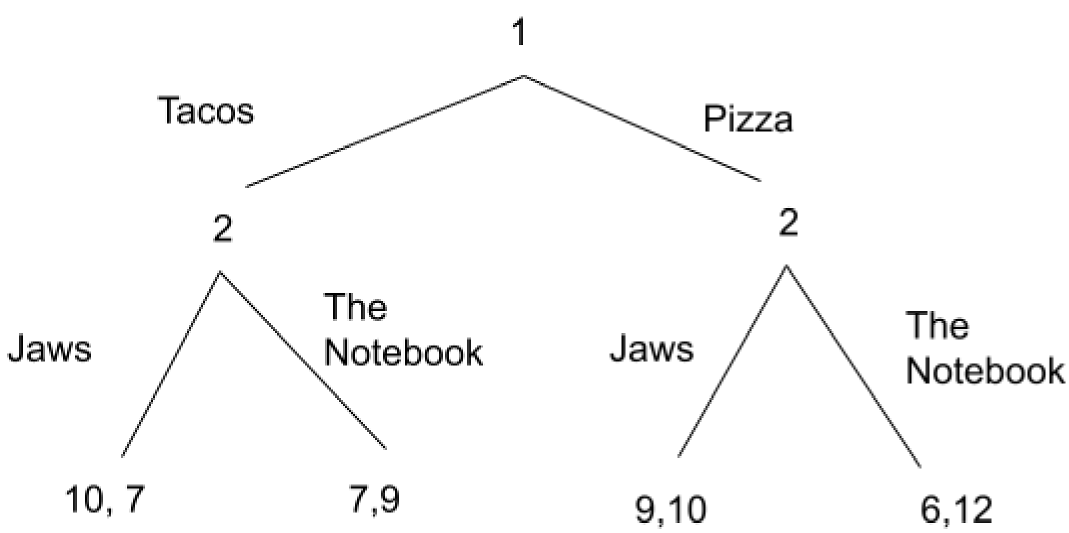
Introduction

Sequential Move Games

- Nodes, branches, path of play
- Strategies
- Backward Induction
- Normal Form
- Strict domination

A game with sequential moves involves players making decisions in a defined order. (Example: you and partner must pick movie and dinner)



*List strategies, strategy profiles, preferences over strategy profiles

A Sequence of Choices

Suppose you have to make a sequence of choices. For example you must decide whether to eat or not, then if you decide to eat, you must choose what to eat (Tacos, Pizza).
Drawing
Don't include payoffs when you first draw it.

Strategies
(Don't eat, Tacos), (Don't eat, Pizza), (Eat, Tacos), (Eat, Pizza)

Payoffs
Let's find payoffs that represent an example where we are hungry for tacos.

What is the best response for this case above?
What’s the best response if you aren’t hungry and prefer pizza?

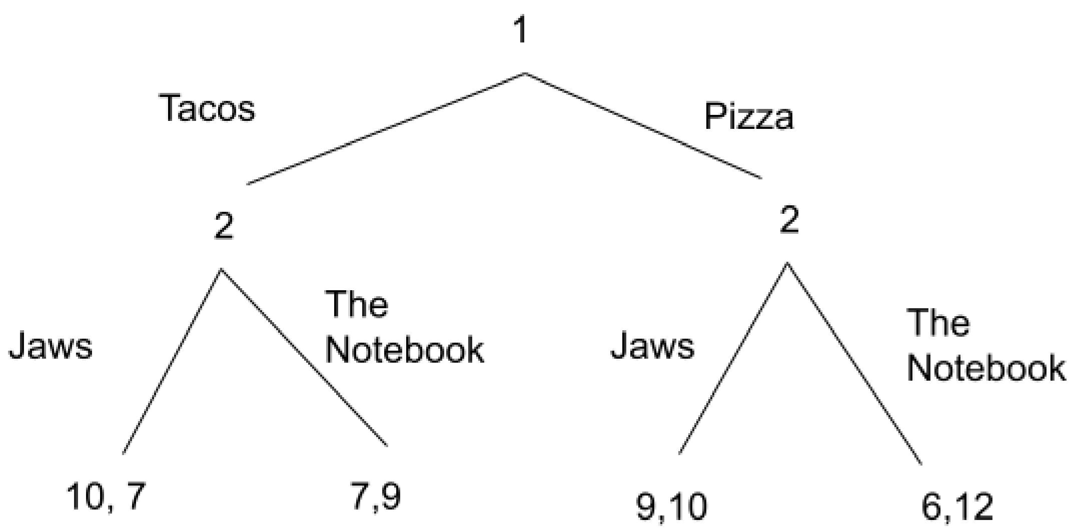
Recall that a sequential game is one in which the players make decisions in some prearranged order. This doesn’t have to be alternating turns—the players can take turns in any order, as long as they both understand what that order is ahead of time.

