

## Neural Associative Skill Memories for Safer Robotics and Modeling Human Sensorimotor Repertoires

**Pranav Mahajan**

*pranav.mahajan@ndcn.ox.ac.uk*

**Mufeng Tang**

*mufeng.tang@ndcn.ox.ac.uk*

*Nuffield Department of Clinical Neurosciences, University of Oxford, OX3 9DU, UK*

**T. Ed Li**

*ed.li@yale.edu*

*Nuffield Department of Clinical Neurosciences, University of Oxford, Oxford OX3 9DU, UK; and Yale School of Medicine, Yale University, New Haven, CT 06510, USA*

**Ioannis Havoutis**

*ioannis@robots.ox.ac.uk*

*Department of Engineering Sciences, University of Oxford, Oxford OX3 9DU, UK; and Oxford Robotics Institute, University of Oxford, Oxford OX2 6NN, UK*

**Ben Seymour**

*ben.seymour@ndcn.ox.ac.uk*

*Nuffield Department of Clinical Neurosciences and Institute of Biomedical Engineering, University of Oxford, OX3 9DU, UK*

Modern robots face a challenge shared by biological systems: how to learn and adaptively express multiple sensorimotor skills. A key aspect of this is developing an internal model of expected sensorimotor experiences to detect and react to unexpected events, guiding self-preserving behaviors. Associative skill memories (ASMs) address this by linking movement primitives to sensory feedback, but existing implementations rely on hard-coded libraries of individual skills. A key unresolved problem is how a single neural network can learn a repertoire of skills while enabling integrated fault detection and context-aware execution. Here we introduce neural associative skill memories (neural ASMs), a framework that uses self-supervised temporal predictive coding to integrate skill learning and expression using biologically plausible local learning rules. Unlike traditional ASMs, which require explicit skill selection, neural ASMs implicitly recognize and express skills through contextual inference, enabling fault detection using “predictive surprise” across the

---

Pranav Mahajan is the corresponding author.

**entire learned repertoire. Compared to recurrent neural networks trained using backpropagation through time, our model achieves comparable qualitative performance in skill memory expression while using local learning rules and predicts a biologically relevant speed-versus-accuracy trade-off. By integrating fault detection, reactive control, and skill expression into a single energy-based architecture, neural ASMs contribute to safer, self-preserving robotics and provide a computational lens to study biological sensorimotor learning.**

## 1 Introduction

---

Animals spend their lives learning, storing, and refining a repertoire of sensorimotor skills, which they must not only express adaptively but also use to detect abnormalities. To recognize an abnormal event, an animal requires an understanding of what normal actions “feel” like. This corresponds to having a robust internal model of the body and its interactions with the world, that is, a generative model that predicts the normal sensory consequences of actions. When reality deviates from these predictions, the resulting prediction error, or predictive surprise, (Clark, 2013; Friston, 2005), serves as a crucial signal. From a biological standpoint, this signal can indicate a sensorimotor conflict or an external perturbation (Shadmehr et al., 2010), or even bodily harm, such as an injury (Seymour & Mancini, 2020; Walters et al., 2023). The brain’s capacity to detect such violations of its internal priors is therefore not a secondary feature but a core component of safe and self-preserving behavior.

Inspired by this principle, roboticists have sought to equip machines with analogous capabilities through associative skill memories (ASMs; Pastor et al., 2013, 2012). The core idea is to link motor commands, often represented by dynamic movement primitives (DMPs) (Ijspeert et al., 2013), with their expected sensory feedback. This allows a robot to detect faults, the engineering equivalent of injuries or perturbations, and react accordingly. However, traditional ASMs have a critical architectural limitation compared to their biological counterparts: they rely on a hard-coded, dictionary-like library of skills. Each skill is learned and stored as an independent module, requiring an external mechanism to explicitly select which one to execute. Consequently, fault detection is constrained to a single skill, failing to capture the integrated nature of biological motor repertoires.

This contrast between engineered libraries and integrated biological systems highlights a key unresolved question: How can a single, unified neural network learn a repertoire of sensorimotor skills in a biologically plausible manner? Such a system would need to solve two fundamental challenges that are handled seamlessly in the brain. First, it must be able to detect abnormalities with respect to all learned skills in the repertoire, without being explicitly told which skill is being performed. Second, it must be able to use

sensory cues to contextually infer which skill memory is most appropriate to express in different scenarios.

This article addresses this question by introducing neural associative skill memories (neural ASMs), a framework that uses a temporal predictive coding network to learn an embodied generative model of a robot's sensorimotor repertoire. We focus on how a sequential memory of sensorimotor observations can be learned from skill demonstrations, eschewing the separate problem of how optimal motor actions leading to these sequences are planned. Our central goal is to demonstrate how a single network, trained with self-supervised, local learning rules, builds on the original concepts of associative skill memories (Pastor et al., 2013, 2012). To this end, we provide basic demonstrations of three outcomes. First, the model performs fault detection across all learned skills in its repertoire. This is achieved by identifying abnormally high network energy, which is indicative of an out-of-distribution state with respect to the demonstration data. Second, it supports reactive correction by minimizing proprioceptive prediction errors. Third, it facilitates the expression of different skills through contextual inference from early-stage cues in a robotics simulation inspired by a human motor experiment (Sheahan et al., 2016). By doing so, we aim to provide a step toward safer, self-preserving robotics and offer a basic computational model for how the brain might learn, express, and monitor its own motor skills. Finally, this study explores the foundational capabilities of a novel learning method involving local learning and inference within a temporal predictive coding framework. We demonstrate these capabilities using streamlined simulations, specifically fault detection in pick-and-place movements and perturbation compensation in point-to-point movements.

## 2 Related Work

---

**2.1 Associative Skill Memories and Movement Primitives.** The foundation of our work lies in ASMs (Pastor et al., 2013, 2012), which extend dynamic movement primitives (DMPs; Ijspeert et al., 2013), a class of attractor-based models for generating stereotyped movements. The key idea of ASMs is to associate these motor primitives with their expected sensory feedback, enabling fault detection and reactive control. However, the standard ASM framework implements this using multiple handcrafted modules: an explicit skill library to store DMPs, a separate system to maintain sensory statistics for each movement, and a prediction module to find the closest match in the library (see Figure 1A). This modular, library-based architecture requires explicit skill selection and can only detect faults relative to one active skill at a time. Our work aims to overcome this limitation by learning a repertoire of skills within a single, unified neural network.

**2.2 Predictive Coding for Sensorimotor Learning.** Our model is built on the principles of predictive coding (Clark, 2013; Friston, 2005; Rao

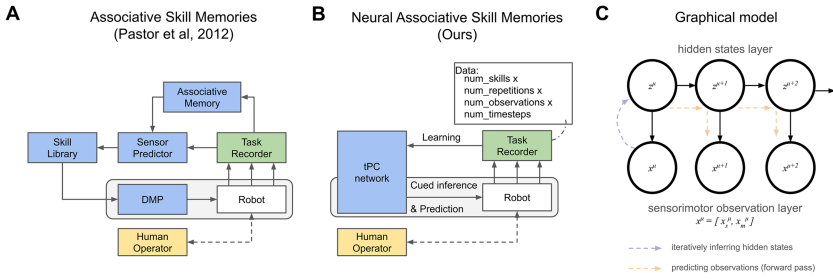


Figure 1: (A) Traditional ASMs (Pastor et al., 2012) use a modular, library-based architecture where each movement and primitive and associated sensory statistics are stored separately. (B) Our neural ASM approach replaces this with a single iPC network that learns multiple skill memories from demonstrations, where each demonstration is a time series of sensorimotor observations. (C) The underlying graphical model is a hidden Markov model (HMM), where hidden states ( $z$ ) capture the dynamics and inferred context and predict observations ( $x$ ). Cued inference (purple arrow) sets the initial state from early observations, which then allows for offline prediction (orange arrows) of the sequence. While prediction can be a simple forward pass in an HMM, more complex generative models would require iterative energy minimization.

& Ballard, 1999), an influential theory in neuroscience positing that the brain continuously generates predictions of sensory input and updates its internal model based on prediction errors. This self-supervised process has recently been proposed as a biologically plausible alternative to back-propagation (Song et al., 2024; Whittington & Bogacz, 2017, 2019) and has been applied to associative memory (Salvatori et al., 2021) and temporal sequence learning (Tang et al., 2024b). In robotics, predictive coding has been used for body state perception and multisensory integration from noisy information to filter internal state (Lanillos & Cheng, 2018). However, these models typically do not learn the dynamics of different movements, which is crucial for differentiating skills. Similarly, novelty detection (akin to abnormality detection in our work) has been demonstrated in static predictive coding networks by identifying inputs that generate high prediction errors (or energy; Li et al., 2024) but not applied to temporal sequence learning or to the domain of neurorobotics. Our work extends these ideas by using a temporal predictive coding network to learn a dynamic model of a full sensorimotor repertoire, using the model’s energy as a natural signal for fault detection across all learned skills.

### 2.3 Temporal Sequence Learning with Recurrent Architectures.

Learning sensorimotor sequences has long been a domain of recurrent neural networks (RNNs). Influential work has used RNNs trained with

backpropagation through time (BPTT) to generate complex sensorimotor sequences by learning associations between initial states and subsequent trajectories (Nishimoto et al., 2008; Nishimoto & Tani, 2004; Yamashita & Tani, 2008). A key insight from this research, particularly from Yamashita and Tani (2008), is the use of hierarchical structures with multiple timescales to capture complex temporal dependencies (MTRNNs). While powerful, these models typically rely on BPTT (Rumelhart et al., 1986), a nonlocal learning rule considered biologically implausible because it requires propagating error signals backward through the network's entire temporal history.

