Binbin Xiong (bxiong@stanford.edu) & Minfa Wang (minfa@stanford.edu)

Project Proposal: Pommerman - FFA

Overview

In this project we propose to participate in the Pommerman competition, which is a play on Bomberman. Our goal is to develop a single agent under FFA mode (Free For All, where four agents enter and one leaves and the board is fully observable) that can beat all other players. More detailed game description could be found here.

Input & Output

Our task is to design a model which reads a state (as a dictionary of observation), outputs an action.

The **state** of the game for each agent is represented as a dictionary of following fields:

- **Board**: 121 Ints. The flattened board.
- **Position**: 2 Ints, each in [0, 10]. The agent's (x, y) position in the grid.
- **Ammo**: 1 Int. The agent's current ammo.
- **Blast Strength**: 1 Int. The agent's current blast strength.
- **Can Kick**: 1 Int, 0 or 1. Whether the agent can kick or not.
- **Teammate**: For FFA, this will be -1.
- **Enemies**: 3 Ints, each in [-1, 3]. Which agents are this agent's enemies.
- **Bombs**: List of Ints. The bombs in the agent's purview, specified by (X int, Y int, BlastStrength int).

For each play, the agents will be put to the corner of a randomly generated board, the game ends when maximum number of time steps are met or only one player leaves.

In each time step, an agent can choose from one of six **actions**:

$$Actions(state) = \{ Stop, Up, Left, Down, Right, Bomb \}$$

The input game board is a randomly drawn symmetric 11x11 grid, and four agents will be placed in each corner. Board contains wood walls and rigid walls. The agents will have an accessible path to each other. Rigid walls are indestructible and impassable.

Wooden walls can be destroyed by bombs. After they are destroyed, they become either a passage or a power-up.

Evaluation Metrics, Baseline & Oracle

The project has both extrinsic metric and intrinsic metric. The **extrinsic** metric is to compete with other trained agents, and maximize the game score over N games. For each game, the final score of an agent is $+1$ if it survives until the end, otherwise -1. We also need to define **intrinsic** metric for fast optimizations. The intrinsic metric may evolve as the project goes along. For now, we choose **random walking agent** as our enemy to compete against to. Each timestamp, each agent will be given an reward = { +1 if it's the only survivor and the game is end, 0 if it's still alive, -x: if it's dead }. We normalize the rewards to make it a zero-sum game.

Baseline: simple_agent provided by the starter code. It is the official baseline model for the competition. The baseline agent's performance measured by intrinsic metric is -0.067.（It is negative because sometimes it kills itself quickly.)
Oracle: An agent that can beat all others. The best in NIPS 2018 competition. Its value on extrinsic metric is 1.0.

Challenges

- **Multi players**: In the game, our agent needs to compete with 3 other close-to-adversarial agents.
- **Simultaneous game**: The agents are taking actions simultaneously. It adds more uncertainties to the game: collisons of players, and prediction of other player's next bomb location, etc.
- **Large search space:** For each state, there would be 1296 ($=6^4$) actions spawning out.
- **Complex Rules:** There are so many different situations of the intermediate states that makes it challenging to create a successful evaluation function.

Research/Modeling Ideas

- N-Person Minimax and Alpha-Beta Pruning: extend minimax to multi-agents scenario, and use alpha-beta pruning for efficient searching.
- Depth-limited search: reduce search space with evaluation function.
- Proximal Policy Optimization Algorithms (PPO): Efficient policy gradient updates.
- Opponent Modeling in Deep Reinforcement Learning: Learn strategy patterns of opponents through encoding observations of opponents into a deep Q-Network (DQN).