M., Manakov, M., Kabra, M., Vollmer, A., Branson, K., & Karpova, A. Y. (2014). Behavioral variability through stochastic choice and its gating by anterior cingulate cortex. *Cell*, *159*(1), 21–32.

Theodorou, E., Buchli, J., & Schaal, S. (2010). A generalized path integral control approach to reinforcement learning. *The Journal of Machine Learning Research*, *11*, 3137–3181.

Todorov, E. (2009). Compositionality of optimal control laws. *Neural Inf Process Syst*, *22*, 1856–1864.

Voloh, B., Maisson, D. J.-N., Cervera, R. L., Conover, I., Zambre, M., Hayden, B., & Zimmermann, J. (2023). Hierarchical action encoding in prefrontal cortex of freely moving macaques. *Cell Rep.*, *42*(9), 113091.

Wang, M. Z., & Hayden, B. Y. (2021). Latent learning, cognitive maps, and curiosity. Current Opinion in Behavioral Sciences, 38, 1-7.

Whittington, J. C. R., McCaffary, D., Bakermans, J. J. W., & Behrens, T. E. J. (2022). How to build a cognitive map. *Nat. Neurosci.*, *25*(10), 1257–1272.

Whittington, J. C. R., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., & Behrens, T. E. J. (2020). The Tolman-Eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell*, *183*(5), 1249-1263.e23.

Wikenheiser, A. M., & Redish, A. D. (2015). Hippocampal theta sequences reflect current goals. *Nat. Neurosci.*, *18*(2), 289–294.

Wikenheiser, A. M., & Schoenbaum, G. (2016). Over the river, through the woods: Cognitive maps in the hippocampus and orbitofrontal cortex. *Nat. Rev. Neurosci.*, *17*(8), 513–523.

Wolpert, D. M., & Kawato, M. (1998). Multiple paired forward and inverse models for motor control. *Neural Netw.*, *11*(7–8), 1317–1329.

Wood, S. N. (2017). *Generalized additive models: An introduction with R*. Chapman and Hall/CRC.

Yang, Q., Lin, Z., Zhang, W., Li, J., Chen, X., Zhang, J., & Yang, T. (2022). Monkey plays Pac-Man with compositional strategies and hierarchical decision-making. *Elife*, *11*.

Yeung, N., & Summerfield, C. (2012). Metacognition in human decision-making: Confidence and error monitoring. *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, *367*(1594), 1310–1321.

Yoo, S. B. M., Hayden, B. Y., & Pearson, J. M. (2021). Continuous decisions. *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, *376*(1819), 20190664.

Yoo, S. B. M., Tu, J. C., Piantadosi, S. T., & Hayden, B. Y. (2020). The neural basis of predictive pursuit. *Nat. Neurosci.*, *23*(2), 252–259.

Zutshi, I., Apostolelli, A., Yang, W., Zheng, Z. S., Dohi, T., Balzani, E., Williams, A. H., Savin, C., & Buzsáki, G. (2025). Hippocampal neuronal activity is aligned with action plans. *Nature*.

Data availability statement:
All data will be made available upon publication

Code availability statement:
All code will be made available at https://github.com/JustFineNeuro/PacManMain

METHODS

All procedures were approved by the Baylor College of Medicine Institutional review board.

Human intracranial neurophysiology. Experimental data were recorded from 13 adult patients (6 males and 7 females) undergoing intracranial monitoring for epilepsy. The hippocampus was not a seizure focus area of any patients included in the study. Single neuron data were recorded from stereotactic (sEEG) probes, specifically AdTech Medical probes in a Behnke-Fried configuration. Each patient had an average of 3 probes terminating in left and right hippocampus. Electrode locations are verified by co-registered pre-operative MRI and post-operative CT scans. Each probe includes 8 microwires, each with 8 contacts, specifically designed for recording single-neuron activity. Single neuron data were recorded using a 512-channel Blackrock Microsystems Neuroport system sampled at 30 kHz. To identify single neuron action potentials, the raw traces were spike-sorted using the WaveClus sorting algorithm (Chaure et al., 2018) and then manually evaluated. Noise was removed and each signal was classified as multi or single unit using several criteria: consistent spike waveforms, waveform shape (slope, amplitude, trough-to-peak), and exponentially decaying ISI histogram with no ISI shorter than the refractory period (1 ms). We only used single neuron activity for all analyses.

Experimental apparatus. To control their computer avatar, subjects used a joystick that was a modified version of a commercially available joystick with a built-in potentiometer (Logitech Extreme Pro 3D).. The joystick position was read out in the MATLAB psychophysics toolbox running on the stimulus control computer.

