
Game Theory (CS4187)

Lectures 19 and 20

Date: 01/10/2024
03/10/2024

Instructor: Gourav Saha

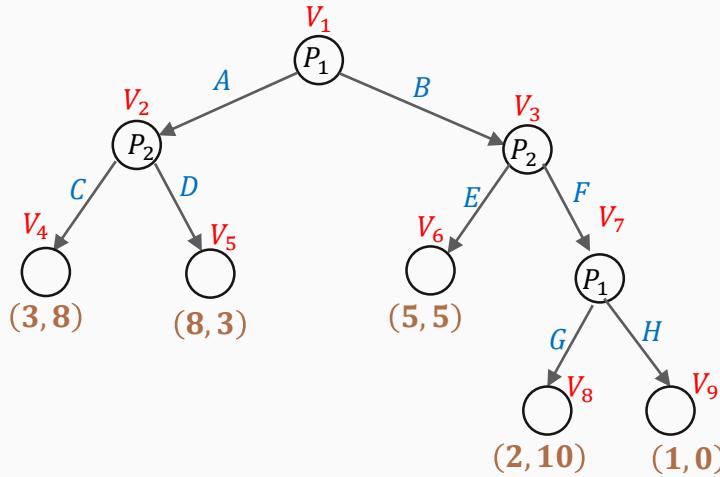
Broad Idea of Today's Lecture

- Till now in module 2, we have discussed six solution concepts related to **strategic form game** (simultaneous move games).

The topics of this lecture is to extend few of the solution concepts of strategic form games to compute solutions for **extended form games** (sequential move games).

Strategies for PIEFG: Examples

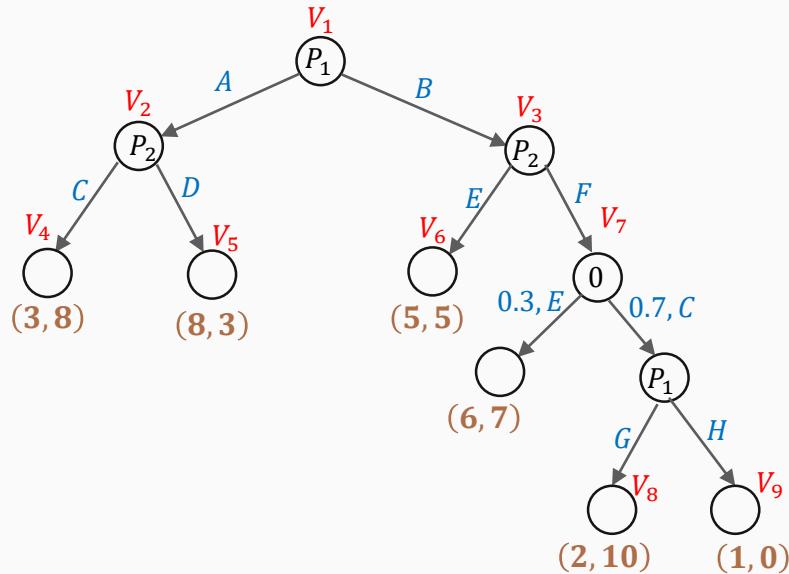
Example 1



- Recall that EFGs are of two type: **perfect information** (PIEFG) and **imperfect information** (IIEFG).
 - Both these types can be further sub-divided into games with **chances moves** and **without chance moves**.
- We start by discussing solution concepts of PIEFG, both with and without chance moves.
- To illustrate the concepts, we will use two examples shown in the left (for example 2, go to the next slide).

Strategies for PIEFG: Examples

Example 2



- Recall that EFGs are of two type: **perfect information** (PIEFG) and **imperfect information** (IIEFG).
 - Both these types can be further sub-divided into games with **chances moves** and **without chance moves**.
- We start by discussing solution concepts of PIEFG, both with and without chance moves.
- To illustrate the concepts, we will use two examples shown in the left (for example 2, go to the next slide).

Strategies for PIEFG

- Recall the definition of PIEFG with and without chance moves.

Definition (PIEFG without Chance Moves): A PIEFG without chance moves is a tuple $\Gamma = \langle N, A, H, Z, \rho, \mathcal{X}, \sigma, u \rangle$.

Definition (PIEFG with Chance Moves): A PIEFG without chance moves is a tuple $\Gamma = \langle N, A, H, Z, \rho, \mathcal{X}, \sigma, u, (p_i)_{i \in H_0} \rangle$.

- N is the set of players,
- A is the **set of actions**,
- H is the **set of non-terminal vertices**,
- Z is the **set of terminal vertices**,
- $\rho : H \rightarrow N$ is the **player function**,
- $\mathcal{X} : H \rightarrow 2^A$ is the **action function**,
- $\sigma : H \times A \rightarrow H \cup Z$ is the **successor function**,
- $u = (u_n)_{n \in N \setminus \{0\}}$, where $u_n : Z \rightarrow \mathbb{R}$ is the **utility function**,
- $H_0 = \{i \in H : \rho(i) = 0\}$ is the **set of non-terminal vertices for which the player is nature**. For every $i \in H_0$, p_i is the **probability distribution** across actions in set $\mathcal{X}(i)$.

Strategies for PIEFG

- A **pure strategy** for a player in PIEFG is a **complete specification** of which (deterministic) action to take at every vertices belonging to that player.
- Let,

$$H_i = \{v \in H : \rho(v) = i\}$$

is the set of non-terminal vertices where player i is playing. Let $a_{i,j}$ denote the action that player i takes corresponding to $H_{i,j}$ ($H_{i,j}$ the j^{th} vertex in H_i). Then the **pure strategy of player i** is,

$$s_i = (a_{i,1}, a_{i,2}, \dots, a_{i,|H_i|})$$

- The **set of strategies of player i** is the cartesian product of all $\mathcal{X}(H_{i,j})$,

