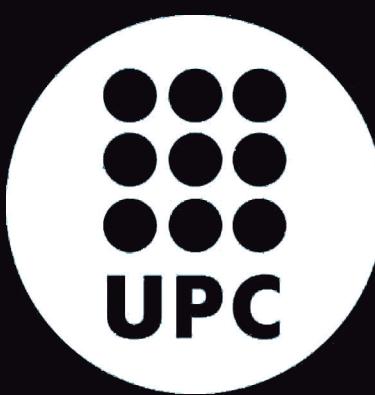
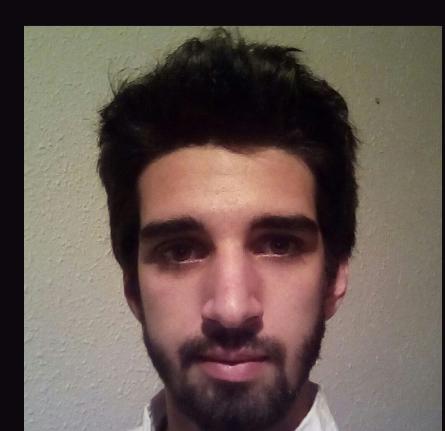


Playing StarCraft II with Reinforcement Learning



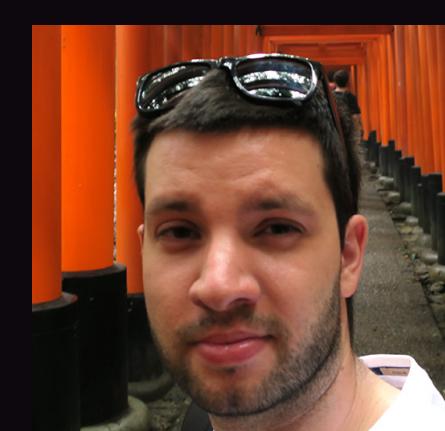
UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



Godefroy
Goffe



Luis
Esteve



Carlos
Roig



Alejandro
Suárez



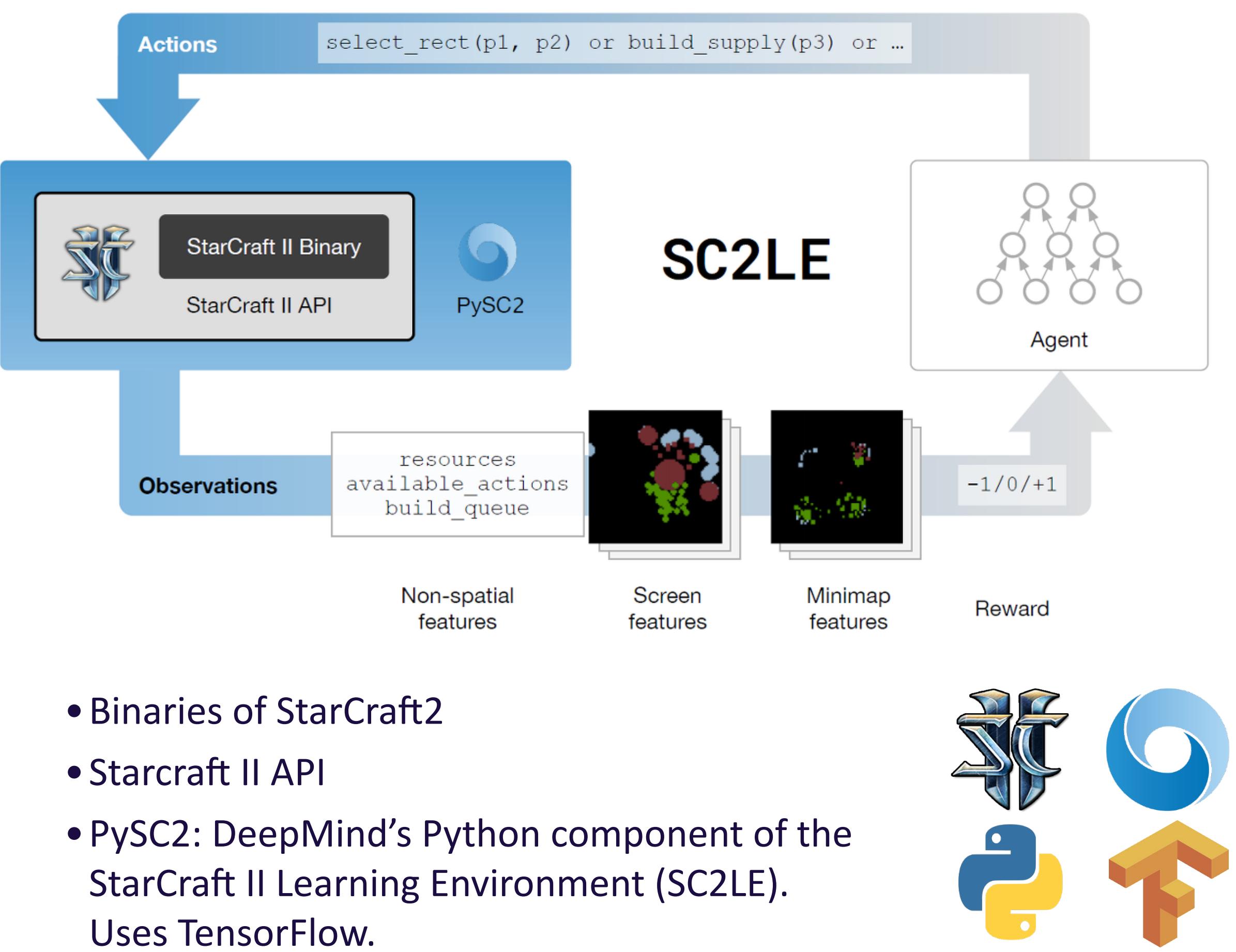
StarCraft II is real time strategy (RTS) game.
Defeating top human players therefore becomes a meaningful
and measurable long-term objective.



Difficulties:

- Multi-agent problem
- Imperfect information game
- Wide range of actions
- Long term strategies

Environment: StarCraft 2 Learning Environment

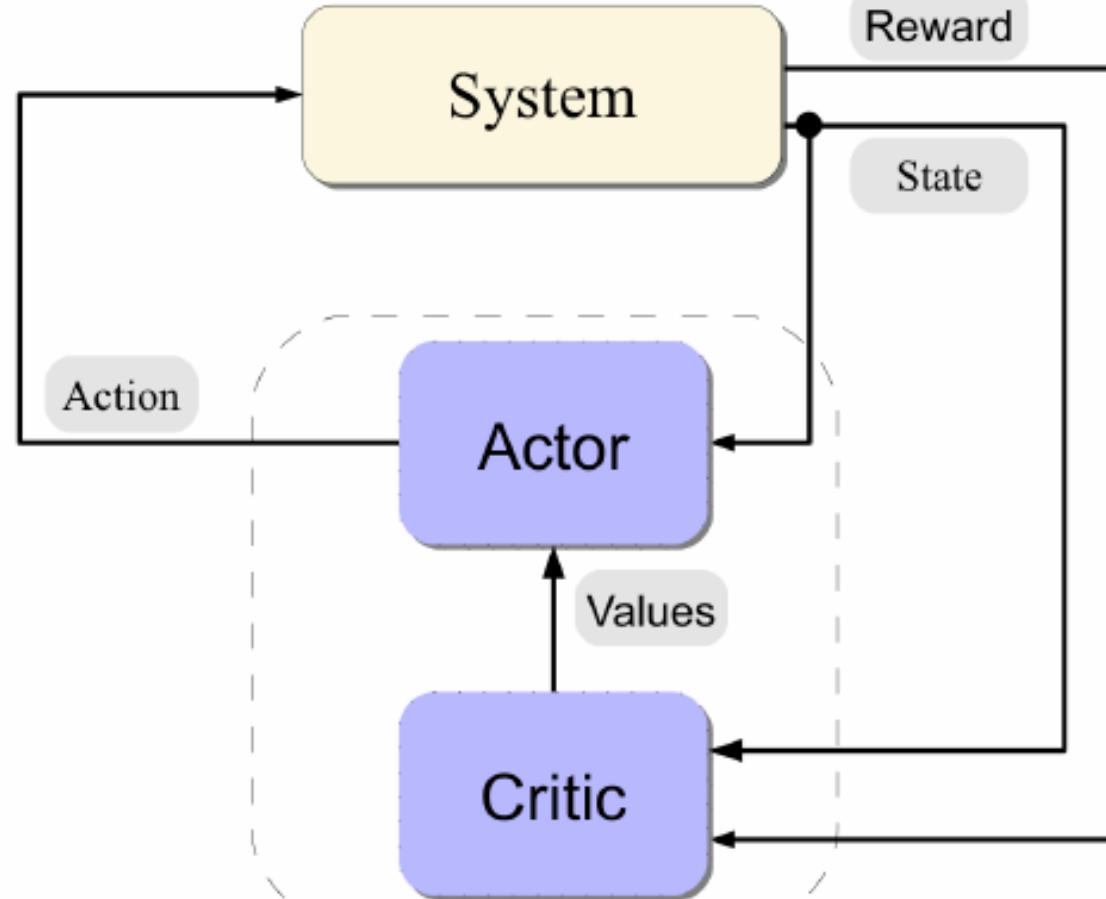


Objectives:

- Training and evaluating a RL agent focused on small mini-games.
- Mini-game: Find and defeat Zerlings. This minigame consist in manage 3 marines in a map. They have to explore the map and find and destroy enemies (Zerlings)
- Experimentation with reward system
- Experimentation with hyperparameters

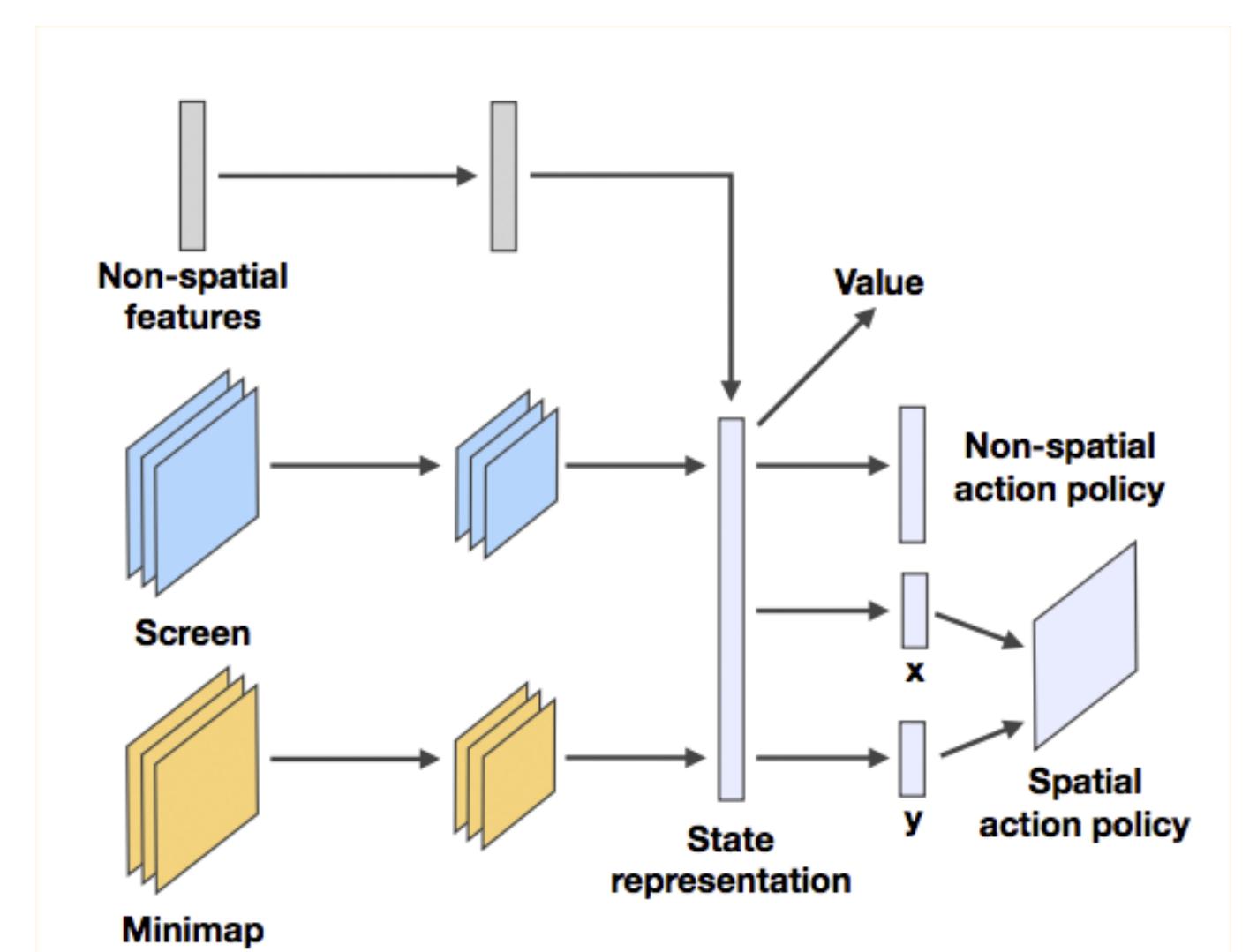
Implementation:

A3C algorithm



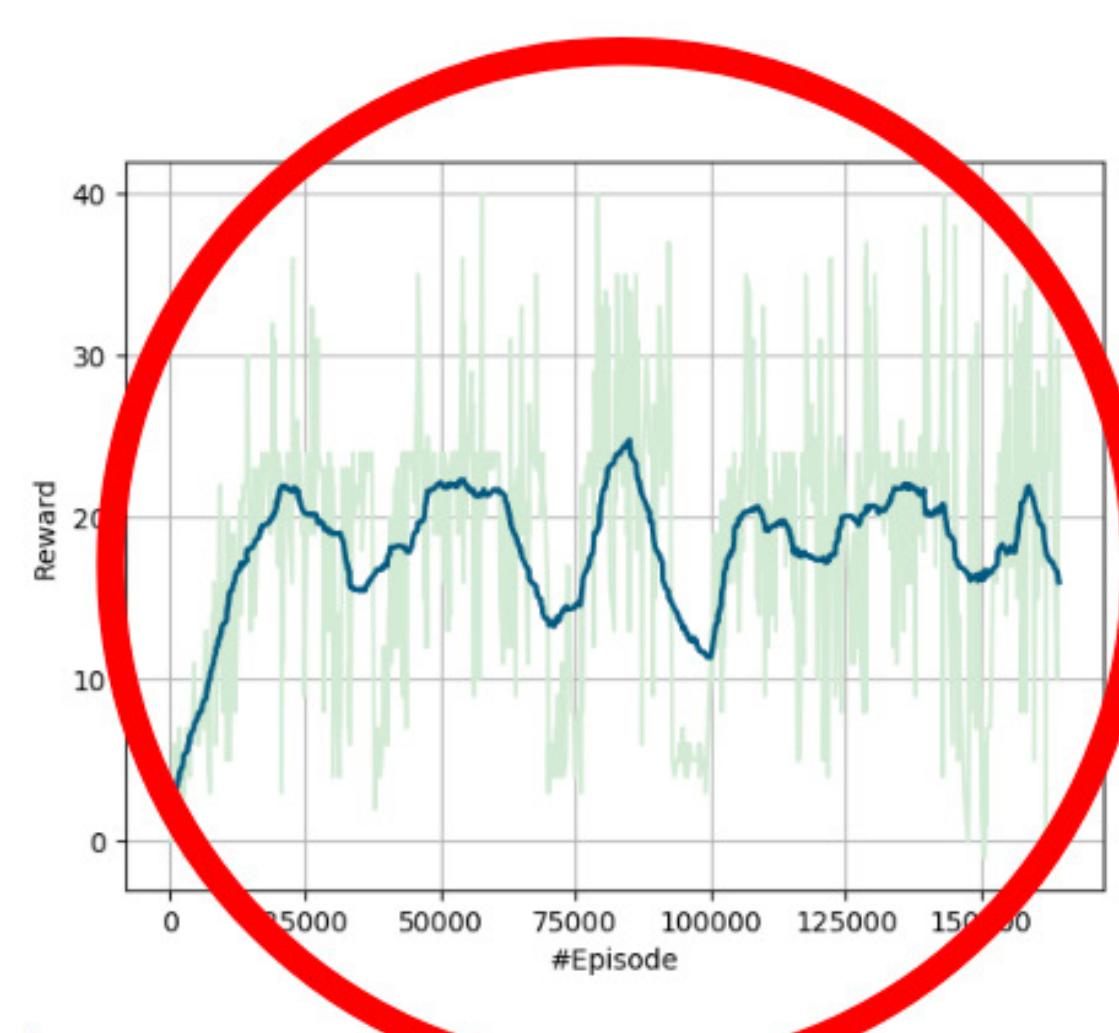
A3C is a policy based RL algorithm, in which the policy is encoded by a neural network that is shared and updated asynchronously by several parallel workers.

AtariNet Neural Network



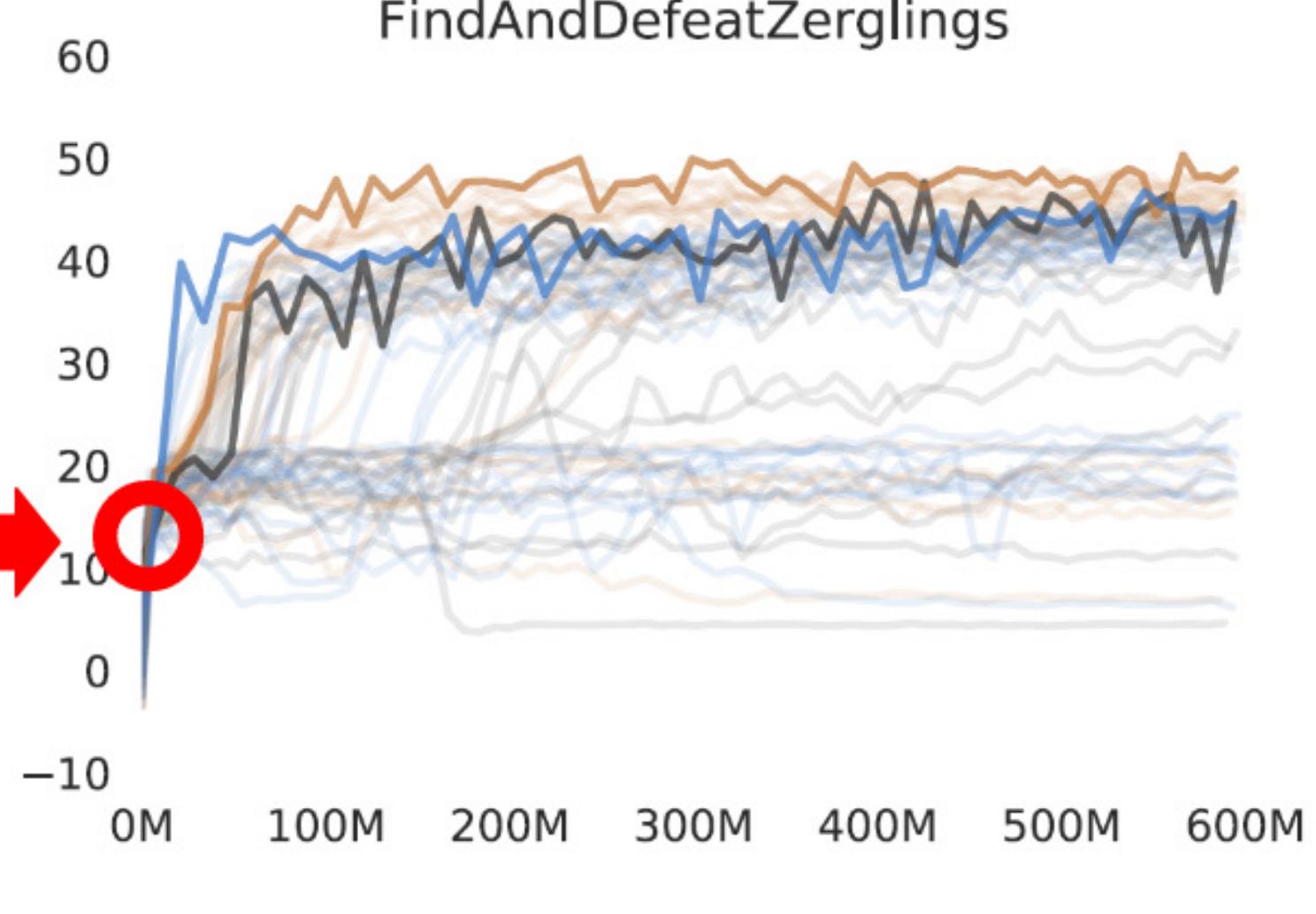
First experiment: comparison with Google Deepmind results

