Learning to Anticipate Physical Failures via Agentic Adversarial Harness

Zhiyuan Li[1], **Dingsheng Liu**[1], **Amir Rasouli**[2]
[1]University of Toronto, [2]Huawei Technologies Canada

a2h-robot.github.io

Keywords: Agentic robotics, code as policies, adversarial task design

Figure 1: **A^{2}H overview.** In simulation, a coding agent iteratively generates policies, adversarially designs task variations to expose failures, repairs policies, and reflects on the failures (1–4). The resulting experiences are distilled into a compact harness H (5), which guides a fresh agent in zero-shot policy generation for unseen real-world tasks (6).

1 Introduction

Recent coding agents have demonstrated promising capabilities in robotic control through the code-as-policies (CaP) paradigm in simulation benchmarks [1, 2, 3, 4]. However, achieving high success rates and robustness requires recursive self-improvement (RSI) across multiple attempts, a process that is difficult to accommodate in real-world deployment [2, 5, 6], where each failed attempt requires a manual reset and risks damaging the robot or its surroundings. Robots may encounter unseen environments, lighting conditions, objects, and sensor configurations, each introducing distinct sources of uncertainty. These challenges are further amplified in complex tasks, where agents must account for numerous physical constraints and interactions while pursuing the specified goal. Existing approaches mitigate these challenges by shifting trial-and-error to simulation through real2sim2real pipelines [7, 8], or by transferring experience across tasks through reusable skill libraries and memories [3, 9, 10]. However, the former requires costly scene reconstruction and depends on simulation fidelity, while the latter is limited by the coverage of previously acquired skills and experiences.

To this end, we propose **Agentic Adversarial Harness (A^{2}H)** that learns knowledge of physical failure modes through self-adversarial task design and iterative reflection, enabling failure anticipation without task-specific execution feedback. Through this process the agent accumulates transferable experience, which can be distilled into a compact harness to guide robust program generation at deployment. We evaluate **A^{2}H** on four real-robot tasks with diverse physical constraints and show that a harness learned in simulation transfers to the real world.

2 Implementation Details

Training. As illustrated in Figure 1, **A^{2}H** alternates a coding agent between two roles during training: a solver that generates code-based policies to achieve a target success rate τ_{succ}, and a task designer that adversarially generates task variations to reduce the success rate of previously generated policies below a failure threshold τ_{break}. Through iterative adversarial task design and policy refinement, a knowledge bank accumulates reflections on identified weaknesses, their causes, and

Late-Breaking Results, 10th Conference on Robot Learning (CoRL 2026), Austin TX, USA.

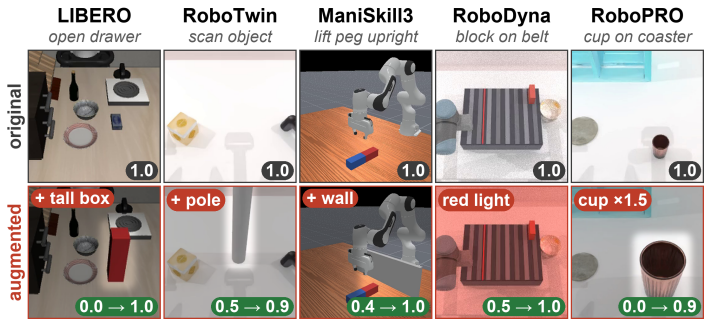

Figure 2: **Agent-designed augmentations.** Top: original task and program success. Bottom: augmentation (rest dimmed) and success of the frozen → repaired program.

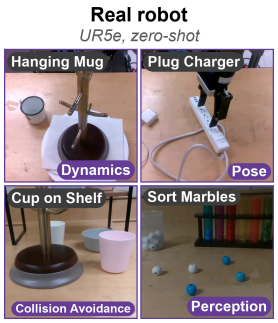

Figure 3: **Real-robot tasks.** Each task (top label) tests one challenge (bottom label).

Table 1: Real-robot experiment results. Pass@1 and pass@4 show first-attempt and within four attempts success rates, and Cost is the API cost of the tokens consumed. Best results are in bold.