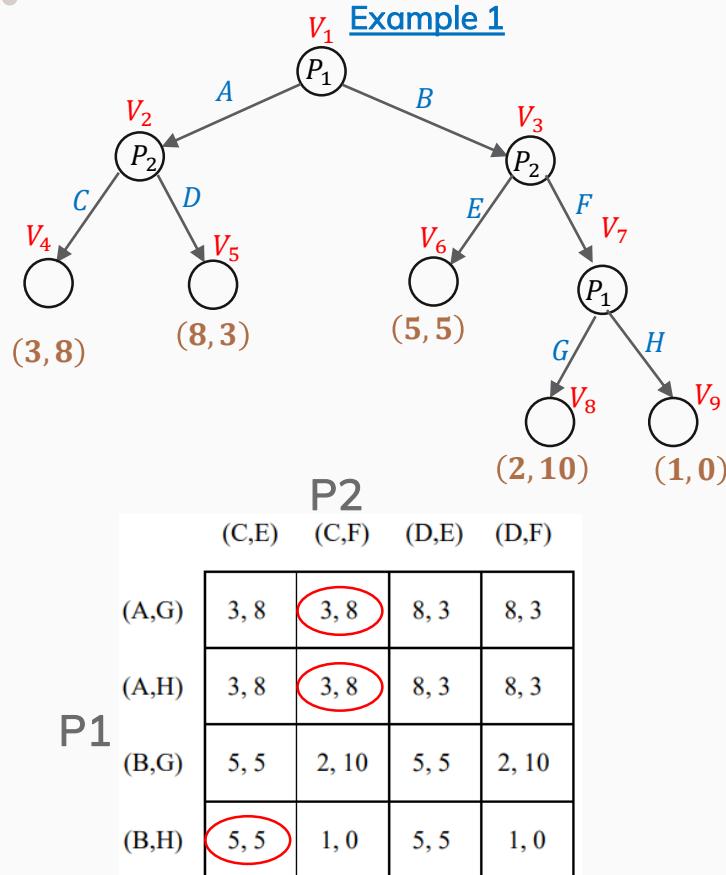
$$S_i = \mathcal{X}(H_{i,1}) \times \mathcal{X}(H_{i,2}) \times \dots \times \mathcal{X}(H_{i,|H_i|})$$

We have $s_i \in S_i$.

Strategies for PIEFG

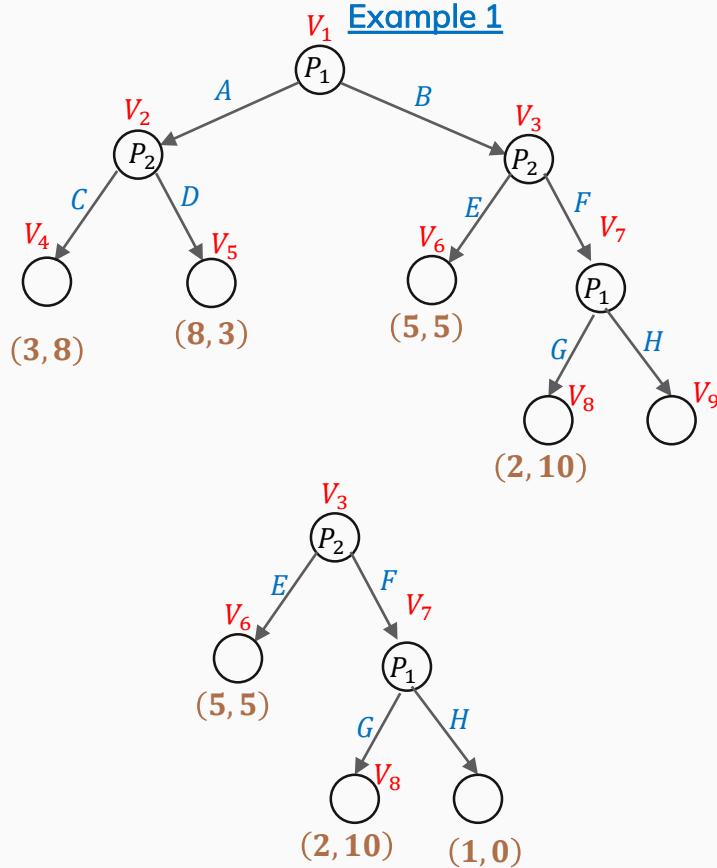
- Just like SFGs, in EFG too there is a concept of **strategy profile**. Strategy profile $s = (s_i, s_{-i})$ is the action of all the players.
 - In EFG, each s_i is a **tuple**. This is one way in which strategy profile of SFGs and EFGs differ.
 - Even in SFGs s_i can be a tuple.
- As we know, we can convert a EFG to and SFG. Let Γ be the EFG. Let $SFG(\Gamma)$ denote the SFG equivalent of Γ .
- Then all the solution concepts that we have learned about SFGs till now can be directly applied to $SFG(\Gamma)$ to find the solution of EFG Γ .
 - The problem is that these solutions does not utilizes the sequential nature of the game and is hence may lead to illogical solutions.

PSNE of PIEFG



- $SFG(\Gamma)$ for this game is shown in the table (directly adapted from Chapter 5 of the book Multiagent Systems by Shoham and Brown).
 - Remember, PSNE in context of EFG is related to $SFG(\Gamma)$, i.e. PSNE is related to the SFG equivalent of EFG.
 - PSNE of the game shown in the left is marked with the red circles. PSNEs are:
 - $((A, G), (C, F))$
 - $((A, H), (C, F))$
 - $((B, H), (C, F))$
 - PSNE $((B, H), (C, E))$ is not logical! Why?

Subgame



Definition (Subgame): Given a EFG Γ , the subgame of Γ at non-terminal vertex h , denoted by $\Gamma(h)$, is the EFG described by the subtree rooted at vertex h .

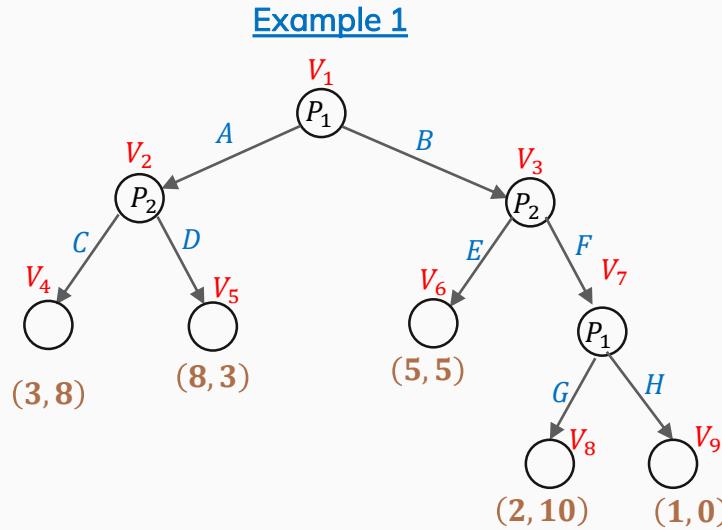
➤ Example: $\Gamma(V_3)$ shown in the left.

Subgame Perfect Nash Equilibrium (SPNE)

Definition (Pure Strategy SPNE): Pure strategy SPNE of an EFG Γ are all strategy profiles s such that for all $h \in H$, the restriction of s to $\Gamma(h)$ is a PSNE of $SFG(\Gamma(h))$.

