COSC-6590/GSCS-6390

Games: Theory and Applications

Lecture 01 - Noncooperative Games

Luis Rodolfo Garcia Carrillo

School of Engineering and Computing Sciences
Texas A&M University - Corpus Christi, USA

August 28, 2018

Table of contents

Elements of a Game

Elements of a Game

To characterize a game one needs to specify several items:

- The **players** are the agents that make decisions.
- The **rules** define the actions allowed by the players and their effects.
- The **information structure** specifies what each player knows before making each decision.

Attention

Chess is a **full-information game** because the current state of the game is fully known to both players as they make their decisions. In contrast, Poker is a **partial-information** game.

- The **objective** specifies the goal of each player.

Elements of a Game

For a mathematical solution to a game, one further needs to make assumptions on the **player's rationality**, regarding questions such as:

- Will the players always pursue their best interests to fulfill their objectives? [YES]
- Will the players form coalitions? [NO]
- Will the players trust each other? [NO]

The answers in square brackets characterize what are usually called **noncooperative games**, and will be implicitly assumed throughout this course.

Note 1.- Human players

Studying noncooperative solutions for games played by humans reveals some lack of faith in human nature.

- When pursuing this approach one should not be surprised by finding **solutions of questionable ethics**.

Noncooperative game theory allows one to find problematic solutions to games, and often indicates how to **fix** the games so that these solutions disappear

- Example: mechanism design.

In ENCS problems, **players are modeling decision processes** not affected by human reason

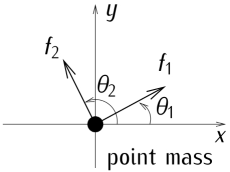
- one can pursue noncooperative solutions without questioning their ethical foundation.

Robust engineering designs and evolutionary biology are good examples of this.

Cooperative vs Noncooperative Games

Cooperative vs. Noncooperative Games: Rope-Pulling

The **Rope-Pulling** game is depicted schematically as



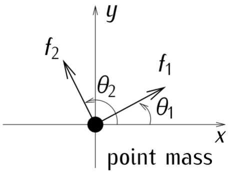
Rules:

- Two players push a mass by exerting on it forces f_1 and f_2.
- Players exert forces with the same magnitude: $|f_1| = |f_2|$.
- Players pull in different directions $\theta_1(t)$ and $\theta_2(t)$.
- The game is played for 1 second.

Note:

$\theta_1(t)$ and $\theta_2(t)$ correspond to the decisions made by the players.

Cooperative vs. Noncooperative Games: Rope-Pulling



Assume **unit forces** and a unit mass. Initially mass is at rest.

According to Newton's law, the point mass moves according to

$$\ddot{x} = \cos\theta_1(t) + \cos\theta_2(t), \qquad \dot{x}(0) = 0, \qquad x(0) = 0$$
$$\ddot{y} = \sin\theta_1(t) + \sin\theta_2(t), \qquad \dot{y}(0) = 0, \qquad y(0) = 0$$

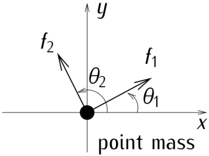
These equations encode the **rules** of the game: they determine how the player's **decisions** affect the **outcome** of the game.

Cooperative vs. Noncooperative Games: Rope-Pulling

Zero-Sum Rope-Pulling Game

Objective (zero-sum):

- P_1 wants to **maximize** $x(1)$
whereas
- P_2 wants to **minimize** $x(1)$.



Notation: zero-sum-game

A game where **players have opposite objectives**.

One could also imagine that

- P_1 wants to **maximize** $x(1)$
- P_2 wants to **maximize** $-x(1)$

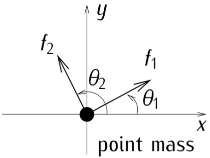
According to this view **the two objectives add up to zero**

Cooperative vs. Noncooperative Games: Rope-Pulling

Zero-Sum Rope-Pulling Game

Objective (zero-sum):

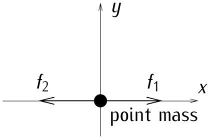
- P_1 wants to **maximize** $x(1)$
whereas
- P_2 wants to **minimize** $x(1)$.