Task description. At trial start, two or three shapes appeared on a gray computer monitor placed directly in front of the subject. The yellow circle (15-pixels in diameter) was an avatar that represented the subject and began at the center of the screen. Subject position was determined by the joystick and was limited by the screen boundaries. A square (30 pixels in length) represented the prey(s). Prey movement was determined by a simple algorithm (see below). Each trial ended with either the successful capture of the prey or after 20 seconds, whichever came first. Successful capture was defined as any spatial tangency between the avatar circle and the prey square. Capture resulted in an immediate numeric reward.

Prey movement was generated interactively using an A-star algorithm. Specifically, for every frame (16.67 ms), we computed the cost of 15 possible future positions the prey could move to in the next time-step. These 15 positions were equally spaced on the circumference of a circle centered on the current position of the prey, with a radius equal to the maximum distance the prey could travel within one time-step. The cost was based on two factors: the position in the field and the position of the subject's avatar. The field that the prey moved in had a built-in bias for cost, which made the prey more likely to move toward the center. The cost due to distance from the avatar of the subject was transformed using a sigmoidal function: the cost became zero beyond a certain distance so that the prey did not move, and it became greater as distance from the avatar of the subject decreased. From these 15 positional costs, the position with the lowest cost was selected for the next movement. If the next movement was beyond the screen range, then the position with the next lowest cost was selected until it was within the screen range.

The maximum subject speed was 23 pixels per frame (each frame = 16.67 ms). The maximum and minimum speeds of the prey varied across subjects and were set by the experimenter to obtain a large number of trials. Specifically, speeds were selected so that subjects could capture prey on no more than 85% of trials. To ensure sufficient time of pursuit, the minimum distance between the initial position of each subject avatar and prey was 400 pixels.

Behavioral modeling

Overview of control choice decomposition. The problem we are studying here is how subject's decide between how to pursue goals on a moment to moment basis. This means we must estimate a decision variable over how targets are selected to drive control actions. We motivate this problem by first considering the low-level, individual goal problem.

For a given target, subject's must deploy some policy that dictates their actions for reaching that goal, $u_k(t)$. Here, we consider the controls as the output of a linear controller that is driven by a set of gains that dictate controls (library of controller types considered in Table 1). For example, the simplest proportional controller with a gain (g_k) that minimizes distance to target (e_k) for a single target k is :

$$u_k(t)=g_k\, e_k$$

When there are multiple goals, and each target has different controllers gains, inferring a decision variable requires inferring how a weighted mixture of targets drives subjects' control actions. To do this, we used ideas from control theory to develop a method for decomposing the subject's instantaneous control (i.e., acceleration) into distinct contributions from each target. This previous work has shown control in a nonlinear cost landscape – when there are multiple targets and thus minima in the landscape – can be approximated by a compositional mixture of linear controllers, and the mixing weights in optimal settings are given by the relative cost/value of individual controllers (Todorov, 2009; Dvijotham and Todorov, 2012). Next, we state the general problem and connection to common mixture models, then our approach to estimating this quantity from behavioral data.

At every time point, we want to decompose the subjects measured control signal ($u_{obs}(t)$ = acceleration) into the proportion of their control signal attributable to individual prey (*K*) targets ($u_k(t)$).

$$p\big(u_{obs}(t)\big) = \sum_{i=1}^{K} p(target_k | u_{obs}(t))\, u_k(t)$$

These controls are weighted by the posterior probability of them following a target, given their control signal. Estimation of these weights implicitly defines a multinomial posterior probability over *K* targets given the control signal. This posterior is the same form as the familiar gaussian mixture model (Murphy, 2022). Posterior latent target probabilities are then computed via Bayes rule:

$$w_k(t) \propto p(target_k | u_{obs}(t)) = \frac{p(u_{obs}(t) | target_{k=1})\, p(target_{k=1})}{\sum_{k=1}^{K} p(u_{obs}(t) | target_{j=1})\, p(target_{j=1})}$$

This shows that estimating $w_k(t)$ will implicitly define the integration of an observed data likelihood and categorical prior. Because solving for the weights is a convex optimization problem, we will explicitly optimize (see Model parameterization) the controller weights $w_k(t)$:

$$p\big(u_{obs}(t)\big) = \sum_{i=1}^{K} w_k(t)\, u_k(t)$$

A final note worth mention is how optimizing $w_k(t)$ (ie., target posterior) as a mixture model mixture approach is different from hidden Markov models. Mixture modeling applies soft-assignment or weighting and, as such, using $w_k(t)$ to explain subject's control signal allows blending of controllers and smooth transitions between them. Hidden Markov models use state decoding or hard-assignment, and effectively apply an argmax[$w_k(t)$] to choose a singular controller. In the supplementary (section: model optimization) we show with simulations using a soft-assignment picks up both cases, where switching is either gradual or binary between targets; thus, using soft-assigned weights will correctly detect the target pursuit and switching strategy for either blending of controllers (soft-assignment) or discrete controller choices (hard-assignment).