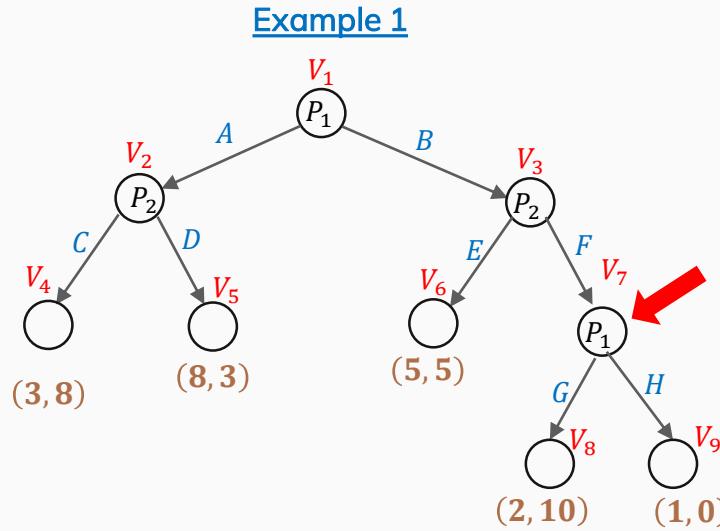
- Basically s should be a PSNE for all the subgames, then only s SPNE. **What is the logic?**
 - In order to capture the sequentially nature of the game, i.e. strategy s has to be rational after a sequence of actions was taken by other players.
- Most of the time, rather than saying “PSNE of $SFG(\Gamma(h))$ ”, we will directly say “PSNE of $\Gamma(h)$ ”.
- What does “restriction of s ”?

Backward Induction



- SPNE for PIEFG can be found using Backward induction. To learn about backward induction, look into chapter 8 of the book:
Harrington, J.E., "Games, Strategies, and Decision Making", 2nd edition.
- Compute SPNE of the PIEFG shown in the left. In the next few slides, we demonstrate backward induction.

Backward Induction



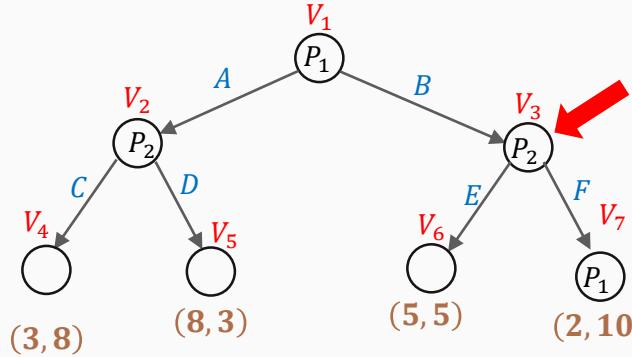
- SPNE for PIEFG can be found using Backward induction. To learn about backward induction, look into chapter 8 of the book:

Harrington, J.E., "Games, Strategies, and Decision Making", 2nd edition.

- P1 is playing in the vertex with marked with the red arrow. Action G has a payoff of 2 and H has a payoff of 1. So, P1 chooses G.

Backward Induction

Example 1

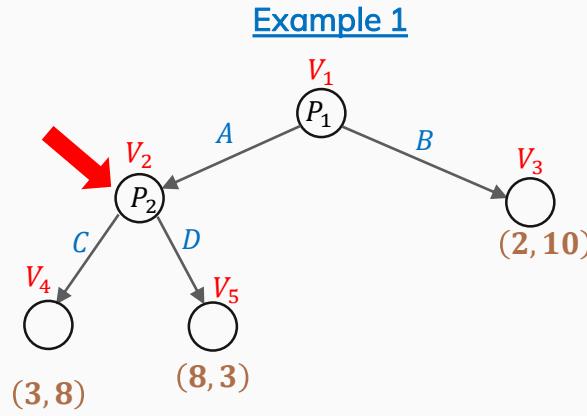


- SPNE for PIEFG can be found using Backward induction. To learn about backward induction, look into chapter 8 of the book:

Harrington, J.E., "Games, Strategies, and Decision Making", 2nd edition.

- P2 is playing in the vertex with marked with the red arrow. Action F has a payoff of 10 and E has a payoff of 5. So, P2 chooses F.

Backward Induction

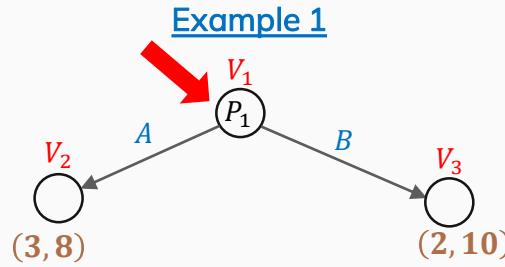


- SPNE for PIEFG can be found using Backward induction. To learn about backward induction, look into chapter 8 of the book:

Harrington, J.E., "Games, Strategies, and Decision Making", 2nd edition.

- P2 is playing in the vertex with marked with the red arrow. Action C has a payoff of 8 and C has a payoff of 3. So, P2 chooses C.

Backward Induction

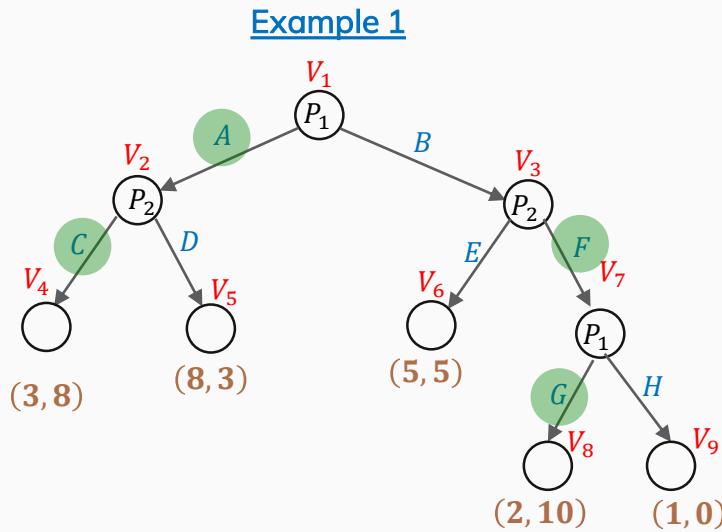


- SPNE for PIEFG can be found using Backward induction. To learn about backward induction, look into chapter 8 of the book:

Harrington, J.E., "Games, Strategies, and Decision Making", 2nd edition.