Solution: the **optimal** solution for this game is given by

$$P_1 : \theta_1(t) = 0, \quad \forall t \in [0, 1], \qquad P_2 : \theta_2(t) = \pi, \quad \forall t \in [0, 1]$$

This results in **no motion**: $\ddot{x} = \ddot{y} = 0$,
leading to: $x(1) = y(1) = 0$



Cooperative vs. Noncooperative Games: Rope-Pulling

The following questions arise:
Is it reasonable to pull at all, given that the mass will not move?
Is the optimal solution **not to push at all**?

This is **not the case** for two reasons

1. Not pushing is not allowed by the rules of the game: each player must exert a force of one Newton.
2. Even if not pulling was an option, it is a **dangerous** choice for the player that decided to follow this action: the other player could take advantage of the situation.

Remember: in noncooperative games players do not trust each other and do not form coalitions.

But why our choice (below) is the **optimal solution**?

$$P_1 : \theta_1(t) = 0, \quad \forall t \in [0,1], \qquad\qquad P_2 : \theta_2(t) = \pi, \quad \forall t \in [0,1]$$

Cooperative vs. Noncooperative Games: Rope-Pulling

Consider now a **Non-Zero-Sum Rope-Pulling Game**

Objective (non-zero-sum):
- P_1 wants to **maximize** $x(1)$, whereas
- P_2 wants to **maximize** $y(1)$.

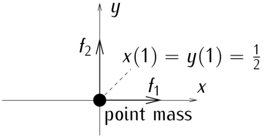
Attention: This is no longer a zero-sum game!

Solution (Nash): The **optimal** solution is given by

$$P_1 : \theta_1(t) = 0, \quad \forall t \in [0,1], \qquad P_2 : \theta_2(t) = \frac{\pi}{2}, \quad \forall t \in [0,1]$$

This leads to constant accelerations
$\ddot{x} = \ddot{y} = 1$
and therefore $x(1) = y(1) = \frac{1}{2}$

Remember: distance $= v_0 t + \frac{1}{2} a t^2$



Cooperative vs. Noncooperative Games: Rope-Pulling

This solution has **two important properties**:

P1.1.- Suppose P_1 follows the course of action $\theta_1(t) = 0$ throughout the whole time period and therefore

$$\ddot{x} = 1 + \cos\theta_2(t), \qquad \ddot{y} = \sin\theta_2(t), \qquad \forall t \in [0, 1].$$

Here, the best course of **action** for P_2 so as to maximize $y(1)$ is precisely to choose

$$\theta_2(t) = \frac{\pi}{2}, \qquad \forall t \in [0, 1] \qquad \Rightarrow \qquad \ddot{y}(t) = 1, \qquad \forall t \in [0, 1].$$

Any **deviation** from this will lead to a **smaller value** of $y(1)$.

Once P_1 decides to stick to its part of the solution, a rational P_2 must necessarily follow its policy.

Cooperative vs. Noncooperative Games: Rope-Pulling

This solution has **two important properties**:

P1.2.- Suppose P_2 follows the course of action $\theta_2(t) = \frac{\pi}{2}$ throughout the whole time period and therefore

$$\ddot{x} = \cos\theta_1(t), \qquad \ddot{y} = \sin\theta_1(t) + 1, \qquad \forall t \in [0, 1].$$

Here, the best course of action for P_1 so as to maximize $x(1)$ is precisely to choose

$$\theta_1(t) = 0, \qquad \forall t \in [0, 1] \qquad \Rightarrow \qquad \ddot{x}(t) = 1, \qquad \forall t \in [0, 1].$$

Any **deviation** from this will lead to a **smaller value** of $x(1)$.

Once P_2 decides to stick to its part of the solution, a rational P_1 must necessarily follow its policy.

Cooperative vs. Noncooperative Games: Rope-Pulling

A pair of policies that satisfy the above properties is called a **Nash equilibrium solution**.

Key feature of a Nash equilibrium solution: **it is stable**

- If the two players start playing at the Nash equilibrium, none of the players gains from deviating from these policies.

Notation: solution

In games a **solution** is a set of policy, one for each player, that jointly satisfy some **optimality condition**.

Policy: a course or principle of action.

Cooperative vs. Noncooperative Games: Rope-Pulling

The solution $(\theta_1(t), \theta_2(t)) = (0, \frac{\pi}{2})$ also satisfies these properties:

P1.3.- Suppose that P_1 follows the course of action $\theta_1(t) = 0$ throughout the whole time period.

- Regardless of what P_2 does, P_1 is guaranteed to achieve $x(1) \geq 0$.
- No other policy for P_1 can guarantee a larger value for $x(1)$ regardless of what P_2 does.

