1 Q1

(a) In the game shown in Figure 1, there are exactly two subgames. One starts from player 2's node, and the other one is the whole game graph.

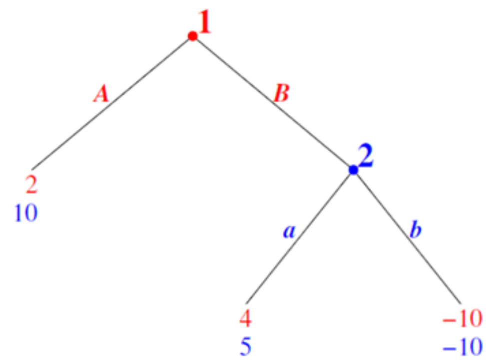

Figure 1: Example for Q1(a).

Using the 'Backward Induction' method, we can easily see that in subgame 1, player 2's must always choose strategy a to maximize its payoff in this subgame, i.e., the payoff of choosing a is 5 which is far greater than choosing b with payoff -10.

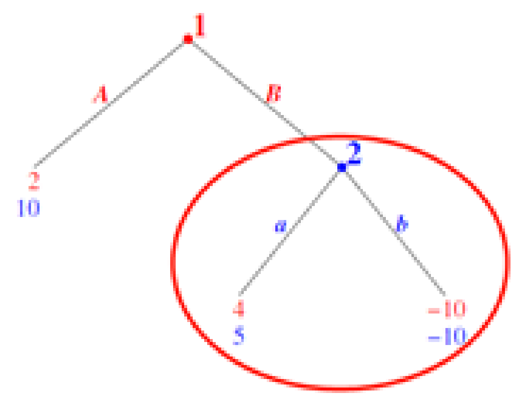

Figure 2: Subgame 1 of game G.

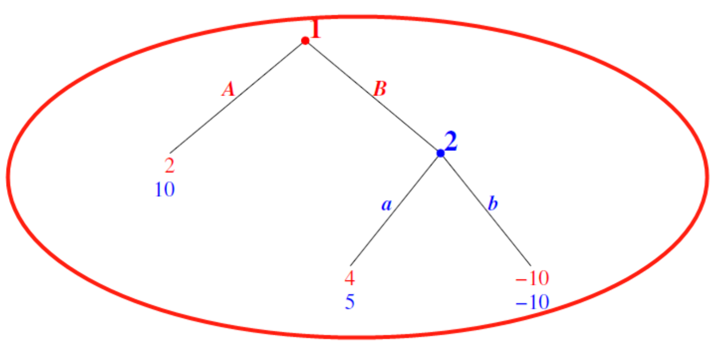

Figure 3: Subgame 2 of game G.

There are two subgames in game G, as shown in Figure 2 and Figure 3. Then, following the 'Backward Induction', we substitute player 2's node to a leaf node with payoff (4, 5). Then, we can find the SPNE of the game, that is player 1 plays B and player 2 plays a, i.e., (B, a) .

player 1/player 2	a	b
A	(2,10)	(2,10)
B	(4,5)	(-10,-10)

Table 1: The NEs in game G.

When we look back to the NE of this game, we can see that there are two NE in this game (A,b) and (B, a). Under both of these situations, neither player 1 nor player 2 can improve their payoff by unilaterally changing their strategy.

The first pure NE is (A,b). This is where player 1 plays A and player 2 plays b, we can see this as a pure Nash equilibrium since neither player can deviate from their strategy to increase their payoff.

The second pure NE is (B, a). This is where player 1 plays B and player 2 plays a, we can see this as a pure Nash equilibrium since neither player can deviate from their strategy to increase their payoff.

As a result, we can find out that (A, b) is NE but it is not SPNE, while (B, a) is NE and it is also SPNE.

(b) We could prove this by 'Backward Induction'.

First, we need to make some assumptions. For a game G with game tree T, and for $w \in T$, define a subtree $T_w \subseteq T$, by $T_w = \{w' = ww''\ for\ w'' \in \Sigma^*\}$. Since the tree is finite, we can just associate payoff with leaf nodes. The depth of node w in T is its length $|w|$ as a string. The depth of tree T is the maximum depth of any node in T. The depth of a game G is the depth of its game tree. And there is a pure profile $s^* = (s_1^*, ..., s_n^*)$ is a SPNE.

