# CSE573 - Paper Report: AlphaGo Zero

Joseph David

May 13 2022

## 1 Background

Much success towards artificial intelligence has involved supervised learning, where a model is trained on reliable data. However, it is sometimes difficult to find such training data or it may not exist at all. In these cases it is beneficial to use reinforcement learning methods where the model simply acts and learns from experience.

One of the main issues with reinforcement learning is in areas where the search space is particularly large. In order to be successful, a system must have an advanced lookahead function. Such is the issue with the game of Go, which artificial intelligence did not successfully tackle until AlphaGo in 2015. The game of Go involves two players taking turns playing their respective pieces (black and white) on the board grid. At each turn, a player may place a piece on any open tile, so with a 19x19 grid it is infeasible to check all possible sequences of mvoes. AlphaGo used two neural networks: a policy network that outputs move probabilities and a value network that evaluates states.

## 2 Research Problem

In this paper, the authors build a system to play Go that relies solely on reinforcement learning. Thus, it is trained only by playing Go, starting with random moves, and is not supervised by humans.

## 3 Main Contributions

AlphaGo Zero uses a single neural network $f_\theta$ that takes as input the current position (of white and black pieces on the board) as well as the past positions. The network then outputs move probabilities given the current state $p(a|s)$ and as well as an evaluation $\nu$ of the likelihood that the current player will win given the current state.

At each state in the game, AlphaGo Zero executes a Monte Carlo tree search to only explore possible future states that have high probability, according to the probabilities given by $f_\theta$. Then the tree search outputs probabilities $\pi$ of playing each move as well as a variable $z$ that denotes whether the current player will win. These probabilities usually correspond to better moves and are therefore used to update and improve the current policy. So the data $(\pi, z)$ is passed into $f_\theta$ in order to align it more closely with these better mvoes and to minimize the error between the game winner $z$ and the expected winner $\nu$.

The network was trained on 4.9 million games. After 36 hours of training, AlphaGo Zero outperformed AlphaGo Lee and eventually beat Lee a hundred out of a hundred times. Furthermore, AlphaGo Zero trained only for two days as opposed to AlphaGo Lee's four months, and required much less computing power.

These results show that reinforcement learning may lead to much better asymptotic performance and is similar to supervised learning in the amount of training required.