Note: Even if P_2 pulls against P_1, which is not very rational but possible.

Cooperative vs. Noncooperative Games: Rope-Pulling

The solution $(\theta_1(t), \theta_2(t)) = (0, \frac{\pi}{2})$ also satisfies these properties:

P1.4.- Suppose P_2 follows the course of action $\theta_2(t) = \frac{\pi}{2}$ throughout the whole time period.

- Regardless of what P_1 does, P_2 is guaranteed to achieve $y(1) \geq 0$.
- No other policy for P_2 can guarantee a larger value for $y(1)$ regardless of what P_1 does.

In view of this, the two **policies** are also called **security policies** for the corresponding player.

The solution is **interesting** in two distinct senses

- these policies form a Nash equilibrium, per **P1.1** and **P1.2**
- these policies are also security policies, per **P1.3** and **P1.4**

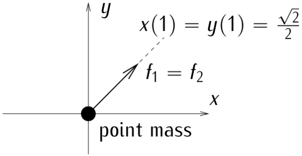
Cooperative vs. Noncooperative Games: Rope-Pulling

Solution (cooperative): consider the following solution

$$P_1 : \theta_1(t) = \frac{\pi}{4}, \quad \forall t \in [0,1], \qquad P_2 : \theta_2(t) = \frac{\pi}{4}, \quad \forall t \in [0,1]$$

leading to constant accelerations $\ddot{x} = \ddot{y} = \sqrt{2}$, and therefore

$$x(1) = y(1) = \frac{\sqrt{2}}{2} > \frac{1}{2}$$



This policy is interesting: **both players do strictly better** than with the Nash policies.

- However, this is not a Nash policy!

Cooperative vs. Noncooperative Games: Rope-Pulling

Suppose that P_1 decides to follow this action $\theta_1(t) = \frac{\pi}{4}$, then

$$\ddot{x} = \frac{\sqrt{2}}{2} + \cos\theta_2(t), \qquad \ddot{y} = \frac{\sqrt{2}}{2} + \sin\theta_2(t), \qquad \forall t \in [0,1]$$

In this case, the best action for P_2 to maximize $y(1)$ is to choose

$$\theta_2(t) = \frac{\pi}{2}, \qquad \forall t \in [0,1],$$

instead of her assigned policy, because this will lead to

$$\ddot{y} = \frac{\sqrt{2}}{2} + 1 \qquad \text{and} \qquad y(1) = \frac{\sqrt{2}+2}{4} > \frac{\sqrt{2}}{2}$$

Unfortunately for P_1, this also leads to

$$\ddot{x} = \frac{\sqrt{2}}{2} \qquad \text{and} \qquad x(1) = \frac{\sqrt{2}}{4} < \frac{1}{2} < \frac{\sqrt{2}}{2}$$

Cooperative vs. Noncooperative Games: Rope-Pulling

The cooperative solution is a **dangerous choice** for P_1

- a **greedy** P_2 will get P_1 even **worse** than with the Nash policy that led to $x(1) = \frac{1}{2}$

Similarly, the cooperative solution is a dangerous choice for P_2.

The **cooperative solution** is not a Nash equilibrium solution,

- despite the fact that both players can do better than with the Nash solution.

Cooperative game theory deals with Cooperative Solutions:

- players negotiate to reach a mutually beneficial solution.
- requires faith/trust among the players.

Solutions arising from cooperation are **not robust** with respect to cheating by one of the players.

Cooperative vs. Noncooperative Games: Rope-Pulling

For certain classes of games, **noncooperative solutions coincide with cooperative solutions**,

- by pursuing ones selfish interests one actually helps other players in achieving their goals.

These games are highly desirable from a social perspective.

It is possible to **reshape** the reward structure of a game to make this happen.

- in economics this is often achieved through pricing, taxation, or other incentives/deterrents.
- in engineering it relates to Mechanism Design.

A network administrator can minimize the total interference between "selfish" wireless users by carefully charging their use of the shared medium.

Cooperative vs. Noncooperative Games: Rope-Pulling

Note 2. Pareto-optimal Solution

A cooperative solution like

$$P_1 : \theta_1(t) = \frac{\pi}{4}, \quad \forall t \in [0,1], \qquad P_2 : \theta_2(t) = \frac{\pi}{4}, \quad \forall t \in [0,1]$$

is called Pareto-optimal because **it is not possible to further improve the gain of one player without reducing the gain of the other**

For the non-zero-sum rope-pulling game, all Pareto-optimal solutions are found by solving the constrained optimization

$$\max_{\theta_1,\theta_2} \{x(1) : y(1) \geq \alpha\} \qquad \text{with} \qquad \alpha \in \mathbb{R}$$

Pareto-optimal solutions are generally **not unique**

- different α result in different pareto-optimal solutions.

