DEEP NEUROETHOLOGY OF A VIRTUAL RODENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Parallel developments in neuroscience and deep learning have led to mutually productive exchanges, pushing our understanding of real and artificial neural networks in sensory and cognitive systems. However, this interaction between fields is less developed in the study of motor control. In this work, we develop a virtual rodent as a platform for the grounded study of motor activity in artificial models of embodied control. We then use this platform to study how motor activity differs across contexts by training a model to solve four complex tasks. Using methods familiar to neuroscientists, we describe the behavioral representations and algorithms employed by different layers of the network using a neuroethological approach to characterize motor activity relative to the rodent's behavior and goals. We find that the model uses two classes of behavioral representations which respectively encode the task-specific behavioral strategies and task-invariant behavioral kinematics. These representations are reflected in the sequential activity and population dynamics of neural subpopulations. Overall, the virtual rodent facilitates grounded collaborations between deep reinforcement learning and motor neuroscience.

1 INTRODUCTION

Animals have nervous systems that allow them to coordinate their movement and perform a diverse set of complex behaviors. Mammals, in particular, are generalists in that they use the same general neural network to solve a wide variety of tasks. This flexibility in adapting behaviors towards many different goals far surpasses that of robots or artificial motor control systems. Hence, studies of the neural underpinnings of flexible behavior in mammals could yield important insights into the classes of algorithms capable of complex control across contexts and inspire algorithms for flexible control in artificial systems.

Recent efforts at the interface of neuroscience and machine learning have sparked renewed interest in constructive approaches in which artificial models that solve tasks similar to those solved by animals serve as normative models of biological intelligence. Researchers have attempted to leverage these models to gain insights into the functional transformations implemented by neurobiological circuits, prominently in vision (Khaligh-Razavi & Kriegeskorte, 2014; Yamins et al., 2014; Kar et al., 2019), but also increasingly in other areas, including audition (Kell et al., 2018) and navigation (Banino et al., 2018; Cueva & Wei, 2018). Efforts to construct models of biological locomotion systems have informed our understanding of the mechanisms and evolutionary history of bodies and behavior (Grillner et al., 2007; Ijspeert et al., 2007; Ramdya et al., 2017; Nyakatura et al., 2019). Neural control approaches have also been applied to the study of reaching movements, though often in constrained behavioral paradigms (Lillicrap & Scott, 2013), where supervised training is possible (Sussillo et al., 2015; Michaels et al., 2019).

While these approaches model parts of the interactions between animals and their environments (Chiel & Beer, 1997), none attempt to capture the full complexity of embodied control, involving how an animal uses its senses, body and behaviors to solve challenges in a physical environment. The development of models of embodied control is valuable to the field of motor neuroscience, which typically focuses on restricted behaviors in controlled experimental settings. It is similarly valuable to AI research, in which flexible models of embodied control could be applicable to robotics.

Here, we introduce a virtual model of a rodent to facilitate grounded investigation of embodied motor systems capable of solving multiple cognitive challenges. The virtual rodent affords a new

opportunity to directly compare principles of artificial control to biological data from real-world rodents, which are more experimentally accessible than humans. We draw inspiration from emerging deep reinforcement learning algorithms which now allow artificial agents to perform complex and adaptive movement in physical environments with sensory information that is increasingly similar to that available to animals (Peng et al., 2016; 2017; Heess et al., 2017; Merel et al., 2019a;b). Similarly, our virtual rodent exists in a physical world, equipped with a set of actuators that must be coordinated for it to behave effectively. It also possesses a sensory system that allows it to use visual input from an egocentric camera located on its head and proprioceptive input to sense the configuration of its body in space.

There are several questions one could answer using the virtual rodent platform. Here we focus on the problem of embodied control across multiple tasks. While some efforts have been made to analyze neural activity in reduced systems trained to solve multiple tasks (Song et al., 2017; Yang et al., 2019), those studies lacked the important element of motor control in a physical environment. Our rodent platform presents the opportunity to study how representations of movements as well as sequences of movements change as a function of goals and task contexts.

To address these questions, we trained our virtual rodent to solve four complex tasks within a physical environment, all requiring the coordinated control of its body. We then ask "Can a neuroscientist understand a virtual rodent?" – a more grounded take on the originally satirical "Can a biologist fix a radio?" (Lazebnik, 2002) or the more recent "Could a neuroscientist understand a microprocessor?" (Jonas & Kording, 2017). We take a more sanguine view of the tremendous advances that have been made in computational neuroscience in the past decade, and posit that the supposed 'failure' of these approaches in synthetic systems is partly a misdirection. Analysis approaches in neuroscience were developed with the explicit purpose of understanding sensation and action in real brains, and often implicitly rooted in the types of architectures and processing that are thought relevant in biological control systems. With this philosophy, we use analysis approaches common in neuroscience to explore the types of representations and dynamics that the virtual rodent's neural network employs to coordinate multiple complex movements in the service of solving motor and cognitive tasks.

2 Approach

2.1 Virtual rodent body

We implemented a virtual rodent body (Figure 1) in MuJoCo (Todorov et al., 2012), based on measurements of laboratory rats (see Appendix A.1). The rodent body has 38 controllable degrees of freedom. The tail, spine, and neck consist of multiple segments with joints, but are controlled by tendons that co-activate multiple joints (spatial tendons in MuJoCo). Upon publication, we will release the rodent model.

The virtual rodent is equipped with a set of proprioceptive inputs as well as "raw" egocentric RGB-camera (64×64 pixels) input from a head-mounted camera. The proprioceptive inputs include internal joint angles and angular velocities, the positions and velocities of the tendons that provide actuation, egocentric vectors from the root (pelvis) of the body to the positions of the head and paws, a vestibular-like upright orientation vector, touch or contact sensors in the paws, as well as egocentric acceleration, velocity, and 3D angular velocity of the root.

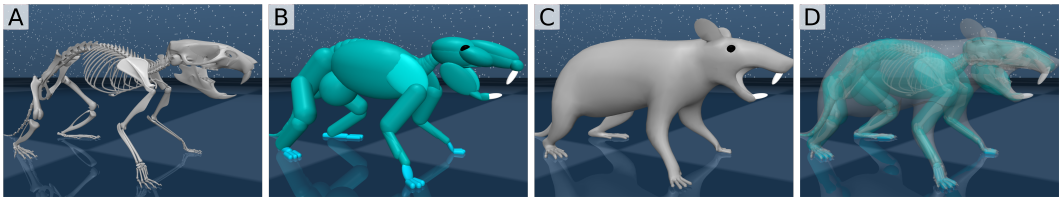

Figure 1: (A) Anatomical skeleton of a rodent (as reference; not part of physical simulation). (B) A body designed around the skeleton to match the anatomy and model collisions with the environment. (C) Purely cosmetic skin to cover the body. (D) Semi-transparent visualization of (A)-(C) overlain.

2.2 VIRTUAL RODENT TASKS

We implemented four tasks adapted from previous work in deep reinforcement learning and motor neuroscience (Merel et al., 2019a; Tassa et al., 2018; Kawai et al., 2015) to encourage diverse motor behaviors in the rodent. The tasks are as follows: (1) Run along a corridor, over "gaps", with a reward for traveling along the corridor at a target velocity (Figure 2A). (2) Collect all the blue orbs in a maze, with a sparse reward for each orb collected (Figure 2B). (3) Escape a bowl-shaped region by traversing hilly terrain, with a reward proportional to distance from the center of the bowl (Figure 2C). (4) Approach orbs in an open field, activate them by touching them with a forepaw, and touch them a second time after a precise interval of 800ms with a tolerance of $\pm$100ms; there is a time-out period if the touch is not within the tolerated window and rewards are provided sparsely on the first and second touch (Figure 2D). We do not provide the agent with a cue or context indicating its task. Rather, the agent must infer the task from the visual input and behave appropriately.

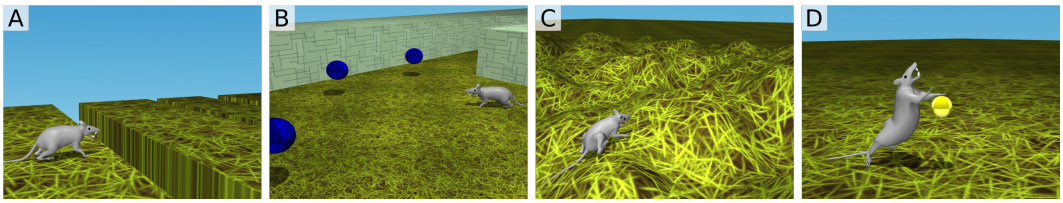

Figure 2: Visualizations of four tasks the virtual rodent is trained to solve: (A) jumping over gaps ("gaps run"), (B) foraging in a maze ("maze forage"), (C) escaping from a hilly region ("bowl escape"), and (D) touching a ball twice with a forepaw with a precise timing interval between touches ("two-tap").

