Evaluating Sensorimotor Abstraction on Curricula for Learning Mobile Manipulation Skills

Oscar Youngquist*, Alenna Spiro†, Khoshrav Doctor‡ and Roderic Grupen§
Manning College of Information and Computer Science, University of Massachusetts Amherst
Email: *oyoungquist@umass.edu, †aspiro@umass.edu, ‡kdoctor@cs.umass.edu, §grupen@cs.umass.edu

***Abstract*—Developmental mechanisms in newborn animals shepherd the infant through interactions with the world that form the foundation for hierarchical skills. An important part of this guidance resides in mechanisms of growth and maturation, wherein patterns of sensory and motor recruitment constrain learning complexity while building foundational expertise and transferable control knowledge. The resulting control policies represent a sensorimotor state abstraction that can be leveraged when developing new behaviors. This paper uses a computational model of developmental learning with parameters for controlling the recruitment of sensory and motor resources, and evaluates how this influences sample efficiency and fitness for a specific mobile manipulation task. We find that a developmental curriculum driven by sensorimotor abstraction drastically improves (by up to an order of magnitude) learning performance and sample efficiency over non-developmental approaches. Additionally, we find that the developmental policies/state abstractions offer significant robustness properties, enabling skill transfer to novel domains without additional training.**

***Index Terms*—Developmental Learning, Hierarchical Skill Acquisition, Mobile Manipulation**

I. Introduction and Related Work

Information embedded in our genes provides support for learning how to control interactions with the environment. A maturational schedule encoded in the genome fosters the acquisition of a foundation of critical skills that support continuing development and survival of the species. In contrast, many approaches to machine learning exploit very little of this kind of curricular structure. Instead, they rely on a large number of sampled interactions to learn from scratch. Learning motor control in this manner is subject to extremely high-dimensional state-spaces [1], where complexity problems lead to a family of exploratory, interactive approaches. However, exploring the full breadth of possible interactions with the unstructured world is a tremendous challenge for computational systems—biological or otherwise.

To constrain this learning complexity during the first 12 months of life, infants follow a pre-configured *maturational curriculum*—a sequence of interaction contexts that controls the incremental complexity of learning using low-dimensional subsets of sensory and motor resources in a manner that highlights critical skills [2], [3]. We explore this concept using a robust, error-suppressing *landscape of attractors*[1] to represent an analog of spinal and developmental reflexes in newborn infants that engage low-dimensional sensorimotor combinations to support learning and development in embodied agents through situated experience. In particular, we show how similar curricula support learning a non-trivial control stack for a simple, simulated mobile manipulator using closed-loop, reflexive controllers that interact with the environment.

It is well understood that animals are born with tacit knowledge encoded in neuroanatomical structures that capture the kinodynamic and perceptual aptitudes of the agent. These structures support developmental reflexes, which guide the infant to develop critical skills using subsets of sensorimotor resources in order to efficiently learn how to interact with the world through exploration [3], [7]. Additionally, they serve as building blocks for more integrated skills that incrementally reveal the latent ability of the agent: affording new opportunities for higher-level control. At each stage of development, the infant is learning to reliably sequence tacitly encoded control knowledge and/or other skills in response to environmental stimuli. Skills can be composed hierarchically and reused to develop increasingly complex behavior. Hierarchically composed skills reduce the complexity of subsequent learning problems by efficiently transferring acquired control knowledge to new domains with little to no additional training [8], [9].

Inspired by this observation, the reinforcement learning (RL) community and roboticists are exploring the curriculum learning (CL) paradigm [10]. CL guides exploration during RL by creating a sequence of sub-goals/tasks, each within the same environment and sharing the same dynamics as the final skill [11], [12]. Foglino et. al. [13] extends this framework to include distinct tasks in the learning sequence and training objectives outside of reducing overall training-time. Further, the automatic generation of developmental curricula has been explored [14]. This line of inquiry considers two major components—decomposing the final skill into sub-skills that encapsulate beneficial control knowledge [15], [16] and determining the optimal training sequence of the generated sub-skills [16], [17]. Nagai et. al. [18] also constrain learning complexity using developmental stages by explicitly modeling the three stages of joint attention development observed in human infants in [19]. Most similar to this work, [20] investigate a Lift-Constraint, Act, Saturate Approach method in which complex skills are developed in stages by removing various learning constraints (sensorimotor, anatomical, environmental, etc.) and saturating the agents acquisition of control knowledge before moving on to the next stage. Additionally, Weber et

