Game -Play (1)

Typical Cases

- 2 person Game
- Players alternate move
- $\boxed{\text{Zero Sum}}$ One player's loss is the other's gain
- $\boxed{\text{Perfect information}}$ Both players have access to complete information about the state of the game. No information is hidden from either player.
- No chance involved.

Game - Play (2)

↳ How to play a game?

Challenge 1

Representing the board.

⇝ Consider all the $\boxed{\text{legal}}$ moves you can make

challenge 2 Generating all legal next boards

⇝ Compute the new position resulting from each move

* ⇝ Evaluate each resulting position and determine which is best.

Challenge 3

Evaluating a position

⇝ Make that move

⇝ Wait for your opponent to move and repeat

Game - Play (3)

- Nodes — State or Position
- Branches — Possible action or decision that a player can make at that Point
- Payoff value:
  → Associated with the terminal nodes of the game tree

  → It quantifies the benefit or loss to each player when the game reaches that node

  → Single value (Zero Sum / Two Player game)

  → Vector of values (when more than two Players / non Zero-sum)

Game Play (4)

Evaluation function [f(n)]

→ "Goodness" of a game Position

$f(n) \gg 0$ → Position n is good for me and bad for you

$f(n) \ll 0$ → position n is bad for me and good for me

$f(n)$ near 0 → Neutral position

$f(n) = +\infty$ → <u>WIN</u> for me

$f(n) = -\infty$ → <u>WIN</u> for You

Game – Play (5)

Zero Sum * One player's gain or loss is exactly balanced by the loss or gain of the other players.

* The total sum of gain / loss ⇒ Zero

↳ when one player wins, then the other one losses.

{ Two player → A
→ B
Player A wins → +1
→ −1 (for Player B) }

Non Zero Sum * The sum of the outcomes for all players does not neccessarily equal Zero.

* The gain of one player does not neccessarily mean a loss of another player.

* All can gain. All can lose. Gain/loss can be distributed equally.

Game Play (6)

Mini Max Problem

(1) Create Start node as MAX node with current board configuration

(2) Expand nodes down to some depth of look ahead in the game

(3) Apply the evaluation function at each of the leaf nodes

(4) Back-up values for each of the non-leaf nodes untill a value is computed to the root node.

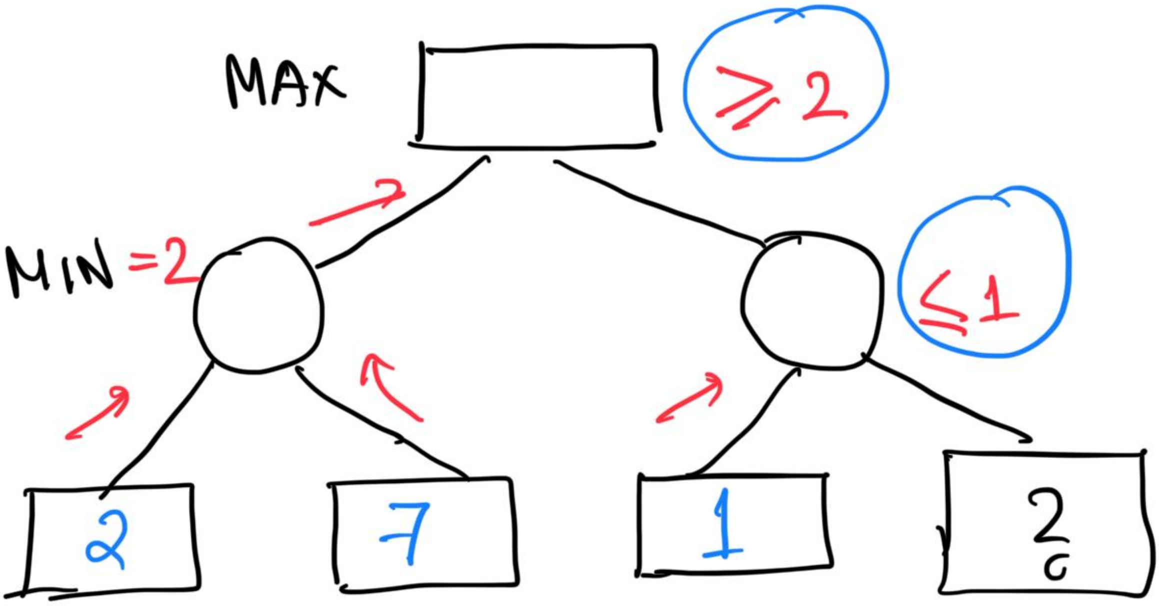
↳ MIN Node: backed up value is the MIN value of Children

↳ MAX Node: backed up value is the MAX value of Children

Game Play (7)