Our model, based on temporal predictive coding (tPC; Millidge et al., 2024; Tang et al., 2024b), offers an alternative. Temporal predictive coding networks learn temporal dependencies using local, Hebbian-like updates that operate only between adjacent layers and time steps. This locality, while biologically plausible, is equivalent to BPTT truncated to one time step (Tang et al., 2024a). A benefit of tPC networks is that they integrate models of temporal sequence learning in the rich field of Bayesian inference. This allows us to use the notions of "energy" of a model to formalize the concept of faults in this work. A key feature of our predictive coding framework is the iterative inference process used to converge on a hidden state at each time step. This process introduces an additional, faster timescale for online inference, distinct from the slower timescale of weight updates during learning. This mechanism shares some conceptual similarities with fast-weight RNNs (Ba et al., 2016) and error regression in MTRNNs (Ahmadi & Tani, 2017). It further provides a candidate neural process for motor preparation, allowing contextual cues to shape a subsequent motor plan. These kinds of iterative inference from early-stage contextual cues further align with the hypothesis of the role of motor preparatory activity in setting the initial state of a dynamical system rather than explicitly representing movement parameters (Churchland, Byron et al., 2006; Churchland, Cunningham et al., 2010; Churchland, Santhanam et al., 2006, Cisek, 2006; Fetz, 1992).

**2.4 Computational Models of Homeostasis and Injury-Related Behaviors.** The concept of fault detection in our model aligns with computational theories of interoception and homeostasis, the processes by which the brain senses, predicts, and regulates the body's internal state (Barrett, 2017; Seth, 2013). Our model adopts the view that the brain must infer its bodily state from multiple, often noisy, sensory signals (interoceptive, exteroceptive, and proprioceptive) to guide control (Seymour & Mancini, 2020). This contrasts with theories that assume direct access to (noise-free) internal states (Keramati & Gutkin, 2014) or rely purely on potentially noisy peripheral nociceptive signals to direct self-preserving behaviors (Walters et al., 2023). In this Bayesian inference-based view, the prediction errors arising from unexpected bodily signals drive an updated belief that the body is damaged, which in turn elicits protective behaviors (Mahajan et al., 2025; Seymour

et al., 2023). Our model's use of prediction errors to detect deviations from a learned "normal" repertoire provides an analogous mechanism.

While other computational models have explored goal-directed homeostatic and interoceptive control (Mahajan et al., 2025; Tschantz et al., 2022), simulating interesting behaviors like investigating one's injury to gain information about the injury state despite associated phasic pain (Mahajan et al., 2025), they typically rely on hand-crafted generative models or restrictive notions of homeostatic set points (Keramati & Gutkin, 2014). This leaves two questions unanswered: (1) How can these ideas be extended to more complex robotics tasks, and (2) How can these computational-level theories be implemented with biologically plausible learning rules? Our work addresses both points by demonstrating that a generative model can be learned directly from demonstrations using local learning rules, providing a framework for fault detection and reactive control in a neuro-robotic system while leaving goal-directed, self-preserving behaviors as future work.

### 3 Neural ASMs: A Theory Sketch

---

Neural associative skill memories (neural ASMs) replace the modular, library-based architecture of traditional ASMs with a single, unified network. This network is a temporal predictive coding (tPC) model (Millidge et al., 2024; Tang et al., 2024b) that learns a generative model of sensorimotor sequences from demonstrations (see Figure 1). The model has a hierarchical structure analogous to a hidden Markov model (HMM), where hidden states ( $z$ ) predict observations ( $x$ ) as well as their own future states. The sensory and motor observations are concatenated at the observation layer ( $x$ ), similar to previous work (Nishimoto et al., 2008; Yamashita & Tani, 2008). The entire system learns by minimizing a single energy function via local, Hebbian-like updates (see equations 6.1 to 6.4).

The model operates in two distinct phases: memorization and recall. During memorization, the network learns from demonstrated sensorimotor sequences. For each step in a sequence, it infers a hidden state and updates its weights, creating an attractor in its energy landscape that corresponds to that sequence (Salvatori et al., 2021; Tang et al., 2024b). This process allows the model to store multiple skill memories within a single set of weights.

During recall, the network retrieves a stored sequence from a partial cue. This begins with cued inference, where initial sensorimotor observations (e.g., the robot's starting position and sensory state) are used to infer the corresponding hidden state (see Figure 1C, purple arrow). This inference process is analogous to motor preparation, where context sets the initial conditions for a movement (Churchland et al., 2010). Once the initial state is set, the network autonomously generates the predicted sensorimotor trajectory for the remainder of the skill (see Figure 1C, orange arrows). In this work, we use offline recall (Tang et al., 2024b), where the trajectory is generated ballistically after inferring the hidden state from a few early-stage

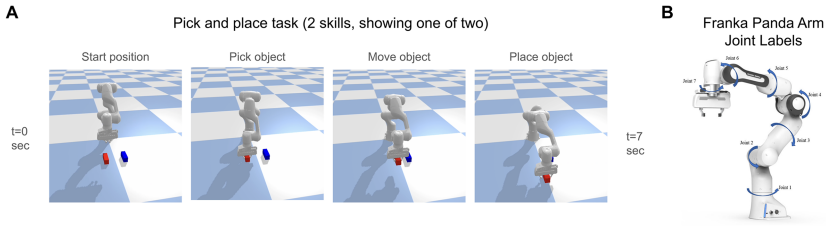


Figure 2: (A) Demonstration of neural ASMs learning two pick-and-place skills in simulation. The sensorimotor sequences in the data set used for learning from demonstrations are generated using predefined end-effector goals and use inverse kinematics to get joint angles. This is intended to be a proxy for teleoperation in simulation. (B) A schematic of Franka Panda arm joint labels. (Adapted from Rogel et al., 2022, under CC BY 4.0 license. A title line was added.)

contextual cues. Examples of such offline processes in biology include sequences guided from working memory (Mizes et al., 2023, 2024) or mental simulation, where trajectories are generated without producing actual movements (Jeannerod, 1994; Tani, 1996; Yamashita & Tani, 2008). An alternative, online recall, would involve continuous inference using incoming sensory data at each step, allowing for dynamic switching between learned skills, but this is not explored here.

Finally, the predicted sequence of motor commands (e.g., joint angles) from the neural ASM serves as a high-level dynamical policy. This policy provides a reference trajectory to a low-level controller, which executes the movement and can make minor reactive adjustments online (see Schaal et al., 2007; see Figure 7 in appendix A).

## 4 Results

**4.1 Memorized Skills Are Useful in Fault Detection and Simple Reactive Correction.** Having introduced neural ASMs as a viable alternative to ASMs, we now demonstrate their core functionalities: fault detection and reactive fault correction in a simple simulation. We train a model to memorize two pick-and-place skills from demonstrations (see Figure 2A) and then test its ability to detect and react to simulated faults. The model uses implicit skill recognition during cued inference to retrieve the correct skill from initial observations using offline recall, unlike traditional ASMs, which require explicit selection of a movement primitive (Pastor et al., 2013, 2012).

We first demonstrate fault detection qualitatively by measuring the network’s energy (i.e., sum of squared prediction errors). Figure 3 shows two examples of a joint-locking fault. In a minor fault where joint 5 gets stuck (Figure 3A), the network’s energy increases abnormally, successfully detecting the fault (Figure 3B) using out-of-distribution detection on the

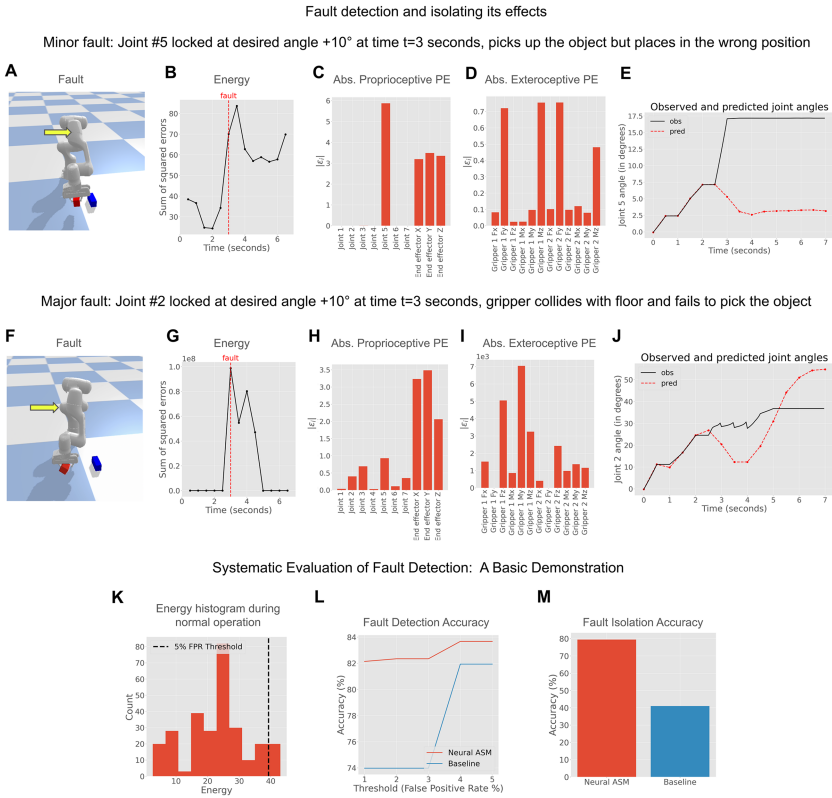


Figure 3: (A–E) Minor fault example with fault detection using energies and correct fault isolation using absolute prediction errors (denoted by abs. PE in the panels) along with the joint angle time series. (F–J). Major fault example with fault detection using energies and incorrect fault isolation using absolute prediction errors (denoted by abs. PE in the panels) along with the joint angle time series. The yellow arrows in panels A and F point to the joints where the respective minor and major faults took place. The  $x$ -axis in panels C and H labels the proprioceptive sensors for joint angles (1–7, see Figure 2B) and end-effector coordinates ( $X$ ,  $Y$ ,  $Z$ ). The  $x$ -axis in panels D and I labels the exteroceptive sensors measuring  $X$ ,  $Y$ , and  $Z$  components of forces and torques at the two grippers. The  $y$ -axis in panels C, D, H, and I represents the absolute prediction error (abs. PE). (K–M) A basic demonstration of the systematic evaluation of fault detection and isolation.