Model parameterization

To infer an estimated $w_k(t)$ time-series per trial, we estimated maximum a posteriori parameters from behavioral data, including the parameters of w_k ($[\phi, \psi]$) and controller gains ($g_k = \theta$). Both sets of parameters are explained below. Optimization of these parameters proceeded by minimizing the negative log posterior:

$$-log\ p(u_{obs}) \propto \sum_{t=1}^{T} -log\ p\big(u_{pred,K}(t;\theta,\phi,\psi)\big) - log\ p(\theta,\phi,\psi)$$

The first term on the r.h.s. is the negative log likelihood (priors are second). u_{pred} is the model predicted control signal that is determined by the mixture of controls across targets:

$$u_{pred}(\theta,\phi,\psi) = w_1(t;\phi,\psi)\ u_{pred,k=1}(t;\theta) + w_2(t;\phi,\psi)\ u_{pred,k=2}(t;\theta)$$

w_k **parameterization**. Mixture weights (w_k) were parameterized by implicitly modeling time dependent logits with (n=30) radial basis functions equally-spaced over each trial's time. Basis parameters included weights (ϕ) and a shared width parameter (ψ). We chose to implicitly parameterize w_k with basis functions as it allows a flexible and smooth parametrization of transitions between target control blending. Practically, w_k are defined though a softmax over the logits (eq. Xand X). Logits are computed by multiplying basis functions by their optimized parameters.

$$w_1(t) = softmax\big(logits_1(t;\phi,\psi)\big)$$
$$w_2(t) = 1 - w_1(t)$$

Note, this approach is readily extended to the case of K > 2 targets by constraining one target's logits to 1 and increasing the basis parameters to n*K-1.

$u_{pred,k}$ **parameterization**. For generating predicted controls per target, $u_{pred,k}$, we tested different controller types and estimate their gain parameters per controller (g_k). We model the predicted subject's states (x_{pred} ={positions, velocities}) and controls as a linear dynamic system mapping joystick input to the cursor. Joystick position controlled the cursor velocity, and therefore control signals from the subject (u_{obs}) are acceleration, which are composed of the controls from distinct targets ($u_{pred,k}$). These controls and states step the predictions forward and generate a new set of control predictions in each optimization iteration. The linear system describing the subject's cursor is:

$$x_{pred}(t+1) = Ax_{pred}(t) + Bu_{pred}(t)$$

Where **A** and **B** are known system parameters (Δt = sampling rate (1/60 Hz)) that determine the mapping from joystick to subject cursor dynamics.

$A = [1\ 0\ \Delta t\ 0\ ;\ 0\ 1\ 0\ \Delta t\ ;\ 0\ 0\ 1\ 0\ ;\ 0\ 0\ 0\ 1\ ]\ ,\ B = [0\ 0\ \Delta t\ 0\ ; 0\ 0\ 0\ \Delta t\ ]$

Because the subject's control strategy for a given target is unknown, we must also fit the mixing weights using different combinations of controller classes that could generate responses. We considered controllers that use different combinations of error signals to drive target pursuit. Model classes included either proportional or integral errors, where the error signals were of different classes described below, including feedback and feedforward predictive controller errors. All model classes are listed in Table 1

Table 1

Equation	Description
$u_{pred} = g_p\, e_p$	Current position error
$u_{pred} = g_p\, e_p + g_v\, e_v$	Current position and velocity errors
$u_{pred} = g_p\, e_p + g_f\, e_f$	Current and future position errors
$u_{pred} = g_p\, e_p + g_i \int e_p\, e_p$	Current and integral position errors
$u_{pred} = g_p\, e_p + g_f\, e_f + g_i \int e_p$	Current, future, and integral position errors
$u_{pred} = g_p\, e_p + g_f\, e_v + g_i \int e_p$	Current, velocity, and integral position errors
$u_{pred} = g_p\, e_p + g_f\, e_v + g_f\, e_f$	Current, velocity, and future position errors

Error signals are denoted e with subscripts indicating type (e.g., e_p is current position error)

e.g., $e_p = x_{position,cursor} - x_{position,target=k}$

Note that we also included models with future position errors. This controller class allows the subject's predicted controls to be driven by feedforward signals based on the target's predicted (1-step ahead) position. The error signal encodes the difference between the subject's current (t) position and target's future (t+1) position. Target future position was computed using the forward kinematics of the target:

$$x_{target}(t+1) = x_{target}(t) + \Delta t x'_{target} + \Delta t^2 x''_{target}$$

