Lecture 10:
Games in Extensive Form

Jan Zouhar

Games and Decisions

Sequential Move Games

- so far, we have only dealt with *simultaneous* games (players make the decisions at the same time, or simply without knowing what the action of their opponent is)
- in most games played for amusement (cards, chess, other board games), players make moves in turns → **sequential move games**
- many economic problems are in the sequential form:
  - English auctions
  - price competition: firms repeatedly charge prices
  - executive compensation (contract signed; then executive works)
  - monetary authority and firms (continually observe and learn actions)
  - …
- formally, we describe these as **games in extensive form** (representation using a **game tree**)

Class Game: Century Mark

- **rules**:
  - played by fixed pairs of players taking turns
  - at each turn, each player chooses a number (integer) between 1 and 10 inclusive
  - this choice is added to sum of all previous choices (initial sum is 0)
  - the first player to take the cumulative sum *above* 100 (*century*) loses the game

- **prize**: *5 extra points for the final test* (!!!)

Volunteers?

Class Game: Century Mark (cont'd)

Analysis of the game

- what's the winning strategy?
  - broadly speaking, bring the sum to 89; then your opponent can't possibly win
  - actually, it's enough to bring it to 78; then you can make sure to make it 89 later
  - reasoning like this, the winning positions are:
    100, 89, 78, 67, 56, 45, 34, 23, 12, **1**

→ the first mover can guarantee a win!

  - *winning strategy*:
    - in the first move, pick **1**
    - then, choose 11 minus the number chosen by the second mover
- *note*: strategy = a complete plan of action

Sequential Move Games with Perfect Information

- models of strategic situations where there is a strict order of play
- perfect information implies that players know…
  - … the rules of the game
    - possible actions of all players
    - resulting payoffs
  - … everything that has happened prior to making a decision

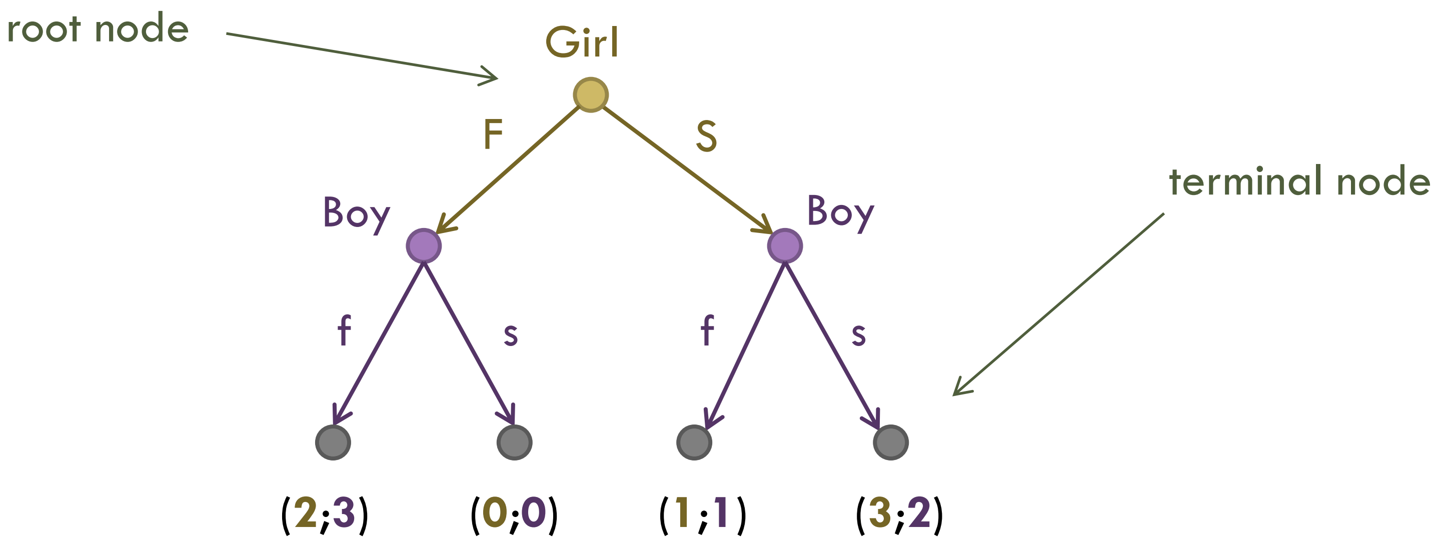
- most easily represented using a **game tree**
  - tree = graph, *nodes* connected with *edges*
    - *nodes* = decision-making points, each non-terminal (see below) node belongs to one of the players
    - *edges* = possible *actions* (*moves*)
    - *root node*: beginning of the game
    - *terminal nodes* (*end nodes*): end of the game, connected with payoffs

