Heuristic Function

As mentioned above (in the agents section) the algorithms use different heuristics to form various evaluation functions. Here we will describe our attempts at creating several fine heuristics..

1: Eval I – Piece to value

Our most basic function counts for the player who builds the tree the value of his pieces and subtracts from it the value of opponent’s pieces. Since Kings are considered more powerful than regular pawn, we double their
value in comparison to them.
Specifically:

Pawn’s value = 1
King’s value = 2

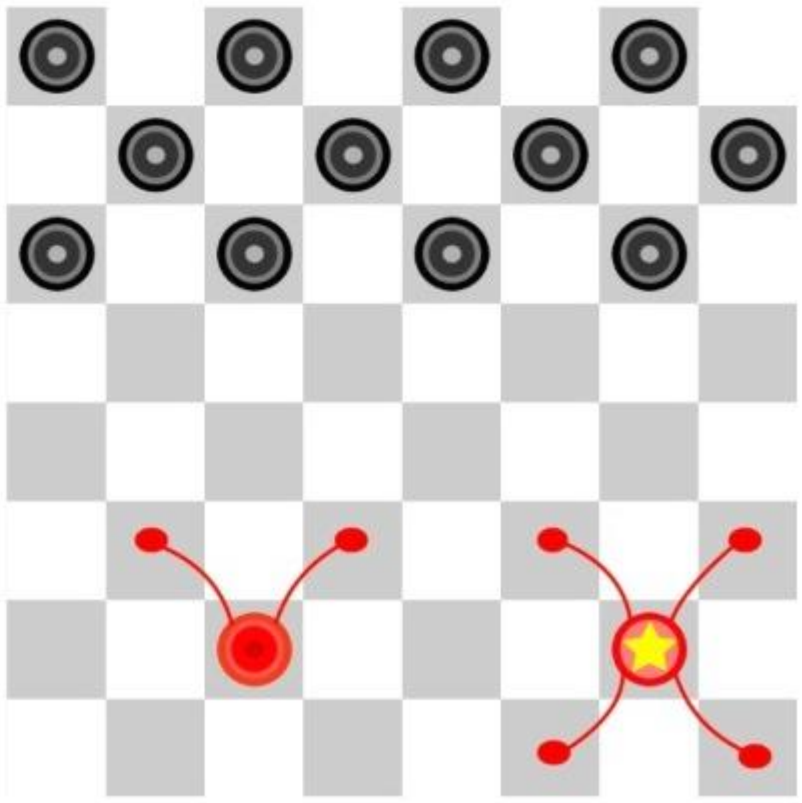


Figure 1.1: Move of a man and move of a king.

2: Eval II – Piece & Board part to value

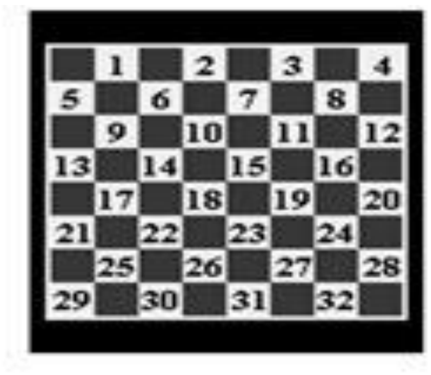
This function is built over the Piece to Value function (in a sense of again evaluating pieces and subtracting) and
attempts to take into account some more properties of the game. It's easy to understand that advanced pawns are more threatening than pawns that are on

the back of the board. Therefore, since advanced pawns are much
closer to become Kings, we give them extra value in our evaluation. Of course, we still evaluate kings more
than any pawn.
Specifically:
We split the board into halves.
Pawn in the opponent's half of the board value = 7
Pawn in the player's half of the board value = 5
King’s value = 10

Figure 3.3: Special situation for describing the rules proper implementation

3: Eval III – Piece & Row to value

This function is a small modification to the previous function in a sense that this function gives specific value of
row to heuristic.
Pawn’s value: 5 + row number
King’s value = 5 + # of rows + 2



Similar Apps In Checker

1-Draughts

2-Halma

3-Konane