Part3- Extensive Form Games

Based on slides by David Sarne and Jackson, Leyton-Brown & Shoham

The Notion of Sequence

- The notion of sequence, or time, is an important ingredient of a game.
- Normal form game does not involve any explicit representation of sequence
- Extensive form game is an alternative representation for that purpose

Perfect Information Extensive Form Game

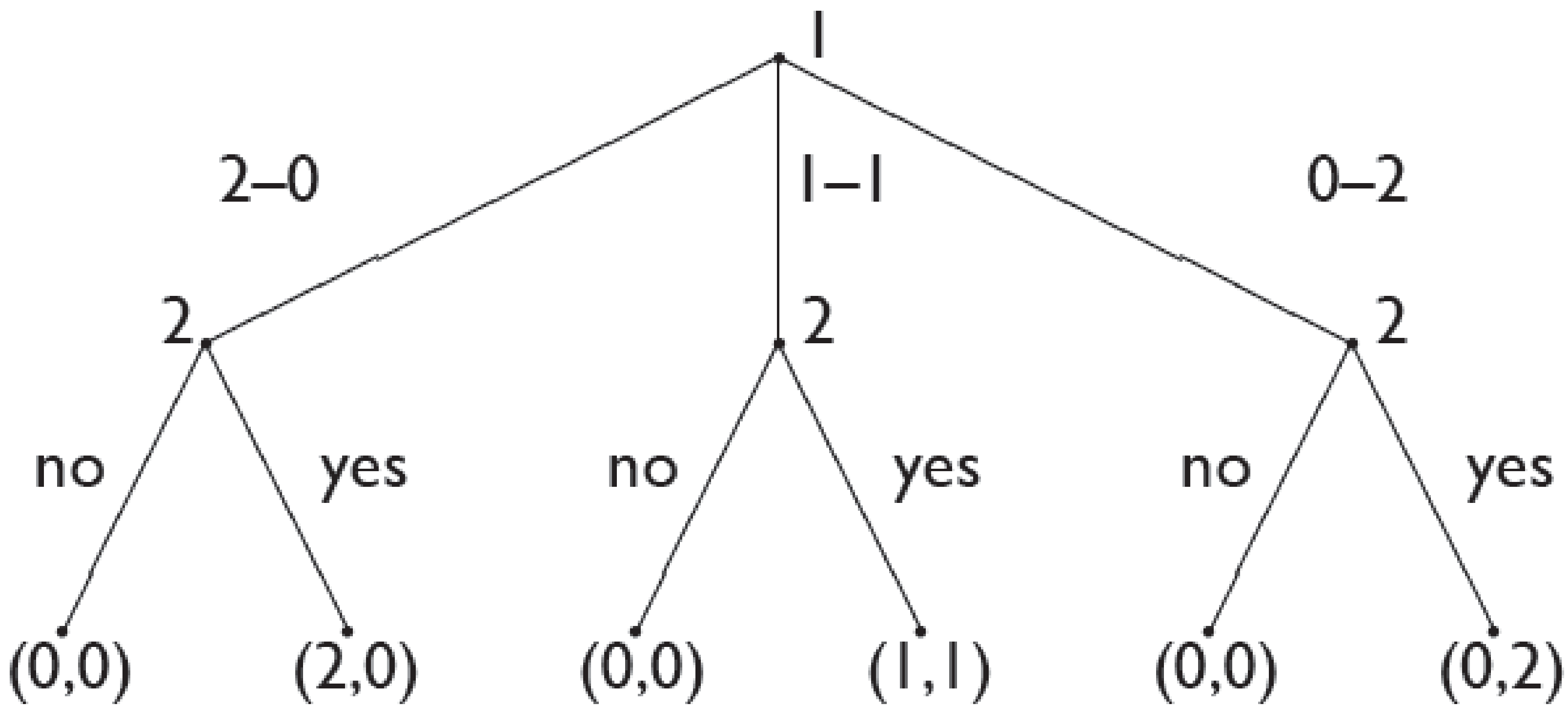
- A (finite) perfect-information game (in extensive form) is defined by the tuple (N ; A ; H ; Z ; χ ; ρ ; σ ; u), where:
  - Players: N
  - Actions: A
  - Choice nodes and labels for these nodes:
    - Choice nodes: H
    - Action function: $\chi : H \rightarrow 2^A$ assigns to each choice node a set of possible actions
    - Player function: $\rho : H \rightarrow N$ assigns to each non-terminal node h a player $i \in N$ who chooses an action at h

Perfect Information Extensive Form Game

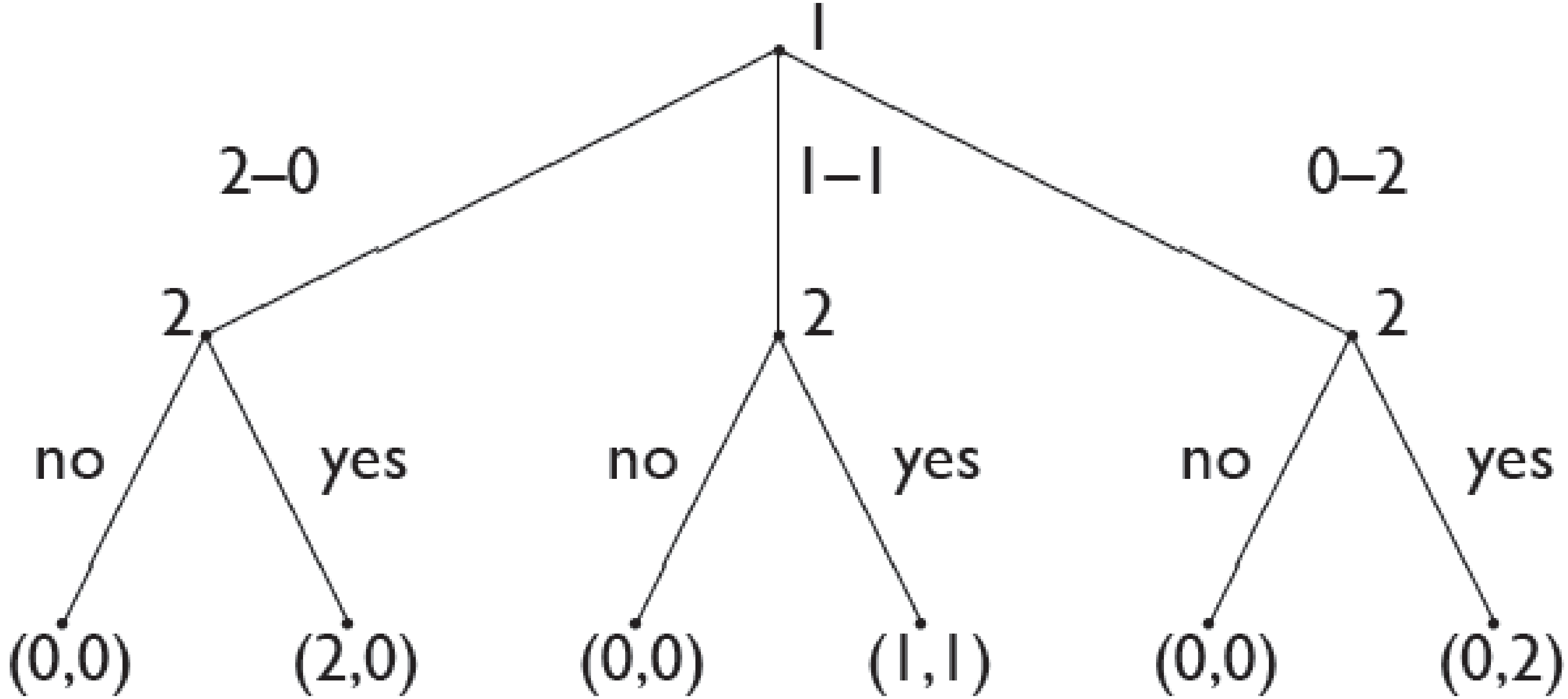
- A (finite) perfect-information game (in extensive form) is defined by the tuple (N ; A ; H ; Z ; χ ; ρ ; σ ; u), where:
  - Terminal nodes: Z is a set of terminal nodes, disjoint from H
  - Successor function: $\sigma : H \mathbf{x} A \rightarrow H \cup Z$ maps a choice node and an action to a new choice node or terminal node such that for all $h_1, h_2 \in H$ and $a_1, a_2 \in A$, if $\sigma(h_1, a_1) = \sigma(h_2, a_2)$ then $h_1 = h_2$ and $a_1 = a_2$
    - Choice nodes form a tree: nodes encode history
  - Utility function: $u = (u_1, \ldots, u_n)$; $u_i : Z \rightarrow \mathbb{R}$ is a utility function for player i on the terminal nodes Z.

Example- The Sharing Game

- A brother and a sister needs to divide 2 indivisible and identical presents from their parents.
- The brother is the first to suggest a split.
- If the sister doesn't agree, they both get nothing.



Strategies



- How many pure strategies are there?
  - Player 1 has 3 strategies (2-0,1-1,0-2)
  - Player 2 has 8 strategies

Pure Strategies

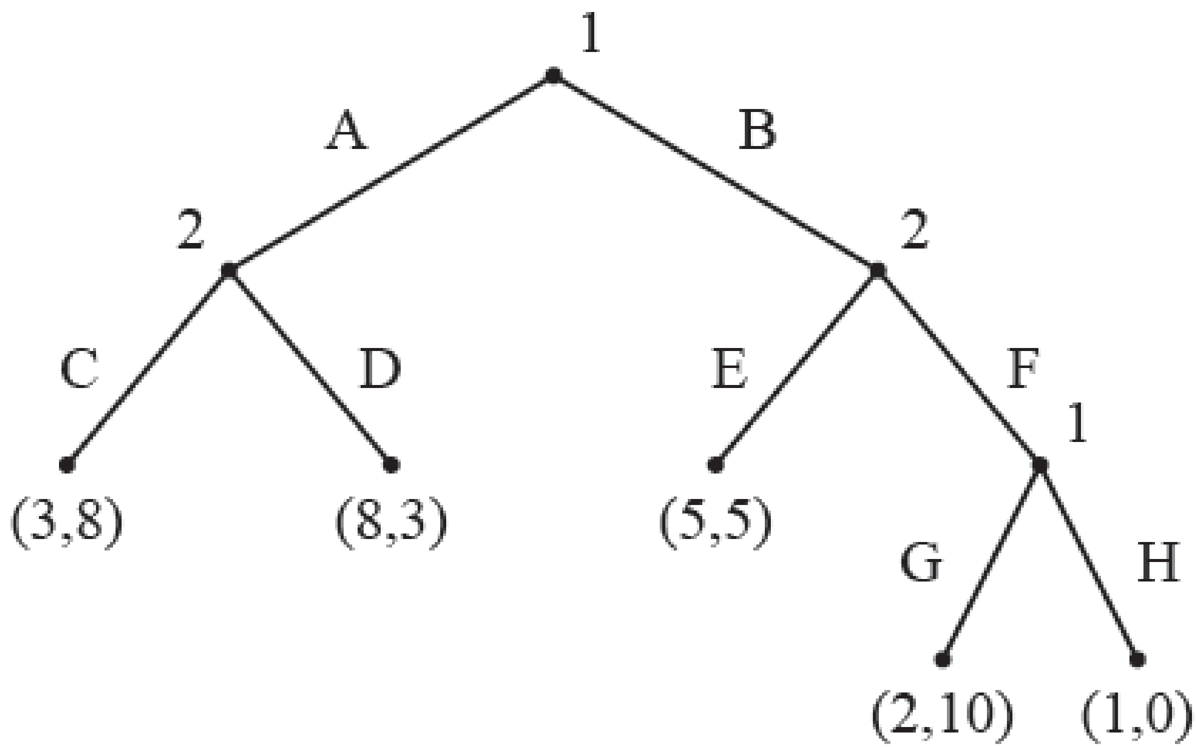
A pure strategy for a player in a perfect-information game is a complete specification of which deterministic action to take at every node belonging to that player.