Cooperative vs. Noncooperative Games: Rope-Pulling

Note: In some cases, all Pareto-optimal solutions can be found by solving unconstrained optimization problems.

For this example, all Pareto-optimal solutions are found by solving

$$\max_{\theta_1,\theta_2} \beta x(1) + (1-\beta) y(1)$$

and picking different values for β in the interval [0,1].

Robust Designs

Robust Designs

Game theory is used in engineering applications as tool to solve design problems that do not start as a game.

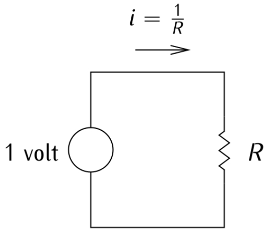
- **Step 1:** take the original design problem.
- **Step 2:** "discover" a game theoretical formulation that leads to a desirable solution.

In these games, the **players** are

- the **system designer**
- the **opponent**: a fictitious entity that tries to challenge the choices of the designer.

Robust Designs: Resistive Circuit

Goal: pick a resistor so that the current $i = \frac{1}{R}$ is as close as possible to 1.



Challenge: for a resistor with nominal resistance $= R_{\text{nom}}$, the actual resistance R may exhibit an error up to 10%, i.e.,

$$R = (1 + \delta)R_{\text{nom}}$$

where δ is an unknown scalar in the interval [-0.1,0.1].

Robust Designs: Resistive Circuit

This is a **robust design problem**: a game between the **circuit designer** and an **unforgiving nature** that does her best to foil the designer's objective:

- P_1: the circuit **designer**. Picks the nominal resistance R_{nom} to **minimize** the current error

$$e = \left|\frac{1}{R} - 1\right| = \left|\frac{1}{(1+\delta)R_{\text{nom}}} - 1\right|$$

- P_2: **nature**. Picks the value of $\delta \in [-0.1, 0.1]$ to **maximize** the same current error e.

Robust designs lead to non-cooperative zero-sum games
- cooperative solutions make no sense in robust design problems.

Robust Designs: Resistive Circuit

Solution (security): A possible solution is

$$P_1 : R_{\text{nom}} = \frac{100}{99}, \qquad\qquad P_2 : \delta = 0.1$$

which leads to a current error of

$$e(R_{\text{nom}}, \delta) = \left| \frac{1}{(1+\delta)R_{\text{nom}}} - 1 \right| = \left| \frac{99}{110} - 1 \right| = \left| \frac{99 - 110}{110} \right| = 0.1$$

This solution exhibits the following properties

P1.5 Once P_1 picks $R_{\text{nom}} = \frac{100}{99}$, the error e will be maximized for $\delta = 0.1$ and is exactly $e = 0.1$.

P1.6 But, if P_2 picks $\delta = 0.1$, then P_1 can pick

$$\frac{1}{(1+\delta)R_{\text{nom}}} = 1 \Leftrightarrow R_{\text{nom}} = \frac{1}{1+\delta} = \frac{1}{1.1} = \frac{100}{110}$$

and get the **error exactly equal to zero**.

Robust Designs: Resistive Circuit

Conclusion: The solution

$$P_1 : R_{\text{nom}} = \frac{100}{99}, \qquad\qquad P_2 : \delta = 0.1$$

is not a safe choice for P_2

- consequently, this solution is **not** a Nash equilibrium.

However, this solution is safe for P_1

- then, $R_{\text{nom}} = \frac{100}{99}$ is a **security policy** for the designer

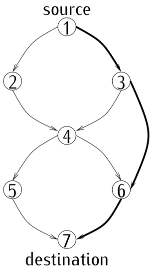
Note: As defined, this game does not have a Nash equilibrium.

It does however have a generalized form of Nash equilibrium that we will encounter in Lecture 4.

Mixed Policies

Mixed Policies: Network Routing

Consider the computer network, and suppose that our **goal** is to **send data packets from source to destination**.



The **3-hop shortest path** from source to destination is highlighted.

Usually, one selects a **path** that **minimizes** the number of hops transversed by the packets. However, this formulation

- does not explore all possible paths, and
- tends to create hot spots.