Model loss priors. The parameter priors for the loss are denoted as:

$$p(\theta)\, p(\psi)\, p(\phi) = N\left(0, I\sigma_\theta^2\right) N\left(0, \sigma_\psi^2\right) N(0, (\lambda S)^{-1})$$

The prior over controller gains (θ) was a zero-mean normal distribution, with identity variance – constraining gains independently. A similar prior was used for the basis function width (ψ). For both priors, increasing σ^2 decreases the regularization of the prior. For basis function weights (ϕ), we wanted to penalize deviations from smoothness. For this, we used a gaussian laplacian prior (multivariate gaussian) often used in spline models to penalize the wiggliness of functions (Wood, 2017). This prior uses the inverse precision matrix with a regularization coefficient $(\lambda S)^{-1}$. S is a penalty matrix defined as the 2nd order difference between adjacent weights, penalizing the curvature:

$$\Delta^2\phi = \phi_{i+1} - 2\phi_i + \phi_{i-1}$$

Model optimization. For parameter estimation, we used trust-const in scipy.optimize. During each parameter update, the current parameter set (θ, ϕ, ψ) is used to simulate the predicted dynamics and compute the loss. Loss Jacobians and Hessians were computed automatically using Jax and parameters were iteratively optimized by maximizing the negative log posterior. To ensure positively constrained controller gains (θ) and basis width (ψ) we optimized their log, and used a softplus transform. Each trial was optimized separately. For each optimization run, we fit a specific controller class (same for both targets). This was repeated 4 times for each controller class, with different parameter initializations, and the one with best loss was retained for cross model performance comparisons (next section). This was repeated for each of the seven model classes (Table 1).

Quantifying model performance and computing mixture. After optimizing parameters and retaining the best fit for each model class (and each trial), we computed the evidence lower bound (ELBO) for each model using a Laplace approximation to the parameter posteriors (see Murphy, 2022). The ELBO provides an estimate of lower bound on the marginal log-likelihood of each model, balancing model fit and penalizing the number of parameters.

The next step is computing a mixing trajectory ($e.g., w_1$) for each trial that marginalizes over each model, while accounting for each model's uncertainty in predicted parameters. Because the model probability is quantified via the ELBO, we can use these ELBOs to compute posterior probabilities for each model.

$$p(M_k \mid u_{obs}) = \frac{exp(ELBO_m)}{\sum_1^j exp\left(ELBO_j\right)}$$

We use these model probabilities to perform Bayesian model averaging to obtain a singular w_1 trajectory. In short, the w_1 trajectory around the MAP parameters for each model class is weighted and summed. Computing $w_1$through model averaging accounts for model uncertainty by marginalizing over the generating models, and this is the target blending estimate we use for downstream analysis.

$$\text{BMA}(w_1) = w_{1,k}\, p(M_k \mid u_{obs})$$

Comparison of different control mixing strategies

We also want to compare a smooth controller blending model to two other possible controller selection models that could drive behavior. We can use the same model fitting framework as above. For one comparison model, we fit a model that assumes subject's choose a single controller at an given moment. This is equivalent to argmax model over individual target controllers, is nested within the smooth blending model, and is akin to how an HMM would choose the generative controller. Comparison of this model to the smooth blending effectively tells us how much of behavior is explainable by a continuous mixing versus discrete selection strategy. This argmax model is formulated for the predicted control as:

$$u_{pred}(\theta, \phi, \psi) = argmax_{\,u_{pred,k}}\, (w_1(t; \phi, \psi), w_2(t; \phi, \psi))$$

Another strategy that subjects could use is blending inputs with a single controller. Note that this control strategy is not nested within the smooth controller blending model, as control signals produced by multiple controllers versus one produced by weighting multiple inputs are not related through any linear transform (Supplementary note X). The input blending model is formulated using a single set of controller gains rather than distinct gains per target, and weighting the inputs:

$$u_{obs}(t) = \theta \sum_{i=1}^{K} w_k(t)\, x_k(t)$$

Both comparison models were fit using the same approach as the controller blending model. Bayesian model selection with the ELBO was used to select the best fitting model for each trial. For each subject, we computed the mean posterior probability of each model type for group level model selection.