	Baseline			RSI Harness			$\mathbf{A^2H}$		
Task	pass@1	pass@4	Cost ($)	pass@1	pass@4	Cost ($)	pass@1	pass@4	Cost ($)
Sort Marbles	20%	45%	8.4	40%	55%	15.9	**100%**	**100%**	12.0
Cup on Shelf	0%	35%	14.5	20%	50%	20.2	**60%**	**100%**	19.1
Plug Charger	0%	15%	19.5	0%	20%	35.5	**60%**	**100%**	28.5
Hanging Mug	0%	15%	23.4	0%	25%	43.1	**20%**	**80%**	40.0
Average	5%	27.5%	16.5	15%	37.5%	28.7	**60%**	**95%**	24.9

corresponding fixes. These experiences are then distilled into a task-agnostic harness that provides a workflow for code-based policy generation and guidelines for anticipating physical failures before execution, such as clearing obstacles along manipulation paths and ensuring safe object release. We train $\mathbf{A^2H}$ on 25 tasks across five simulation benchmarks, LIBERO [11], RoboTwin 2.0 [12], ManiSkill3 [13], RoboDyna [14] and RoboPRO [15], covering diverse robots, simulators, and physical challenges, as shown in Figure 2. See the project page for full details.

Evaluation. We evaluate on four challenging real-robot tasks (Figure 3): hang a mug on a rotating holder (Dynamics), plug a charger into an inclined socket (Pose), manipulate cups and bowls on a cluttered shelf (Collision Avoidance), and sort marbles under low-light conditions (Perception). We compare agents in three settings: without a harness (Baseline), with a harness distilled from conventional RSI [16, 17] on the same 25 training tasks (RSI Harness), and $\mathbf{A^2H}$. Each setting is evaluated with four independent agents, and each agent's policy is tested over five trials per task.

3 Results and Conclusion

As shown in Table 1, $\mathbf{A^2H}$ consistently outperforms both baselines across all real-robot tasks. $\mathbf{A^2H}$ achieves 60% pass@1, improving over RSI Harness and Baseline by 45 and 55 percentage points, respectively. Remarkably, its first-attempt success rate exceeds the pass@4 of both baselines (37.5% and 27.5%), suggesting that the learned harness helps agents anticipate physical constraints before execution rather than relying on real-world trial and error to discover them. This advantage extends to multi-attempt execution, with $\mathbf{A^2H}$ achieving 100% pass@4 on three tasks and 80% on Hanging Mug, demonstrating robustness across diverse physical and perceptual challenges. Notably, $\mathbf{A^2H}$ also reduces average token cost by 13.2% compared with RSI Harness ($24.9 vs. $28.7), despite achieving substantially higher success rates. This suggests that self-adversarial harness learning provides more effective guidance than transferring experience from conventional RSI alone.

Conclusion. We proposed $\mathbf{A^2H}$, a method that learns knowledge of physical failure modes through self-adversarial task design and reflection. Experiments on real-robot demonstrated substantial improvements in zero-shot success rates and robustness over iterative execution, highlighting the potential of transferring failure anticipation from simulation to real-world robotic manipulation.

References

[1] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng. Code as Policies: Language model programs for embodied control. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[2] L. Fu, J. Yu, K. El-Refai, E. Kou, H. Xue, H. Huang, W. Xiao, G. Wang, D. Niu, L. Fei-Fei, G. Shi, J. Wu, S. S. Sastry, Y. Zhu, K. Goldberg, and L. J. Fan. CaP-X: A framework for benchmarking and improving coding agents for robot manipulation. In *International Conference on Machine Learning (ICML)*, 2026.

[3] R. Lu, Y. Wu, E. Kou, H. Lin, L. Fu, W. Xiao, A. Mandlekar, Y. Xu, G. Shi, K. Goldberg, A. Chen, M. Chowdhury, J. Kautz, Y. Zhu, L. J. Fan, and G. Wang. ASPIRE: Agentic /skills discovery for robotics. *arXiv preprint arXiv:2607.00272*, 2026.