2.3 TRAINING A MULTI-TASK POLICY

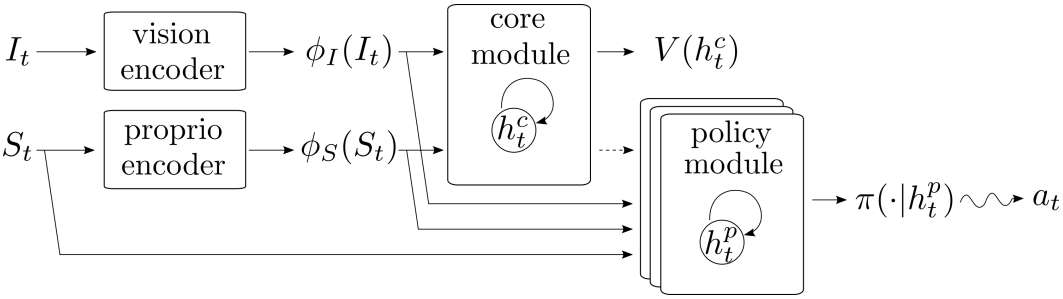

Figure 3: The virtual rodent agent architecture. Egocentric visual image inputs are encoded into features via a small residual network (He et al., 2016) and proprioceptive state observations are encoded via a small multi-layer perceptron. The features are passed into a recurrent LSTM module (Hochreiter & Schmidhuber, 1997). The core module is trained by backpropogation during training of the value function. The outputs of the core are also passed as features to the policy module (with the dashed arrow indicating no backpropogation along this path during training) along with shortcut paths from the proprioceptive observations as well as encoded features. The policy module consists of one or more stacked LSTMs (with or without skip connections) which then produce the actions via a stochastic policy.

Emboldened by recent results in which end-to-end RL produces a single terrain-adaptive policy (Peng et al., 2016; 2017; Heess et al., 2017), we decided to train a single model on the multiple motor-control-reliant tasks. To train a single policy to perform all four tasks, we used an IMPALA-style setup for actor-critic DeepRL (Espeholt et al., 2018); parallel workers collected rollouts, logged them to a replay, from which a central learner sampled data to perform updates. The value-function critic was trained using off-policy correction via V-trace. To update the actor, we used a variant of MPO (Abdolmaleki et al., 2018) where the E-step is performed using advantages determined from

the empirical returns and the value-function, instead of the Q-function (Anonymous, 2019). Empirically, we found that the "escape" task was more challenging to learn during interleaved training relative to the other tasks. Consequently, for the results we present, we trained a single-task expert on the escape task and trained the multi-task policies using kickstarting for that task (Schmitt et al., 2018), with a weak coefficient (.001 or .005). Kickstarting on this task made the seeds more reliably solve all four tasks, facilitating comparison of the multi-task policies with different architectures (i.e. the policy having 1, 2, or 3 layers, with or without skip connections across those layers). The procedure yields a single neural network that is capable of using visual inputs to determine how to behave as well as coordinating its body to move in ways required to solve the tasks. See video examples of a single policy solving episodes of each task: gaps, forage, escape, and two-tap.

3 Analysis

We analyzed the virtual rodent's neural network activity in conjunction with its behavior to attempt to characterize how it solves multiple tasks. We apply analyses and perturbation techniques adapted from neuroscience, where a range of techniques have been developed to highlight the properties of real neural networks. Biological neural networks have been hypothesized to control, select, and modulate movement through a variety of debated mechanisms, ranging from explicit neural representations of muscle forces and behavioral primitives, to more abstract production of the dynamics that could enable movement (Churchland et al., 2012; Graziano, 2006; Kalaska, 2009). A challenge with nearly all of these models however is that they have largely been inspired by findings from individual behavioral tasks, making it unclear how to generalize them to a broader range of naturalistic behaviors. To provide insight into mechanisms underlying movement in the virtual rodent, and to potentially give insight by proxy into the mechanisms underlying behavior in real rats, we thus systematically tested how the different network layers encoded and generated different aspects of movement.

For all analyses we logged the virtual rodent's kinematics, joint angles, computed forces, sensory inputs, and the cell unit activity of the LSTMs in core and policy layers during 25 trials per task from each network architecture.

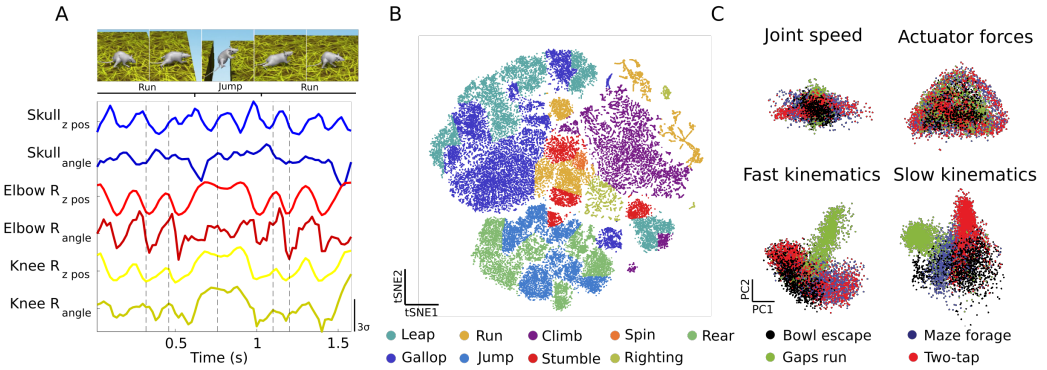

Figure 4: Ethology of the virtual rodent. (A) Example jumping sequence in gaps run task with a representative subset of recorded behavioral features. Dashed lines denote the time of the corresponding frames (top). (B) tSNE embedding of 60 behavioral features describing the pose and kinematics of the virtual rodent, colored by hand-labeling of coarse behavioral categories. The behavioral feature set is sufficient to segment meaningful behaviors into stereotyped behavioral clusters. (C) The first two principal components of different behavioral features reveals that behaviors are shared across tasks at short, 5-25 Hz timescales (fast kinematics), but no longer 0.3-5 Hz timescales (slow kinematics).

3.1 Virtual rodents exhibit behavioral flexibility.

We began our analysis by quantitatively describing the behavioral repertoire of the virtual rodent. A challenge in understanding the neural mechanisms underlying behavior is that it can be described at

many timescales. On short timescales, one could describe rodent locomotion using a set of actuators that produce joint-specific patterns of forces and kinematics. However on longer timescales, these force patterns are organized into coordinated, re-used movements, such as running, jumping, and turning. These movements can be further combined to form behavioral strategies or goal-directed behaviors. Relating neural representations to motor behaviors therefore requires analysis methods that span multiple timescales of behavioral description. To systematically examine the classes of behaviors these networks learn to generate and how they are differentially deployed across tasks, we developed sets of behavioral features that describe the kinematics of organismal-scale behavior of the animal on fast (5-25 Hz), intermediate (1-25 Hz) or slow (0.3-5 Hz) timescales (Appendix A.2 A.3). As validation that these features provided a feature set capable of distinguishing between different behaviors, embedding these features using tSNE (Maaten & Hinton, 2008) produced a behavioral map in which virtual rodent behaviors, including rearing, jumping, running, climbing and spinning, were segregated to different regions of the map (Figure 4B)(see video).

Interestingly, we found that the many of the behaviors that the virtual rodent generated were shared across tasks, suggesting that the agent exhibited behavioral flexibility. On short timescales, patterns of activation of both low level motor features like joint speed and actuator forces, and longer timescale behavioral features were largely shared across tasks (Figure 4C). In contrast, longer timescale behavioral features were more differentiated across tasks, suggesting that the network learned to adapt similar movements in a selective manner for different tasks.