Behavioral analysis

Finding behavioral switch points. To separate instantaneous moments into pursuit versus change-of-mind points between targets, we first searched each trial to find points where W_t =0.5. These W_t points denote crossings between targets. We excluded any putative switch points that occurred within 150 ms of the trial start time, under the assumption that these likely represent a correction to an initial choice rather than true pursuit switching or change-of-mind. Using the remaining potential switch points, we created a window of -300 to +300 ms around these points. The switch trajectories were then categorized as falling into one of three categories. The main category (1) was those where a full switch was made between targets, the other two were either (2) partial switches or (3) were unclassifiable. To achieve this classification, we used a PCA across all switch trajectories from within a session. The 1st principal component (PC) always captured the sigmoidal shaped full switch trajectory of interest. We computed the cosine similarity of this 1st PC score and all switch trajectories and retained those with a similarity of > 0.9. We ran a PCA again on this subset, with the 1st PC again producing the canonical sigmoidal trajectory. In this case, we retained those with cosine

similarity of >0.97. This threshold proved to be conservative enough to remove all partial switches, based on visual inspection. In total, an average of 35% of W_t crossing-points were labeled as legitimate switches and forwarded for further analysis. Switch onset and offset timing were determined using a linear fitting change-point algorithm that looked for changes in mean and slope of the W_t trajectory. To find the switch start, we searched backward in time from the crossing of W_t = 0.5. Switch ends were found by searching forward in time.

Switch statistics. The factors underlying the propensity to make target switches were modeled using a Bayesian binomial mixed effects model implemented in statsmodels in python (Seabold and Perktold, 2010). We used this approach to allow modeling all participants together, while accounting for individual variation in predictors. Predictor significance was determined using the 95% posterior credible interval. In terms of predictors, we asked if switch probability was sensitive to the time elapsed in trial, the difference in prey reward, and several prey distance variables. Specifically, we included the mean distance to the prey being pursued and the mean distance of the prey switched to, where the mean is calculated in the 100 ms leading up to a switch. We also included a factor to determine the mean rate of change of distance (distance closure) between both preys, too. The distance closure captures whether either prey getting closer or farther away (on average) altered switch probability. Each of these variables was centered by using their median values extracted from trials in which there were no switches. This allows interpreting the coefficients in terms of changes with respect to control, no-switch trials. We used the mixed

Neural analysis

Single neuron encoding of task variables. To assess how neurons are tuned to task variables, we employed a linear-nonlinear Poisson model as a Generalized additive model (GAM; Balzani et al., 2020; Wood, 2017). Specifically, we aimed to quantify how neurons were tuned to subject speed, relative prey distance, relative prey speed, relative time in trial, prey reward difference, and the model-based blending variable

(W_t). As our main goal was estimating tuning W_t, we included all of these terms and interactions with W_t in the model to control for any confounding.

The GAM was fit with a penalized regression smoothing spline model (Wood, 2017). Penalized smoothing spline GAMs were chosen as they have been shown to offer good recovery of nonlinear neuron tuning, particularly in tasks where tuning is estimated with respect to continuously changing variables (e.g., speed; Balzani et al., 2020). Natural cubic regression splines served as a basis function. Cubic splines are convenient as a smooth interpolant basis because their only hyperparameter is the number of bases (J; Wood, 2017). The spike counts y_t are fit in a GAM with univariate and interaction effects with basis functions b_{j,x_1} (basis j and variable x_1). As an example, in a two-variable model the firing is modeled as:

$$log(y_t) = \sum_{j=1}^{J} b_{j,x_1}\beta_{j,x_1} + \sum_{j=1}^{J} b_{j,x_2}\beta_{j,x_2} + \sum_{j=1}^{J} (b_{j,x_1} \otimes b_{j,x_2})\beta_{j,x_{1\otimes 2}}$$

The first two terms are examples of single (marginal) variable tuning effects. The latter component encodes the interaction (nonlinear mixed selectivity) term as a tensor over the two univariate basis functions.

The model was fit in a Bayesian framework using stochastic variational inference in NumPyro. To estimate smooth and regularized tuning functions, we follow previous implementations (Balzani et al., 2020; Wood, 2017) and implement two types of regularization. Specifically, regularization was implemented through two types of priors on model coefficients that (1) control for basis wiggliness and enforce smoothness across basis functions, and (2) force sparse sets of bases similar to an L1 regularization. Combining these two effectively performs model selection by both smoothing across bases (1) and (2) forcing bases that don't contribute to the model likelihood to be zero. These priors are implemented for basis coefficients as a multivariate normal distribution, for a given variable as:

$$\beta \sim MVN(0, \Sigma_\beta)$$

$$\Sigma_\beta = (\lambda_j S + \lambda_* S_*)$$

The smoothing/regularization terms (λ) are treated as random variables in the Bayesian approach, with Half-Cauchy priors. This allowed estimating separate smoothing terms per univariate and tensor model term directly from the data, without cross-validation. S encodes a 2nd-order difference penalty between basis coefficients, inducing smoothing as a Laplacian graph prior that is used to construct the coefficients prior covariance. We point the reader to Wood (2017) for more specifics on how S and S_* are constructed.

