UNIVERSITY OF OREGON

Combining Simultaneous & Sequential Moves

Dante Yasui

Winter 2024

Outline

Games with both types of moves

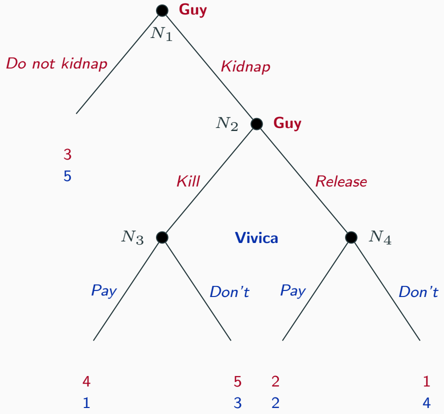
Combining Sequential and Strategic Form

SPNE vs NE

Games with both types of moves

- So far, we have only seen games that were either made up of *only sequential* or *only simultaneous* moves.
- But realistically, many strategic interactions will contain both types of moves.
- Learning to move between extensive and strategic form will help us examine more complicated games, as well as introduce us to the roll of *information* and beliefs which will become useful to later topics.

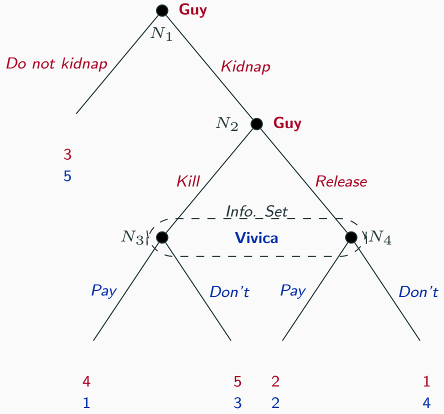
Original Kidnap Game



Modifying the game

- When we first saw this game, Vivica could observe whether Guy had killed Orlando before she paid the ransom.
- Now we will consider a variation of this game where Orlando's fate is decided by Guy *without Vivica knowing the result*.

Kidnap Game with a simultaneous subgame



- The dotted box around Vivica's nodes N_3 and N_4 indicate that she can't tell them apart.
- They are part of the same information set; to Vivica, they are indistinguishable from each other.

Definition 1

An *information set* is the set of information available to a player when making a decision. In the extensive form, every decision node belongs to one and only one info set, but multiple nodes may be in the same info set if the player at that node cannot tell them apart.

Combining Sequential and Strategic Form

Using Strategic Form in Extensive Games

Consider the situation when Guy has already kidnapped Orlando.

List the remaining strategies for each player:

Using Strategic Form in Extensive Games

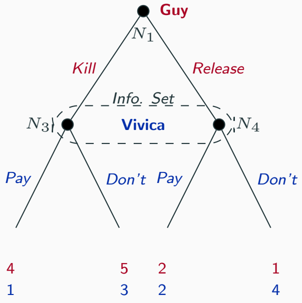
With these strategy profiles, we can write this subgame out in its strategic form:

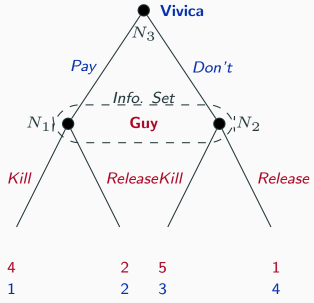
		Vivica	
		Pay	*Dont Pay*
Guy	*Kill*	$4, 1$	$2, 2$
	Release	$5, 3$	$1, 4$

Going from Strategic to Extensive

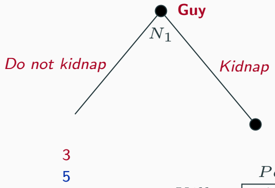
For every strategic form game, there are multiple extensive form game trees which can be drawn.