3.2 Networks primarily reflect behaviors, not forces

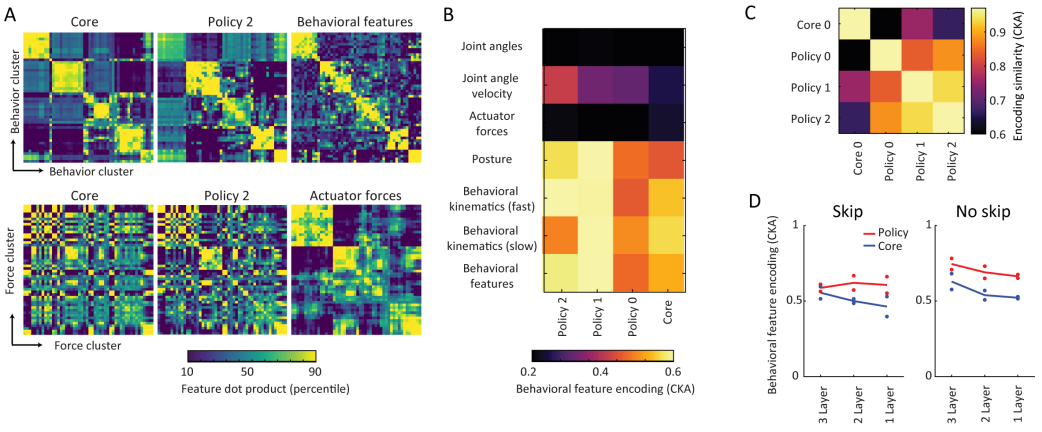

Figure 5: Representational structure of the rodent's neural network. (A) Example feature similarity matrices, of the core and policy networks, behavioral features, and actuator forces, computed across 50 different behavioral or force categories. Similarity was assessed by computing the dot product of either the neural population vector or the behavioral feature vector within each category. (B) CKA index of neural and behavioral feature similarity matrices for 3 and 1 policy layer architectures. (C) CKA index of feature similarity matrices across all pairs of network layers. (D) Average CKA index between core and policy layers and behavioral features, compared across architectures. Points show values from individual network seeds.

We next examined the neural activity patterns underlying the virtual rodent's behavior, to test if networks produced behaviors through explicit representations of forces, kinematics or behaviors. As expected, core and policy units operate on distinct timescales (See Appendix A.3, Figure 9). Units in the core typically fluctuated over timescales of 1-10 seconds, likely representing variables associated with context and reward. In contrast, units in policy layers were more active over subsecond timescales, potentially encoding motor and behavioral features.

To quantify which aspects of behavior were encoded in the core and policy layers, and how these encoding patterns varied across layers, we turned to representational similarity analysis (RSA) (Kriegeskorte et al., 2008; Kriegeskorte & Diedrichsen, 2019). RSA provides a global measure of how well different features are encoded in different layers of a neural network by analyzing the

geometries of network activity upon exposure to several stimuli, such as objects or odorants. To apply RSA, first a representational similarity (or equivalently, dissimilarity) matrix is computed that quantifies the similarity of neural population responses to a set of stimuli. To test if different neural populations show similar stimulus encodings, these similarity matricies can then be directly compared across different network layers. Multiple metrics, such as the matrix correlation or dot product can be used to compare these neural representational similarity matricies. Here we used the linear centered kernel alignment (CKA) index, which shows invariance to orthonormal rotations of population activity (Kornblith et al., 2019).

RSA can also be used to directly test how well a particular stimulus feature is encoded in a population. If each stimuli can be quantitively described by one or more feature vectors, a similarity matrix can also be computed across the set of stimuli themselves. The strength of encoding of a particular set of features can by measured by comparing the correlation of the stimulus feature similarity matrix and the neuronal similarity matrix. The correlation strength directly reflects the ability of a linear decoder trained on the neuronal population vector to distinguish different stimuli (Kriegeskorte & Diedrichsen, 2019). Unlike previous applications of RSA in the analysis of discrete stimuli such as objects, (Khaligh-Razavi & Kriegeskorte, 2014; Yamins et al., 2014) behavior evolves continuously. To adapt RSA to behavioral analysis, we partitioned time by discretizing each behavioral feature into 50 clusters (Appendix A.4).

As expected, RSA revealed that core and policy layers encoded distinct behavioral features. Policy layers contained greater information about fast timescale behaviors in a manner that was largely conserved across layers, while core layers showed more moderate behavioral encoding that was stronger for slow behavioral features (Figure 5B,C). This difference in encoding was largely consistent across all architectures tested (Figure 5D).

More interestingly, the feature encoding of policy networks was somewhat consistent with the emergence of a hierarchy of behavioral abstraction. In networks trained with three policy layers, representations were distributed in timescales across layers, with the last layer (Policy 2) showing stronger encoding of fast behavioral features, and the first layer (Policy 0) instead showing stronger encoding of slow behavioral features. However, policy layers, even those close to the motor periphery, did not show strong explicit encoding of behavioral kinematics or forces.

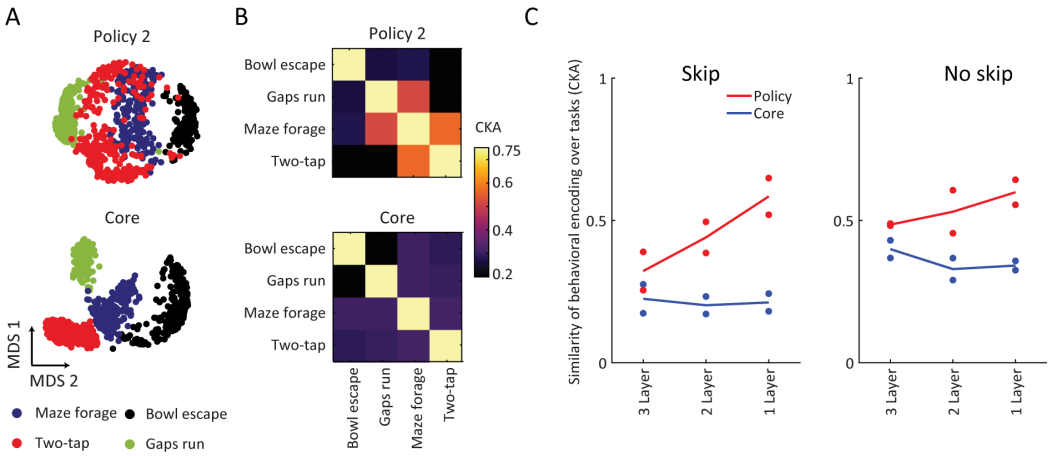

Figure 6: Policy representations are shared across behaviors. (A) Two-dimensional multidimensional scaling embeddings of core and policy activity shows that core representations are largely distinct across tasks. (B) CKA index of the policy 2 and core network representations of behavioral features across different tasks. Policy 2, but not core networks show similar encoding patterns across the across the maze forage and two-tap tasks, as well as the gaps run and maze forage tasks, consistent with the shared behaviors used across these tasks. (C) The similarity of behavioral feature encoding (CKA index) across different architectures demonstrates that networks with fewer layers show greater similarity across tasks. Points show values from individual seeds.

3.3 BEHAVIORAL REPRESENTATIONS ARE SHARED ACROSS TASKS

We then investigated the degree to which the rodent's neural networks were able to use the same neural representations to produce behaviors, such as running or spinning, that were shared across tasks. Embedding population activity into two-dimensions using multidimensional scaling revealed that core neuron representations were highly distinct across all tasks, while policy layers contained more overlap (Figure 6A), suggesting that some behavioral representations were re-used. Comparison of representational similarity matricies for behaviors that were shared across tasks revealed that policy layers tended to possess a relatively similar encoding of behavioral features, especially fast behavioral features, over tasks (Figure 6C; Appendix A.4). This was validated by inspection of neural activity during individual behaviors shared across tasks Appendix A.5, Figure 10). Core layer representations across almost all behavioral categories were more variable across tasks, consistent with encoding a longer timescale representation of behavioral sequences or abstract encodings of task variables.

Interestingly, when comparing this cross-task encoding similarity across architectures, we found that one layer networks showed a marked increase in the similarity of behavioral encoding across tasks (Figure 6D). This suggests that in networks with lower computational capacity, animals must rely on a smaller, shared behavioral representation across tasks.

3.4 NEURAL POPULATION DYNAMICS ARE SYNCHRONIZED WITH BEHAVIOR

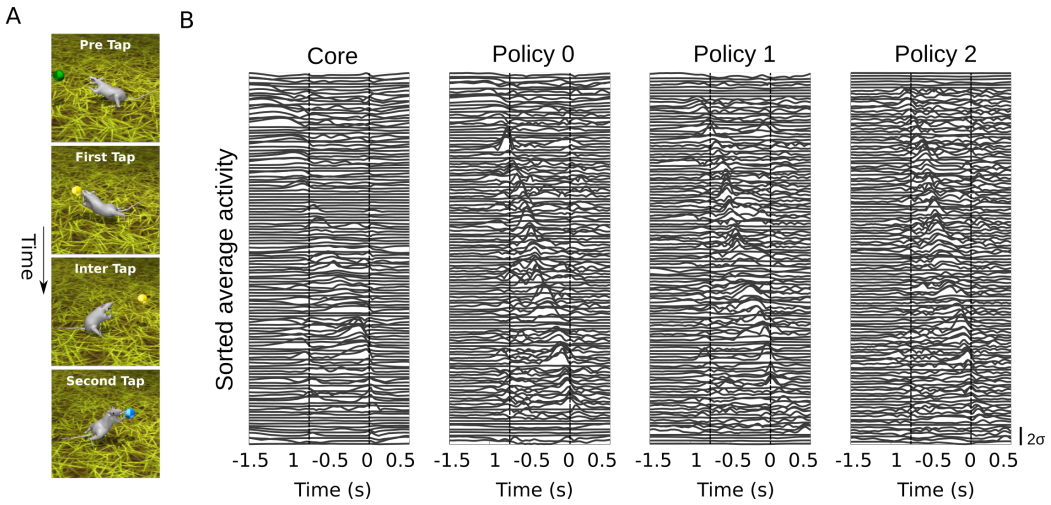

Figure 7: Neurons in core and policy networks show sequential activity during stereotyped behavior. (A) Example video stills showing the virtual rodent engaged in the two-tap task (B) Average absolute z-scored activity traces of all 128 neurons in each layer during performance of the two-tap sequence. Traces are sorted by the time of peak average firing rate. Dashed lines indicate the times of first and second taps. Sequential neural activity is present during the two-tap sequence.

