# Other Solution Concepts and Extensive-Form Games

Branislav Bošanský

Artificial Intelligence Center,
Department of Computer Science,
Faculty of Electrical Engineering,
Czech Technical University in Prague

*branislav.bosansky@agents.fel.cvut.cz*

October 22, 2019

Previously ... on multi-agent systems.

1. Mixed Strategies
2. Minimax Theorem
3. Linear Program for computing NE in zero-sum normal-form games

# Computing NE in Zero-Sum Games

We can now compute Nash equilibrium for two-player, zero-sum games using a linear programming:

$$\max_{s,U} \quad U \tag{1}$$

$$\text{s.t.} \quad \sum_{a_1 \in \mathcal{A}_1} s(a_1) u_1(a_1, a_2) \geq U \qquad \forall a_2 \in \mathcal{A}_2 \tag{2}$$

$$\sum_{a_1 \in \mathcal{A}_1} s(a_1) = 1 \tag{3}$$

$$s(a_1) \geq 0 \qquad \forall a_1 \in \mathcal{A}_1 \tag{4}$$

Computing a Nash equilibrium in zero-sum normal-form games can be done in polynomial time.

# Computing NE in General-Sum Games

The problem is more complex for general-sum games (LCP program):

$$
\begin{aligned}
\sum_{a_2 \in \mathcal{A}_2} u_1(a_1, a_2) s_2(a_2) + q(a_1) = U_1 \qquad & \forall a_1 \in \mathcal{A}_1 \\
\sum_{a_1 \in \mathcal{A}_1} u_2(a_1, a_2) s_1(a_1) + w(a_2) = U_2 \qquad & \forall a_2 \in \mathcal{A}_2 \\
\sum_{a_1 \in \mathcal{A}_1} s_1(a_1) = 1 \qquad \sum_{a_2 \in \mathcal{A}_2} s_2(a_2) = 1 & \\
q(a_1) \geq 0,\ w(a_2) \geq 0,\ s_1(a_1) \geq 0,\ s_2(a_2) \geq 0 \qquad & \forall a_1 \in \mathcal{A}_1, \forall a_2 \in \mathcal{A}_2 \\
s_1(a_1) \cdot q(a_1) = 0,\ s_2(a_2) \cdot w(a_2) = 0 \qquad & \forall a_1 \in \mathcal{A}_1, \forall a_2 \in \mathcal{A}_2
\end{aligned}
$$

Computing a Nash equilibrium in two-player general-sum normal-form game is a PPAD-complete problem. The problem gets even more complex (FIXP-hard) when moving to $n \geq 3$ players.

# Regret

The concept of regret is useful when the other players are not completely malicious.

| | **L** | **R** |
|---|---|---|
| **U** | $(100, a)$ | $(1-\varepsilon, b)$ |
| **D** | $(2, c)$ | $(1, d)$ |

## Definition (Regret)

A player $i$'s *regret* for playing an action $a_i$ if the other agents adopt action profile $a_{-i}$ is defined as

$$\left[\max_{a_i' \in \mathcal{A}_i} u_i(a_i', a_{-i})\right] - u_i(a_i, a_{-i})$$

# Regret

## Definition (MaxRegret)

A player $i$s *maximum regret* for playing an action $a_i$ is defined as

$$\max_{a_{-i} \in \mathcal{A}_{-i}} \left( \left[ \max_{a'_i \in \mathcal{A}_i} u_i(a'_i, a_{-i}) \right] - u_i(a_i, a_{-i}) \right)$$

## Definition (MinimaxRegret)

Minimax regret actions for player $i$ are defined as

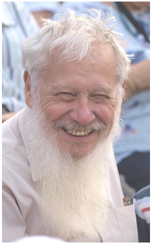
$$\arg\min_{a_i \in \mathcal{A}_i} \max_{a_{-i} \in \mathcal{A}_{-i}} \left( \left[ \max_{a'_i \in \mathcal{A}_i} u_i(a'_i, a_{-i}) \right] - u_i(a_i, a_{-i}) \right)$$

# Correlated Equilibrium

Consider again the following game:

| | **L** | **R** |
|---|---|---|
| **U** | $(2,1)$ | $(0,0)$ |
| **D** | $(0,0)$ | $(1,2)$ |

Wouldn't it be better to coordinate 50:50 between the outcomes (U,L) and (D,R)? Can we achieve this coordination? We can use *a correlation device*—a coin, a streetlight, commonly observed signal—and use this signal to avoid unwanted outcomes.


