Breadth vs Depth: Benchmarking Generalists and Specialists in Robot Agility Learning

Sayan Mondal
CMU RI

Praveen Venkatesh
CMU RI

Siddharth Saha
CMU RI

Khai Nguyen
CMU MechE

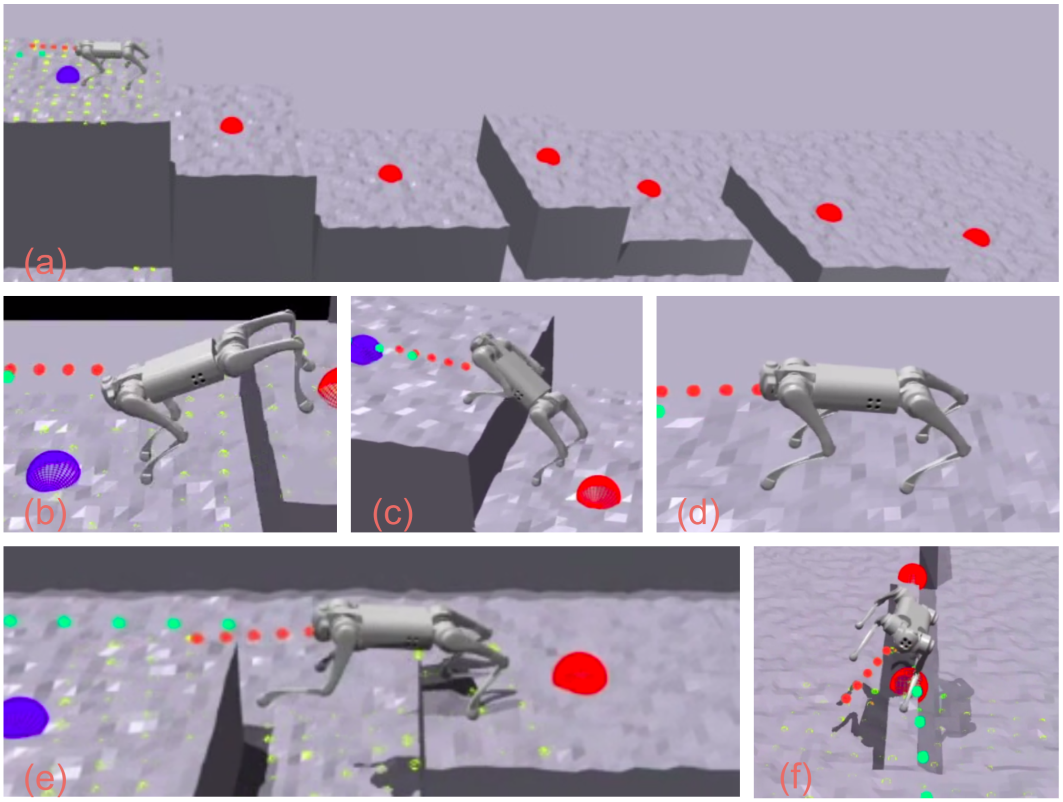

Figure 1: We introduce a benchmark for learning robot agility on low-priced robots. Our study assesses both specialist and generalist policies, enabling robots to climb up and down high obstacles, leap over large gaps, and walk over narrow beams only using one-side legs. Videos are provided at this link.

Abstract: In the realm of deep reinforcement learning, achieving generalization over unforeseen variations in the environment often necessitates extensive policy learning across a diverse array of training scenarios. Empirical findings reveal a notable trend: an agent trained on a multitude of variations (termed a generalist) exhibits accelerated early-stage learning, but its performance tends to plateau at a suboptimal level for an extended period. In contrast, an agent trained exclusively on a select few variations (referred to as a specialist) frequently attains high returns within a constrained computational budget. To reconcile these contrasting advantages, we experiment with various combinations of specialists and generalists in the quadrupedal locomotion setting. Our investigation delves into determining the impact of each skill when they are trained to be specialists and the impact of combining them together into creating a more generalist agent.

1 Motivation

In pursuit of an optimal training strategy, our project navigates the delicate balance between cultivating a versatile, generalist policy proficient in a diverse range of tasks and the potential pitfalls associated with its implementation. The inherent challenge lies in the risk of developing a robust policy that, while competent in numerous trades, may master none or only a few due to catastrophic forgetting.