Representational similarity analysis revealed that on policy and core activity was structured into different subspaces for different behaviors. However RSA also revealed that these networks did not posess explicit low-level representations of forces or kinematics, suggesting networks used different mechanisms to directly generate behavior. Inspection of the activity patterns underlying individual behaviors in the two-tap task showed peak activity in core and policy units was sequentially organized (Figure 7), uniformly tiling time between both taps of the two-tap sequence. This sequential activation in policy networks was observed across tasks and behaviors, including during running (see video) where, consistent with policy networks encoding short-timescale kinematic features in a task-invariant manner, neural activity sequences were largely conserved across tasks (See Appendix A.5, Figure 10). Interestingly, we also observed that neural variability is reduced during the inter-

trial interval when more stereotyped behavior is demanded by the two-tap task, which may have bearing on classic results in motor control (See Appendix A.6, Figure 11).

To systematically examine the types of sequential or dynamical activity present in core and policy networks, we examined the dynamics of the networks' population vector over time. We first applied principal components analysis (PCA) to the activity during the performance of single tasks, and visualized the gradient of the population vector as a vector field. Figure 8A shows such a vector field representation of the first two principal components of the core and final policy layer during the two-tap task. We generated vector fields by discretizing the PC space into a two-dimensional grid and calculating the average neural activity gradient with respect to time for each bin. The vector fields showed strong signatures of rotational dynamics across all layers. To quantify the frequency of these dynamical patterns, we turned to jPCA, a dimensionality reduction method that extracts rotational dynamics in neural activity by fitting a rotational linear dynamical system to neural data (Churchland et al., 2012). The resulting jPCs form an orthonormal basis that spans the same space as the first six traditional PCs, while maximally emphasizing rotational dynamics. Figure 8B shows the vector fields of the first two jPC planes for the core and final policy layers along with their characteristic frequency. Consistent with our previous findings, jPC planes in the core have lower characteristic frequencies than those in policy layers across tasks (Figure 8C). The jPC planes also individually explained a large percentage of total neural variability (Figure 8D).

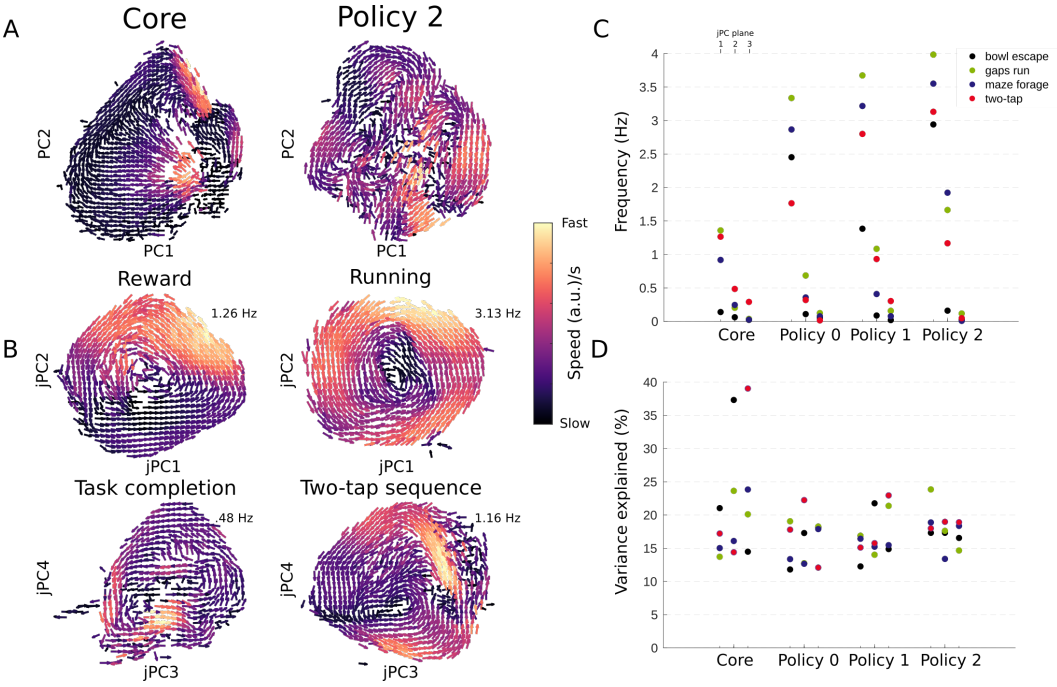

Figure 8: Latent network dynamics within tasks reflect rodent behavior on different timescales. (A) Vector field representation of the first two principal components of neural activity in the core and final policy layers during the two-tap task. PC spaces show signatures of rotational dynamics. (B) Vector field representation of first two jPC planes for the core and final policy layers during the two-tap task. Apparent rotations within the different planes are associated with behaviors and behavioral features of different timescales, labeled above. Columns denote layer (as in (A)), while rows denote jPC plane. (C) Characteristic frequency of rotations within each jPC plane. Groups of three points respectively indicate the first, second, and third jPC planes for a given layer. Rotations in the core are slower than those in the policy. (D) Variance explained by each jPC plane.

These rotational dynamics in the policy and core jPC planes were respectively associated with the production of behaviors and the reward structure of the task. For example, in the two-tap task, rotations in the fastest jPC plane in the core were concurrent with the approach to reward, while rotations in the second fastest jPC were concurrent with long timescale transitions between running to the orb

and performing the two-tap sequence. Similarly, the fastest jPC in policy layers was correlated with the phase of running, while the second fastest was correlated with the phase of the two-tap sequence (see video). This trend of core and policy neural dynamics respectively reflecting behavioral and task-related features was also present in other tasks. For example, in the maze forage task, the first two jPC planes in the core respectively correlated with reaching the target orb and discovering the location of new orbs, while those in the policy were correlated with low-level locomotor features such as running phase (see video). Along with our representational similarity analysis, these findings further support a model in which the core layer transforms sensory information into a sensorimotor contextual signal in a task-specific manner while the policy uses this signal and sensory inputs to generate appropriate behaviors in a more task-independent fashion. For a more complete set of behaviors with neural dynamics visualizations overlain, see Appendix A.7.

3.5 Neural perturbations corroborate distinct roles across layers

To causally demonstrate the differing roles of core and policy units in respectively encoding task-relevant features and movement, we performed perturbations of different neuronal subsets in the two-tap task. We identified two stereotyped behaviors (rears and spinning jumps) that were reliably used in two different seeds of the agent to reach the orb in the task. We ranked neurons according to the degree of modulation of their z-scored activity during the performance of these behaviors. We then inactivated subsets of neurons by clamping their activity to their mean values between the first and second taps and observed the effects of inactivation on trial success and behavior.

In both seeds analyzed, inactivation of policy units had a stronger effect on motor behavior than the inactivation of core units. For instance, in the two-tap task, ablation of 64 neurons in the final policy layer disrupts the performance of the spinning jump (Appendix A.8 Figure 12B video). In contrast, ablation of behavior-modulated core units did not prevent the production of the behavior, but mildly affected the way in which the behavior is directed toward objects in the environment. For example, ablation of a subset of core units during the performance of a spinning jump had a limited effect, but sometimes resulted in jumps that missed the target orbs (video; See Appendix A.8, Figure 12C).

We also performed a complementary perturbation aimed to elicit behaviors by overwriting the cell state of neurons in each layer with the time-varying average trajectory of neural activity measured during natural performance of a target behavior. The efficacy of stimulation was found to depend on the gross body posture and behavioral state of an animal, but was nevertheless successful in some cases. For example, during the two-tap sequence, we were able to elicit spinning movements common to searching behaviors in the forage task (video; See Appendix A.8, Figure 12D, E). The efficacy of this activation was more reliable in layers closer to the motor output (Figure 12D). In fact, activation of core units rarely elicited spins, but rather elicited sporadic dashes reminiscent of the searching strategy of many models during the forage task (video).