Definition (pure strategies)

Let $G = (N, A, H, Z, \chi, \rho, \sigma, u)$ be a perfect-information extensive-form game. Then the pure strategies of player i consist of the cross product

$$\prod_{h \in H, \rho(h)=i} \chi(h)$$

Example



- What are the pure strategies of player2
  - $S2 = \{(C,E),(C,F),(D,E),(D,F)\}$
- What are the pure strategies of player 1?
  - $S1 = \{(B,G),(B,H),(A,G),(A,H)\}$

We will never reach to H, but still, (A,G) is different from (A,H)

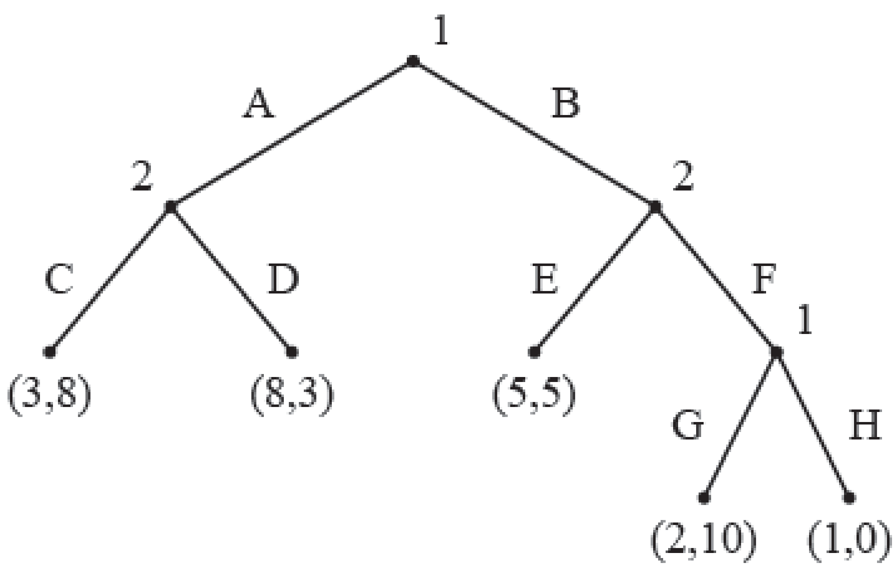
Nash Equilibrium

Using the basic definition of pure strategy it is easy to redefine:

- Mixed strategy
- Best response
- Nash Equilibrium

Converting to a Normal-form Game

For every perfect-information game there exists a corresponding normal-form game with the same strategy space.



	CE	CF	DE	DF
AG	$3,8$	$3,8$	$8,3$	$8,3$
AH	$3,8$	$3,8$	$8,3$	$8,3$
BG	$5,5$	$2,10$	$5,5$	$2,10$
BH	$5,5$	$1,0$	$5,5$	$1,0$

Note:
Every result/proof that is true for normal form games is also true for extensive form games

The Advantages of Extensive Form Game

- An explicit representation of time
- A compact representation
  - The temporal structure of the extensive-form representation can result in a certain redundancy within the normal form.
  - E.g., in the previous example we had 16 outcomes in the normal form game versus only 5 with the extensive form game.

- But…
  - the reverse transformation— from the normal form to the perfect-information extensive form—does not always exist
  - E.g. matching coins

What about Mixed Strategies?

Theorem

Every perfect information game in extensive form has a PSNE

- Why?
  - Since the game is sequential
- Since players take turns, and everyone gets to see everything that happened thus far before making a move, it is never necessary to introduce randomness into action selection.
- Note: this is true only for perfect information extensive form game.

“Let’s Burn the Boats”

- What does it mean?
  - You give away alternative B
  - So the other player won’t have a choice other than trying to achieve alternative A, which is better for you.

- **Paradox of commitment**
  - It is by limiting my own options that I can manage to influence the rival’s course of actions in my interest.

Hernan Cortes and the Aztecs

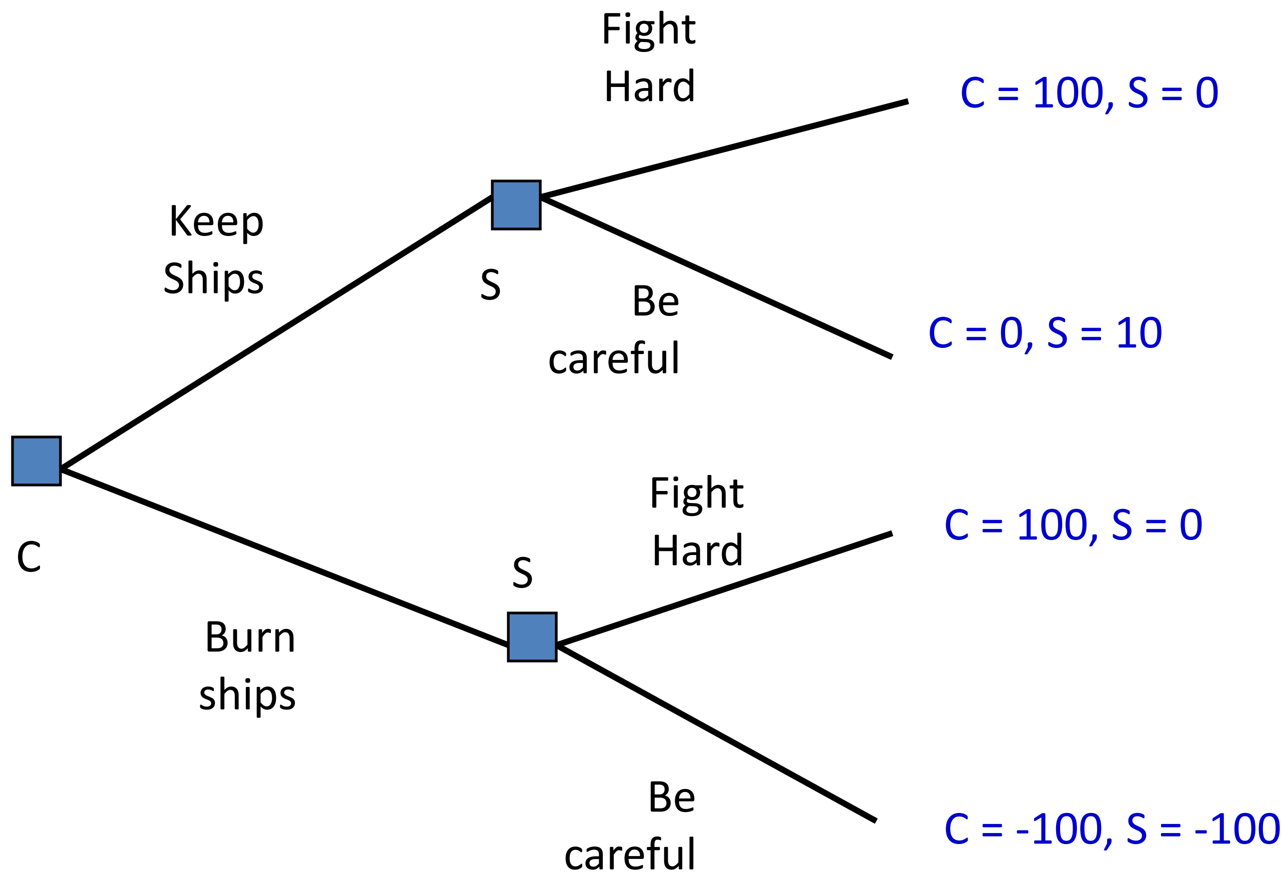
- From a good Spanish family.
  - Failed at being a European soldier.
  - Failed at becoming a lawyer.

Cortes was hungry for wealth and recognition.

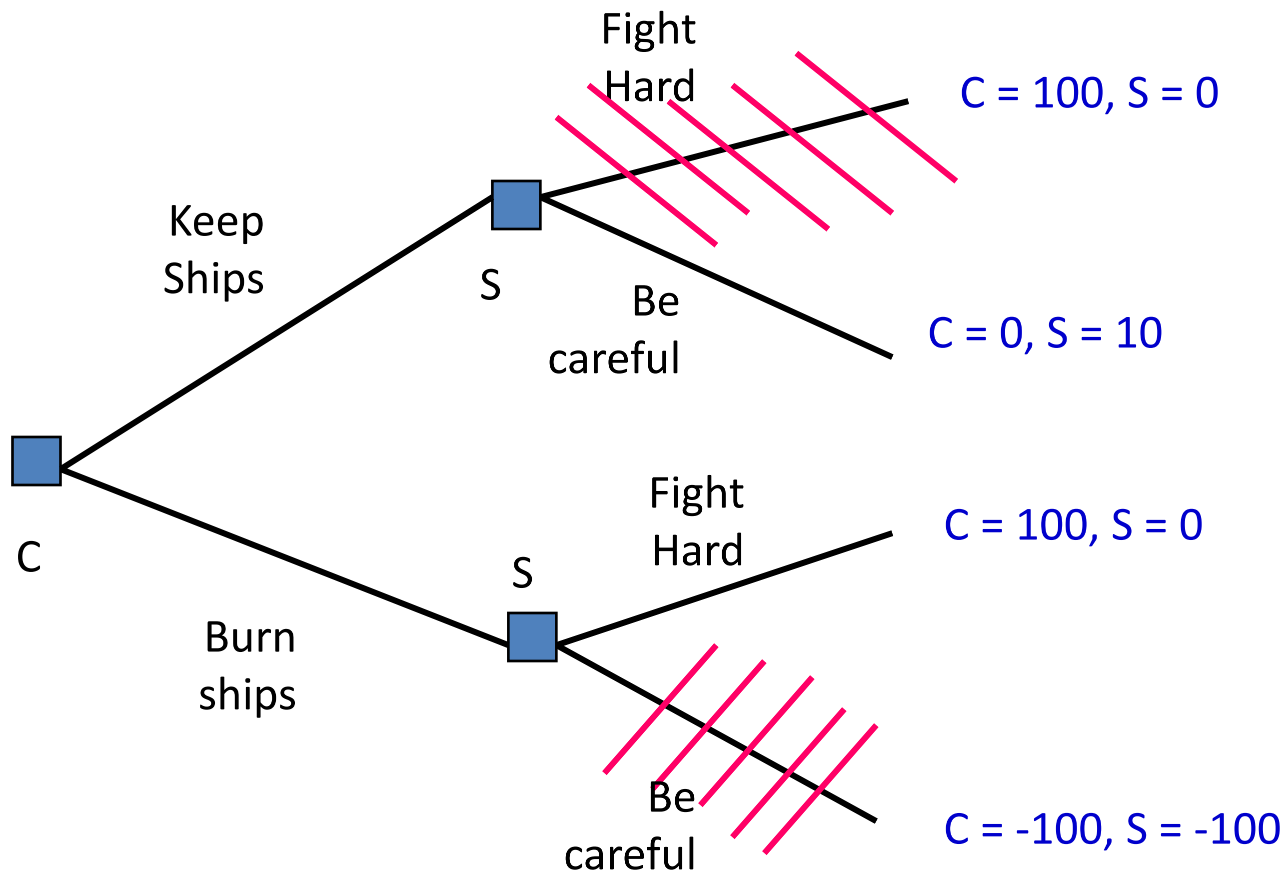
Cortes

- 1519 – landed on the coast of Mexico with :
  - 600 men, 16 horses, and a few cannon.
- WHY WOULD CORTES BURN HIS SHIPS WHEN HE GOT TO THE NEW WORLD?

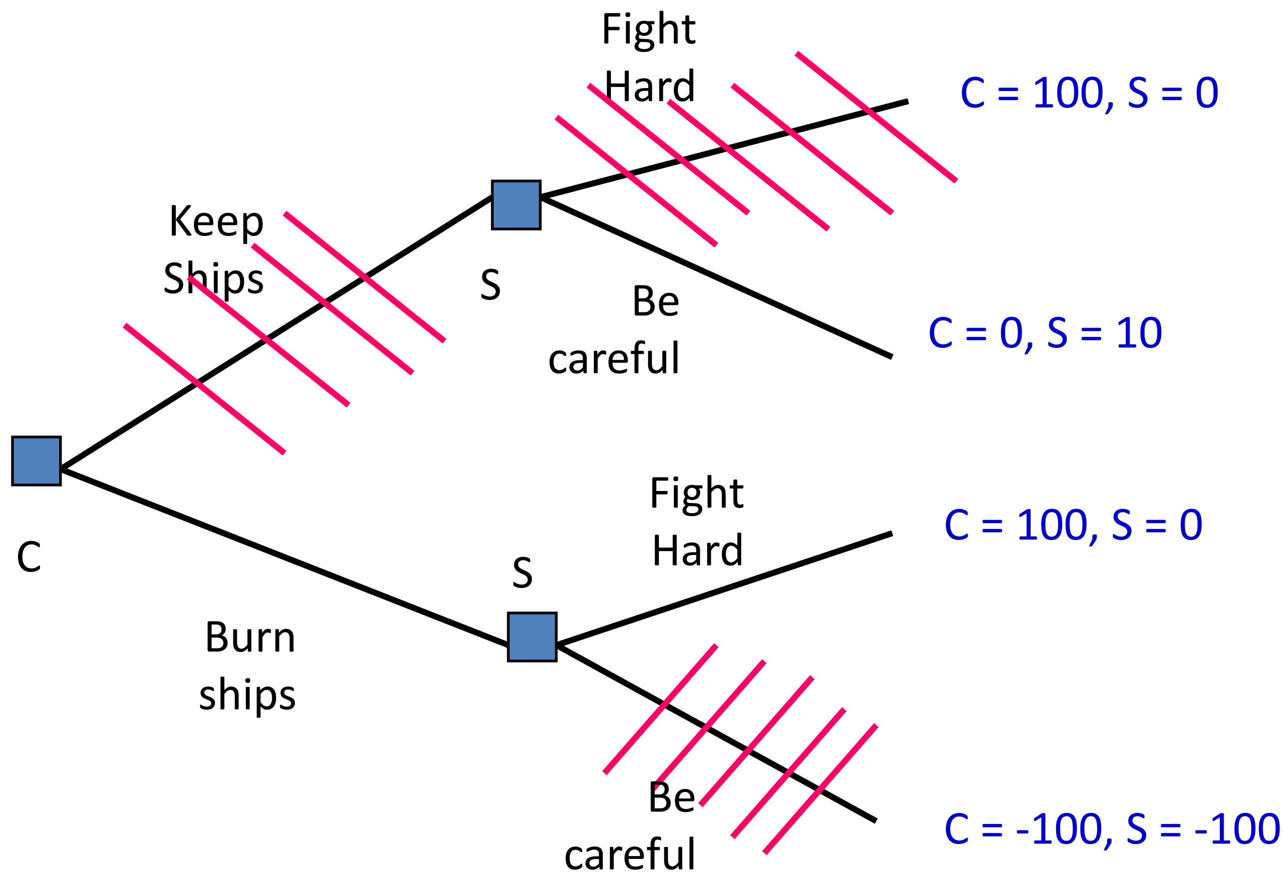
Think of Cortes trying to motivate his own soldiers



If no retreat possible, will fight hard or die. But if retreat is possible, may fight less hard and ‘run away’



So Cortes wants to burn his ships. It is a credible commitment not to retreat – and this alters how his own troops behave.