Conversely, our investigation explores the viability of training specialized policies independently. This approach aims to curtail the potential influence of one skill on another, posing a compelling yet open research problem regarding the positive or negative impacts of training one skill on another. For example, the beneficial impact of sports on writing, attributed to the development of hand muscles, contrasts with its potentially adverse effects on singing, where excessive shouting could strain the vocal cords.

In this report, we undertake a comprehensive exploration to gain insights into the comparative performance of specialist skills versus their generalist counterparts. This endeavor is anchored in a meticulous study conducted on a simulated quadrupedal Unitree Go1 robot. Our intent is to distill a nuanced understanding of the most effective approaches to training various skill sets.

The ramifications of this research extend to the practical optimization of quadrupedal locomotion performance. By discerning the best practices for training some skills together and others separately, we aim to harmonize the strengths of both approaches, thereby achieving the pinnacle of performance in quadrupedal locomotion.

2 Prior Work

The current landscape of research in agile locomotion through reinforcement learning has seen several recent methodologies aimed at addressing this intricate challenge. A notable contribution from CMU [1] introduces a two-stage approach, focusing on cultivating a single generalist policy for a quadruped capable of autonomously navigating a parkour course. In parallel, a contemporaneous endeavor [2] adopts a different strategy, emphasizing the acquisition of a repertoire of skills and subsequent distillation into a unified generalist policy.

An alternative perspective is presented by [3], wherein a systems-based approach is proposed for quadruped locomotion. Here, a navigation module utilizes local terrain information to selectively employ specialist policies. Furthermore, [4] delves into the benchmarking of agile locomotion in quadrupeds, employing a hand-crafted waypoint-based mechanism to skill selection based on location and the quadruped's current position. Their findings showcase enhanced performance with specialist agents compared to generalist agents.

While these methodologies exist independently across various papers, a critical gap remains unexplored—there exists no comprehensive study elucidating the advantages or disadvantages of a single generalist policy versus the alternative approach of selecting a specialist policy from a bank of skills. This project endeavors to address this gap by probing into the fundamental question: can the design of a policy selection mechanism empower robots to execute high-agility tasks more effectively than a single generalist policy? Through this exploration, we aim to contribute insights into the nuanced dynamics of policy selection mechanisms for optimal performance in agile locomotion tasks.

3 Approach

3.1 Specialist Approach

3.1.1 Specialist Controller Policies

Our approach involves a meticulous training regimen tailored to enhance the robot's proficiency in specific locomotion tasks, fostering excellence in navigating terrains unique to each skill. This

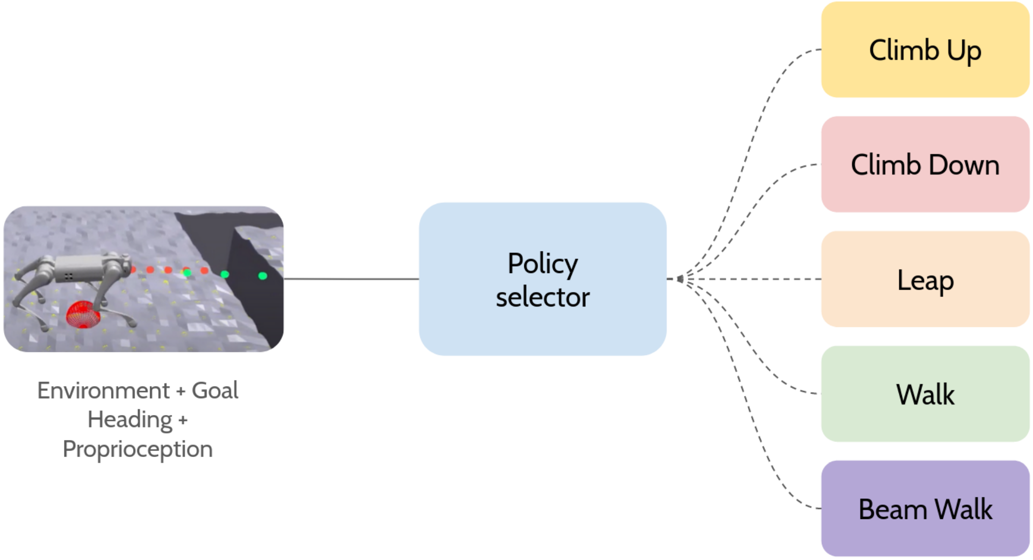

Figure 2: Policy Selector framework to select among the library of specialist skills depending on the local terrain and the heading direction information.

specialist-centric methodology is designed to lay the foundation for robust baselines in individual locomotion skills. By prioritizing the intricacies of each distinct task, the robot develops expertise in overcoming specific challenges, demonstrating a nuanced understanding of diverse terrains and obstacles.