[1] The landscape is implemented using the control basis framework [4]–[6]

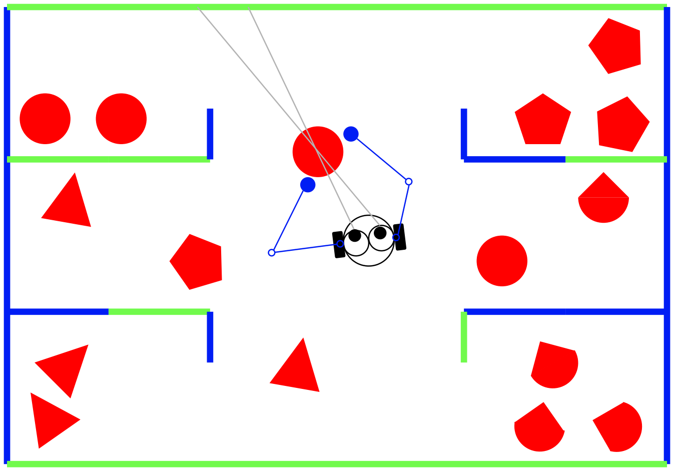

Fig. 1. A depiction of the *Roger* simulator and a sorting task that makes repeated use of an object relocation mobile manipulation control stack: each object type must be sorted into predetermined regions.

al. [21] investigate RL techniques for identifying early life options—useful recurring sensorimotor policies—specifically geared towards accelerating an agent's learning performance in the very early stages of their lives.

Maturational mechanisms defeat computational learning complexity by engaging small subsets of sensorimotor resources (such as proximodistal and cephalocaudal development in infants) to determine what stimuli in the world can be controlled given these constraints and acquire policies for controlling them. These control policies represent sensorimotor abstractions that can be leveraged by later development processes, thereby advancing the frontier of computational decision policies as development proceeds. In this work, we propose a class of developmental curricula over combinations of sensorimotor resources and closed-loop controllers in a landscape of attractors, driven by advancing controllability through managing the complexity of combinatoric decision spaces. A variety of developmental curricula are developed to teach a simulated robot *Roger* to locate, grasp, and transport objects in an unobstructed environment and we evaluate how hierarchical control abstraction influences development. The contributions of this work are preliminary results supporting a method for embodied, situated agents to compose tacit and acquired knowledge into transferable skills through exploration; demonstrating how the hierarchical abstraction provided by such skills improve sample efficiency and reduce the training time required to learn transferable skills.

II. A Platform for Experiments in Developmental Learning

A. *Roger-the-Crab*

The *Roger* simulator (Figure 1) was inspired by a kinematic platform created by Paul Churchland to examine the neural basis for hand-eye coordination [22]. *Roger* is a mobile manipulator that exists in flatland (2D) and consists of three elastic bodies (a body and two hands). These bodies inherit the kinodynamic properties of its nonholonomic base and a pair of planar, 2R manipulators. *Roger*'s sensor suite includes a stereo pair of cameras and additionally, *Roger*'s body and hands act as tactile sensors.

B. *The Control Basis*

In the control basis framework, control is derived online by associating task-independent potential functions (ϕ) with sensory features (σ) and output effectors (τ) to create closed-loop processes as a proxy for the innate developmental reflexes built into animals. These primitives can be composed concurrently to create analogs of the intersegmental responses seen in vertebrates, and sequentially to form hierarchical skills [4]–[6]. Concurrent control is achieved using a prioritized composition of actions where a subordinate controller c_2 is projected into the nullspace of a superior controller c_1 using the subject-to operator: $c_2 \triangleleft c_1$ [23]. As in [4]–[6], the $\triangleleft$ "subject-to" operator represents the Moore-Penrose right-pseudoinverse. This allows the subordinate objective to be optimized without hindering the superior objective.

