Assignment – 2

Foundation of Artificial Intelligence – CS2701

Name: Khushboo Thaker
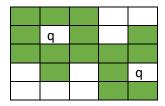
ID: 4147510

Email: kmt81@pitt.edu
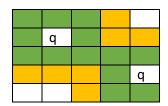
1. **The start of the game has many possible moves. How does your program deal with the huge branching factor?**

   To deal with huge branching factor I have generated successor nodes in step:

   1. First of all, the blocks adjacent to other player queens will be used to generate the moves. Also the program considers the blocks which can block the path of two of other player's queens together

   2. If these child nodes are not enough, then the blocks adjacent to the locations in first step will be included to generate child nodes

   This will go iteratively till we get the enough child nodes

   NOTE: Assumption is the blocks near the opposite queens are of prime importance

2. **There is a time limit for the turn. What does your program do to make the most use out of the time you have?**

   To deal with time limit my program has modified minmax algorithm for iterative deepening – level based. And there is a timer which signals the minmax program at the end of 14 seconds and the program returns the latest minmax result in case it is not done with all the levels

3. How well does your automatic player compete against a human player?

I tried playing with my game. But most of the time I won the game except the times if I missed an important point and played a wrong step. As my strategy is to not let them in the second half of the board which is not the part of my heuristing

4. What different heuristic functions have you tried before settling on your final choice? What ideas worked? What didn't? Be sure to describe your final heuristic function in sufficient details.

The different heuristics I tried are as below:

1. Number of blocks reachable by the opposite queens
2. 1/ Number of blocks reachable by my queen
3. Number of adjacent blocks open for my queens
4. Number of adjacent blocks open for opposite player queen
5. 1/ (Number of blocks I am reachable – Number of black opposite player is reachable)
6. Random

I tried playing all the heuristics against random player for 100 runs and I found that the No of adjacent blocks heuristics wins most of the time although sometimes with large number of moves and very few blocks left. So I selected it to be the final heuristic

My final heuristic is 4) No of adjacent blacks open for opposite player queen. As my code is based on prime locations this heuristic makes more sense

Below is the example: Table 1	Intial Board condition given to Player with white queen -Q

Initial Board (Table 1): board with Q at row2 col3, Q at row3 col3, q at row7 col4.

		Q							
		Q							
		q							

Possible Move1

		q							
		x							
				Q					
				q					

So as the move 1 helps in blocking 2 queens this would be given priority over the move 2

Possible Move 2

		q							
						x			
			Q						
			q						

References:

Lecture Notes

Class Notes

https://en.wikipedia.org/wiki/Game_of_the_Amazons
Jens Lieberum, An evaluation function for the game of amazons, Theoretical Computer Science, Volume 349, Issue 2, 2005, Pages 230-244, ISSN 0304-3975, http://dx.doi.org/10.1016/j.tcs.2005.09.048.