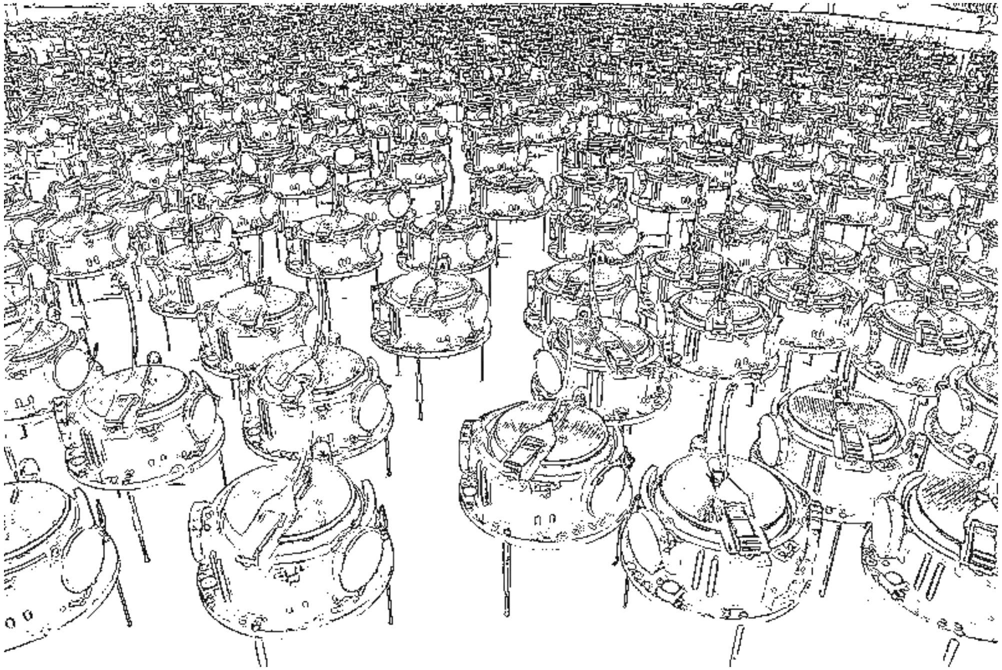
Intelligent Multi Agent Systems



Games in Extensive Form and Bayesian Games

Gerhard Neumann

Slides edited from Tom Lanaerts

Agenda

What we will do today…

- Many games are not in the normal form
- Sequential (non-simultaneous decisions)
- Chance moves
- Imperfect information

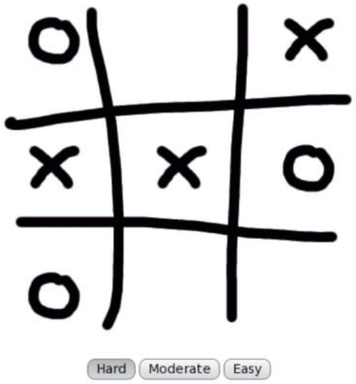
We will there introduce the extensive form and discuss Bayesian games for the imperfect information case

Extensive Form Games

Strategic games assume that each decision maker chooses her actions once and for all

Hence it does not take into account the sequential structure of decision making

Examples:



➡ Often, decision are made sequentially, not simultanously

Example

In an **Entry game** there are two players:

- A (the incumbent) and B (the challenger)
- B may decide to challenge (or to stay out)
- After B challenges A may either allow entry or fight against entry

Some terminology

A **history** is the sequence of actions taken by the players up to some decision point

A **terminal history** is a history that contains the action choices of all the players up until the point where the payoff is distributed

The Entry game has 3 **terminal histories**

1.(Challenge, Allow entry),
2.(Challenge, Fight entry) and
3.(Stay out)

A **sub-history** is a history that contains part of a terminal history

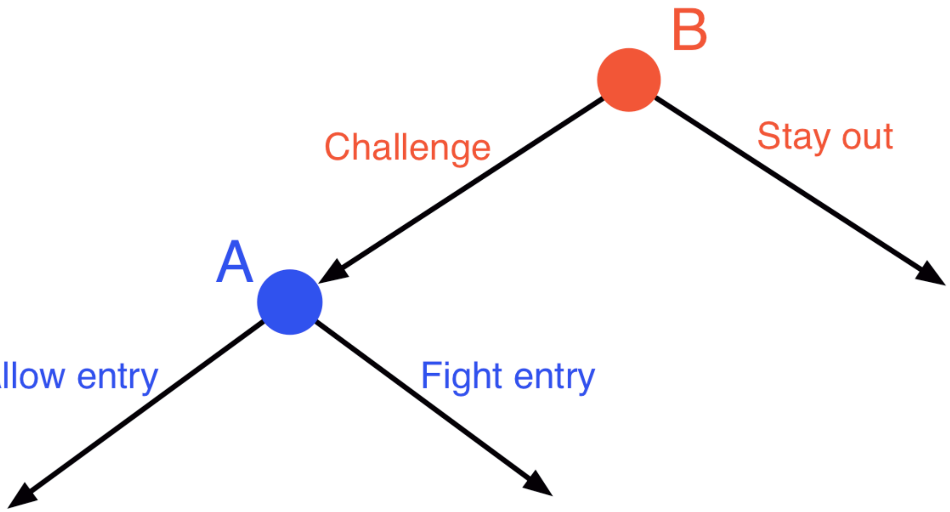
Histories

One can assign to every sub-history which is not the complete history *(= proper sub-history)* a player using a **player function** *P(proper sub-history):*

- (Challenge) and Ø are proper sub-histories of (Challenge, Allow entry) and (Challenge, Fight entry)
- *P(Challenge)* indicates that player A (the incumbent) acts after that point
- Thus *P(Ø)* indicates that player B (the challenger) acts after that point (which is the start of the game)

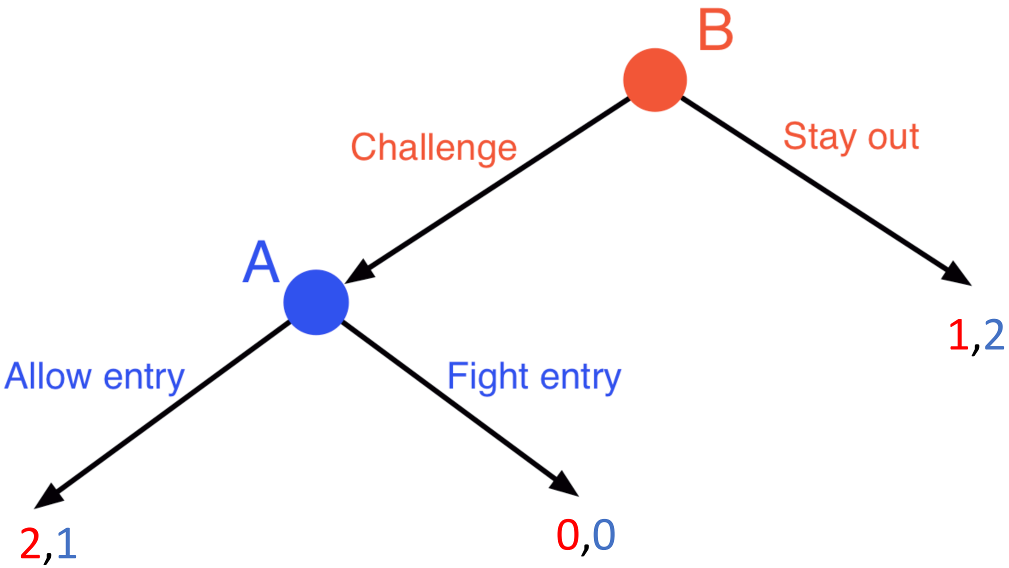
Tree Representation

An extensive form game is typically **represented as tree**



Tree Representation

An extensive form game is typically **represented as tree**



- **B** prefers (Challenge, Allow entry) over (Stay out) over (Challenge, Fight entry)
- **A** prefers (Stay out) over (Challenge, Allow entry) over (Challenge, Fight entry)

Definition

An extensive game with perfect information consists of:

- a set of players
- a set of terminal histories with the property that none of these histories is a proper sub-history of another
- A player function P(h) that assigns a player to every proper sub-history that can be derived from the terminal histories
- For each player, preferences over the set of terminal histories

Perfect Information

What:

- ➡Players know the node they are in
- ➡They know all the prior choices, including those of other agents

What happens when agents have only incomplete knowledge of the actions taken by others or no longer remember their past actions?

- ➡Games with *Imperfect Information* (see later)

More terminology

If all terminal histories are **finite**, then the game has a **finite horizon**

If a game has a finite horizon and finitely many terminal histories then the game is called **finite**

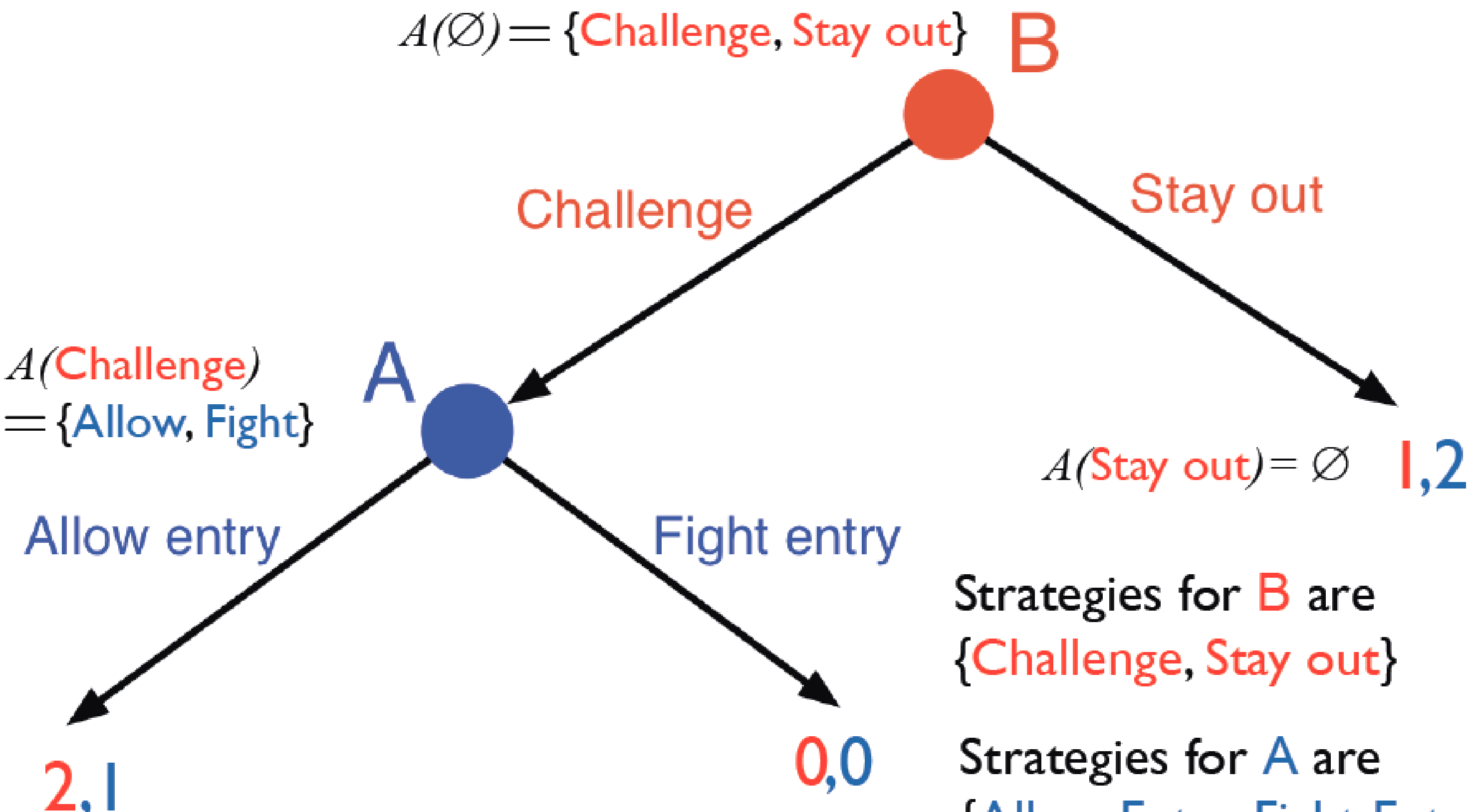
Definition :

A **strategy** of a player *i* in an extensive game with perfect information is a function that assigns to **each history *h*** after which it is player *i*'s turn to move (*P(h)=i,* where *P* is the player function) an action in *A(h)*, i.e., the set of available actions after *h*

Strategies

Action sets for different histories:

$A(\varnothing) = \{\text{Challenge}, \text{Stay out}\}$



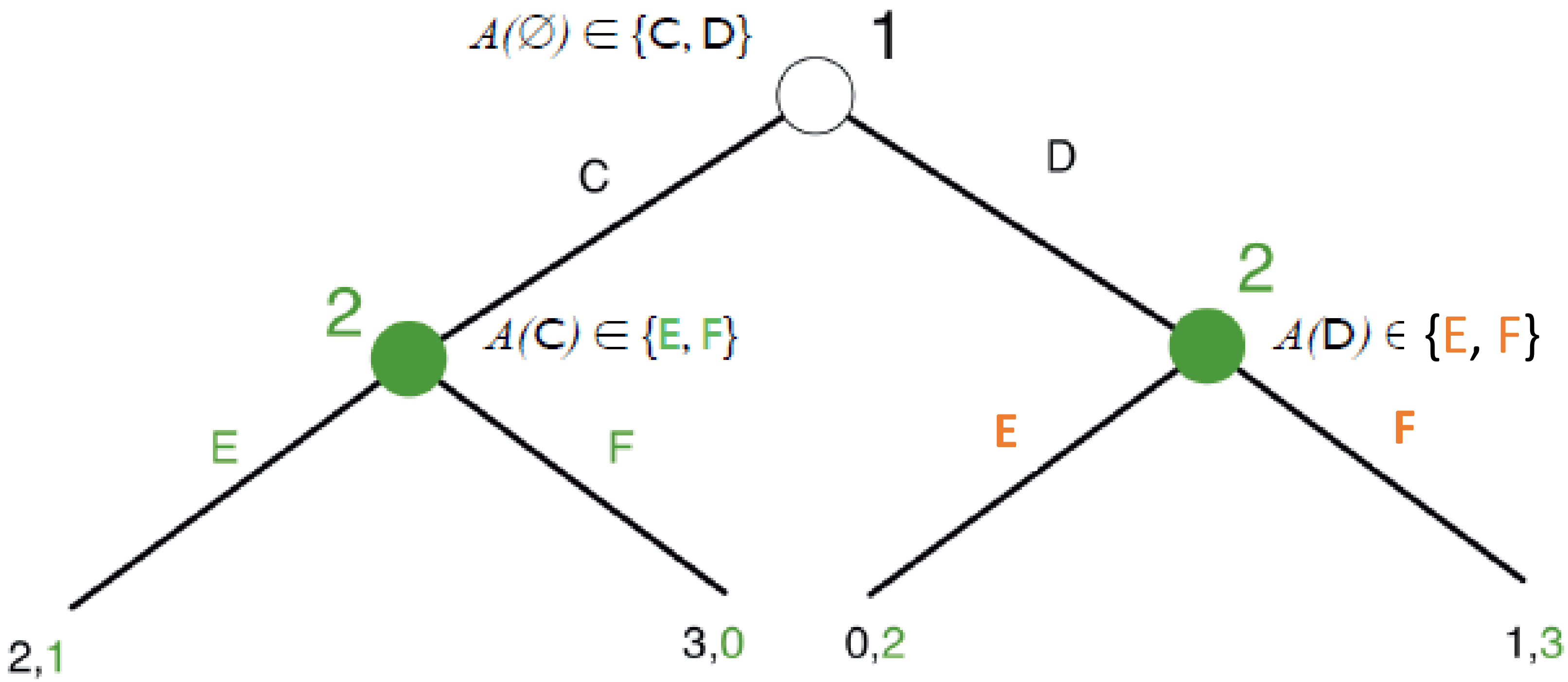
$A(\text{Challenge}) = \{\text{Allow}, \text{Fight}\}$

