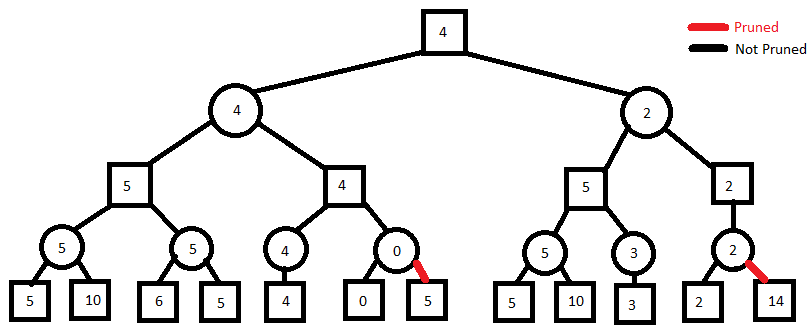
Artur Meletyan

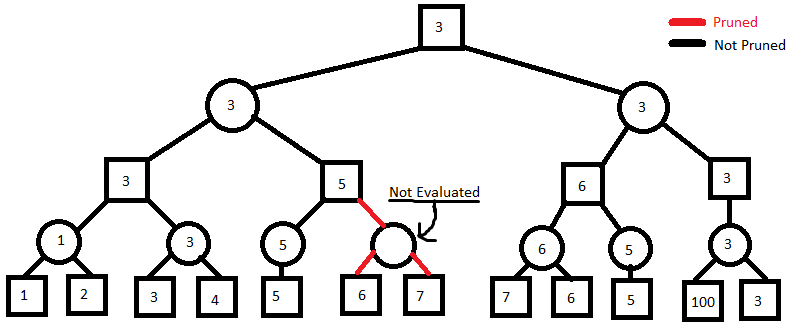
9/20/15

CSCI 3202 Assignment 4

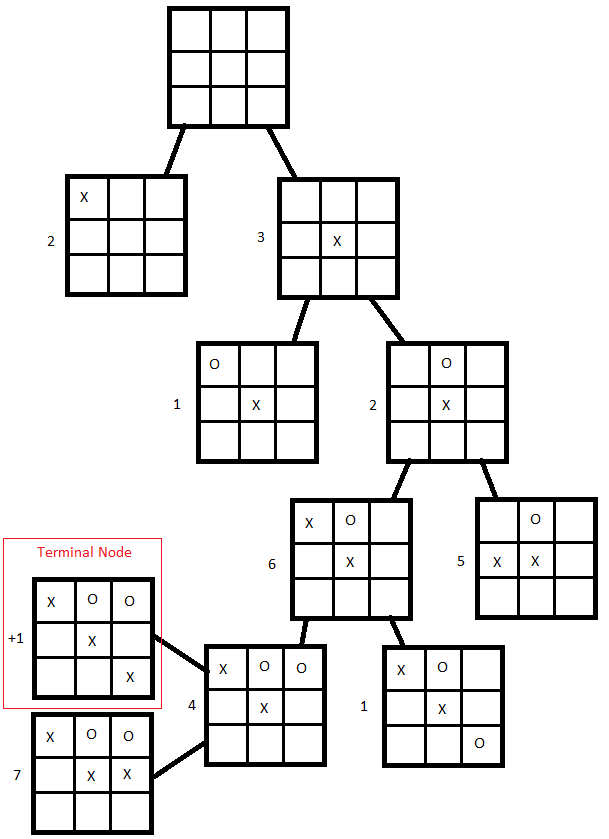
- -

1. 

- -

1. 