Example 1: Sequential Battle of the Sexes

- simultaneous moves:
  a bimatrix game (*normal form*)

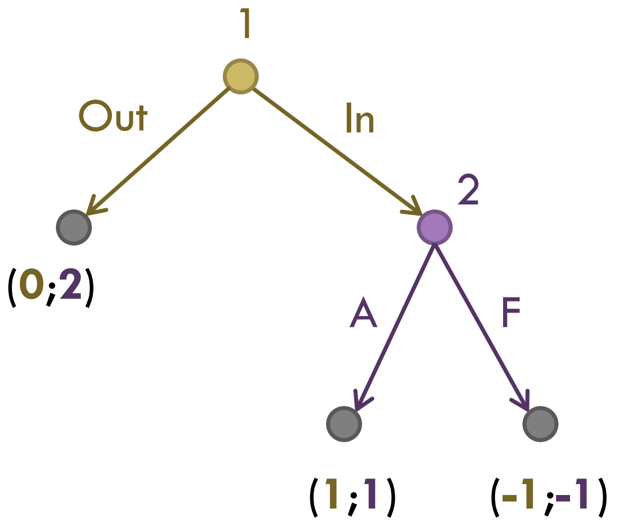
Girl \ Boy	f	s
F	2 ; 3	0 ; 0
S	1 ; 1	3 ; 2

- girl moves first: a sequential move game (*extensive form*)



Example 2: Model of Entry

- currently, firm 2 is an incumbent monopolist
- firm 1 has the opportunity to enter
- after firm 1 makes the decision to enter (*In* or *Out*), firm 2 will have the chance to choose a pricing strategy; it can choose either to *fight* (F) the entrant or to *accommodate* (A) it with higher prices



Exercise 1: Stackelberg Duopoly

- suppose firm 1 develops a new technology before firm 2 and as a result has the opportunity to build a factory and commit to an output level q_1 *before* firm 2 starts
- firm 2 then observes firm 1 before picking its output level q_2
- assume that:
  - output levels can only be 0,1, or 2 units of production
  - market price function (inverse demand) is: p = 3 – (q_1 + q_2)
  - production costs are 0

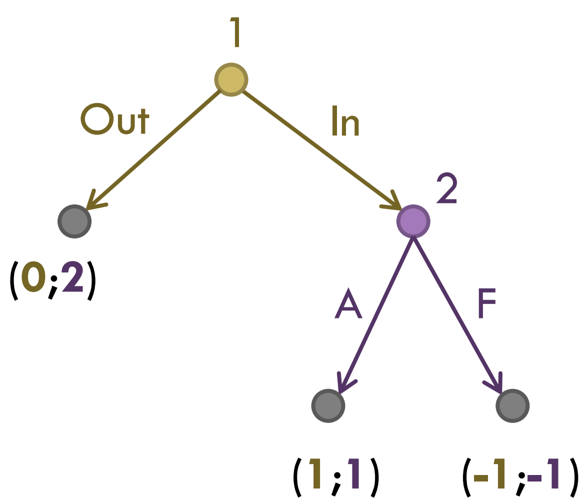
1. How many decision (= non-terminal) nodes are there in the game tree?
2. Draw the game tree

Strategies vs. Actions

- **action** = decision taken in any node of the game tree
- **strategy** = *complete contingent plan* explaining what a player will do in any situation that arises
  - specifies the choice to be made at each decision node
  - the sort of advice you would give to somebody playing on your behalf
- example strategies:
  - *Model of entry*
    - firm 1: *In*, *Out*
    - firm 2: *A*, *F*
  - *Battle of the sexes*
    - Girl: *F*,*S*
    - Boy: *ff*, *fs*, *ss*, *sf* (first letter: case *F*, second letter: case *S*)
  - *Stackelberg duopoly*
    - firm 1: 0,1,2
    - firm 2: 000, 001, 002, 010, 011, 012, 020, 021, … ($3^3 = 27$ strategies)

Normal Form Analysis of Move Games

- every extensive form game can be translated into a normal form game by listing the available strategies
- **Example**:
  - Model of entry:

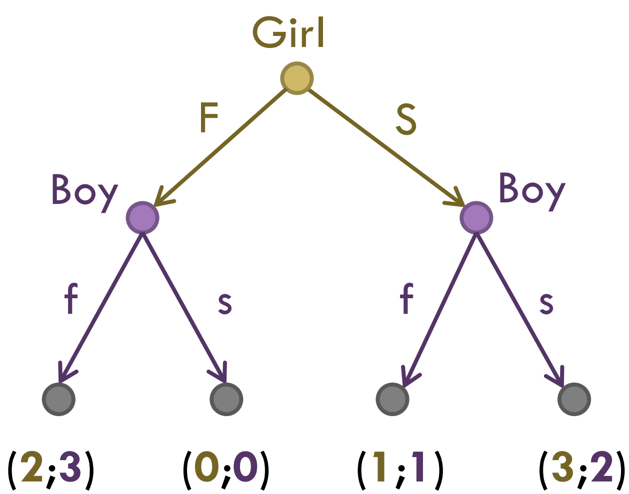


1 \ 2	A	F
Out	0 ; 2	0 ; 2
In	1 ; 1	-1 ; -1

- normal form allows us to find NE's
  - here: (*In*,*A*) and (*Out*,*F*) ← "Stay out or I will fight!"

Normal Form Analysis of Move Games (cont'd)

- **Another example**:
  - Battle of the sexes:

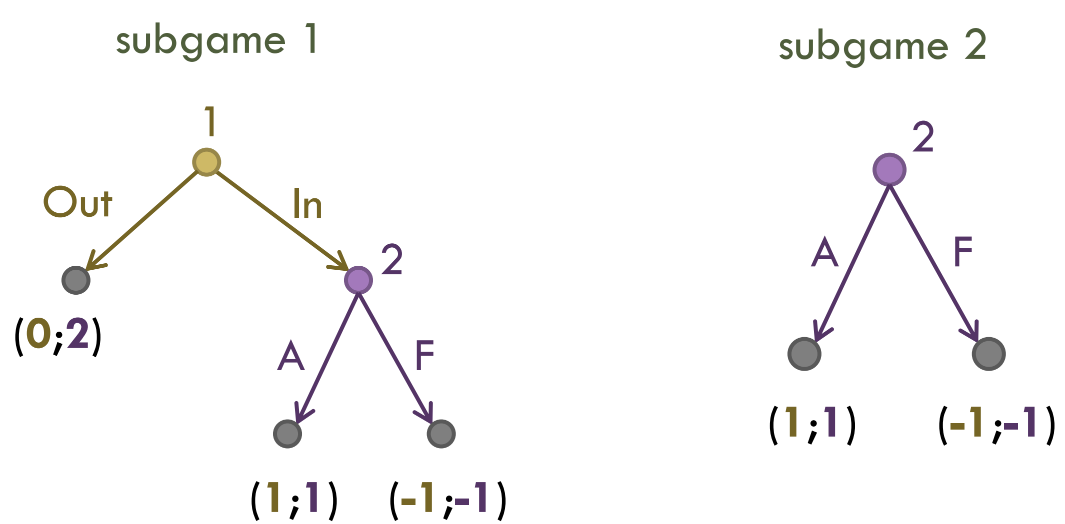


G \ B	ff	fs	sf	ss
F	2 ; 3	2 ; 3	0 ; 0	0 ; 0
S	1 ; 1	3 ; 2	1 ; 1	3 ; 2

- *criticism*:
  1. too many strategies even for simple trees (consider Stackelberg)
  2. too many NE's, some of them not very plausible (non-credible threats: "If you go shopping, I'll go to the football game anyway")

Backward Induction, or SPNE's

- **backward induction** is a method to find a NE that is "plausible" (or, in some games, a *winning strategy*)
- solutions found using backward induction are so called **subgame perfect nash equilibria** (SPNE's)
  - definition: *A strategy profile s* is a subgame perfect equilibrium of game G if it is a Nash equilibrium of every subgame of G.*
  - **subgames**: Model of entry



Backward Induction, or SPNE's (cont'd)

- *backward induction* finds best responses from terminal nodes upwards:

Step 1: start at the last decision nodes (neighbors to terminal nodes). For each of the decision nodes, find the deciding player's best action (payoff-maximizing one).