network’s energy. The individual prediction errors also correctly identify joint 5 as the primary location of the fault (Figures 3C–3E). In a more severe fault where joint 2 gets stuck and causes a collision (Figure 3F), the model again detects the fault via a spike in energy (Figure 3G). However, in this

case, the downstream effects of the collision cause larger prediction errors in other joints and sensors, illustrating that while the effects of a fault can be isolated, identifying the root cause remains a challenge (Figures 3H–3J). Section 6 provides more details on how the faults are simulated. A subtlety about neural ASMs is that since the skill recognition is entirely implicit, the fault detection only depends on the current predictions, which in turn depend on the inferred context. This is unlike ASMs, which require knowing the explicit movement being performed to compare the observations to the corresponding signal statistics.

These qualitative examples of fault detection in different sensors are similar to Pastor et al. (2013, 2012). In addition, we perform a systematic evaluation of fault detection and isolation. For simplicity, we use percentile-based thresholding of the energy distribution during normal operation without faults to set the threshold for (out-of-distribution) fault detection. Figure 3K shows the energy distribution during our pick-and-place task over 10 trials of normal operation for each skill, along with the 95th percentile threshold as an example, which is equivalent to a 5% false-positive rate (FPR). We compare the performance of our neural ASM against a baseline analogous to traditional ASMs that uses normalized Z-scores for error detection (see section 6 for full details). By setting the detection threshold to correspond to an FPR of 1% to 5%, we find that the neural ASM correctly detects 82% to 83% of simulated faults, a modest improvement over the baseline's 74% to 82% detection rate. However, the neural ASM demonstrates a substantial advantage in fault isolation, correctly identifying the specific joint responsible for the fault in 79.5% of cases, whereas the baseline is only able to do so in 41% of cases. This concludes our basic demonstration of systematic evaluation, which can be extended to real-world robots with more realistic faults and comprehensive comparisons with alternate benchmarks in future work.

Finally, in cases where faults can be corrected on-the-fly, neural ASMs enable reactive correction. In ASMs, this is facilitated by DMPs, which themselves provide the reactive movement dynamics in end-effector space. In neural ASMs, reactive correction is modeled by minimizing proprioceptive prediction errors in either end-effector or configuration (joint) space, which is used for control. We demonstrate reactive fault correction by simulating a fault caused by a falling cube colliding with the robot, leading to temporary prediction errors. By minimizing proprioceptive prediction errors in configuration space based on predicted trajectory, the low-level controller automatically corrected for this disturbance (see Figure 4). In this simulation experiment, almost all faults can be corrected on the fly unless the grip strength is too weak and the object slips out of the grip due to the collision. A more systematic evaluation will require extending neural ASMs to real-world robots along with alternative human-like methods for fault correction (e.g., Collins et al., 2005); see section 5 for more details.

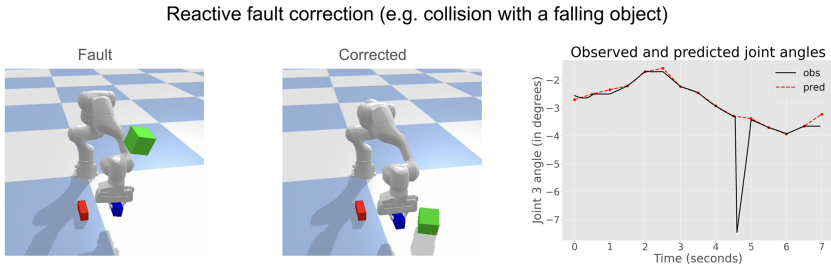


Figure 4: Demonstration of a fault resulting from a collision with a falling object. The fault is corrected reactively on the fly by having the low-level controller minimize proprioceptive prediction errors in joint configuration space using the proprioceptive predictions from neural ASMs.

In summary, this simple setup showcases that the core aspects of ASMs can be implemented using neural ASMs: (1) fault detection, enabling the robot to halt and seek assistance for unresolvable faults, and (2) reactive fault correction, supporting real-time adjustments for robust, fault-tolerant control.

#### 4.2 Contextual Inference in Skill Memory Separation and Expression.

Having demonstrated the utility of neural ASMs for self-preserving robots, we now demonstrate the role of contextual inference in the separation and expression of skill memories in our model. We use a robotics setup loosely inspired by Sheahan et al. (2016), who showed that motor planning of a follow-through motion, but not simply its execution, separates sensorimotor memories. However, we radically simplify the setup and eschew the optimal control that goes into arriving at the optimal trajectories, which compensate for the perturbations applied in their experiment. We rather assume that the appropriate sensorimotor sequences comprising the motor plan for each context are available in the demonstration data set (for learning from demonstrations) and focus solely on under which conditions our model can or cannot learn and express these memories. We aim to explain and qualitatively simulate certain aspects of human motor behavior in this robotics task. In doing so, we will also compare neural ASM with less biologically plausible counterparts (baselines): sequence-to-sequence recurrent neural networks (RNNs) trained using backpropagation through time (BPTT). This simulation highlights the challenges shared by humans and machines that use neural networks to learn multiple skills and would not usually arise if the system stored each skill independently in a library-like manner.

Our simulation is inspired by the work of Sheahan et al. (2016), who investigated how motor planning affects the separation of opposing motor skills. In their study, participants reached toward a target while counteracting one of two opposing force fields. A visual cue, available

either before or during the movement, indicated which force field was active. The crucial finding was that participants only learned to separate the two motor memories, producing distinct compensatory trajectories for each field, when the contextual cue was available before the movement, allowing for motor preparation (see Figures 5A and 5B). Therefore, in “planning only” and “planning and execution” (full follow-through) conditions, participants could separate the memories, whereas they could not do so in the execution-only condition, where the cue appeared mid-movement. This highlights the critical role of contextual inference during motor preparation for separating and selecting skill memories, a principle we test with our model.

To evaluate the capabilities of our framework, we compare the neural ASM against two standard baseline models: a sensory-to-motor (S-to-M) RNN and a sensorimotor-to-sensorimotor (SM-to-SM) RNN (see Figure 5D). The crucial difference between these models lies in how they process contextual information. While our neural ASM uses an iterative inference phase to model motor preparation from early cues, the baseline RNNs are trained with backpropagation through time (BPTT) and lack this distinct inference mechanism at runtime. All models are then tested on their ability to perform offline recall, generating a full trajectory from initial cues, with the specific architectures and training procedures detailed in section 6.

We find that in the planning only and planning and execution conditions, all models successfully separate the two skill memories, but fail to do so in the execution only condition, where the contextual cue is unavailable during preparation. To quantify this, we measure the directional compensatory deviation (DCD), which captures the (postexposure) maximum trajectory deviation specifically in the direction that would correctly compensate for the expected perturbation (see section 6 for a formal definition). As shown in Figure 5E, the DCD is significantly greater than zero for the planning conditions across all models, indicating successful memory separation. In contrast, the DCD is near zero for the execution-only condition, confirming the models’ failure to express the correct motor plan. This result is further illustrated by the recalled trajectories in Figure 5F, which show distinct, opposing paths for the two contexts, qualitatively replicating the postexposure hand paths reported by Sheahan et al. (2016). Sheahan et al. (2016) visualise their post-exposure hand path trajectories with four different starting points on the outside and ending in the central target, whereas we simplify when plotting our recalled motor plan in Figure 5F. The same results can also be observed with alternative metrics such as mean square error in predicted trajectories in comparison to the motor plan (see appendix C).

