1 INTRODUCTION

1.1 Background

Oware, a well-known two-player board game that is particularly significant in African countries, presents itself in many variants, two of which are: the *Abapa* and *Nam-Nam* variants [1], which are most common in Ghana. Oware *Abapa*, often seen in international competitions, differs in its gameplay style from *Nam-Nam*, which is derived from the *Twi* dialect meaning *to roam* [2]. This name was given because it embodies a seemingly random and wandering nature in its strategic play. While this feature of the game diminishes its international competitive appeal, it presents a fascinating subject for exploration in reinforcement learning (RL). A lot of the terms used to describe the actions in the game communicate the concept of farming, as the game was invented during a time when such activities were the prevalent occupation [1].

Figure 1. The traditional oware board (Retrieved from: [3])

Oware *Nam-Nam* is played on a board with 12 pits and 48 seeds [2]. The board is divided into two rows of six pits each, and the players sit on either side of the board. The row of pits closest to a player is their territory. Figure 1 shows the board and two players represented as A and B.

The game starts with 4 seeds in each pit. The first player is usually chosen either randomly or by an agreed method, such as tossing a coin. Suppose that player A goes first, then player A picks up all the seeds from any pit in their territory and sows them separately

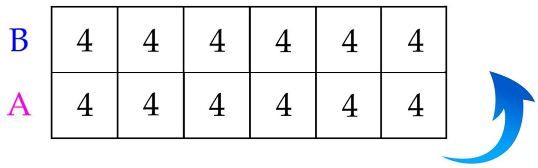

Figure 2. Oware board structure with number of seeds

and in succession individually in a counter-clockwise direction, starting from the next pit. If the last seed lands in an empty pit, player A's turn ends, otherwise, all the seeds located where the last seed landed are picked up by player A, and the sowing process continues until an empty pit is reached [2].

Figure 3 shows the board after player A's first turn. It takes four turns for player A to finish their first round, regardless of which pit they start from. The last seed of the round always lands in the first pit that was emptied at the beginning of the turn.

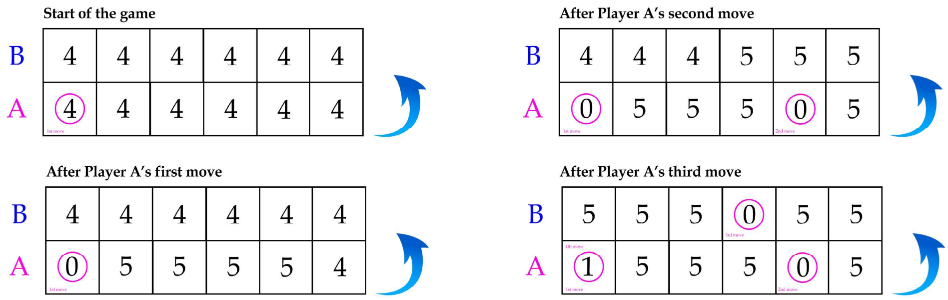

Figure 3. Oware board before and after Player A plays once

The game becomes more complex after the first round, as the board state depends on where player B decides to start their turn. The objective of the game is for a player to be able to capture more seeds than their opponent over several rounds of play. Seeds are captured when a player sows a seed that makes a group of four seeds in any pit on their territory. The player then takes all four seeds and removes them from the board. If the last seed of a turn makes a group of four seeds on the opponent's territory, the player can also capture those seeds. Captured seeds are stored separately and counted at the end of the round [2]. Some oware boards have special pits at the ends of the rows for this purpose, as can be seen in the first picture in Figure 1.

A round ends when all the seeds in the game have been captured. When there are only eight seeds left on the board, the first player to capture four seeds also gets the remaining four seeds. The winner of the round is the person who has captured the largest number of seeds during the round. In the case where, say, player A captures more seeds than the opponent, player A is said to have captured one territory (pit) of the opponent. This means that in the subsequent rounds, the territory of Player A is all the pits in front of him, plus one pit (usually starting from the rightmost pit of the opponent) of the opponent whose territory has been captured. The game ends when one player wins six rounds more than the opponent, meaning the player has captured all six territories of the opponent. This can take a minimum of six rounds but usually takes much longer when both players capture each other's territories continuously.

Many factors affect the complexity of a game's implementation in reinforcement learning (RL), such as the size and structure of the state and action spaces, the dynamics and stochasticity of the game environment, the reward function and its sparsity or delay, the presence and behavior of other agents or opponents, the observability and partiality of the game state, the need for exploration and exploitation, the generalization and transferability of the learned policies, and the computational and memory requirements of the RL algorithm [4].