To prove this statement, by using induction, we will show that every subgame G_w has a unique pure strategy SPNE. Because game G with perfect information, so the whole game can be identified as a subgame. Thus, until the final subgame (the whole game tree) is shown to have a unique pure strategy SPNE, the statement will be proved.

Depth 0: At leaf node w, each player i gets its payoff $u_i(w)$, and the strategies in the SPNE s* are empty.

Depth 1: Let's prove the case when depth=1, there must be a unique SPNE. As assumptions show in the statement, all the payoff at each leaf node is unique and there are no chance nodes in the game, therefore, there must be a leaf node at every subgame with depth=1 that has a strict better payoff than others and the subgame is controlled by only one player's strategy. As a result, the player to whom the node belongs, cannot change other strategies to maximize its payoff. In fact, there must be a unique edge or action that maximizes the payoff for the player who owns the node.

Inductive step: We assume the whole game tree's maximum depth is k+1. Let $Act(W) = a_1', ..., a_r'$ be the set of be the set of actions available at the root of game G_w. The subtrees $T_{wa_j'}$, for $j = 1, .., r$, each define a PI-subgame $G_{wa_j'}$, of depth $<= k$. Thus, by induction, each game $G_{wa_j'}$ has a pure strategy SPNE, $s^{wa_j'} = (s_1^{wa_j'}, ..., s_n^{wa_j'})$.

Next, we need to prove that this also works for game G_w when depth=k+1. For the root node $\epsilon \in PI_i$, which belongs to player i, where PI_i is the perfect information set for player i. Let $a' = argmax_{a \in Act(\epsilon)} h_i^{\epsilon a}(s^{\epsilon a}$ be the action that player i would take to get maximum payoff for itself, where $h_i^{\epsilon a}(s^{\epsilon a})$ is the expected payoff to player i in the subgame $G_{\epsilon a}$. As shown in Depth 0 and Depth 1, we know that each such node could be replaced by a leaf node with its maximum payoff based on the subgame's NE. Because all the nodes' payoffs are unique, so there must be unique action taken at each node, otherwise, the payoff wouldn't be the best one.

That is to say, each subgame with depth=k can be replaced by a leaf node that contains its subgame's expected payoff. And all the replaced node's payoffs are unique, therefore, the game with depth k+1 must have a unique choice among all the replaced nodes that maximize the root node's player i's payoff.

Therefore, we can define s_i^* be the union of player i's pure strategies in each subgame based on the subgame's NE. Composing all player's pure strategies together, we can get a unique SPNE $s^* = (s_1^*, ..., s_n^*)$ for game G. By using induction of showing the statement is true in Depth=0 and Depth=1, given the assumption in Depth=k, we successfully make our statement holds in Depth=k+1, thus, the statement 'every finite extensive game of perfect information where there are no chance nodes and where no player gets the same payoff at any two distinct leaves, must have a unique pure-strategy SPNE' is true.

(c) As shown in Figure.4, we can see the example for this sub-question. If we consider the normal form of this game, we will get a payoff matrix, shown in the following Table 3.

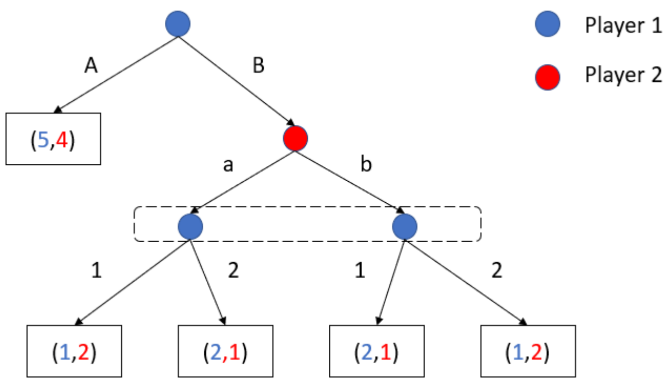

Figure 4: Example for Q1(c).

player 1/player 2	a	b
A1	**(5,4)**	**(5,4)**
A2	**(5,4)**	**(5,4)**
B1	(1,2)	(2,1)
B2	(2,1)	(1,2)

Table 2: The payoff matrix in game G.