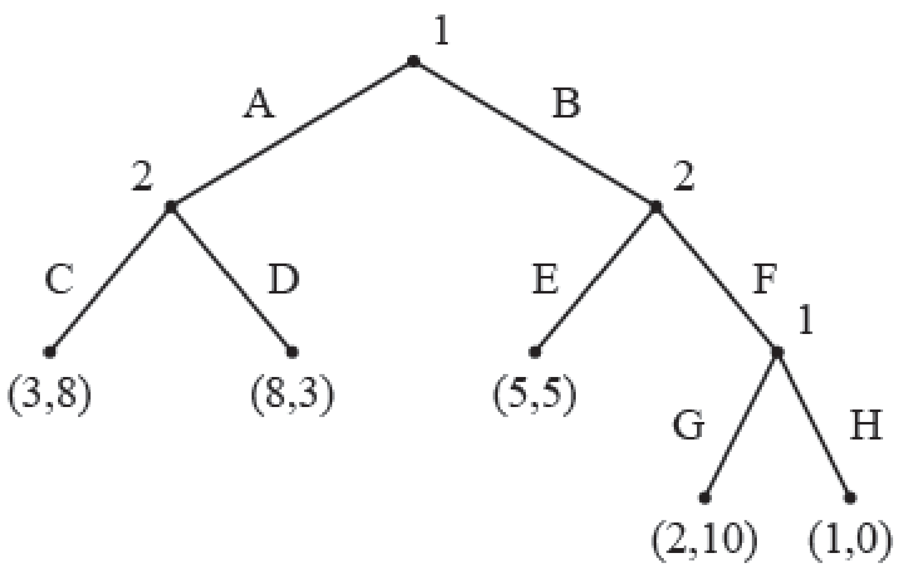
Where else did we see it?

- Nick from Golden Balls
- Chicken game

- Business guru, Tom Peters swears by Cortés' destructive strategy by going so far as to sug-gest that every company hire a CDO – a Chief Destructive Officer.

Subgame-perfect equilibrium

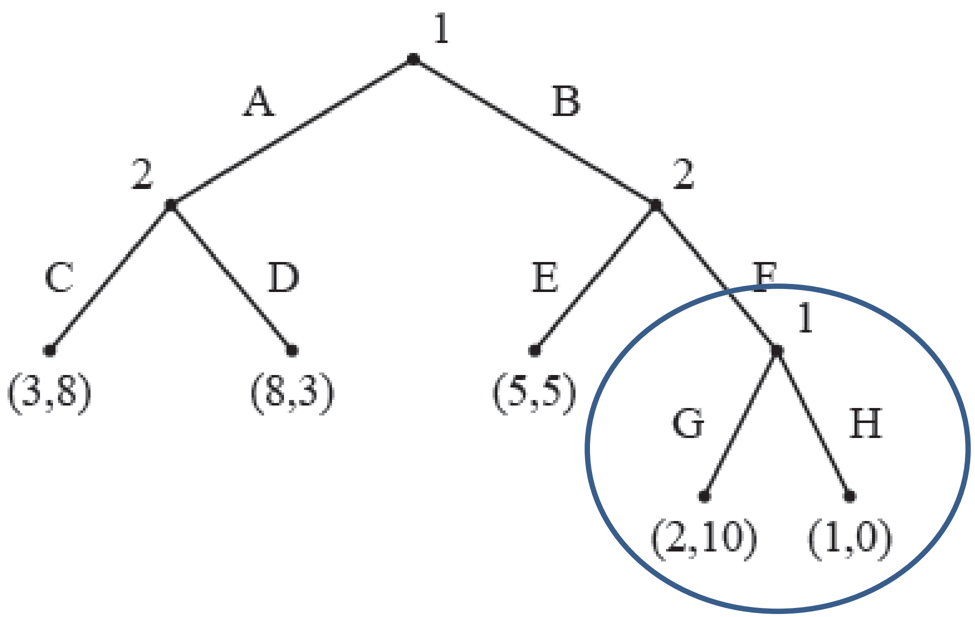
- Find all the Nash equilibria in the game



	CE	CF	DE	DF
AG	3, 8	3, 8	8, 3	8, 3
AH	3, 8	3, 8	8, 3	8, 3
BG	5, 5	2, 10	5, 5	2, 10
BH	5, 5	1, 0	5, 5	1, 0

- Is (AG,CF) a reasonable equilibrium?

Subgame-perfect equilibrium



	CE	CF	DE	DF
AG	$3,8$	$3,8$	$8,3$	$8,3$
AH	$3,8$	$3,8$	$8,3$	$8,3$
BG	$5,5$	$2,10$	$5,5$	$2,10$
BH	$5,5$	$1,0$	$5,5$	$1,0$

- Is (BH,CE) a reasonable equilibrium?
  - Why player 1 would play H if he can play G?
  - Since the choice in H is a threat that keeps the equilibrium. Player 2 know that if he chooses F player 1 will choose H and therefore player 2 prefers to keep choosing E.
  - However, is this a credible threat? Will player 1 actually play H if we reach to that point?

Formal Definition

Definition (subgame of G rooted at h)

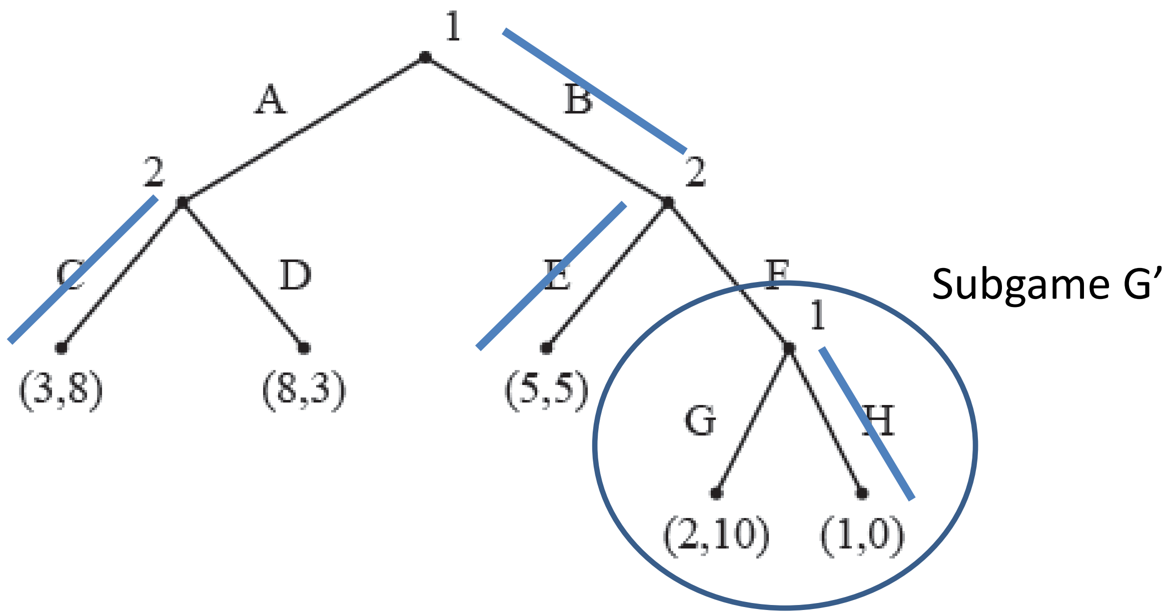
The subgame of G rooted at h is the restriction of G to the descendents of h.

Definition (subgames of G)

The set of subgames of G is defined by the subgames of G rooted at each of the nodes in G.

- *The subgame-perfect equilibria (SPE) of a game G are all strategy profiles s such that for any subgame G′ of G, the restriction of s to G′ is a Nash equilibrium of G′.*

Recall our example

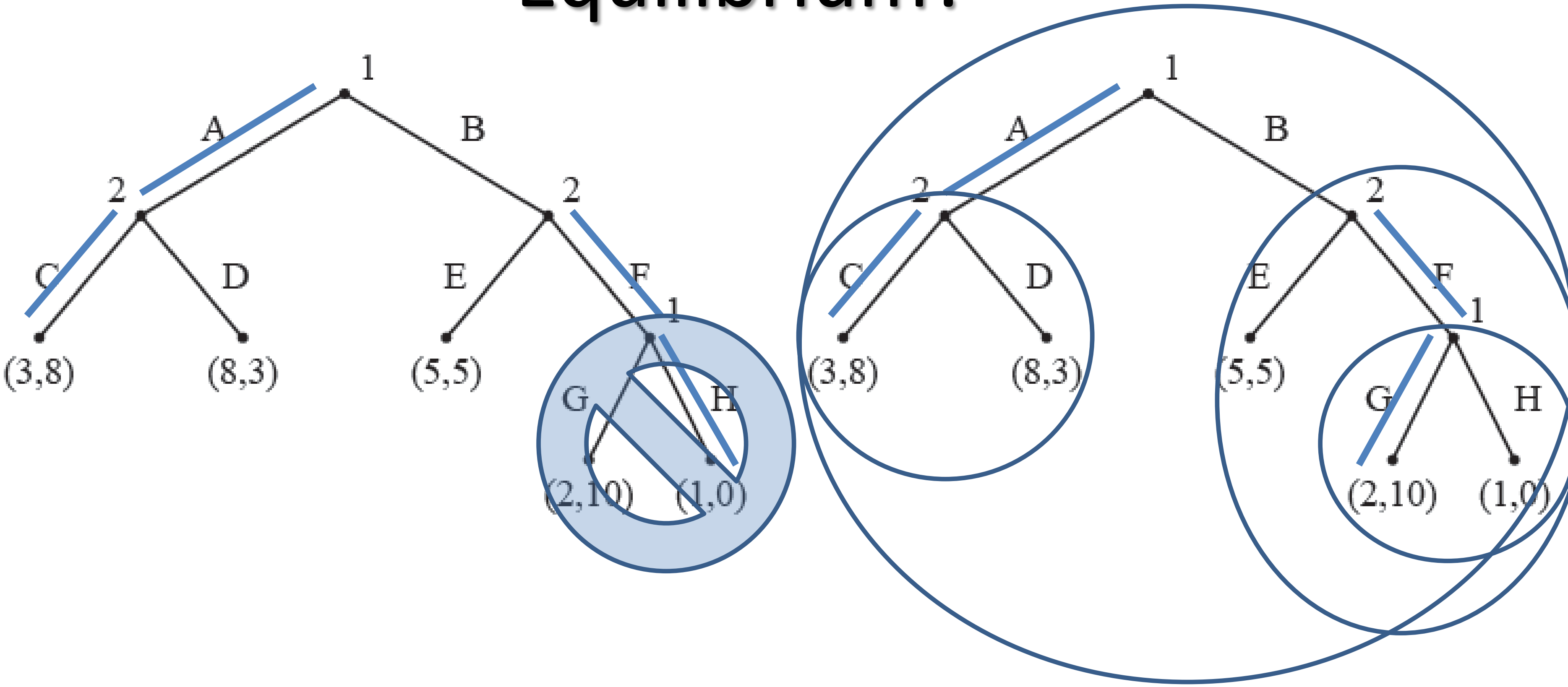


In this subgame, H is not part of a N.E. Thus, (BH,CE) is not subgame-perfect equilibrium

Additional Notes

- The definition of subgame-perfect rules out "noncredible threats".
- Since G is its own subgame, every SPE is also a Nash equilibrium.
- Every perfect-information extensive-form game has at least one subgame-perfect equilibrium (proof is by induction on the height of the game tree).

