3-6-9: Prolog Game

Artificial Intelligence Project

16/01/2015

Team Members

Ahmed Hassan

Omar Samir

Tho Vo

1. Description

3-6-9 is a 2-player board game, played on a 3x3,6x6 and 9x9 board. The objective of the game is to get highest points at the end of the game. The game ends when every cell in the board is completely filled by stones

2. Rules

At each turn, player will put a stone on an empty cell. How to calculate the points?

- For every 3 stones in any line (orthogonal or diagonal) the player gets 1 point
- For every 6 stones in any line (orthogonal or diagonal) the player gets 2 points
- For every 9 stones in any line (orthogonal or diagonal) the player gets 3 points

	●		●		●
		●	●		
			1		
				●	
●	●	●	2	●	●

1. If the next player plays at [1] he will get 2 points (for making two lines with exactly 3 stones, a vertical and a diagonal one). He also made a line with 4 stones, which does not earn him any point.
2. If he plays at [2], he will get 3 points, 2 for the horizontal line of 6 stones, and 1 for the vertical line with 3 stones.[1]

3. Heuristics

We use 3 type of heuristics for this game. All these heuristics will be applied to a weight matrix where the computer can use to decide its next move on the board:

1. **Heuristic 1:** Compute the sum at each turn for each cell dimension (row, column, diagonals), then fill the weights matrix with the sums: weight = maximum sum of cell dimensions

3	●	3	●	3	●
2	2	●	●	2	2
1	2	2	3	2	2
2	1	2	2	●	2
1	2	2	2	2	3
●	●	●	5	●	●

The sum will be calculated by adding the maximum stone at the row, column and in two diagonal . We add this sum on every cell and the computer will choose the first one highest value.
In this example, the computer will choose to place the stone at [6,4]
The green one indicates computer's choice

[1] http://www.di.fc.ul.pt/~jpn/gv/369.htm

2. **Heuristic 2:** Compute the weight at each turn for each cell dimension using the same method as the first heuristic, then quantize the weights to discrete level, where the levels should take into consideration the significance of a certain move relative to other moves

2	●	2	●	2	●
5	5	●	●	5	5
1	5	5	2	5	5
5	1	5	5	●	5
1	5	5	5	5	2
●	●	●	7	●	●

The level of the stone relates to the player's next move:
1: Level = 1, The 1st stone, it means when the computer put the 2nd stone here, the player can put 3rd stone to get point, we avoid it by add the low level for it
2: Level = 5, The 2nd stone,it means the computer can put a stone here to get 1 point,
3: Level = 2
4: Level = 0, if the computer puts the stone right after this one, the player can put the next stone and get 2 points
5: Level = 7, computer may get 2 points
6: Level = 1 … and so on

3. **Heuristic 3:** Compute the weights for each cell using the same method as the second heuristic, then multiply those weights by a dynamic ratio that depends on the number of turns played, as well as the size of the board.
This heuristic aims to collect as much point as possible.

```
Weight =< 3,
    Dynamic_weight is (Weight_leveled / ((Turns+1)/Size*Size)).
Weight > 3,
    Dynamic_weight is (Weight_leveled * ((Turns+1)/Size*Size)).
```

6	●	6	●	6	●
15	15	●	●	15	15
3	15	15	15	3	15
15	15	15	15	15	15
3	15	15	15	3	15
●	●	●	2.3	●	●

Turns = 11
Size = 6
[1,1] has Weight = 3 and Weight_leveled = 2 so the Dynamic_weight = 6

4. Tests

- **Comparison of AI using 1st heuristic and 2nd heuristic:**

Computer play first, put the stone at [1,1][2] in both cases

1st	[2,3]	[1,2]	[1,3](1)[3]	[1,4]	[3,3](2)	[1,5]
2nd	[2,3]	[1,2]	[1,3](1)	[3,3](1)[4]	[2,2](2)	[2,1](2)

[2] Blue indicates the computer's move
[3] The number in bracket indicates the player's score
[4] The number in this blue bracket indicates the computer's score

1st heuristic has an intention to get the 6 stone in one line and it really doesn't care about the small points it can get just because base on how it calculate the weight, the row, column or diagonal with have the maximum stone.
2nd heuristic aims to take as much point as possible.

- **Full game moves with AI using 2nd heuristic :**

[1,1] → [2,3] → [1,2] → [1,3](1) → [3,3](1) → [2,2](2) → [2,1](2) → [4,4] → [3,1](4) → [3,2](4) → [4,5](5) → [4,3](6) → [1,4](6) → [3,4](7) → [2,4] → [4,2](8) → [5,1] → [6,5] → [1,5](7) → [1,6](11) → [2,5] → [2,6](13) → [3,5](9) → [3,6](17) → [5,5](11) → [6,1] → [4,1](13) → [5,2](20) → [4,6](15) → [6,2](23) → [6,6](17) → [5,6](25) → [5,3](18) → [5,4](28) → [6,3](21) → [6,4](33)

- **Selected game moves with AI using 3rd heuristic :**

The test shows that the aim of 3rd heuristic is collect small points instead of large points in early turn. Here we calculate the weight matrix at (turn 8)

●	●	20	20	20	20	Instead of picking [5,5], it choose to pick [1,3] to get 1 point first
20	●	●	20	20	20	
20	20	●	20	4	4	
20	20	20	●	20	4	
20	20	20	4	1.7	20	
●	20	20	20	20	●	

Then we calculate the weight matrix at a later stage of the game (turn 26)

●	●	●	●	●	●	with the maximum weight, it now chooses the large points instead of small one
5.2	●	●	●	●	●	
●	0	●	●	●	0	
●	0	●	●	0	●	
●	5.2	●	●	●	0	
●	0	5.2	●	0	●	