$A(\text{Stay out}) = \varnothing$

Strategies for B are {Challenge, Stay out}

Strategies for A are {Allow Entry, Fight Entry}

Strategies



Strategies for 1 are {C, D}

Strategies for 2 are {EE, EF, FE, FF}

In general, the number of strategies can increase exponentially with the depth of the tree

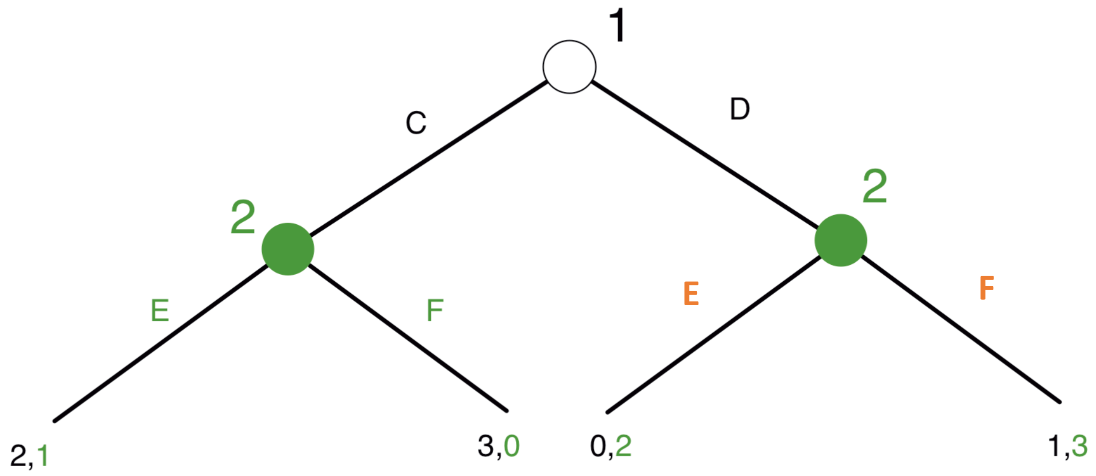
Pure Strategies

Overall, a pure strategy for a player in a perfect-information game is a complete specification of which deterministic action to take at every node belonging to that player.

- Strategies for 1 are {C, D}
- Strategies for 2 are {EE, EF, FE, FF}

We can always convert an extensive form game into a normal form game

	(E,E)	(E,F)	(F,E)	(F,F)
C	**2,1**	**2,1**	**3,0**	**3,0**
D	**0,2**	**0,2**	**1,3**	**1,3**



Finding an Equilibrium

Given our new definition of pure strategy, we are able to reuse our old definitions of:

- Mixed Strategies
- Best Response
- Nash Equilibrium Theorem

Nash Equilibrium

Theorem:

Every perfect information game in extensive form has a pure strategy Nash Equilibrium.

This is easy to see, since the players move sequentially.

Finding an Equilibrium

We can always convert an extensive form game into a normal form game

Example: Entry Game

	Allow	Fight
Challenge	1 2	0 0
Stay out	2 1	2 1

Finding an Equilibrium

We can always convert an extensive form game into a normal form game

Example: Entry Game

	Allow	Fight
Challenge	2, 1	0, 0
Stay out	1, 2	1, 2

Finding an Equilibrium

We can always convert an extensive form game into a normal form game

Example: Entry Game

➡2 Nash Equilibria

	Allow	Fight
Challenge	2, 1	0, 0
Stay out	1, 2	1, 2

Finding an Equilibrium

- This NE ignores the sequential structure of the game
  - Player 1 “knows” that player 2 would choose the fight action
- Player 2 uses the fight action as a threat
- But how creditable is this threat?
  - Is it rational to fight once “challenge” is chosen?

	Allow	Fight
Challenge	1 2	0 0
Stay out	2 1	2 1

Equilibrium

- We need a new definition of a NE that considers the sequential structure
- To reach this new definition, we first need to define the notion of a sub-game
- The idea is that this equilibrium **requires each player's strategy to be optimal**, given the other players' strategies, not only at the start **but at every possible history**

Sub-Games

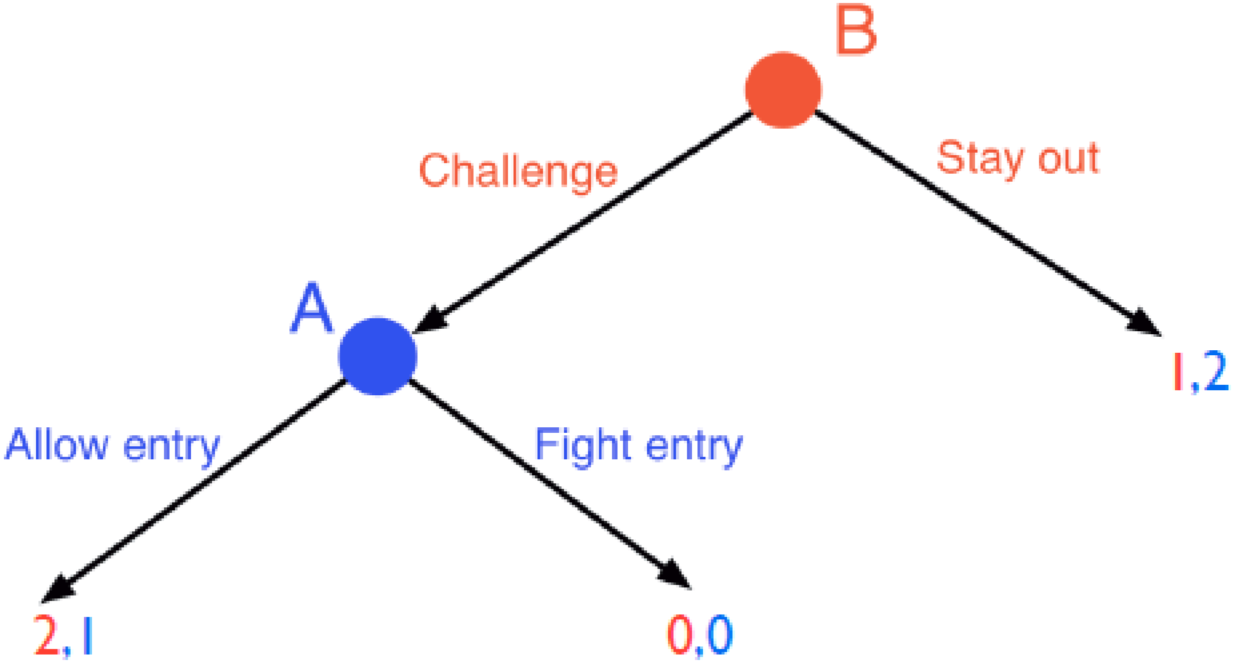
Definition :

Let *Γ* be an extensive game with perfect information, with player function *P*. For any nonterminal history *h* of *Γ*, the subgame *Γ(h)* following the history *h* is the following extensive game

- **Players** The players in *Γ*
- **Terminal histories** The set of all sequences *h'* of actions such that *(h,h')* is a terminal history of *Γ*
- **Player function** The player function *P(h,h')* is assigned to each proper sub-history *h'* of a terminal history
- **Preferences** each player prefers *h'* to *h''* if and only if she prefers *(h,h')* to *(h,h'') in Γ*

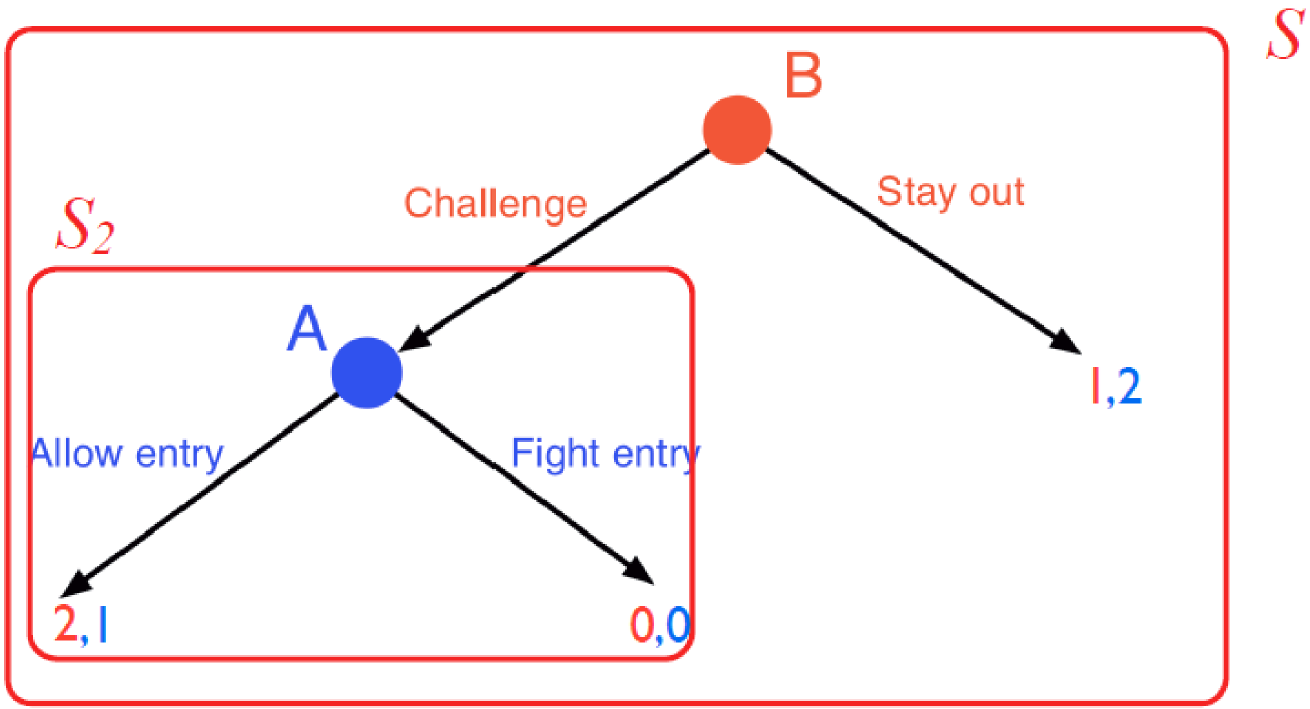
Sub-Games

The entry game has 2 sub-games



Sub-Games

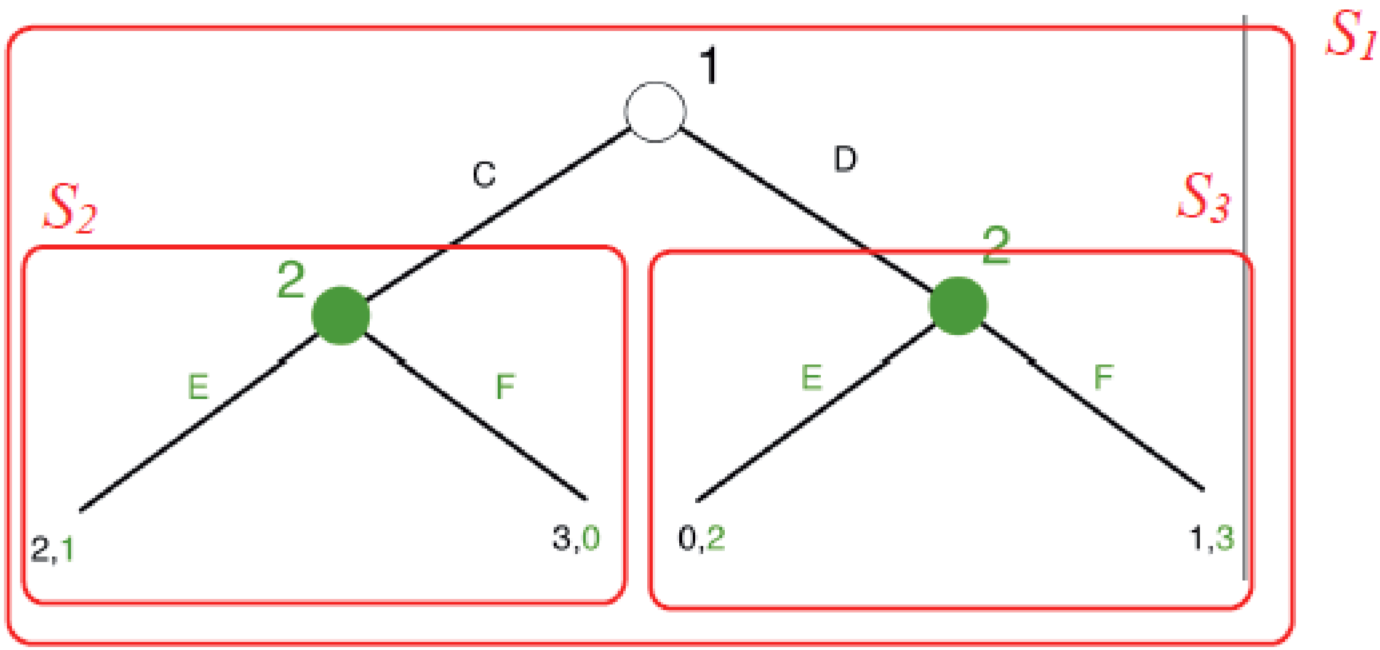
The entry game has 2 sub-games



S_2 is a proper sub-game

Sub-Games

The game below has 3 sub-games



S_2 and S_3 are proper sub-games

Sub-game perfect equilibrium

Definition:

The strategy profile $s*$ in an extensive game with perfect information is a **sub-game perfect equilibrium (SPE),** if for every player i and every history h after which it is player i's turn to move,

$u_i(O_h(s^*)) \geq u_i(O_h(r_i, s_{-i}^*))$ for every strategy r_i of player i,

where u_i is a payoff function that represents the player i's preferences and $O_h(s)$ is the terminal history consisting of h followed by the sequence of actions generated by s after h

Every sub-game perfect equilibrium is a Nash equilibrium

Finding an Equilibrium

Best response analysis claims there are 2 Nash Equilibria

➡ Are they also sub-game perfect equilibria?

