

StarCraft II: A New Challenge for Reinforcement Learning

Paper Review

By Daniel Bourke

Original paper can be viewed here: <https://deepmind.com/documents/110/sc2le.pdf/>

Introduction and Goals

A new grand challenge for reinforcement learning (RL) has appeared. DeepMind and Blizzard have partnered together to create **SC2LE** (StarCraft II Learning Environment).

Reinforcement learning coupled with deep learning has proven successful in the past with games such as Go (**AlphaGo**) and Atari (**DeepMind Atari Agent**). The goal of the SC2LE is to take the knowledge gained from previous successful implementations of RL and furthering it in a domain more advanced than any previous environment.

StarCraft II can be considered a more challenging environment than others used in the past because of a number of variables an agent has to consider. StarCraft II involves multiple agents (several different kinds of characters), delayed credit assignment (some actions aren't rewarded until later in the game), a lack of ability to see the entire game space (fog of war covering unpopulated sections of the map) and multiple other players (games range from 1v1 to 3v3). This complexity makes defeating top human players a meaningful and measurable long-term goal.

SC2LE includes an open source Python-based interface for communicating directly with the game engine, a suite of mini-games focusing on the different elements of StarCraft II gameplay and a dataset of game replay data from human expert players.

With the introduction of SC2LE, DeepMind and Blizzard hope to make research towards the goal of creating agents able to take defeat professional human players more accessible for all.

Outcomes and Results

Established RL algorithms and two human players (a novice DeepMind game tester and a StarCraft GrandMaster) were used to determine baseline performance levels within SC2LE. Baseline performance levels are critical for further comparative research.

Established RL algorithms included *Atari-net Agent*, *FullyConv Agent*, *FullyConv LSTM Agent* and two random agents. The Atari-net Agent was the deep reinforcement learning architecture used by DeepMind in previous research. The *FullyConv Agent* and *FullyConv LSTM Agent* both involved *fully convolution* networks except the *FullyConv LSTM Agent* included a convolutional LSTM module. The two random agents were *random policy*, an agent which picked uniformly at random among all valid available actions and *random search*, an agent which evaluated randomly initialised policy networks and keeping the one with the highest score.

When tested on the full game, the agents used did not learn to win a single game. Agents trained with a ternary reward strategy (win = 1, draw = 0, loss = -1) were unable to develop a strategy which lasted for the entire game and achieved very poor results. Agents trained with the *Blizzard score* (a running score throughout the game indicating progress, only available to human players at the end of the game) ended up choosing basic strategies such as keeping worker units mining minerals to ensure a steady improvement in the score. These results show how challenging StarCraft II is as an RL domain.

The RL algorithms performed much better on mini-games than the full game. *Fully convolutional* agents achieved the best results. When compared to the GrandMaster player, all agents performed sub-optimally except for in the simplest mini-game, MoveToBeacon. In mini-games such as DefeatRoaches and FindAndDefeatZerglings, agents were competitive with the novice DeepMind player. This demonstrates how even simple mini-games can present difficult challenges. Future RL agents should be able to achieve human-level performance on these games with ease if it is to have a chance at conquering the full game.

Finding the baseline performance levels of RL agents in the SC2LE included several simplifications of the game as it is played by humans, such as constant access to the Blizzard score and unlimited computing time between each step rather than being in real-time.

Future Work

Future updates to the environment will introduce access to human replays for RL agents and removal of the simplifications to move closer towards the goal of training agents humans consider to be worthwhile opponents.

More details on the SC2LE, architectures and simplifications for the RL agents, results of the paper and future work can be viewed in the [full text](#).

Authors Notes

This kind of research excites me. As a lifelong gamer, competing against AI agents is something I'm very familiar with. The idea of creating an agent able to learn on its own is mind-blowing to me. If the goal of creating a worthy opponent for professional players is achieved in a game as complex as StarCraft II, imagine where else those techniques could be used. If the Atari-agent can master close to 50 Atari games with a single algorithm, could the StarCraft II agent do the same with other complex games?

Reinforcement learning combined with other techniques such as deep learning enable us to gain insight into how we learn as humans. I'm not sure these techniques will be the final frontier on the journey to solving intelligence but for now, they are state of the art. After writing this brief summary, I'm left wondering how I would go about creating an RL agent for games I actively play.