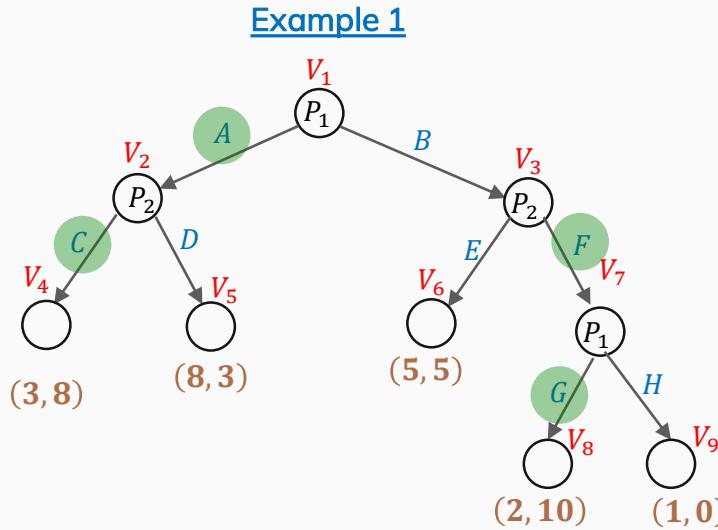
- P1 is playing in the vertex with marked with the red arrow. Action A has a payoff of 3 and B has a payoff of 2. So, P1 chooses A.

Backward Induction



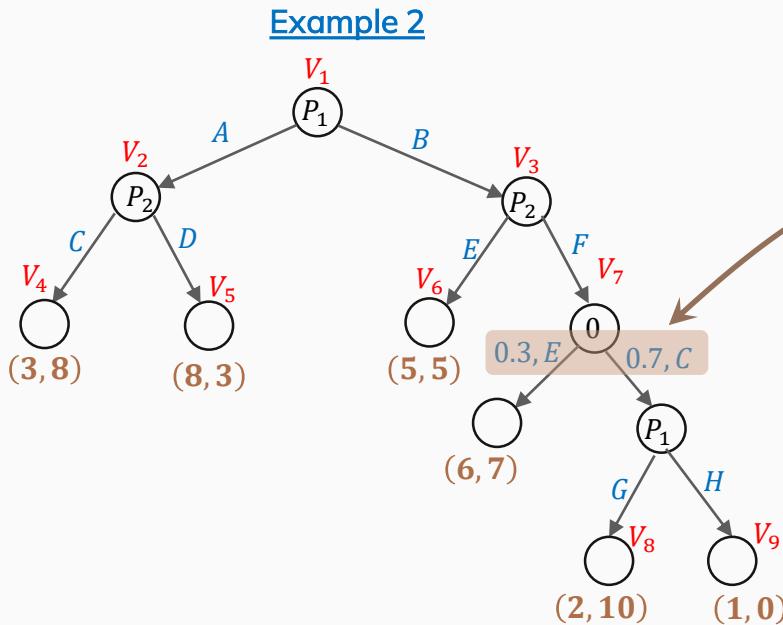
➤ So, the SPNE is,
 $((A, G), (C, F))$

Backward Induction



- So, the SPNE is,
 $((A, G), (C, F))$
- Is the computed SPNE one of the NEs that was discussed in [this slide \(click here\)](#)?
- Is it always true that an SPNE of an EFG Γ will be an NE of the induced SFG, $SFG(\Gamma)$?

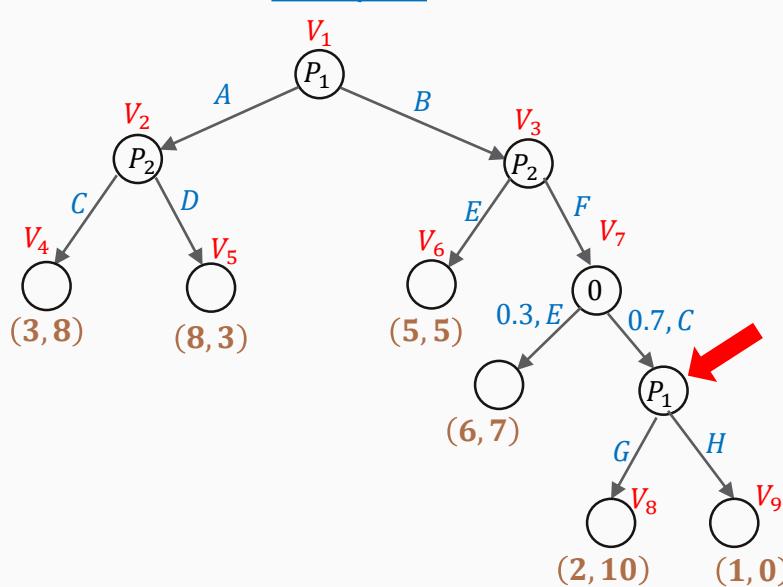
Backward Induction



➤ Compute SPNE of the PIEFG **with chance moves** shown in the left. In the next few slides, we demonstrate backward induction.

Backward Induction

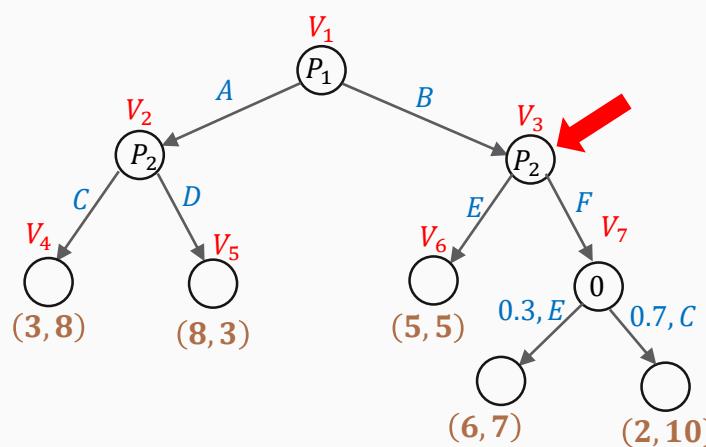
Example 2



- Compute SPNE of the PIEFG **with chance moves** shown in the left. In the next few slides, we demonstrate backward induction.
- P_1 is playing in the vertex with marked with the **red arrow**. Action G has a payoff of 2 and H has a payoff of 1. So, P_1 chooses G .

Backward Induction

Example 2



➤ Compute SPNE of the PIEFG **with chance moves** shown in the left. In the next few slides, we demonstrate backward induction.

