HOMEWORK – 2

CS 6343

Submission Date: February 22, 2015

Submitted By: Kanika Kapoor

kxk140230

Ques-1 4-square Game

1	2	3	4
A			B

← starting position

Given

→ Player A moves first

→ each player must move his token to an open adjacent space in either direction

→ If opponent is in adjacent space, then player may jump over the opponent to the next open space

jump possible

1	2	3	4
↙	B	A	

→ Goal: when one player reaches the opposite end of the board

→ If player A reaches space 4 first, then value of the game is +1. If player B reaches space 1 first, then value of game is −1.

Part (a) Designing Game Tree

Conventions

① state (S_A, S_B) → token locations

② Put terminal state in and write game value $(+1, -1)$ in O.

③ Put loop states in .
Put ? as value to loop states

(a)

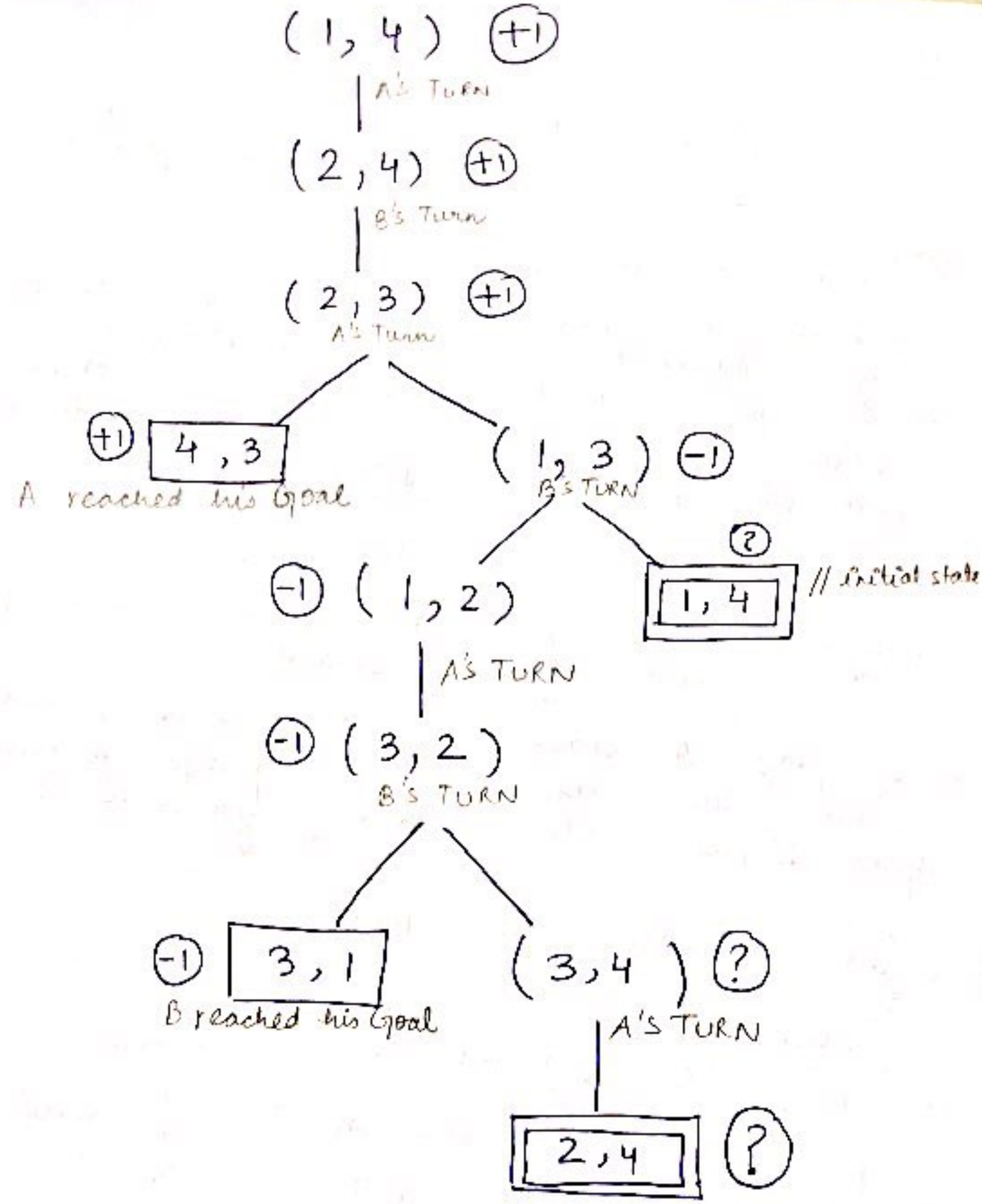


Fig: Game tree for Four-Square Game using the given conventions:

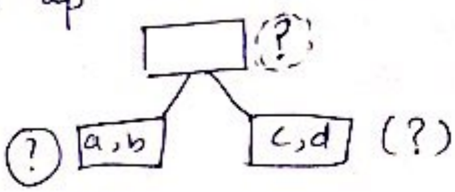
loop states

Terminal states

(b) Given the "?" values, they are handled based on the assumption ~~of~~ minmax algo that we try to maximize the utility because the opponent will also play perfectly to win ie if the agent has a choice between winning the game and entering a ? state, he will choose to win.

Thus, $\min(-1, ?) = -1$ and
$\max(+1, ?) = +1$

In case where all the successors have ?, then ~~the~~ the backed up value is ?



The ~~po~~ backing up of values continues from leaf nodes towards the root, one layer at a time, and eventually reach the top of tree, at the point MAX chooses the move that leads to the heighest value.

Part (c)

The standard minmax algorithm uses the Depth first approach and as we know, DFS is not complete - it may fall into an infinite loop. So in this case minimax algo may end up in infinite loop.

We can try to fix this issue by comparing current state against the stack; and if we see any repeated states, we will return a ? value.

This may work fine for this case, but it does not guarantee perfect working always.

For example: We are given that when A reaches 4, the value of game is +1, when B reaches 1, value of game is -1. But we have no clarification for "draw" state i.e. it is not clear how to compare the ? state with draw position. A "draw" position is neither good nor bad for either player, but it may help us to bring the game to end.

Part (d) For $n > 2$

let us assume $\underline{n=3}$ ie $\underline{\text{odd}}$

Initial state

1	2	3
A		B

According to game rule, A has to move first and will ~~jup~~ jump to 2.

1	2	3
	A	B

In the next turn, B makes a moves and jumps over A to reach to position 1 because there are no adjacent squares empty for B and thus only option for B is to jump over A to move to 1 which is the goal state for B and A looses.

1	2	3
B	A	

Here n is odd, and A looses.
But we need to consider more values of n to make sure we land up to correct conclusion

Assume $\underline{n=4}$ ie $\underline{\text{even}}$

Initial State

1	2	3	4
A	·		B

According to the game rule, A moves first and goes to square 2.

1	2	3	4
	A		B

Now B's turn and B comes to sq 3.

1	2	3	4
	A	B	

Now again A's turn and he gets two options to move to sq 1 or sq 4. But A knows that going to sq 1 will take him away from goal node, so player A moves to sq 4 and reaches the goal node.

