Rapport Assignment 4

Mancala

rICKARD sÖRlin

|  |  |
| --- | --- |
| *Kurs: Artificiell intelligence 1*  *Kurskod:* DVA264  *Högskolepoäng:* [7.5 hp]  *Program: Kandidatprogrammet tillämpad AI* | Handledare: Miguel LeonOrtiz  Examinator: Miquel LeonOrtiz  *Datum: 2023*-06-27  *E-post: r*sn21005@student.mdu.se |

def evaluate(boardstate):

""" Function handels the Heuristic evaluation of the current state of the Mancala game to guide the algorithm using different game domain tactics that is calculated using weighted linear functions. """

# Nr 1 - Calculate the score difference in players Mancala , only needed to beat player 3 but most important weight!

scoreweight = 1 # (1 <=> )

# Get the amount of marbels in player1 mancala

playermancala = boardstate[6]

# Get the amount of marbels in opponents mancala

opponentmancala = boardstate[13]

scorediffrens = scoreweight \* (playermancala - opponentmancala)

# Need to use more game domain knowledge to win over 4 and 5 loses every time

# Player 4 playstyle looks like he steals when he can sow need to take that in to account

# Nr 2 Check for empty holes on each player side it open ups for stealing but also losing stones

emptyholesweight = 0.5 # (0.5 <=> 0,7 )

playeremptyholes = 0

for i in range(6):

if boardstate[i] == 0:

playeremptyholes += 1

opponentemptyholes = 0

for i in range(7, 13):

if boardstate[i] == 0:

opponentemptyholes += 1

# Calculate emptyholesdiff higher playeremptyholes value gives positive else negative.

emptyholesdiff = emptyholesweight \* (playeremptyholes - opponentemptyholes)

# Nr 3 Gain score by stealing should be prioritized as well important to prevent get stolen from that move

stealweight = 0.45 # (0.45 <=> 0.65 )

playersteal = 0

# Check amount of empty holes player1 has and where opposite hole that belongs to opponent is not empty

for i in range(6):

if boardstate[i] == 0 and boardstate[12-i] > 0:

playersteal += 1

# Check amount of empty holes opponent has where opposite hole belongs to player1 and is not empty

opponentsteal = 0

for i in range(7, 13):

if boardstate[i] == 0 and boardstate[12-i] > 0:

# Player1 will feel double as mutch pain "negativ opponentsteal" as opponent if they have same amount of empty hole! I really need to watch my back for 4 and 5 sow i prevent them from stealling from me.

opponentsteal += 2

# Calculate scoresteal value higher playersteal gives positive value else negative to "punish" that move

stealpotential = stealweight \* (playersteal - opponentsteal)

# Nr 4 - Try to starw 4 & 5 have hard to beat them count the amount of stones each player has during game.

# Give small positive score for more marbels on player1 side then opponent else negative if less marbels.

# At the end it will be counted also to the players score and during the game it will be less for opponent to play

# with and gain score. Maybe try later make the weight dynamic during game play if about to lose to tweak more!

marbelsweight = 0.1 # (0.05 <=> 0.1)

# Count amount of marbels each plater has on their side

playermarbels = sum(boardstate[:6])

opponentmarbels = sum(boardstate[7:13])

# Calculate the marbel diffrens more marbels on player1 "max" side will give positive value else negative.

marbelsdiffrens = marbelsweight \* (playermarbels-opponentmarbels)

# Calculate the sum of total evaluation value based on the different mancala game domain tactics

# using weighted linear functions of current state!

# "Max" will try get high positive evaluationvalue and "Min" wants as large negativ evaluation value as possible

evaluationvalue = scorediffrens + stealpotential + emptyholesdiff + marbelsdiffrens

return evaluationvalue