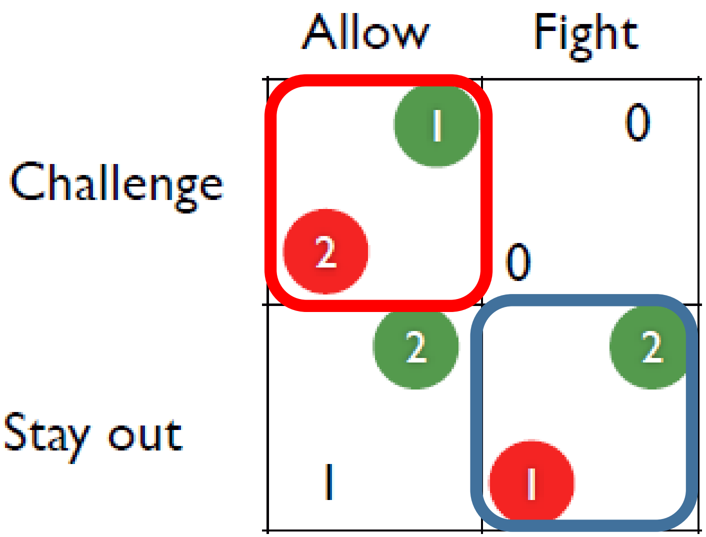
Take the NE s^*=(stay out,fight): ***Sub-game S2***

Player A (i=A) moves at h=challenge→$O_h(s^*)$ =(challenge, allow)

So now we check the payoffs for every action r_A of A, given $O_h(s^*)$

- ➡ r_A^*=allow → $u_A(O_h(s^*)) = 1$
- ➡ r_A=fight → $u_A(O_h(r_A, s_{-A}^*)) = 0$

So (challenge,allow) is a subgame perfect equilibirum



Finding an Equilibrium

Best response analysis claims there are 2 Nash Equilibria

➡ Are they also sub-game perfect equilibria?

Take the NE *s=(Stay out,fight):** ***Sub-game S1***

Player B (i=B) moves at $h=\emptyset \rightarrow O_h(s^*)$ =(Stay out, fight)

So now we check the payoffs for every action r_B of B, given $O_h(s^*)$

- ➡ $r_B^* = \text{stay out} \rightarrow u_B(O_h(s^*)) = 1$
- ➡ $r_B = \text{challenge} \rightarrow u_B(O_h(r_B, s_{-B}^*)) = 0$

	Allow	Fight
Challenge	2, 1	0, 0
Stay out	1, 2	1, 2

Finding an Equilibrium

Best response analysis claims there are 2 Nash Equilibria

➡ Are they also sub-game perfect equilibria?

Take the NE s^*=(Stay out,fight): ***Sub-game S2***

Player A (i=A) moves at ***h*=challenge**$\rightarrow O_h(s^*)$ =(**challenge**, fight)

So now we check the payoffs for every action r_A of A, given $O_h(s^*)$

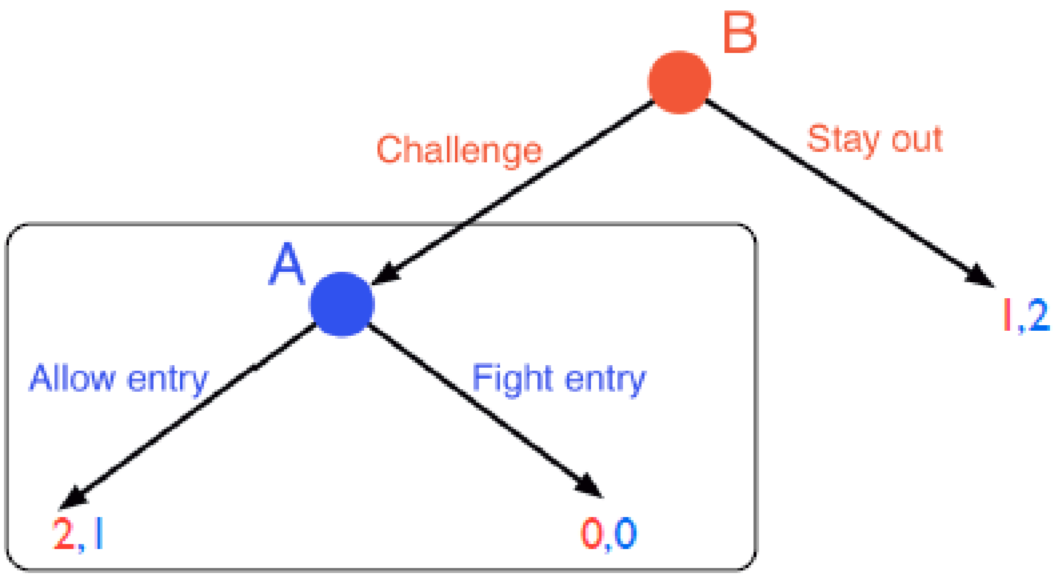
- ➡ r_A^*=fight $\rightarrow u_A(O_h(s^*)) = 0$
- ➡ r_A=allow $\rightarrow u_A(O_h(r_a, s_{-A}^*)) = 1$

So (stay out, fight) is a **NOT** subgame perfect equilibirum

	Allow	Fight
Challenge	2, 1	0, 0
Stay out	1, 2	1, 2

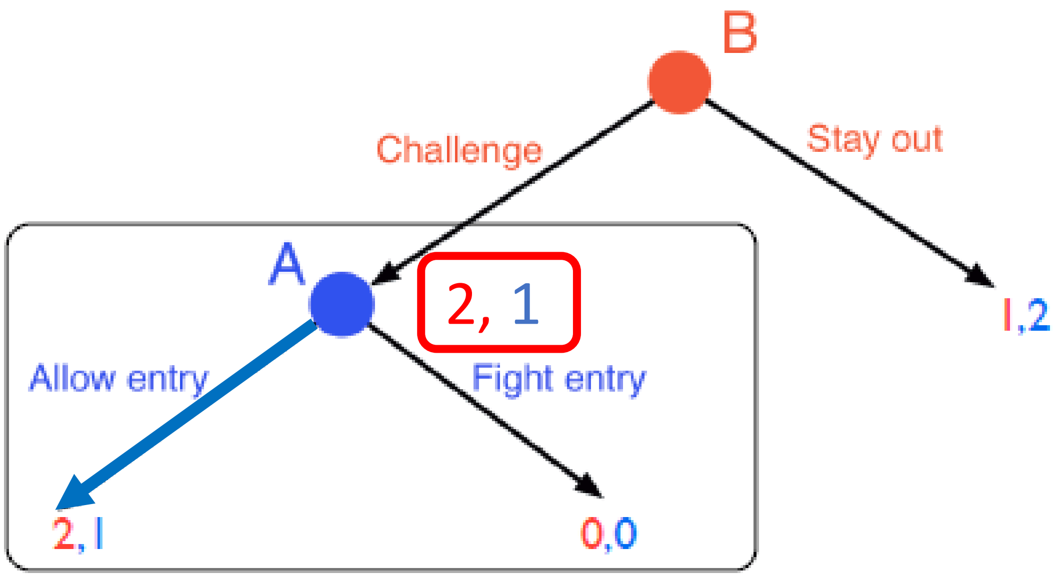
Finding the SPE

In a game with a finite horizon, the set of subgame perfect equilibria can be found more easily by using a procedure that extends the **backward induction** process:



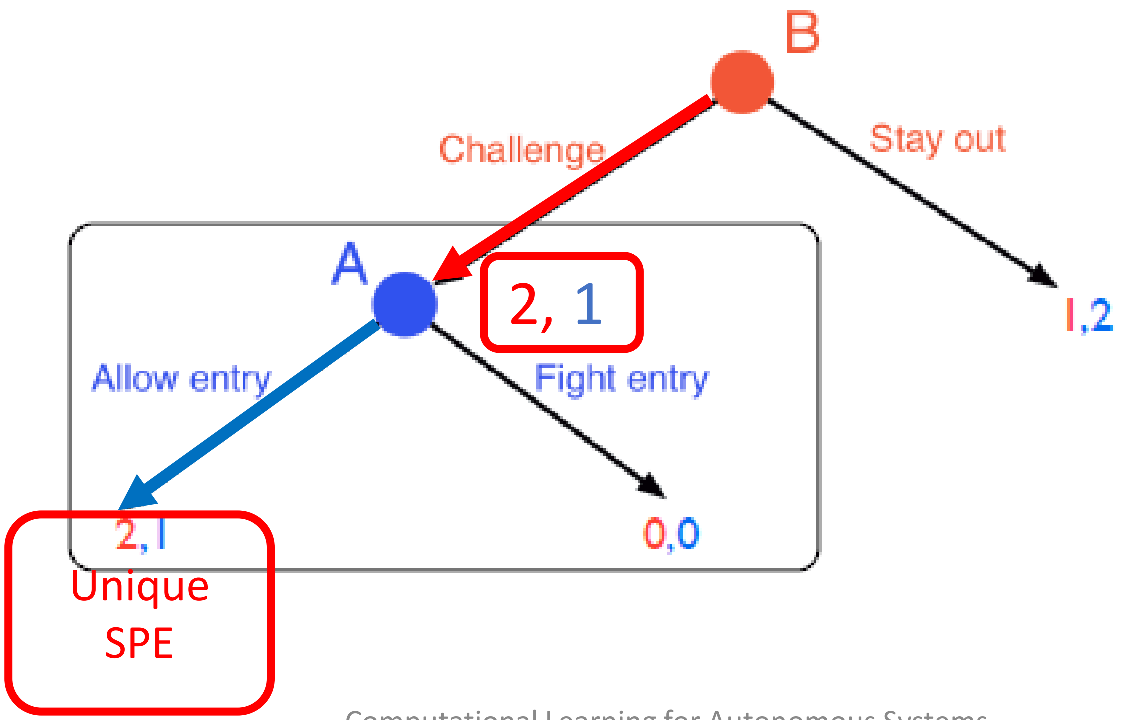
Finding the SPE

In a game with a finite horizon, the set of subgame perfect equilibria can be found more easily by using a procedure that extends the **backward induction** process:



Finding the SPE

In a game with a finite horizon, the set of subgame perfect equilibria can be found more easily by using a procedure that extends the **backward induction** process:



Backwards induction algorithm

Idea: Identify the equilibria in the bottom-most trees, and adopt these as one moves up the tree

function BACKWARDINDUCTION(node h) **returns** u(h)
if h ∈ Z **then**
    return u(h) // h is a terminal node
bestUtil ← −∞
forall a ∈ A(h) **do**
    utilAtChild ←BACKWARDINDUCTION(*σ(h, a)*)
    if $utilAtChild_{P(h)} > bestUtil_{P(h)}$ **then**
        bestUtil ← *utilAtChild*
return *bestUtil*

- *A(h)* is the action set for node h
- *P(h)* is the player function for node h
- *σ(h, a)* is the transition function

- For zero-sum games, BackwardInduction has another name: the minimax algorithm

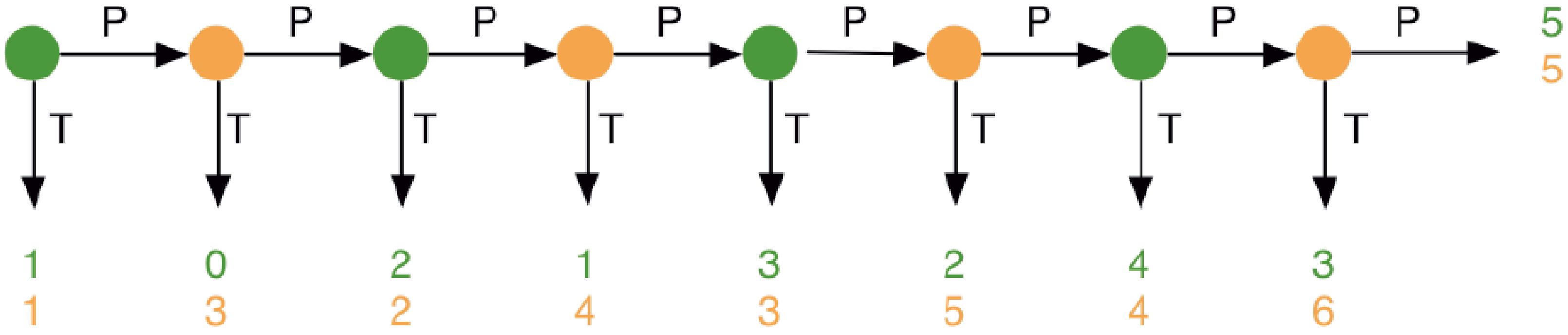
Backward Induction

2 Examples:

- Centripede Game
- 5 Pirates Game

The Centipede game

8 move centipede game: Originally invented by Rosenthal in 1982 (with 100 moves)

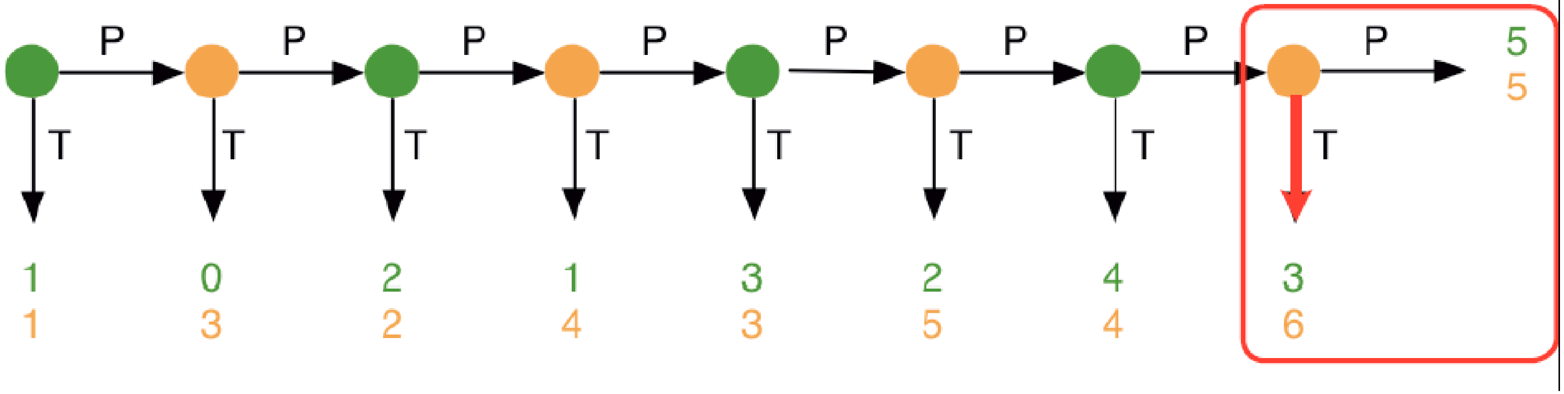


➡ Note that passing (P) the money always means that you may receive less than currently possible

➡ What is the Sub-game perfect Nash equilibrium here?