➤ P2 is playing in the vertex with marked with the red arrow. Action E has a payoff of 5 and F has a payoff of:

$$0.3 * 7 + 0.7 * 10 = 9.1$$

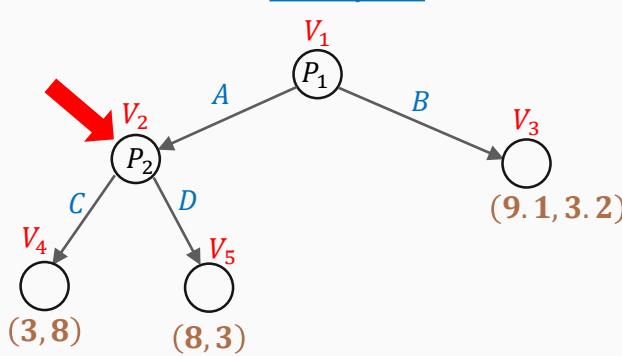
The above payoff is for P2. The payoff of P1 for action F is,

$$0.3 * 6 + 0.7 * 2 = 3.2$$

P2 chooses F because 9.1 is greater than 5.

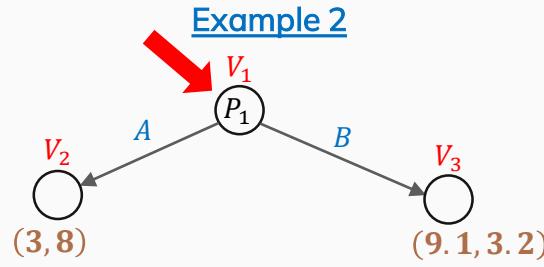
Backward Induction

Example 2



- Compute SPNE of the PIEFG **with chance moves** shown in the left. In the next few slides, we demonstrate backward induction.
- P2 is playing in the vertex with marked with the **red arrow**. Action C has a payoff of 8 and C has a payoff of 3. So, P2 chooses C.

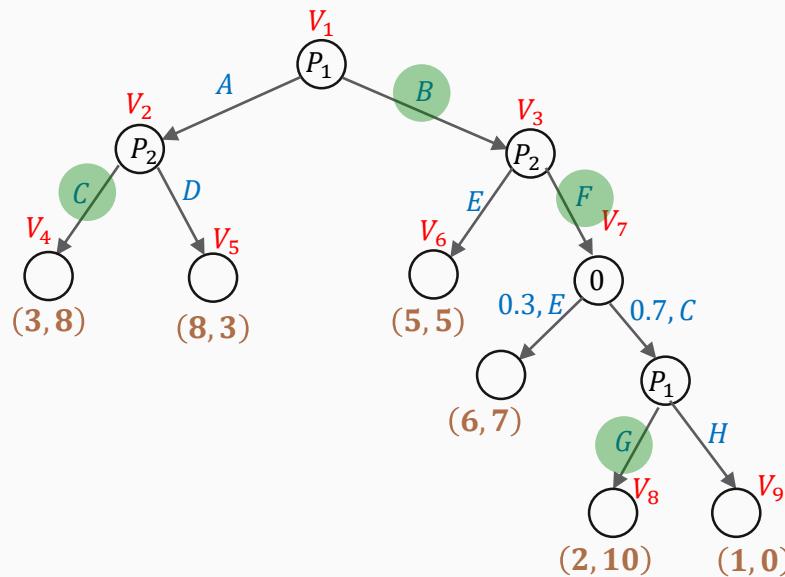
Backward Induction



- Compute SPNE of the PIEFG **with chance moves** shown in the left. In the next few slides, we demonstrate backward induction.
- P_1 is playing in the vertex with marked with the **red arrow**. Action B has a payoff of 9.1 and A has a payoff of 3 . So, P_1 chooses B .

Backward Induction

Example 2



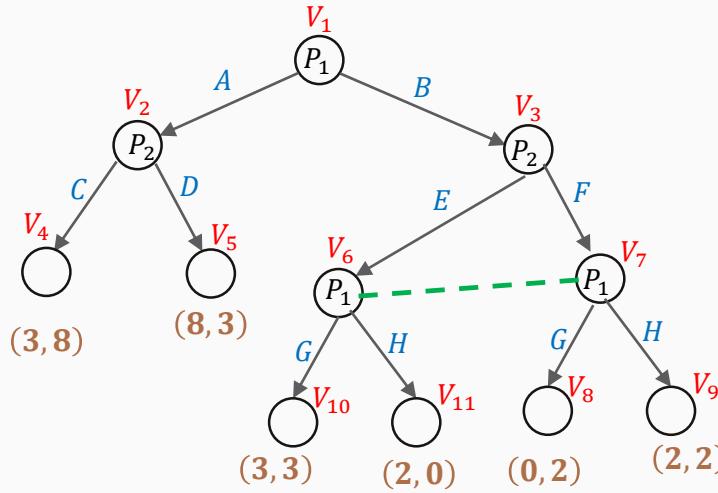
➤ So, the SPNE is,

$$((B, G), (C, F))$$

Solution Concepts for IIEFG

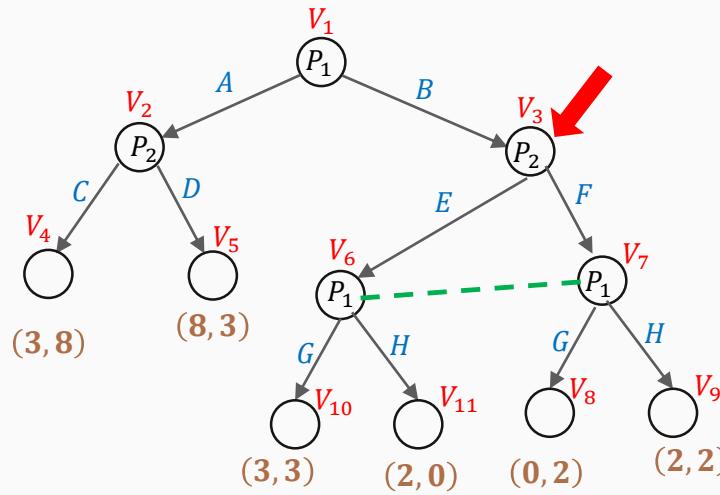
- For PIEFG, a pure strategy SPNEs/PSNEs always exists. This is a natural consequence of backward induction.
- But for IIEFG, it is not guaranteed that pure strategy SPNEs will exist. How?
 - All SFGs can be expressed as EFGs.
 - We know that there are SFGs for which PSNEs don't exist.
 - Since SPNE is a subset of PSNE, SPNE may not exist.
- We will talk about the following solution concepts of IIEFGs:
 - SPNEs.
 - Sequential equilibrium (not coming for minor 2).