Interestingly, our model predicts a speed-accuracy trade-off in such ballistic actions, which is not predicted by baseline RNNs (see Figure 5G). Unlike baseline RNNs, our model has a cued iterative inference phase akin to setting the initial condition of the hidden state from multisensory

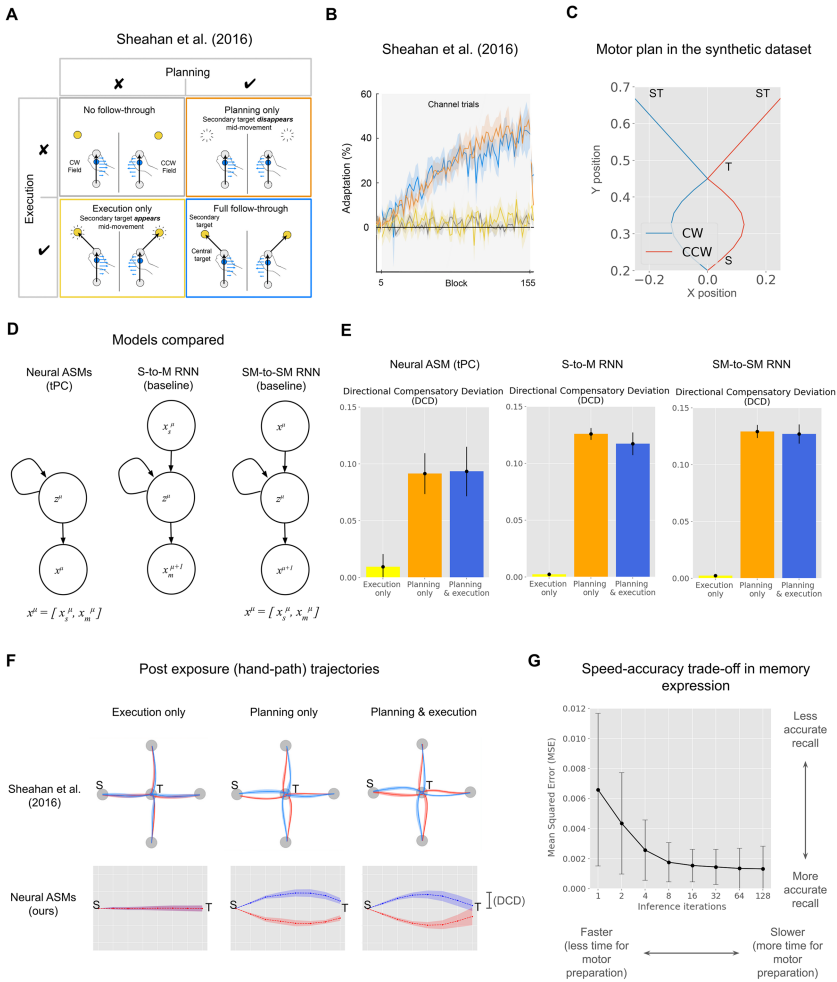


Figure 5: (A) Schematic from Sheahan et al. (2016) describing their experiment, which inspires our robotics experiment. (Adapted under CC BY 4.0 license. The figure was cropped and a title line was added.) (B) Adaptation result from Sheahan et al. (2016). (C) Motor plan (end-effector positions) in the synthetic data set, S: starting point, T: central target, ST: secondary target. Further details in Figure 8D in appendix B. Simplified representations of our model and baseline RNNs. S-to-M RNN predicts the motor observations at the next discrete time step ( $\mu + 1$ ) using the sensory observations at discrete time step ( $\mu$ ) as input. SM-to-SM predicts the sensorimotor observations at the next discrete time step ( $\mu + 1$ ) using sensorimotor observations at the current time step ( $\mu$ ). (E) Results qualitatively replicated by our model, on par with baseline models. (F) Qualitative comparison of postexposure (hand-path) trajectories (G). Speed accuracy trade-off, predicted in-memory expression here demonstrated in offline skill recall.

integration. Since we hypothesize such iterative inference of the hidden state as a potential mechanism of motor preparation, fewer inference iterations mean a shorter or constrained time of motor preparation. We find that very few iterations may be inadequate for complete hidden state inference, leading to less accurate skill recall, which improves with an increase in inference iterations. This additional timescale in our model predicts such speed-accuracy recall during skill recall or expression. A similar speed-accuracy trade-off has been observed in the expression of habitual actions in human experiments by Hardwick et al. (2019). (Section 5 for more details.)

We have demonstrated that our model can capture essential aspects of the results by Sheahan et al. (2016) on the role of contextual inference in sensorimotor memory separation and expression in our simplified robotics setup, and is qualitatively on par with RNN baselines using nonlocal learning rules. Further, it predicts a speed-accuracy trade-off during skill memory expression, which is not predicted by baseline RNNs.

## 5 Discussion

---

In this work, we introduced neural associative skill memories, a framework that leverages temporal predictive coding to learn an integrated generative model of a sensorimotor repertoire. Our results demonstrate that this approach, using biologically plausible local learning rules, can unify fault detection, reactive control, and contextual skill expression within a single network.

The model's behavior can be understood as a form of procedural memory, capturing the automatic, stimulus-driven recall of a well-learned sequence of actions (Dezfouli & Balleine, 2012; Éltető et al., 2022; Mizes et al., 2023; Simor et al., 2019). Our main finding is that the model's energy, representing the sum of squared prediction errors, serves as a natural and effective signal for fault detection. This aligns with the neuroscientific view of the brain as a prediction machine, where predictive surprise is fundamental for adaptive behavior. While our fault-detection performance showed only a modest improvement over a simple baseline, we argue that the key contribution is conceptual. Unlike traditional approaches that require a separate module to compare observations against the sensory statistics of an explicitly selected skill, in our framework, this capacity emerges inherently from the model's primary objective of predicting its sensorimotor stream.

Furthermore, we showed that the iterative inference process, where the model settles on a hidden state based on early cues, acts as a model of motor preparation and selects the appropriate skill memory to express. This provides a potential mechanistic account of motor preparatory activity, hypothesized to initialize a dynamical system without explicitly encoding movement parameters (Churchland, Byron et al., 2006; Churchland, Cunningham et al., 2010; Churchland, Santhanam et al., 2006, Cisek, 2006; Fetz, 1992). Iterative inference before plasticity refers to performing multiple inference steps to settle on a stable hidden state representation of the current observa-

tion before any synaptic weight updates are made. Such iterative inference before plasticity has also recently been demonstrated to have benefits over backpropagation in some biologically plausible tasks (Song et al., 2024). This provides a computational foundation for testing whether motor preparation (in terms of iterative inference) provides benefits in skill learning.

Further, a speed-accuracy trade-off arises from the model's iterative inference mechanism, where fewer iterations (less preparation time) can lead to a less accurate initial state for recall. We note that this trade-off does not arise in off-the-shelf recurrent neural networks. It directly links the time available for preparation (i.e., the number of inference iterations) to the accuracy of the recalled skill, a phenomenon observed in human habitual actions (Hardwick et al., 2019). When it comes to motor control, on one hand, neural ASMs appeal to the concepts such as the equilibrium point hypothesis (Feldman & Levin, 1995) and passive motion paradigm (Mohan et al., 2019), aligning with the free energy principle (Friston, 2010; Friston et al., 2010). On the other hand, neural ASMs are also compatible with views by Schaal et al. (2007), where the dynamical systems policy sits atop the optimal control system, thus providing a possible unification of both approaches to motor control (see Figure 7 in appendix A).

**5.1 Limitations.** Our study serves as a proof of concept, and several limitations should be acknowledged. The experiments used a simplified robotics setup with a limited skill repertoire and basic evaluation metrics; future work should test the scalability, generalization capabilities, and robustness of neural ASMs in more complex, dynamic environments. To simplify the demonstration, we simulate a limited repertoire of two pick-and-place skills, although the framework supports more. We also find that the rate of learning depends on the number of skills, consistent with observations in Howard et al. (2015). (See Figure 10 in appendix D.) The reliance on learning from demonstrations means the model is not learning optimal control strategies itself, a key distinction from reinforcement learning approaches.

Furthermore, our experiment, inspired by Sheahan et al. (2016), is not an exact replica. We omit their “no follow-through” condition based on findings that no adaptation occurs from static cues alone (Gandolfo et al., 1996; Howard et al., 2012, 2013, 2015; Sheahan et al., 2016). The inability to learn to counteract perturbations from static cues is a limitation of the learning-from-demonstration paradigm itself, where optimal trajectories are provided rather than our specific model. Nonetheless, this does not affect our central finding regarding how contextual inference supports the separation and expression of already learned skills.

On a framework level, further limitations exist. First, the model, under offline recall, relies on early-stage cues for contextual inference and cannot adapt to goal changes midtrajectory. Online recall could be used for re-recognition or switching between alternative skills in the repertoire. This would involve inferring the hidden states using the sensorimotor

observations at each time step, allowing the model to continuously infer the skill it is performing and then predict the sensorimotor observations for the next time step. However, it would not be able to perform novel goal-directed planning to change goals or to achieve certain preferred sensory observations. Second, our model concatenates sensorimotor channels into a single input vector, whereas a more neurally plausible architecture might involve separate interacting pathways for different modalities. Contextual selection of the correct skill would require motor and sensory observations to be connected to a shared hidden state. Third, the capacity of the tPC network is constrained by its local-in-time learning rule, which restricts its ability to learn complex, long-horizon sequences. Finally, all associative skill memory models (Pastor et al., 2013, 2012) assume that skill-related movements are stereotyped; high sensory variability would make the learned predictions unreliable.

**5.2 Future Work.** These limitations highlight several promising directions for future research. A critical next step is to test the scalability and robustness of the neural ASM framework in more complex, dynamic environments with a larger repertoire of skills. For robotic applications, improving reactive controllers, for instance, by incorporating principles from passive dynamic machines (Collins et al., 2005), would be a valuable step toward more energy-efficient and human-like reactive behaviors.