Step 2: replace the decision nodes from step 1 with terminal nodes, the payoffs being the profit-maximizing payoffs from step 1. (i.e., suppose the players always maximize profit)

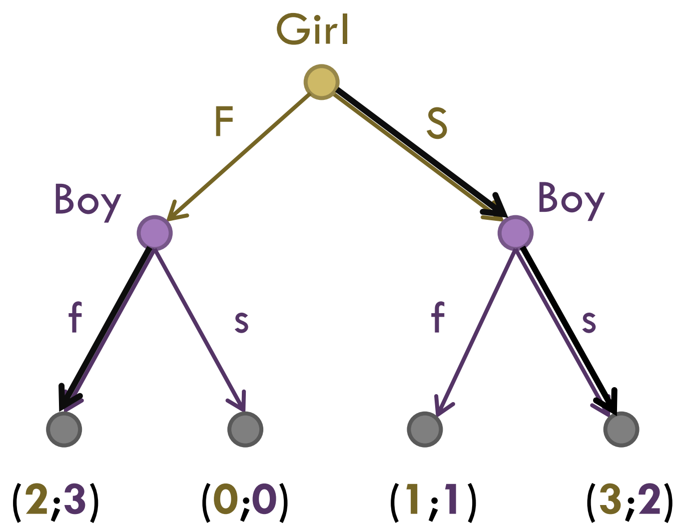
Step 3: repeat steps 1 and 2 until you reach the root node.

- Notes:
  - in step 1, each decision node with its terminal neighbors constitutes a *subgame* (→ we find *subgame perfect* NE's)
  - in principle, this resembles the approach we used in the *Century mark* game (100 wins → 89 wins → … → 1 wins)
  - in practice, we're not exactly replacing nodes in the tree; instead, we mark the used branches and "prune" the unused ones

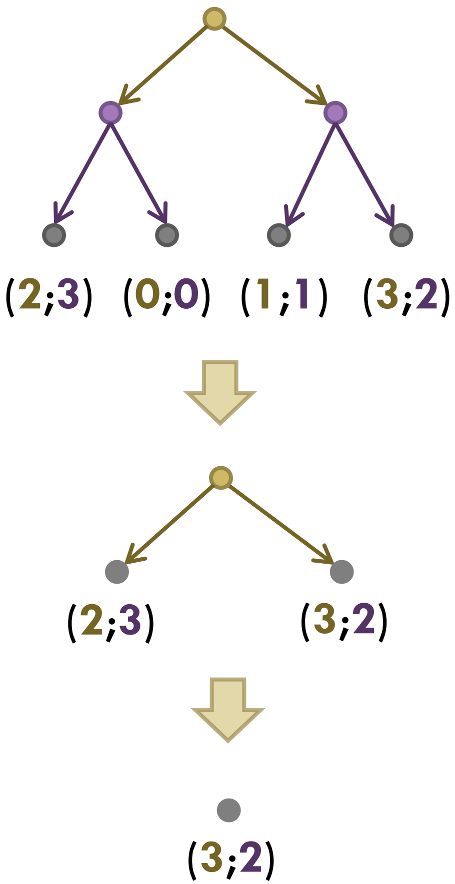
Backward Induction, or SPNE's (cont'd)

Example: Sequential battle of the Sexes

Notation (solving manually)

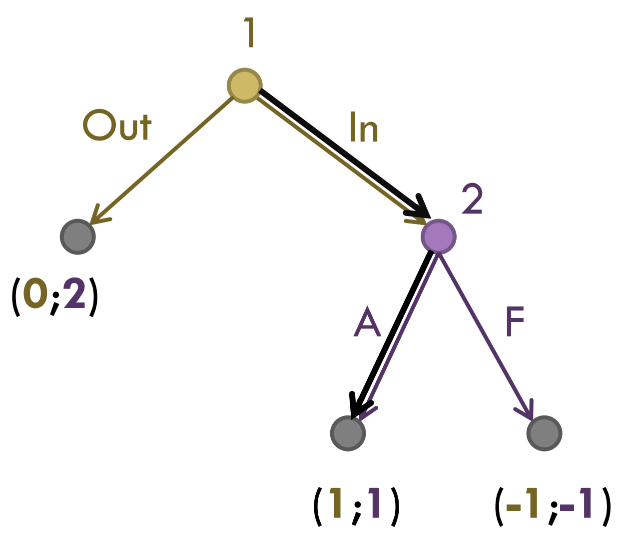


Algorithm reference:



- SPNE move sequence: S-s

Example: Model of entry



- SPNE move sequence: *In-A*

Exercise 2: Reversed Sequential BoS

- consider Battle of sexes again
- this time, the *boy* moves first

Girl \ Boy	f	s
F	2 ; 3	0 ; 0
S	1 ; 1	3 ; 2

1. Draw the game tree for this game
2. Find a SPNE using backward induction.
3. Compare the results with the "girls-first" version of the game. Is it an advantage to be a first mover in this game?