4 Discussion

For many computational neuroscientists and artificial intelligence researchers, an aim is to reverse-engineer the nervous system at an appropriate level of abstraction. In the motor system, such an effort requires that we build embodied models of animals equipped with artificial nervous systems capable of controlling their synthetic bodies across a range of behavior. Here we introduced a virtual rodent capable of performing a variety of complex locomotor behaviors to solve multiple tasks using a single policy. We then used this virtual nervous system to study principles of the neural control of movement across contexts and described several commonalities between the neural activity of artificial control and previous descriptions of biological control.

A key advantage of this approach relative to experimental approaches in neuroscience is that we can fully observe sensory inputs, neural activity, and behavior, facilitating more comprehensive testing of theories related to how behavior can be generated. Furthermore, we have complete knowledge of the connectivity, sources of variance, and training objectives of each component of the model, providing a rare ground truth to test the validity of our neural analyses. With these advantages in mind, we evaluated our analyses based on their capacity to both describe the algorithms and representations employed by the virtual rodent and recapitulate the known functional objectives underlying its creation without prior knowledge.

To this end, our description of core and policy as respectively representing value and motor production is consistent with the model's actor-critic training objectives. But beyond validation, our analyses provide several insights into how these objectives are reached. RSA revealed that the cell activity of core and policy layers had greater similarity with behavioral and postural features than with short-timescale actuators. This suggests that the representation of behavior is useful in the moment-to-moment production of motor actions in artificial control, a model that has been previously proposed in biological action selection and motor control (Mink, 1996; Graziano, 2006). These behavioral representations were more consistent across tasks in the policy than in the core, suggesting that task context and value activity in the core engaged task-specific behavioral strategies through the reuse of shared motor activity in the policy.

Our analysis of neural dynamics suggests that these reused motor activity patterns are often organized as sequences. Specifically, the activity of policy units uniformly tiles time in the production of several stereotyped behaviors like running, jumping, spinning, and the two-tap sequence. This finding is consistent with reports linking sequential neural activity to the production of stereotyped motor and task-oriented behavior in rodents (Berke et al., 2009; Rueda-Orozco & Robbe, 2015), including during task delay periods (Akhlaghpour et al., 2016), as well as in singing birds (Albert & Margoliash, 1996; Hahnloser et al., 2002). Similarly, by relating rotational dynamics to the virtual rodent's behavior, we found that different behaviors were seemingly associated with distinct rotations in neural activity space that evolved at different timescales. These findings are consistent with a hierarchical control scheme in which the sequences and dynamics in the policy that mediate the generation of reused behaviors are modulated and patterned by sensorimotor signals from the core.

This work represents an early step toward the constructive modeling of embodied control for the purpose of understanding the neural mechanisms behind the generation of behavior. Incrementally and judiciously increasing the realism of the model's embodiment, behavioral repertoire, and neural architecture is a natural path for future research. Our virtual rodent possesses far fewer actuators and touch sensors than a real rodent, uses a vastly different sense of vision, and lacks integration with olfactory, auditory, and whisker-based sensation (see Zhuang et al., 2017). While the virtual rodent is capable of locomotor behaviors, an increased diversity of tasks involving decision making, memory-based navigation, and working memory could give insight into "cognitive" behaviors of which rodents are capable. Furthermore, biologically-inspired design of neural architectures and training procedures should facilitate comparisons to real neural recordings and manipulations. We expect that this comparison will help isolate residual elements of animal behavior generation that are poorly captured by current models of motor control, and encourage the development of artificial neural architectures that can produce increasingly realistic behavior.

AUTHOR CONTRIBUTIONS

Contributions masked for anonymity.

ACKNOWLEDGMENTS

The rodent skeleton that we used as a reference for designing the virtual rodent body was purchased from leo3Dmodels on TurboSquid. Other acknowledgements masked for anonymity.

REFERENCES

Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018.

Hessameddin Akhlaghpour, Joost Wiskerke, Jung Yoon Choi, Joshua P Taliaferro, Jennifer Au, and Ilana B Witten. Dissociated sequential activity and stimulus encoding in the dorsomedial striatum during spatial working memory. *Elife*, 5:e19507, 2016.

C Yu Albert and Daniel Margoliash. Temporal hierarchical control of singing in birds. *Science*, 273 (5283):1871–1875, 1996.

Anonymous. V-MPO: On-policy maximum a posteriori policy optimization for discrete and continuous control. *Under Review for International Conference on Learning Representations*, 2019.

Andrea Banino, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J Chadwick, Thomas Degris, Joseph Modayil, et al. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429, 2018.

Joshua D Berke, Jason T Breck, and Howard Eichenbaum. Striatal versus hippocampal representations during win-stay maze performance. *Journal of neurophysiology*, 101(3):1575–1587, 2009.

Gordon J Berman, Daniel M Choi, William Bialek, and Joshua W Shaevitz. Mapping the stereotyped behaviour of freely moving fruit flies. *Journal of The Royal Society Interface*, 11(99):20140672, 2014.

Hillel J Chiel and Randall D Beer. The brain has a body: adaptive behavior emerges from interactions of nervous system, body and environment. *Trends in neurosciences*, 20(12):553–557, 1997.

Mark M Churchland, M Yu Byron, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, et al. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature neuroscience*, 13(3):369, 2010.

Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487 (7405):51, 2012.

Christopher J Cueva and Xue-Xin Wei. Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. *arXiv preprint arXiv:1803.07770*, 2018.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymyr Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. IMPALA: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International Conference on Machine Learning*, pp. 1406–1415, 2018.

Michael Graziano. The organization of behavioral repertoire in motor cortex. *Annu. Rev. Neurosci.*, 29:105–134, 2006.

Sten Grillner, Alexander Kozlov, Paolo Dario, Cesare Stefanini, Arianna Menciassi, Anders Lansner, and Jeanette Hellgren Kotaleski. Modeling a vertebrate motor system: pattern generation, steering and control of body orientation. *Progress in brain research*, 165:221–234, 2007.

Richard HR Hahnloser, Alexay A Kozhevnikov, and Michale S Fee. An ultra-sparse code underliesthe generation of neural sequences in a songbird. *Nature*, 419(6902):65, 2002.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Nicolas Heess, TB Dhruva, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, Ali Eslami, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*, 2017.

Norman C Heglund and C Richard Taylor. Speed, stride frequency and energy cost per stride: how do they change with body size and gait? *Journal of Experimental Biology*, 138(1):301–318, 1988.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Auke Jan Ijspeert, Alessandro Crespi, Dimitri Ryczko, and Jean-Marie Cabelguen. From swimming to walking with a salamander robot driven by a spinal cord model. *Science*, 315(5817):1416–1420, 2007.

Eric Jonas and Konrad Paul Kording. Could a neuroscientist understand a microprocessor? *PLoS computational biology*, 13(1):e1005268, 2017.

John F Kalaska. From intention to action: motor cortex and the control of reaching movements. In *Progress in Motor Control*, pp. 139–178. Springer, 2009.

Kohitij Kar, Jonas Kubilius, Kailyn Schmidt, Elias B Issa, and James J DiCarlo. Evidence that recurrent circuits are critical to the ventral streams execution of core object recognition behavior. *Nature neuroscience*, 22(6):974, 2019.

Risa Kawai, Timothy Markman, Rajesh Poddar, Raymond Ko, Antoniu L Fantana, Ashesh K Dhawale, Adam R Kampff, and Bence P Ölveczky. Motor cortex is required for learning but not for executing a motor skill. *Neuron*, 86(3):800–812, 2015.

Alexander JE Kell, Daniel LK Yamins, Erica N Shook, Sam V Norman-Haignere, and Josh H McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644, 2018.

Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLoS computational biology*, 10(11):e1003915, 2014.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. Similarity of neural network representations revisited. *CoRR*, abs/1905.00414, 2019.

Nikolaus Kriegeskorte and Jörn Diedrichsen. Peeling the onion of brain representations. *Annual Review of Neuroscience*, 42(1):407–432, 2019.

Nikolaus Kriegeskorte, Marieke Mur, and Peter A Bandettini. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4, 2008.

Yuri Lazebnik. Can a biologist fix a radio? Or, what I learned while studying apoptosis. *Cancer cell*, 2(3):179–182, 2002.