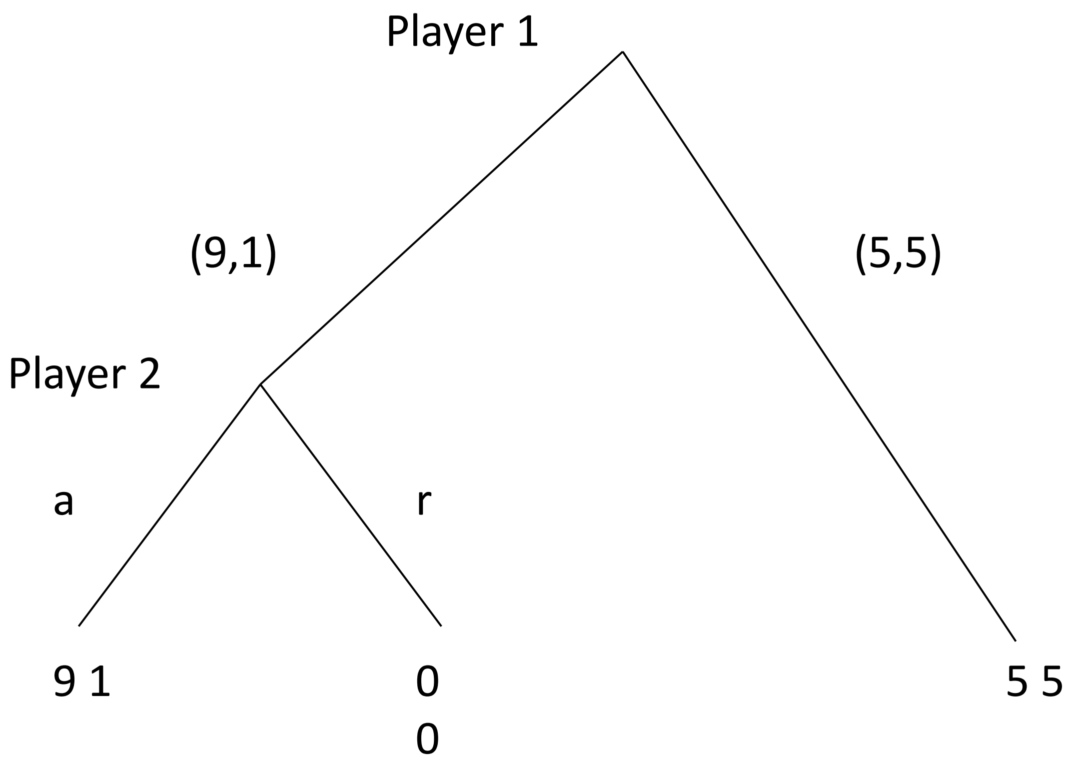
What is the Subgame-perfect Equilibrium?



The strategy H in (AH,CF) is called off path

Example: Mini Ultimatum Game

- Proposer (Player 1) can suggest one of two splits of £10: (5,5) and (9,1).
- Responder (Player 2) can decide whether to accept or reject (9,1), but has to accept (5,5). Reject leads to 0 for both



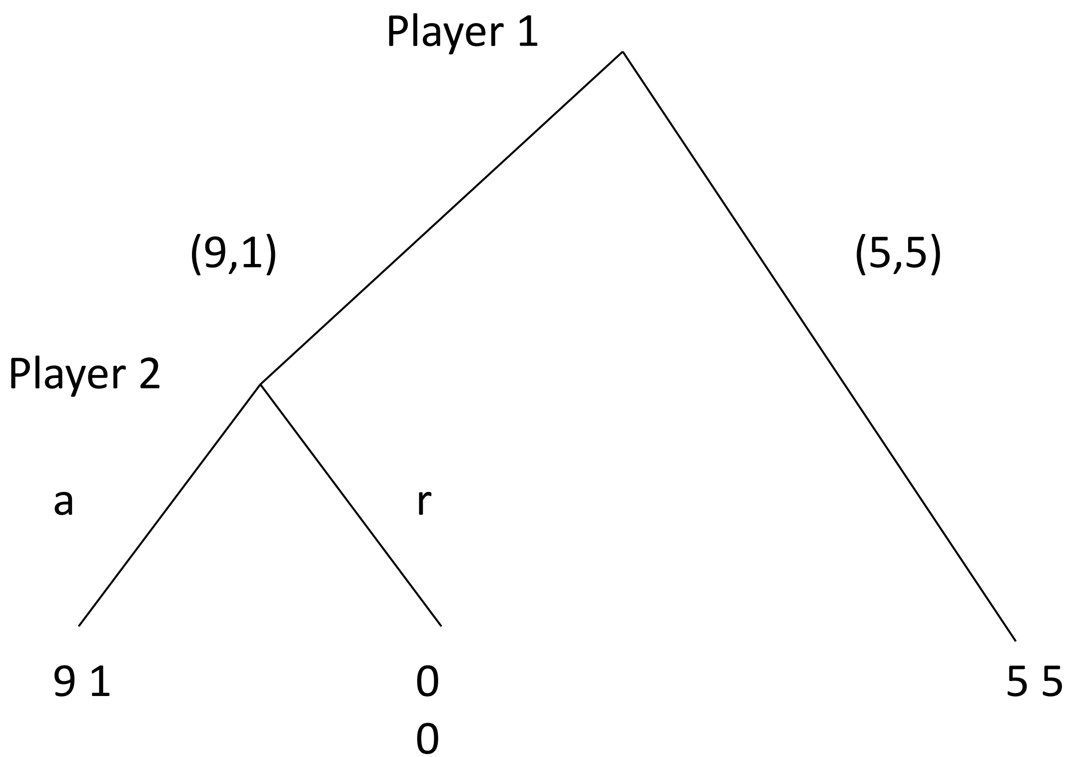
Mini Ultimatum Game in Strategic Form

		Player 2	
		accept (9,1)	reject (9,1)
Player 1	propose (5,5)	5,5	5,5
	propose (9,1)	9,1	0,0

- There are **two** equilibria:
  1. (propose (9,1), accept (9,1))
  2. (propose (5,5), reject (9,1)).
- Equilibrium 2 is in **weakly dominated** strategies (reject (9,1) is weakly dominated)

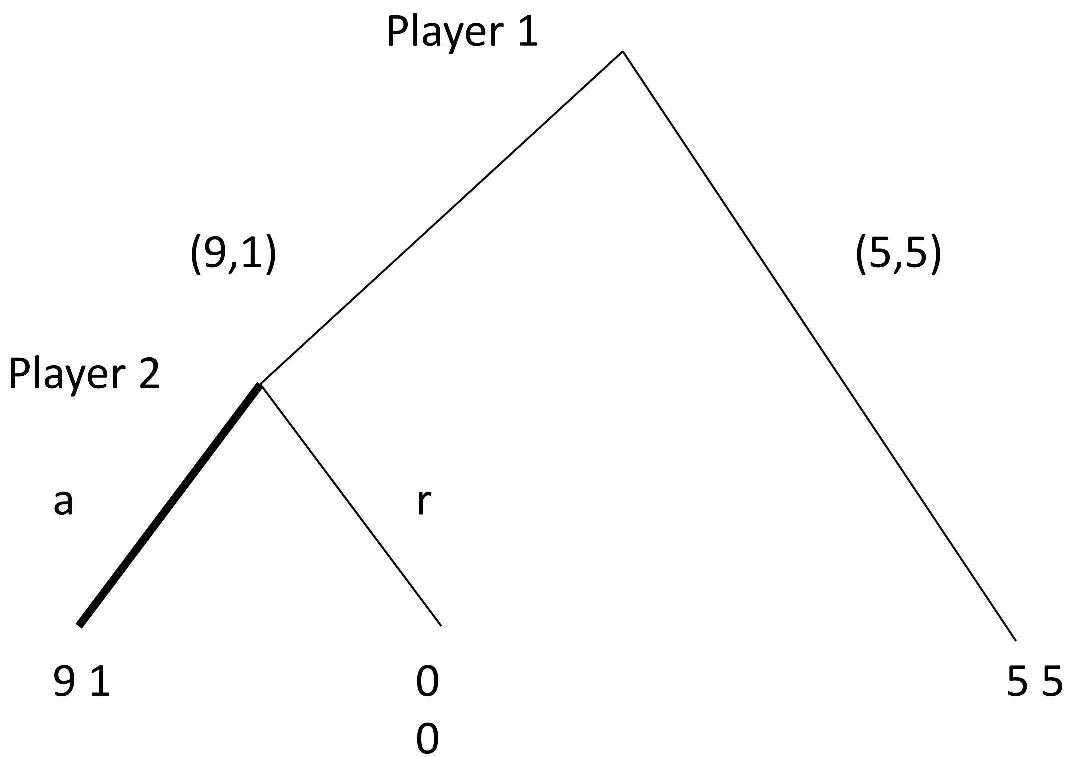
Example: Mini Ultimatum Game

- There is one subgame of length 1, following (9,1)



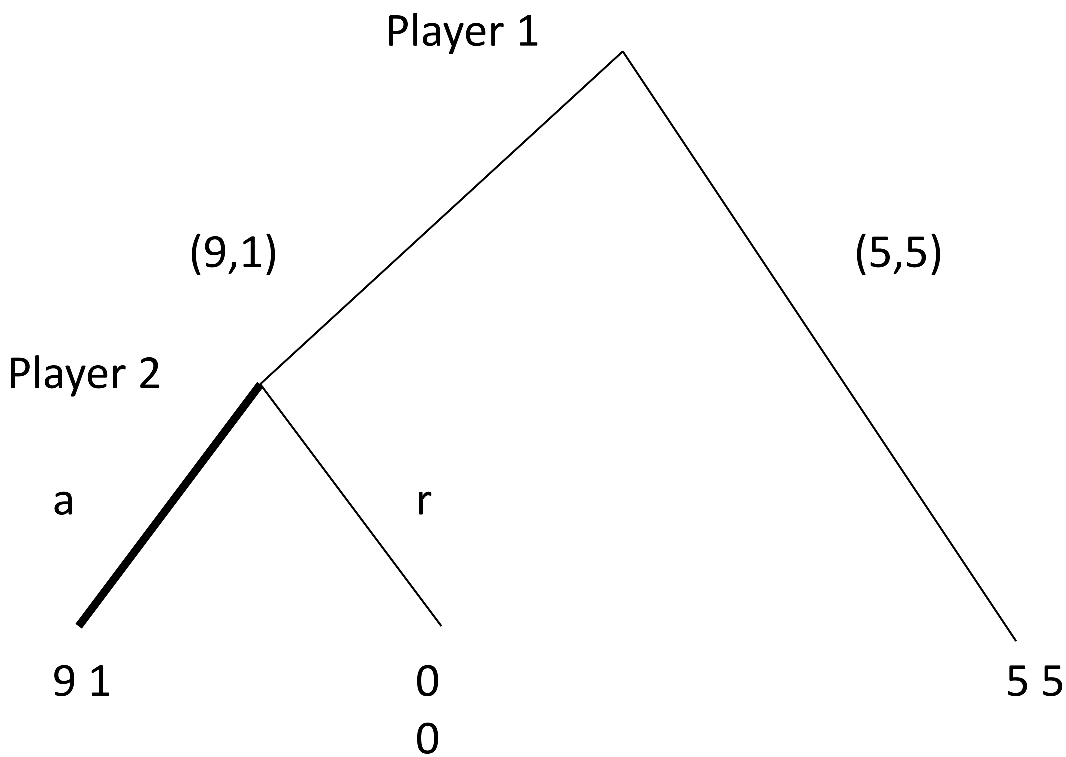
Example: Mini Ultimatum Game

- There is one subgame of length 1, following (9,1)
- The optimal action is accept



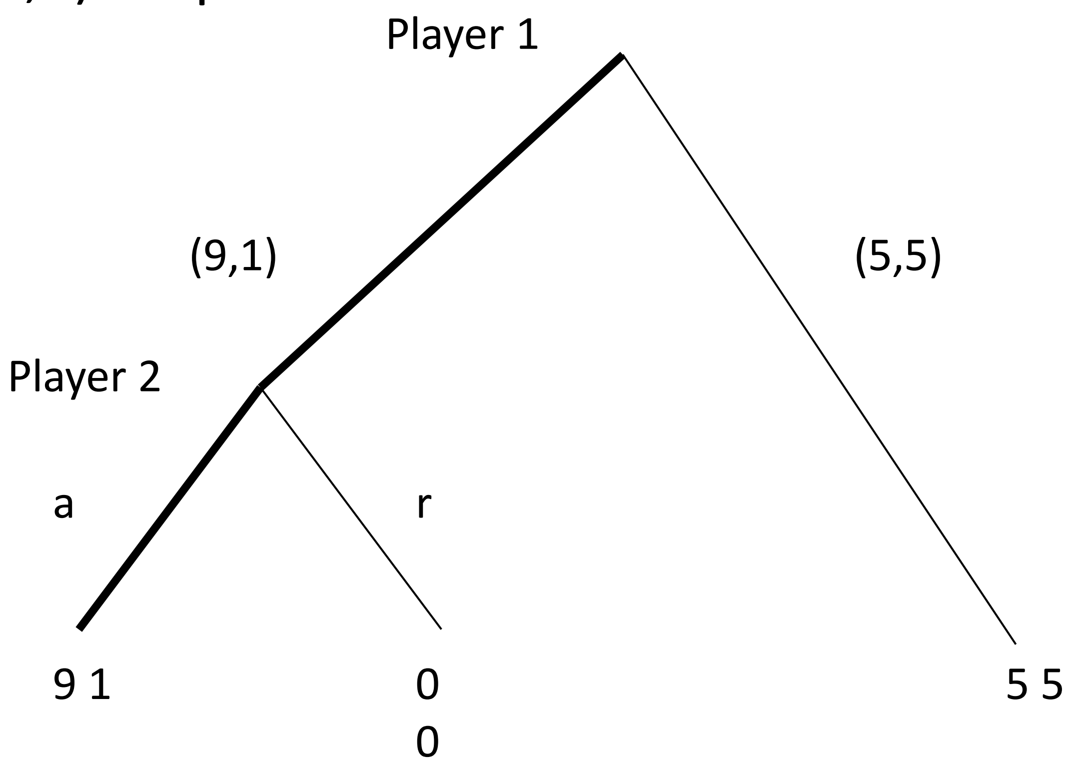
Example: Mini Ultimatum Game

- There is one subgame of length 1, following (9,1)
- The optimal action is accept
- There is one subgame of length 2, the whole game