[4] Z. Shen, H. You, Y. Liu, Z. Zheng, L. Zha, K. Yamazaki, M. Zhang, S. Huang, J. Sun, Q. Chen, L. He, K. Liu, H. Chang, K. Fragkiadaki, D. Shah, M. Schwager, P. Henderson, I. Abraham, and C. Xu. EmbodiedSWE: Coding agents for long horizon dexterous robotics. *arXiv preprint arXiv:2609.27308*, 2026.

[5] W. Xiao, J. Xie, T. Zhang, H. Lin, L. Fu, H. Xue, J. Lu, Y. Yang, C. Dai, Z. Wang, J. Wu, G. Wang, S. S. Sastry, K. Goldberg, L. J. Fan, Y. Zhu, and G. Shi. ENPIRE: Agentic robot policy self-improvement in the real world. In *Conference on Robot Learning (CoRL)*, 2026.

[6] Y. Lee, R. Nair, Q. Zhang, K. Lee, O. Khattab, and C. Finn. Meta-Harness: End-to-end optimization of model harnesses. *arXiv preprint arXiv:2603.28052*, 2026.

[7] M. Torne, A. Simeonov, Z. Li, A. Chan, T. Chen, A. Gupta, and P. Agrawal. Reconciling reality through simulation: A real-to-sim-to-real approach for robust manipulation. In *Robotics: Science and Systems (RSS)*, 2024.

[8] Y. Li, Y. Feng, S. Sun, J. Chen, H. Ren, B. Yang, W. Xu, Q. Wu, H. Cheng, and R. Xu. Agentic RSR: Real-to-sim-to-real through scene reconstruction and execution-grounded robot policies. *arXiv preprint arXiv:2610.10479*, 2026.

[9] T. Kagaya, S. Lakshmi, Y. Lou, T. J. Yuan, J. Karlekar, S. Pranata, N. Murakami, A. Kinose, and Y. You. Memory transfer planning: LLM-driven context-aware code adaptation for robot manipulation. *arXiv preprint arXiv:2509.24160*, 2025.

[10] X. Wang, B. Li, H. Wang, L. Zhang, A. Y. Cui, Z. Tang, S. Buch, and D. Xu. SEAR: Self-evolving agentic robotics. *arXiv preprint arXiv:2610.18792*, 2026.

[11] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

[12] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, X. Lin, Y. Ge, Z. Gu, W. Deng, Y. Guo, T. Nian, X. Xie, Q. Chen, K. Su, T. Xu, G. Liu, M. Hu, H.-a. Gao, K. Wang, Z. Liang, Y. Qin, X. Yang, P. Luo, and Y. Mu. RoboTwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

[13] S. Tao, F. Xiang, A. Shukla, Y. Qin, X. Hinrichsen, X. Yuan, C. Bao, X. Lin, Y. Liu, T.-k. Chan, Y. Gao, X. Li, T. Mu, N. Xiao, A. Gurha, V. N. Rajesh, Y. W. Choi, Y.-R. Chen, Z. Huang, R. Calandra, R. Chen, S. Luo, and H. Su. ManiSkill3: GPU parallelized robotics simulation and rendering for generalizable embodied AI. In *Robotics: Science and Systems (RSS)*, 2025.

[14] Z. Li, Yang, X. Zhao, and A. Rasouli. Benchmarking robotic manipulation under physically realistic dynamics. In *NeurIPS Workshop on Robot Learning with World Models: Capabilities, Frontiers, and Challenges*, 2026.

[15] Z. Li, C. Eret, X. Zhao, Y. Wu, and A. Rasouli. Data and evaluation protocols for robust robotic manipulation. In *RSS Workshop on Data-Centric Robotics*, 2026. URL https://robopro-bench.github.io/RoboPRO/.

[16] N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[17] P. Xia, R. Han, Z. Wang, Y. Chen, Y. Zhuang, Y. Lee, C. Huang, H. Yu, Z. CuiZhu, Y. Ming, H. Yao, B. Gokturk, T. Pfister, and C.-Y. Lee. RRSI: Regularized recursive self-improvement of agent harnesses. *arXiv preprint arXiv:2609.24972*, 2026.