Closed-loop systems $\phi_i|_\tau^\sigma$ are derived by following gradients $\nabla\phi_i$, and the state of closed-loop interaction is determined using membership functions γ_i defining partitions of the phase portrait of $(\phi_i, \dot{\phi}_i)$. In this work, a coarse partition defined by Equation 1 is used to represent states where the environment affords no reference stimuli (NOREF), stimuli are detected but gradient descent has not reached equilibrium (!CONV), or when the system is in the set of equilibrium states (CONV).

$$\gamma_i(\phi_i, \dot{\phi}_i) = \begin{cases} \text{NOREF} & \sigma \text{ undetected} \\ \text{!CONV} & ||\nabla\phi_i|| > \epsilon \\ \text{CONV} & ||\nabla\phi_i|| \leq \epsilon \end{cases} \tag{1}$$

Each objective function ϕ_i defines a subset of the domain called the region of attraction where $\gamma_i \neq$ NOREF. In the interior of this region, gradient descent on ϕ_i may be engaged to funnel the control state to an equilibrium set. When multiple functions share the same domain, this generates a multi-modal landscape of attractors [24]. The arrangement of attractors reveals transitions that form probabilistic roadmaps through the situated state-space, as shown in Figure 2. These transitions can be used to generate task-independent models that greatly reduce the complexity of the agent-environment interaction.

A skill takes the form of a sequence of actions that leads the system through a set of control equilibria following high-probability transitions in the landscape of attractors in order to reach a goal reward state. Each skill encapsulates control knowledge afforded by the agent's embodiment and environment—directly encoding which of these interactions are observable and controllable. Subsequent behavior may leverage control knowledge encoded in existing skills, rather than construct entirely new, task-specific behavior from scratch. This has the effect of insulating high-level skills from the low-level details required to effectively coordinate motor and sensory resources.

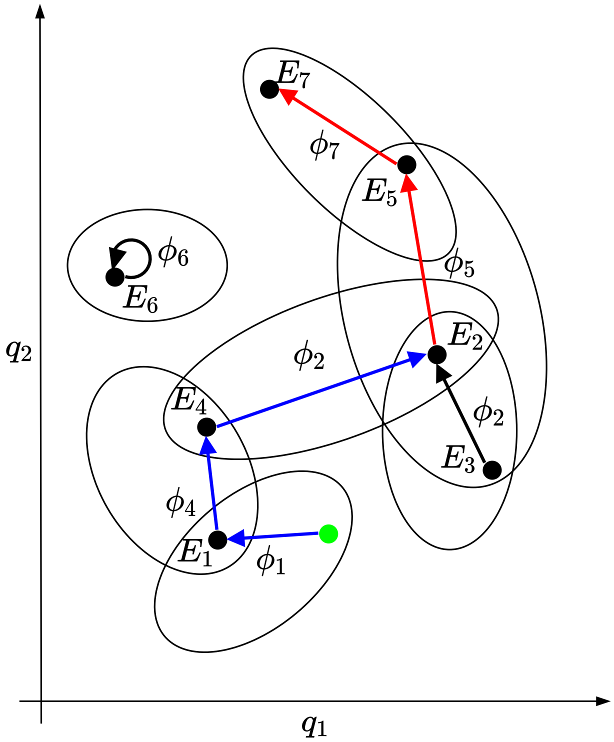

Fig. 2. A landscape of attractors with 7 regions of attraction (ellipses) in the domain $\mathbf{q} = [q_1, q_2]$. Starting at the green dot, equilibrium set E_7 can be reached through the sequential activation of five controllers. However, in the presence of high-probability pathways, skills encapsulating these sequential transitions (such as the blue and the red skills) could be used instead to reduce the state-space.

To learn skills, we define a Markov decision process (MDP) $M = \langle S_i, A, \Psi_i, T, R \rangle$ for each stage i of a developmental curriculum. A represents the complete set of actions (primitive and acquired skills); Ψ_i consists of the actions available at learning stage i; and S_i is the set of states defined by all possible values of the vector $\boldsymbol{\gamma}$, created from the partition function values of each of the individual actions in Ψ_i, $\boldsymbol{\gamma} = [\gamma_0, \gamma_1, \ldots, \gamma_n]$, defining the developmental context (DC) of the new skill. Finally, $R : S \times A \rightarrow \mathbb{R}$ is a reward function and T is a set of conditional transition probabilities between states, $T(s'|s, a)$, that we are attempting to learn.

C. A Control Basis for Mobile Manipulation

In this section, we introduce reflexive control actions $a \in A_0$ for use in the experiments. For simplicity, we drop the full $\phi_i|_\tau^\sigma$ notation for closed-loop actions and refer to them using the parameterized objective function ϕ_i.