To fit the models and approximate the posterior distribution of parameters, we employed a variational inference approach with a Gaussian guide. Variational parameters (mean and covariance) were optimized by minimizing the negative evidence lower bound (ELBO). This approach balances computational efficiency with parameter uncertainty estimation. All models were optimized using Adam (learning rate: 1e-3), each run for 10,000 steps.

To further determine coefficient significance, 99% posterior credible intervals were used to evaluate whether they included zero. Those variables with intervals non-overlapping with zero were retained. To determine whether the full interaction (nonlinear mixing with blending) model was a significantly better fit only a model with marginal effects, or a baseline intercept-only model, we computed a z-score test using the difference in expected log predictive density (ELPD) between each set of two models and the standard of the difference in ELPD (Gelman, 2013). The ELPD was computed using WAIC. Models with a z-score corresponding to $p < 0.01$ were denoted as significant. Specifically, we first tested whether the marginal effects model only was a better fit than the baseline. Then we used the ELPD test again to select whether the full model with interactions was a more probable model than the marginal effects model. The winning model was retained as the tuning coefficients for that neuron.

Fisher information of W tuning. We computed a cross-validated and denoised estimate of fisher information using neuron tuning to W_t. We computed tuning curves for each neuron in twelve bins. We used 5-fold cross-validation when computing the tuning

curves. Tuning curves from the training set for each brain region were reduced using principal components analysis, retaining enough dimensions to account for 90% of variance. Fisher information was computed using PCA projections of the test set, and taking the euclidean distance between adjacent levels of W_t of these projections (Kriegeskorte and Wei, 2021). To test for significant information, we created a null distribution for permutation testing by permuting W_t labels during calculation of the tuning curves. This distribution was created by repeating this process 1000 times, and the p-value was computed by comparing the mean actual 5-fold information to the permutation distribution.

Demixed principal components analysis. To examine the low-dimensional population dynamics related to behavioral controller blending, we performed dimensionality reduction on neural data using demixed principal components analysis (dPCA; Kobak et al., 2016). Using dPCA, we can extract linear subspaces of population activity that maximize the variance specifically related to separating levels of blending. In effect, this approach allows us to determine whether the population has linearly decodable representations of blending.

Because our aim was assessing whether neural population dynamics closely tracked the behavioral blending variable, we focused on time windows in which there was an explicitly consistent blending trajectory: change of mind between prey targets. To decompose the neural data, we first sorted all switch points, during a trial, for a given subject, and within a session, into those that switched from a low to high or high to low value target. Under the assumption that blending differs in a mirrored fashion (high to low versus low to high W_t), across time, for these switch directions, we can take the decoding subspace as the dynamic subspace for W_t. We additionally separated the data in low and high levels of prey reward difference.

We fit the dPCA model to firing rate tensors as pseudo-populations that included all subjects. The model included the effects for W_t , prey reward difference, and their interaction. Each fit was done separately for dACC (n =149) and Hippocampus (n = 314). The fitting and significance testing procedure was carried out by performing 1000

repetitions randomly sampling (stratified monte-carlo) the data, and holding-out 5% of trials (balanced between switch directions). A dPCA model was then trained using a fixed regularization of $1x10^{-5}$. Firing rate test-sets were then projected along the top decoder dimension for the switch direction. We took the decoded switch direction to be the minimum distance of the projected test data to the projected training data. This classification was computed for each of the 30 time-points in the switch window. Second, we established significance thresholds by repeating the above procedure, but randomly swapping trial labels to create a null accuracy distribution. We created the null distribution by repeating the 1000 iterations 50 times. For each of the 50 sets, we computed the null decoding accuracy. We determined that decoding was statistically significant when the decoding accuracy was greater than the null for 100% of the 50 sets.

Ramp decoding. We estimated ramping population signals used in Figure 4 by adopting a decoding method previously used for perceptual decision making (Steinemann et al., 2024). Ramping signals were estimated by assessing whether we could decode if a switch decision was going to happen within 100 ms.

For pre-processing neural data, we first aligned spikes to the target switch onset. Then we split the data into the separate switch directions. These splits were then split into training and test folds (5 folds), retaining only neurons that had at least 25 instances in each condition. We then sampled time-windows of firing of length equal to the switch decision period (30 samples) that did not contain a switch. We sampled control data equal to the length of train and test folds. Sampling of control condition spikes was repeated for each training-test fold. These firing windows provide the comparator neural activity for decoding.

For each fold, we z-scored and performed PCA on the concatenated control and training data. The z-scoring and PCA transforms were then applied to the concatenated control and test fold data. Time-based decoding was performed on 80ms windows that were stepped by one window (16.67 ms sampling rate). We then trained and tested a linear support vector machine on each window, and used the average performance over

folds for decoding accuracy. Decoding significance was estimated by repeating this trial sampling procedure 1,000 times but permuting the condition labels. All decoding was implied in scikit-learn (Pedregosa et al., 2011).