Mixed Policies: Network Routing

Alternative formulation: consider two players

- P_1 is the **router**: selects the path for the packets
- P_2 is an **attacker**: selects a link to be disabled

The two players make their **decisions independently** and without knowing the choice of the other player.

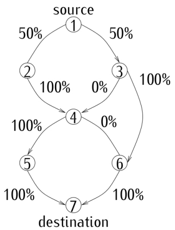
Objective:

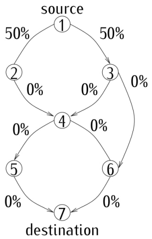
- P_1: wants to maximize the probability that a packet reaches its destination.
- P_2: wants to minimize this probability.

Note: P_2 is **purely fictitious** and its role is to drive P_1 away from routing decisions that would lead to hot spots.

Mixed Policies: Network Routing

Solution: A saddle-point solution for which 50% of the packets will reach their destination.





Stochastic routing policy
Percentages indicate how traffic should be distributed among the outgoing links of a node.

Stochastic attack policy
Percentages indicate the probability by which the attacker will disable that link.

Mixed Policies: Network Routing

This solution exhibits two key properties

P1.7 Once player P_1 picks her policy, P_2's policy is the best response from this players perspective.

P1.8 Once player P_2 picks her policy, P_1s policy is the best response from this players perspective.

These are also **security policies**

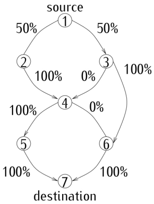
- Each policy guarantees for that player a percentage of packet arrivals **no worse** than 50%.
- No other policies can lead to a guaranteed better percentage of packet arrivals.

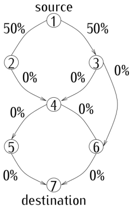
Note: no worse may mean **larger than** or **smaller than**, depending on the player (router or attacker).

Mixed Policies: Network Routing

The solution policies are **mixed policies**

- they call for each player to randomize among several alternatives with **carefully** chosen probabilities.





For this game, there are no Nash equilibrium that do not involve some form of **randomization**.

Notation: policies that do not require randomization are called **pure policies**.

Nash Equilibrium

Nash Equilibrium

(meta) Definition 1.1: Nash Equilibrium

Consider a game with two Players P_1, P_2.

A pair of policies (π_1,π_2) is said to be a **Nash equilibrium** if the following two conditions holds:

- **C1** : If P_1 uses the policy π_1, then there is no admissible policy for P_2 that does strictly better than π_2.
- **C2** : If P_2 uses the policy π_2, then there is no admissible policy for P_1 that does strictly better than π_1.

Attention!

Condition C1 does not require π_2 to be strictly better than all the other policies, just no worse. Similarly for π_1 in C2.

Nash Equilibrium

Definition 1.1 leaves open several issues that can only be resolved in the context of specific games:

1. What exactly is a policy?
2. What is the set of admissible policies against which π_1 and π_2 must be compared?
3. What is meant by a policy **doing strictly better** than another?

The key feature of a **Nash equilibrium** is that it is **stable**

- if P_1 and P_2 start playing at the Nash equilibrium (π_1,π_2), none of the players gain from deviating from these policies.

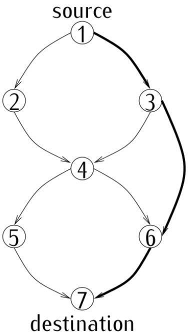
Attention!
The definition of Nash equilibrium does not preclude the existence of **multiple Nash equilibria** for the same game.

Also, there are games for which there are **no Nash equilibria**

Practice Exercise

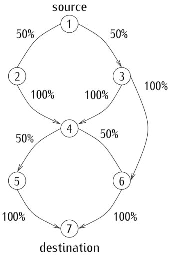
Practice Exercise 1.1.

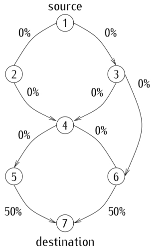
Find **other saddle-point solutions** to the previously introduced network routing game



Solution to Exercise 1.1.

Another saddle-point solution that also satisfies the Nash equilibrium conditions **C1.1-C1.2** in the (meta) Definition 1.1.





Stochastic routing policy
Percentages indicate how traffic should be distributed among the outgoing links of a node.

Stochastic attack policy
Percentages indicate the probability by which the attacker will disable that link.

End of Lecture

01 - Noncooperative Games

Questions?