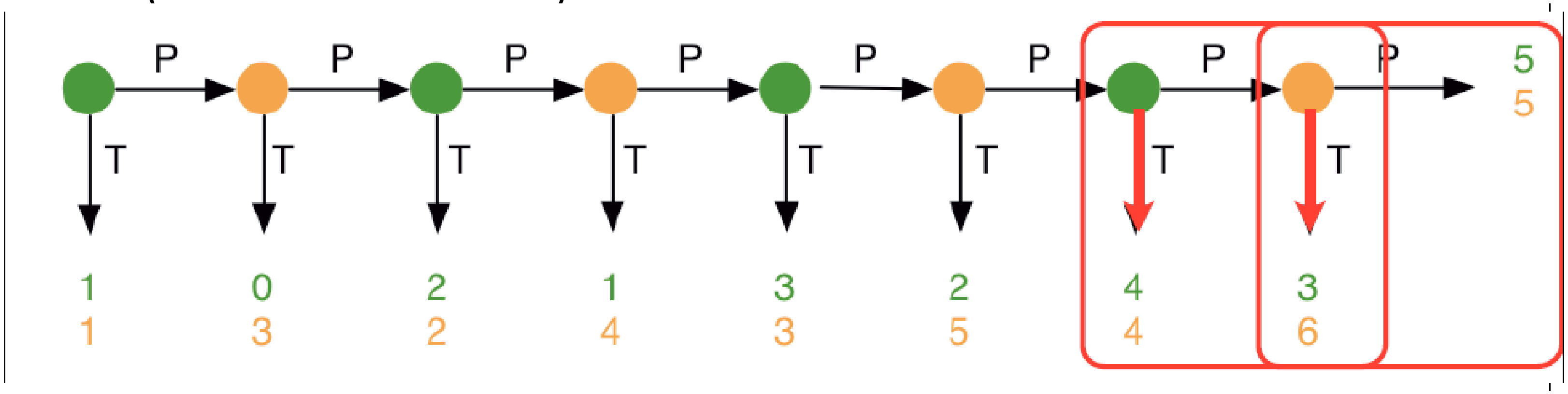
The Centipede game

8 move centipede game: Originally invented by Rosenthal in 1982 (with 100 moves)



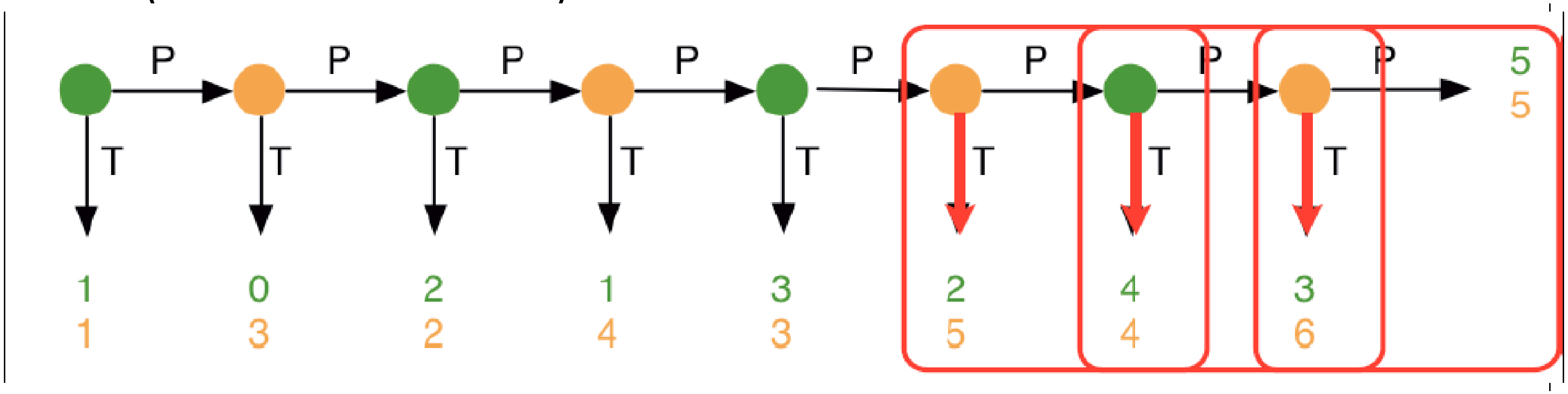
Example: The Centipede game

8 move centipede game: Originally invented by Rosenthal in 1982 (with 100 moves)



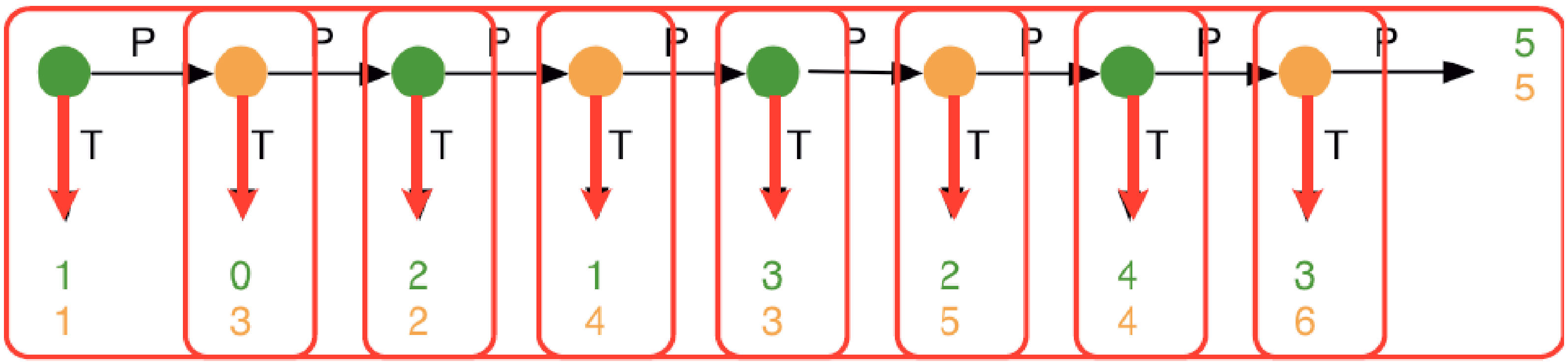
Example: The Centipede game

8 move centipede game: Originally invented by Rosenthal in 1982 (with 100 moves)



Example: The Centipede game

8 move centipede game: Originally invented by Rosenthal in 1982 (with 100 moves)



So the rational choice is to take immediately the money

- However, this outcome is Pareto-dominated by all but one other outcome.

Example: The Centipede game

But: Human subjects do not go down right away!

PROPORTION OF OBSERVATIONS AT EACH TERMINAL NODE

	Session		N	f_1	f_2	f_3	f_4	f_5	f_6	f_7
Four Move	1	(PCC)	100	.06	.26	.44	.20	.04		
	2	(PCC)	81	.10	.38	.40	.11	.01		
	3	(CIT)	100	.06	.43	.28	.14	.09		
	Total	1–3	281	.071	.356	.370	.153	.049		
High Payoff	4	(High-CIT)	100	.150	.370	.320	.110	.050		
Six Move	5	(CIT)	100	.02	.09	.39	.28	.20	.01	.01
	6	(PCC)	81	.00	.02	.04	.46	.35	.11	.02
	7	(PCC)	100	.00	.07	.14	.43	.23	.12	.01
	Total	5–7	281	.007	.064	.199	.384	.253	.078	.014

from McKelvey and Palfrey (1992) An Experimental Study of the Centipede Game. *Econometrica* 60(4):803-836

5 Pirates Puzzle

5 pirates of different ages have a treasure of 100 gold coins. On their ship, they decide to split the coins using this scheme:

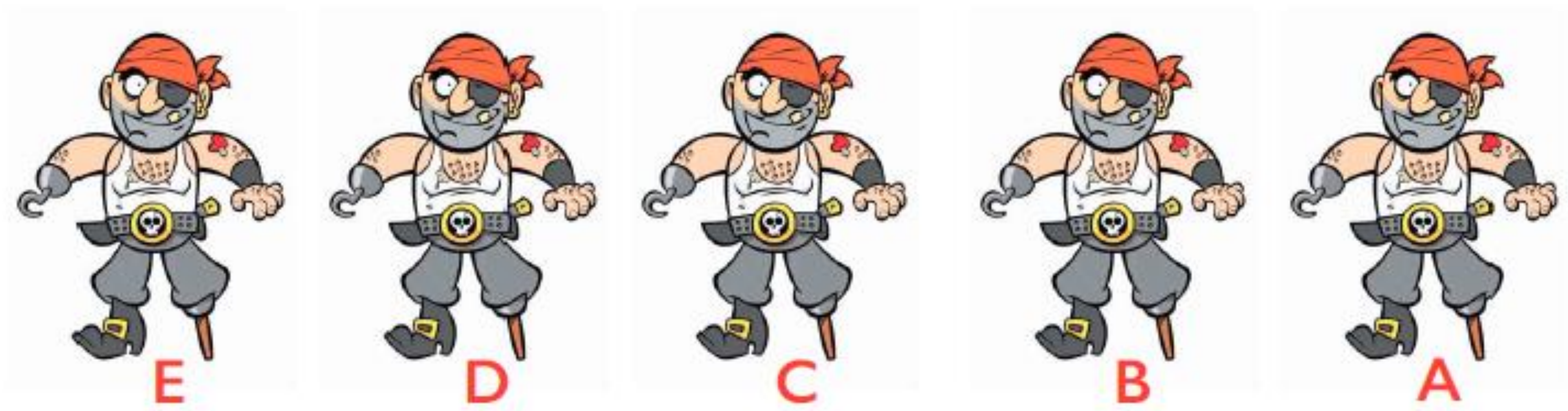
- The oldest pirate proposes how to share the coins, and ALL pirates (including the oldest) vote for or against it.
- If 50% or more of the pirates vote for it, then the coins will be shared that way. Otherwise, the pirate proposing the scheme will be thrown overboard, and the process is repeated with the pirates that remain.
- As pirates tend to be a bloodthirsty bunch, if a pirate would get the same number of coins if he voted for or against a proposal, he will vote against so that the pirate who proposed the plan will be thrown overboard.

Assuming that all 5 pirates are intelligent, rational, greedy, and do not wish to die, (and are rather good at math for pirates) what will happen?

5 Pirates Puzzle

What proposal should the oldest pirate make?

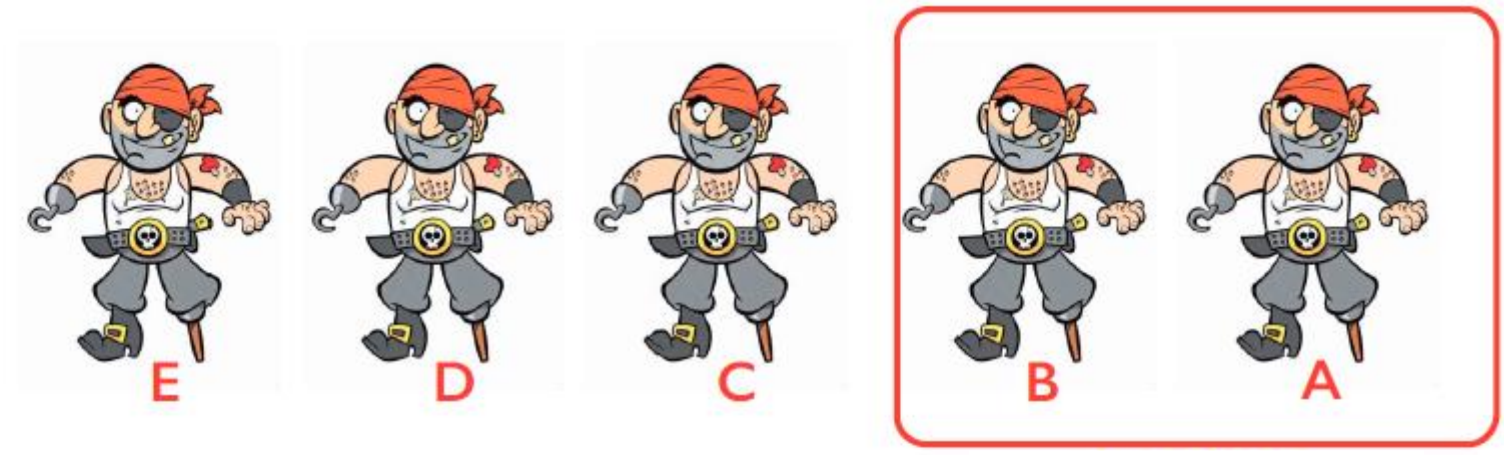
- Should he (or she) give away most of the loot?
- The problem can be solved by backwards induction