At a systemic level, incorporating goal-directed planning would require disentangling sensory and motor modalities. The concatenated input structure, similar to Nishimoto et al. (2008) and Yamashita and Tani (2008), would need to be replaced by a more neurally plausible architecture with separate, interacting pathways for different modalities. Future work can aim to integrate ideas on goal-directed planning to achieve a preferred sensory state similar to Matsumoto et al. (2022). One could potentially use a sensory tPC network for inferring the internal states from sensory observations and a “model inversion” of a motor tPC network to implement goal-directed motor planning, exploiting the duality of Bayesian inference and optimal control (Doya, 2021). This would allow for more flexible behavior, such as adapting to midtrajectory goal changes, and is an avenue for future work. Additionally, enabling neural ASMs to track posterior variance, for example, via Monte Carlo predictive coding (Oliviers et al., 2024) would make them viable as planning-capable world models. To overcome the limited temporal credit assignment of the current tPC network, one could investigate multi-timescale tPC architectures (Yamashita & Tani, 2008) or alternative biologically plausible approximations to BPTT using memory traces, such as e-prop (Bellec et al., 2020).

**5.3 Conclusion.** In conclusion, this work presents a biologically inspired framework for sensorimotor learning in robots that unifies skill learning, fault detection, and contextual expression within a single predictive coding network. Neural ASMs further offer a plausible approximation

of how associative sensorimotor memories might be implemented in the brain using local learning rules. The conceptual contribution of this work provides a step toward creating more adaptive, self-preserving robots and offers a tractable computational test bed for exploring theories of biological motor control.

## 6 Methods

**6.1 Model Algorithm.** The sequential memory in neural ASMs is modeled using a temporal predictive coding (tPC) network (Millidge et al., 2024; Tang et al., 2024b). Predictive coding models learn in a self-supervised fashion with the aim of best predicting the incoming input based on its own learned generative model. The model evaluates the actual input against its prediction by determining the difference in activity in the respective error neurons and minimizes these “errors” through adjusting neural activities and synaptic weights, which corresponds to the processes of inference and learning, respectively (Bogacz, 2017; Clark, 2015).

In mathematical terms, the task of neural ASMs can be seen as learning a sequence of sensorimotor observations (such as motor coordinates and associated sensory events like haptic feedback)  $(\mathbf{x}^t)_{t=0}^T$ . This can be reduced to learning the dynamics in these sensorimotor observations for each skill, that is, learning to associate each  $\mathbf{x}^\mu$  with the next  $\mathbf{x}^{\mu+1}$  ( $\mu = 0, 1, \dots, T - 1$ ). We use a two-layer tPC model, whose underlying graphical structure is that of a hidden Markov model (HMM). The lower layer of the tPC predicts the sensorimotor observations ( $\mathbf{x}^\mu$ ), and the upper layer predicts the next hidden state ( $\mathbf{z}^{\mu+1}$ ). This predictive processing account loosely models the hierarchical processing of raw sensory inputs by the neocortex, where hidden value neurons  $\mathbf{z}^\mu$  models the brain’s internal neural responses to the sequential sensory inputs  $\mathbf{x}^\mu$  (see Figure 6A).

The working of the tPC model in neural ASMs can be divided into two stages: (1) memorization and (2) recall (see Figure 6B). During memorization, tPC tries to minimize the sum of squared errors at step  $\mu$  with respect to the weights and the hidden activities:

$$F_\mu(\mathbf{z}^\mu, \mathbf{W}_H, \mathbf{W}_F) = \|\mathbf{z}^\mu - \mathbf{W}_H f(\hat{\mathbf{z}}^{\mu-1})\|_2^2 + \|\mathbf{x}^\mu - \mathbf{W}_F f(\mathbf{z}^\mu)\|_2^2, \quad (6.1)$$

where  $\mathbf{W}_H$  governs the temporal prediction in the hidden state, and  $\mathbf{W}_F$  are the weights governing predictions from  $\mathbf{z}^\mu$  to  $\mathbf{x}^\mu$ , with  $\hat{\mathbf{z}}^{\mu-1}$  being the hidden state inferred at the previous time step. During memorization, the model first infers the hidden representation of the current sensorimotor observational input  $\mathbf{x}^\mu$  by

$$\dot{\mathbf{z}}^\mu \propto -\frac{\partial F_\mu(\mathbf{z}^\mu, \mathbf{W}_H, \mathbf{W}_F)}{\partial \mathbf{z}^\mu} = -\varepsilon^{\mathbf{z},\mu} + f'(\mathbf{z}^\mu) \odot \mathbf{W}_F^\top \varepsilon^{\mathbf{x},\mu}, \quad (6.2)$$

where  $\odot$  denotes the element-wise product between two vectors, and  $\varepsilon^{\mathbf{z},\mu}$  and  $\varepsilon^{\mathbf{x},\mu}$  are defined as the hidden temporal prediction error

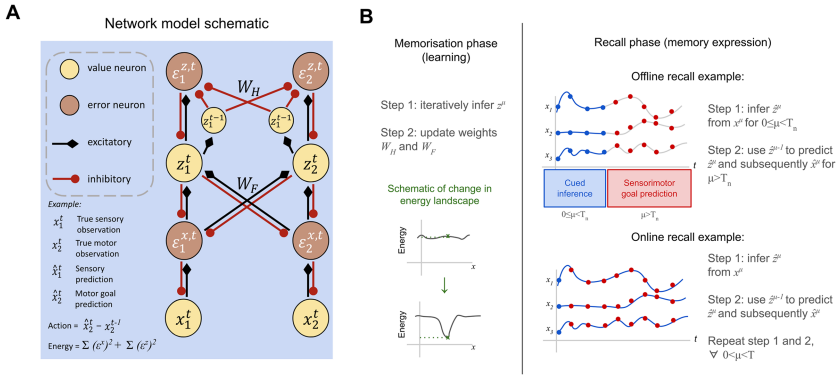


Figure 6: (A) Neural network implementations of temporal predictive coding (tPC) model used in section 4 for simulations. The network is illustrated with a single sensory and motor observation input for didactic purposes (though in reality, the network will have multiple sensorimotor observations as inputs). (B) The model learns in the memorization phase in a self-supervised manner, and the weight updates change the energy landscape to store these memories as attractors in the energy landscape (which is crucial in recognizing memorized skills). During the recall phase, the learned memories are expressed and can be either done offline (e.g., ballistic actions) or in an online manner where ground-truth observations are used to provide online feedback through model inversion at each step.

$\mathbf{z}^\mu - \mathbf{W}_H f(\hat{\mathbf{z}}^{\mu-1})$  and the top-down error  $\mathbf{x}^\mu - \mathbf{W}_F f(\mathbf{z}^\mu)$ , respectively. After  $\mathbf{z}^\mu$  converges,  $\mathbf{W}_H$  and  $\mathbf{W}_F$  are updated following gradient descent on  $F_\mu$ :

$$\Delta \mathbf{W}_H \propto - \frac{\partial F_\mu(\mathbf{z}^\mu, \mathbf{W}_H, \mathbf{W}_F)}{\partial \mathbf{W}_H} = \varepsilon^{\mathbf{z}, \mu} f(\hat{\mathbf{z}}^{\mu-1})^\top, \quad (6.3)$$

$$\Delta \mathbf{W}_F \propto - \frac{\partial F_\mu(\mathbf{z}^\mu, \mathbf{W}_H, \mathbf{W}_F)}{\partial \mathbf{W}_F} = \varepsilon^{\mathbf{x}, \mu} f(\mathbf{z}^\mu)^\top, \quad (6.4)$$

which are performed once for every presentation of the full sequence. Importantly, the converged  $\mathbf{z}^\mu$  is then used as  $\hat{\mathbf{z}}^\mu$  for the memorization at time step  $\mu + 1$ .

After memorization (learning) is completed, the model enters the recall stage, where all weights no longer change and the previously learned memories are expressed in response to certain cued input observations (also referred to as queries  $q$ ). Note that during the recall phase, the observation layer has no access to the correct patterns for the complete movement. Instead, it needs to dynamically change its value to retrieve the memories to predict these sensorimotor observations. The sequential memories are recalled or expressed using the learned weights  $\mathbf{W}_H$  and  $\mathbf{W}_F$ . The loss thus becomes

$$F_\mu(\mathbf{z}^\mu, \hat{\mathbf{x}}^\mu) = \|\mathbf{z}^\mu - \mathbf{W}_H f(\hat{\mathbf{z}}^{\mu-1})\|_2^2 + \|\hat{\mathbf{x}}^\mu - \mathbf{W}_F f(\mathbf{z}^\mu)\|_2^2, \quad (6.5)$$

where  $\hat{\mathbf{x}}^\mu$  denotes the activities of value neurons in the observation layer during recall. Both the hidden and observation layer value neurons are updated to minimize the loss. The hidden neurons will follow similar dynamics specified in equation 6.2, whereas the observation layer neurons are updated according to

$$\dot{\hat{\mathbf{x}}}^\mu \propto -\frac{\partial F_\mu(\mathbf{z}^\mu, \hat{\mathbf{x}}^\mu)}{\partial \hat{\mathbf{x}}^\mu} = -\varepsilon^{\mathbf{x}^\mu}, \quad (6.6)$$

and the converged  $\hat{\mathbf{x}}^\mu$  is the final retrieval.

