

Shreyas Shankar (20171128)

Swetanjal Datta (20171077)

Artificial Intelligence 25 February 2019

Intermediate Assignment Report

Introduction:

We plan to use alpha-beta pruning with minimax algorithm. The heuristic used will be a combination of the possible small square captures, with importance given to small square captures resulting in game victory (Big square capture).

Heuristic:

The heuristic will evaluate the following positional advantages:

1. States leading to victory would be given a very high value
2. States leading to game loss would be penalized the heaviest
3. In case of a draw, state will be evaluated according to a linear combination of values of the squares captured as given.
4. In Big Square, capture of center square is valued most, followed by corner squares and then edge squares.
5. In Big Square, Small square captures leading to capturing that row, column or diagonal in that corresponding Big square is valued
6. In small squares, an immediate capture of the small square is preferred foremost.
7. Moves leading to potential row/column/diagonals in small square is valued.
8. In the beginning, seek to move in the center square of the board (both big square and small square).

Implementation of Alpha beta pruning:

```

1 evaluate (node, alpha, beta)
2     if node is a leaf
3         return the utility value of node
4     if node is a minimizing node
5         for each child of node
6             beta = min (beta, evaluate (child, alpha, beta))
7             if beta <= alpha
8                 return beta
9         return beta
10    if node is a maximizing node
11        for each child of node
12            alpha = max (alpha, evaluate (child, alpha, beta))
13            if beta <= alpha
14                return alpha
15    return alpha

```

Note: utility value is the value computed according to above mentioned heuristic.

Competency of Heuristic/Utility Function

We feel our heuristic is competent because we are giving very high utility to a win, a very low(negative) utility for a loss and for a draw the utility is proportional to the scoring as mentioned in the description of the assignment.

Once, we have a strong heuristic for the larger boards, we turn our attention to each smaller board where we give winning and blocking the opponents win the highest priority, followed by trying to capture squares in a way so as to win that square entirely.

In such an open ended game, we strongly feel that winning small squares greedily is the best possible option, more so because of the bonus move. In other words trying to win a small square can never put me in a bad situation and blocking the small square win of the opponent can never put me in a worse situation compared to other moves. All these observations led us to develop the following consistent heuristics.

Though we are making a greedy choice, we are very confident that this heuristic will turn out to be strong enough.

Challenges Ahead

The detailed heuristic function clearly indicates that a lot of tedious coding needs to be done. We feel we are prepared to handle such implementation details.

We need to assign appropriate values for utility which we plan to do by Monte Carlo Search techniques if time/infrastructure permits, otherwise by simple intuition and logic.