Two extensive subgames





Another way to represent the entire Kidnap Game



Vivica

Guy		Pay	$Dont\ Pay$
	$Kill$	$4, 1$	$2, 2$
	$Release$	$5, 3$	$1, 4$

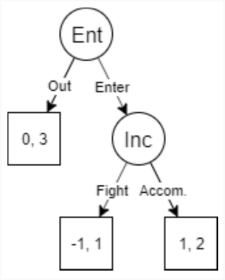
A Strategic Form representation

$$\begin{array}{llcccc} & & \multicolumn{4}{c}{\text{Guy}} \\ & & dnk, k & dnk, r & kid, k & kid, r \\ \text{Vivica} & pay & 5, 3 & 5, 3 & 1, 4 & 3, 5 \\ & no\ pay & 5, 3 & 5, 3 & 2, 2 & 4, 1 \end{array}$$

SPNE vs NE

Market Entry Game

- Consider this variation on the Entry Game, in which the Entrant is not as efficient as the Incumbent:



		Incumbent	
		Fight	*Accom.*
Entrant	*Enter*	-1, 1	1, 2
	Out	0, 3	0, 3

- Note that this game has two NEs: (Enter, Accommodate) and (Out, Fight).
- Does it really make sense for the Incumbent to ever Fight?

Refining Nash Equilibrium

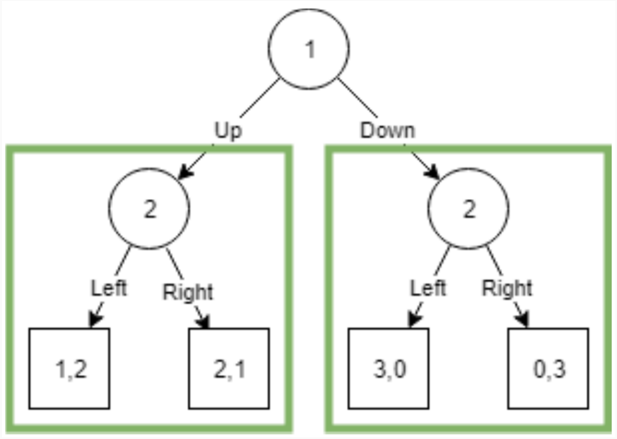
- The more you think about it, the less sense this Nash equilibrium makes.
- The only reason why the Entrant would choose to stay Out is if they really, truly believe that if they Entered, the Incumbent would Fight.
- . . . but it's hard to justify that belief when Fighting is costly for the Incumbent.
    - The Incumbent would have to somehow convince the Entrant that they really would Fight them, perhaps by making a credible threat.
    - We will talk more about this kind of strategic move later in the course.
- We often don't want this kind of Nash Equilibrium. To rule it out, we have the concept of subgame perfection.

Subgames

- A subgame is a subset of an extensive-form game which, itself, is a complete game.
- In the context of a game tree, a subgame is a **single node**, and all of the branches and nodes descending from it, such that it forms a complete game.

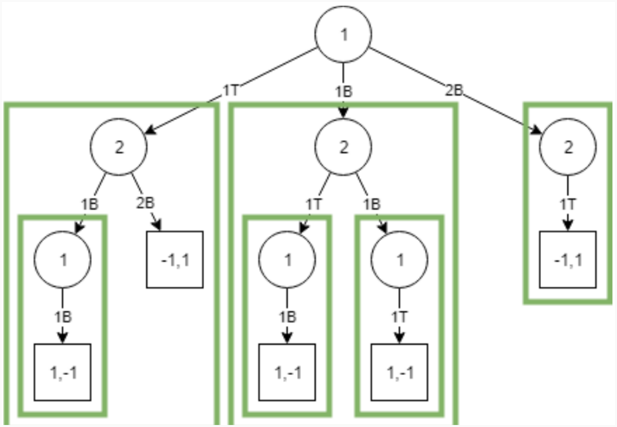
Examples of Subgames

- Each of these green boxes contains a subgame.