5 Pirates Puzzle

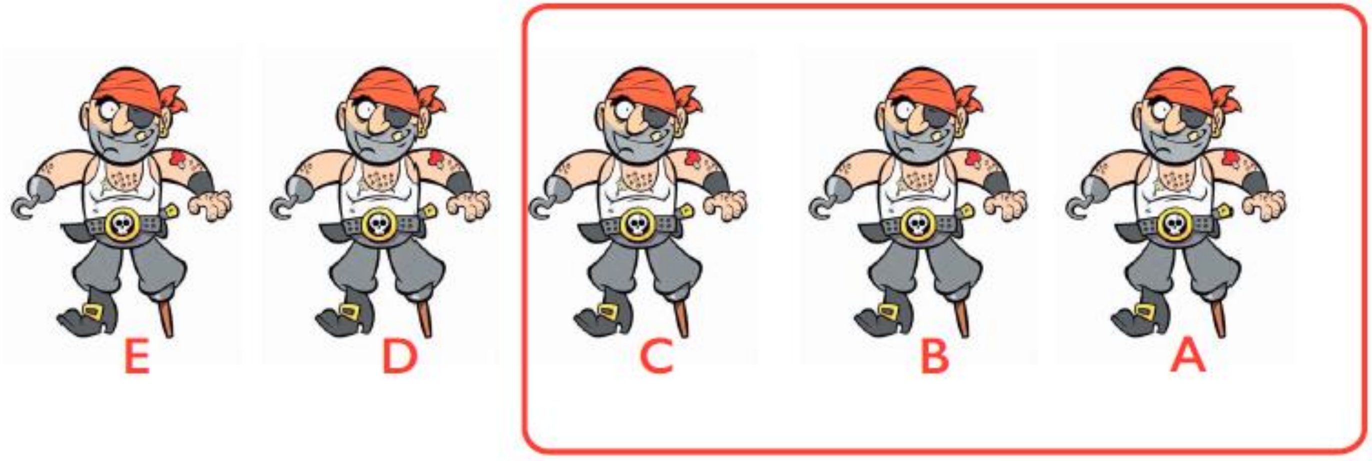
The problem can be solved by backwards induction

Let's start with game where only 2 pirates remain : **A** and **B**



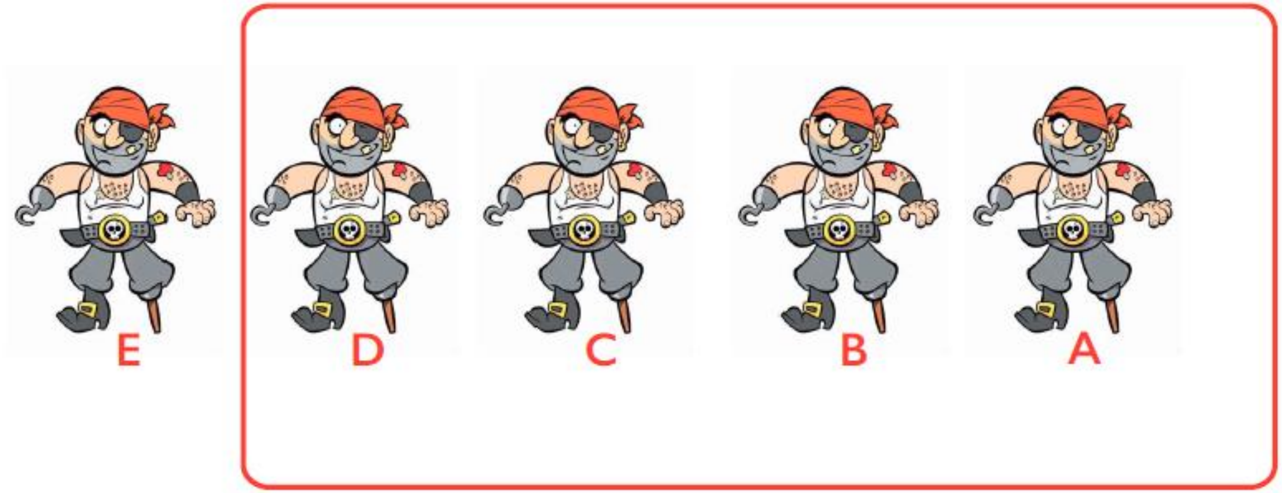
5 Pirates Puzzle

The problem can be solved by backwards induction



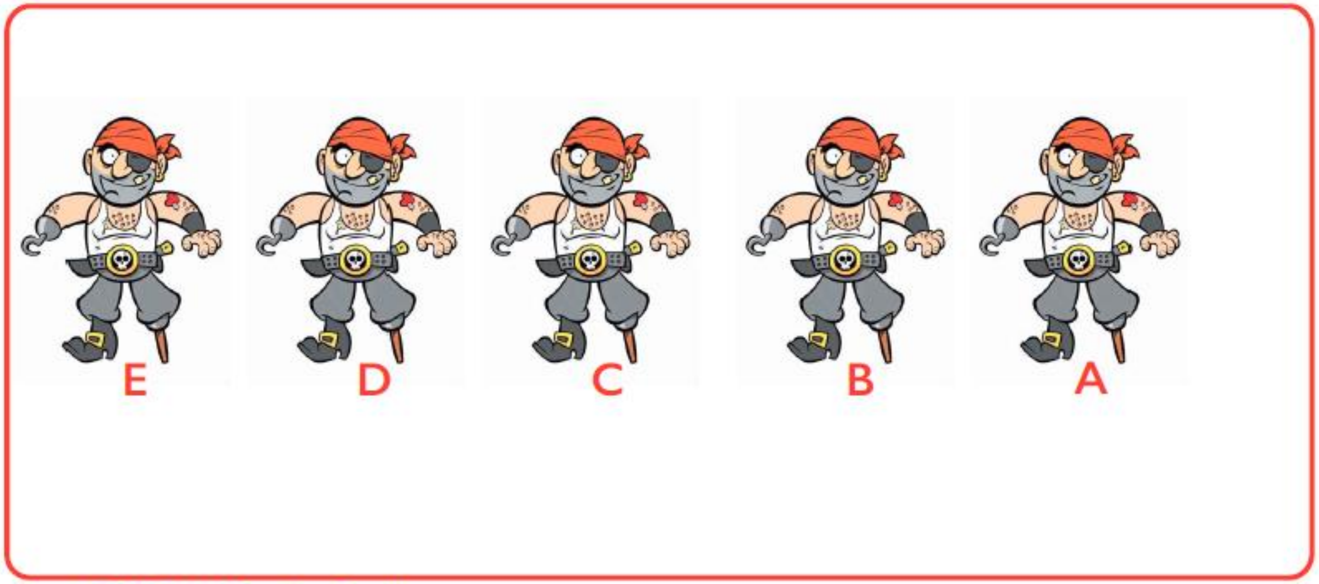
5 Pirates Puzzle

The problem can be solved by backwards induction



5 Pirates Puzzle

The problem can be solved by backwards induction



Extensions of Extensive Form Games

- **Simultaneous moves:** In some situations, after some sequence of actions of the players, the players may need to choose the next action simultaneously
- **Chance moves:** Sometimes, random events may occur that alter the sequence of actions
- **Bayesian games:** Sometimes you lack information about the opponent

Simultaneous moves

Take for instance the following variant of the battle of the sexes:

- First one player decides to stay home and watch television or to attend a concert.
- When he or she decided to stay home, the game ends
- If he or she decides to attend the concert, then both players have to choose simultaneously which concert, Bach or Stravinsky

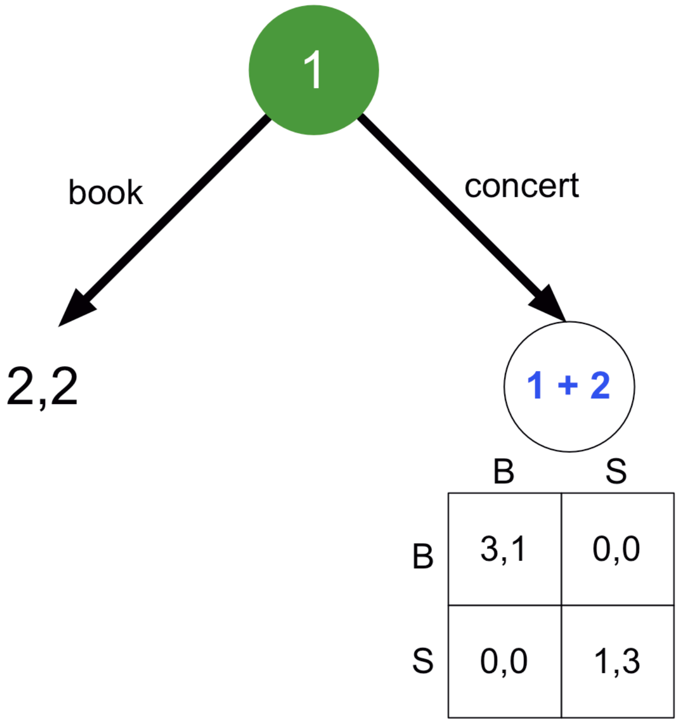
Simultaneous moves

This defines the following extensive form game:

- ➡a set of players: *{1,2}*
- ➡a set of terminal histories: *{Book, (Concert, (B,B)), (Concert,(B,S)), (Concert, (S,B)), (Concert, (S,S))}*
- ➡A player function: *P(Ø) = {1}, P(Concert)={1,2}*
- ➡Preferences:
  - Player 1: *(Concert,(B,B)) > Book > (Concert,(S,S)) > (Concert,(S,B)) = (Concert,(B,S))*
  - Player 2: *(Concert,(S,S)) > Book > (Concert,(B,B)) >(Concert,(S,B)) = (Concert,(B,S))*

Simultaneous moves

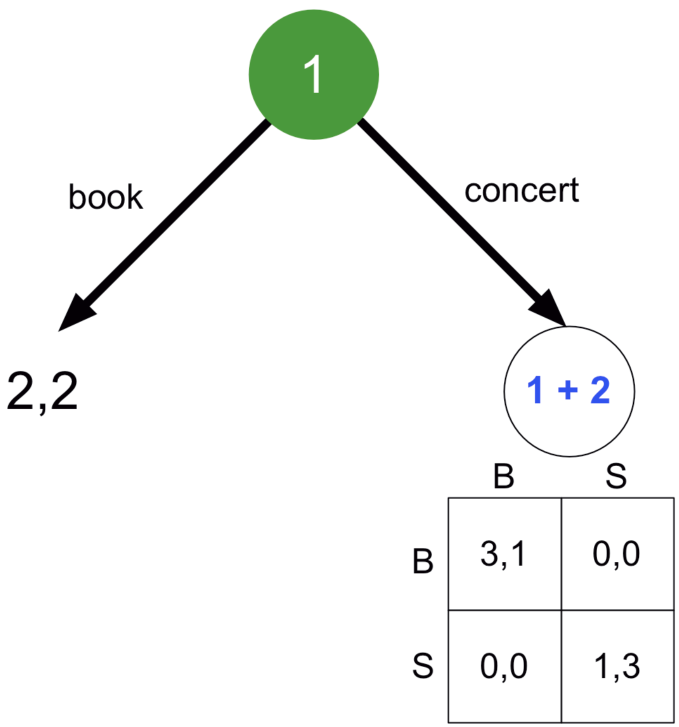
How to find the equilibrium?



- **Actions** of player 1:
  A1(Ø)={Concert, Book},
  A1(Concert)={B, S},
- **Actions** of player 2:
  A2(Concert)={B, S}
- **Strategies** player 1: *(Concert, B), (Concert, S), (Book, B)* and *(Book, S)*
- **Strategies** player 2: *B* and *S*

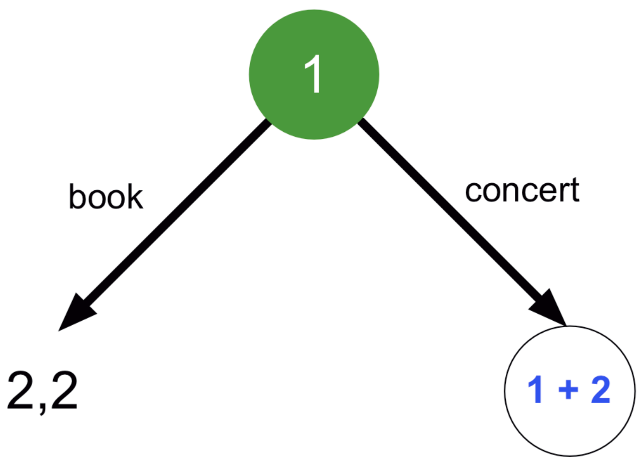
Simultaneous moves

How to find the equilibrium?



	B	*S*
(concert, B)	3,1	0,0
(concert, S)	0,0	1,3
(book, B)	2,2	2,2
(book, S)	2,2	2,2

Simultaneous moves



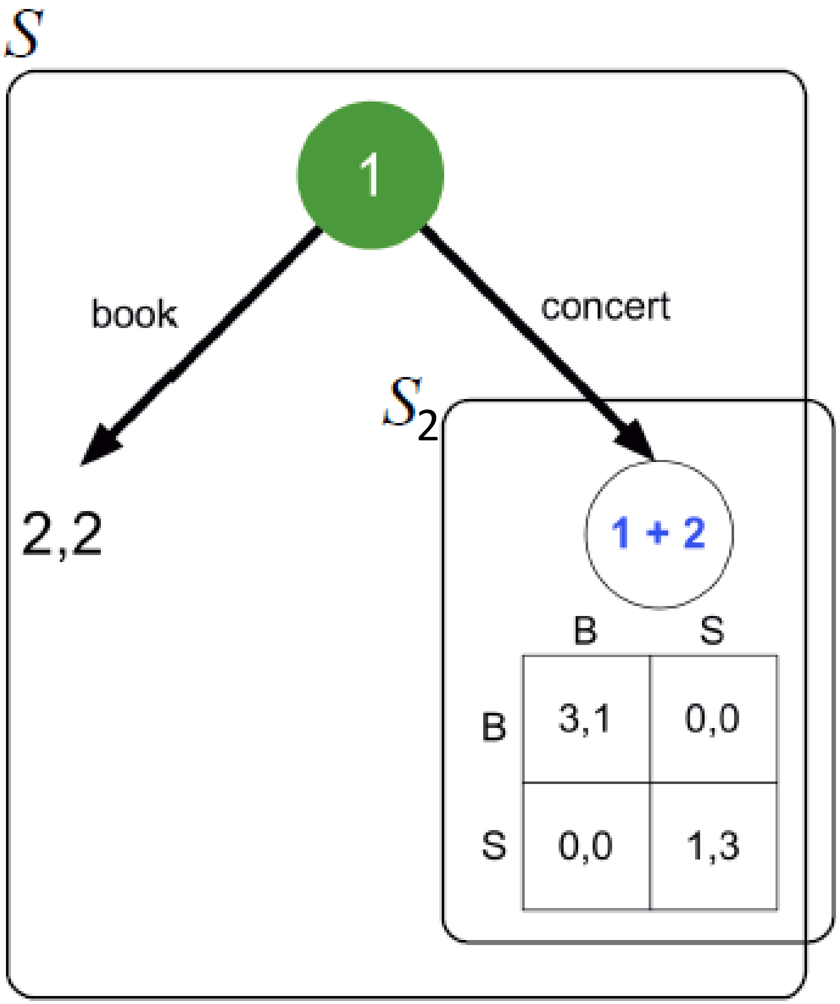
	B	S
B	3,1	0,0
S	0,0	1,3

How to find the equilibrium?