Robert Aumann

# Correlated Equilibrium

## Definition (Correlated Equilibrium (simplified))

Let $\mathcal{G} = (\mathcal{N}, \mathcal{A}, u)$ be a normal-form game and let $\sigma$ be a probability distribution over joint pure strategy profiles $\sigma \in \Delta(\mathcal{A})$. We say that $\sigma$ is a correlated equilibrium if for every player $i$, every signal $a_i \in \mathcal{A}_i$ and every possible action $a'_i \in \mathcal{A}_i$ it holds

$$\sum_{a_{-i} \in \mathcal{A}_{-i}} \sigma(a_i, a_{-i}) u_i(a_i, a_{-i}) \geq \sum_{a_{-i} \in \mathcal{A}_{-i}} \sigma(a_i, a_{-i}) u_i(a'_i, a_{-i})$$

## Corollary

*For every Nash equilibrium there exists a corresponding Correlated Equilibrium.*

# Computing Correlated Equilibrium

Computing a Correlated equilibrium is easier compared to Nash and can be found by linear programming even in general-sum case:

$$\sum_{a_{-i} \in \mathcal{A}_{-i}} \sigma(a_i, a_{-i}) u_i(a_i, a_{-i}) \geq \sum_{a_{-i} \in \mathcal{A}_{-i}} \sigma(a_i, a_{-i}) u_i(a_i', a_{-i}) \qquad \forall i \in \mathcal{N}, \forall a_i, a_i' \in \mathcal{A}_i$$

$$\sum_{a \in \mathcal{A}} \sigma(a) = 1 \qquad \sigma(a) \geq 0 \qquad \forall a \in \mathcal{A}$$

# Stackelberg Equilibrium

Finally, consider a situation where an agent is a central public authority (police, government, etc.) that needs to design and publish a policy that will be observed and reacted to by other agents.

- *the leader* – publicly commits to a strategy
- *the follower(s)* – play a Nash equilibrium with respect to the commitment of the leader

Stackelberg equilibrium is a strategy profile that satisfies the above conditions and maximizes the expected utility value of the leader:

$$\arg\max_{s \in \mathcal{S}; \forall i \in \mathcal{N} \setminus \{1\} s_i \in BR_i(s_{-i})} u_1(s)$$

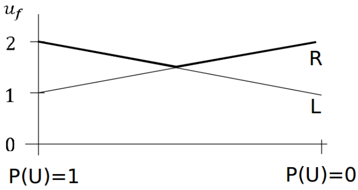
# Stackelberg Equilibrium

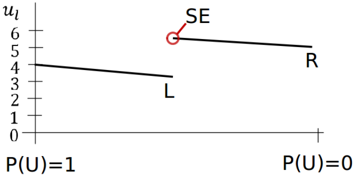
Consider the following game:

| | $\mathbf{L}$ | $\mathbf{R}$ |
|---|---|---|
| $\mathbf{U}$ | $(4,2)$ | $(6,1)$ |
| $\mathbf{D}$ | $(3,1)$ | $(5,2)$ |

$(\mathbf{U},\mathbf{L})$ is a Nash equilibrium.

What happens when the row player commits to play strategy $\mathbf{D}$ with probability $1$? Can the row player get even more?



# Computing a Stackelberg equilibrium in NFGs

The problem is polynomial for two-players normal-form games; $1$ is the leader, $2$ is the follower.

Baseline polynomial algorithm requires solving $|\mathcal{A}_2|$ linear programs:

$$\max_{s_1 \in \mathcal{S}_1} \sum_{a_1 \in \mathcal{A}_1} s_1(a_1) u_1(a_1, a_2)$$

$$\sum_{a_1 \in \mathcal{A}_1} s_1(a_1) u_2(a_1, a_2) \geq \sum_{a_1 \in \mathcal{A}_1} s_1(a_1) u_2(a_1, a_2') \quad \forall a_2' \in \mathcal{A}_2$$

$$\sum_{a_1 \in \mathcal{A}_1} s_1(a_1) = 1$$

one for each $a_2 \in \mathcal{A}_2$ assuming $a_2$ is the best response of the follower.