Timothy P Lillicrap and Stephen H Scott. Preference distributions of primary motor cortex neurons reflect control solutions optimized for limb biomechanics. *Neuron*, 77(1):168–179, 2013.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Josh Merel, Arun Ahuja, Vu Pham, Saran Tunyasuvunakool, Siqi Liu, Dhruva Tirumala, Nicolas Heess, and Greg Wayne. Hierarchical visuomotor control of humanoids. In *International Conference on Learning Representations*, 2019a.

Josh Merel, Leonard Hasenclever, Alexandre Galashov, Arun Ahuja, Vu Pham, Greg Wayne, Yee Whye Teh, and Nicolas Heess. Neural probabilistic motor primitives for humanoid control. In *International Conference on Learning Representations*, 2019b.

Jonathan A Michaels, Stefan Schaffelhofer, Andres Agudelo-Toro, and Hansjörg Scherberger. A neural network model of flexible grasp movement generation. *bioRxiv*, pp. 742189, 2019.

Jonathan W Mink. The basal ganglia: focused selection and inhibition of competing motor programs. *Progress in neurobiology*, 50(4):381–425, 1996.

John A Nyakatura, Kamilo Melo, Tomislav Horvat, Kostas Karakasiliotis, Vivian R Allen, Amir Andikfar, Emanuel Andrada, Patrick Arnold, Jonas Lauströer, John R Hutchinson, et al. Reverse-engineering the locomotion of a stem amniote. *Nature*, 565(7739):351, 2019.

Xue Bin Peng, Glen Berseth, and Michiel Van de Panne. Terrain-adaptive locomotion skills using deep reinforcement learning. *ACM Transactions on Graphics (TOG)*, 35(4):81, 2016.

Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel Van De Panne. DeepLoco: Dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Transactions on Graphics (TOG)*, 36(4):41, 2017.

Pavan Ramdya, Robin Thandiackal, Raphael Cherney, Thibault Asselborn, Richard Benton, Auke Jan Ijspeert, and Dario Floreano. Climbing favours the tripod gait over alternative faster insect gaits. *Nature communications*, 8:14494, 2017.

Alfonso Renart and Christian K Machens. Variability in neural activity and behavior. *Current opinion in neurobiology*, 25:211–220, 2014.

Pavel E Rueda-Orozco and David Robbe. The striatum multiplexes contextual and kinematic information to constrain motor habits execution. *Nature neuroscience*, 18(3):453, 2015.

Simon Schmitt, Jonathan J Hudson, Augustin Zidek, Simon Osindero, Carl Doersch, Wojciech M Czarnecki, Joel Z Leibo, Heinrich Kuttler, Andrew Zisserman, Karen Simonyan, et al. Kickstarting deep reinforcement learning. *arXiv preprint arXiv:1803.03835*, 2018.

H Francis Song, Guangyu R Yang, and Xiao-Jing Wang. Reward-based training of recurrent neural networks for cognitive and value-based tasks. *Elife*, 6:e21492, 2017.

Greg J Stephens, Bethany Johnson-Kerner, William Bialek, and William S Ryu. Dimensionality and dynamics in the behavior of c. elegans. *PLoS computational biology*, 4(4):e1000028, 2008.

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

Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. DeepMind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014.

Guangyu Robert Yang, Madhura R Joglekar, H Francis Song, William T Newsome, and Xiao-Jing Wang. Task representations in neural networks trained to perform many cognitive tasks. *Nature neuroscience*, 22(2):297, 2019.

Chengxu Zhuang, Jonas Kubilius, Mitra JZ Hartmann, and Daniel L Yamins. Toward goal-driven neural network models for the rodent whisker-trigeminal system. In *Advances in Neural Information Processing Systems*, pp. 2555–2565, 2017.

A APPENDIX

A.1 RAT MEASUREMENTS

In order to construct the virtual rodent model, we obtained the mass and lengths of the largest body segments that influence the physical properties of the virtual rodent. First, we dissected cadavers of two female Long-Evans rats, and measured the mass of relevant limb segments and organs. Next, we measured the lengths of body segments over the skin of animals anesthetized with 2% v/v isoflurane anesthesia in oxygen. We confirmed that these skin based measurements approximated bone lengths by measuring bone lengths in a third cadaver. The care and experimental manipulation of all animals were reviewed and approved by the appropriate Institutional Animal Care and Use Committee.

	Animal (#)		
	63	**64**	
Body part	**Mass (g)**		**Average mass (g)**
Hindlimb L	21	26	23.5
Hindlimb R	21	26	23.5
Tail	8	10	9
Forelimb R	11	14	12.5
Forelimb L	12	13	12.5
Full torso	176	187	181.5
Head	26	26	26
Upper torso	78	71	74.5
Lower torso	98	114	106
Torso without organs	54	58	56
Intestines and stomach	22	32	27
Liver	26	17	21.5
Pelvis and kidneys	74	80	77
Jaw	2.43	4.70	3.57
Skull	23	21	22
Tail (base to mid)	5.92	7.20	6.56
Tail (mid to tip)	1.78	2.30	2.04
Scapula L	3.19	4.70	3.94
Humerus L	6.25	4.70	5.48
Radius/ulna L	2.61	2.8	2.70
Forepaw L	0.53	0.5	0.52
Scapula R	2.23	3.9	3.07
Humerus R	6.08	6.7	6.39
Radius/ulna R	2.17	3.3	2.74
Forepaw R	0.53	0.5	0.52
Hindpaw L	1.66	1.7	1.68
Tibia L	9	9	9
Femur L	13	16	14.5
Hindpaw R	1.81	1.6	1.71
Tibia R	5	6	5.5
Femur R	13	18	15.5
Total	281	301	291

Table 1: Before weighing, limb segments were divided at their respective joints. Mass of all segments includes all bones, skin, muscle, fascia and adipose layers. L and R refer to the left and right sides of the animal. Precision of measurements listed without decimal places is ± 0.5g

	Animal (#)							
	48	**62**	**55**	**56**	**64**	**63**	**62***	**Average ± std**
Age (days)	382	82	330	330	83	83	83	
Mass (g)	325	273	389	348	283	269	273	309 ± 47
Body part				**Length (mm)**				
Ankle to claw L	40.2	39.5	39.7	37.8	39.9	41.5	39.8	39.8 ± 1.1
Ankle to toe L	38.4	38.12	37.7	35.6	36.6	39.3	38	37.7 ± 1.2
Ankle to pad L	23.4	22.2	23	22.12	22.5	23.3	6.4	20.4 ± 6.2
Ankle to claw R		38.2	40.4	38.3	39.3	39.6	38.3	39.0 ± 0.9
Ankle to toe R		37	38.7	36.3	37.7	38.6	36.2	37.4 ± 1.1
Ankle to pad R		22.4	23.3	21.9	21.8	23.1	24.1	22.8 ± 0.9
Tibia L	50	36.3	38.5	49.2	35.8	38.7	34.1	40.4 ± 6.5
Femur L	44.5	31.6	32.1	37.9	33.4	35.35	32.4	35.3 ± 4.6
Tibia R		36.7	39.1	37.9	35.1	38.4	36.18	37.2 ± 1.5
Femur R		32.9	32.1	38.7	31.9	32.1	32.6	33.4 ± 2.6
Pelvis	25.8	32	31.7	30.2	26.7	27.2		28.9 ± 2.7
Wrist to claw L	15	18.8	17.6	18.6	16	19.02	19.2	17.7 ± 1.6
Wrist to finger L		16	15.8	17.4	15.5	17.07	17.6	16.6 ± 0.9
Wrist to pad L		6	6.4	8.34	4.9	6.1	6.4	6.4 ± 1.1
Wrist to olecranon L	29.1	34	32.5	31.7	33.9	32.1	29.9	31.9 ± 1.9
Humerus L	31.9	29.52	31	28.2	27	31.2	25.4	29.2 ± 2.4
Scapula L	22.7	24	26.4	29.3	25.9	29.1	26.2	26.2 ± 2.4
Wrist to claw R		16.8	17	17.8	15.9	16.3	18.1	17.0 ± 0.8
Wrist to finger R		14.1	13	15.6	15.6	15.3	16.9	15.1 ± 1.4
Wrist to pad R		5.6	5.8	6.55	5.2	5	5.8	5.7 ± 0.5
Wrist to olecranon R		30.6	33.5	31.2	30.4	31.8	29.9	31.2 ± 1.3
Humerus R		28.2	33.5	28.8	25	28.2	25.2	28.1 ± 3.1
Scapula R		23.8	29.5	25.9	26.2	28.8	24.4	26.4 ± 2.3
Headcap width	39							39
Headcap length	30							30
Skull width	38.8	23.35	23	21.8	22.8	23.9	22.2	25.1 ± 6.1
Skull length	57	51.1	61	56.48	53.16	58.13	48	55.0 ± 4.5
Skull height					21.59	21.5	21	21.4 ± 0.3
Head to thoracic		48.6	71.4	68.68	65	60.4	71.2	64.2 ± 8.7
Thoracic to sacral		73.1	73.6	62.9	65.04	64.7	68.8	68.0 ± 4.6
Head to sacral	145	126	145.5	127.05	127.2	123.7	140.9	133.6 ± 9.7
Head width	53.4							53.4
Ear	18	17.55	19.3	17.9	19.2	18.8		18.5 ± 0.7
Eye	7.2	8.25	8.6	8.8	8.2	8.3		8.2 ± 0.6

Table 2: Length measurements of limb segments used to construct the virtual rodent model from 7 female Long-Evans rats. Measurements were performed using calipers either over the skin or over dissected bones (*). Thoracic and sacral refer to vertebral segments. L and R refer to the left and right sides of the animal's body.