We can see that we have four pure NEs in this game, namely $\{(A1, a), (A1, b), (A2, a), (A2, b)\}$, and no player can improve their payoff by unilaterally changing their strategy.

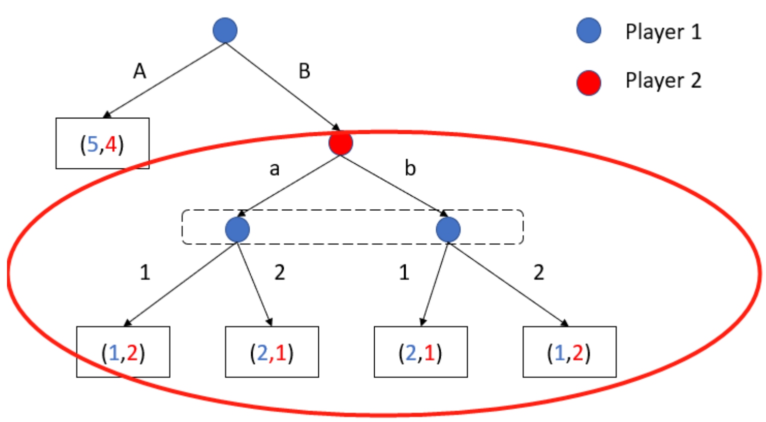

Figure 5: One subgame of game G..

For SPNE, we can use 'Backward Induction' again. One subgame is shown in Figure.5. Below is the payoff matrix of this subgame:

player 1/player 2	a	b
1	(1,2)	(2,1)
2	(2,1)	(1,2)

Table 3: The payoff matrix in one subgame of game G.

However, this subgame has no pure NE, since each pure strategy profile's payoff can be improved by one of the players unilaterally changing its strategy.

Because this subgame has no pure NE, the whole game wouldn't have a pure SPNE that covers all the subgames in the game. Thus, this example is the game that contains a pure NE but does not contain any subgame perfect pure Nash Equilibrium.

2 Q2

(a) The answer is yes.

An imperfect information EFG has perfect recall if each player i never "forgets" its sequence of prior actions and information sets, i.e., an EFG has perfect recall if whenever $w, w' \in Info_{i,j}$ belong to the same information set, then the "visible history" for player i (sequence of information sets and actions of player i during the play) before hitting node w and w' must be the same.

We can easily see from the game that it satisfies perfect recall. For each player i at node w after action took in the previous node x, all the nodes at the same information set which contains w also have the same "visible history". Those nodes that are in the information set with only one node do satisfy this requirement because the "visible histories" of one node and itself are the same. For the nodes of player 2 in the only information set with more than one node, we can see that the "visible history" of both nodes are $\{\o\}$, so the nodes in this information set also satisfy the requirement.

After checking all nodes in the game tree, we can say that this game satisfies "perfect recall".

(b) The trivial subgames are shown in Figure 6. Apart from the game itself, there are 5 trivial subgames. Because the nodes for player 2 after player 1 plays M or R belong to the same information set, we cannot get the subgame rooted at these two nodes. Thus, the whole game contains 6 subgames. To find all SPNEs in this game, we plan to use the 'Backward Induction' algorithm.

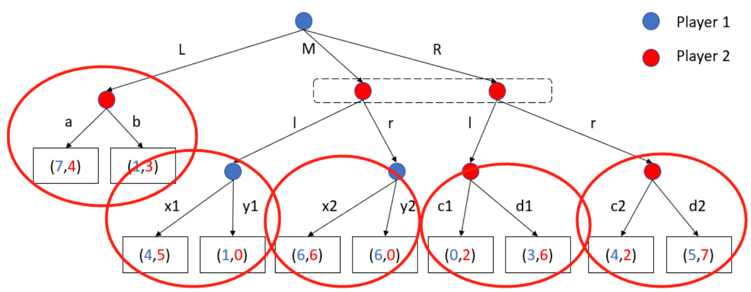

Figure 6: The trivial subgames in this extensive form game are highlighted in red circles.

First, we need to find all NEs in the subgames. We simply name the trivial subgames from left to right to subgame1, subgame2,..., subgame5.