The order of turns can depend on decisions made by either player—consider the card game UNO, in which some cards reverse the order of play or skip the next player.
The order of turns can also depend partly (or wholly) on chance—consider a card game in which the first player is determined by drawing a card. (We’ll see more on this kind of game in a few weeks.)

The important thing is that, in a sequential game, the players make decisions one after the other. For now, we’ll also be assuming that when players make their decisions, they also know what decisions other players have already made—although we’ll relax that assumption later on.

Depicting Sequential Games

In order to visualize a sequential game, we use a game tree, which looks something like this



The important features of a game tree include:
Nodes, the “dots” or “vertices” in the game tree. There are two types of nodes:

–Decision nodes, which each represent a decision that a player must make. Each decision node is associated with one specific player. One decision node represents the first decision made in the game and is called the initial node.

Terminal nodes, which each represent a different outcome of the game. At the abterminal node, there are no more decisions to be made by any player.

Branches, the “lines” or “edges” in the game tree. Branches represent the different choices that a player has at a decision node, and they show which player acts next after a decision is made. Each decision node must have at least two branches coming from it, and other than the initial node, each node must have exactly one branch going into it. Terminal nodes do not have any branches coming from them.

Payoffs for each player, at each terminal node. These payoffs can be listed in any order, so long as it is always the same order throughout the game tree, and that order is clearly explained. (A common convention is to list payoffs in the order that players get to make decisions.)

Path of Play

In a sequential game, a path of play is a sequence of actions taken, leading from the initial node all the way to a terminal node. This sequence fully describes one way that the game may be played. However, the path of play by itself is not useful for finding equilibria: we need to work with strategies, instead.

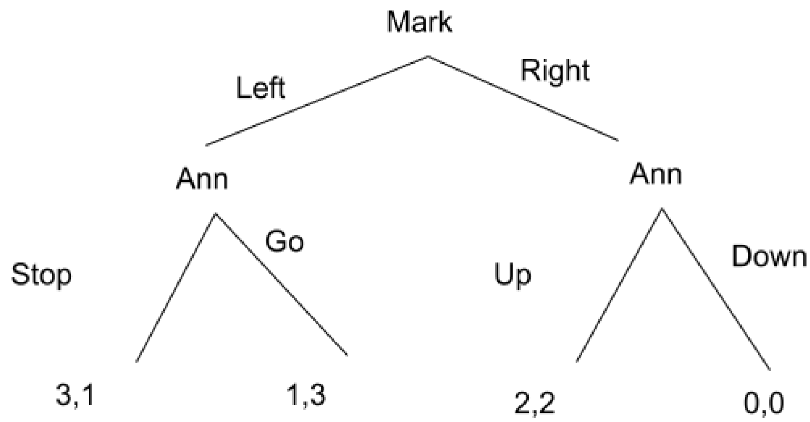
Strategies in Sequential Games

Recall that a strategy is an information-contingent plan of action. In a sequential game, that means that in order to be complete, a strategy has to describe what a player would do in any possible situation: that is, at any node of the game tree. In other words, when we describe a player's strategy for a sequential game, we need to list the action they would take at every single node. That includes nodes that are never reached in the path of play!

Counting Strategies

It's straightforward to count the number of strategies available to a player (and this will be useful in a couple of weeks, when we start talking about strategic-form games). Because strategy must include one action at each of a player's decision nodes, we can use the counting principle to find the number of possible strategies: simply count the number of actions available at each of a player's decision nodes, and then take the product of all of these numbers.

Example 1: Mark and Ann



1. List Strategies VERY IMPORTANT
   Mark: (Left), (Right)
   player 2: (Stop, Up), (Stop, Down), (Go, Up), (Go, Down)

2. Strategy profiles

Strategy profiles	Payoffs
(Left,(Stop, Down))	(3,1)
(Left,(Stop, Up))	(3,1)
(Left,(Go, Down))	(1,3)
(Left,(Go, Up))	(1,3)
(Right,(Stop, Up))	(2,2)
(Right,(Go, Up))	(2,2)
(Right,(Stop, Down))	(0,0)
(Right,(Go, Down))	(0,0)

Backward Induction or Best Response

Subgame: is a subset of an extensive-form game which, itself, is a complete game. It starts at a decision node and must contain all payoffs that descend from that node.

