A Framework for Curriculum Schema Transfer from Low-Fidelity to High-Fidelity Environments

Yash Shukla
Department of Computer Science
Tufts University
Medford, USA
yash.shukla@tufts.edu

Jivko Sinpov
Department of Computer Science
Tufts University
Medford, USA
jivko.sinapov@tufts.edu

***Abstract*—Emergence of Deep Neural Networks in Reinforcement Learning (RL) have enabled robots to learn a wide range of behaviors. Despite these advances, in many tasks, the number of interactions required to learn a policy are prohibitively expensive. This is infeasible in real world settings, where interactions are expensive. Curriculum learning tackles this problem, but generating a curriculum still requires costly interactions in the environment. Learned behaviors transferred from simulation to a real world scenario suffer from 'Sim2Real' gap because of the differences in the two domains. In this work, we provide a holistic framework aimed at generating a low-fidelity (LF) environment for a complex robotic high-fidelity (HF) environment, optimizing the curriculum in the LF environment and transferring it to the HF environment. We demonstrate that our approach improves learning performance in robotic navigation and manipulation domains, by requiring fewer interactions to learn a policy for a sequential decision making task.**

***Index Terms*—Curriculum Learning, Reinforcement Learning, Sim2Real**

I. Introduction

Sim2Real Transfer [1], [2] allows a model trained in a simulation to be deployed on a physical robot. However, it suffers from "Reality Gap" [1], where the simulation policy performs poorly on transfer to the real world. Continual learning on incrementally realistic simulations may help mitigate this problem [3]. One key limitation of Sim2Real approaches is the need for a simulation whose task Markov Decision Process (MDP) representation exactly matches the complex dynamics of the realistic scenario, which is not always feasible [4]. A common technique to achieve Sim2Real Transfer is to abstract the information from the simulator so that it is better suited for the realistic domain [5]. This still requires the system dynamics of the simulation and realistic domains to match. In this work, we aim to study cross-domain knowledge transfer, and how it can help aid learning in the realistic scenario.

Cross-domain knowledge transfer techniques aid learning in domains that do not share the MDP representations, but are semantically related [6], [7]. Most of these approaches require an explicit mapping between the two domains, usually provided by a human expert. In this work, we provide a framework to develop a curriculum for the complex realistic task in the high-fidelity (HF) domain by optimizing the task curriculum in a newly generated low-fidelity (LF) representation of the HF domain, and transfer the curriculum [8], [9]. This would generate curriculum for the HF, without having to interact in the HF domain. Through experiments in a robotic navigation and manipulation domain, we show that this requires fewer interactions than learning from scratch, and is applicable in settings where Sim2Real fails i.e. when the simulation model has different MDP representation than the physical model.

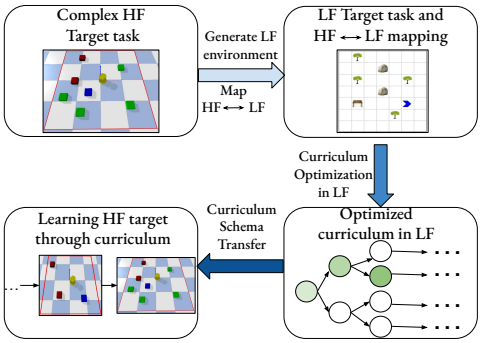


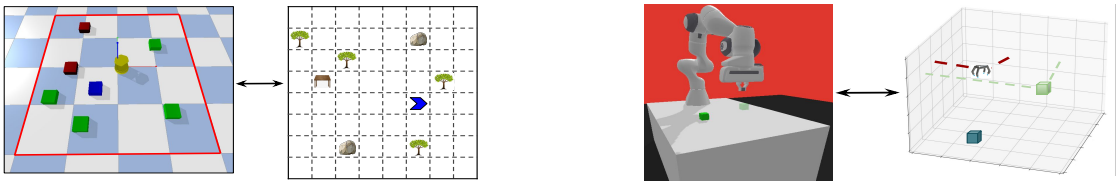
Fig. 1: Overview of the curriculum schema transfer procedure. This work focuses on generating the low-fidelity environment given a high-fidelity environment and attaining the mapping function

II. Curriculum Transfer Framework

The overall procedure of performing a Curriculum Schema Transfer involves four stages and is described in Fig 1: (1) Generating a low-fidelity (LF) version of the high-fidelity (HF) task along with the LF↔HF mapping; (2) LF curriculum optimization; (3) Transferring curriculum schema from LF to HF (4) Learning HF task trough the curriculum.