SPNE for IIEFGs



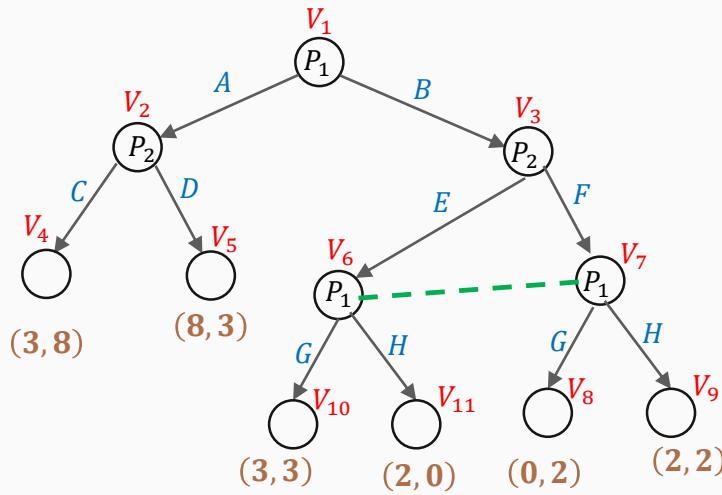
- We will demonstrate the concept using the example shown in the left.
- We can't solve imperfect information EFGs directly using backward induction. That said the overall idea of backward induction is still useful of IIEFGs.
- Backward induction can't be directly used because two nodes in the same information set can't have two different strategies as it is not possible to distinguish between the nodes in an information set. Example: P1 can take action G at vertex V6 and H at vertex V7 just because those actions gives better payoff in the corresponding vertices.

SPNE for IIEFGs



- We can see however that the subgame starting at vertex V_3 (marked with red arrow) can definitely be solved. Infact, the subgame at V_3 is a **simultaneous move game** between P1 and P2.
- Since, it is a simultaneous move game we can find its NEs as we did for SFGs. This is shown in the next slide.

SPNE for IIEFGs



		P2	
		E	F
P1	G	3, 3	0, 2
	H	2, 0	2, 2

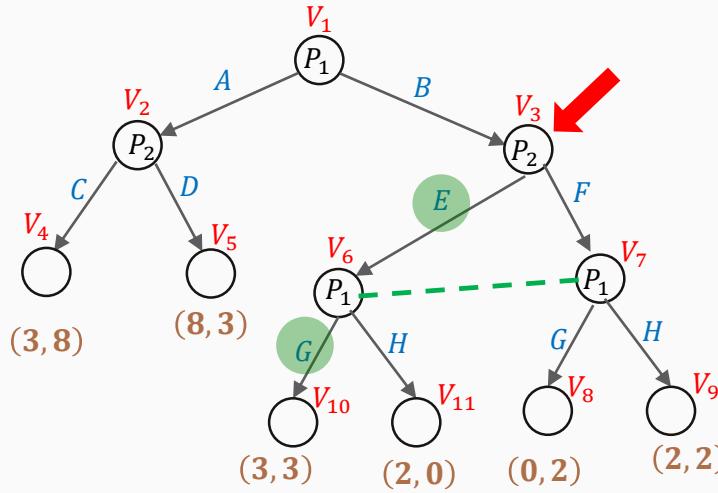
➤ NEs are:

$$\left(\left(\frac{2}{3} (G), \frac{1}{3} (H) \right), \left(\frac{2}{3} (E), \frac{1}{3} (F) \right) \right)$$

} PSNEs} MSNE

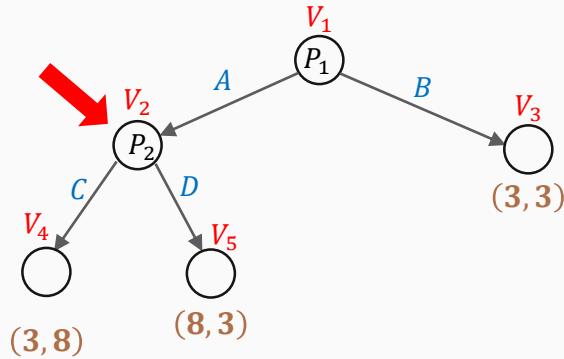
➤ For each of these NEs, we can do backward induction as before. We show it for one PSNE and one MSNE in the next few slides.

SPNE for IIEFGs



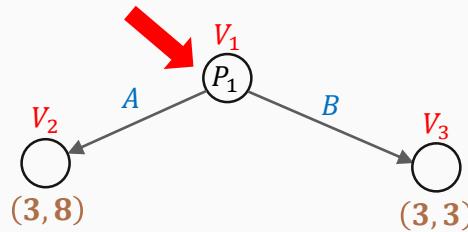
- Backward induction for NE (G, E) .
- The actions are highlighted using green circles. The corresponding payoff is (3, 3).

SPNE for IIEFGs



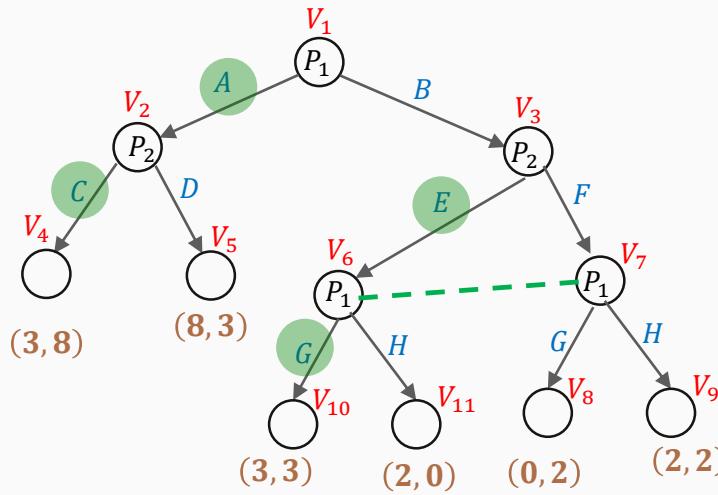
- Backward induction for NE (G, E) .
- P2 is playing in the vertex with marked with the red arrow. Action C has a payoff of 8 and C has a payoff of 3. So, P2 chooses C.

SPNE for IIEFGs



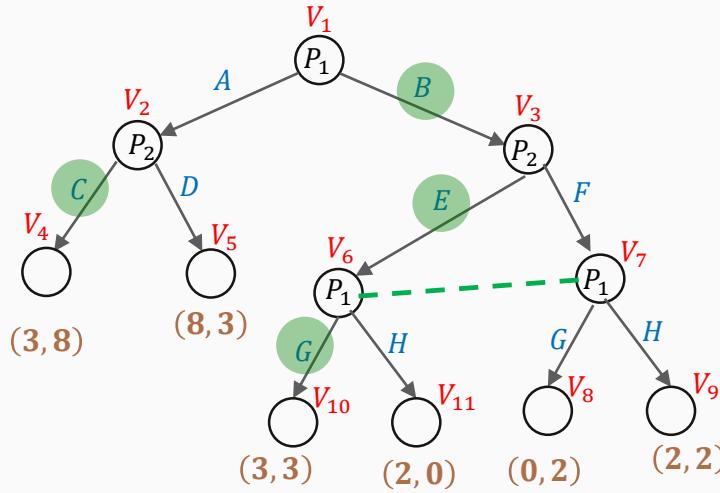
- Backward induction for NE (G, E) .
- P2 is playing in the vertex with marked with the red arrow. Both actions A and B has the same payoffs. So both are possible. In the next two slides, we will show the SPNE for actions A and B separately.