Clustering ramp. We clustered firing during the target switches to assess whether classes of single neurons also exhibit ramping behavior. To obtain a meaningful clustering based on latent neural dynamics and potential, we first fit a linear dynamic system to trial-averaged, gaussian smoothed (FWHM 60 ms) firing rates using the ssm package in Python. In addition, the models were fit to both switch directions (high to low and low to high value prey) simultaneously, giving them shared parameters but different inputs. We used the normalized relative target distance as inputs to the model. This approach to clustering accounts for similar latent dynamics evolving under different inputs. The number of the latent state dynamics for each model were optimized by computing the loss (ELBO) from a held out test in 5-fold cross-validation. This was done separately for each brain region. We found three latent states were suitable for describing each area.

Clustering was then performed on the firing rate embedding matrix C using K-means. The C matrix links latent states to firing rates. The number of clusters was optimized using the silhouette score, separate for each brain region. We chose clusters to retain based on those that exhibited significant differences in activity between time-locked segments with switched and control segments without. Significance was determined by permuting (1,000 times) the mean clustered firing rate between the actual switches and controls.

Supplementary Material

S1. Mixture of controllers model fitting accuracy and ground-truth recovery

To test the validity of our modeling approach using basis functions and controllers, we simulated ground-truth data with a known W_t that we designed. To cover the space from simple to complex W_t profiles, we considered five different types of W_t. We show examples of these five types in Table S1. To simulate data, we sampled 20 random trials across both subjects' sessions, with 5 of each having an approximate length of 0.91 sec, 1.45 sec, 2.83 sec, 4.1 sec long. In the main text, we fit across a range of number of bases and model-average. We chose a fixed number of bases in simulation to fix the complexity of the model, as increasing basis numbers will always increase fit quality.

As a first test, we compared whether different controller types (see Table 1) could be recovered as the best fitting model compared to a baseline of just a position controller (**Figure S1A**). The posterior model probabilities indicated that for all controller classes we could recover the true generative controller model on average better compared to a baseline (all probabilities > 0.5 for the generative model).

We next examined how well we could recover the generative kinematics for each controller model class. In **Figure S1B** we show the average correlation between predicted and actual positions for 20 simulated trials per controller model type. We get good recovery across all model types, with an average correlation of 0.83 across model classes.

The quality of inferring ground-truth W_t was similar to positions. The correlations between the true and predicted W_t are shown in **Figure S1C.** Across the generative models, we get an average correlation of 0.87. These correlations are averaged over different levels of W_t complexity, which was controlled using a scaling parameter for a generating gaussian process model. In Figure **S1D-I**, we show example fits of the simulated and recovered W_t for each class of generative models.

Overall, these simulations and analysis indicate that our method did a job at detecting the correct model class (**Figure S1A)**, as well as recovering the actual latent target specific controls (**Figure S1B**) and W_t (**Figure S1C**).

