**Experimental environments:**

For the purposes of this project, two experimental environments are used to compare the two selected machine learning approaches: Q-Learning and Learning from observations. The two environments’ goal is to test the performance of the ML algorithms in the same playing area and in a completely new environment using the knowledge acquired from training.

The first environment is a controlled playing area in Minecraft comprising of small, enclosed platforms. The platforms are surrounded by blocks of lava, limiting the state space by not allowing the agent to venture away from the goal. The areas are specialized to train the agent into two key elements of playing Minecraft – reaching objective points (such as collecting diamonds) and placing blocks to help the player get to unreachable locations (for example crossing a gap or building upwards towards a high point). The three platforms used in this environment are the following:

• horizontal (2D) walking – 3x12 blocks arena with the starting and destination blocks at the ends of the platform. Blocks in the floor can be randomly swapped for lava blocks to bring some variety to the testing platform.



• climbing (3D) – 4x4 platform with half the platform being elevated 2 blocks high, requiring the agent to place at least one block near the wall to be able to climb. Starting and destination blocks are at opposite corners of the platform.



• crossing a gap – 6x6 platform with two adjacent rows of blocks replaced with lava. To reach the destination, the player needs to place two consecutive blocks at “0” elevation



Both unsupervised and supervised learning can be used in the first environment for testing the system and benchmarking the performances. After a set amount of training, the agent can be moved to the second experimental environment.

The second environment consists of a procedurally generated Minecraft world created by the game itself. This world is an accurate representation of a real scenario when playing Minecraft. Across the world, goal markers are placed to test the overall performance of the agent when placed in an uncontrolled environment.



The possible agents to test in this environment are: Reinforcement Learning agent with no prior knowledge, Reinforcement Learning agent with prior knowledge, a Random Walking agent and a Learning from demonstrations agent.

Bibliography

[1] Alger, M. (2016) Deep Inverse Reinforcement Learning, The Australian National University.

[2] Amato, C., Shani, G. (2010) High-level Reinforcement Learning in Strategy Games, Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems(AAMAS 2010).

[3] Grosz, Barbara (1996) Collaborative Systems, AAAI-94 Presidential Address, AI Magazine Volume 17 Number 2 (1996)

[4] Watkins, C. (1989) Learning from Delayed Rewards, PhD Thesis, University of Cambridge, England.

[5] Wulfmeier, M., Ondruska, P., Posner, I. (2015) Maximum Entropy Deep Inverse Reinforcement Learning, Mobile Robotics Group, Department of Engineering, Science, University of Oxford

[6] Yannakakis, G., Togelius, J. (2014) A Panorama of Artificial and Computational Intelligence in Games

[7] Hester, T. et al. (2017) Deep Q-learning from Demonstrations, Association for the Advancement of Artificial Intelligence

[8] Krishnan, S., Garg, A., et al. (2016) HIRL: Hierarchical Inverse Reinforcement Learning for Long-Horizon Tasks with Delayed Rewards, EECS IEOR, University of California, Berkeley CA USA28

[9] Mnih, V. et al. (2013) Playing Atari with Deep Reinforcement Learning

[10] Ziebart, B., Maas, A., Bagnel l, J., Dey, A. (2008) Maximum entropy inverse Reinforcement Learning In AAAI, pages 1433–1438, 2008.

[11] Vinyals, O. et al (2017) StarCraft II: A New Challenge for Reinforcement Learning

[12] Silver, D. et al (2017) Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm