EMERGENT COMPLEXITY VIA MULTI-AGENT COMPETITION

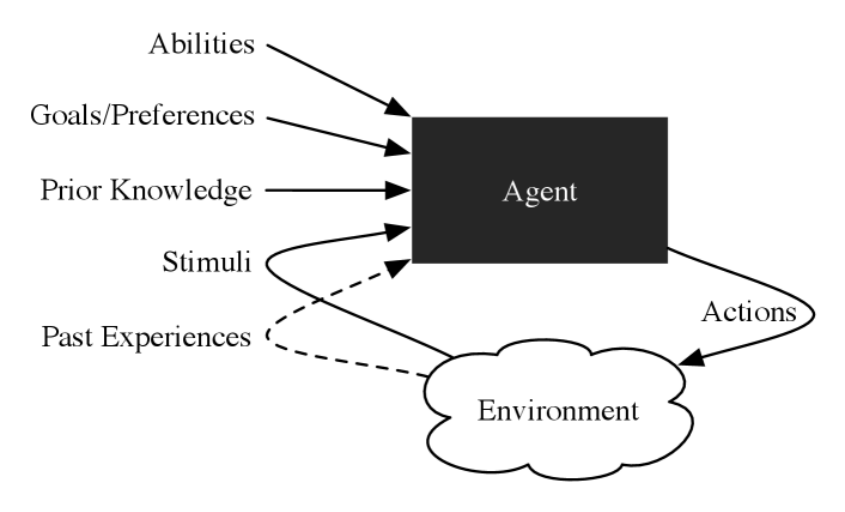
Trapit Bansal, UMass Amherst, Jakub Pachocki, Szymon Sidor, Ilya Sutskever, Igor Mordatch from OpenAI

Paper Presentation 2024-03-29

Dinara Zhussupova

HSE University

Agent. Environment



Reinforcement learning algorithms can train agents that solve problems in complex, interesting environments.

Complex environment

The complexity of the trained agent is closely related to the complexity of the environment.

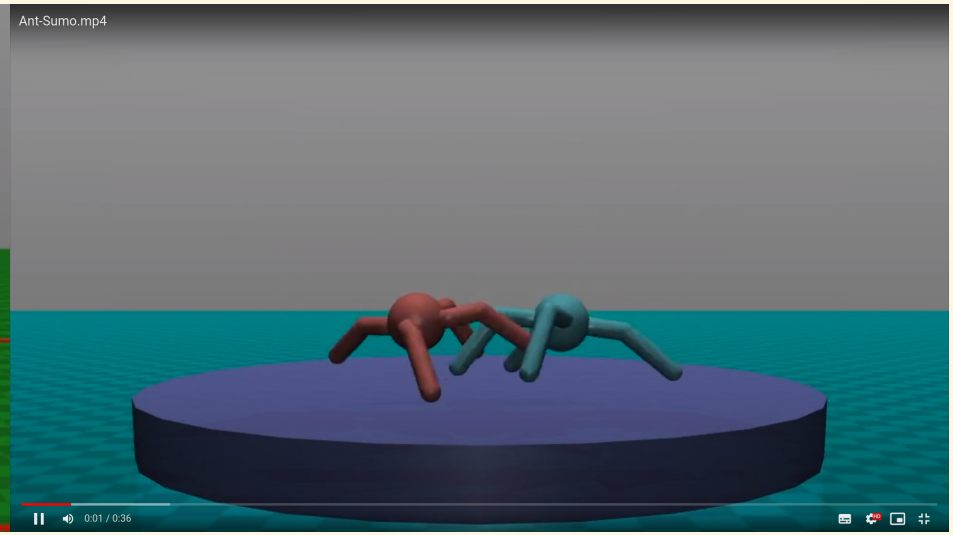
Self play

A competitive multi-agent environment trained with self-play can produce behaviors that are far more complex than the environment itself.

Multi-agent environments[1]

This paper introduces several competitive multi-agent environments where agents compete in a 3D world with simulated physics. The trained agents learn a wide variety of complex and interesting skills, even though the environment themselves are relatively simple. Authors consider two three-dimensional agent bodies: ant and humanoid. The ant is a quadrupedal body with 12 DoF and 8 actuated joints. Humanoid has 23 DoF and 17 actuated joints.

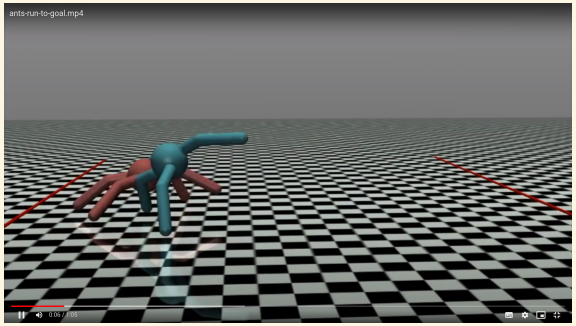
Kick and defend task for humanoids

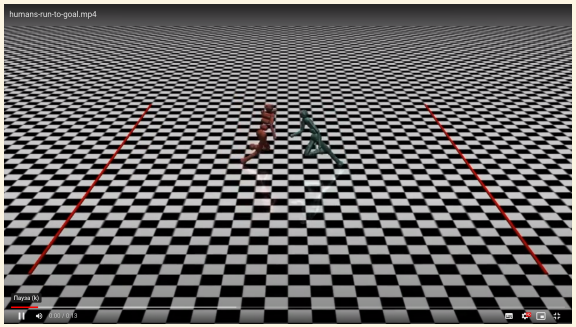


Sumo task for ants

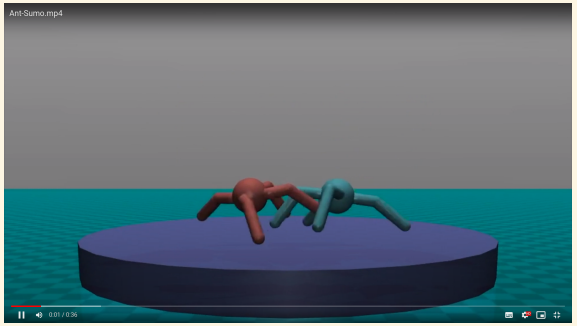
Four tasks

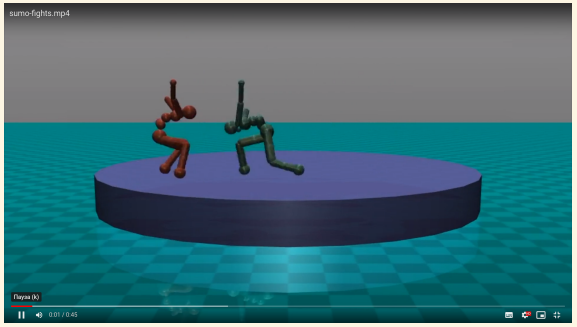
Run to Goal



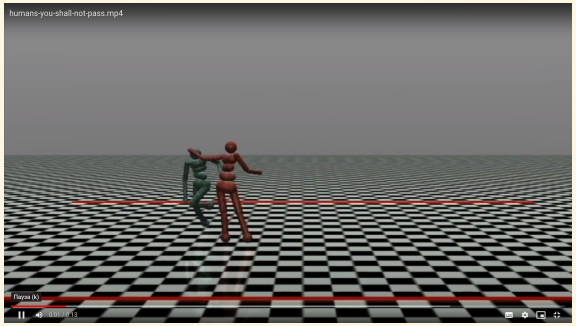


Sumo

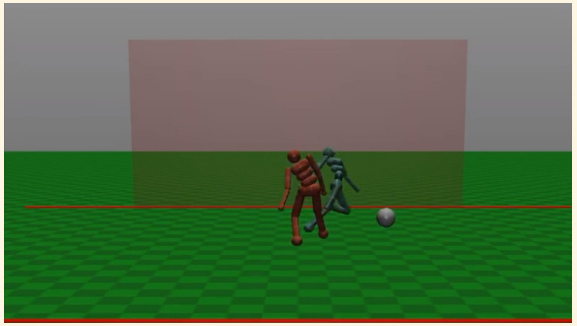




You Shall Not Pass



Kick and Defend



Run to Goal

- The agents start by facing each other in a 3D world and they each have goals on the opposite side of the word.
- The agent that reaches its goal first wins.
- Reaching the goal before the opponent gives a reward of +1000 to the agent and -1000 to the opponent.
- If no agent reaches its goal then they both get -1000.

You Shall Not Pass

This is the same world as the previous task, but one agent (the blocker) now has the objective of blocking the other agent from reaching it's goal while not falling down.

- If the blocker is successful in preventing the opponent from reaching the goal and is standing at the end of episode then it gets +1000 reward.
- If it is not standing then it gets 0 reward, and the opponent gets -1000 reward.
- If the opponent is successful in reaching it's goal then it gets +1000 reward and the blocker gets -1000 reward.

Sumo

The agents compete on a round arena and the goal of each agent is to either knock the other agent to the ground or to push them out of the ring.

- The winner gets +1000 and the other agent gets -1000.
- If there is a draw then both agents get -1000.

Kick and Defend

This a standard penalty shootout.

- One agent has to kick a ball through the goal, which has a fixed width of 6 units, while the other agent defends.
- Successful kick or defend gives the agent +1000 reward and the opponent -1000 reward.
- The defender cannot go beyond the goal-keeping area which is a distance 3 units from the goal, doing so terminates the game with a penalty of -1000 for the defender.
- We give two additional rewards for defender: if defender is successful and it made contact with the ball then it gets additional +500 reward, and if the defender is successful and still standing at the end of the game then it gets another additional reward of +500.

Authors found the latter two rewards to yield more realistic looking defending behaviors.

TRAINING COMPETITIVE AGENTS

Algorithm

Proximal Policy Optimization (PPO) John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. https://arxiv.org/abs/1707.06347

Distributed approach

Researchers adopt a decentralized training approach and use a distributed implementation of PPO for very large scale multi-agent training. This allows to use really large batch-sizes during training ameliorating the variance problem to some extent while also aiding in exploration. Our distributed PPO implementation is similar to the implementation of Nicolas Heess, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, Ali Eslami, Martin Riedmiller, et al. Emergence of locomotion behaviours in rich environments. https://arxiv.org/abs/1707.02286

Reward

Instead of estimating a truncated generalized advantage estimate (GAE) from a small number of steps per rollout, authors estimate GAE from the full rollouts. This is important as the competition reward is a sparse reward given at the termination of the episode.

EXPLORATION CURRICULUM

Challenges: earn basic motor skills vs competitive multi agent training

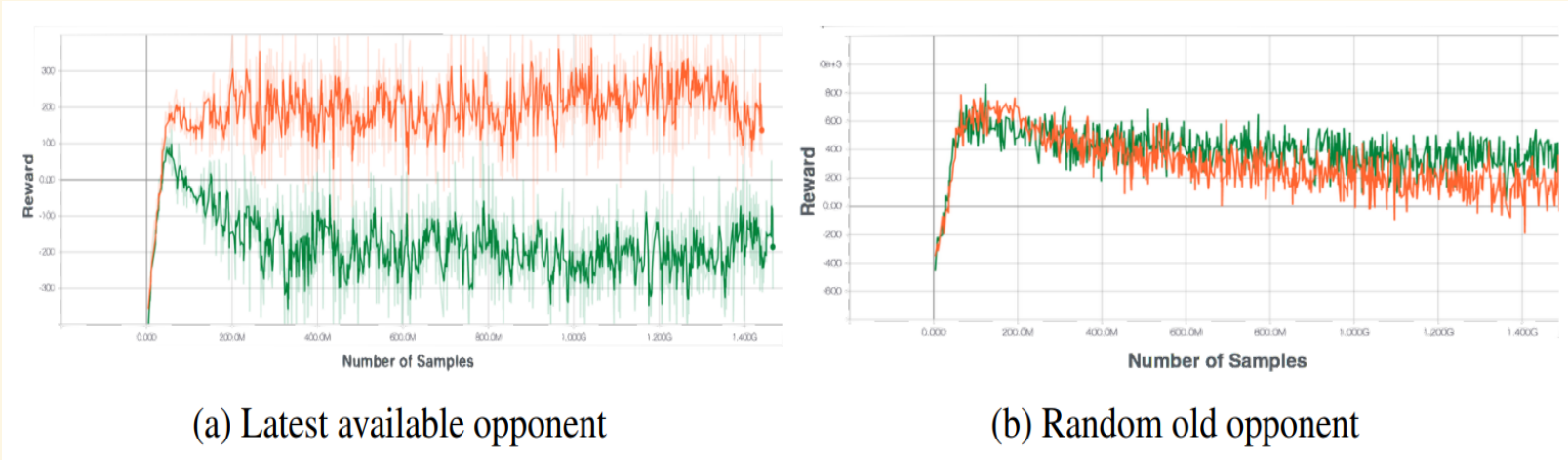
- use a dense reward at every step in the beginning phase of the training to allow agents to learn basic motor skills, like walking forward or being able to stand, which would increase the probability of random actions from the agent yielding a positive reward
- the exploration reward is gradually annealed to zero, in favor of the competition reward, to allow the agents to train for the majority of the training using the sparse competition reward

$$r_t = \alpha_t s_t + (1 - \alpha_t) I[t == T] R$$

OPPONENT SAMPLING

The skill of opponents encountered during training could have significant impact on the learning of the agents.

Training agents against the most recent opponent leads to imbalance in training where one agent becomes more skilled than the other agent early in training and the other agent is unable to recover.



(a) Latest available opponent

(b) Random old opponent

First figure - naive approach, and second - opponent sampling

EXPERIMENTAL DETAILS

Policies and Value Functions

Authors compare both MLP and LSTM for the policies and the value functions. MLP had 2 hidden layers with 128 units each. For LSTM networks, the input was first projected to a 128 dimensional embedding using a fully connected layer with ReLU activation which is then fed into a single-layer LSTM with 128 hidden state dimension and the output is projected to the action dimension using another fully connected layer.

There were used MLP policy and value functions for the `run-to-goal` and `you-shall-not-pass` environments, and

LSTM policy and value function for `sumo` and `kick and-defend`.

Observations

- Ant body: all the joint angles of the agent, its velocity of all its joints, the contact forces acting on the body and the relative position and all the joint angles for the opponent
- Humanoid body: in addition to the above we also give the centre-of mass based inertia tensor, velocity vector and the actuator forces for the body
- Sumo environment: the torso's orientation vector as the input, the radial distance from the edge of the ring of all the agents and the time remaining in the game
- Kick-and-defend: the relative position of the ball from the agent, the relative distance of the ball from goal and the relative position of the ball from the two goal posts

Algorithm Parameters

- Adam with learning rate 0.001
- clipping parameter in PPO = 0.2, discounting factor = 0.995 and generalized advantage estimate parameter = 0.95
- each iteration consists of 409600 samples from the parallel rollouts and perform multiple epochs of PPO training in mini-batches consisting of 5120 samples

Main results

- LEARNED BEHAVIORS
- EFFECT OF EXPLORATION CURRICULUM
- EFFECT OF OPPONENT SAMPLING
- LEARNING ROBUST POLICIES
  - RANDOMIZATION IN WORLD
  - COMPETING AGAINST ENSEMBLE OF POLICIES