ϕ_0: A objective function that actuates *Roger*'s eyes and base to center an environmental stimuli on the image plane.

ϕ_1: A harmonic function path planner [25] used to generate robust, reactive, and collision-free motions.

ϕ_2: A potential function that uses visual estimates to preshape the robot for grasping.

ϕ_3: A contact configuration potential function that uses tactile contact normal feedback to move contacts to optimize forces and moments [26].

ϕ_4: A kinematic conditioning function that optimizes the posture of the body and arms for a set of fixed contact locations.

ϕ_5: A potential function in the same class as ϕ_3 that dynamically adjusts contacts in order to compensate for the inertial properties of an object while in motion.

Furthermore, this work assumes that some simple skills exist with embedded control knowledge that are composed exclusively of primitive controllers[2]. We distinguish policies from primitives by using Φ_i (value function) for closed-loop interaction rather than ϕ_i (potential function).

Φ_6: A control policy that searches for and then tracks prescribed stimuli using ϕ_0.

Φ_7: A control policy that combines ϕ_1 and $\phi_4 \triangleleft \phi_2$ sequentially to approach an object while pre-shaping the arms for prehensile grasping.

III. Learning Experiments

To demonstrate our approach, we implement a two-stage curriculum in which *Roger* first learns a new skill Φ_{SG} to **S**earch for and then approach and **G**rasp an object. The Φ_{SG} skill is reused in the next stage to learn how to **T**ransport objects in an unobstructed environment, resulting in the Φ_T skill. We evaluate the most effective ways to configure Ψ in each stage to acquire the right control knowledge and, therefore, to maximize cumulative learning performance while minimizing combinatorial complexity.

A. Learning to Search and Grasp

The first stage of this curriculum is to learn how to coordinate sensor and motor resources to locate a red ball in an unobstructed environment, approach it, and form a prehensile grasp on the ball.

1) Experimental Settings: For the Φ_{SG} stage, we consider three experimental settings, listed below:

- **Baseline #1 (B1)**: In this setting, we only consider primitive actions, concurrent combinations, and the composite skill Φ_6 in Ψ_0: $\Psi_0 = \{\phi_1, \phi_3, \phi_4 \triangleleft \phi_2, \Phi_6\}$. Learning performance in this setting establishes a baseline for comparing different developmental curricula.
- **Developmental Context #1 (DC1)**: Next, all the primitive actions and skills are available to the agent: $\Psi_0 = \{\phi_1, \phi_3, \phi_4 \triangleleft \phi_2, \Phi_6, \Phi_7\}$.
- **Developmental Context #2 (DC2)**: Last, we rely exclusively on the Φ_7 skill/abstraction to express the preshape behavior and make primitives ϕ_1 and $\phi_4 \triangleleft \phi_2$, on which Φ_7 depend, ineligible: $\Psi_0 = \{\phi_3, \Phi_6, \Phi_7\}$.

All states for which $\gamma(\phi_3) = \text{CONV}$ are absorbing states and in each of the developmental contexts the reward R

[2] In other experiments, these policies have been learned following a similar procedure to that outlined in Section III. For the sake of space we omit the details of this work.

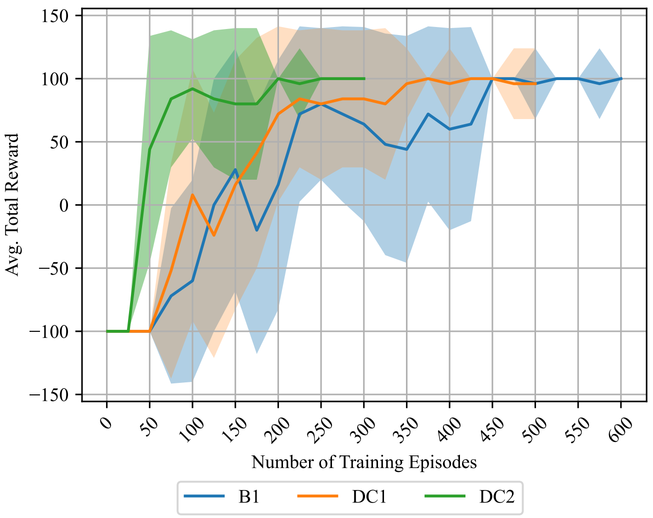

Fig. 3. Learning curves of Φ_{SG} training across all experimental settings. The average (and standard deviation) episodic reward across 50 trials is reported as a function of time.