- Subgame1: In this subgame, player 2 has two actions: a and b. Player 2's aim is to choose one action that maximizes its payoff in this subgame. Choosing action a gets payoff 4, while choosing action b gets payoff 3. Because $3 < 4$, player 2 will always choose to play action a with expected payoff (7,4).
- Subgame2: In this subgame, player 1 has two actions: x_1 and y_1. Player 1's aim is to choose one action that maximizes its payoff in this subgame. Choosing action x_1 gets payoff 4, while choosing action y_1 gets payoff 1. Because $4 > 1$, player 1 will always choose to play action x_1 with expected payoff (4,5).
- Subgame3: In this subgame, player 1 has two actions: x_2 and y_2. Player 1's aim is choosing one action that maximizes its payoff in this subgame. Choosing action x_2 gets payoff 6, while choosing action y_2 gets payoff 6. Because $6 = 6$, so player 1 can either choose action x_2 or y_2 with expected payoffs (6,6) or (6,0) respectively.
- Subgame4: In this subgame, player 2 has two actions: c_1 and d_1. Player 2's aim is choosing one action that maximizes its payoff in this subgame. Choosing action c_1 gets payoff 2, while choosing action d_1 gets payoff 6. Because $2 < 6$, so player 2 will always chooses to play action d_1 with expected payoff (3,6).
- Subgame5: In this subgame, player 2 has two actions: c_2 and d_2. Player 2's aim is choosing one action that maximizes its payoff in this subgame. Choosing action c_2 gets payoff 2, while choosing action d_2 gets payoff 7. Because $2 < 7$, so player 2 will always chooses to play action d_2 with expected payoff (5,7).

After getting all NEs in these 5 trivial subgames, we can replace the subtrees of these subgames with leaf nodes that contain the expected payoffs. Since there are two possible actions in subgame3, we then have two versions of the new extensive form game, which are shown in Figure 7 and Figure 8.

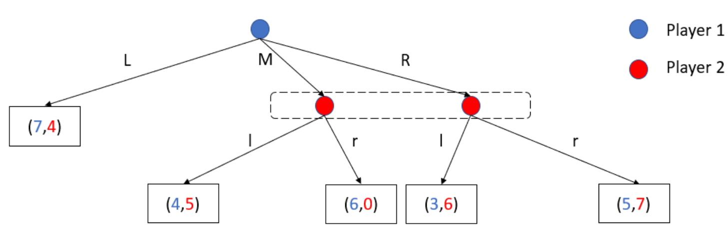

Figure 7: The reduced game G' when player 1 in subgame3 choose y_2.

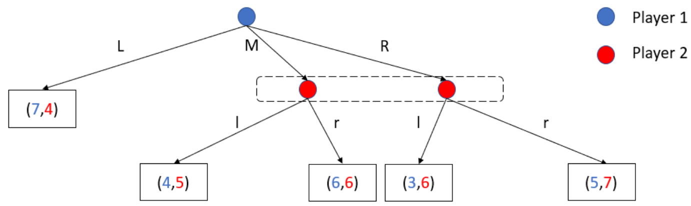

Figure 8: The reduced game G” when player 1 in subgame3 choose x_2.

In both two new games G’ and G”, the only subgame is the game itself, which means that the NE of this new game is also the SPNE of the game. We plan to analyze these two situation one by one.

For situation shows in Figure 7, the expected payoff matrix can be described as Table 4. We can easily find the pure NE in this game G’, namely {(L,l), (L,r)}, since each player cannot increase their payoffs by unilaterally changing their strategies.

player 1/player 2	l	r
L	**(7,4)**	**(7,4)**
M	(4,5)	(6,0)
R	(3,6)	(5,7)

Table 4: The payoff matrix in reduced game G’.

For situation shows in Figure 8, the expected payoff matrix can be described as Table 5. We can easily find the pure NE in this game G’, namely {(L,l), (L,r)}, since each player cannot increase their payoffs by unilaterally changing their strategies.

player 1/player 2	l	r
L	**(7,4)**	**(7,4)**
M	(4,5)	(6,6)
R	(3,6)	(5,7)

Table 5: The payoff matrix in reduced game G”.