	B	S
$(concert, B)$	3,1	0,0
$(concert, S)$	0,0	1,3
$(book, B)$	2,2	2,2
$(book, S)$	2,2	2,2

Can we also find the sub-game perfect equilibria?

Simultaneous moves



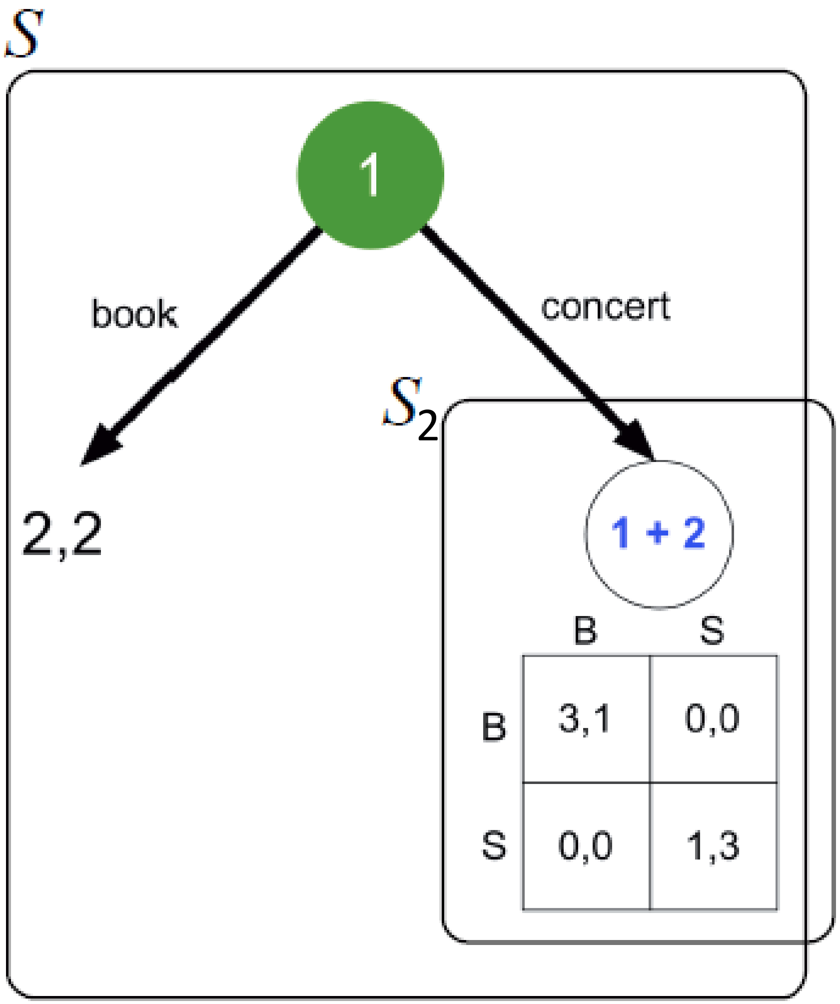
How to find the equilibrium?

1. What are the pure strategy equilibria for the sub-game S_2?

2. What is the optimal choice in S_1 if the outcome of S_2 is (B,B)?

3. What is the optimal choice in S_1 if the outcome of S_2 is (S,S)?

Simultaneous moves



How to find the equilibrium?

1. What are the pure strategy equilibria for the sub-game S_2?

   (B,B) and (S,S)

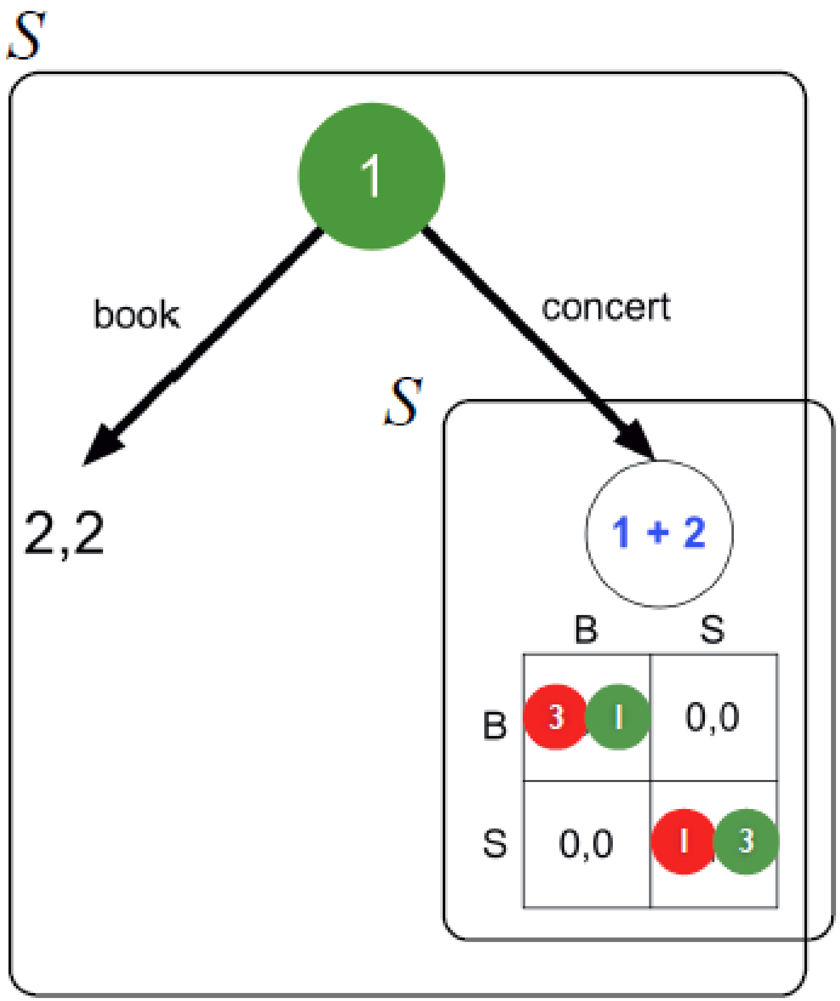
2. What is the optimal choice in S_1 if the outcome of S_2 is (B,B)?

   concert

3. What is the optimal choice in S_1 if the outcome of S_2 is (S,S)?

   book

Simultaneous moves



Thus, the sub-game perfect equilibria are :

- **((concert,B),B)**
- **((book,S),S)**

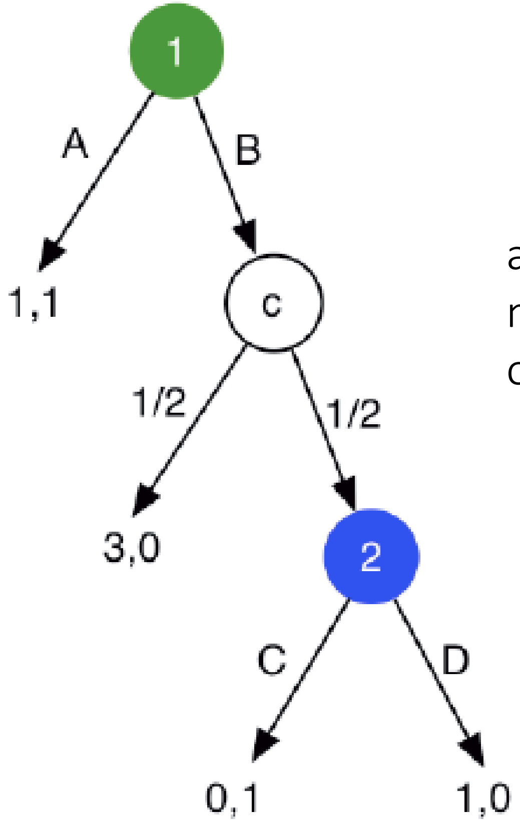
Neither **((book,B),S)** nor **((book,B),B)** are sub-game perfect. Do you see why?

Note that we ignored the Mixed Strategy Equilibrium of BoS here…

Chance Moves

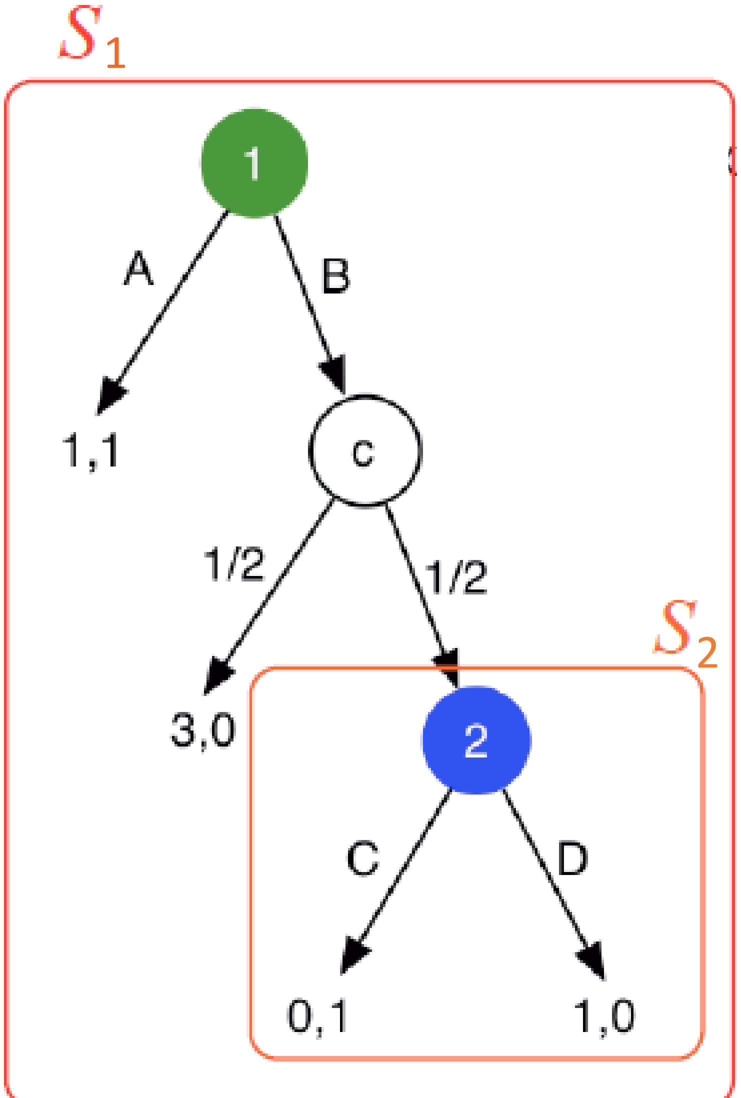
- In the definition of extensive-form games, there is a function P that assigns a player to each history.
- Here, one can also assign chance as opposed to a player
- As a consequence, the preferences of the players become defined over the set of lotteries (probability distribution) over terminal histories

Chance Moves



after the move of player 1, chance may direct you to a terminal history or to a decision of player 2

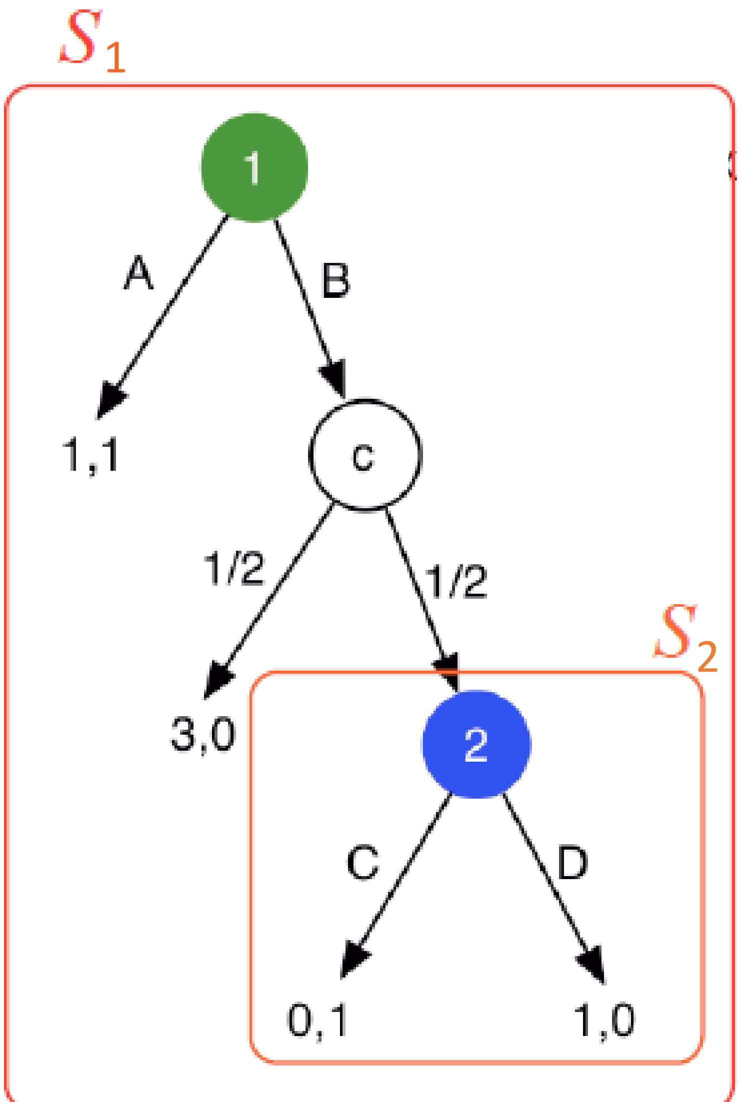
Chance Moves



Backward induction:

- In sub-game S_2, player 2 chooses C
- In sub-game S_1, we consider the consequences of the actions of player 1

Chance Moves



Backward induction:

- In sub-game S_2, player 2 chooses C
- In sub-game S_1, we consider the consequences of the actions of player 1
  - If A is selected then 1 obtains a payoff of 1
  - If B is selected then 1 obtains a payoff of 3 with probability 1/2 and a payoff of 0 with probability 1/2!
  - Hence an expected payoff of 3/2

Thus the SPE is here is (B, C)

Imperfect information

- Often players do not know the preferences of their opponents
- or they may not know how well the opponent knows their preferences
- **Bayesian games** allow us to analyze any situation in which a player is not completely informed about an environmental aspect that may be relevant for her choice of action
- Lets start with pure strategies