Example: Mini Ultimatum Game

- There is one subgame of length 1, following (9,1)
- The optimal action is accept
- There is one subgame of length 2, the whole game
- Taking "accept" in the subgame of length 1 as given, we see that (9,1) is optimal



Computing one SPE - Backward Induction

all leaf nodes

function BACKWARDINDUCTION (node h) **returns** $u(h)$
if $h \in Z$ **then**
  return $u(h)$

set of all actions available to player at that node

$best\_util \leftarrow -\infty$
forall $a \in \chi(h)$ **do**
  $util\_at\_child \leftarrow$ BACKWARDINDUCTION$(\sigma(h, a))$
  if $util\_at\_child_{\rho(h)} > best\_util_{\rho(h)}$ **then**
    $best\_util \leftarrow util\_at\_child$
return $best\_util$

vector denoting the utility for each player

Computing one SPE - Backward Induction

- This procedure does not return an equilibrium strategy for each of the n players, but rather describes how to label each node with a vector of n real numbers.
- This labeling can be seen as an extension of the game's utility function to the nonterminal nodes H.
- The players' equilibrium strategies follow straightforwardly from this extended utility function: every time a given player i has the opportunity to act at a given node $h \in H$ (i.e., $\rho(h) = i$), that player will choose an action $a_i \in \chi(h)$ that solves argmax $a_i \in \chi(h)$ $u_i(\sigma(a_i,h))$.

Example: Voting for a Payraise

Three legislators are voting on whether to give themselves a pay raise.
All three want the pay raise; however,
Each face a small cost in voter resentment c>0.
The benefit for the raise is greater than cost: b>c
They vote in the order 1-2-3. Simple majority rule
What is the outcome obtained by backward induction?

(N; NY; nyyn)

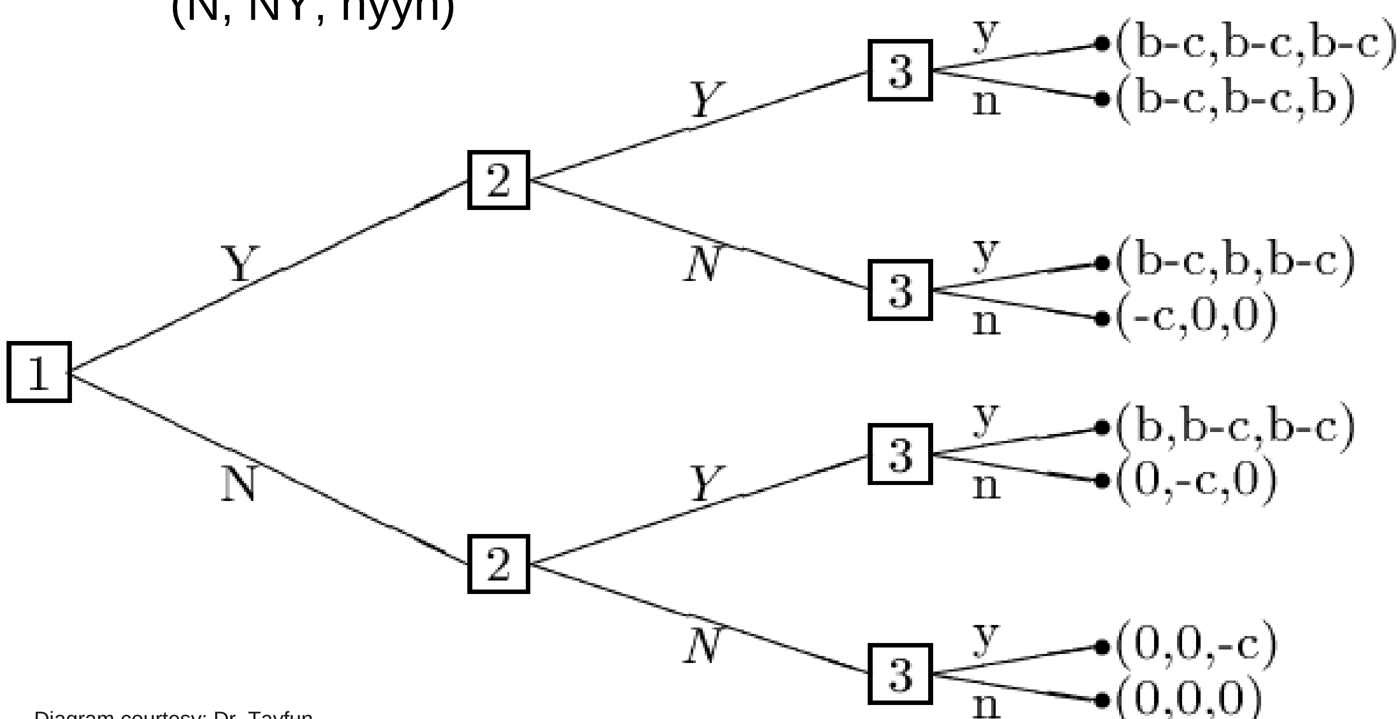
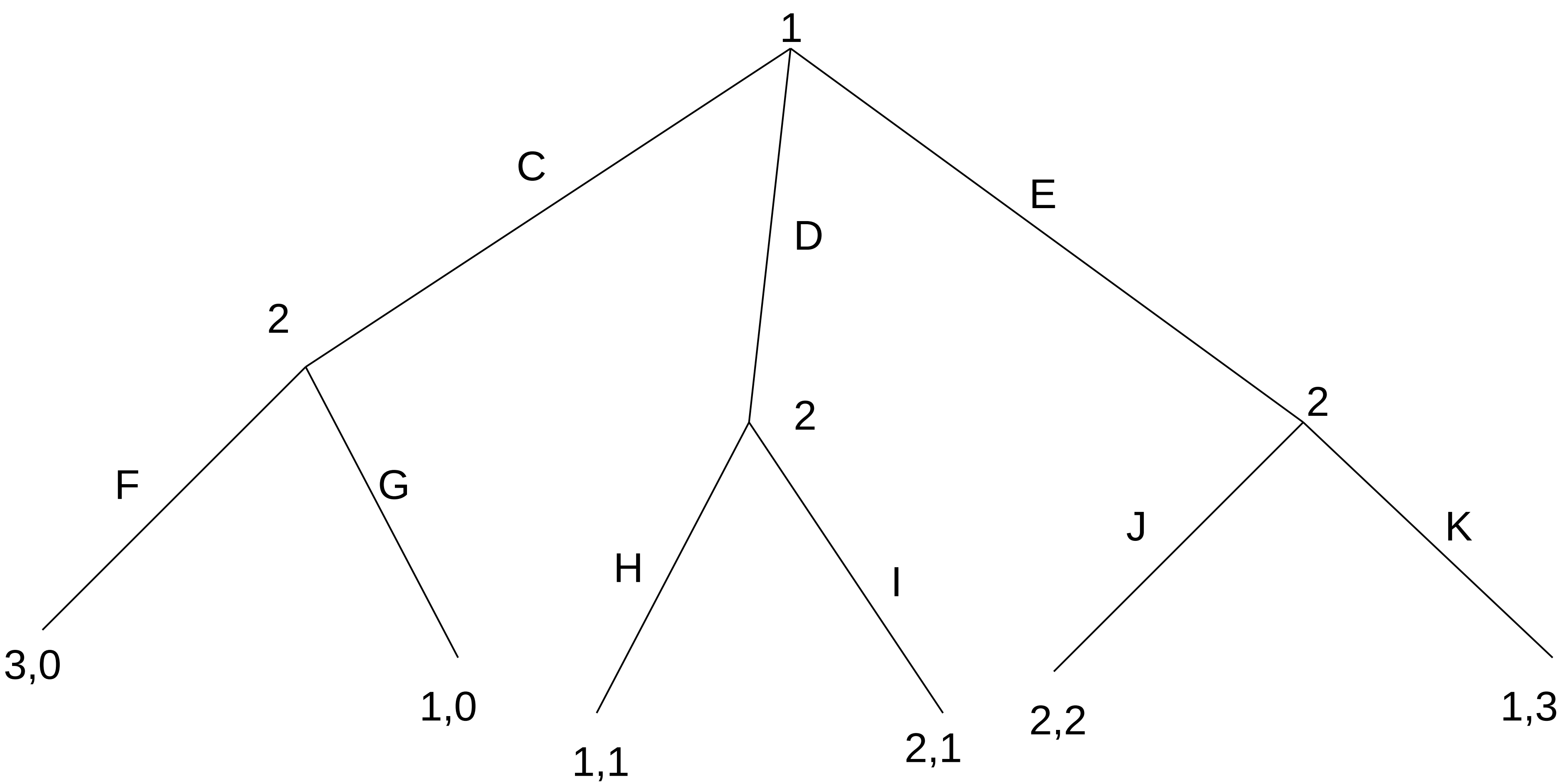


