Juan Pablo Sarmiento (jps327)
Nikita Kiriy (nk327)
Brian Wojcik (bmw75)

Code Review 1 - Team King Fish

Executive Summary

We have worked on implementing Alpha Beta search for our AI but have run into bugs that prevent us from beating the Random AI. We plan on fixing this as soon as possible and then begin implementing further refinements to our strategy.

Proposal Summary

Our project involves a two-stage implementation of our AI for Modified Chinese Checkers. The first stage is to build an AI capable to beating a random player by heuristic move choices. The second stage is to create an adaptive AI that can learn and analyze the opponent's strategy in real time

System Architecture Summary

From our proposal we identified several components to be developed: a time distribution decision system to determine how much time should be spent in planning and on deciding each turn, a learning system that can train off of previous games, as well as a shortest-path finding system as an essential component to calculating costs of different paths that can serve as a heuristic. We have now also included the need for an alpha-beta search component to our AI as a separate module.

Implementation and Evaluation Status

Time distribution system: this is being included into our alpha-beta search component. It is currently in the implementation phase (almost finished with coding).

Learning system: we have not begun work on this yet. It still has not even been designed yet as it is part of our Stage 2 AI.

Shortest-path finding system: this has been included into our alpha-beta search component as part of our utility cost calculation. It has been revised and no longer entails calculating an actual shortest path, because the shortest path to the target area for each marble will change each turn due to movements that the opponent makes, as well as due to any special marbles that may be added. Calculating a tree of shortest paths would be too costly memory-wise. Instead, we use the functions given in Board.java to calculate distances to the target area for each marble, and we then use this in the calculation of a state's utility.

Alpha-beta search: This is currently the core of our Stage 1 AI. We are in the implementation phase (almost finished with coding). We had implemented it, but our results were not as expected (as in, our AI was not beating the Random AI player given to us due to making illegal moves as a result of bugs in our state representation), so we had to debug what was going on. We decided to change how our system was being modularized. We are now implementing an "Alpha Beta Black Box" - a separate module to which our AI will be able to ask for the next move it must perform.

The AI will either receive the next move that must be done or be told that it needs wait some more time to compute the next move, which is where our time distribution system will come in. This component will be evaluated by using our "two-move checkmate" strategy described in our project proposal. We will give a series of initial board states, starting from the trivial board state where it will only require two moves to win. We will then start moving back, each time requiring more moves in order to achieve a win. Ultimately, we should be able to give it the default initial board state (all pieces in their home area) and our AI should be able to defeat the Random AI and the Greedy AI players that are given to us.

Project schedule timeline

We have fallen slightly behind schedule due to bugs in our Alpha Beta search; however, in our project proposal, we had planned to have a smarter AI by October 18, so we should have alpha-beta search working by then and begin refining our implementation with new strategies. By Nov 1 (second code review), our Stage 1 AI should be complete and we should have begun working on our Stage 2 AI.

One reason for not moving as fast as we initially planned can be that we used to have 3 members in our team but one member dropped the course before we had even begun implementing our AI, so all work until now has had to be done with only two members. This will change now with the recent addition of a new member, so dividing work should be more comfortable and progress can be made quicker.

Proposal Modifications

The details behind any deviations from our initial proposal have already been described in our Implementation and Evaluation Status where we describe how we've been implementing different components of our AI as opposed to how we had initially planned to do them in our project proposal. As far as high-level modifications go, there are no changes: we are still striving for a two-stage implementation of our AI and still plan on adhering to the same deadlines, with the ultimate goal being a learning AI that can analyze the opponent's strategy in real time and adapt to it.