In the case of Oware *Nam-Nam*, the structure of the state space is such that it is determined by the number of seeds in each of the 12 pits (6 per player). Each pit can have 0 to 48 seeds (the total number of seeds in the game). However, not all distributions are possible due to the game's rules. For instance, it's unlikely for all seeds to end up in a single pit. The state space can be estimated by the number of ways to distribute seeds across 12 pits, which is a problem of distributing n indistinguishable items into k distinguishable bins. The action space also consists of all the possible pits from which a player can start playing. This action space can be represented by all integers, from 1 to 12, however, the number of possible actions a player can take changes over the entire game for both players.

Oware has a deterministic environment since the environment changes only according to the movements made by the players. It also has a non-episodic or sequential environment since any action taken in the present directly affects the future. Oware also has an observable environment, since players can see what the state of the board is at any given point in time, and can determine what it will be like after their opponent takes a turn playing. Oware has a static environment since the environment of the game does not change while a player takes an action. The environment of Oware is discrete, since the number of possible states on the board is finite, although it is a considerably large number.

1.2 Objectives

This research focuses on the application of reinforcement learning in playing Oware *Nam-Nam*. The specific objectives of the study are highlighted below.

- We present a bot that plays and beats a human and a random player at the game Oware *Nam-Nam*
- We explore the seemingly random nature of winning in Oware *Nam-Nam* and gain insights into the significance of some hyperparameters of Deep Q-Networks in the process.

1.3 Structure of the Thesis

In the introduction chapter (1), the groundwork for the thesis is laid with a brief introduction to the Oware game, RL, and the reason RL is being used for Oware *Nam-Nam*. The literature review chapter (2) explores the existing literature on RL and its applications, especially in games, highlighting where Oware fits in this context. Details of the RL algorithms and models developed for Oware are presented in the methodology chapter (3), after which come the applications of these RL methods to the game, and a comparison of the performance with human players is done in the experiments chapter (4). The findings from the previous chapter and their implications are analyzed and discussed in the concluding chapter (5) as well as a summary of the main contributions and future research in the field of RL and Oware.

REFERENCES

[1] Wikipedia contributors. Oware. https://en.wikipedia.org/wiki/Oware, 2021. Retrieved [5th June 2024].

[2] The Oware Society. The Oware Society, 1994. Retrieved [29 May, 2024].

[3] Africa Heartwood Project. Oware Mancala Seed Game. https://africaheartwoodproject.org/product/oware-mancala-seed-game/, n.d. Retrieved [6th June 2024].

[4] Erdem Emekligil and Ethem Alpaydın. What's in a game? the effect of game complexity on deep reinforcement learning. In *Computer Games: 7th Workshop, CGW 2018, Held in Conjunction with the 27th International Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, July 13, 2018, Revised Selected Papers 7*, pages 147–163. Springer, 2019.

[5] Sutton, Richard S. and Barto, Andrew G. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[6] Rothmann, Marc and Porrmann, Mario. A Survey of Domain-Specific Architectures for Reinforcement Learning. *IEEE Access*, 10:13753–13767, 2022.

[7] Mnih, Volodymyr, Kavukcuoglu, Koray, Silver, David, Rusu, Andrei A., Veness, Joel, Bellemare, Marc G., Graves, Alex, Riedmiller, Martin, Fidjeland, Andreas K., Ostrovski, Georg, et al. Human-Level Control Through Deep Reinforcement Learning. *Nature*, 518(7540):529–533, 2015.

[8] Mohamed, Abdel-rahman, Dahl, George, and Hinton, Geoffrey. Deep Belief Networks for Phone Recognition. In *NIPS Workshop on Deep Learning for Speech Recognition and Related Applications*, volume 1, page 39. NIPS Workshop, 2009.

[9] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25, 2012.

[10] Das, Arghyadeep, Shroff, Vedant, Jain, Avi, and Sharma, Grishma. Knowledge Transfer Between Similar Atari Games Using Deep Q-Networks to Improve Performance. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–8. IEEE, 2021.

[11] Mnih, Volodymyr, Kavukcuoglu, Koray, Silver, David, Graves, Alex, Antonoglou, Ioannis, Wierstra, Daan, and Riedmiller, Martin. Playing Atari with Deep Reinforcement Learning. *arXiv preprint arXiv:1312.5602*, 2013.