A Self-Play Policy Optimization Approach to Battling Pokémon

Dan Huang
San Francisco, CA, United States
hello@yuzeh.com

Scott Lee
Irvine, CA, United States
randomperson2727@gmail.com

***Abstract*—Pokémon is a popular role-playing video game franchise with a long-lived competitive scene that has evolved throughout the last two decades. The game exhibits several properties that come together to present a worthy challenge for AI agents to tackle. In this work, we present a low-cost self-play based reinforcement learning approach to the competitive battling aspect of the game. The proposed agent was tested and trained in a variety of environments designed to simulate possible use cases of such an AI. Experiments demonstrate that the agent is capable of performing on par with the state of the art in search-based Pokémon AI, as well as being competitive with human players on a popular matchmaking ladder. Furthermore, we investigate the transferability of trained skill–whether an agent trained in one environment performs well in a different environment.**

I. Introduction

In game-playing AI, there are several known challenges that that make both design and implementation of agents difficult. For example, some techniques, like minimax search, often require a means of simulating the impact of a decision on a game state. The simulation itself can be computationally expensive, and implementation is a non-trivial task, particularly for complex, partially observable, and stochastic games. On

II. Background

Pokémon is a popular video game franchise featuring a combat system around which a competitive scene has developed. Battles are turn-based adversarial games in which players construct teams of six Pokémon each and aim to defeat their opponent's Pokémon before they themselves are defeated. Over the course of seven major iterations, the game's ruleset and competitive metagame have changed drastically, with the game mechanics changing after each iteration. As an AI benchmark, modern Pokémon has several properties that make it challenging [1]. Game states in Pokémon are high-dimensional and the majority of its features are both categorical and partially observable. Player decisions are also processed atomically, and individual moves tend to have large impacts on game state. As a result, a game state can change drastically and unpredictably between steps. The game also features teambuilding on a scale that is several orders of magnitude more complex than many MOBAs, like Dota.

Several members of the competitive Pokémon community have developed agents to tackle this problem. The methodologies comprising the current state of the art consist primarily of search-based and heuristic-based systems, with limited forays into machine learning [2] [3] [4]. We believe the RL approach