# Beyond Normal-Form Representations

One representation does not rule them all

# Beyond Normal-Form Representations

# Beyond Normal-Form Representations

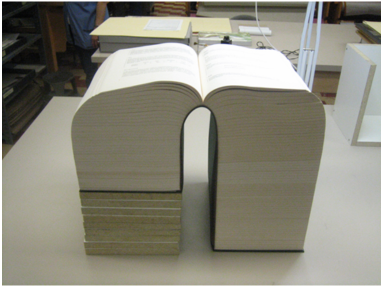
We can represent such dynamic scenarios using the normal-form representation.

A strategy in a dynamic game has to reflect all possible situations we can encounter in a game (including due to the moves by the opponent and/or stochastic events). Therefore, we need to have an action prescribed to be played in each situation that can happen.

The obvious drawback of using this representation is that there is exponentially many possible strategies given a description of the game.

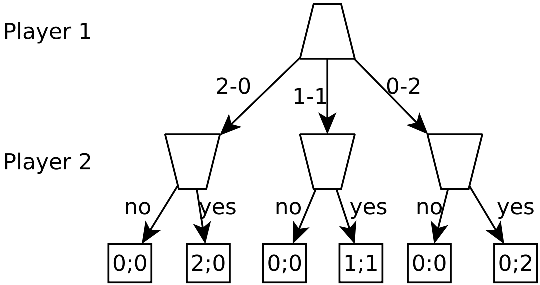
# Strategies in Sequential Games

A strategy in a dynamic game has to reflect all possible situations we can encounter in a game (including due to the moves by the opponent and/or stochastic events). Therefore, we need to have an action prescribed to be played in each situation that can happen.



# Extensive-Form Representation

We can use a more compact representation that is suitable for finite games termed *extensive-form games*.



# Extensive-Form Games (EFGs)

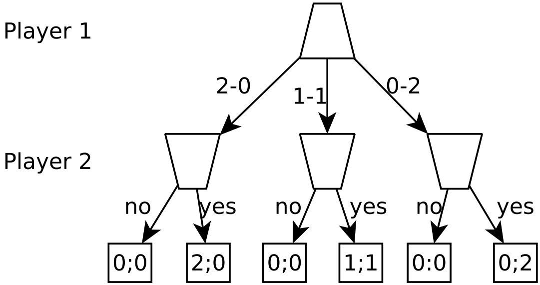
Formal Definition:

- players $\mathcal{N} = \{1, 2, \ldots, n\}$
- actions $\mathcal{A}$
- choice nodes (histories) $\mathcal{H}$
- action function $\chi : \mathcal{H} \to 2^{\mathcal{A}}$
- player function $\rho : \mathcal{H} \to \mathcal{N}$
- terminal nodes $\mathcal{Z}$
- successor function $\varphi : \mathcal{H} \times \mathcal{A} \to \mathcal{H} \cup \mathcal{Z}$
- utility function $u = (u_1, u_2, \ldots, u_n)$; $u_i : \mathcal{Z} \to \mathbb{R}$

A pure strategy of player $i$ in an EFG is an assignment of an action for each state where player $i$ acts

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

# Strategies in EFGs

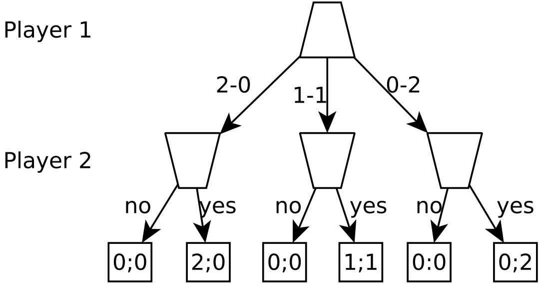



What are actions and strategies in this game?

$$\mathcal{A}_1 = \{2-0, 1-1, 0-2\};\ \mathcal{S}_1 = \{2-0, 1-1, 0-2\}$$

$$\mathcal{A}_2 = \{no, yes\};\ \mathcal{S}_2 = \{(no, no, no), (no, no, yes), \ldots, (yes, yes, yes)\}$$