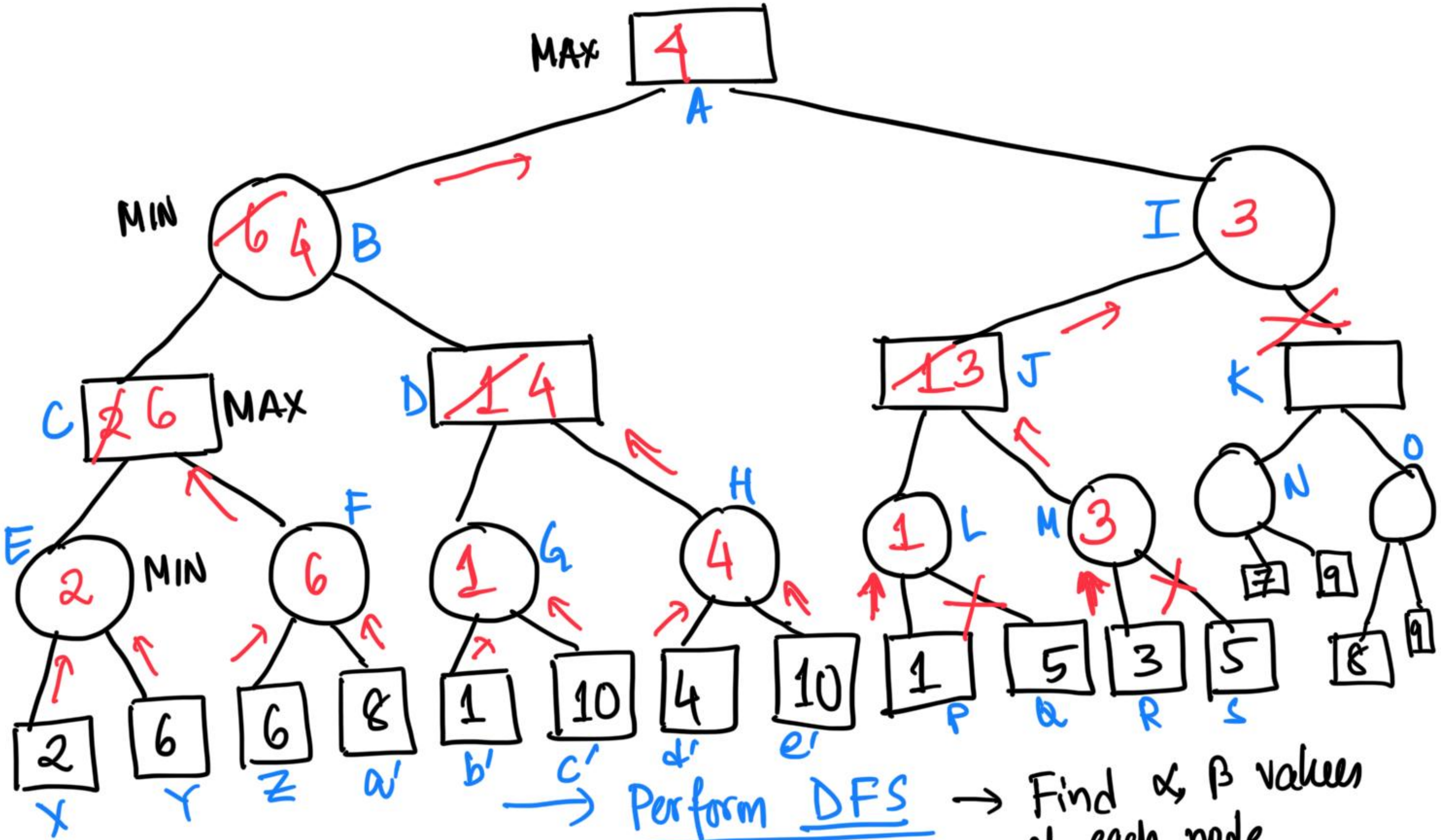
How to improve Minimax search?

↳ If you have an idea that is surely bad, Don't take the time to see how truly awful it is!!



It can't affect the value of root node ← No need to compute the value of this node

Game Play (7)



→ Perform DFS → Find α, β values at each node

→ X → Pruning ← $\boxed{\alpha \geq \beta}$

Game Play (8)

α - β Pruning

α: Tentative value at MAX Node
β: Tentative value at MIN node

→ Traverse the search tree in DFS

→ At each MAX node n
$\alpha(n)$ = max value found so far

→ At each MIN node n
$\beta(n)$ = min value found so far

→ α value start at -INF and only increases

→ β value ~~start~~ at +INF and only decreases

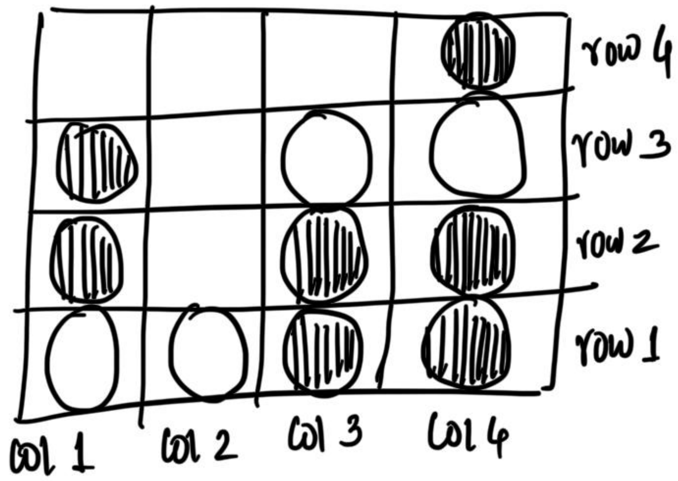
→ β cutoff Given a max node n, cutoff the search below n if $\alpha(n) \geq \beta(i)$ for some MIN node ancestor i of n

Don't generate or examine any more children of n

→ α cutoff Stop searching below MIN node n if $\beta(n) \leq \alpha(i)$ for some MAX node ancestor i of n

Game Play (9)

"MINI - FOUR"



- Two Players (●, O) board game
- Connect four of one's own disc in a row/column/ diagonal.

Game Play

- Players alternate turns, dropping disc into any column.
- Disc falls to the lowest available position in the chosen column.
- Game terminates → when one player successfully connects four disc in a row, column, diagonally <u>OR</u> the board is completely <u>filled</u>.

Game Play (10)

How to design static evaluation function for MINI-FOUR

Input : <board state>
Output : <Number>

Only considers current state

There can be different features that we can consider.

For example * number of unblocked runs of length 2,3 You have — number of unblocked runs of length 2,3 Your opponent has.

* IS the State a winning configuration?

* How many open space are adjacent to You vs opponent

....