Introduction

In this column, we describe the Model AI Assignment “FairKalah: Fair Mancala Competition”. After introducing the rules of Mancala (a.k.a. Kalah), we discuss the primary difficulty that its unfairness causes for AI competition assessment, and present a solution along with a description of a set of resources to aid in assignment adoption.

Rules of Mancala

Mancala, originally “Kalah”, is a variant of the Mancala game family (Russ, 2000) by William J. Champion, invented in 1940, patented in 1955 (Champion, U.S. Patent US2720362A, 1955-10-11), and enjoying popularity to this day in the United States. According to the Mancala World wiki (Kalah, n.d.), “In 1959, Kalah became the first remotely played computer game, when it was programmed by MIT students for the DEC’s PDP-1 computer.”

The Mancala board (Figure 1) is rectangular with 6 play pits per side for each player. To each player’s right is their larger score pit. Initially, 48 pieces are distributed 4 per play pit in the standard game as shown.

A player’s move in Mancala consists of selecting one of their non-empty play pits, picking up all pieces from that pit, and “sowing” them counter-clockwise, one per pit, skipping their opponent’s score pit. The Player 1 sowing pattern is shown in Figure 2.

If a player’s last piece is sown to their score pit, they take another turn. If a player’s last piece is sown to an empty play pit on their side, the player captures that piece and any in the opponent’s opposite pit (which may be empty). Any captured piece(s) are placed in the player’s score pit.

At the end of a turn, when no pieces remain in at least one player’s play pits, the opponent will score any remaining pieces. The player that scores the most pieces wins the game. If both players score the same number of pieces, the game is a draw (i.e. tie).

The Problem with Mancala

Mancala has been analyzed and shown to be unfair, with perfect play resulting in a first player win by 10 points (Irving et al., 2000). Indeed, any player with significant experience of the game knows that a first free move (4 in Figure 1) followed by a move closest to the first player score pit (1) will begin a sequence...
of capture threats that keep the second player on the defense and at a disadvantage.

While not only causing problems with gameplay, we have observed that this causes problems with Mancala AI assignments. We have used this game as a regular competitive introductory AI assignment since the fall of 2000, and have observed that the strong first player advantage makes it difficult to discern when a heuristic change yields an improvement to play. Attempts to randomize the initial distribution of pieces or having an initial sequence of forced play did not adequately address this problem.

The Solution: FairKalah

In the summer of 2019, this author and Taylor C. Neller computed a 24 piece endgame database and applied the MTD($f$) algorithm (van Horssen, 2019) to compute all fair initial Mancala boards where one or two pieces have been redistributed to different play and/or score pits. We call this fair Mancala play "FairKalah", and have observed that it not only improves play variety and quality, but it also makes it much easier for students to discern when they have improved upon a game play heuristic design.

This forms the basis for the Model AI Assignment "FairKalah: Fair Mancala Competition" which also includes Java and Python object-oriented implementations of:

- a Mancala/FairKalah game tree node,
- depth-limited minimax search,
- a text-based human player interface for testing and demonstration,
- a simple, real-time player using depth-limited minimax and a score difference heuristic, and
- round-robin FairKalah tournament code that produces game transcripts and a spreadsheet summary.

Also provided are suggested readings, video presentation of rules and an example demonstration game, and integration with the Ludii general game system.

In a typical two-week assignment for an introductory AI course, I have students work in pairs to:

- devise an improvement to the score difference heuristic, empirically testing at least two different heuristic evaluation performances with provided tournament code and a given simple player,
- implement alpha-beta pruning, thus speeding search and allowing greater search depth limits in the same real-time limits, and
- devise improved time-management, seeing iterative-deepening as an anytime algorithm and empirically testing how to better distribute reasoning time across a game.

When a current research project is concluded, the Model AI Assignments page for FairKalah will also include access to an optimal play dataset with each row containing an input state description (pieces per pit), output game value of that state, and output Boolean indications of optimal move(s). A longer project student project could apply machine learning to this data or reinforcement learning to gathered play data so as to learn superior heuristic evaluations, node ordering, etc.

Further, I anticipate that, while very strong heuristics may be learned thus, this assignment will remain relevant and valuable by applying tighter time and memory constraints for AI players.

It is my hope that these developments will breathe new life into Mancala game play, and that this assignment would benefit AI education. Contributions of ports to other programming languages, as well as additional resources to add to our Model AI Assignment repository would be appreciated.

References


---

**Todd W. Neller** is a Professor of Computer Science at Gettysburg College. A game enthusiast, Neller researches game AI techniques and their uses in undergraduate education.