TABLE I
EXPERIMENTAL RESULTS FOR Φ_{SG} LEARNING EXPERIMENTS

Approach	**Train Time** (min:sec)	**%** **Improvement**	**Avg. Reward ± Std**
B1	49:54	-	100.0 ± 0.0
DC1	41:35	16.67%	96.0 ± 28.0
DC2	24:55	50.07%	100.0 ± 0.0

is 100 if $\gamma(\phi_3) = $ CONV, -100 if the training episode times-out when $\gamma(\phi_3) \neq$ CONV, and is zero otherwise. We employ epsilon-greedy Q-Learning—with training episodes lasting five simulated seconds. Over the course of training, exploratory actions were decreased linearly from 100% to 5%.

2) Learning Results: Learning performance, in terms of training time, is summarized in Table I for each method. Additionally, the final policy of each developmental context is evaluated on 50 episodes and the average reward is also reported and displayed graphically in Figure 3. The final policy for each setting performs equally well, resulting in nearly equal average rewards over 50 random trials. However, DC2 took the least amount of training time—requiring half of the interactions needed to learn an equivalent policy as the baseline. DC2 additionally makes the greatest use of the abstraction provided by skills, offering support that abstraction can accelerate learning.

The impact of hierarchically encoded control knowledge can be assessed by comparing the overall usage frequency of the composite Φ_7 skill to that of it's constituent actions ϕ_1 and $\phi_4 \lhd \phi_2$ in the final policy for DC1. We find the hierarchical Φ_7 skill is called 66.67% of the time on average and results in a 16.67% reduction in training time indicating that Φ_{SG} makes effective use of transferable control knowledge encoded in skills from previous stages of development.

B. Learning to Transport

The objective of the second stage of the development is to learn to transport objects in the same environment. This involves learning to coordinate searching for an object, forming an initial grasp, and moving the grasped object to a new Cartesian goal location; all while maintaining the grasp with respect to external perturbations.

We also introduce a new skill Φ_{STG} which combines the Φ_{SG} skill with the primitive $\phi_4 \lhd \phi_5$ in order to **S**earch for and then **T**rack a **G**rasp whenever all the contacts are on the ball and Φ_{SG} is converged.

1) Experimental Settings: We consider each of the five experimental settings listed below:

- **Baseline #2 (B2)**: In this setting, we consider primitive actions, and the previously existing learned policies: $\Psi_1 = \{\phi_3, \phi_4 \lhd \phi_5, \Phi_6, \Phi_7, \phi_1 \lhd \phi_4 \lhd \phi_5\}$. This setting establishes a baseline for comparing different choices for Ψ_1.
- **Developmental Context #3 (DC3)**: Next, we consider the impact of the Φ_{SG} skill and the control knowledge it contains: $\Psi_1 = \{\phi_3, \phi_4 \lhd \phi_5, \Phi_6, \Phi_7, \Phi_{SG}, \phi_1 \lhd \phi_4 \lhd \phi_5\}$.
- **Developmental Context #4 (DC4)** We remove the constituent actions and skills of Φ_7 and Φ_{SG} to evaluate the impact of using these control abstractions: $\Psi_1 = \{\Phi_7, \phi_4 \lhd \phi_5, \Phi_{SG}, \phi_1 \lhd \phi_4 \lhd \phi_5\}$
- **Developmental Context #5 (DC5)**: We evaluate the impact of a developmental stage in which the transitions between Φ_{SG} and $\phi_4 \lhd \phi_5$ have been implicitly captured in the Φ_{STG} skill: $\Psi_1 = \{\phi_4 \lhd \phi_5, \Phi_{SG}, \Phi_{STG}, \phi_1 \lhd \phi_4 \lhd \phi_5\}$.
- **Developmental Context #6 (DC6)**: Finally, in this most restrictive setting we assess the impact of removing the constituent skills and primitives for Φ_{STG}: $\Psi_1 = \{\Phi_{STG}, \phi_1 \lhd \phi_4 \lhd \phi_5\}$

The absorbing states are defined by $\{\gamma(\phi_1) = \text{CONV} \wedge \gamma(\phi_4) = \text{CONV} \wedge \gamma(\phi_5) = \text{CONV}\}$, indicating the object is grasped at the goal location in the environment. In each developmental context the reward R is 100 in the absorbing state, -100 if the training episode times-out outside of the absorbing state, and is zero otherwise. As before, we employ epsilon-greedy Q-Learning, with training episodes lasting ten simulated seconds, with the same linear exploration decay rate as in the previous experiment.