First-Mover Advantage

- Is there a first-mover advantage?
  - in many real-life and artificial games, yes:
    - BoS
    - Model of Entry
    - Stackelberg duopoly
    - Chess, checkers, many other board games
  - can you think of any games with a second-mover advantage?

- Games with a second-mover advantage:
  - English auction.
  - Cake-cutting: one person cuts, the other gets to decide how the two pieces are allocated
  - some versions of the game of *nim* (see next slides)

Example 3: Game of Nim

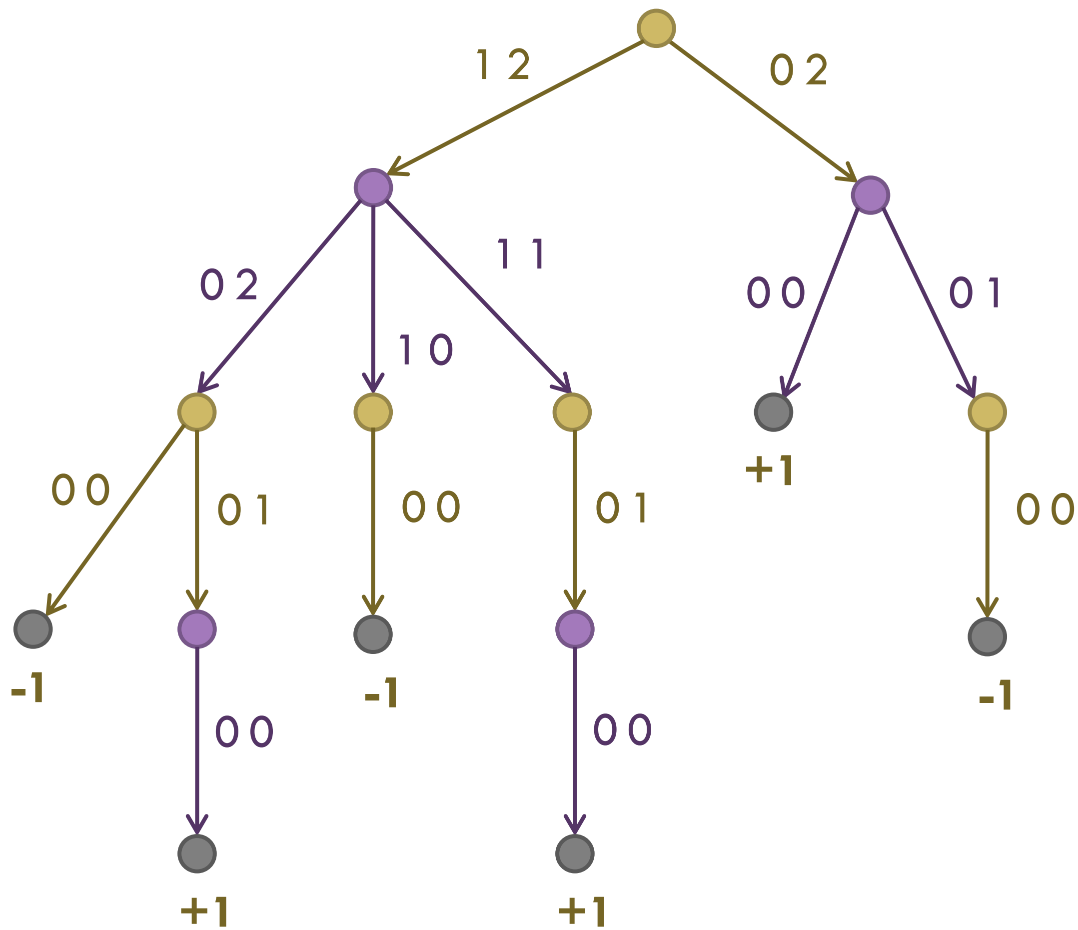
- two players take turns removing objects (matches, tokens) from distinct heaps (piles, rows)
- on each turn, a player must remove an arbitrary number of objects (one or more) from a single heap
- the player to remove the last object loses the game (*zero-sum game*)
- *origins*: centuries ago; mathematical description by Bouton in 1901, the name probably comes from the German word *"nimm"* = *"take!"*
- *notation*: numbers of objects in heaps:

3,4,5 nim game

1,3,5,7 nim game

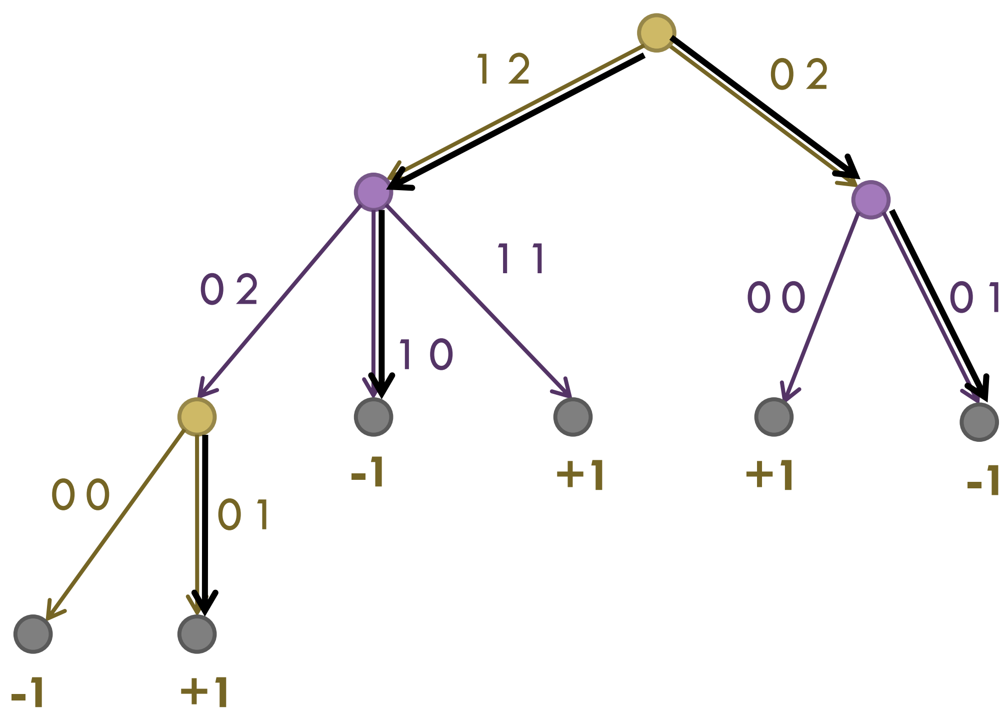
Example 3: Game of Nim (cont'd)

- game tree for 2,2 nim (symmetric moves omitted):



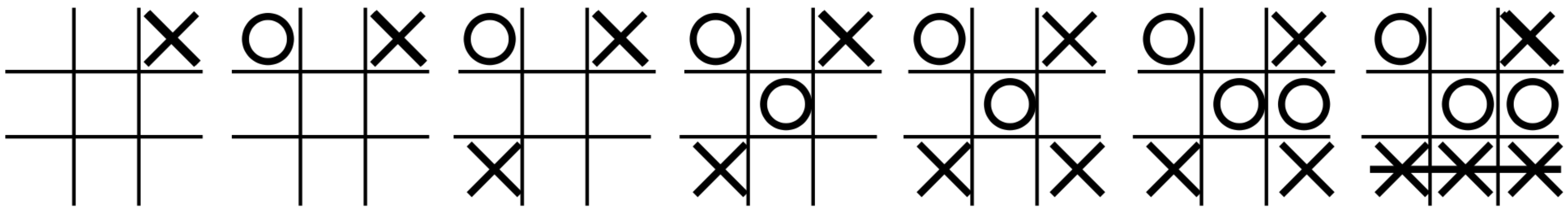
Example 3: Game of Nim (cont'd)

- simplified game tree (non-branching nodes omitted):