A. Generating a low-fidelity (LF) environment

A suitable characteristic for a LF environment is that the time-to-threshold for the LF task M_i^{LF} is lower than the time-to-threshold [10] for the equivalent HF task M_i^{HF}, i.e. $\Delta(M_i^{LF} - M_i^{HF}) > 0$. To generate a LF environment, we define the HF task using an Object-Oriented MDP (OOMDP) representation [11], [12] along with its existing MDP representation. OOMDP representation helps in abstracting the HF environment using a set of classes, and each class has parameters that take values within a range. At any given instance, an environment consists of a set of objects, where each object is an instance of a class and is defined by the values of the parameters for that class. The OOMDP state is given by the union of all object states, where each object state is the



(a) HF (left) and LF (right) for Crafter-TurtleBot (b) HF (left) and LF (right) in Panda-Pick-and-Place

Fig. 2: Target task illustration in the HF and LF environments for a Crafter-TurtleBot domain and Panda-Pick-and-Place domain. In Crafter-TurtleBot, agent's goal is to craft a stone-axe by collecting two trees and a rock. The goal in the manipulation domain is to place an object it at a target location.

value of the parameters of that object. The LF environment is generated by assuming an equivalency in the classes between the HF and LF environments $\mathcal{C}^{HF} \equiv \mathcal{C}^{LF}$, but differences in the range of parameter values for a class in LF environment. A human expert describes a discretization hyperparameter for a class parameter, that helps in reducing the range of the class parameter in the LF environment. Thus, by iterating over all parameters for all classes in the HF environment, the LF environment bounds the range of the parameter space, thereby decreasing the number of OOMDP states, while keeping the objective of the task unchanged. This accelerates the curriculum optimization procedure, and provides a convenient LF↔HF mapping for curriculum schema transfer.

B. Curriculum Optimization in low-fidelity (LF) environment

Once we have a suitable LF candidate, the next step is to optimize the curriculum in the LF environment. This is done using beam search algorithm, where a source task candidate is obtained by varying the OOMDP state parameters of the LF target task. This is carried out N times, where N is the length of the beam. Out of these N source tasks, the W source task that took the least amount of interactions to converge to an optimal policy are selected ($W<N$). Then, for each of these W source tasks, we again initialize N source tasks by varying LF OOMDP target task parameters. This process is continued until we reach the final target task. The idea behind this approach is to sequence source tasks in increasing order of difficulty, to reduce the overall interactions to reach convergence in target task. The converged policy of a source task is used as the initial policy of the next source task in curriculum.

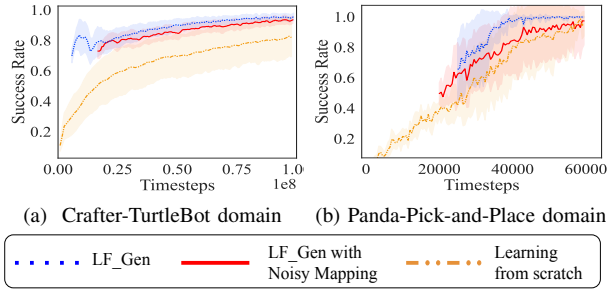
C. Low-fidelity (LF) to high-fidelity (HF) schema transfer

Once we have a curriculum in LF, the next step involves utilizing the LF↔HF mapping to transfer the schema of the optimzied curriculum from the LF to HF environment. Thus, now we have a curriculum in HF environment, without ever having to interact in the expensive HF environment.

D. Learning high-fidelity (HF) target task through curriculum

The curriculum schema transfer yields us a curriculum in HF. The agent starts with a random policy to learn the first source task in the curriculum. Once the agent learns the first source task, this policy is transferred to the next source task. Thus, the agent iteratively learns the tasks in the curriculum, culminating in the final target task[1].

[1]More details and code information in Appendix https://github.com/shukla-yash/LF_Gen/blob/master/TechnicalAppendix.pdf



(a) Crafter-TurtleBot domain (b) Panda-Pick-and-Place domain

Fig. 3: Learning Curves for Crafter-TurtleBot navigation domain (a) and Panda-Pick-and-Place manipulation domain (b).