The 4 pure SPNEs we have found so far are {(Lx_1x_2, ald_1d_2), (Lx_1x_2, ard_1d_2),(Lx_1y_2, ald_1d_2), (Lx_1y_2, ard_1d_2)}

Next, we need to be concerned about whether there are mixed SPNEs in the game. For subgame1. subgame2, subgame4, and subgame5, it is clear that there are no mixed NEs because there is only one player’s choice and the payoff of each choice is unique. But for subgame3, the leaf nodes’ payoffs are both 6 with player 1. We define a $p \in [0, 1]$ that represents the probability of player 1 choosing action x_2, so the probability of player 1 choosing action y_2 is $(1 - p)$. Based on the ’Backward Induction’ algorithm and useful corollary for Nash Equilibria, the expected payoff of the leaf node that replaces the subgame3 would be (6,6p). And the new updated game G”’ is shown in Figure 9. The corresponding payoff matrix in new game G”’ is shown in Table 6.

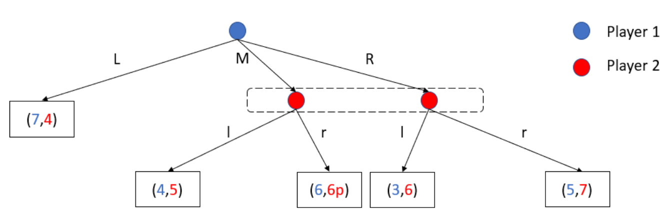

Figure 9: The reduced game G”’ when player 1 in subgame3 choose x_2 with p and y_2 with $(1 - p)$.

player 1/player 2	l	r
L	**(7,4)**	**(7,4)**
M	(4,5)	(6,6p)
R	(3,6)	(5,7)

Table 6: The payoff matrix in reduced game G"'.

We can see that in this payoff matrix, the small changes in subgame3's best payoff don't make any difference in the new game G"'. Player 1's pure strategy L strictly dominates other pure strategies. Based on the assumption that player 1 will always choose L, both l and r are player 2's best response to it. We can use the corollary of the Nash Equilibrium theorem, i.e., if player 2 is playing against player 1's mixed strategy, both of player 2's pure strategies must be the best response to player 1. We define $q \in [0, 1]$ as the probability of player 2 choosing l, and the probability of player 2 choosing r is $(1 - q)$. The mixed NE of game G"' is $\{(\text{L}x_1(x_2^p,y_2^{(1-p)}),\ \text{a}(l^q,r^{(1-q)})d_1d_2)\}$, where $(x_2^p,y_2^{(1-p)})$ means the mixed strategy that player 1 has probability p to choose strategy x_2 and probability $(1 - p)$ to choose strategy y_2, the same to $(l^q,r^{(1-q)})$. If we set the range of p and q to $[0, 1]$, our expression will also contain the pure SPNEs.

As a result, all SPNEs are given by:

- (Lx_1x_2, ald_1d_2), in behavior strategy: Player 1: ((1, 0, 0), (1, 0), (1, 0)); Player 2: ((1,0), (1,0), (0,1), (0,1))
- (Lx_1x_2, ard_1d_2), in behavior strategy: Player 1: ((1, 0, 0), (1, 0), (1, 0)); Player 2: ((1,0), (0,1), (0,1), (0,1))
- ((Lx_1y_2), (ald_1d_2), in behavior strategy: Player 1: ((1, 0, 0), (1, 0), (0, 1)); Player 2: ((1,0), (1,0), (0,1), (0,1))
- (Lx_1y_2, ard_1d_2), in behavior strategy: Player 1: ((1, 0, 0), (1, 0), (0, 1)); Player 2: ((1,0), (0,1), (0,1), (0,1))
- (L$x_1(x_2^p,y_2^{(1-p)})$, a$(l^q,r^{(1-q)})d_1d_2$), in behavior strategy: Player 1: ((1, 0, 0), (1, 0), (p, 1-p)); Player 2: ((1,0), (q,1-q), (0,1), (0,1)), where $p \in [0, 1]$ and $q \in [0, 1]$.

Why these are all SPNEs in this game?

Following the 'Backward Induction' algorithm, we can get the NEs in those trivial subgames, four of which are fixed. Only subgame3 has two possible actions to choose from. Then, the reduced game doesn't have trivial subgames, thus the NEs in the reduced game is also fixed. A profile of behavior strategies can be named SPNE if defines a Nash equilibrium for every subgame in this game. And the subgames in this game contain those 5 trivial subgames and the whole game (or the reduced game). By composing the NEs of those subgames, we can easily prove that SPNEs we calculated are all SPNEs in this game.