During the training of each specialist controller policy, we intentionally limit exposure to skill-specific terrains, ensuring that each controller remains focused on its designated skill set without unwarranted generalization across other skills. This targeted training strategy aims to hone the specialist's capabilities and maximize its effectiveness in executing specific locomotion tasks.

3.2 Selector Policy

In order to translate the acquired specialization into practical utility during execution, we implement a sophisticated high-level navigation module. This module is governed by a learned selector policy, tasked with extracting terrain features and discerning the most fitting specialist skill for the given context. At each time step, the selector policy exclusively outputs the index corresponding to the identified specialist skill. Subsequently, the actual locomotion commands are generated by the executed specialist skill.

Upon execution of the specialist controller, the robot receives rewards, and we employ the Proximal Policy-gradient Optimization approach to back-propagate these rewards. This feedback loop enables the selector policy to learn to recognize obstacles and determine the optimal specialist policy based on the received rewards. This iterative process ensures that the selector becomes adept at making informed decisions, refining its ability to select the most effective specialist skill in varying environmental conditions. This version of the policy selector is the **RL+PPO based Selector Policy**.

The other version of the policy selector is the **Supervised Learning based Selector Policy**, where it is trained using Supervised Learning.

3.2.1 Implementation

We train a network that chooses which skill to use at which point using depth information [3]. This is the policy selector network. We model our task as a classification problem, where our network architecture for the selector policy is a 3-layer MLP where the inputs are scandots, proprioception, and heading, and we have k output classes, where each class corresponds to a policy of interest. The scandots contain privileged information about the height map surrounding the robot that is directly obtained from Isaac Gym. The scandots capture the local terrain information.

Since we have complete control over the generation of terrain, we can also define the locations at which a particular policy needs to be used. We name this an oracle policy. An image of the generated policies can be viewed in fig 3.

We use five base skills (Fig. 1 sequentially):

- Climb down - The robot learns only to climb down.
- Climb up - The robot learns only to climb up.
- Walk - The robot learns to walk efficiently and quickly in varied terrains.
- Leap - The robot learns to jump across a gap.
- Beam walk - here, we provide an environment where there is a thin beam, and the robot has to struggle to place all its legs on the beam for stable walking. This is a hard skill to execute, and we notice very interesting emergent behaviors from the robot during training.

Our approach uses the following methods to train the selector network.

- Trained using the same PPO formulation used during training of low-level skills.
- Trained using the same PPO formulation used during training, with rewards replaced with an oracle reward obtained from the environment.
- Supervised training using data collected from an oracle policy.

3.2.2 RL+PPO based Selector Policy

We perform two experiments for reward shaping:

- Rewards are as per original paper
- We retain only an oracle reward — an L1 norm where a correct action is rewarded, and an incorrect action is penalized with the same weightage.

3.2.3 Supervised Learning based Selector Policy

- First we collect all of the data required through multiple hundred thousands runs on the terrain, and collect all oracle policies required.
- Next we train a 132(scan dots) x128x128xnpolicies MLP with this data
- Then we use this model at runtime, and run it perfectly.
- It works really well compared to everything else. The only issue is that it may not generalize to completely to new environments as it is fully out-of-distribution.

VIDEO LINK

4 Results

4.1 Experimental Setup

We use Isaac Gym to train our agents and evaluate their performance. We use the Unitree Go1 URDF and generate terrains inside Isaac Gym to learn walking policies.

4.2 Experimentation

- Action stacking (repeating the same action for multiple timesteps) worsened the performance primarily because the timesteps would change midway a critical skill. However, if tuned well, this could be an interesting parameter to tune. The action reward helps instead.
- Curriculum is crucial for the robot to learn anything.
- We get unusual behaviors at times -
    - In one run, the robot learns to fake a climb down policy by abandoning the leap policy midway, and executing climb-up. This sort of behvaiour, while is acceptable, is not the intended behavior, and is sort of emergent within the framework that we have defined.
    - In another run, the robot learns to leap instead of climb down. This is likely because it moves fast and somewhat reaches all the goals eventually primarily because leap and climb down are overlapping skills. VIDEO LINK
    - Since the beam walk has not been trained in terrains that contain climbing up, we obtain modal collapse where the robot goes close to the upstep and makes a weird kicking motion to set it back far. VIDEO LINK
- PPO doesn't seem to give promising results when we add the beam walking skill. We encounter mode-collapsing issues very frequently. However, we do notice that it does work somewhat well, even though the skills keep flickering. VIDEO LINK
- It makes sense to use the local terrain information simply. Since the scandots are constrained, we can use supervised learning from the Oracle policy to make decisions. The advantage of an RL framework comes in the fact that we can vary terrains during training and obtain policies that work from observation where generating an oracle network is non-trivial. However, it is difficult to avoid some of the issues that were stated above.