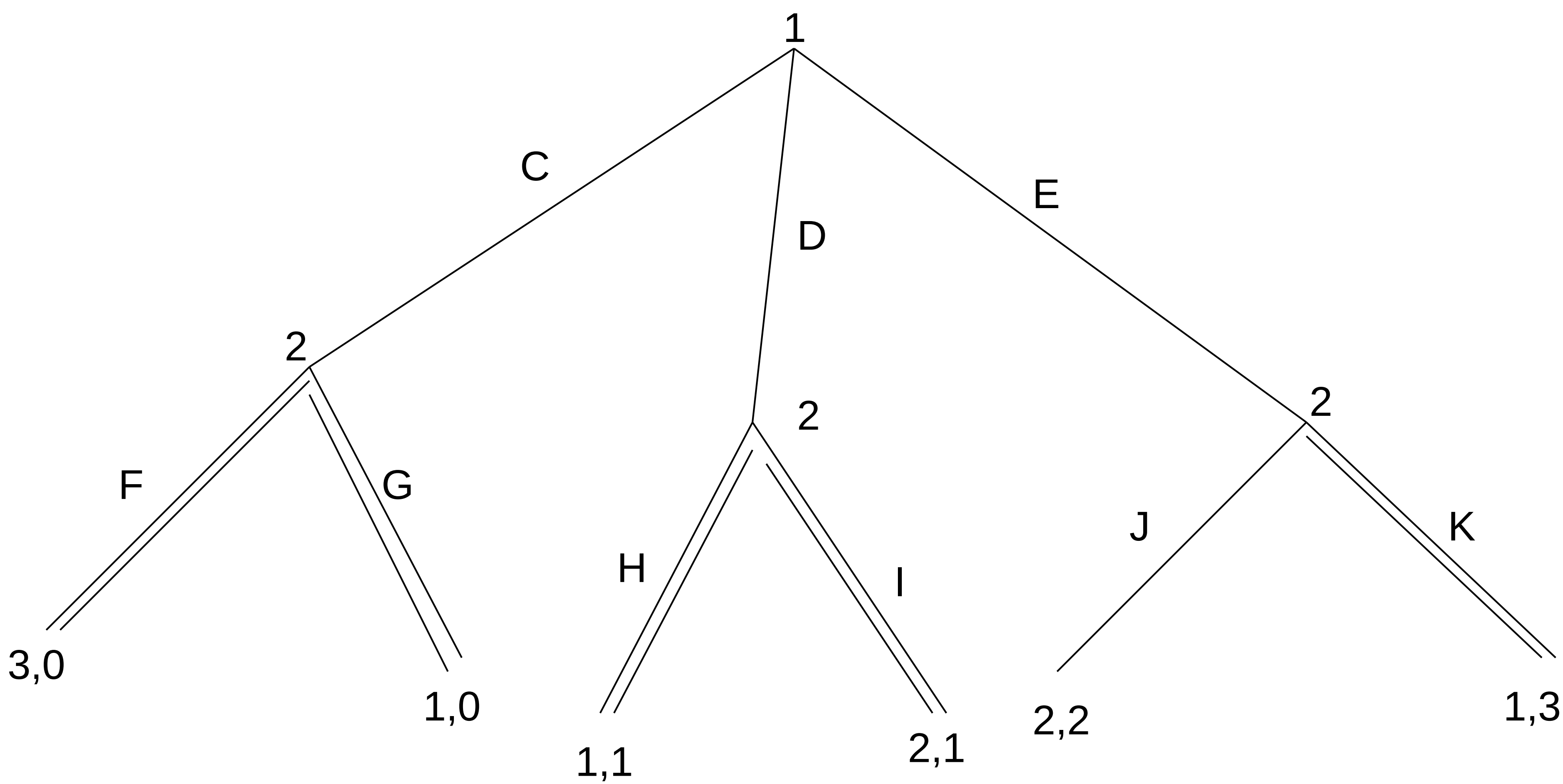
Diagram courtesy: Dr. Tayfun Sönmez

Multiple SPE



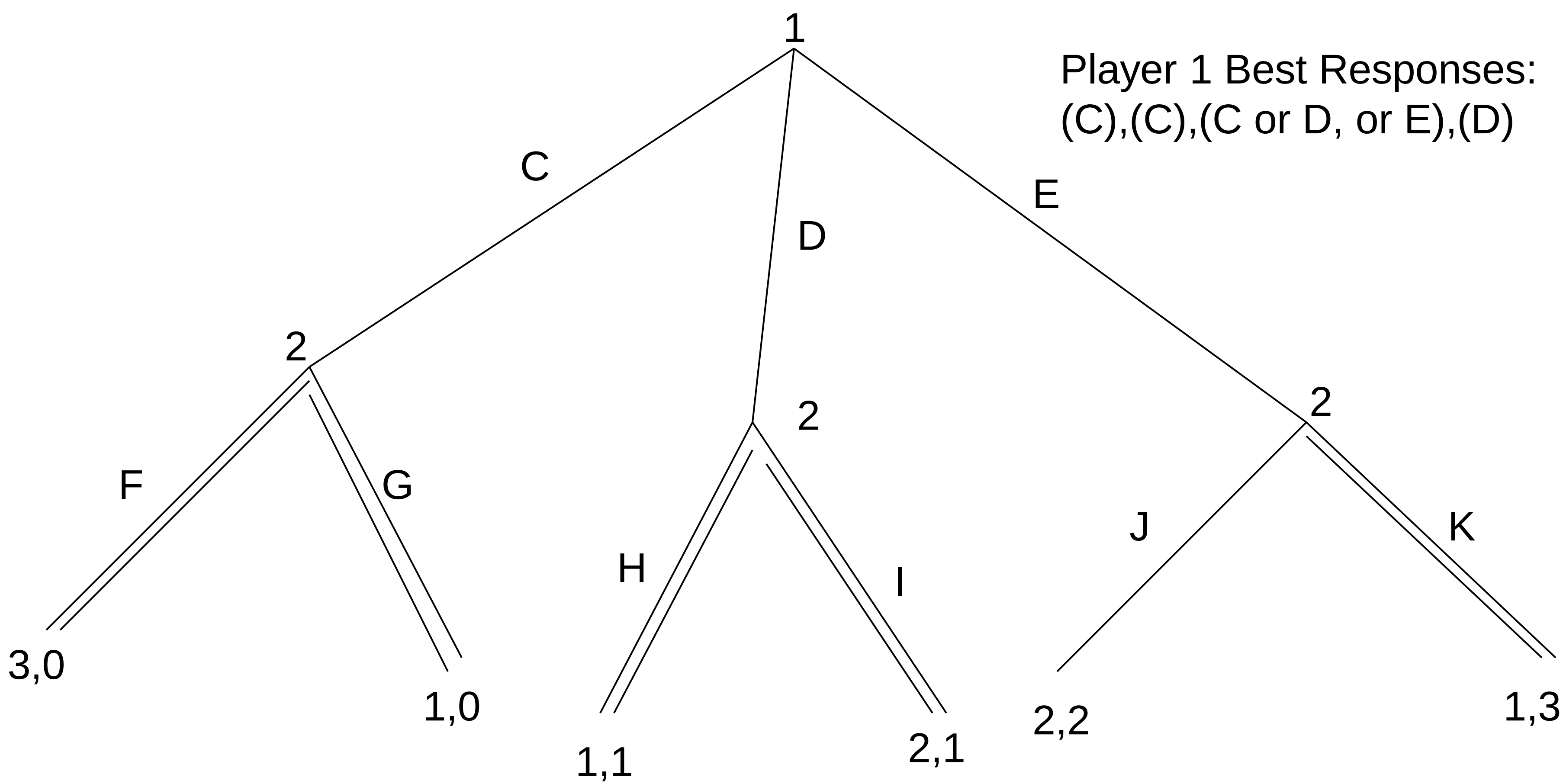
Multiple SPE

Player 2 Optimal Strategies:
(FHK),(FIK),(GHK),(GIK)



Multiple SPE

Player 2 Optimal Strategies:
(FHK),(FIK),(GHK),(GIK)



Player 1 Best Responses:
(C),(C),(C or D, or E),(D)

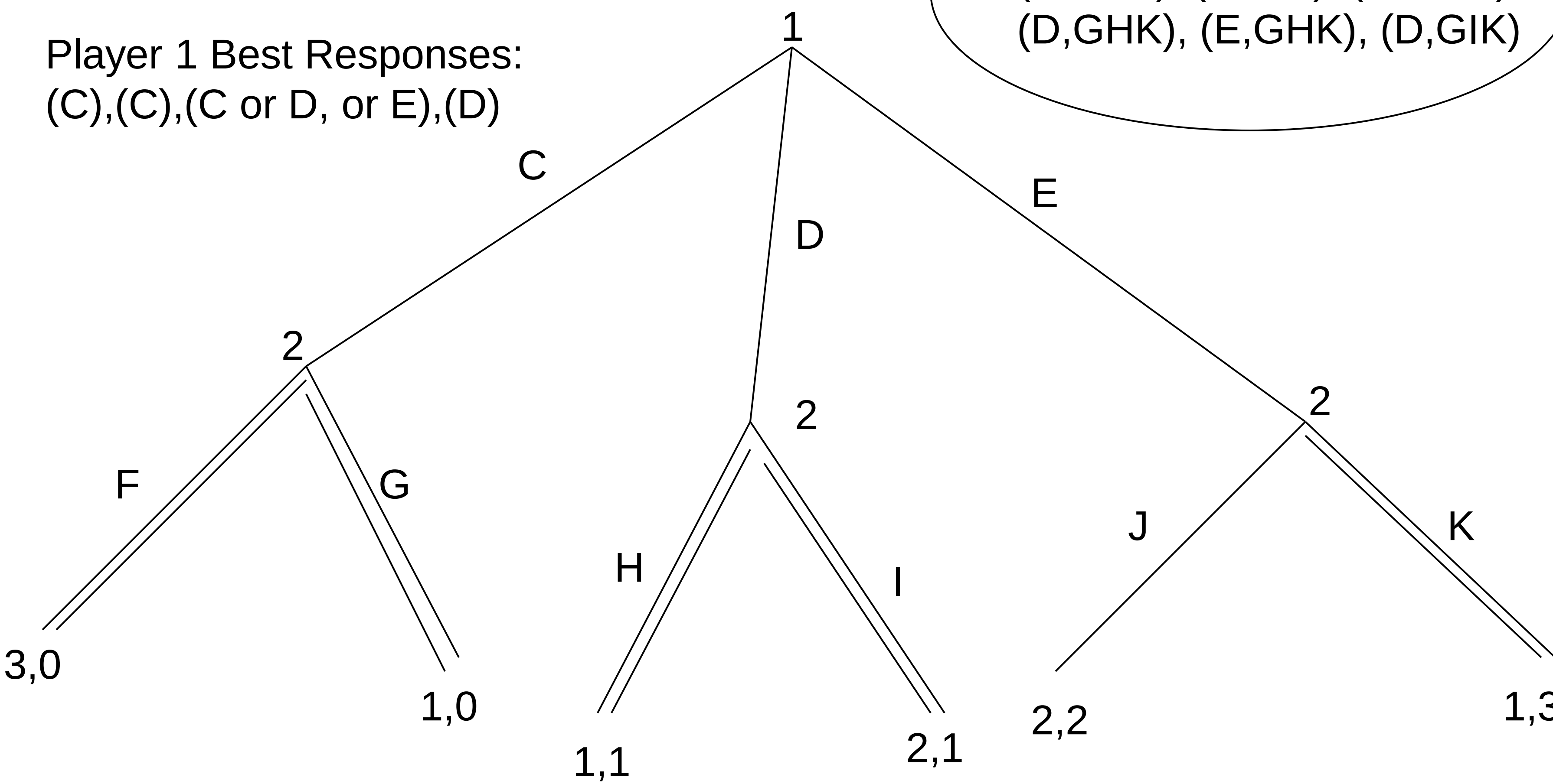
Multiple SPE

Player 2 Optimal Strategies:
(FHK),(FIK),(GHK),(GIK)

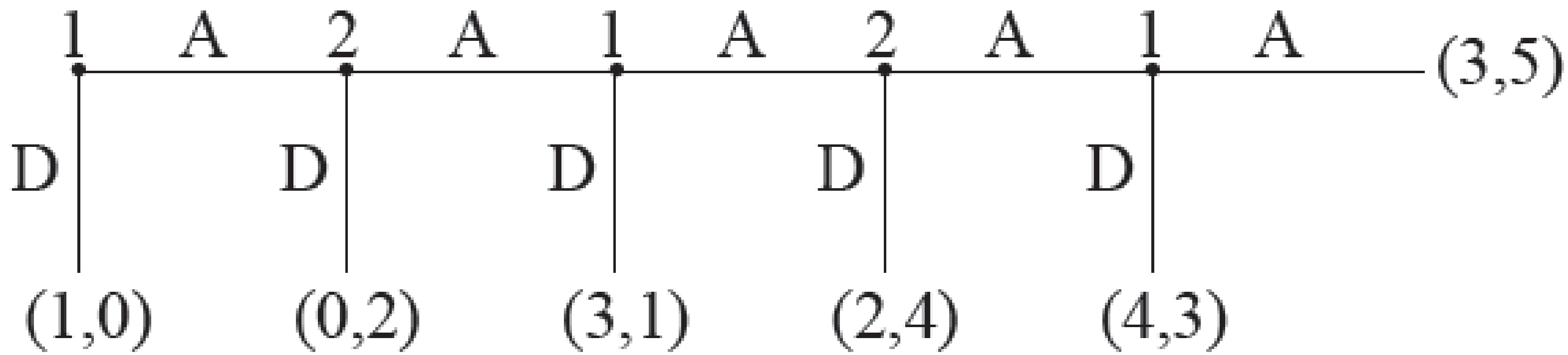
Player 1 Best Responses:
(C),(C),(C or D, or E),(D)

All SPE

Strategy Pairs:
(C,FHK), (C,FIK), (C,GHK),
(D,GHK), (E,GHK), (D,GIK)

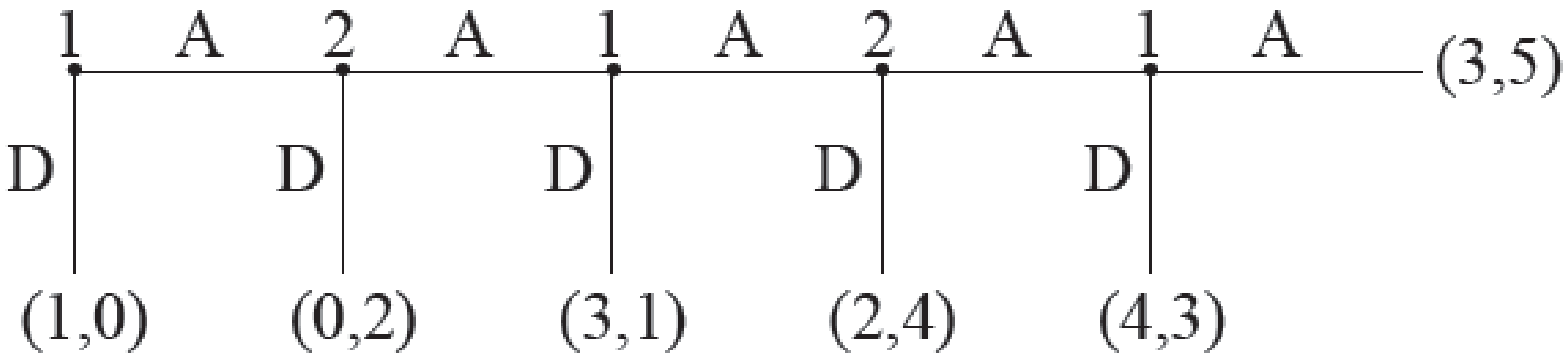


Centipede game



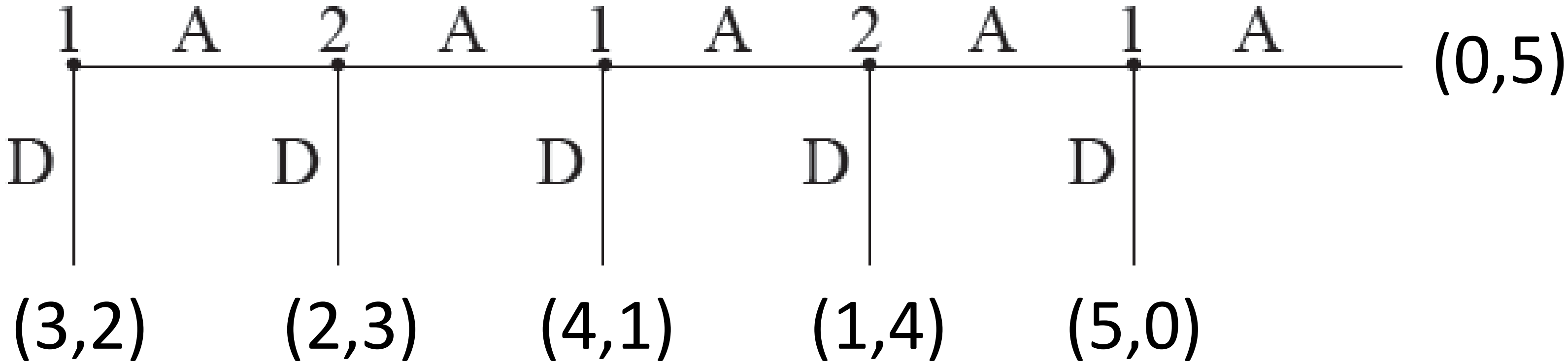
- What is the subgame-perfect equilibrium?
  - Player 1 chooses D at the first round
- Note that this outcome is Pareto-dominated by almost all the other outcomes.