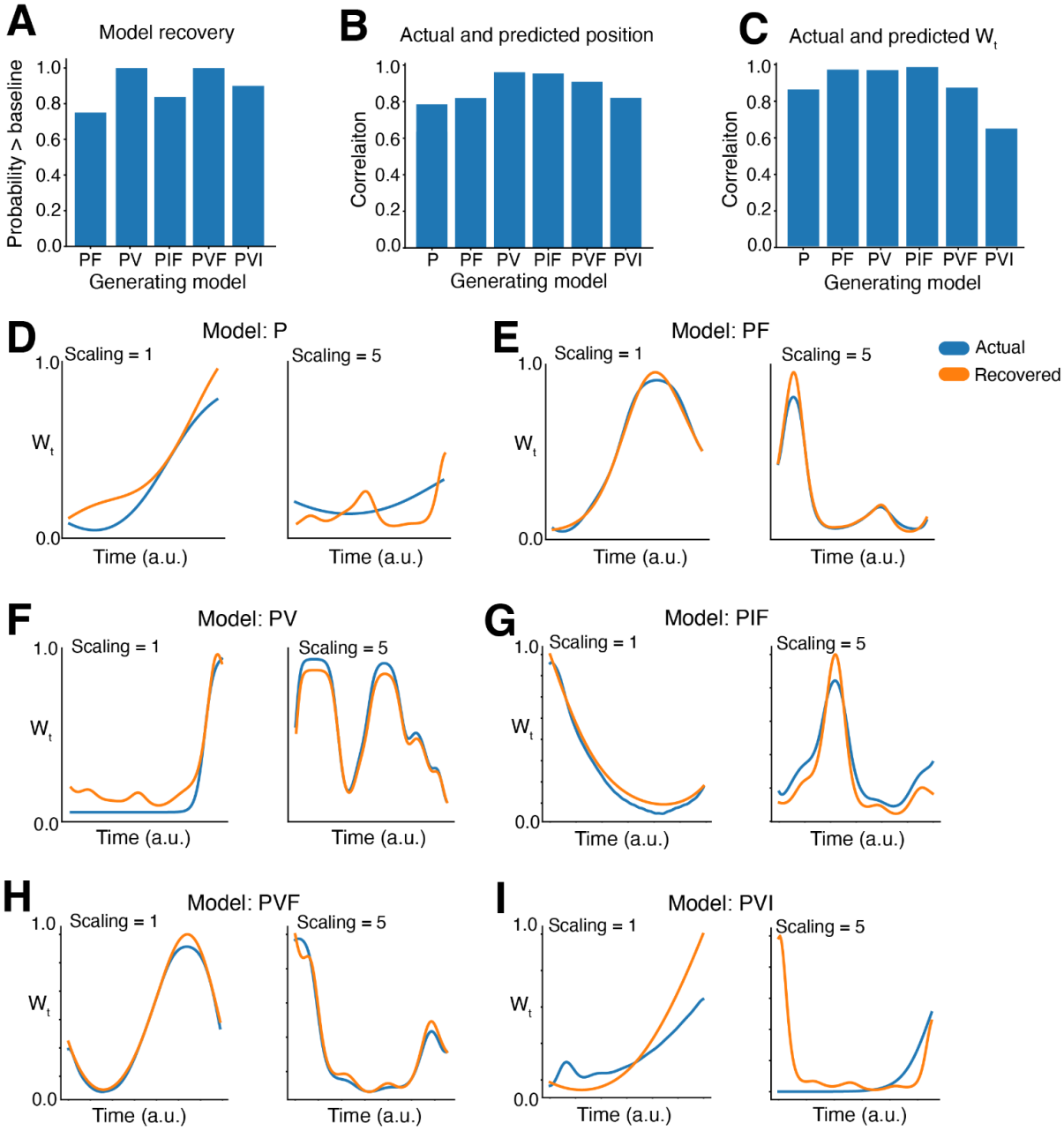

Figure S1.| **Model recovery of Wt and controllers.** (**A**) Probability of choosing the generative model as being better than a baseline, position only controller model. Higher values indicate better recovery. (**B**). Correlation between true and predicted controller position time series based on fit of generative model class. Higher correlations indicate better recovery, and each bar is the average of 30 runs. (**C**). Correlation between true and predicted Wt time series based on fit of generative model class. Higher correlations indicate better recovery, and each bar is the average of 30 runs. (**D**) - (**I**) examples from simulations and fits showing the generative Wt and the predicted model recovered Wt. Each set of plots is a different generative model. The left

columns in each (scaling = 1) are those with simple generating Wt, and the right right columns are more complex Wt (scaling = 5). The scaling is the scale parameters used in a gaussian process generator.

S2. Proof of controller model non-equivalencies

We outline a simple algebraic proof here that two of the considered model classes – mixing of controllers versus mixing of inputs – are not related by any linear transform, making them constitute different behavioral model classes. We formulate this argument for a single time-point, input feature and controller dimension. An assumption we make is that controller gains are at steady state, as formulated in an infinite horizon or average cost scenario; ie., controller gains are not state or time-dependent. This problem is formulated is as follows:

$$u_{obs:controller\ mixing} = \sum_{k=1}^{K} w_k\, u_k$$

Which represents the controller blending model, where control signals for two controllers are specified with gains (L_k):

$$u_1 = -L_1\, x_1$$
$$u_2 = -L_2\, x_2$$

Using this, it is convenient to rewrite this as:

$$u_{obs:controller\ mixing} = -w_1 L_1\, x_1 - w_2 L_2\, x_2$$

For comparison, we create the same setup for input mixing models:

$$u_{obs:input\ mixing} = -L \sum_{k=1}^{K} w_k\, x_k$$

Rewriting in expanded form:

$$u_{obs:input\ mixing} = -L\,(w_1\, x_1 + w_2\, x_2)$$

Now testing for equivalence across models, we find that are only equivalent when the controller gains are identical:

$$u_{obs:controller\ mixing} = u_{obs:input\ mixing}$$

$$-w_1 L_1\, x_1 - w_2 L_2\, x_2 = -L\,(w_1\, x_1 + w_2\, x_2)\ \ iif\ \ L_1 = L_2$$

Thus, the two models can only be seen as equal when the gains of both controllers are equal.