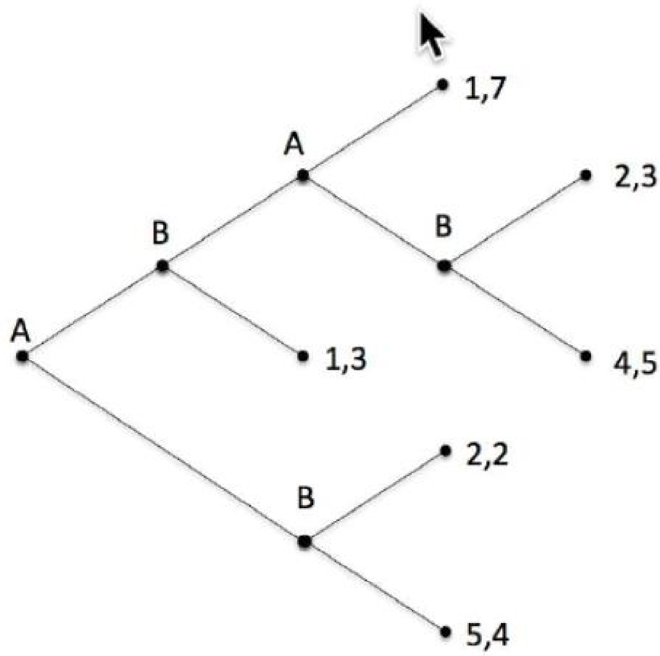
Subgame perfect Nash equilibria: a refinement of Nash equilibria in which players must choose the best strategy in each subgame.
Or
a refinement of Nash equilibria in which players must choose the best response for every choice even the ones not reached.

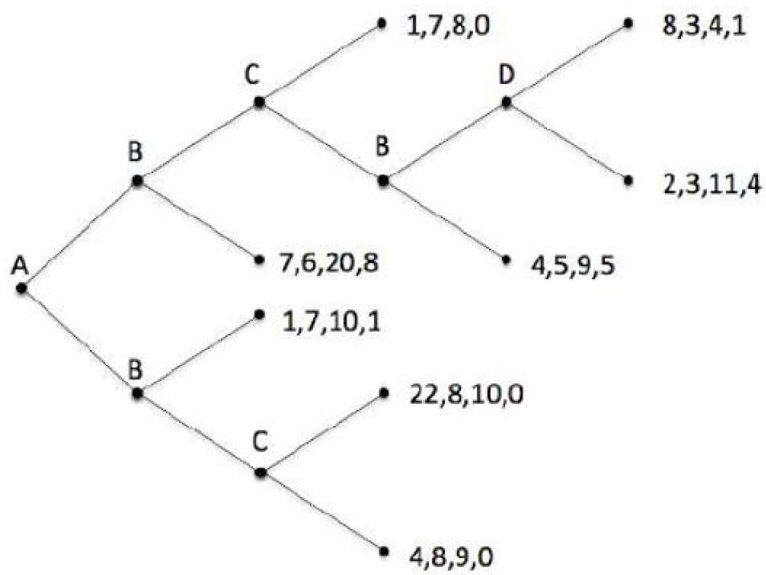
Backward Induction or best response in each subgame

- We start by looking at all of the “last decisions” in the game tree—those are decision nodes with no other decision nodes that come after them—and we mark the outgoing branch at each node which gives the decision-maker the highest payoff.
- Then we consider the “second-to-last” decision nodes. This time, when we’re comparing payoffs, we take into account what we’ve figured out about the last decisions
- We continue rolling back like this, until we get all the way to the initial node. At this point, we should have marked one outgoing branch at each decision node: these marked branches constitute a strategy profile which is a Nash Equilibrium. It’s not necessarily the ONLY Nash Equilibrium,
  Because
  Backward induction rules out any Nash Equilibrium that relies on a threat: a player might want to threaten to do something irrational at one decision node, so that another player makes choices that avoid that decision node.

Backward Induction

(Not three players)





Nash Equilibrium

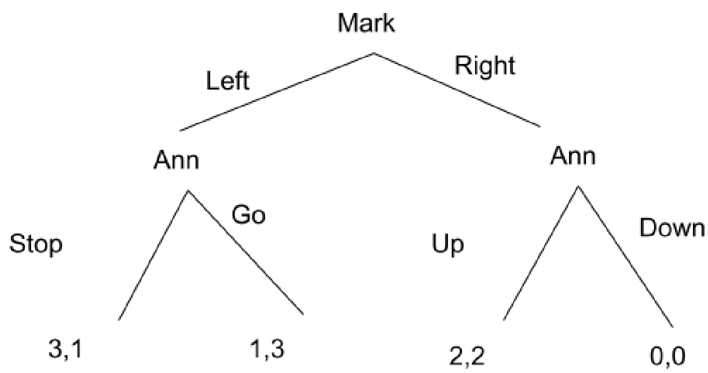
A Nash Equilibrium is a strategy profile (a set of strategies, one for each player) in which no player wants to change their own strategy. In other words, no player can improve their own payoff by making changes to their strategy, assuming that the other players don't make any changes either.