SPNE for IIEFGs



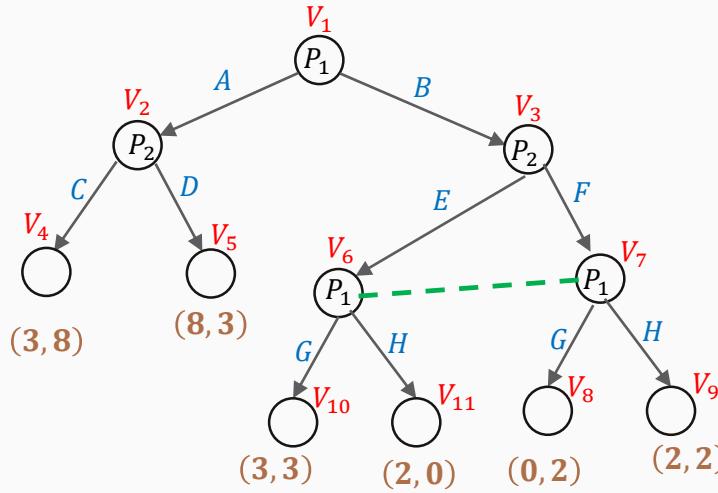
- Backward induction for NE (G, E) .
- Let the action taken by P1 at vertex V_1 be A.
- SPNE is:
 $((A, G), (C, E))$

SPNE for IIEFGs



- Backward induction for NE (G, E) .
- Let the action taken by P1 at vertex V1 be B.
- SPNE is:
 $((B, G), (C, E))$
- So essentially, there are two SPNEs when the subgame starting at node V3 has NE (G, E) . They are:
 $((A, G), (C, E))$
 $((B, G), (C, E))$

SPNE for IIEFGs



➤ Backward induction for NE

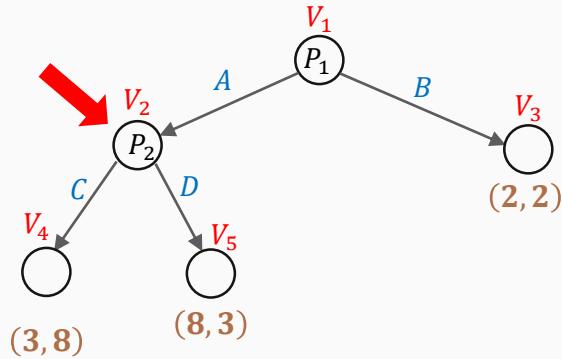
$$\left(\left(\frac{2}{3} (G), \frac{1}{3} (H) \right), \left(\frac{2}{3} (E), \frac{1}{3} (F) \right) \right)$$

➤ The payoff of the P1 and P2 for the above NE for the subgame at vertex V3 is

$$P1: \frac{2}{3} * \frac{2}{3} * 3 + \frac{2}{3} * \frac{1}{3} * 2 + \frac{1}{3} * \frac{2}{3} * 0 + \frac{1}{3} * \frac{1}{3} * 2 = 2$$

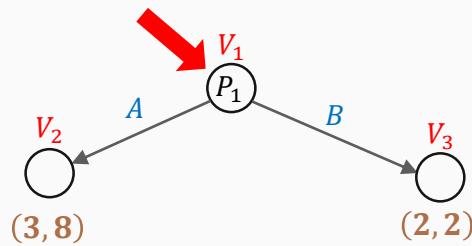
$$P2: \frac{2}{3} * \frac{2}{3} * 3 + \frac{2}{3} * \frac{1}{3} * 0 + \frac{1}{3} * \frac{2}{3} * 2 + \frac{1}{3} * \frac{1}{3} * 2 = 2$$

SPNE for IIEFGs



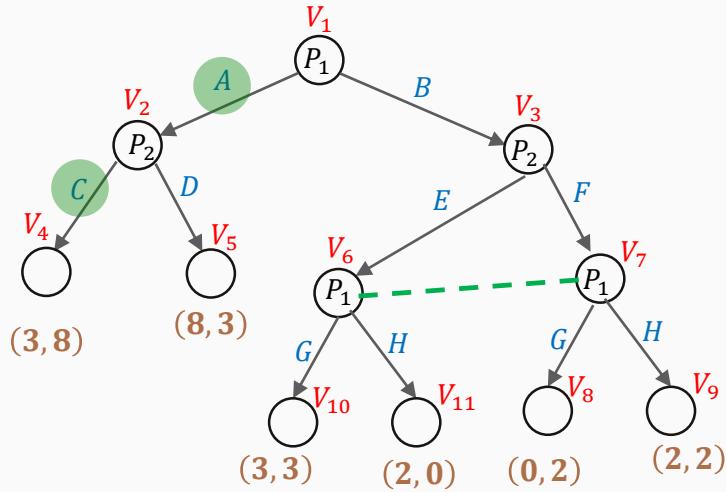
- Backward induction for NE
 $\left(\left(\frac{2}{3} (G), \frac{1}{3} (H)\right), \left(\frac{2}{3} (E), \frac{1}{3} (F)\right)\right)$
- P2 is playing in the vertex with marked with the red arrow. Action C has a payoff of 8 and C has a payoff of 3. So, P2 chooses C.

SPNE for IIEFGs



- Backward induction for NE $\left(\left(\frac{2}{3} (G), \frac{1}{3} (H)\right), \left(\frac{2}{3} (E), \frac{1}{3} (F)\right)\right)$
- P1 is playing in the vertex with marked with the red arrow. Action A has a payoff of 3 and action B has a payoff of 2. P1 choose A.