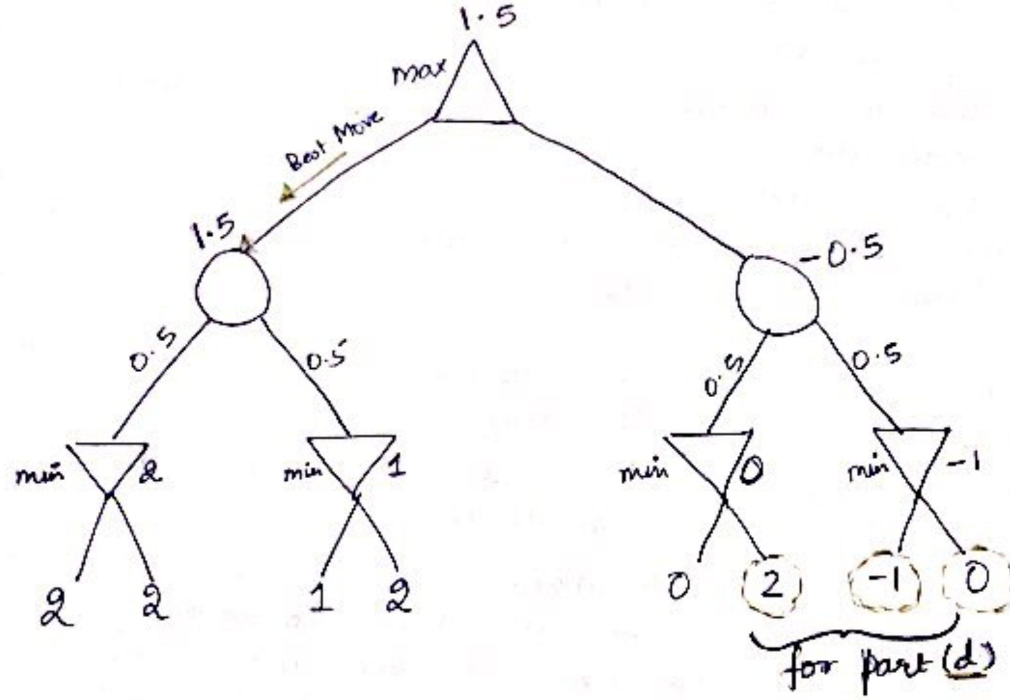
1	2	3	4
		B	A

Q-1 3

Therefore we can say that if n is even, player A will have a priority to reach to goal node, but if n is odd, then player B will have a priority in reaching to goal node, until and unless any of the player moves back or blocks path or draws the game. Thus we can say that whener n is even, player A wins and when n is odd, player A looses.

Ques-2 Pruning

Part (a)



Starting from Bottom Left

min (2, 2) = 2

min (1, 2) = 1

Value of internal node (L) = 1.5

Bottom Right

min (0, 2) = 0

min (-1, 0) = -1

value of internal node (R) = -0.5

At top, max(1.5, -0.5) = 1.5

Part (b)

Given values of first six leaves, do we need to evaluate the 7^{th} and 8^{th} leaves

→ Yes, we need to evaluate.

Value of left branch = 1.5

We need to evaluate because if the 7^{th} and 8^{th} ~~...~~ leaf are both greater than 2, then the best move will change to right side.

(If they have value > 2, the value of min node and chance node will change too, thus changing best move)

Given the values of first seven leaves, do we need to evaluate the eighth leaf?

→ once we evaluate the 7^{th} leaf, we know that Value of right branch is at most −0.5. The min node cannot be worth more than −1, so our chance node on right side can no more be ~~...~~ greater than −0.5 and in that case the best move will not change. thus we do not need to evaluate the 8^{th} leaf.

Part (c)

The chance of the node will be between 0 to 2 → ~~...~~ a case where third and fourth leaves is −2, so we get the value of chance node as 0.

Part(d)

Marked on the diagram with dotted circles

Ques. 3

(a) α is valid if and only if True $\models \alpha$

$\frac{\alpha}{True}$ // to prove

Valid: A sentence is valid if it is True for all models.

So to prove α to be valid, we need to assert that α is true in all worlds.

given True $\models \alpha$.

True entails α if and only if for every model in which True is true, α is true (by definition).

True	α
true	true

(b) For any α, False $\models \alpha$

given false $\models \alpha$.

False entails α if and only if for all models in which False is true, α is true as per definition of entailment.

Thus, the statement is Impassively true, since there are no models in which False is true.

(c) $\alpha \models \beta$ if and only if the sentence $(\alpha \to \beta)$ is valid

If $\alpha \to \beta$ is valid, then if α is true, β is true. And we have $\alpha \models \beta$, and if α is false, then β is ~~false~~ true for all the models where α is false because false cannot be true. Thus, if $\alpha \to \beta$ is valid, then $\alpha \models \beta$.

$\alpha \models \beta$

α	β
T	T
F	T
F	F

$\alpha \to \beta$

α	β	$\alpha \to \beta$	
T	T	T	✓ valid instan
T	F	F	
F	T	T	✓ valid instan
F	F	T	✓ Valid insta

(d) $\alpha \equiv \beta$ if and only if the sentence $\alpha \Leftrightarrow \beta$ is valid

We know $P \Leftrightarrow Q = (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

Similarly $\alpha \Leftrightarrow \beta = (\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$.

$\alpha \Rightarrow \beta$: when α is true, β is true for all the models where α holds true. Hence, α entails β

$\beta \Rightarrow \alpha$: when beta(β) implies ~~α~~ α, it means that wherever β is true, α is true for all world/models where β is true.
Hence β entails α.

Combining both the results, we see that the two statements are logically equivalent whenever they entail each other. Thus we ~~say~~ can say $\alpha \equiv \beta$ when $\alpha \Leftrightarrow \beta$ is valid.

(e) $\alpha \models \beta$ if and only if the sentence $(\alpha \wedge \neg\beta)$ is unsatisfiable

Unsatisfiable : A sentence is unsatisfiable if it is true in no models which means it is always false. in other words.