# Strategies in EFGs

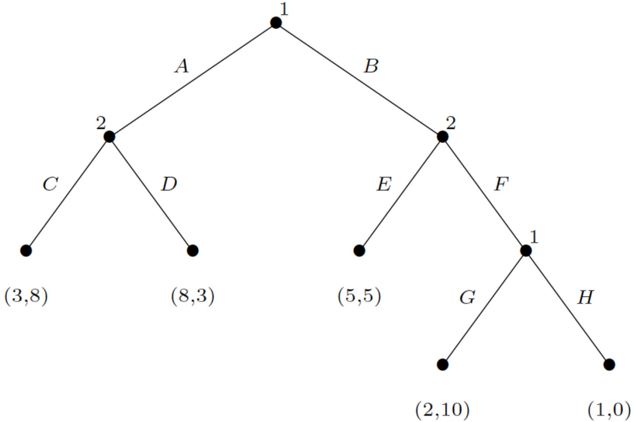



We can replace the function $\chi$ by multiplying actions so that an action can be applied only in a single state.

$$\mathcal{A}_2 = \{no_{\{2-0\}}, yes_{\{2-0\}}, no_{\{1-1\}}, yes_{\{1-1\}}, no_{\{0-2\}}, yes_{\{0-2\}}\};$$

$$\mathcal{S}_2 = \{(no_{\{2-0\}}, no_{\{1-1\}}, no_{\{0-2\}}), \ldots, (yes_{\{2-0\}}, yes_{\{1-1\}}, yes_{\{0-2\}})\}$$

# Strategies in EFGs

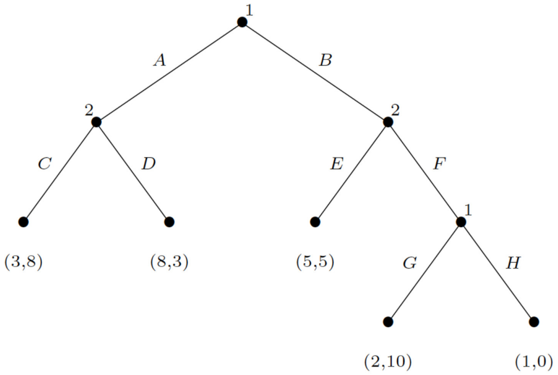



What are actions and strategies in this game?

$$\mathcal{S}_1 = \{(A, G), (A, H), (B, G), (B, H)\}$$

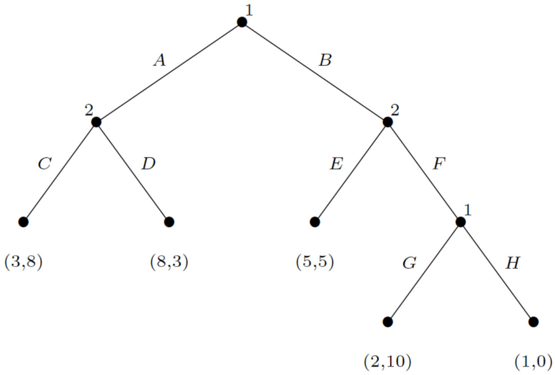
$$\mathcal{S}_2 = \{(C, E), (C, F), (D, E), (D, F)\}$$

# Induced Normal Form



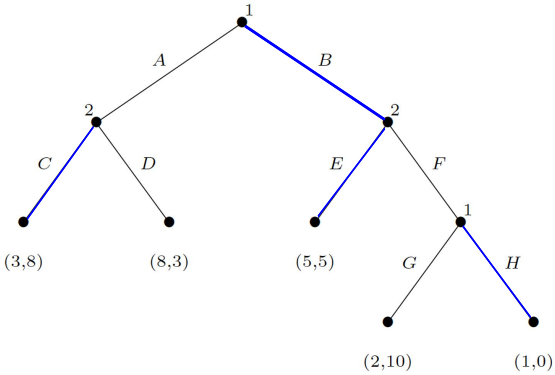

| | $(C,E)$ | $(C,F)$ | $(D,E)$ | $(D,F)$ |
|---|---|---|---|---|
| $(A,G)$ | $(3,8)$ | $(3,8)$ | $(8,3)$ | $(8,3)$ |
| $(A,H)$ | $(3,8)$ | $(3,8)$ | $(8,3)$ | $(8,3)$ |
| $(B,G)$ | $(5,5)$ | $(2,10)$ | $(5,5)$ | $(2,10)$ |
| $(B,H)$ | $(5,5)$ | $(1,0)$ | $(5,5)$ | $(1,0)$ |