4.3 Insights on Selector Network

We present the results of our experimentation in Table 1.

- The order of terrain difficulty for learning selector is climb = leap = plank $<$ climb and leap.
- The Oracle and SL-based Policy Selector (PS-SL-5) outperform the generalists in skill-specific environments (Gap, Beam, Climb)
- However, the generalist outperforms in more difficult and dynamic terrains (Climb+Gap). We note that the noise and perturbations present during final evaluations heavily affect the performance of the selector network in predicting the policy to execute, leading to a reduction in performance.
- The RL+PPO-based Selector (PS-RL-3) does not perform well as it keeps switching its predicted policy and does not know what to do in certain scenarios. We have frequently run into mode collapse, where it learns to predict one single policy at all times.
- However, we see that the RL+PPO-based Selector outperforms the generalist in reaching waypoints in the hardest terrain (Climb+Leap).
- We sometimes see the RL+PPO-based Selector leap instead of climbing down. This indicates a high overlap between the two skills and could potentially be fixed using better reward shaping.

5 Conclusion, Limitations and Future Work

Our work establishes a baseline for benchmarking various learning-based approaches in robot agility. We propose three baselines: i) specialist policies that learn single skills using on-policy RL; ii) a selector policy that learns to select prior specialist policies at the right time; and iii) a true generalist policy that learns to handle all tasks at once. We further investigate the performance of the selector policy which can be trained via on-policy RL or imitating an oracle.

Despite this progress, the ultimate goal remains unresolved, leaving considerable room for advancing agility through the acquisition of more useful behaviors, as well as enhancements in speed and robustness. We contend that closing this gap requires a collective endeavor from the research community, and our benchmark stands poised to significantly contribute to the progress of athletic intelligence. One limitation of our current work is the reliance on privileged information, including terrain scandots and properties. Previous research has successfully bridged the sim2real gap through a two-phase student-teacher training approach, wherein a teacher trained with privileged scandots information is distilled to a student equipped only with on-board sensors such as depth cameras. We propose the logical next step of distilling both low-level locomotion skills and high-level navigation controllers to seamlessly operate with on-board sensors, subsequently benchmarking their performance on real robots. An extension to our work involves incorporating the approach proposed in [2], wherein all specialists and potentially the selector policy are distilled into a unified generalist policy (referred to as post-generalist, in contrast with the true-generalist in [1]). Another limitation pertains to the learning settings for each approach. Establishing quantitatively identical training procedures, including hyper-parameters, proves challenging and raises concerns about fairness. While previous work typically emphasizes reporting their best results based on different high-level performance criteria and hardware setups, an equally compelling avenue for future research is to propose a standardized training procedure. This would facilitate the evaluation of different approaches based on algorithmic criteria, such as performance per training iteration, providing valuable insights into their suitability for particular applications and resources.

References

[1] X. Cheng, K. Shi, A. Agarwal, and D. Pathak. Extreme parkour with legged robots. *arXiv preprint arXiv:2309.14341*, 2023.

[2] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao. Robot parkour learning. *arXiv preprint arXiv:2309.05665*, 2023.

[3] D. Hoeller, N. Rudin, D. Sako, and M. Hutter. Anymal parkour: Learning agile navigation for quadrupedal robots. *arXiv preprint arXiv:2306.14874*, 2023.

[4] K. Caluwaerts, A. Iscen, J. C. Kew, W. Yu, T. Zhang, D. Freeman, K.-H. Lee, L. Lee, S. Saliceti, V. Zhuang, et al. Barkour: Benchmarking animal-level agility with quadruped robots. *arXiv preprint arXiv:2305.14654*, 2023.