Our results



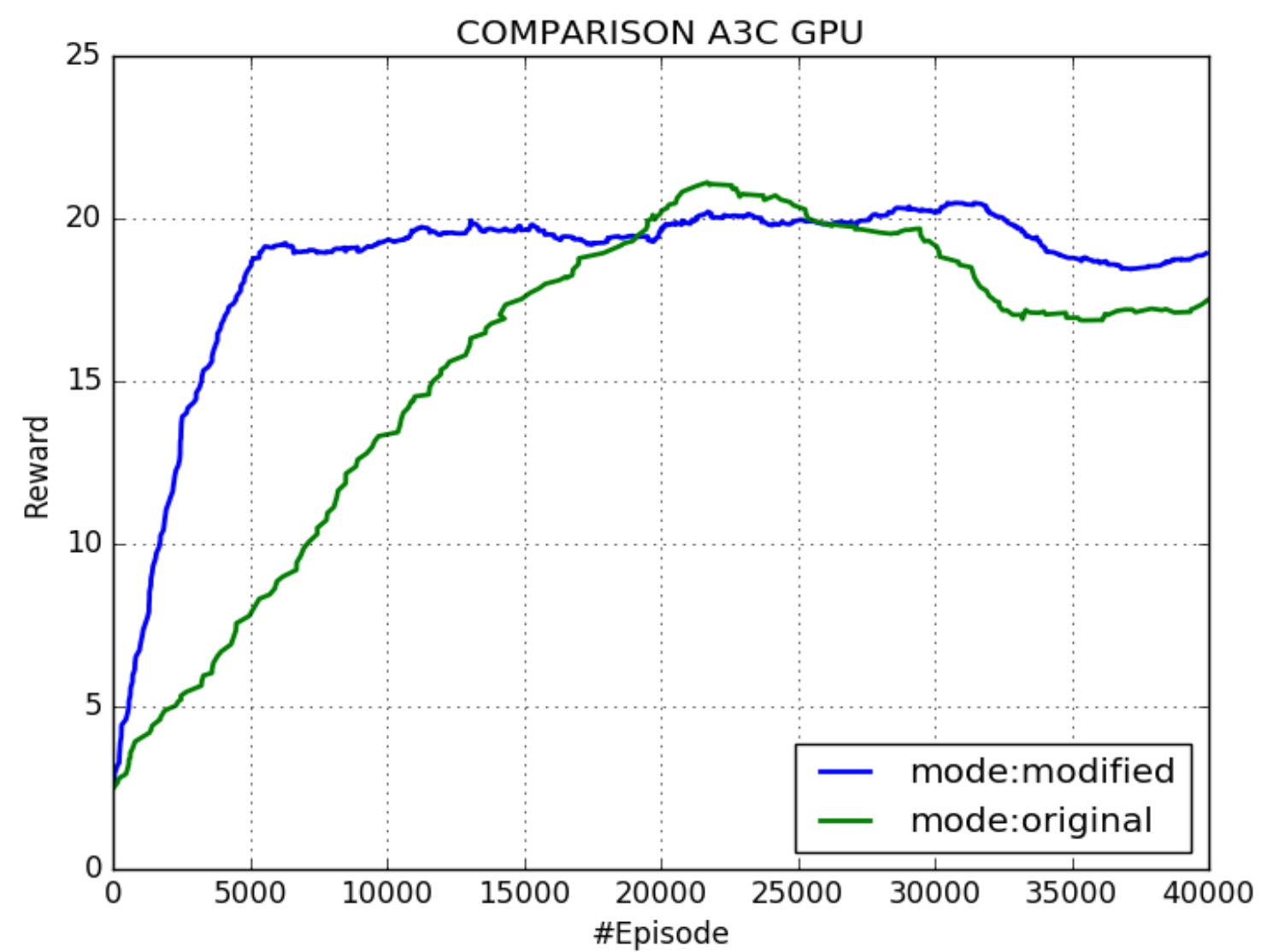
Deepmind results

FindAndDefeatZerglings



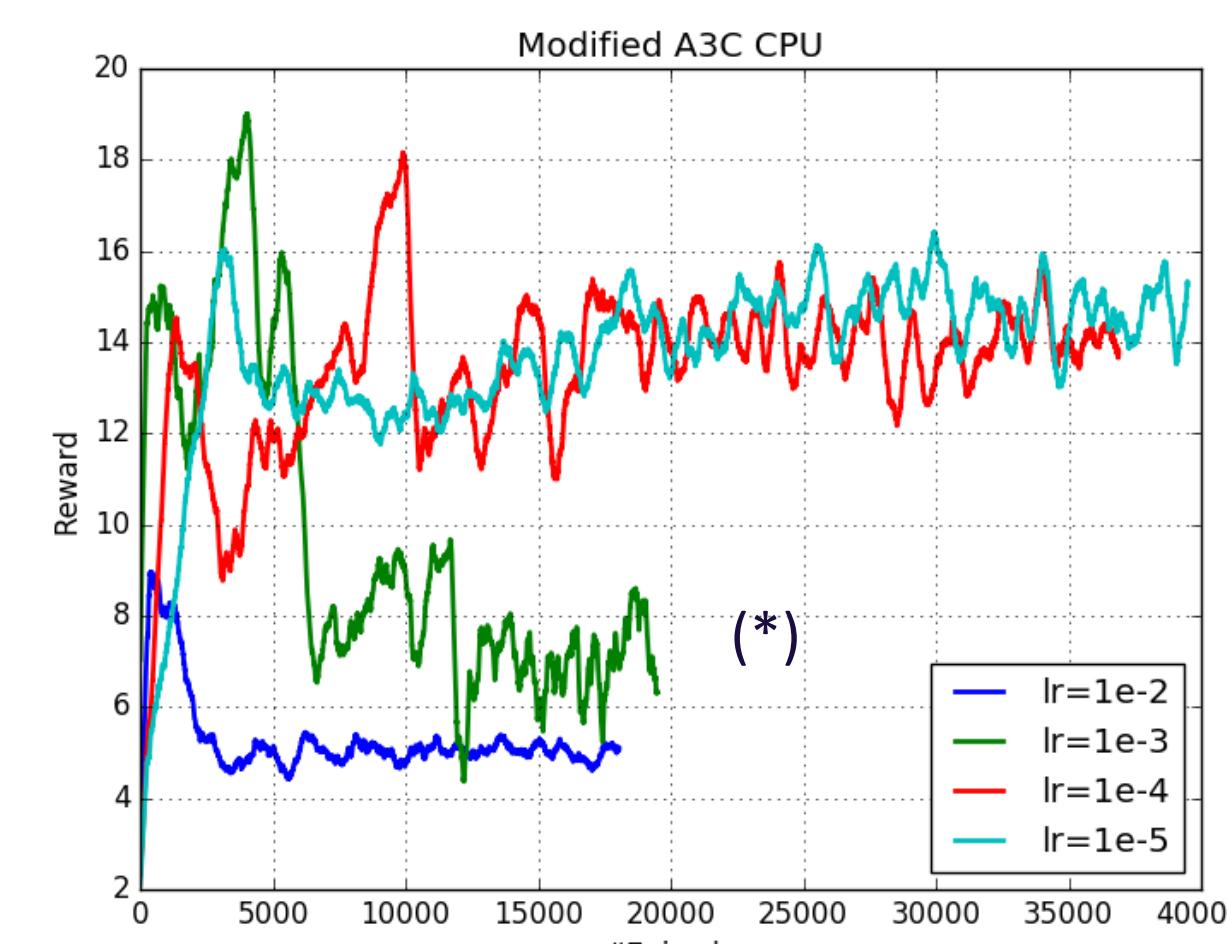
In order to achieve Deepmind results more simulations are needed. But is so expensive: in time (more than 63 years of simulations with our setup) and in cash (640k \$ in Google fees)

Second experiment: reward hacking



The agent with the modified reward learns faster than the original one. It needs fewer iterations to achieve similar rewards

Third experiment: learning rates



Optimal Learning Rate:
1e-4 / 1e-5

Bigger LR make the algorithm not converge

Sponsored by:

