ARTIFICIAL INTELLIGENCE

Russell & Norvig

Games, evaluation functions
Tic-Tac-Toe

- The first player is X and the second is O
- Object of game: get three of your symbol in a horizontal, vertical or diagonal row on a 3x3 game board
- X always goes first
- Players alternate placing Xs and Os on the game board
- Game ends when a player has three in a row (a wins) or all nine squares are filled (a draw)
Evaluation functions

• It is usually impossible to solve games completely
  • Connect 4 has been solved
  • Checkers has not
• This means we cannot search entire game tree
  • we have to cut off search at a certain depth
    • like depth bounded depth first, lose completeness
• Instead we have to estimate cost of internal nodes
• We do this using a evaluation function
  • evaluation functions are heuristics
• Explore game tree using combination of evaluation function and search
Evaluation function for Tic-Tac-Toe

• A simple evaluation function for Tic-Tac-Toe
  • count number of lines where X can win
  • subtract number of lines where O can win

• Value of evaluation function at start of game is zero
  • on an empty game board there are 8 possible winning lines for both X and O

8-8 = 0
Evaluating Tic-Tac-Toe

\[
\text{evalX} = \text{(number of lines where X can win)} - \text{(number of lines where O can win)}
\]

- After X moves in center, score for X is +4
- After O moves, score for X is +2
- After X’s next move, score for X is +3
Evaluating Tic-Tac-Toe

\[ \text{evalO} = (\text{number of lines where O can win}) - (\text{number of lines where X can win}) \]

- After X moves in center, score for O is -4
- After O moves, score for O is +2
- After X’s next move, score for O is -3

\[
\begin{align*}
8-8 & = 0 \\
4-8 & = -4 \\
4-6 & = -2 \\
2-6 & = -4 \\
\end{align*}
\]
A Better Evaluation Function

\[ \text{Eval} = 3 \times X_2 + X_1 - (3 \times O_2 + O_1) \]

Where

- \( X_2 \) is the number of lines with 2 X’s and a blank
- \( X_1 \) is the number of lines with 1 X and 2 blanks
- \( O_2 \) is the number of lines with 2 O’s and a blank
- \( O_1 \) is the number of lines with 1 O and 2 blanks

Return \( \text{MaxInt} \) (or Infinity) if X wins, -\( \text{MaxInt} \) (or –Infinity) if O wins

This is static evaluation function. Returns one value, positive for advantage to one player, negative means advantage to the other. Zero indicates it is even.
• Simple code for Tic Tac Toe:

• [http://www.cs.olemiss.edu/~dwilkins/CSCI531/tic.c](http://www.cs.olemiss.edu/~dwilkins/CSCI531/tic.c)