Normal Form

Another way to think of a sequential game is the normal form. Players choose their strategies simultaneously.

- list all possible strategies, it's important to get these correct
- List strategy profiles and payoffs for those strategies
- Construct a payoff matrix using the strategies, and strategy profile payoffs.

Mark and Ann



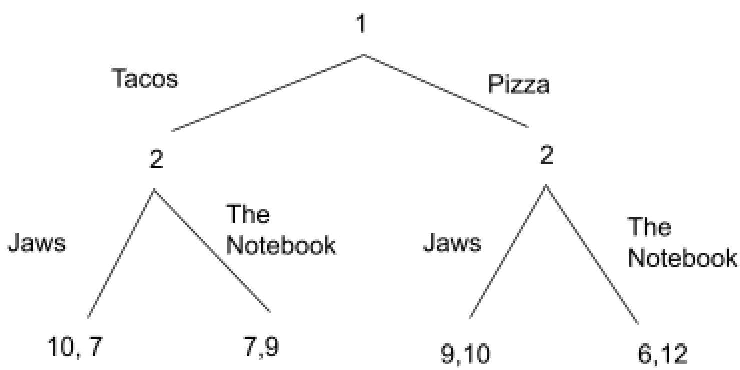
1. List Strategies
   Mark: (Left), (Right)
   player 2: (Stop, Up), (Stop, Down), (Go, Up), (Go, Down)
2. Backward Induction
3. Strategy profiles

Strategy profiles	Payoffs
(Left,(Stop, Down))	(3,1)
(Left,(Stop, Up))	(3,1)
(Left,(Go, Down))	(1,3)
(Left,(Go, Up))	(1,3)
(Right,(Stop, Up))	(2,2)
(Right,(Go, Up))	(2,2)
(Right,(Stop, Down))	(0,0)
(Right,(Go, Down))	(0,0)

4. Normal Form: Construct a payoff matrix

	Left	Right
SD	3,1	0,0
SU	3,1	2,2
GD	1,3	0,0
GU	1,3	2,2

Data Night



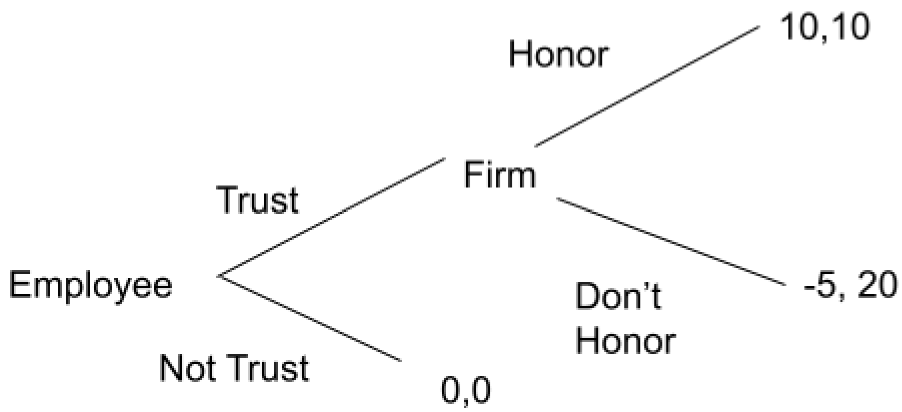
1. Strategies
   player 1: Tacos or Pizza
   player 2: (Jaws, Jaws), (Jaws, Notebook), (Notebook, Jaws), (Notebook, Notebook)
2. Backward Induction
3. Strategy Profiles

Strategy Profiles	payoffs
(Tacos, (Jaws, Jaws))	10,7
(Pizza,(Jaws, Jaws))	9,10
(Tacos, (Jaws, Notebook))	10,7
(Pizza, (Jaws, Notebook))	6,12
(Tacos,(Notebook, Jaws))	7,9
(Pizza,(Notebook, Jaws))	9,10
(Tacos,(Notebook, Notebook))	7,9
(Pizza,(Notebook, Notebook))	6,12

4. Normal Form: Construct a payoff matrix

	Tacos	Pizza
JJ	10,7	9,10
JN	10,7	6,12
NJ	7,9	9,10
NN	7,9	6,12

Trust Honor Game



1. Strategies
   Firm: Don't Honor and Honor
   Employee: Trust and Don't Trust
2. Backward Induction
3. Strategy Profile — Payoff

Strategy Profile	Payoff
(T,DH)	(-5,20)
(T,H)	(10,10)
(DT, H)	(0,0)
(DT,DH)	(0,0)

4. Normal Form: Construct a payoff matrix

		Firm	
		Honor	No Honor
Employee	Trust	(10,10)	(-5,20)
	Don't Trust	(0,0)	(0,0)