A.2 BEHAVIORAL ANALYSIS

We generated behavioral features describing whole-body pose and kinematics of the virtual rodent on fast, intermediate, and slow temporal scales. To describe the whole-body pose, we took the top 15 principal components of the virtual rodent's joint angles and joint positions to yield two 15 dimensional sets of eigenpostures (Stephens et al., 2008). We combined these into a 30 dimensional set of postural features. To describe the animal's whole-body kinematics, we computed the continuous wavelet transform of each eigenposture using a Morlet wavelet spanning 25 frequencies. This yielded a 375 dimensional time-frequency representation of the underlying kinematics. We then computed the top 15 principal components of each 375 dimensional time-frequency representation. This yielded a 30 dimensional representational description of the animal's behavioral dynamics. To facilitate comparison of organismal behavior to neural representations on different timescales, we generated three sets of Morlet wavelet frequencies on 1 to 25 Hz (intermediate), 0.3 to 5 Hz (slow) or 5-25 Hz (fast) timescales. In separate work, we have found that combining postural and dynamical information improves separation of animal behaviors in behavioral embeddings. Therefore, we combined postural and dynamical features, the later on intermediate timescales, to yield a 60 dimensional set of behavioral features that we used to map the animal's behavior using tSNE (Figure 4C) (Berman et al., 2014). tSNEs were made using the Barnes-Hut approximation with a perplexity of 30.

A.3 POWER SPECTRAL DENSITY OF CORE AND POLICY LAYERS

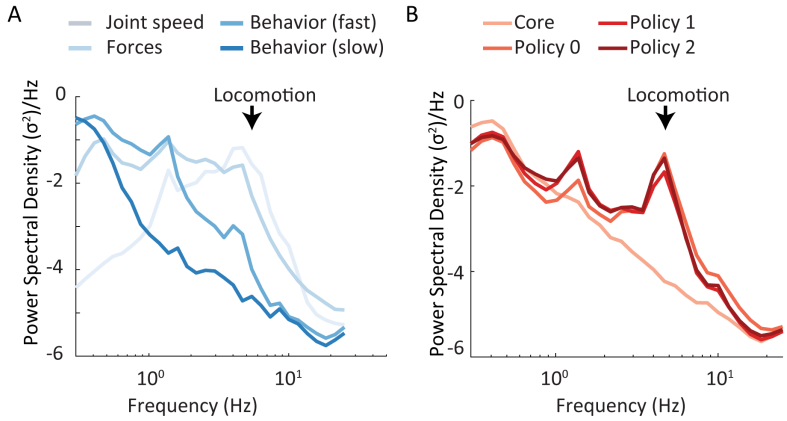

Figure 9: Power spectral density estimates of four different network layers, computed by averaging the spectral density of the top ten principal components of each matrix of activations, weighted by the variance they explain. Notice that policy layers have more power in high frequency bands than core layers. Arrows mark peaks in the power spectra corresponding to locomotion. Notably, the 4-5 Hz frequency of galloping in the virtual rat matches that measured in laboratory rats (Heglund & Taylor, 1988). Power spectral density was computed using Welch's method using a 10 s window size and 5 s overlap.

A.4 REPRESENTATIONAL SIMILARITY ANALYSIS

We used representational similarity analysis to compare the population representations across network layers and to different subsets of behavioral features. Representational similarity analysis has in the past been used to compare neural population responses in tasks where behavioral stimuli are discrete, for instance corpuses of objects or faces (Kriegeskorte et al., 2008; Kriegeskorte & Diedrichsen, 2019). A challenge in scaling such approaches to neural analysis in the context of behavior is that behavior unfolds continuously in time. It is thus *a priori* unclear how to discretize behavior into discrete chunks in which to compare representations.

Formally, we defined a set of eight different behavioral feature sets $B_{i=1...8}$ describing different aspects of the behavior of the animal on different timescales. These included instantaneous features such as joint angles, the angular speed of the joint angles, the eigenpostures of the animal, and the actuator forces, behavioral kinematics consisting of the wavelet transform of the eigenpostures computed on slow, fast, and intermediate timescales, and behavioral features, which consisted of both behavioral kinematics and the eigenposture features. Each feature set is a matrix $B_i \in \mathbb{R}^{Mxq_i}$ where M is the number of timepoints in the experiment and q_i is the number of features in the set. We discretized each set B_i using k-means clustering with $k = 50$ to yield a partition of the timepoints in the experiment P_i.

Using the discretization defined in P_i, we can perform representational simirity analysis to compare the structure of population responses across neural network layers L_m and L_n or between a given network layer and features of the behavior B_i. Following notation in (Kornblith et al., 2019) we let $X \in \mathbb{R}^{kxp}$ be a matrix of population responses across p neurons and the k behavioral categories in p_{B_i}. We let $Y \in \mathbb{R}^{nxq}$ be either the matrix of population responses from q neurons in a distinct network layer, or a set of q features describing the behavior of the animal in the feature set B_j. It is not necessary that the behavioral feature being compared, j, is the same as used to create the discretization partition, i, but we do so in all cases here.

After computing the response matricies in a given partition, we compared the representational structure of the matricies XX^T and YY^T. To do so, we compute the similarity between these matricies using the linear Centered Kernel Alignment index, which invariance under orthonormal rotations of the population activity. Following (Kornblith et al., 2019), for the CKA coeffient is:

$$CKA(XX^T, YY^T) = \frac{\|XY^T\|_F}{\|XX^T\|_F \|YY^T\|_F} \tag{1}$$

Where $\| \cdot \|_F$ is the Frobenius norm. For centered X and Y, the numerator is equivalent to the dot-product between the vectorized responses $\|XY^T\|^F = \langle \text{vec}(XX^T), \text{vec}(YY^T) \rangle$.

For a given network layer L_m, and a behavioral partition P_i, we can denote $XX^T = D_{P_i}^{L_m} = D_i^m$. Similarly, for a given feature set B_i, let $D_{P_i}^{B_i} = D_i^i$. Thus we are interested in characterizing both

$$CKA\left(D_i^m, D_i^n\right) \tag{2}$$

and

$$CKA\left(D_i^m, D_i^i\right). \tag{3}$$

An additional challenge comes when restricting this analysis to comparing the neural representations of behavioral across different tasks T_a, T_b, where not all behaviors are necessarily represented in each task. To do so, we denote $B_i(T_a)$ to be the set of behavioral clusters observed in task T_a, and $B_i^{T_a T_b} = B_i(T_a) \cap B_i(T_a)$ to be the set of behaviors used in each of the two tasks. We can then define a restricted partition of timepoints for each task $P_i^{T_a,T_b}$ or $P_i^{T_b,T_a}$ that includes only these behaviors, and compute the representational similarity between the same layer across tasks:

$$CKA\left(D_{i,T_a}^m, D_{i,T_b}^{L_m}\right). \tag{4}$$

We have presented a means of performing representational similarity analysis across continuous time domains, where the natural units of discretization are unclear and likely multi-fold. While we focused on analyzing responses on the population level, it is likely that different subspaces of the population may encode information about distinct behavioral features at different timescales, which is still an emerging domain in representational similarity analysis.