Bayesian Game

Definition:

A Bayesian game is a tuple (N, A, Θ, p, u)

- where N is a set of agents,
- $A = (A_1, \ldots, A_n)$, where A_i is the set of actions available to player *i*,
- $\Theta = (\Theta_1, \ldots, \Theta_n)$, where Θ_i is the type space of player *I*
  - The type specifies the utility function
- $p : \Theta \to [0, 1]$ is the common prior over types,
- $u = (u_1, \ldots, u_n)$, where $u_i : A \times \Theta \to \mathbb{R}$ is the utility function for player *i*.
- Each player knows his own type

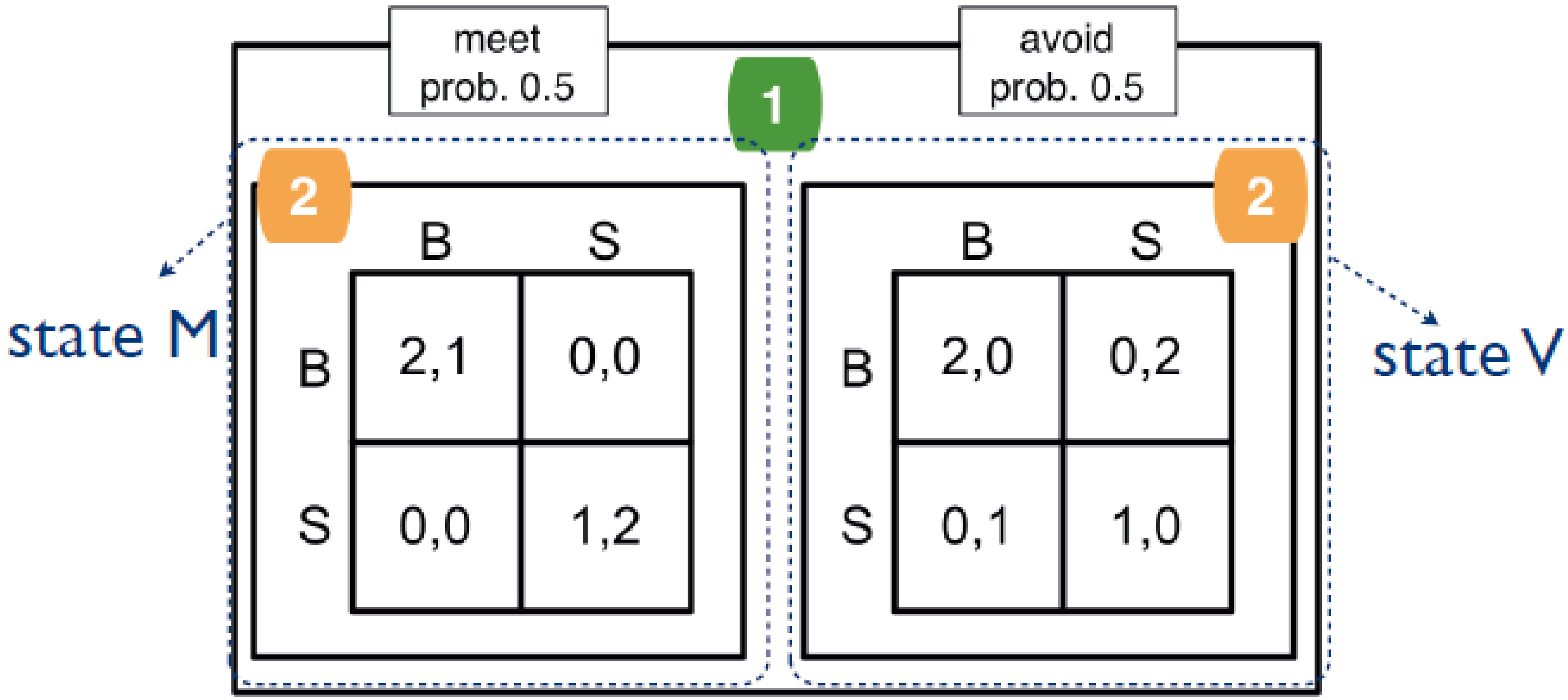
Example: Bayesian Game

Consider another variant of the **battle of the sexes:**

- player 1 is unsure whether player 2 prefers to go out with her or prefers to avoid her.
- Let's assume that there is equal chance for both (which can be based on player 1's personal assessment)
- So player 1 beliefs that with probability 1/2 she plays two different games
- **Player 2 knows which of the two games is being played.**

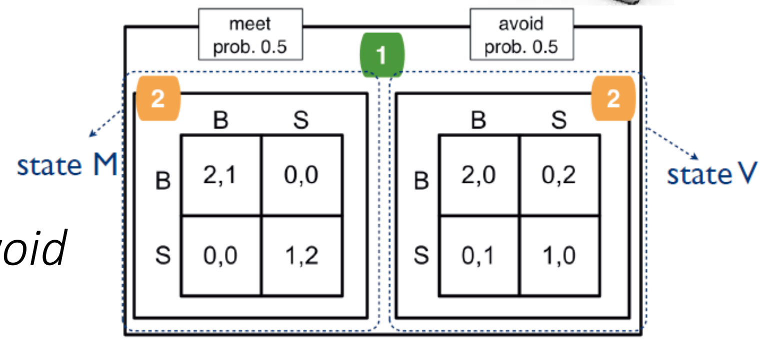
Example: Bayesian Game

There are 2 potential games (= types for player 2):



What are the equilibria in this kind of game?

Example: Bayesian Game



Players: 1 and 2

Actions: for each player {B,S}

Types: Player 1: *meet*, Player 2: *meet* and *avoid*

Priors:

$$p(\theta_2 = meet) = p(\theta_2 = avoid) = 0.5$$

Payoffs:

- the payoff $u_i(a, meet)$ for each player are given by the first matrix in the figure and the
- payoff $u_i(a, avoid)$ for each player are given by the second matrix in the same figure

Strategies for Bayesian Games

Pure strategy: $a_i(\theta_i) = (a_i^1, \ldots, a_i^j) \in A_i^{|\Theta_i|}$

a vector that contains an action for every possible type θ_i^j of agent *i*

We can again convert a Bayesian game to a game in normal form over the set of pure strategies of the players

Mixed strategy: $\pi_i : \Theta_i \times A_i \rightarrow [0, 1]$

$\pi_i(a_i^j | \theta_i)$ the probability that agent j plays action a_j, given that *j*'s type is *j*.

Expected Utilities

We can again convert a Bayesian game to a game in normal form over the set of pure strategies of the players

➡ **But how to compute the utilities?**

Three meaningful notions of expected utility:

- ex-ante: the agent knows nothing about anyone's actual type
- ex-interim: an agent knows his own type but not the types of the other agents
- ex-post: the agent knows all agents' types.

Ex-post expected utility

Definition: Agent i's ex-post expected utility in a Bayesian game (N, A, Θ, p, u), where i's type is θ_i and where the agents' strategies are given by the mixed strategy profile π, is defined as

$$U_i^P(\pi, \theta) = \sum_{a \in A} \left(\prod_{j \in N} \pi_j(a_j | \theta_j) \right) u_i(a, \theta).$$

Vector of all types of all agents

- Agent i must weight the utility value by the probability $p(a|\theta) = \prod_j \pi_j(a_j|\theta_j)$ that the joint action a would be realized given all players' mixed strategies and types

The type of the agents are known to everybody

Ex-interim expected utility

Definition: Agent i's ex-interim expected utility in a Bayesian game(N, A, Θ, p, u), where i's type is θ_i and where the agents' strategies are given by the mixed strategy profile π, is defined as

$$U_i^I(\pi|\theta_i) = \sum_{\theta_{-i}\in\Theta_{-i}} p(\theta_{-i}|\theta_i) U_i^P(\pi, (\theta_{-i}, \theta_i)).$$

Vector of all types of all agents except agent i

or, equivalently,

$$U_i^I(\pi|\theta_i) = \sum_{\theta_{-i}\in\Theta_{-i}} p(\theta_{-i}|\theta_i) \sum_{a\in A} \left(\prod_{j\in N} \pi_j(a_j|\theta_j) \right) u_i(a, \theta_{-i}, \theta_i).$$

➡ Agent i must consider every θ_{-i} and weight the ex-post utility value by the probability that the other players' types would be θ_{-i} given that his own type is θ_i.

Ex-ante expected utility

Definition (Ex-ante expected utility):

Agent i's ex-ante expected utility in a Bayesian game (N, A, Θ, p, u), where i's type is θ_i and where the agents' strategies are given by the mixed strategy profile π, is defined as

$$U_i^A(\pi) = \sum_{\theta_i \in \Theta_i} p(\theta_i) U_i^I(\pi|\theta_i).$$

Or equivalently

$$U_i^A(\pi) = \sum_{\theta} p(\theta) \sum_{a \in A} \left(\prod_{j \in N} \pi_j(a_j|\theta_j) \right) u_i(a, \theta_{-i}, \theta_i).$$

NE in a Bayesian Game

Definition (Best response in a Bayesian game)

The set of agent i's best responses to mixed strategy profile π_{-i} are given by

$$\mathrm{BR}_i(\pi_{-i}) = \arg\max_{\pi'_i \in \Pi_i} U_i^A(\pi'_i, \pi_{-i}).$$

- it may seem odd that *BR* is calculated based on *i*'s ex-ante utility.
- **However:** $U^A(\pi'_i, \pi_{-i}) = \sum_{\theta_i \in \Theta_i} p(\theta_i) U_i^I(\pi'_i, \pi_{-i} | \theta_i)$
  $U_i^I(\pi'_i, \pi_{-i} | \theta_i)$ does not depend on strategies for *i* where *i* is using a different type θ'_i
- Thus, we are in fact performing independent maximization of i's ex-interim expected utility conditioned on each type that he could have.

NE in a Bayesian Game

- Thus, we are in fact performing independent maximization of i's ex-interim expected utility conditioned on each type that he could have.

Alternative Definition (Best response in a Bayesian game) π_{-i}

A mixed strategy π_i is a best response to mixed strategy profile if

$$\pi(\cdot|\theta_i^j) \in \arg \max_{\pi_i' \in \Pi_i(\theta_i^j)} U_i^I(\pi_i', \pi_{-i}|\theta_i^j)$$

$$= \arg \max_{\pi_i' \in \Pi_i(\theta_i^j)} \sum_{a_i^k} \pi_i(a_i^k|\theta_i^j) U_i^I(a_i^k, \pi_{-i}|\theta_i^j)$$

for each $\theta_i^j \in \Theta_i$.

NE in a Bayesian Game

Simplified Definition for pure strategies and 2 players

A pure strategy $a_i(\theta_i) = (a_i^1, \ldots, a_i^m)$ is a best response to pure strategy profile $a_{-i}(\theta_{-i})$ if

$$
\begin{aligned}
a_i^j &\in \arg\max_{a_i^{j'} \in A_i} U_i^I(a_i^{j'}, a_{-i}(\theta_{-i}) | \theta_i^j), \quad \forall j = 1 \ldots m \\
&= \arg\max_{a_i^{j'} \in A_i} \sum_{k=1}^{m} p(\theta_{-i}^k | \theta_i^j) u_i(a_i^{j'}, a_{-i}^k | \theta_i^j, \theta_{-i}^k), \quad \forall \theta_i^j \in \Theta_i
\end{aligned}
$$

➡ For each type θ_i^j , we need to find the best action a_i^j, where we average over all possible types θ_{-i}^k of the other player (using actions a_{-i}^k)

Nash Equilibrium

Definition (Bayes-Nash Equilibrium):

A (mixed strategy) Bayes-Nash equilibrium is a mixed strategy profile $\pi = (\pi_1, \ldots, \pi_n)$ that satisfies:

$$\forall i \quad \pi_i \in \mathrm{BR}_i(\pi_{-i})$$

- ➡ Same definition as for standard NE
- ➡ we can construct an induced normal form for Bayesian games (similar to extensive form)
- ➡ the numbers in the cells correspond to ex-ante expected utilities of all different action combinations

Imperfect Information

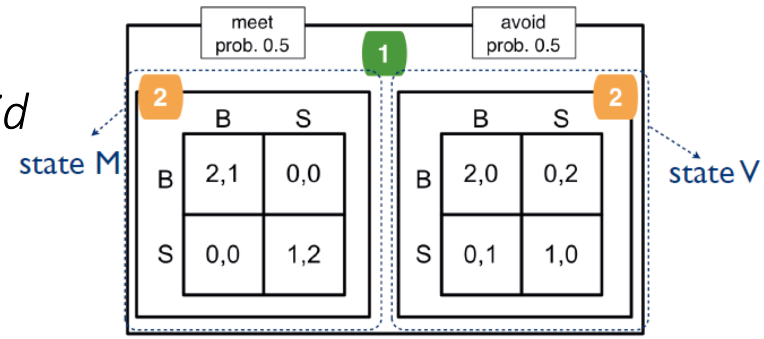
Players: 1 and 2

Actions: for each player {B,S}

Types: Player 1: *meet*, Player 2: *meet* and *avoid*

Strategies: Player 1: **{B,S},**

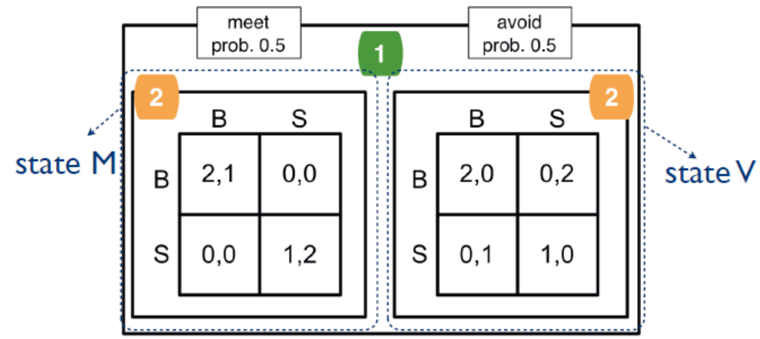
Player 2: **{BB, BS, SB, SS}**



Imperfect Information

What are the (pure) equilibria in this kind of game?