# Nash Equilibria in EFGs




| | $(C, E)$ | $(C, F)$ | $(D, E)$ | $(D, F)$ |
|---|---|---|---|---|
| $(A, G)$ | $(3, 8)$ | $(\mathbf{3}, \mathbf{8})$ | $(8, 3)$ | $(8, 3)$ |
| $(A, H)$ | $(3, 8)$ | $(\mathbf{3}, \mathbf{8})$ | $(8, 3)$ | $(8, 3)$ |
| $(B, G)$ | $(5, 5)$ | $(2, 10)$ | $(5, 5)$ | $(2, 10)$ |
| $(B, H)$ | $(\mathbf{5}, \mathbf{5})$ | $(1, 0)$ | $(5, 5)$ | $(1, 0)$ |

# Nash Equilibria in EFGs - threats




| | $(C,E)$ | $(C,F)$ | $(D,E)$ | $(D,F)$ |
|---|---|---|---|---|
| $(A,G)$ | $(3,8)$ | $(\mathbf{3},\mathbf{8})$ | $(8,3)$ | $(8,3)$ |
| $(A,H)$ | $(3,8)$ | $(\mathbf{3},\mathbf{8})$ | $(8,3)$ | $(8,3)$ |
| $(B,G)$ | $(5,5)$ | $(2,10)$ | $(5,5)$ | $(2,10)$ |
| $(B,H)$ | $(\mathbf{5},\mathbf{5})$ | $(1,0)$ | $(5,5)$ | $(1,0)$ |

# Nash Equilibria in EFGs

Not all Nash strategies are entirely "sequentially rational" in EFGs. Off the equilibrium path, the players may use irrational actions.

We use *refinements of NE* in EFGs to avoid this. The best known (for EFGs with perfect information) is **Subgame-perfect equilibrium**.

**Definition (Subgame)**

Given a perfect-information extensive-form game $G$, the subgame of $G$ rooted at node $h$ is the restriction of $G$ to the descendants of $h$. The set of subgames of $G$ consists of all of subgames of $G$ rooted at some node in $G$.

# Subgame-Perfect Equilibrium (SPE)

**Definition (Subgame-perfect equilibrium)**

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 $G'$.

```
function BackwardInduction(node h)
    if h ∈ Z then
        return u(h)
    end if
    best_util ← ∞
    for all a ∈ χ(h) do
        util_at_child ← BackwardInduction(φ(h, a))
        if util_at_child_ρ(h) > best_util_ρ(h) then
            best_util ← util_at_child
        end if
    end for
end function
```

# Subgame-Perfect Equilibrium (SPE)

This is the same algorithm (in principle) that you know as Minimax (or Alpha-Beta pruning, or Negascout) and works (in general) for $n$-player games.

**Corollary**

*Every extensive-form game with perfect information has at least one Nash equilibria in pure strategies that is also a Subgame-perfect equilibrium.*

Is this correct? We have seen examples of games that do not have pure NE.

Not every game can be represented as an EFG with perfect information.

# EFGs with Chance

We introduce a new "player" termed chance (or Nature) that plays using a fixed randomized strategy.

Formal Definition:

- players $\mathcal{N} = \{1, 2, \ldots, n\} \cup \{c\}$
- actions $\mathcal{A}$
- choice nodes (histories) $\mathcal{H}$
- action function $\chi : \mathcal{H} \to 2^{\mathcal{A}}$
- player function $\rho : \mathcal{H} \to \mathcal{N}$
- terminal nodes $\mathcal{Z}$
- successor function $\varphi : \mathcal{H} \times \mathcal{A} \to \mathcal{H} \cup \mathcal{Z}$
- stochastic transitions $\gamma : \Delta\{\chi(h) \mid h \in \mathcal{H}, \rho(h) = c\}$
- utility function $u = (u_1, u_2, \ldots, u_n)$; $u_i : \mathcal{Z} \to \mathbb{R}$