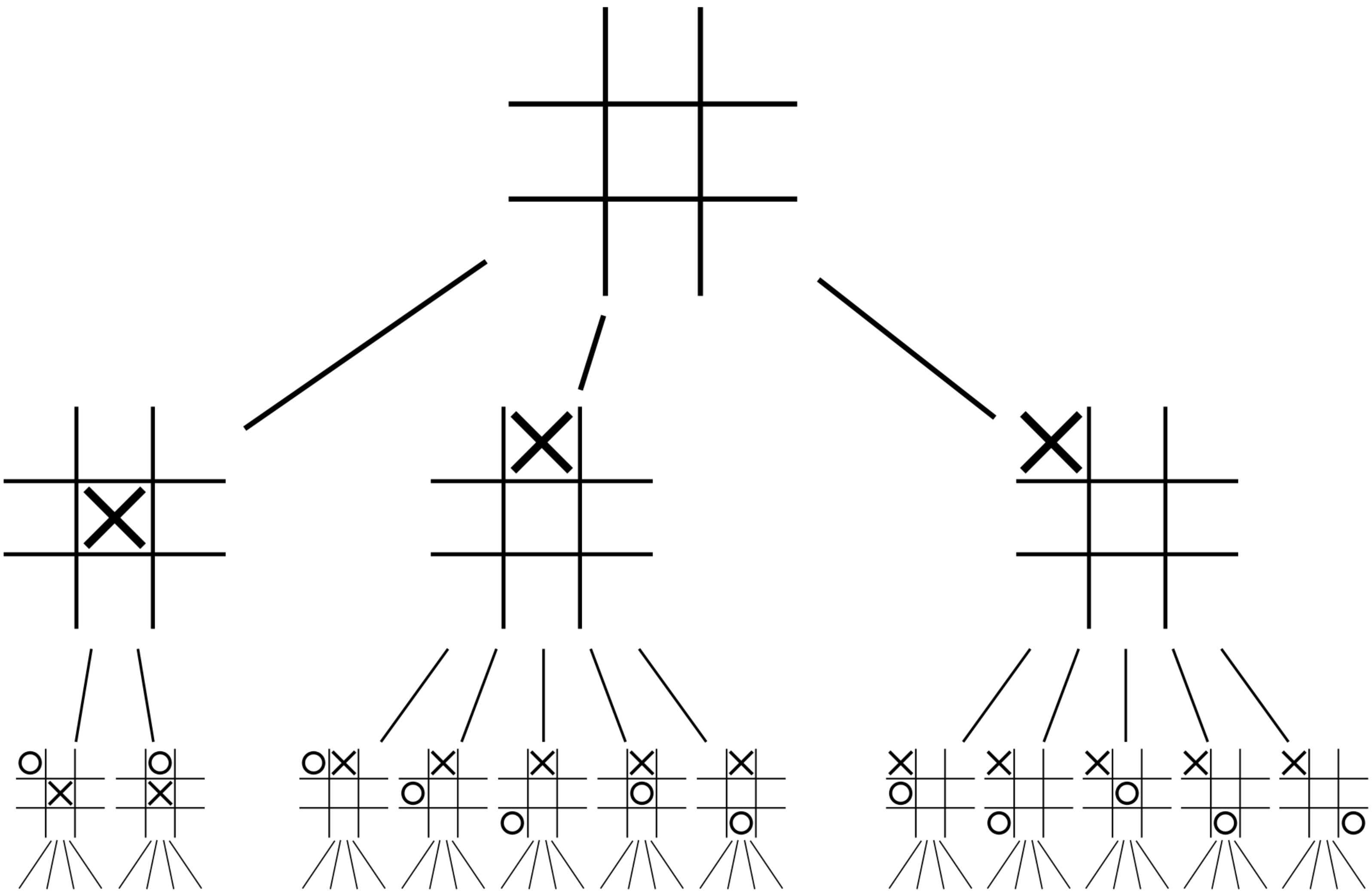
- player 1 can never win here (unless by fault of player 2)
- simple winning strategy for 2 heaps – *leveling up*: as long as both heaps at least 2, make them equal size with your move

Example 4: Tic-tac-toe



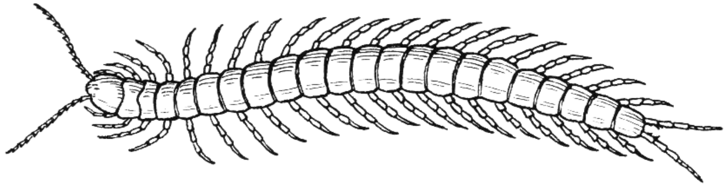
- game tree quite complex (see next slide)
  - manual calculation of the backward induction is cumbersome
  - suitable for computer analysis
  - one of the first "video games"
  - conclusions from the analysis can be formulated in a set of tactical rules (or: strategic algorithm)
    - similar to the *nim* situation with more than 2 heaps (more-or-less simple rules to apply the best strategic decisions)
  - for other complex game trees, this is not possible