In the case of sequential memory, there are two types of recall, offline and online. In case of offline recall, first  $\hat{\mathbf{z}}^\mu$  is iteratively inferred from  $T_n$  cued input ground-truth observations or queries  $q = \mathbf{x}^\mu$ , where  $\mu = (0, 1, \dots, T_n)$  (here,  $0 \leq T_n < T$ ). Once  $\hat{\mathbf{z}}^\mu$  is converged, it is used to recall  $\hat{\mathbf{x}}^\mu$  for  $\mu > T_n$ . In case of online recall, we query the model with  $q = \mathbf{x}^\mu$  (ground-truth), use the query to infer  $\hat{\mathbf{z}}^\mu$ , and then use  $\hat{\mathbf{z}}^\mu$  for the recall the next time step  $\mu + 1$  for  $\mu = 0, 1, \dots, T - 1$ . This distinction is important in our results; here, we only present offline recall results for skill memory expression in ballistic movements.

**6.2 Network Training Details.** The network details for the tPC network are as follows: The number of hidden units was 256, and the number of sensorimotor observation units depended on the task. The learning rate for the weight updates was  $10^{-4}$ , and the default learning iterations were 1000 per skill, trained with a batch size of 1. The iterative inference learning rate for hidden state update was  $10^{-2}$  with default inference iterations set to 100. The same hyperparameters were used for all simulation experiments. Kaiming uniform initialization was used in hidden layers for all networks.

The S-to-M RNN and SM-to-SM RNN are sequence-to-sequence RNNs. They used the same number of hidden units, learning rate, learning iterations, and batch size at the tPC network. They did not have iterative inference functionality, like the tPC network. The input unit size is the number of sensory observations, and the output unit size is the number of motor observations. The skill memory expression experiment trains the RNNs to predict the output sequence using inputs only from the first time step to mimic the offline recall used in tPC.

The sensorimotor sequences were Z-score normalized for each observation channel before being provided as inputs to all neural networks. Outputs were again unnormalized to original units before movement.

### 6.3 Experimental Setup and Robot Simulation Details.

**6.3.1 Task 1: Fault Detection and Reactive Correction.** In our simulations, the neural ASM operates at 2 Hz, guiding an underlying low-level position

controller operating at 40 Hz. The demonstration data set was generated in simulation by providing end-effector goals, from which joint angles were calculated using inverse kinematics and sensory observations were recorded. This process acts as a proxy for teleoperation. The experiment used 10 repetitions for each of two distinct pick-and-place skills. Each repetition is a sensorimotor sequence of 15 time steps (at 2 Hz), involving 25 observations (including desired joint angles, end-effector positions, gripper states, and sensed forces and torques), with realistic noise from the simulation process.

The minor fault is simulated as follows: joint 5 overshoots the desired goal by  $10^\circ$  while attempting to pick up the object ( $t = 3$  seconds) and gets stuck in that configuration for the remaining duration (see Figure 3A). The major fault is simulated as follows: joint 2 overshoots the desired goal by  $10^\circ$  and gets stuck while attempting to pick up the object, which results in the arm colliding with the floor (see Figure 3F).

For the systematic evaluation of fault detection, we use percentile-based thresholding of the model’s energy distribution during normal operation to establish a detection threshold, as illustrated in Figure 3K. This method allows us to control for the false-positive rate (FPR); for instance, a 95th percentile threshold corresponds to a 5% FPR. We then measure accuracy on 980 simulated fault trials, created by systematically varying the locked joint (1 to 7), the time of fault (1 to 6 seconds in 0.5 s steps), and the degree of joint angle overshoot ( $-15$  to  $+15$  in five steps). Fault isolation accuracy is measured as the proportion of trials where the joint with the highest absolute prediction error matches the simulated fault location.

We compare our model’s performance against a simple baseline analogous to traditional ASMs (Pastor et al., 2012), which relies on stored signal statistics. This baseline computes normalized errors for each observation channel using Z-score normalization ( $\epsilon = (x_i^t - \bar{x}_i)/\sigma_i$ ). The sum of squared normalized errors is then used for fault detection, and the channel with the maximum absolute normalized error is used for fault isolation. We acknowledge that this represents one possible implementation of a statistics-based approach and that other methods may exist. A comprehensive comparison with alternate methods is left for future work.

*6.3.2 Task 2: Contextual Skill Expression.* Inspired by the experimental paradigm of Sheahan et al. (2016), we designed a simulation to test the model’s ability to separate skill memories based on contextual cues. It is important to note that our goal was not to model the learning of optimal trajectories to counteract perturbations but rather to assess if the model could learn and express distinct, predefined motor plans from demonstrations under different contextual conditions. The synthetic data set therefore provides these optimal compensatory trajectories directly. For each skill, the data include end-effector positions, joint angles calculated via inverse kinematics, and a one-hot coded visual cue representing the context (see appendix B for plots). The sensory and motor sequences used in the skill

memory expression are presented in appendix B. We did not model the “no follow-through” condition from the original study, for reasons explained in the section 5.

We compared our neural ASM against two baseline recurrent neural network (RNN) models (see Figure 5D): a sensory-to-motor (S-to-M) RNN and a sensorimotor-to-sensorimotor (SM-to-SM) RNN, a discrete-time variant of the model used in Nishimoto et al. (2008). Unlike our tPC-based model, these baselines are trained using backpropagation through time (BPTT) and lack an iterative inference phase. To test skill expression from preparation, all models were evaluated using an offline recall procedure, generating the full trajectory based only on the inputs from the first time step. To parallel the human experiment, we trained all networks on each condition using six different random seeds for 1200 learning trials, and posttraining recall was averaged over 24 trajectories.

To quantify memory separation, we introduce the directional compensatory deviation (DCD) metric, which, unlike a simple absolute deviation, accounts for the direction of movement. DCD is defined as the maximum deviation of the recalled end-effector trajectory perpendicular to the straight line connecting the start (S) and target (T) points, projected onto the axis of correct compensation for a given context (e.g., leftward for a clockwise field). A positive DCD value indicates that the trajectory deviates in the appropriate direction to counteract the expected force field, while a value near zero indicates a failure to express the correct motor memory. This metric serves as a direct proxy for the directional adaptation measured by Sheahan et al. (2016).

## Appendix A: Unifying View on Dynamical Systems versus Optimal Control

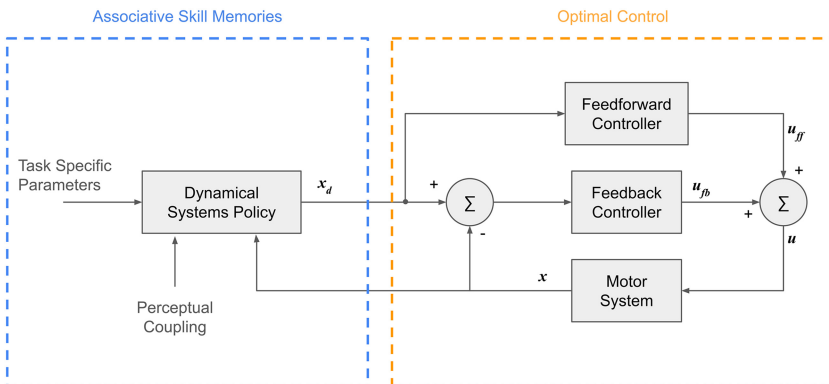


Figure 7: Unifying view proposed by Schaal et al. (2007), where the dynamic systems policy from associative skill memories can employ an optimal control-based low-level controller.

### Appendix B: Demonstration Data Used in Skill Memory Expression Task

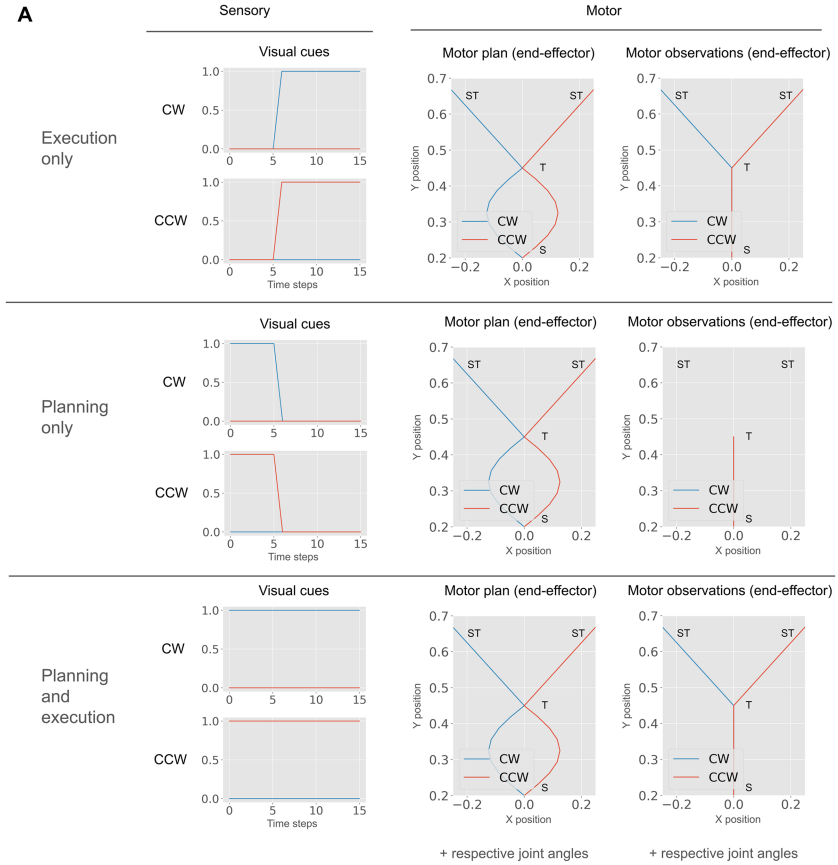


Figure 8: Plots of the sensory and (hypothetically optimal) motor sequences used for demonstrations in robot experiments on skill memory expression inspired by Sheahan et al. (2016).