SPNE for IIEFGs



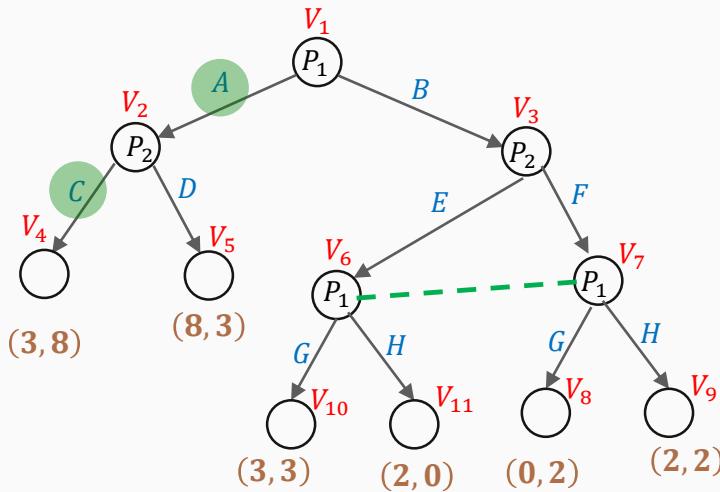
➤ Backward induction for NE

$$\left(\left(\frac{2}{3} (G), \frac{1}{3} (H) \right), \left(\frac{2}{3} (E), \frac{1}{3} (F) \right) \right)$$

➤ SPNE is,

$$\left(\left(A, \left(\frac{2}{3} (G), \frac{1}{3} (H) \right) \right), \left(C, \left(\frac{2}{3} (E), \frac{1}{3} (F) \right) \right) \right)$$

SPNE for IIEFGs



➤ So the SPNEs that we found till now are:

$$(A, G), (C, E)$$

$$(B, G), (C, E)$$

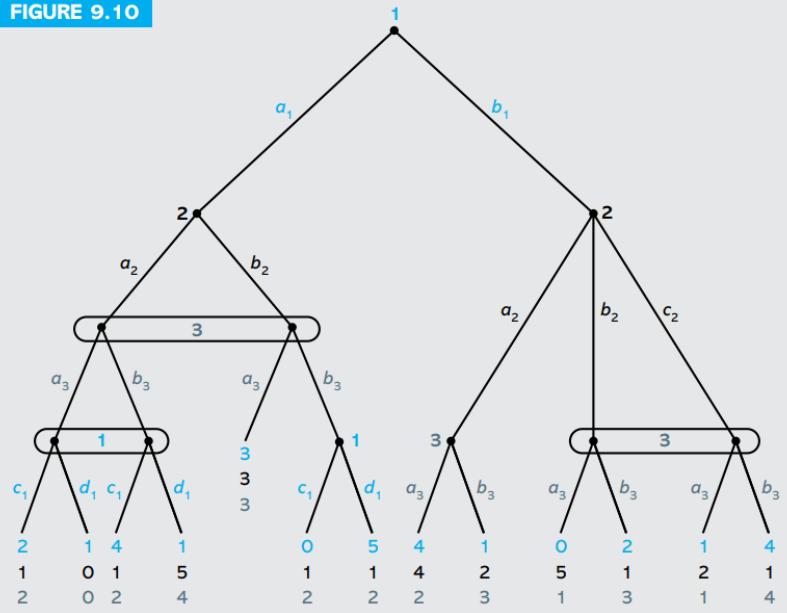
$$\left(\left(A, \left(\frac{2}{3} (G), \frac{1}{3} (H) \right) \right), \left(C, \left(\frac{2}{3} (E), \frac{1}{3} (F) \right) \right) \right)$$

➤ Find the SPNE when the NE for the subgame at vertex V3 is,

$$(H, F)$$

Solution Concepts for IIEFG

FIGURE 9.10

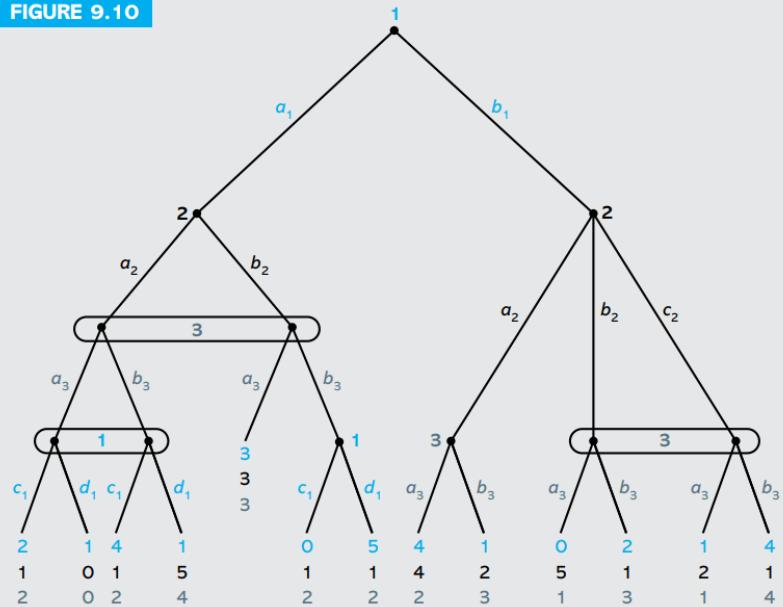


The Need for a Systematic Approach to Find Sub-Games

- For the problem that we discussed in the last few slides, the sub-game was easy to identify. This is always not so in case of imperfect information game.
- Consider the game shown in the left (directly adapted from the book by Harrington). What are the subgames?

Solution Concepts for IIEFG

FIGURE 9.10



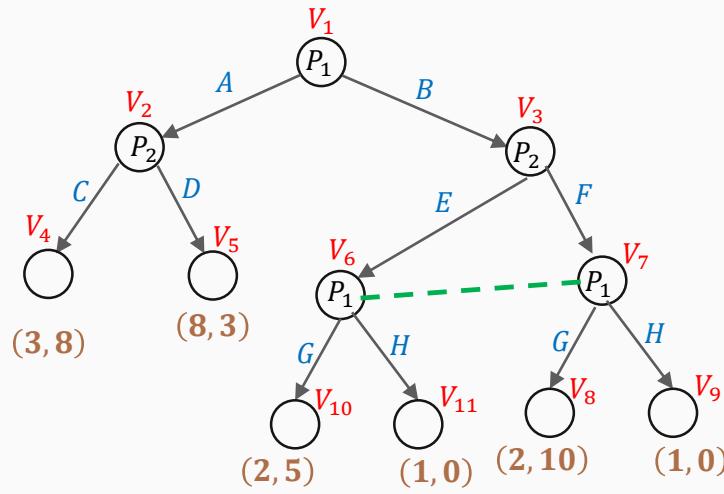
The Need for a Systematic Approach to Find Sub-Games

Definition (Subtree): A sub-tree of T is a tree T' consisting of a node in T (called the **root of the sub-tree**) and **ALL of its descendants in T**, along with ALL the edges that connect these nodes.

Definition (Regular Subtree): A regular sub-tree is a subtree that contains all information sets that have at least one node in the subtree.

In other words, to be a regular subtree, a subtree that includes one node of an information set must include all nodes of that information set.

Solution Concepts for IIEFG



Mixed Strategies vs Behavioral strategies



Thank You!