Game of the Amazons Bot

Computer Science 322 - Artificial Intelligence

This bot is designed to play the Amazons board game against other players or bot which are communicating through the server which is automatically configured in the included cosc322-2015 library.

Features

* Minimax Search
  + Separate parallel and sequential minimax search implementations with alpha-beta pruning
  + Iterative deepening implemented for both sequential and parallel searches
  + Uses memory consumption and computation time as cut-off conditions
* Evaluation Functions
  + Complex evaluation function based on Martin Muller and Theodore Tegos' minimum distance evaluation function
    - <http://library.msri.org/books/Book42/files/muller.pdf>
    - Minimum distances on board are efficiently calculated and colored with iterative deepening instead of a depth first search traversal
  + Simple evaluation function based on the number of available moves for each side
    - Simple evaluation is able to traverse deeper in the game tree than the complex evaluation function but does not perform as well.
* GUI
  + A graphical representation of the game board is implemented and functions fully with local as well as server play
  + An ASCII game board representation is also outputted in the console
  + An ASCII minimum distance evaluation of the game board is also available for output
* Object Oriented Design
  + Leveraged abstract classes to create a framework for performing the searches
    - GameSearch and Evaluation abstract classes
  + Custom XML parser using JAXB
    - Replaces nanoXML implementation used in the example code provided
    - Object Oriented
    - Transformation layer between our internal board layout and the standard layout
    - Simple unmarshaling and marshaling of XML strings
* Utilities
  + Illegal move detection
  + End-of-game state checks
  + Remote play via the Amazons game server
  + Method for getting all possible moves for each queen
  + Record time-to-make-move and display after each turn with other statistics
  + Used in cut-off conditions for iterative deepening minimax search
  + Use previous information to stop search if the next depth is not feasible to traverse within the remaining time (does not waste time at the beginning of the game)