2) Learning Results: The learning performance of each of the five considered settings can be seen in Figure 4, as well as in Table II. As before, the final policy of each approach performed equally well. However, it is clear that the settings that exploit a highly-restrictive Ψ_1, vastly outperform the baseline setting with respect to training time. Performing the same action selection analysis as before for DC3, which introduces the hierarchical Φ_{SG} skill, we see that the Φ_{SG} skill is called a total of 40% of the time in comparison to it's subsumed actions. This, coupled with a reduced training time compared to the baseline, once again indicates that learning algorithms can exploit the state-space abstraction provided

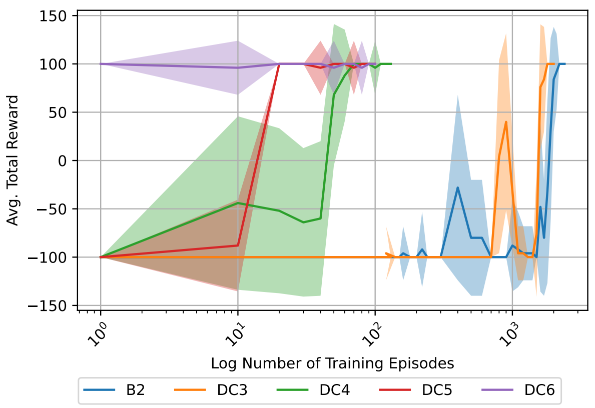

Fig. 4. Learning performance of Φ_T skill across all experimental settings. The average (and standard deviation) episodic reward across 50 trials is reported as a function of time. The x-axis is in log scale.

TABLE II
EXPERIMENTAL RESULTS FOR Φ_T LEARNING EXPERIMENTS

Approach	**Train Time** (min:sec)	**% Improvement**	**Avg. Reward ± Std**
B2	399:49	-	100.0 ± 0.0
DC3	333:10	16.67%	100.0 ± 0.0
DC4	21:30	94.62%	100.0 ± 0.0
DC5	16:30	95.87%	100.0 ± 0.0
DC6	1:30	99.62%	100.0 ± 0.0

by skills from previous developmental stages to accelerate acquiring new behaviors, even in higher-dimensional decision spaces.

When comparing the top three performing settings, removing all the low-level actions/skills that are subsumed by a later hierarchical skill accelerated learning rates dramatically; resulting in an improvement in training time of greater than 94% for each. This is due to the fact that removing these subsumed actions results in a significantly smaller state-space, requiring less exploration to learn. However, overall performance of the final policy suggests that these reduced-state MDP's do not compromise the robot's ability to learn to successfully complete the task.

Lastly, we compare the cumulative training time for each of the developmental contexts for the Φ_T skill. Cumulative training time includes the Φ_T training time from the second stage, as well as the training time for the Φ_{SG} policy from the previous level of the curriculum, as appropriate[3]. The results of this comparison can be seen in Figure 5. The three approaches that most heavily rely on the abstraction provided by control policies from previous developmental stages took significantly less time to learn the final Φ_T skill. This is most apparent in DC6, which spent 93.52% of it's training time across both stages learning the Φ_{SG} skill. As a result of the large amounts of control knowledge encoded in the Φ_{STG} skill, learning Φ_T is dramatically simpler, as all the agent needs to learn is to coordinate the execution of two abstract actions across a maximum of nine total states.

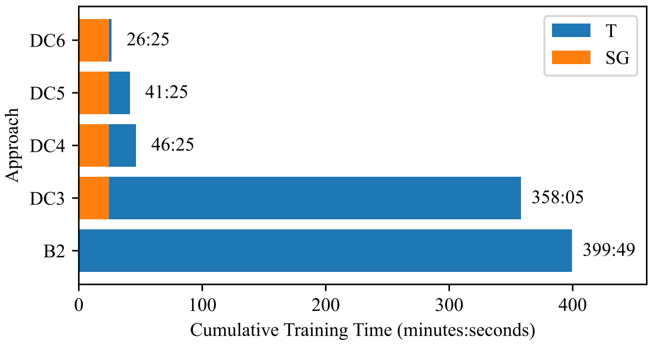

Fig. 5. Total elapsed training time across both stages of the developmental curriculum for each approach. The approach number matches that of the appropriate description in section III-B1. Here SG denotes the time spent training the Φ_{SG} skill, and T is the time spent training the Φ_T skill.