α	β	$\neg\beta$	$\alpha \wedge \neg\beta$
T	T	F	False
T	F	T	True
F	T	F	False
F	F	T	False

$\alpha \models \beta$

α	β
T	T
F	T
F	F

$\alpha \wedge \neg\beta$ is unsatisfiable in cases where $\alpha \rightarrow \beta$ is valid. $\alpha \rightarrow \beta$ means that β will hold in all models where α is true. Thus, the entailment when $\alpha \wedge \neg\beta$ is unsatisfiable

Ques-4

(a) $A \longleftrightarrow (B \vee E)$

We know ~~A⇔(B∨E)~~ $P \Leftrightarrow Q = (P \Rightarrow Q) \wedge (Q \Rightarrow P)$

Similarly here,

$$\left(A \Rightarrow (B \vee E) \right) \wedge \left((B \vee E) \Rightarrow A \right)$$

We know $P \Rightarrow Q = \neg P \vee Q$

Applying here,

$$\left(\neg A \vee (B \vee E) \right) \wedge \left(\neg (B \vee E) \vee A \right)$$

Applying De Morgan's Law

$$\left(\neg A \vee (B \vee E) \right) \wedge \left((\neg B \wedge \neg E) \vee A \right)$$

Applying distributive law

$$\underbrace{(\neg A \vee B \vee E)}_{C_1} \wedge \underbrace{(\neg B \vee A)}_{C_2} \wedge \underbrace{(\neg E \vee A)}_{C_3}$$

$C1 : (\neg A \vee B \vee E)$

$C2 : (\neg B \vee A)$

$C3 : (\neg E \vee A)$

(b) $E \rightarrow D$

We know $P \rightarrow Q = \neg P \vee Q$

Here we get

$$\underbrace{\neg E \vee D}_{C_4}$$

$C_4 : \neg E \vee D$

(c) $C \wedge F \longrightarrow \neg B$

We know $P \rightarrow Q = \neg P \vee Q$

Here P is $C \wedge F$

Applying ,

$$\neg (C \wedge F) \vee (\neg B)$$

Applying De Morgans Law

$$\underbrace{(\neg C \vee \neg F \vee \neg B)}_{C_5}$$

$C_5 : (\neg C \vee \neg F \vee \neg B)$

(d) $E \rightarrow B$

We know $P \rightarrow Q = \neg P \vee Q$

we get , $\underbrace{\neg E \vee B}_{C_6}$

$C_6 : \neg E \vee B$

(e) $B \rightarrow F$

We know $P \rightarrow Q = \neg P \vee Q$

We get,

$$\underbrace{\neg B \vee F}_{C_7}$$

$C_7 : \neg B \vee F.$

(f) $B \rightarrow C$

We know $P \rightarrow Q = \neg P \vee Q$

We get, $\underbrace{\neg B \vee C}_{C_8}$

$C_8 : \neg B \vee C$

Goal $\Rightarrow \neg A \wedge \neg B$

$\neg$Goal $\Rightarrow \neg(\neg A \wedge \neg B)$

$A \vee B$ // **To prove**

All clauses

$C_1 : (\neg A \vee B \vee E)$

$C_2 : (\neg B \vee A)$

$C_3 : (\neg E \vee A)$

$C_4 : \neg E \vee D$

$C_5 : \neg C \vee \neg F \vee \neg B$

$C_6 : \neg E \vee B$

$C_7 : \neg B \vee F$

$C_8 : \neg B \vee C$

$C_9\ (\neg \text{Goal}) : A \vee B$

$C_{10}: \neg A \vee B \rightarrow$ from C_1 & C_6

$C_{11}: \neg C \vee \neg B \rightarrow$ from C_5 & C_7

$C_{12}: \neg B \rightarrow$ from C_{11} & C_8

$C_{13}: \neg A \rightarrow$ from C_{12} & C_{10}

$C_{14}: A \rightarrow$ from C_{12} & C_9

$\{\phi\} \rightarrow$ from C_{13} & C_{14}

Since we get an empty set, the proof is complete. Our assumption was wrong, so Goal $(\neg A \wedge \neg B)$ is true

---

Alternative Way

from $C_1\ (\neg A \vee B \vee E)$
$C_6\ (\neg E \vee B)$

we get $\underline{\neg A \vee B}$

Take $C_{10} \wedge C_2$

$$(A \vee \neg B) \wedge (\neg A \vee B)$$

$$(A \vee \overset{\text{True}}{\cancel{\neg A}}) \wedge (A \vee B) \wedge (\neg B \wedge \neg A) \wedge (\overset{\text{True}}{\cancel{\neg B} \vee B})$$

$$(A \vee B) \wedge (\neg B \wedge \neg A)$$

$$(A \vee B) \wedge (\neg (A \vee B))$$

If $A \vee B = K$

$$K \wedge \neg K$$

$$= \text{FALSE}$$

Thus our assumption was wrong. This means $\neg A \vee \neg B$ is proved using resolution.