Centipede game



- Will people play according to this SPE?
- What should player 2 do if player 1 chose A at the first round?
  - According to the SPE, he should choose D.
  - But player 1 was also supposed to choose D according to this rational.
  - In other words, you have reached a state to which your analysis has given a probability of zero.

Constant-Sum Centipede game



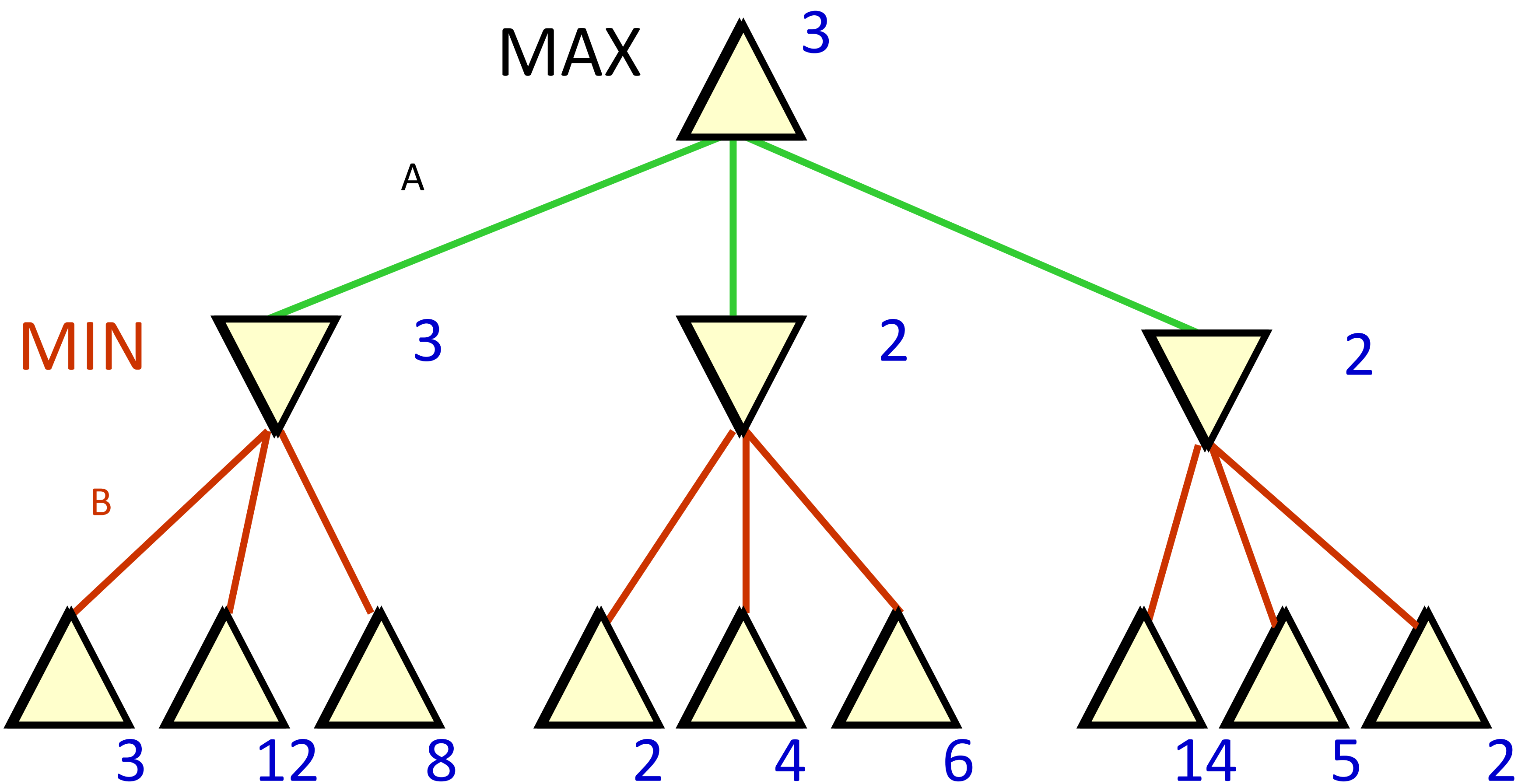
- Instead of having increasing payoffs for both players, the sum of their payoffs is always the same.
- In this case, backward induction gives much more accurate results.

Computing SPE, 2-player Zero-sum Games

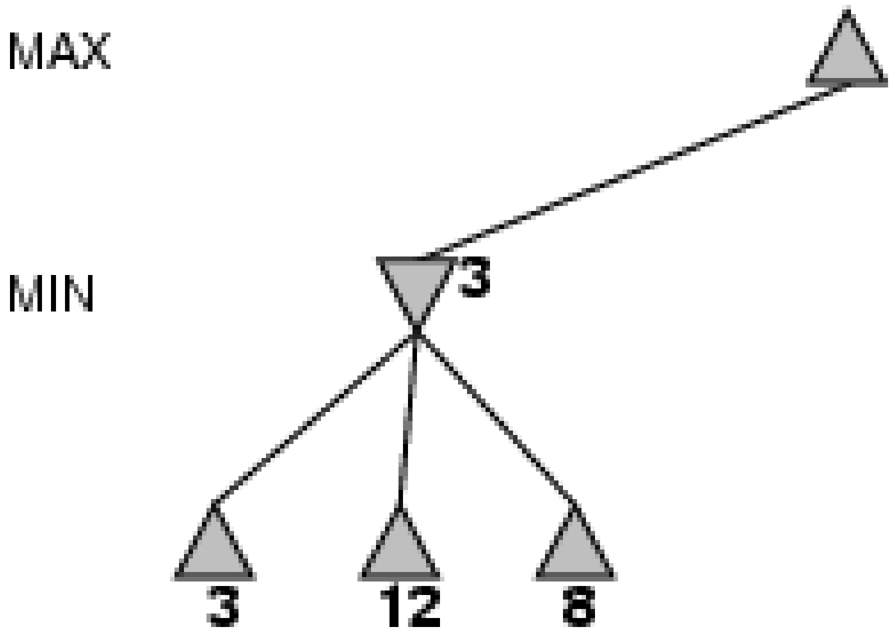
- Backward Induction has another name in the two-player, zero-sum context: the minimax algorithm.

- Recall that in such games, only a single payoff number is required to characterize any outcome:
  - Player 1 wants to maximize this number, while player 2 wants to minimize it.
  - We call player 1 the MAX player, and player 2 the MIN player.