- -

1. 
2. X = highest number of adjacent X’s (by either row, column, or diagonal)

O = highest number of adjacent O’s (by either row, column, or diagonal)

eval(s) = X(s) – O(s)

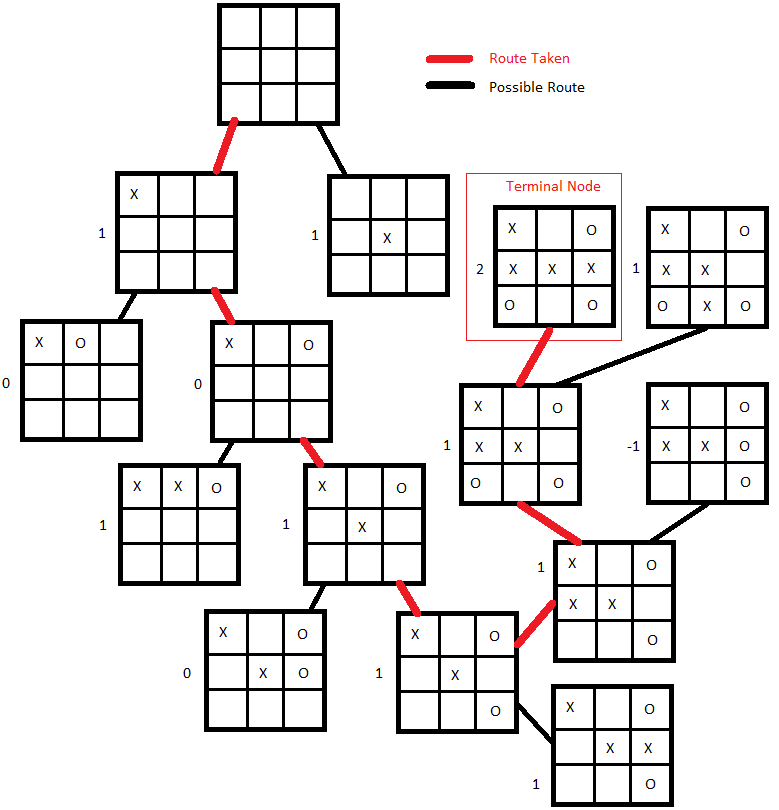
This evaluation function for a game of tic-tac-toe weighs the highest number of adjacent X’s to the highest number of adjacent O’s in a future state. The highest scoring future state is chosen. If the future states are equal, either one is arbitrarily chosen. If one of the future states has X = 3, it is chosen regardless of the value of O. If one of the future states has O = 3, it is avoided regardless of the value of X. If both future states have either X = 3 or O = 3, either one is arbitrarily chosen.

I believe this evaluation function is a reasonable approach in tic-tac-toe because it has a very direct method to reaching the goal of having three X’s in a row, column or diagonal by simply trying to place three X’s next to each other as quick as possible. Also, it accounts for how much a future state could benefit an opponent. In the case that a route is blocked by an O, the evaluation function would adaptively build off the next highest number of adjacent X’s.

Nonetheless, this evaluation function has many shortcomings. The amount of times that it will make an arbitrary move is too much to be consistently reliable. The number of X’s and O’s on the board will even out every other turn and given the limited space, the probability that the variables X and O will equal is high. Another shortcoming comes to the actual act of making an arbitrary move. The function neglects any merits that a future state has if it is not related to the amount of X’s that are adjacent to each other (such as blocking the opponent from matching three O’s). Hence, it will ignore possible strategic victories by chance.

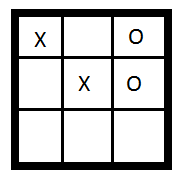
This evaluation function that I created is capable of favorable results, but it is unreliable when compared to the one given for this assignment due to its lack of robust analysis.

NOTE: The partial tree diagram generated by my evaluation function is on the next page.



1. The utility obtained by MAX using minimax decisions against a suboptimal MIN will never by lower than the utility obtained playing against an optimal MIN because an optimal MIN inherently tries to decrease the utility of MAX. A suboptimal MIN, by playing in a suboptimal manner, will only ever yield utility to MAX that is greater than or equal to an optimal MIN (one that always yield the lowest possible utility to MAX).

EXAMPLE:

Given the following tic-tac-toe scenario and assuming it is O’s turn…

An optimal O would decide to place an O at the bottom left corner of the board. A suboptimal O could do the same thing, but is also able to place an O in positions that would not win the match since any of them would be suboptimal. Therefore, the assertion since the suboptimal MIN would only ever be able to match an optimal MIN or do worse.