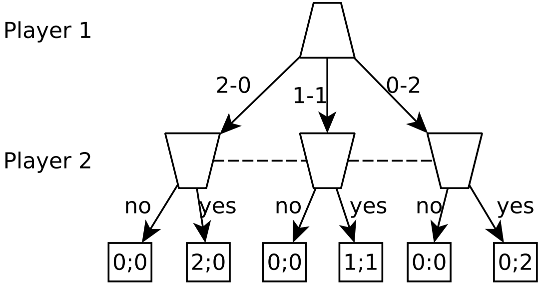
# EFGs with Imperfect Information

When players are not able to observe the state of the game perfectly, we talk about *imperfect information games*. The states that are not distinguishable to a player belong to a single *information set*.

Formal Definition:

- $\mathcal{G} = (\mathcal{N}, \mathcal{A}, \mathcal{H}, \mathcal{Z}, \chi, \rho, \varphi, \gamma, u)$ is a perfect-information EFG.
- $\mathcal{I} = (\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_n)$ where $\mathcal{I}_i$ is a set of equivalence classes on choice nodes of a player $i$ with the property that $\rho(h) = \rho(h') = i$ and $\chi(h) = \chi(h')$, whenever $h, h' \in I$ for some information set $I \in \mathcal{I}_i$
- we can use $\chi(I)$ instead of $\chi(h)$ for some $h \in I$

# Strategies in EFGs with Imperfect Information




What are actions and strategies in this game?

$$\mathcal{A}_1 = \{2-0, 1-1, 0-2\};\ \mathcal{S}_1 = \{2-0, 1-1, 0-2\}$$

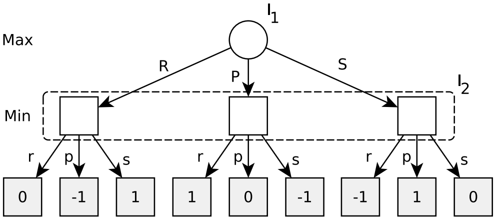
$$\mathcal{A}_2 = \{no, yes\};\ \mathcal{S}_2 = \{no, yes\}$$

# Strategies in EFGs with Imperfect Information

There are no guarantees that a pure NE exists in imperfect information games.

Every finite game can be represented as an EFG with imperfect information.

| | **R** | **P** | **S** |
|---|---|---|---|
| **R** | $(0, 0)$ | $(-1, 1)$ | $(1, -1)$ |
| **P** | $(1, -1)$ | $(0, 0)$ | $(-1, 1)$ |
| **S** | $(-1, 1)$ | $(1, -1)$ | $(0, 0)$ |



# Strategies in EFGs with Imperfect Information

Mixed strategies are defined as before as a probability distribution over pure strategies.

There are also other types of strategies in EFGs, namely *behavioral strategies*:

- A *behavioral strategy* of player $i$ is a product of probability distributions over actions in each information set

$$\beta_i : \prod_{I \in \mathcal{I}_I} \Delta(\chi(I))$$

There is a broad class of imperfect-information games in which the expressiveness of mixed and behavioral strategies coincide – *perfect recall games.* Informally, no player forgets any information she previously knew in these games.

# Perfect Recall in EFGs

## Definition

Player $i$ has perfect recall in an imperfect-information game $G$ if for any two nodes $h$,$h'$ that are in the same information set for player $i$, for any path $h_0, a_0, \ldots, h_n, a_n, h$ from the root of the game tree to $h$ and for any path $h_0, a'_0, \ldots, h'_m, a'_m, h'$ from the root to $h'$ it must be the case that:

1. $n = m$
2. for all $0 \leq j \leq n$, $h_j$ and $h'_j$ are in the same equivalence class for player $i$
3. for all $0 \leq j \leq n$, if $\rho(h_j) = i$, then $a_j = a'_j$

## Definition

We say that an EFG has a *perfect recall* if all players have perfect recall. Otherwise we say that the game has an *imperfect recall*.

# How to solve an EFG with imperfect information?

Does a backward induction work?

Does a limited-lookahead search work?

Existing algorithms:

- algorithms based on linear programming
- algorithms based on no-regret learning (reinforcement learning)
- algorithms based on convex optimization