IV. Generalization Experiment

In many instances, the hierarchical policies constructed under the control-basis framework generalize to changes in the interaction context, without the need for expensive additional training. This is due to the high-level policy relying on the robustness of the control knowledge encapsulated in their supporting skills. Therefore, high-level policies are able to rely on the actions they subsume to absorb the changes at run-time, enabling these policies to adapt to changes in the domain on the fly.

To demonstrate the Φ_T policy's robustness, we perform trials of the Φ_T skill as described in Section III-B1—but now introduce two changes in the interaction context. First, three randomly placed obstacles are added into the environment at the start of each episode. The obstacles introduce both navigation obstructions and partial observability, however they are not large enough to entirely occlude the objects. Further, three additional object geometries, a triangle, pentagon, and spade (Figure 1) are introduced to evaluate the policies ability to adapt to novel objects. Other than these additions, the experimental conditions remain the same as in training for this skill.

TABLE III
EXPERIMENTAL RESULTS FOR Φ_T GENERALIZATION EXPERIMENT

Object	Avg. Reward ± Std	Success Rate
Circle	88.0 ± 47.497	94.0%
Triangle	92.0 ± 39.192	96.0%
Pentagon	80.0 ± 60.0	90.0%
Spade	96.0 ± 28.0	98.0%
Total	91.334 ± 38.946	94.5%

We perform the Φ_T training task in a randomly obstructed environment 50 times for each object. The results of this experiment, as seen in Table III, demonstrate that the Φ_T policy is able to successfully adapt to both new object geometries and cluttered environments. Despite the average episodic return

[3] For this comparison, we only consider the training time of the DC2 Φ_{SG} experimental setting.

and success rate for the circle object being lower than that for the unobstructed environment, out of the three unsuccessful circle trials, only one was the result of an unsuccessful grasp. In the other two trials, *Roger* ran out of time while navigating around the obstacles to transport the object to the origin. This is also the case for one out of the two unsuccessful triangle trials, and three of the five unsuccessful pentagon trials. We hypothesize that with a longer time limit before the episode is terminated, we would see fewer failures, but do not do so to mimic the restrictions placed during the learning phase.

V. Conclusion

Most skill learning literature focuses on developing intelligent algorithms for managing the complexity of learning problems. However, embodied robots often introduce a high degree of learning complexity that requires a prohibitively large amount of training time. In addition, the skills learned with these approaches are often non-generalizable, requiring significant amounts of additional training to adapt to new domains.

The approach presented in this work avoids these limitations by emulating a developmental curriculum in which an embodied and situated agent learns hierarchical control policies that model behavior as the sequential activation of task-independent objective functions. By limiting the sensor and motor resources available at each stage of development, this system is able to learn to locate, grasp, and transport objects in an unobstructed environment with up to an order of magnitude fewer interactions than the non-hierarchical approach in the most restrictive settings. The resulting high-level control policy is robust to changes in interaction context and was shown to be able to successfully adapt to transporting novel object geometries in cluttered environments without additional training.

One limitation is that there is currently no mechanism to automatically decompose complex tasks into an appropriate developmental curriculum. To address this shortcoming, we plan to investigate such methods taking inspiration from the work of [11], [16], [17]. Additionally, we plan to explore transferring high-level policies into large number of control contexts without retraining by modifying the skills which make up its hierarchical control stack.

Acknowledgments

The authors would like to thank members of the Laboratory for Perceptual Robotics at UMass Amherst for their support. This work was supported by an Early Stage Innovations grant from NASA's Space Technology Research Grants Program.

References

[1] J. Canny, B. R. Donald, J. Reif, and P. G. Xavier, "On the complexity of kinodynamic planning," Cornell University, Tech. Rep., 1988.

[2] T. Elliott and N. R. Shadbolt, "Developmental robotics: manifesto and application," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 361, no. 1811, pp. 2187–2206, 2003.