III. Experiments and Results

We test our framework on two robotic domains. Fig 2a shows a continuous robotic navigation domain, in which a turtlebot needs to collect two pieces of trees, one piece of rock and approach the crafting table to craft a stone-axe. The LF domain involves an agent navigating in a 2D grid, where objects have discrete locations on the grid. Fig 2b is based on panda-gym [13] shows a continuous robotic manipulation domain, in which a robotic arm needs to pick an object and place it at target location. The LF domain involves an agent manipulating in a 3D grid, where objects have discrete locations on the grid. The discretization reduces OOMDP state space and helps in quicker optimization of curriculum. Fig 3 shows the performance of the curriculum schema transfer on the two domains as compared to learning from scratch. We also test our approach in scenarios where an exact LF↔HF mapping is not available, by introducing a multivariate gaussian noise over the parameters. The learning curves shows that it performs better than learning from scratch.

IV. Conclusion and Future Work

In this work, we proposed a framework that can optimize a curriculum for a complex robotic task in a simplified representation, and can transfer the schema of the curriculum. We showed that this method does not require an equivalent LF and HF MDP representation, and works even when Sim2Real fails. In future work, we would like to use multiresolution manifold representations to generate the LF environment, as it will help us to transfer policy and value functions across the LF and HF environments. Additionally, we would like to explore LF↔HF↔ real-world schema transfer.

REFERENCES

[1] Tobin J., Fong R., Ray A., Schneider J., Zaremba W., and Abbeel P. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In 2017 IEEE/RSJ Intl Conf. on Intelligent Robots and Systems (IROS). IEEE, 23–30.

[2] Höfer, S., Bekris, K., Handa, A., Gamboa, J.C., Golemo, F., Mozifian, M., Atkeson, C., Fox, D., Goldberg, K., Leonard, J. and Liu, C.K., 2020. Perspectives on sim2real transfer for robotics: A summary of the R: SS 2020 workshop. arXiv preprint arXiv:2012.03806.

[3] Josifovski J., Malmir M., Klarmann N., and Alois and Knoll. Continual Learning on Incremental Simulations for Real-World Robotic Manipulation Tasks. In 2nd Workshop on Closing the Reality Gap in Sim2Real Transfer for Robotics at Robotics: Science and Systems (R:SS) 2020.

[4] Paull, Liam, and Courchesne A. On assessing the value of simulation for robotics. In 2nd Workshop on Closing the Reality Gap in Sim2Real Transfer for Robotics at Robotics: Science and Systems (R:SS) RSS. 2020.

[5] Antonova R., Maydanskiy M., Kragic D., Devlin S., Hofmann K. Modular Latent Space Transfer with Analytic Manifold Learning In 2nd Workshop on Closing the Reality Gap in Sim2Real Transfer for Robotics at Robotics: Science and Systems (R:SS) RSS. 2020.

[6] Ammar, H. B., Eaton, E., Luna, J. M., & Ruvolo, P. (2015, June). Autonomous cross-domain knowledge transfer in lifelong policy gradient reinforcement learning. In Twenty-fourth international joint conference on artificial intelligence.

[7] Ammar, H.B., Eaton, E., Ruvolo, P. and Taylor, M.E., 2015, February. Unsupervised cross-domain transfer in policy gradient reinforcement learning via manifold alignment. In Twenty-Ninth AAAI Conference on Artificial Intelligence.

[8] Shukla, Y., Thierauf, C., Hosseini, R., Tatiya, G. and Sinapov, J., 2022, May. ACuTE: Automatic Curriculum Transfer from Simple to Complex Environments. In Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (pp. 1192-1200).

[9] Shukla, Y., Loar, K., Wright, R., Sinapov, J, 2022, June. An Object-Oriented Approach for Generating Low-Fidelity Environments for Curriculum Schema Transfer. In 1st Workshop on Scaling Robot Learning at the International Conference on Robotics and Automation, 2022.

[10] Narvekar, S., Peng, B., Leonetti, M., Sinapov, J., Taylor, M.E. and Stone, P., 2020. Curriculum Learning for Reinforcement Learning Domains: A Framework and Survey. Journal of Machine Learning Research, 21, pp.1-50.

[11] Carlos D., Cohen A., and Littman M. "An object-oriented representation for efficient reinforcement learning." Proceedings of the 25th international conference on Machine learning. 2008.

[12] Silva, F., and Costa A. "Object-oriented curriculum generation for reinforcement learning." Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. 2018.

[13] Gallouédec, Quentin and Cazin, Nicolas and Dellandréa, Emmanuel and Chen, Liming. "panda-gym: Open-Source Goal-Conditioned Environments for Robotic Learning." 4th Robot Learning Workshop: Self-Supervised and Lifelong Learning at NeurIPS.