### Appendix C: Additional Results in the Skill Memory Expression Task

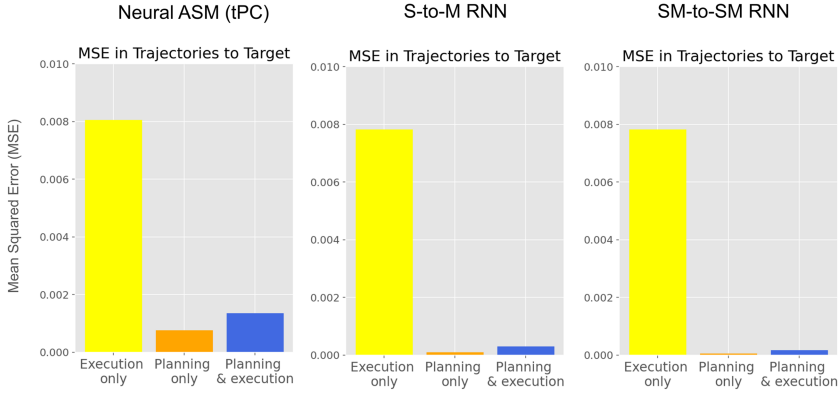


Figure 9: Mean squared errors (MSE) in trajectories to the target as an alternate metric for quantifying the results in the skill memory expression task, inspired by Sheahan et al. (2016). MSE will show an inverse relationship to the DCD metric that we constructed or seen in the trajectory plots of Sheahan et al. (2016): failure to separate the skill memories results in low DCD and high MSE and vice versa.

### Appendix D: Learning Rate Varies with Number of Skills

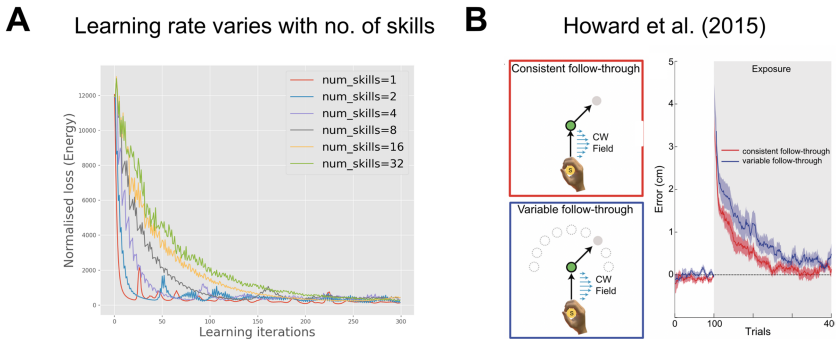


Figure 10: The fewer the number of distinct skills to memorize, the faster is the learning rate of our model, as seen in energy (normalized loss) over epochs. This potentially explains why consistent follow-throughs improve learning rates as variable follow-throughs can split the learning into different skill memories rather than a single memory, as observed by Howard et al. (2015). (Adapted from Howard et al., 2015, under CC BY 4.0 license. The figure was cropped, and a title line was added.)

## Author Contributions

---

P.M., I.H., and B.S. conceptualized the core idea, and M.T. and T.E.L. contributed further discussions. P.M. designed and performed the simulation experiments. M.T. provided the initial code on the tPC network from his previous work. P.M. wrote the first draft of the manuscript with feedback from B.S. and I.H. All authors contributed to editing and approving the manuscript.

## Acknowledgments

---

We thank the anonymous reviewers from *Neural Computation* for their helpful feedback on the original submission. P.M. thanks Rafal Bogacz, Michael Browning, Gaspard Oliviers, and anonymous reviewers from COSYNE 2024 and IWAI 2024 for their helpful feedback on earlier iterations of this work. P.M. and B.S. are funded by Wellcome Trust (214251/Z/18/Z, 203139/Z/16/Z, and 203139/A/16/Z), IITP (MSIT 2019-0-01371), and JSPS (22H04998). M.T. is supported by the E. P. Abraham Scholarship in the Chemical, Biological/Life and Medical Sciences. T.E.L. was supported by the Gruber Science Fellowship and the Interdepartmental Neuroscience Program at Yale University, which is funded by T32 NS041228 from the National Institute of Neurological Disorders and Stroke. This research was also partly supported by the NIHR Oxford Health Biomedical Research Centre (NIHR203316). The views expressed are our own and not necessarily those of the NIHR or the Department of Health and Social Care.

We have no conflicts of interest to declare.

## Code Availability

---

The code is available at <https://github.com/PranavMahajan25/NeuralASMs>.

## References

---

- Ahmadi, A., & Tani, J. (2017). How can a recurrent neurodynamic predictive coding model cope with fluctuation in temporal patterns? Robotic experiments on imitative interaction. *Neural Networks*, 92, 3–16. 10.1016/j.neunet.2017.02.015
- Ba, J., Hinton, G. E., Mnih, V., Leibo, J. Z., & Ionescu, C. (2016). Using fast weights to attend to the recent past. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in neural information processing systems*, 29. Curran.
- Barrett, L. F. (2017). The theory of constructed emotion: An active inference account of interoception and categorization. *Social Cognitive and Affective Neuroscience*, 12(1), 1–23.

- Bellec, G., Scherr, F., Subramoney, A., Hajek, E., Salaj, D., Legenstein, R., & Maass, W. (2020). A solution to the learning dilemma for recurrent networks of spiking neurons. *Nature Communications*, *11*(1), 3625. 10.1038/s41467-020-17236-y
- Bogacz, R. (2017). A tutorial on the free-energy framework for modelling perception and learning. *Journal of Mathematical Psychology*, *76*, 198–211. 10.1016/j.jmp.2015.11.003
- Churchland, M. M., Byron, M. Y., Ryu, S. I., Santhanam, G., & Shenoy, K. V. (2006). Neural variability in premotor cortex provides a signature of motor preparation. *Journal of Neuroscience*, *26*(14), 3697–3712. 10.1523/JNEUROSCI.3762-05.2006
- Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Ryu, S. I., & Shenoy, K. V. (2010). Cortical preparatory activity: Representation of movement or first cog in a dynamical machine? *Neuron*, *68*(3), 387–400. 10.1016/j.neuron.2010.09.015
- Churchland, M. M., Santhanam, G., & Shenoy, K. V. (2006). Preparatory activity in premotor and motor cortex reflects the speed of the upcoming reach. *Journal of Neurophysiology*, *96*(6), 3130–3146. 10.1152/jn.00307.2006
- Cisek, P. (2006). Preparing for speed: Focus on “preparatory activity in premotor and motor cortex reflects the speed of the upcoming reach.” *Journal of Neurophysiology*, *96*(6), 2842–2843. 10.1152/jn.00857.2006
- Clark, A. (2013). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behavioral and Brain Sciences*, *36*(3), 181–204. 10.1017/S0140525X12000477
- Clark, A. (2015). *Surfing uncertainty: Prediction, action, and the embodied mind*. Oxford University Press.
- Collins, S., Ruina, A., Tedrake, R., & Wisse, M. (2005). Efficient bipedal robots based on passive-dynamic walkers. *Science*, *307*(5712), 1082–1085. 10.1126/science.1107799
- Dezfouli, A., & Balleine, B. W. (2012). Habits, action sequences and reinforcement learning. *European Journal of Neuroscience*, *35*(7), 1036–1051. 10.1111/j.1460-9568.2012.08050.x
- Doya, K. (2021). Canonical cortical circuits and the duality of Bayesian inference and optimal control. *Current Opinion in Behavioral Sciences*, *41*, 160–167. 10.1016/j.cobeha.2021.07.003
- Éltető, N., Nemeth, D., Janacsek, K., & Dayan, P. (2022). Tracking human skill learning with a hierarchical Bayesian sequence model. *PLOS Computational Biology*, *18*(11), e1009866.
- Feldman, A. G., & Levin, M. F. (1995). The origin and use of positional frames of reference in motor control. *Behavioral and Brain Sciences*, *18*(4), 723–744. 10.1017/S0140525X0004070X
- Fetz, E. (1992). Are movement parameters recognizably coded in the activity of single neurons? *Behavioral and Brain Sciences*, *15*, 679–690.
- Friston, K. (2005). A theory of cortical responses. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *360*(1456), 815–836. 10.1098/rstb.2005.1622
- Friston, K. (2010). The free-energy principle: A unified brain theory? *Nature Reviews Neuroscience*, *11*(2), 127–138. 10.1038/nrn2787
- Friston, K. J., Daunizeau, J., Kilner, J., & Kiebel, S. J. (2010). Action and behavior: A free-energy formulation. *Biological Cybernetics*, *102*, 227–260. 10.1007/s00422-010-0364-z