[3] E. Thelen, "Dynamic systems theory and the complexity of change," *Psychoanalytic dialogues*, vol. 15, no. 2, pp. 255–283, 2005.

[4] M. Huber, W. S. MacDonald, and R. A. Grupen, "A control basis for multilegged walking," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 4. IEEE, 1996, pp. 2988–2993.

[5] M. Huber and R. A. Grupen, "Learning to coordinate controllers-reinforcement learning on a control basis," in *IJCAI*, 1997, pp. 1366–1371.

[6] S. Hart and R. Grupen, "Natural task decomposition with intrinsic potential fields," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2007, pp. 2507–2512.

[7] G. Metta, G. Sandini, L. Natale, and F. Panerai, "Development and robotics," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*. Citeseer, 2001, pp. 33–42.

[8] J. Law, M. Lee, M. Hülse, and A. Tomassetti, "The infant development timeline and its application to robot shaping," *Adaptive Behavior*, vol. 19, no. 5, pp. 335–358, 2011.

[9] N. A. Bernstein, M. L. Latash, and M. T. Turvey, *Dexterity and its development*. Psychology Press, 2014.

[10] M. Riedmiller, R. Hafner, T. Lampe, M. Neunert, J. Degrave, T. Wiele, V. Mnih, N. Heess, and J. T. Springenberg, "Learning by playing solving sparse reward tasks from scratch," in *International conference on machine learning*. PMLR, 2018, pp. 4344–4353.

[11] S. Sukhbaatar, Z. Lin, I. Kostrikov, G. Synnaeve, A. Szlam, and R. Fergus, "Intrinsic motivation and automatic curricula via asymmetric self-play," *arXiv preprint arXiv:1703.05407*, 2017.

[12] C. Florensa, D. Held, X. Geng, and P. Abbeel, "Automatic goal generation for reinforcement learning agents," in *International conference on machine learning*. PMLR, 2018, pp. 1515–1528.

[13] F. Foglino, C. C. Christakou, and M. Leonetti, "An optimization framework for task sequencing in curriculum learning," in *2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2019, pp. 207–214.

[14] F. L. Da Silva and A. H. R. Costa, "A survey on transfer learning for multiagent reinforcement learning systems," *Journal of Artificial Intelligence Research*, vol. 64, pp. 645–703, 2019.

[15] S. Narvekar, J. Sinapov, M. Leonetti, and P. Stone, "Source task creation for curriculum learning," in *Proceedings of the 2016 international conference on autonomous agents & multiagent systems*, 2016, pp. 566–574.

[16] F. L. D. Silva and A. H. R. Costa, "Object-oriented curriculum generation for reinforcement learning," in *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, 2018, pp. 1026–1034.

[17] S. Narvekar, J. Sinapov, and P. Stone, "Autonomous task sequencing for customized curriculum design in reinforcement learning." in *IJCAI*, 2017, pp. 2536–2542.

[18] Y. Nagai, K. Hosoda, and M. Asada, "How does an infant acquire the ability of joint attention?: A constructive approach," 2003.

[19] G. Butterworth and N. Jarrett, "What minds have in common is space: Spatial mechanisms serving joint visual attention in infancy," *British journal of developmental psychology*, vol. 9, no. 1, pp. 55–72, 1991.

[20] M. H. Lee, Q. Meng, and F. Chao, "Staged competence learning in developmental robotics," *Adaptive Behavior*, vol. 15, no. 3, pp. 241–255, 2007.

[21] A. Weber, C. P. Martin, J. Torresen, and B. C. da Silva, "Identifying reusable early-life options," in *2019 Joint IEEE 9th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)*. IEEE, 2019, pp. 335–340.

[22] P. M. Churchland, *Matter and consciousness*. MIT press, 2013.

[23] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," 1986.

[24] H. J. S. Feder and J.-J. Slotine, "Real-time path planning using harmonic potentials in dynamic environments," in *Proceedings of International Conference on Robotics and Automation*, vol. 1. IEEE, 1997, pp. 874–881.

[25] C. I. Connolly, "Harmonic functions and collision probabilities," *The International Journal of Robotics Research*, vol. 16, no. 4, pp. 497–507, 1997.

[26] J. A. Coelho Jr and R. A. Grupen, "A control basis for learning multifingered grasps," *Journal of Robotic Systems*, vol. 14, no. 7, pp. 545–557, 1997.