AlphaZero-Othello

(a truly unoriginal name)

Jonathan Hayase

December 13, 2020

University of Washington

What is Othello?

What is Othello?

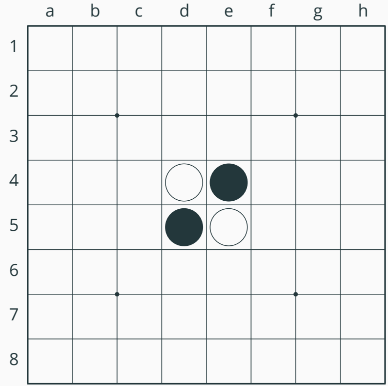


Figure 1: Opening Position

What is Othello?

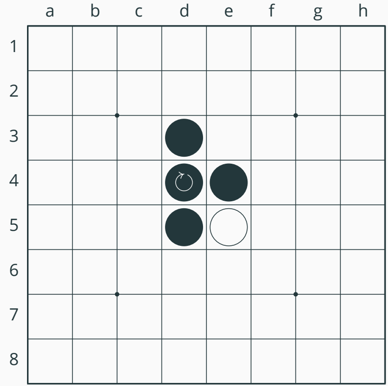


Figure 2: One option for black's 1st move

What is Othello?

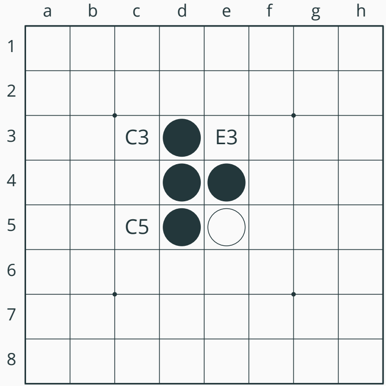


Figure 3: All of white's responses

What is Othello?

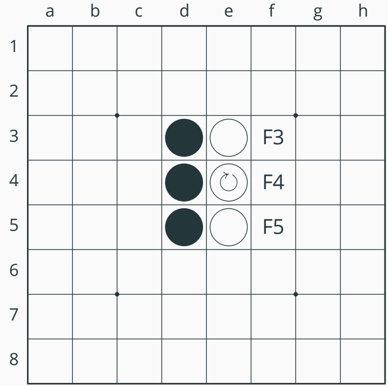


Figure 4: White chose E3 and all of black’s responses

What is Othello?

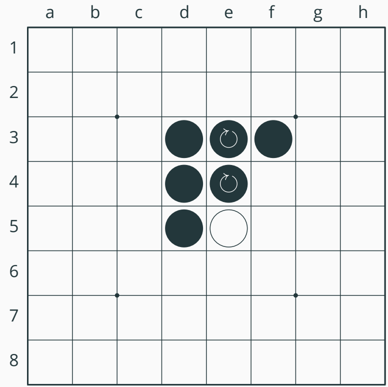
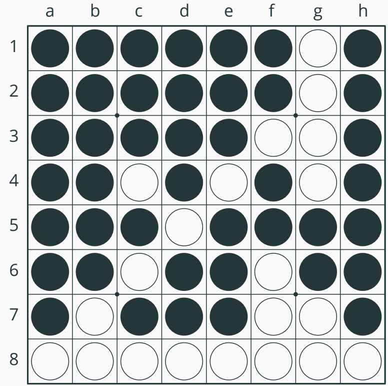


Figure 5: Black chose F3

What is Othello?



	a	b	c	d	e	f	g	h
1	●	●	●	●	●	●	○	●
2	●	●	●	●	●	●	○	●
3	●	●	●	●	●	○	○	●
4	●	●	○	●	○	●	○	●
5	●	●	●	○	●	●	●	●
6	●	●	○	●	●	○	●	●
7	●	○	●	●	●	○	○	●
8	○	○	○	○	○	○	○	○

Figure 6: AZ-O (black, 43) wins against Iagno “Medium” (white, 21)

What is AlphaZero?

What is AlphaZero?

AlphaZero uses a neural network which takes a state s and computes two things:

1. A policy $p_\theta(s)$ which is a distribution over the set of actions
2. A value $v_\theta(s) \in [-1, 1]$ which predicts the eventual winner of the game

What is AlphaZero?

The goal is to minimize the loss

$$L(\theta) = \sum_t \left((v_\theta(s_t) - z_t)^2 - \hat{\pi}(s_t)^T \log\left(p_\theta(s_t)\right) \right)$$

where

1. z_t is the outcome of the game from the perspective of move t
2. $\hat{\pi}(s_t)$ is an improved policy.

How to compute an improved policy

In order to calculate $\hat{\pi}(s)$ we use Monte Carlo Tree Search (MCTS). Define:

1. $Q(s, a)$ is the average z after taking action a from state s.
2. $N(s, a)$ is the number of times action a was taken at state s.
3. $P(s, a)$ is the probability of taking a at state s (from $p(s)$)

choose a maximizing the Upper Confidence Bound

$$U(s, a) = Q(s, a) + c_{\text{puct}} P(s, a) \frac{\sqrt{\sum_b N(s, b)}}{1 + N(s, a)}$$

AlphaZero-Othello?

What is AlphaZero-Othello?

1. 100% of the code is written by me
2. Multithreaded self-play
3. Multithreaded evaluation arena
4. Uses a single GPU on a single node (i.e. it is not distributed)
5. Self-play, evaluation, and training all happen synchronously (unlike in the original AlphaZero)

Results

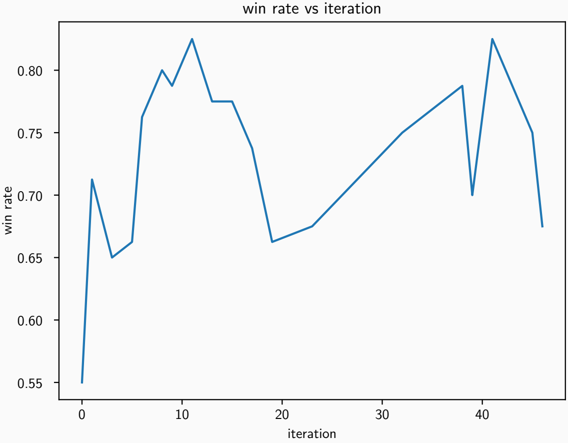


Figure 7: Win rate vs random agent

Results

Current best model (Iteration 43)

1. Reliably beats me (a novice)
2. Reliably beats Iagno “Easy”
3. Sometimes beats Iagno “Medium”
4. Never beats Iagno “Hard”

Conclusion: Not great but it did learn something

Excuses

Q: Why doesn't AlphaZero-Othello consistently improve?

A: The games of self play for AlphaZero-Othello are probably far too noisy to reliably improve on the policy.

- AlphaGo Zero: 7.84×10^9 MCTS iterations.
- AlphaZero-Othello: 1.075×10^5 MCTS iterations.

Solution: crank up the simulation count and (probably) the number of games.

I am not aware of a very strong Othello agent trained using RL techniques.

Thank you!