(c) The answer is yes.

As shown in Figure 10, we can see that the pure SPNEs we find at the Q2(b) are in purple and highlighted by the blue box. But the NEs of this game are highlighted in the red box, especially, those profiles are NEs but no SPNEs are in red, any of which could be shown as examples to prove that this game contains NEs, other than the SPNEs identified above.

player1/player2	a;l;c1;c2	a;l;c1;d2	a;l;d1;c2	a;l;d1;d2	a;r;c1;c2	a;r;c1;d2	a;r;d1;c2	a;r;d1;d2	b;l;c1;c2	b;l;c1;d2	b;l;d1;c2	b;l;d1;d2	b;r;c1;c2	b;r;c1;d2	b;r;d1;c2	b;r;d1;d2
L;x1;x2	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)
L;x1;y2	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)
L;y1;x2	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)
L;y1;y2	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(7,4)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)	(1,3)
M;x1;x2	(4,5)	(4,5)	(4,5)	(4,5)	(6,6)	(6,6)	(6,6)	(6,6)	(4,5)	(4,5)	(4,5)	(4,5)	(6,6)	(6,6)	(6,6)	(6,6)
M;x1;y2	(4,5)	(4,5)	(4,5)	(4,5)	(6,0)	(6,0)	(6,0)	(6,0)	(4,5)	(4,5)	(4,5)	(4,5)	(6,0)	(6,0)	(6,0)	(6,0)
M;y1;x2	(1,0)	(1,0)	(1,0)	(1,0)	(6,6)	(6,6)	(6,6)	(6,6)	(1,0)	(1,0)	(1,0)	(1,0)	(6,6)	(6,6)	(6,6)	(6,6)
M;y1;y2	(1,0)	(1,0)	(1,0)	(1,0)	(6,0)	(6,0)	(6,0)	(6,0)	(1,0)	(1,0)	(1,0)	(1,0)	(6,0)	(6,0)	(6,0)	(6,0)
R;x1;x2	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)
R;x1;y2	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)
R;y1;x2	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)
R;y1;y2	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)	(0,2)	(0,2)	(3,6)	(3,6)	(4,2)	(5,7)	(4,2)	(5,7)

Figure 10: The payoff matrix of whole extensive form game.

For example, in the profile (Mx_1x_2, brd_1d_2), player 1 cannot improve its current payoff 6 by unilaterally changing its strategy, neither player 2 can improve its current payoff 6 by unilaterally changing its strategy.

(d) The answer is no.

As we illustrated above, there are 4 pure SPNEs in this game. We take SPNE (Lx_1x_2, ard_1d_2) and NE (Mx_1x_2, brd_1d_2) as examples. Because NE (Mx_1x_2, brd_1d_2), which is not SPNE, player 2 could make a non-credible threat at node we highlighted in Figure 11 to choose b. Player 2 threat player 1 that if player 1 choose L then player 2 will choose b to make player get payoff 1 instead of the maximum payoff 7. If player 1 believe this threat, then player 1 would choose M instead to get payoff 6, which at least is better than 1 by taking action L. Player 2 has the possibility to increase its own payoff from 4 to 6 by threatening if player 1 also takes action x2, so player 2 has a reasonable incentive to make non-credible threats.

Thus, this game exists some NEs that have a non-credible threat and the statement that all the equilibria in this game are "credible" is wrong.

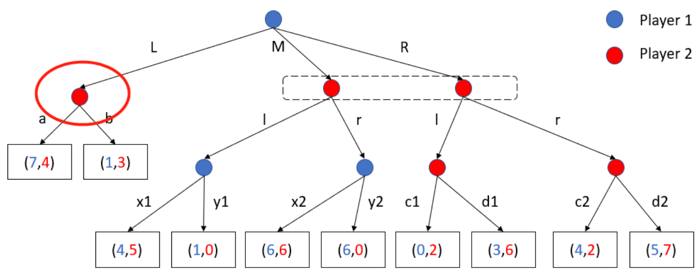

Figure 11: The example of Q2(d).