Upper part of the tic-tac-toe game tree
(symmetric moves omitted)



Exercise 4: Centipede Game

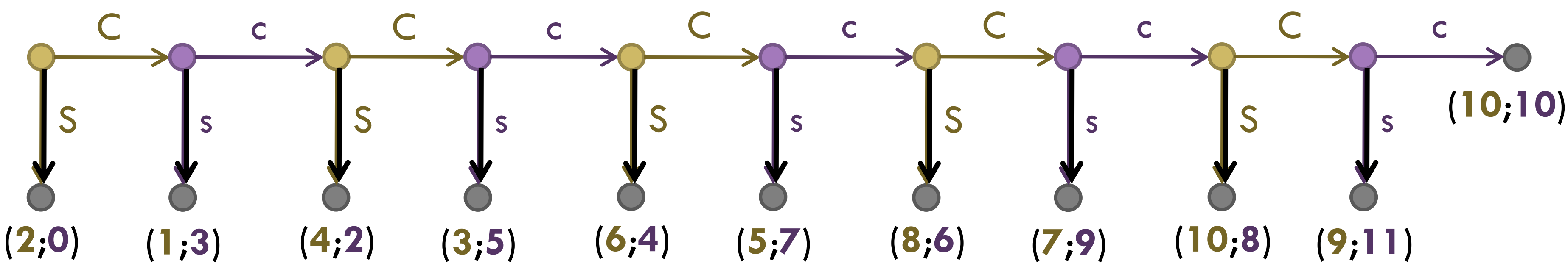
- two players at a table, two heaps of money (initially: \$0 and \$2)
- on his/her move, a player can either:
  - take the larger heap and leave the smaller one for the other player (*stop*, *S*)
  - push the heaps across the table to the other player, which increases both heaps by \$1 (*continue*, *C*)
- this can go on up until 10th round (player 2's 5th move), where instead of increasing the amounts in heaps, the heaps are distributed evenly amongst the players in case of *C*



1. Play the game in pairs.
2. Can you draw the game tree (or part of it, at least)?
3. Try to find the SPNE in the game.

Exercise 4: Centipede Game (cont'd)

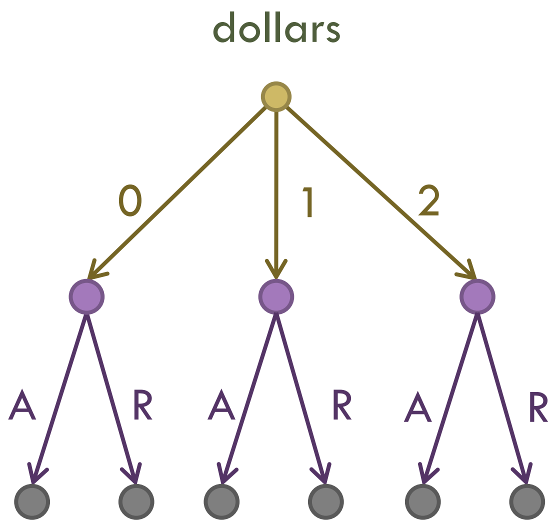
- backward induction:

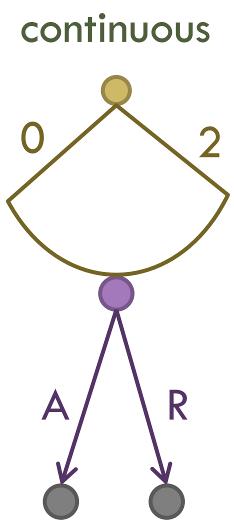


- critique of the SPNE:
  - doesn't reflect the way people behave in complicated games (limited *normativity*)
  - real decision-makers can only go 3-4 nodes "deep"

Example 5: Ultimatum Game

- two players interact to decide how to divide a sum of money offered to them (say, $2)
- player 1 proposes how to divide the sum, player 2 either accepts (A), or rejects (R)
  - if player 2 accepts, player 1's proposal is carried out
  - if player 2 rejects, *neither player receives anything*
- number of possible divisions: dollars, cents or continuous





Example 5: Ultimatum Game (cont'd)

- strategies:
  - player 1: "proposal" number x in [0,10]
  - player 2: "reject threshold" number y in [0,10]
- equilibria
  - NE: any pair of strategies $x = y$
  - SPNE: any pair of strategies $x = y$ = smallest non-zero number or 0

Lecture 10:
Games in Extensive Form

Jan Zouhar

Games and Decisions