- Gandolfo, F., Mussa-Ivaldi, F. A., & Bizzi, E. (1996). Motor learning by field approximation. In *Proceedings of the National Academy of Sciences*, 93(9), 3843–3846. 10.1073/pnas.93.9.3843
- Hardwick, R. M., Forrence, A. D., Krakauer, J. W., & Haith, A. M. (2019). Time-dependent competition between goal-directed and habitual response preparation. *Nature Human Behaviour*, 3(12), 1252–1262. 10.1038/s41562-019-0725-0
- Howard, I. S., Ingram, J. N., Franklin, D. W., & Wolpert, D. M. (2012). Gone in 0.6 seconds: The encoding of motor memories depends on recent sensorimotor states. *Journal of Neuroscience*, 32(37), 12756–12768. 10.1523/JNEUROSCI.5909-11.2012
- Howard, I. S., Wolpert, D. M., & Franklin, D. W. (2013). The effect of contextual cues on the encoding of motor memories. *Journal of Neurophysiology*, 109(10), 2632–2644. 10.1152/jn.00773.2012
- Howard, I. S., Wolpert, D. M., & Franklin, D. W. (2015). The value of the follow-through derives from motor learning depending on future actions. *Current Biology*, 25(3), 397–401. 10.1016/j.cub.2014.12.037
- Ijspeert, A. J., Nakanishi, J., Hoffmann, H., Pastor, P., & Schaal, S. (2013). Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Computation*, 25(2), 328–373. 10.1162/NECO\_a\_00393
- Jeannerod, M. (1994). The representing brain: Neural correlates of motor intention and imagery. *Behavioral and Brain Sciences*, 17(2), 187–202. 10.1017/S0140525X00034026
- Keramati, M., & Gutkin, B. (2014). Homeostatic reinforcement learning for integrating reward collection and physiological stability. *eLife*, 3, e04811. 10.7554/eLife.04811
- Lanillos, P., & Cheng, G. (2018). Adaptive robot body learning and estimation through predictive coding. In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 4083–4090).
- Li, T. E., Tang, M., & Bogacz, R. (2024). Predictive coding model can detect novelty on different levels of representation hierarchy. bioRxiv, 2024–06.
- Mahajan, P., Dayan, P., & Seymour, B. (2025). Homeostasis after injury: How intertwined inference and control underpin post-injury pain and behaviour. bioRxiv, 2025–02.
- Matsumoto, T., Ohata, W., Benureau, F. C., & Tani, J. (2022). Goal-directed planning and goal understanding by extended active inference: Evaluation through simulated and physical robot experiments. *Entropy*, 24(4), 469. 10.3390/e24040469
- Millidge, B., Tang, M., Olanlouy, M., Harper, N. S., & Bogacz, R. (2024). Predictive coding networks for temporal prediction. *PLOS Computational Biology*, 20(4), 1–31.
- Mizes, K. G., Lindsey, J., Escola, G. S., & Ölveczky, B. P. (2023). Dissociating the contributions of sensorimotor striatum to automatic and visually guided motor sequences. *Nature Neuroscience*, 26(10), 1791–1804. 10.1038/s41593-023-01431-3
- Mizes, K. G., Lindsey, J., Escola, G. S., & Ölveczky, B. P. (2024). The role of motor cortex in motor sequence execution depends on demands for flexibility. *Nature Neuroscience*, 27(12), 1–10. 10.1038/s41593-024-01792-3
- Mohan, V., Bhat, A., & Morasso, P. (2019). Muscleless motor synergies and actions without movements: From motor neuroscience to cognitive robotics. *Physics of Life Reviews*, 30, 89–111. 10.1016/j.plrev.2018.04.005

- Nishimoto, R., Namikawa, J., & Tani, J. (2008). Learning multiple goal-directed actions through self-organization of a dynamic neural network model: A humanoid robot experiment. *Adaptive Behavior*, 16(2–3), 166–181. 10.1177/1059712308089185
- Nishimoto, R., & Tani, J. (2004). Learning to generate combinatorial action sequences utilizing the initial sensitivity of deterministic dynamical systems. *Neural Networks*, 17(7), 925–933. 10.1016/j.neunet.2004.02.003
- Oliviers, G., Bogacz, R., & Meulemans, A. (2024). Learning probability distributions of sensory inputs with Monte Carlo predictive coding. *PLOS Computational Biology*, 20(10), e1012532. 10.1371/journal.pcbi.1012532
- Pastor, P., Kalakrishnan, M., Meier, F., Stulp, F., Buchli, J., Theodorou, E., & Schaal, S. (2013). From dynamic movement primitives to associative skill memories. *Robotics and Autonomous Systems*, 61(4), 351–361. 10.1016/j.robot.2012.09.017
- Pastor, P., Kalakrishnan, M., Righetti, L., & Schaal, S. (2012). Towards associative skill memories. In *Proceedings of the 2012 12th IEEE-RAS International Conference on Humanoid Robots* (pp. 309–315).
- Rao, R. P., & Ballard, D. H. (1999). Predictive coding in the visual cortex: A functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1), 79–87. 10.1038/4580
- Rogel, A., Savery, R., Yang, N., & Weinberg, G. (2022). Robogroove: Creating fluid motion for dancing robotic arms. In *Proceedings of the 8th International Conference on Movement and Computing* (pp. 1–9).
- Rumelhart, D. E., McClelland, J. L., & PDP Research Group. (1986). *Parallel distributed processing, vol. 1: Explorations in the microstructure of cognition: Foundations*. MIT Press.
- Salvatori, T., Song, Y., Hong, Y., Frieder, S., Sha, L., Xu, Z., . . . Lukasiewicz, T. (2021). Associative memories via predictive coding. In M. A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. Wortman Vaughan (Eds.), *Advances in neural information processing systems*, 34. Curran.
- Schaal, S., Mohajerian, P., & Ijspeert, A. (2007). Dynamics systems vs. optimal control—a unifying view. *Progress in Brain Research*, 165, 425–445. 10.1016/S0079-6123(06)65027-9
- Seth, A. K. (2013). Interoceptive inference, emotion, and the embodied self. *Trends in Cognitive Sciences*, 17(11), 565–573. 10.1016/j.tics.2013.09.007
- Seymour, B., Crook, R. J., & Chen, Z. S. (2023). Post-injury pain and behaviour: A control theory perspective. *Nature Reviews Neuroscience*, 24(6), 378–392. 10.1038/s41583-023-00699-5
- Seymour, B., & Mancini, F. (2020). Hierarchical models of pain: Inference, information-seeking, and adaptive control. *NeuroImage*, 222, 117212. 10.1016/j.neuroimage.2020.117212
- Shadmehr, R., Smith, M. A., & Krakauer, J. W. (2010). Error correction, sensory prediction, and adaptation in motor control. *Annual Review of Neuroscience*, 33(1), 89–108. 10.1146/annurev-neuro-060909-153135
- Sheahan, H. R., Franklin, D. W., & Wolpert, D. M. (2016). Motor planning, not execution, separates motor memories. *Neuron*, 92(4), 773–779. 10.1016/j.neuron.2016.10.017
- Simor, P., Zavecz, Z., Horváth, K., Élterő, N., Török, C., Pesthy, O., . . . Nemeth, D. (2019). Deconstructing procedural memory: Different learning trajectories and

- consolidation of sequence and statistical learning. *Frontiers in Psychology*, 9, 2708. 10.3389/fpsyg.2018.02708
- Song, Y., Millidge, B., Salvatori, T., Lukasiewicz, T., Xu, Z., & Bogacz, R. (2024). Inferring neural activity before plasticity as a foundation for learning beyond back-propagation. *Nature Neuroscience*, 27(2), 348–358. 10.1038/s41593-023-01514-1
- Tang, M., Barron, H., & Bogacz, R. (2024a). *Learning grid cells by predictive coding*. arXiv:2410.01022.
- Tang, M., Barron, H., & Bogacz, R. (2024b). Sequential memory with temporal predictive coding. In A. Oh, T. Naumann, A. Globerson, M. Saenko, M. Hardt, & S. Levine (Eds.), *Advances in neural information processing systems*, 36. Curran.
- Tani, J. (1996). Model-based learning for mobile robot navigation from the dynamical systems perspective. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(3), 421–436. 10.1109/3477.499793
- Tschantz, A., Barca, L., Maisto, D., Buckley, C. L., Seth, A. K., & Pezzulo, G. (2022). Simulating homeostatic, allostatic and goal-directed forms of interoceptive control using active inference. *Biological Psychology*, 169, 108266. 10.1016/j.biopsycho.2022.108266
- Walters, E. T., Crook, R. J., Neely, G. G., Price, T. J., & Smith, E. S. J. (2023). Persistent nociceptor hyperactivity as a painful evolutionary adaptation. *Trends in Neurosciences*, 46(3), 211–227. 10.1016/j.tins.2022.12.007
- Whittington, J. C., & Bogacz, R. (2017). An approximation of the error backpropagation algorithm in a predictive coding network with local Hebbian synaptic plasticity. *Neural Computation*, 29(5), 1229–1262. 10.1162/NECO\_a\_00949
- Whittington, J. C., & Bogacz, R. (2019). Theories of error back-propagation in the brain. *Trends in Cognitive Sciences*, 23(3), 235–250. 10.1016/j.tics.2018.12.005
- Yamashita, Y., & Tani, J. (2008). Emergence of functional hierarchy in a multiple timescale neural network model: A humanoid robot experiment. *PLOS Computational Biology*, 4(11), e1000220.

---

Received May 20, 2025; accepted August 24, 2025.