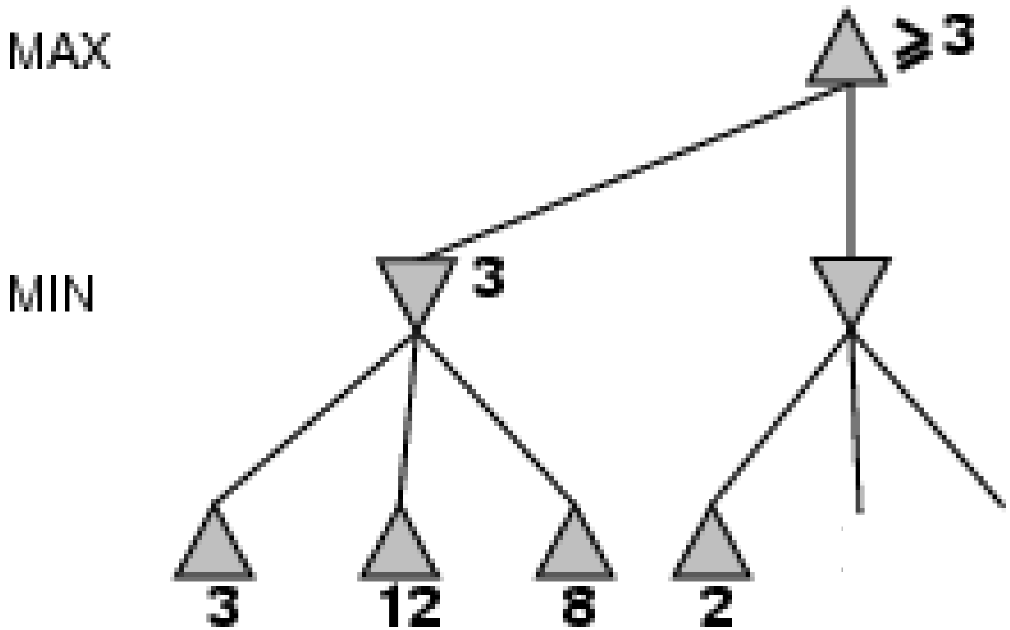
The Minimax Algorithm



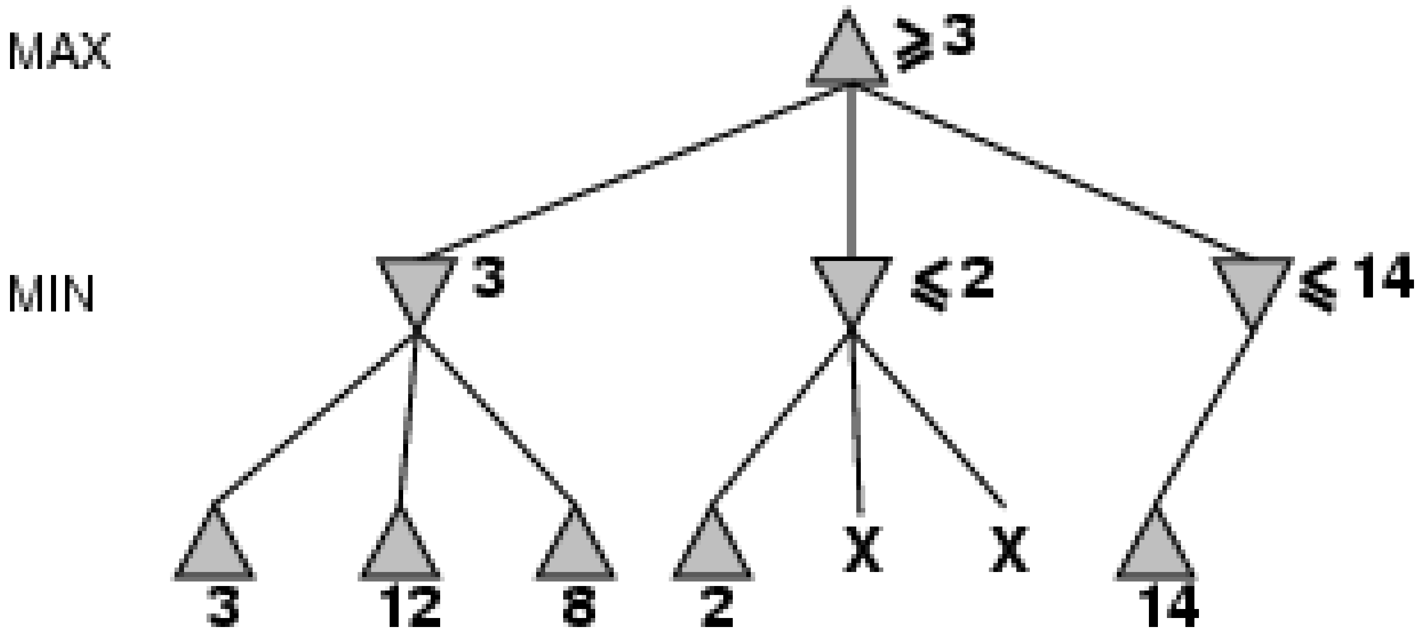
α-β pruning example



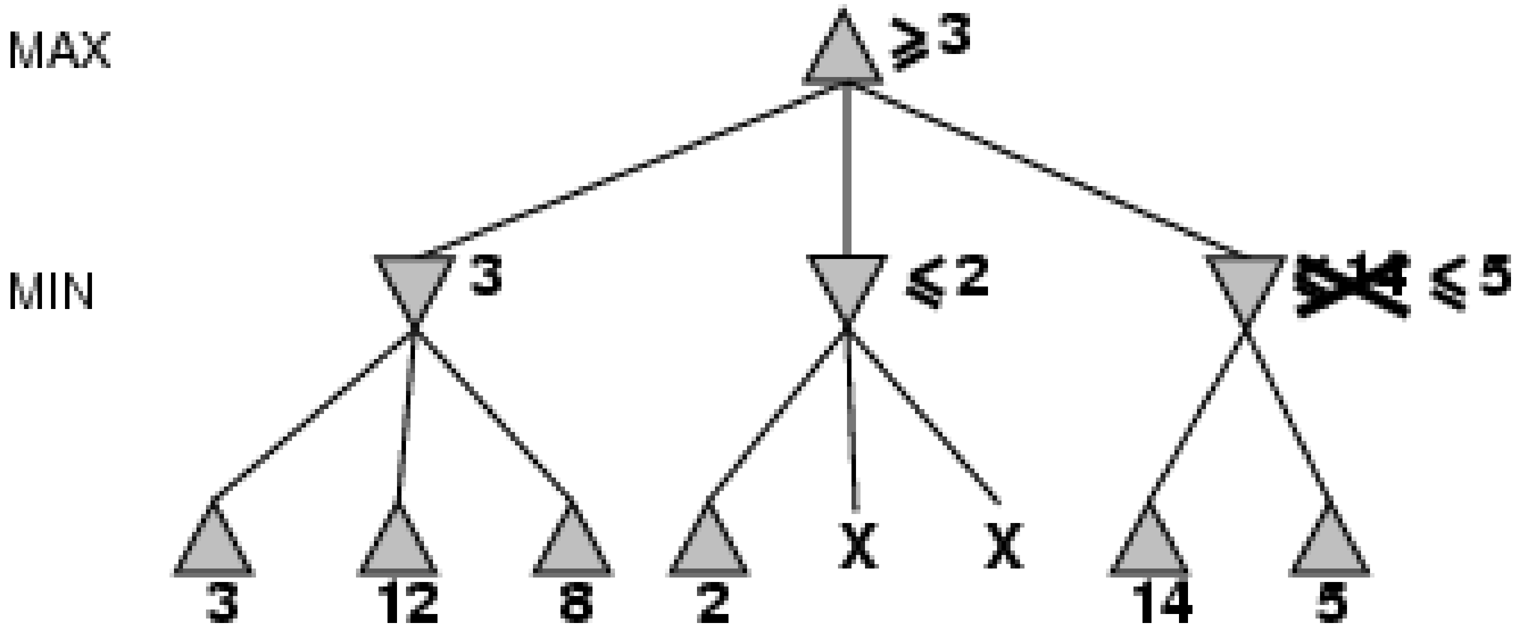
α-β pruning example



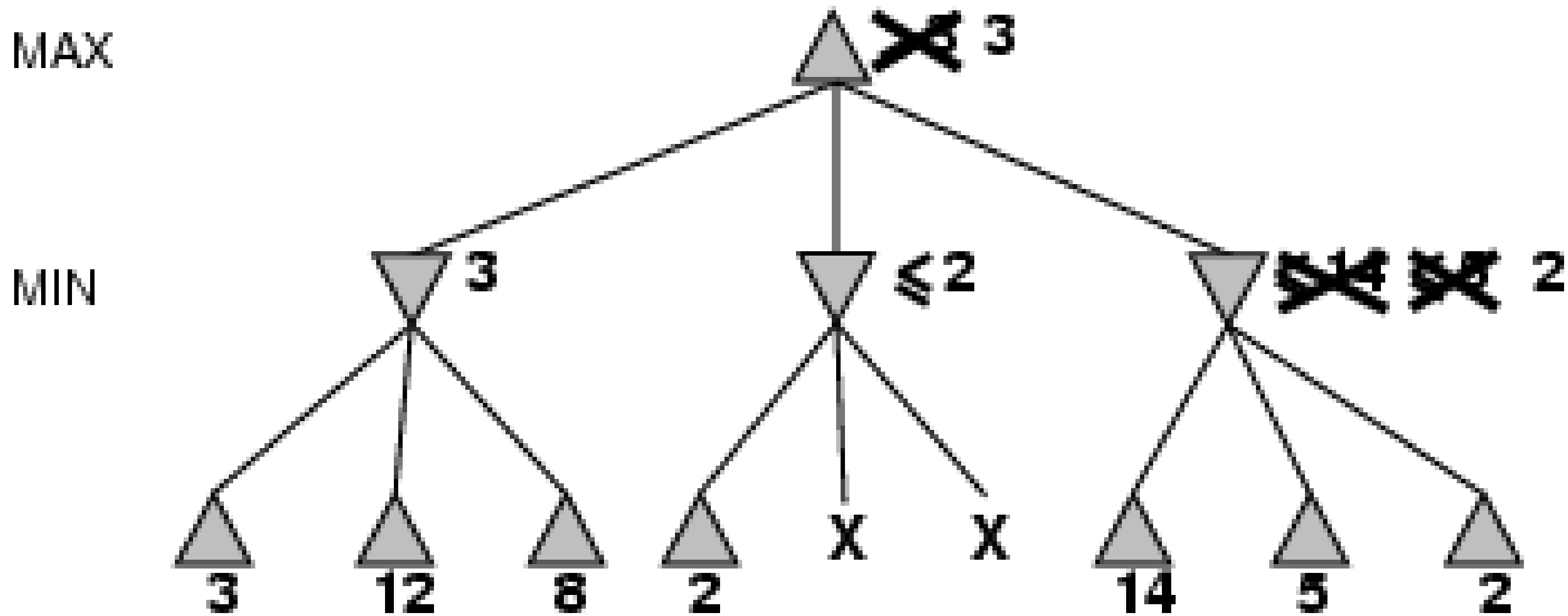
α-β pruning example



α-β pruning example

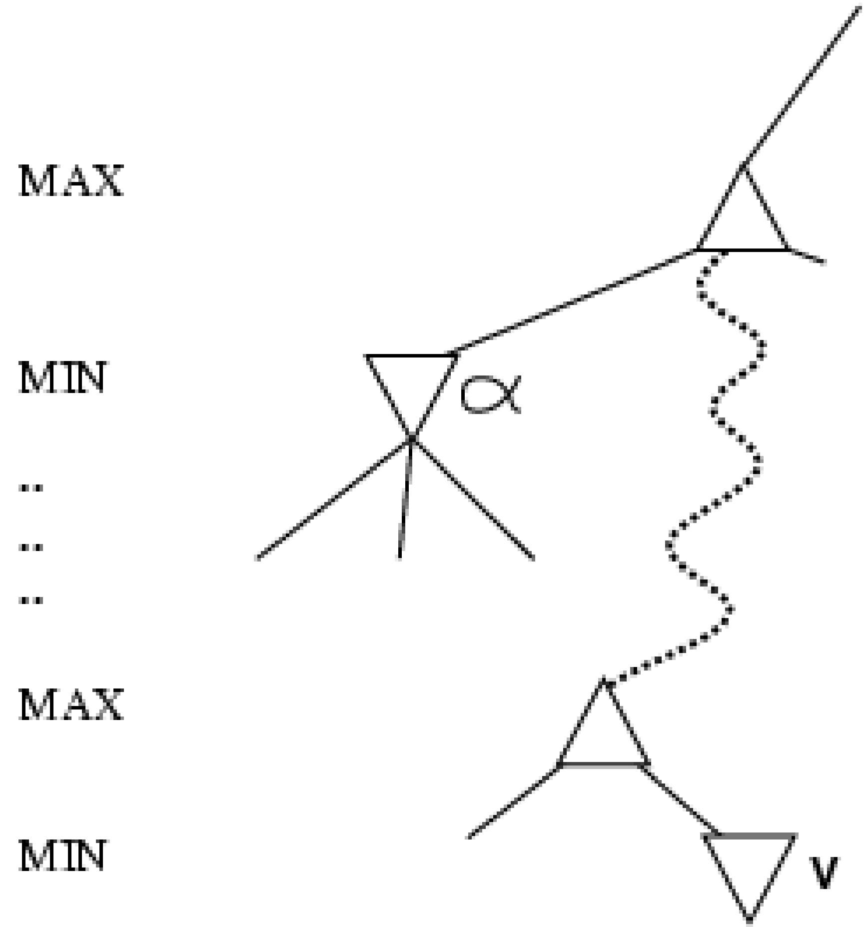


α-β pruning example



Why is it called α-β?

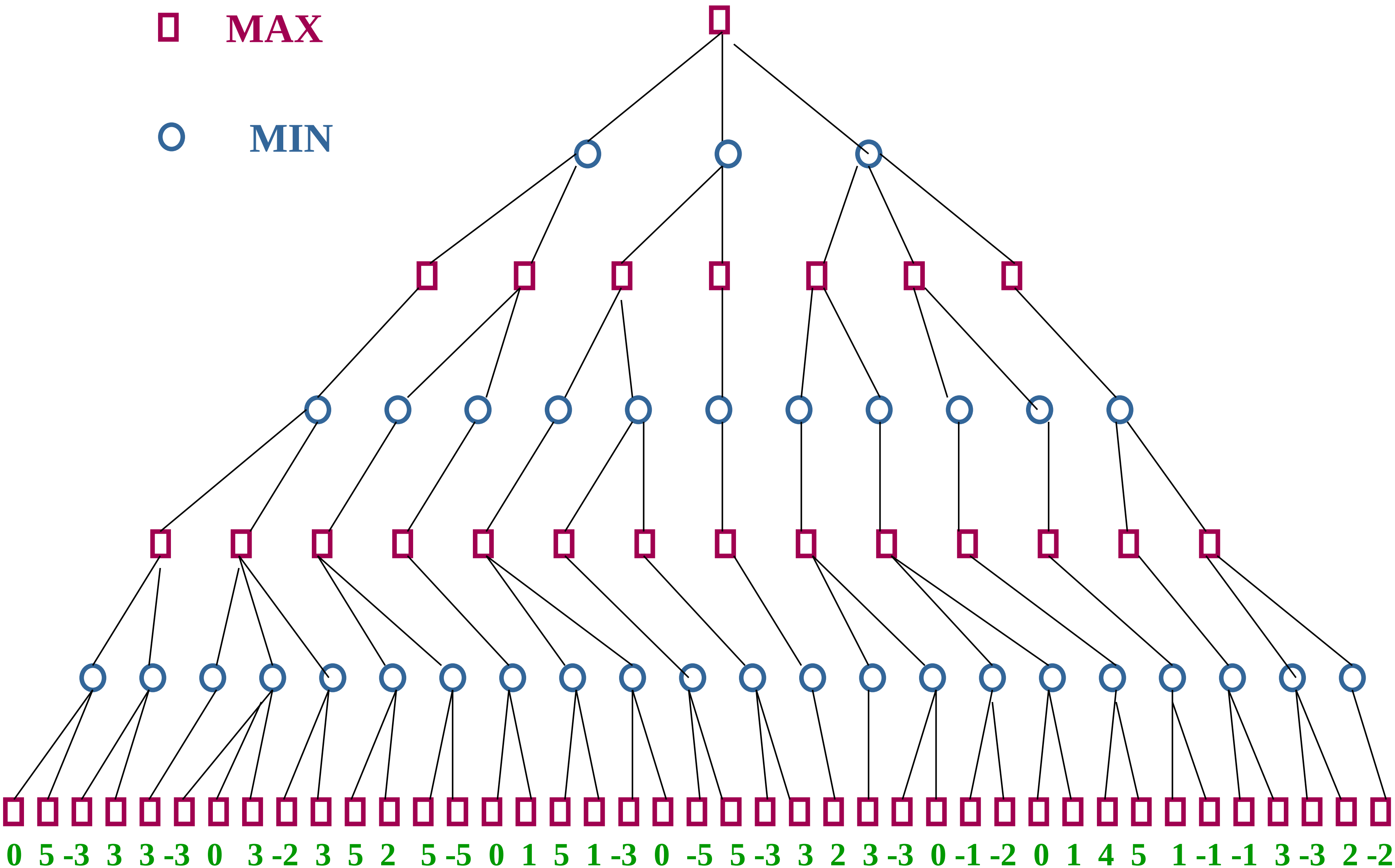
- α is the value of the best (i.e., highest-value) choice found so far at any choice point along the path for *max*
- If *v* is worse than α, *max* will avoid it
  - →prune that branch
- Define β similarly for *min*



The α-β algorithm

```
function ALPHABETAPRUNING (node h, real α, real β) returns u_1(h)
if h ∈ Z then
    return u_1(h)                                        // h is a terminal node
best_util ← (2ρ(h) − 3) × ∞                              // −∞ for player 1; ∞ for player 2
forall a ∈ χ(h) do
    if ρ(h) = 1 then
        best_util ← max(best_util, ALPHABETAPRUNING(σ(h, a), α, β))
        if best_util ≥ β then
            return best_util
        α ← max(α, best_util)
    else
        best_util ← min(best_util, ALPHABETAPRUNING(σ(h, a), α, β))
        if best_util ≤ α then
            return best_util
        β ← min(β, best_util)
return best_util
```

α- β pruning



α - βpruning