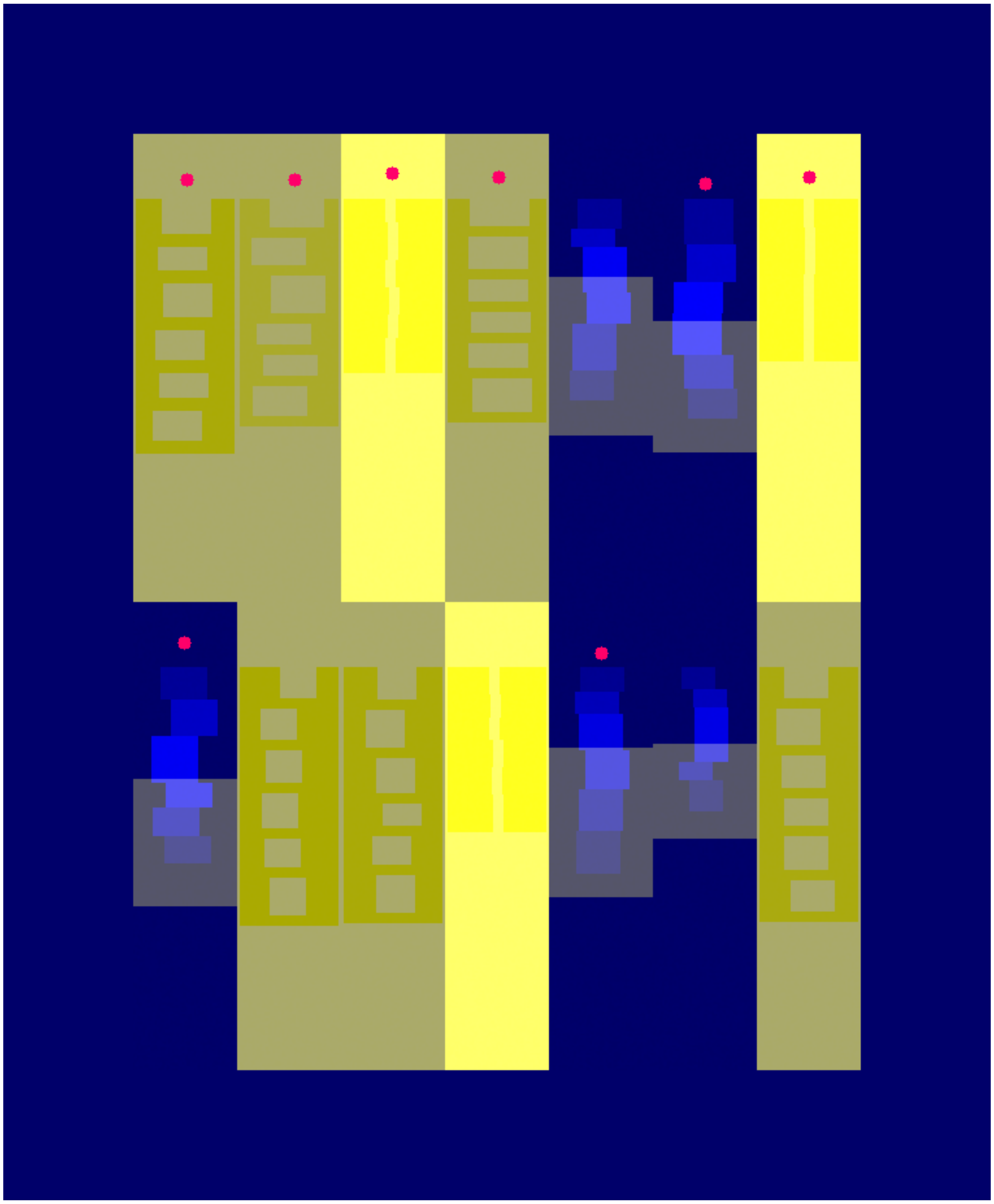

Figure 3: Oracle Policy Map, Top Down View. Pink dots represent agents, and colored overlays represent policy regions. A walk policy is an oracle if there is a low deviation in the scandots.