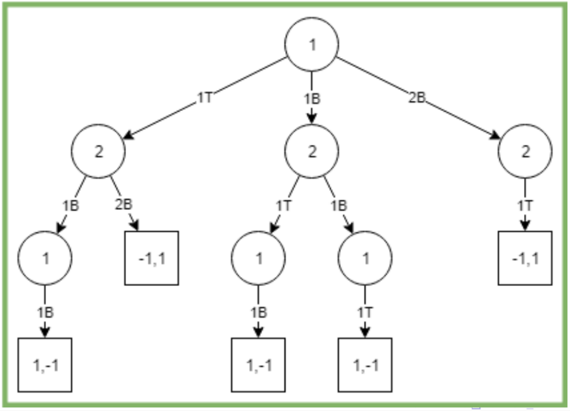
Examples of Subgames

- Each of these green boxes contains a subgame.
  - The smaller boxes are trivial subgames, because they don't contain any meaningful choices.



Examples of Subgames

- A game is, technically, a subgame of itself, but it is an improper subgame. (Any subgames other than the entire game are proper subgames.)

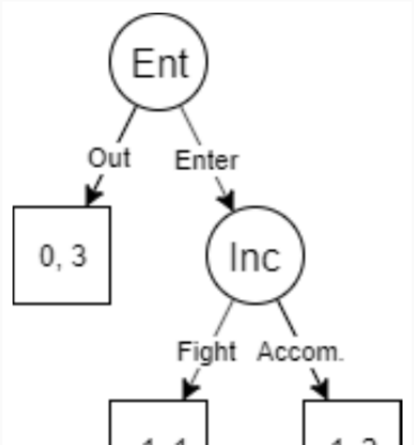


Subgame Perfection

- So why do we care about subgames?
- Because of a refinement of Nash equilibrium, called subgame-perfect Nash equilibrium, that does not produce the questionable equilibria we saw earlier.
- All subgame-perfect Nash equilibria are, obviously, regular Nash equilibria.
- A subgame-perfect Nash equilibrium also has to produce a Nash equilibrium **in every subgame**.

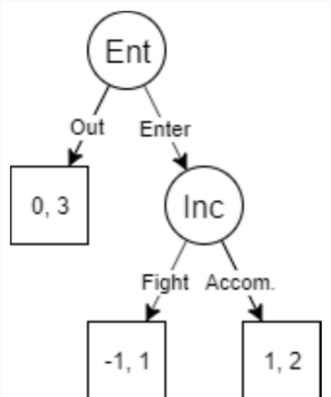
SPNE in the Entry Game

- There is only a single proper subgame in the Entry Game.
- A SPNE of this game must have a NE in that subgame.
- The only NE of the subgame is for the Incumbent to Accommodate.



SPNE in the Entry Game

- Given that the Incumbent Accommodates, the Entrant now faces a choice between staying Out, or Entering and being Accommodated.
- The Entrant's best response is to Enter, and the SPNE is (Enter, Accommodate)



Rationality and Subgame Perfection

- To justify looking for SPNEs instead of NEs, we must make the assumption of commonly known sequential rationality. There are two parts to this assumption:
  - Players are aware that other players are sequentially rational.
  - Knowing that other players are sequentially rational, each player can predict the others' rational behavior and will use this to choose an action which provides the largest possible payoff **at each node where they have a choice of actions**.
- One way of thinking about this is that sequentially rational players choose their strategy "on the fly," one move at a time, instead of all at once at the beginning of the game.

Finding SPNEs: Backward Induction

- The method for finding SPNEs is straightforward—we've already been using it without spelling it out formally.
    1. Find all the decision nodes in the **lowest row** of decision nodes. (This assumes that the game tree is neat and orderly—which it will be if I'm giving you the game tree.)
    2. Find the optimal choice for players to make at each of these nodes, and mark the corresponding branch (or erase/cross out all the other branches).
    3. If you just solved the game's initial node, you're done!
    4. Otherwise, identify all of the decision nodes in the **next row up** and return to step 2.
- At each step, remember to take into account the optimal moves that you've already found lower in the tree.