A.5 COMPARISON FOR RUNNING NEURAL POPULATION ACTIVITY ACROSS TASKS

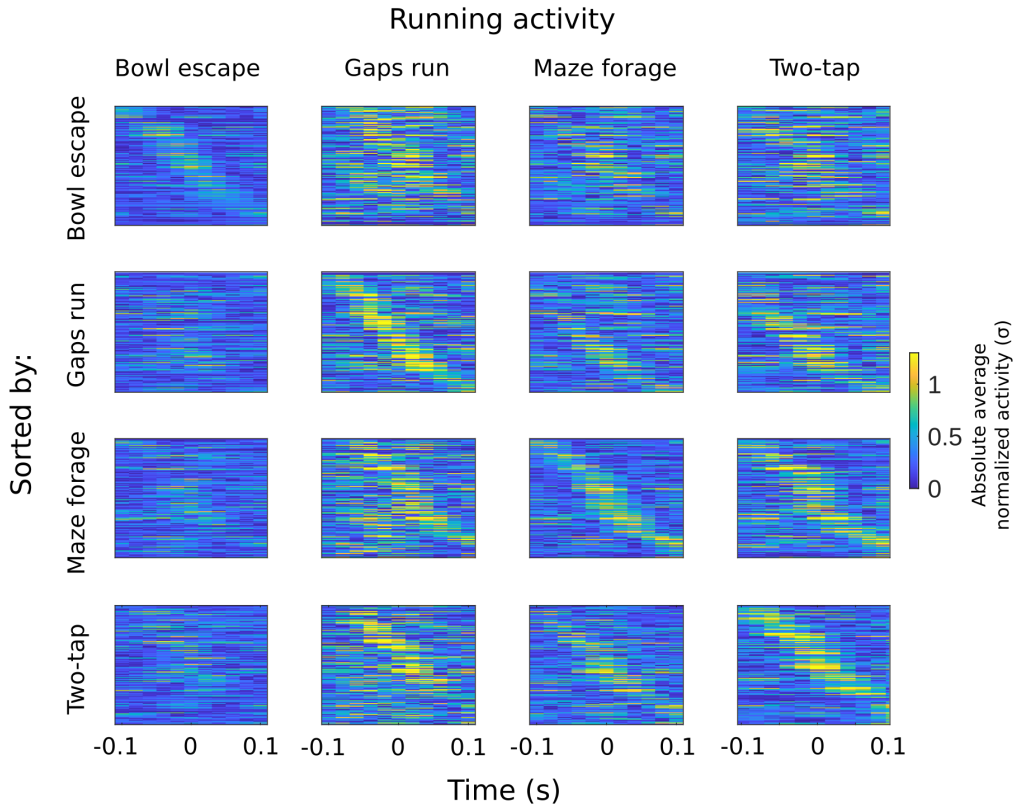

Figure 10: Average activity in the final policy layer during running cycles across different tasks. In each heatmap, rows correspond to the average absolute zscored activity for individual neurons, while columns denote time relative to the mid stance of the running phase. Across heatmaps, neurons are sorted by the time of peak activity in the tasks denoted on the left, such that each column of heatmaps contains the same average activity information with rearranged rows. Aligned running bouts were acquired by manually segmenting the the principal component space of policy 2 activity to find instances of mid-stance running and analyzing the surrounding 200 ms.

A.6 STEREOTYPED BEHAVIOR INITIATION AND NEURAL VARIABILITY

During the execution of stereotyped behaviors, neural variability is reduced (Figure 11). Recall that in our setting, neurons have no intrinsic noise, but inherit motor noise through observations of the state (i.e. via sensory reafference). This effect loosely resembles, and perhaps informs one line of interpretation of the widely reported phenomenon of neural variability reducing with stimulus or task onset (Churchland et al., 2010). Our reproduction of this effect, which simply emerges from training, suggests that variance modulation may partly arise from moments in a task that benefit from increased behavioral precision (Renart & Machens, 2014).

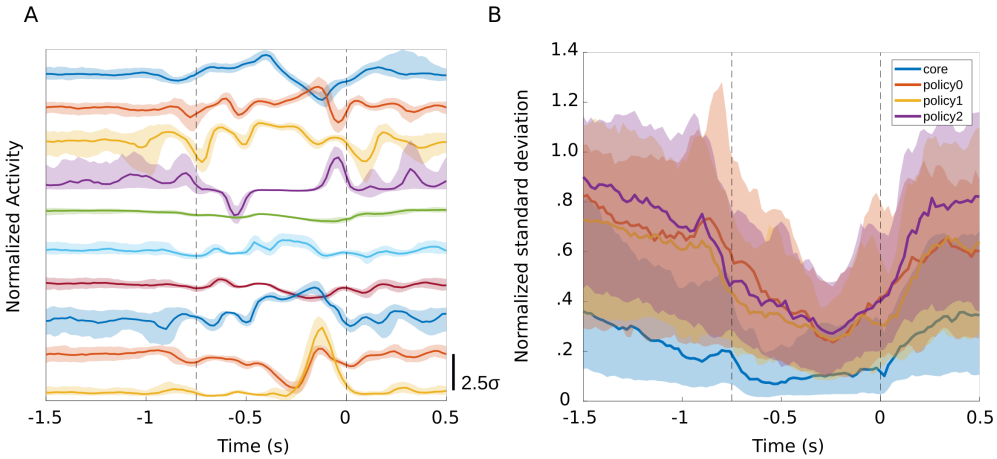

Figure 11: Quantification of neural variability in inter-tap interval of two-tap task relative to the second tap. (A) Example normalized activity traces of ten randomly selected neurons in the final policy layer. Lines indicate mean normalized activity whiles shaded regions range from the 20th percentile to the 80th percentile. Dashed lines indicate the times of first and second taps. (B) Standard deviation of normalized activity across all neurons in the final policy layer as a function of time relative to the second tap. Lines indicate the mean standard deviation while shaded regions range from the 20th percentile to the 80th percentile. Observe that variability is reduced during the two-tap interval.

A.7 NEURAL DYNAMICS VISUALIZED DURING TASK BEHAVIOR

For completeness, we provide links to videos of a few variants of neural dynamics for each task.

Network	Visualization	Task (link)
1-layer policy	PCA	gaps
	PCA	forage
	PCA	escape
	PCA	two-tap
3-layer policy	PCA	gaps
	PCA	forage
	PCA	escape
	PCA	two-tap
3-layer policy	jPCA	gaps
	jPCA	forage
	jPCA	escape
	jPCA	two-tap

Table 3: Links to representative visualizations of neural dynamics and behavior

A.8 PERTURBATION RESULTS

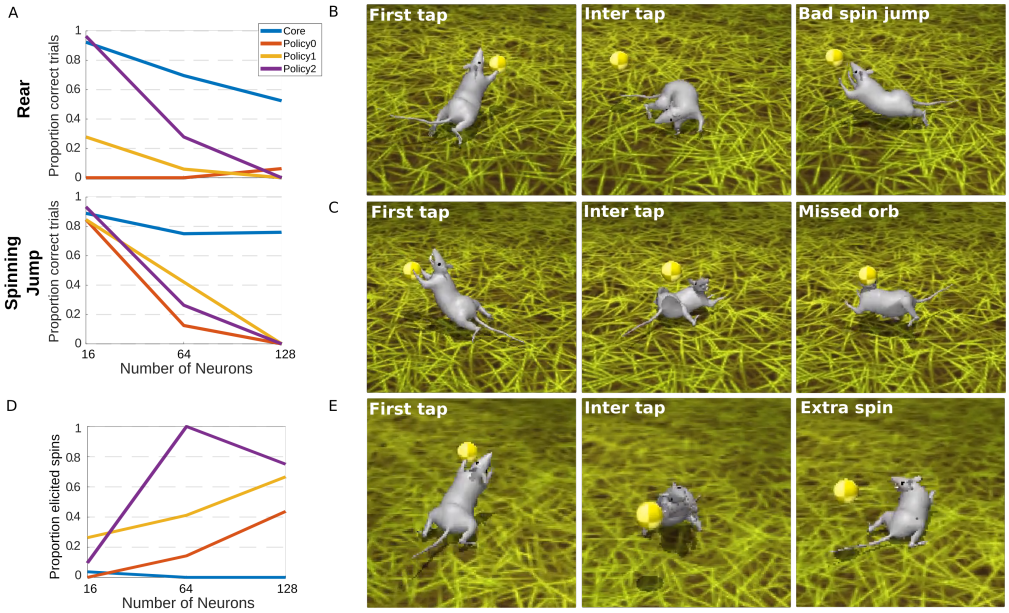

Figure 12: Causal manipulations reveal distinct roles for core and policy layers in the production of behavior. (A) Two-tap accuracy during the inactivation of units modulated by idiosyncratic behaviors within the two-tap sequence. Core inactivation has a weaker negative effect on trial success than policy inactivation for several levels of inactivation. (B) Representative example of a failed trial during core inactivation in a model that performs a spinning jump during the two-tap sequence. The model is still able to perform the spinning jump behavior, but misses the orb. (C) Representative example of a failed trial during inactivation of the final policy layer in a model that performs a spinning jump during the two-tap sequence. The model is incapable of producing the spinning jump behavior while inactivated. (D) Proportion of attempts at stimulation that successfully elicited spin behavior during the two-tap task. The efficacy of this activation was more reliable in layers closer to the motor output. (E) Representative example of a single trial in which an extra spin occurs.