Environment	Policy	Reward↑	Ep. Length↑	%WP↑	Violation↓
Climb+Gap	CU	07.18±40.51	307.92±187.02	0.36±0.22	0.07±0.28
Climb+Gap	CD	03.32±00.75	985.03±093.54	0.14±0.00	1.09±0.59
Climb+Gap	Leap	01.05±00.80	112.35±093.35	0.12±0.06	0.10±0.32
Climb+Gap	Beam	16.89±06.11	947.51±203.84	0.14±0.03	0.07±0.34
Climb+Gap	Walk	00.43±00.58	682.87±414.17	0.06±0.07	0.05±0.28
Climb+Gap	Climb	08.54±04.07	306.80±127.52	0.51±0.26	0.10±0.36
Climb+Gap	Oracle	13.55±20.03	708.98±201.10	**0.90±0.20**	-
Climb+Gap	PS-RL-3	14.75±18.62	626.47±159.17	0.87±0.20	-
Climb+Gap	PS-SL-5	11.68±12.71	541.27±259.42	0.71±0.35	-
Climb+Gap	GEN-3	15.04±05.90	553.03±169.21	0.80±0.28	0.05±0.23
Climb+Gap	GEN-5	14.21±06.19	542.97±193.59	0.81±0.30	0.08±0.30
Gap	CU	05.38±02.48	211.50±102.20	0.25±0.10	0.07±0.28
Gap	CD	07.39±01.52	261.30±088.78	0.28±0.02	0.09±0.34
Gap	Leap	17.78±09.36	548.58±255.08	0.86±0.18	0.01±0.07
Gap	Beam	07.28±02.37	323.98±194.43	0.28±0.05	0.19±0.58
Gap	Walk	00.73±00.91	637.31±419.90	0.07±0.09	0.01±0.09
Gap	Climb	11.20±05.62	393.93±205.04	0.59±0.34	0.04±0.21
Gap	Oracle	20.55±07.65	632.56±154.97	**0.90±0.22**	-
Gap	PS-RL-3	18.66±17.73	644.01±159.28	0.89±0.23	-
Gap	PS-SL-5	20.82±07.25	639.80±148.34	**0.90±0.21**	-
Gap	GEN-3	15.10±05.90	524.67±173.22	0.76±0.30	0.02±0.18
Gap	GEN-5	14.20±05.46	506.09±167.77	0.81±0.29	0.03±0.18
Beam	CU	03.73±02.18	208.82±173.95	0.23±0.14	0.04±0.21
Beam	CD	06.11±02.34	300.53±209.81	0.30±0.11	0.10±0.30
Beam	Leap	03.13±01.35	213.37±227.11	0.25±0.10	0.05±0.25
Beam	Beam	08.58±30.97	317.44±140.27	**0.62±0.27**	0.00±0.00
Beam	Walk	00.44±00.58	602.87±431.72	0.02±0.05	0.01±0.14
Beam	Oracle	02.23±03.63	215.92±088.28	0.39±0.25	-
Beam	PS-RL-3	-	191.57±168.91	0.28±0.04	-
Beam	PS-SL-5	03.56±04.30	255.18±102.25	0.58±0.29	-
Beam	GEN-3	03.47±00.67	225.23±224.72	0.16±0.05	0.02±0.14
Beam	GEN-5	03.45±00.88	298.69±302.03	0.25±0.07	0.05±0.22
Climb	CU	10.90±07.81	461.84±277.98	0.61±0.42	0.06±0.24
Climb	CD	03.72±01.07	985.04±083.70	0.14±0.00	1.22±0.60
Climb	Leap	00.91±00.57	092.13±039.12	0.12±0.06	0.05±0.22
Climb	Beam	22.80±05.58	953.77±203.10	0.14±0.03	0.13±0.45
Climb	Walk	01.04±01.86	907.89±266.62	0.05±0.07	0.07±0.28
Climb	Climb	16.68±03.69	540.25±097.64	**0.99±0.07**	0.02±0.15
Climb	Oracle	21.67±05.31	659.84±115.50	**0.99±0.09**	-
Climb	PS-RL-3	12.75±09.86	567.87±132.48	0.89±0.17	-
Climb	PS-SL-5	20.83±09.04	658.18±128.93	0.98±0.13	-
Climb	GEN-3	15.37±03.70	548.36±103.33	0.96±0.16	0.03±0.19
Climb	GEN-5	15.49±04.79	561.68±144.60	0.93±0.22	0.06±0.27

Table 1: We test our method against several baselines and ablations in the simulation. We measure the reward, episode length, success rates of reaching all waypoints, and edge violation of every policy in different environments averaged across X trials and Y random seeds. Symbols: CU – climb up, CD – climb down, PS-RL-3 – policy selector of 3 skills (leaping, climbing, walking) trained by RL, PS-SL-5 – policy selector of 3 skills (leaping, climbing-up, climbing-down, walking, beam-walking) trained by SL, GEN-3 – true generalists of 3 skills, GEN-5 – true generalist of 5 skills.