We first calculate the expected payoff U_1 for each strategy of 1 given a pure strategy of player 2:



$$U_1^I(a_1^1,(a_2^1,a_2^2),\theta_1) = p(\theta_2=\text{meet}|\theta_1)u_1(a_1,a_2^1,\theta_2=\text{meet})+$$
$$p(\theta_2=\text{avoid}|\theta_1)u_1(a_1,a_2^2,\theta_2=\text{avoid}).$$

$$U_1^I(B,(B,S)) = 0.5u_1(B,B,\text{meet})+$$
$$0.5u_1(B,S,\text{avoid})$$
$$= 0.5\cdot 2 + 0.5\cdot 0 = 1$$

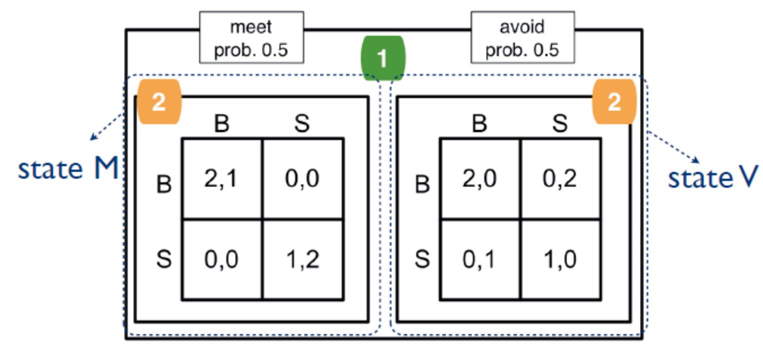
action for both types

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Example

A pure strategy Nash equilibrium in this game is a triple of actions, with the property that

- The **action of player 1 is optimal**, given both actions of the two player 2 types (and player 1's belief about the state)
- The **action of each player 2 type is optimal**, given the action of player 1



	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Example

First, player 1 has to determine his or her best response given the actions of both types of player 2

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Second, determine the best responses of player 2 against player 1 in both types

Type "meet"

1	0
0	2

Type "avoid"

0	2
1	0

Example

The pure strategy NE can be obtained by matching these results

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Type "meet"

1	0
0	2

Type "avoid"

0	2
1	0

Not a NE!

Example

The pure strategy NE can be obtained by matching these results

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	$\boxed{1}$	1	0
S	0	0,5	0,5	1

Type "meet"

$\boxed{1}$	0
0	2

Type "avoid"

0	$\boxed{2}$
1	0

{B, (B,S)} is a NE!

Example

The pure strategy NE can be obtained by matching these results

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Type "meet"

1	0
0	2

Type "avoid"

0	2
1	0

Not a NE!

Example

The pure strategy NE can be obtained by matching these results

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	0,5	0,5	1

Type "meet"

1	0
0	2

Type "avoid"

0	2
1	0

Not a NE!

We have 1 pure strategy NE's:

- **{B, (B,S)}**

Example: 2 unknown types

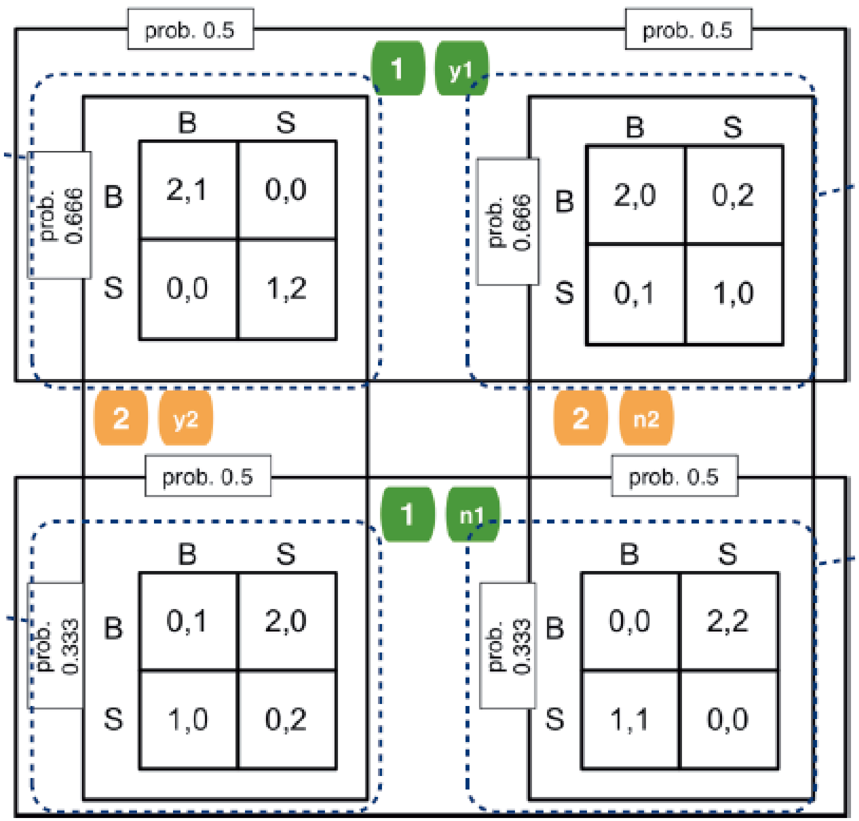
We can make the game even more interesting when both players don't know whether the other one wants to meet or avoid the other one

state yy

every player wants to go out with the other

state yn

only player 1 wants to go out



state ny

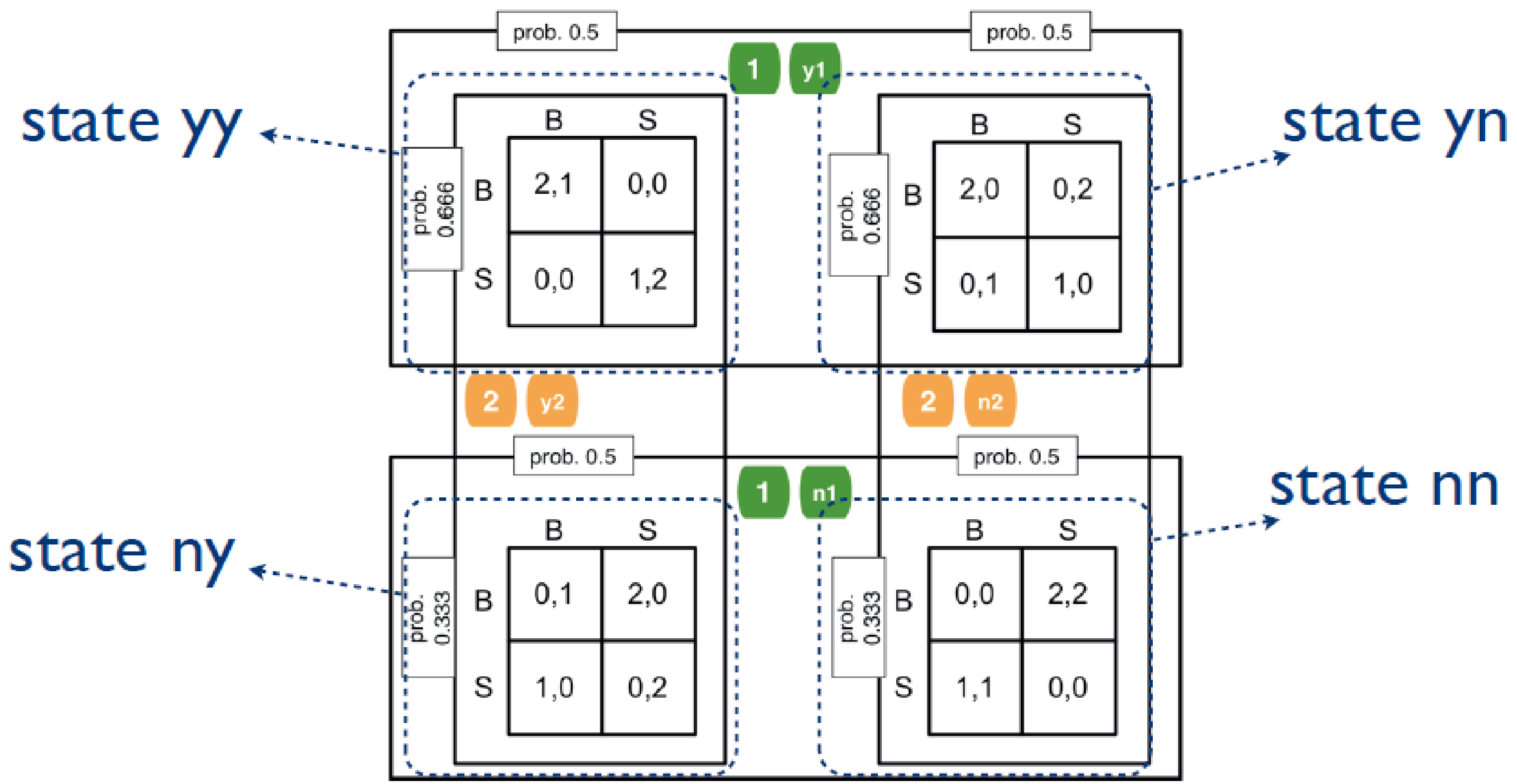
only player 2 wants to go out

state nn

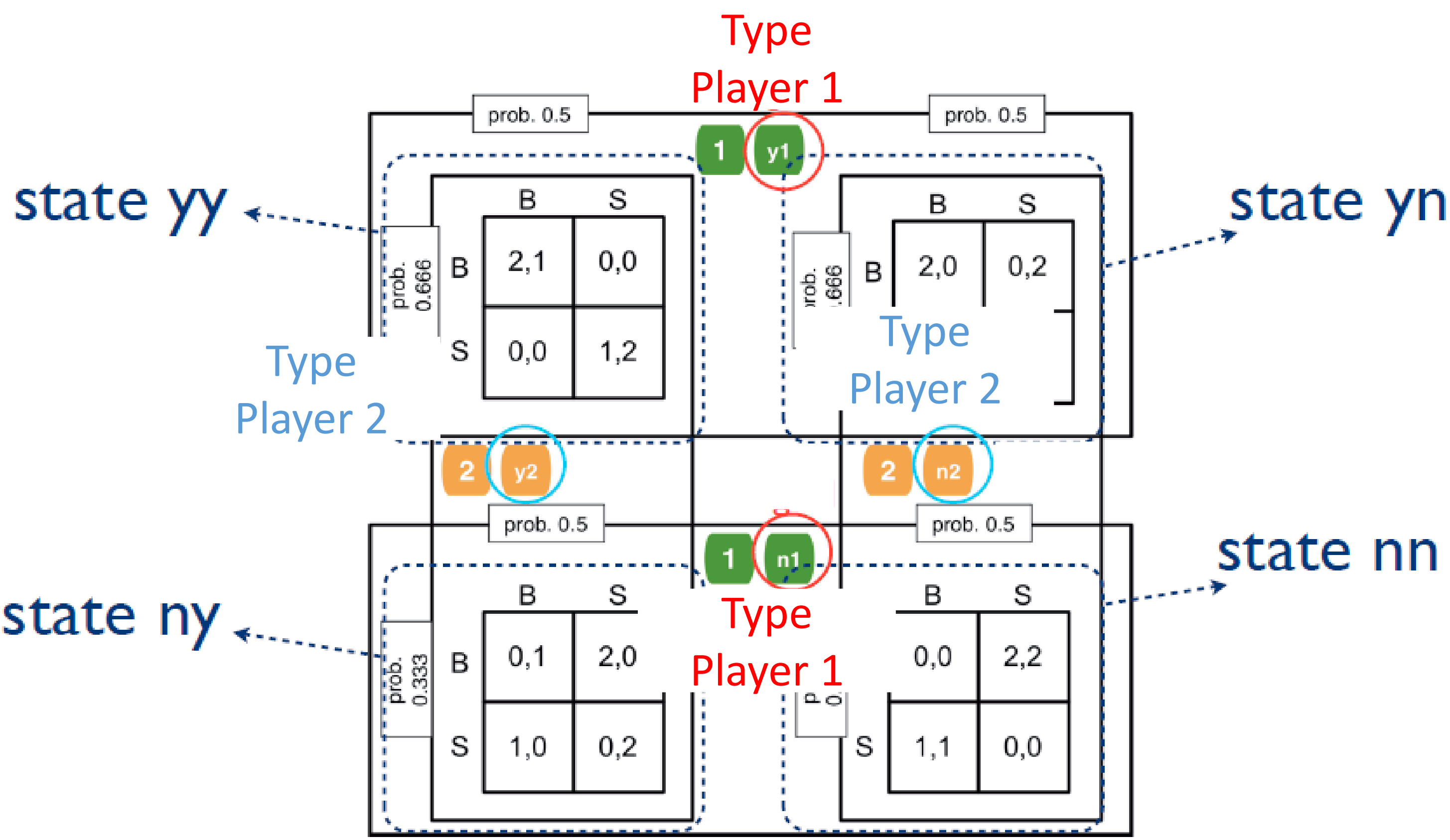
neither of the players wants to go out with the other

Example: 2 unknown types

Note that in this game, player 1 cannot distinguish between states yy and yn, and between ny and nn (vice versa for player 2)



Example: 2 unknown types



Example: 2 unknown types

Depending on his type, each player has to determine what he can expect in terms of payoff, given the probabilities of the types of the other player.

Player 1:

P1

y1

	(B,B)	(B,S)	(S,B)	(S,S)
B	2	1	1	0
S	0	1/2	1/2	1

{B,(B,S)}

n1

	(B,B)	(B,S)	(S,B)	(S,S)
B	0	1	1	2
S	1	1/2	1/2	0

{B,(B,S)}

P2

state yy

B	S
1	0
0	2

state yn

B	S
0	2
1	0

state ny

B	S
1	0
0	2

state nn

B	S
0	2
1	0

Example: 2 unknown types

Depending on his type, each player has to determine what he can expect in terms of payoff, given the probabilities of the types of the other player.

Player 2:

P1

state yy

B	2	0
S	0	1

state ny

B	0	2
S	1	0

state yn

B	2	0
S	0	1

state nn

B	0	2
S	1	0

{(B,S),B} and {(S,B),S}

{(S,B),S}

P2

	(B,B)	(B,S)	(S,B)	(S,S)
B	1	2/3	1/3	0
S	0	2/3	4/3	2

y2

	(B,B)	(B,S)	(S,B)	(S,S)
B	0	1/3	2/3	1
S	2	4/3	2/3	0

n2

Example: 2 unknown types

Putting things together, one can see that there 2 pure strategy Nash equilibria for this Bayesian game

	y1	n1	
P1	{B,(B,S)}	{B,(B,S)}	→ {(B,B),(B,S)}
P2	{(B,S),B} and {(S,B),S}	{(S,B),S}	→ {(S,B),(S,S)}
	y2	n2	

Conclusion

- Extensive form games model sequential interactions
- There is always a pure strategy NE
- Much easier to solve by backwards induction
- **Extensions:** Mix of simultaneous play, chance
- **Imperfect information games:** Bayesian Games
- Use belief over the pay-off functions to compute expected utility