Game Theory

A Theoretical Framework for Analyzing Strategic Interactions

Team: Yassine Dridi, Ryhem Mekni, Islem Fatma Gamha, Youssef Bougriba

Dauphine Tunis

March 15, 2025

The outline

1. Foundations of Game Theory
2. Typology of Games
3. Game Theory Strategies : Solutions concepts
4. Nash equilibrium
5. The Expansion Towards Artificial Intelligence

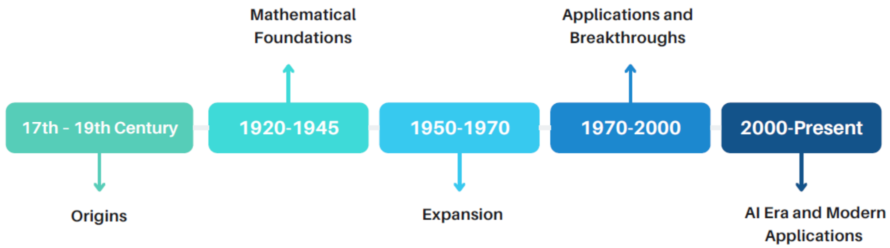
Game Theory

- **What's Game Theory ?**

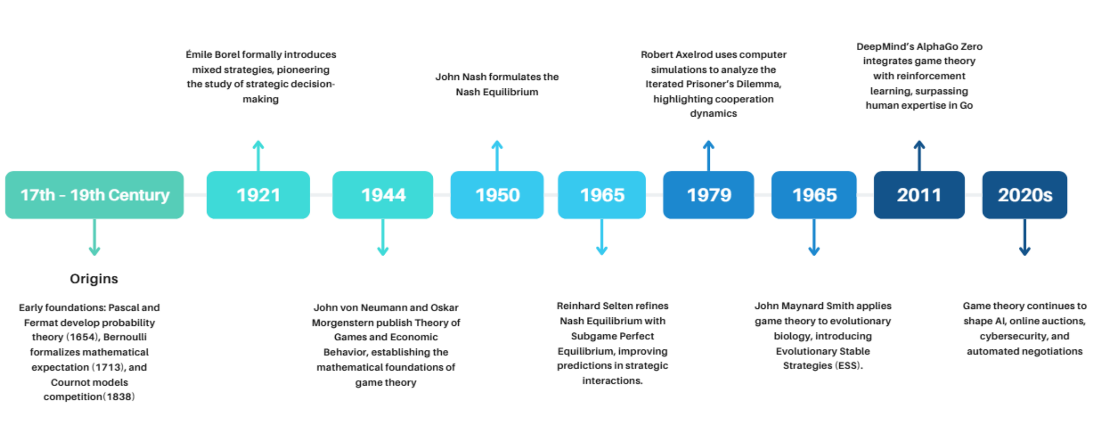
  Game theory is the study of strategies for decision making.

  Formally : As Roger B. Myerson (1991) defines it"the study of mathematical models of conflict and cooperation between intelligent rational decision makers" (Roger B. Myseron, 1991)

History of Game Theory



History of Game Theory: Key Milestones



Foundations of Game Theory

Dauphine Tunis

Game Theory core concepts

We are now ready to formally introduce games and some fundamental concepts,

- Players : Who is interacting?
- Strategies : What are the options of each player? In what order do players act?
- Payoff : How do strategies translate into outcomes? What are players' preferences over possible outcomes?
- Information / beliefs : What do players know/believe about the situation and about one another? What actions do they observe before making decisions?
- Rationality : How do players think?

Game Theory components

GAME THEORY: THE BUILDING BLOCKS OF STRATEGIC DECISION-MAKING

01. Number of players

- 2 players
- N-Players

02. Timing For actions

- Simultaneous actions
- Sequential actions

03. Number of Game stages

- Static Games
- Dynamic Games

04. Behavior of players

- Cooperative Games
- Non-cooperative Games

05. Strategies used

- Pure strategies
- Mixed strategies

Game Theory : Representation of games

A game is just a formal representation of the above information.

This is usually done in one of the following two ways:

1. The extensive-form representation, in which the above information is explicitly described using game trees and information sets.

2. The normal-form (or strategic-form) representation, in which the above information are summarized in a matrix by the use of strategies.

Representation of games

Normal Form

$$G = (N, (S_i)_{i \in N}, (u_i)_{i \in N})$$

- A set of N players, $I \equiv \{1, 2, \ldots, N\}$.
- S_i is the set of strategies/actions available to player i.
- $u_i : S_1 \times \cdots \times S_n \to \mathbb{R}^n$ is the utility function of player i: given the strategies of all players, the function returns the utility of player i.
- An element $s = \langle s_1, \ldots, s_n \rangle$ of $S_1 \times \cdots \times S_n$ is called a **strategy profile**.
- The notation $\langle s'_i, s_{-i} \rangle$ represents the strategy profile that is the same as s except that player i plays strategy s'_i, i.e.,

$$\langle s'_i, s_{-i} \rangle = \langle s_1, \ldots, s_{i-1}, s'_i, s_{i+1}, \ldots, s_n \rangle$$

Normal Form

Game Representation: The following matrix illustrates a normal-form game with two players and their respective payoffs.

	Strategy A (P2)	**Strategy B (P2)**
Strategy X (P1)	(u_1, u_2)	(v_1, v_2)
Strategy Y (P1)	(w_1, w_2)	(z_1, z_2)

Each player i aims to maximize the expected value of u_i, computed based on their own beliefs, making the u_i their utility function.

Representation of Games

Normal Form

- Game is represented by a matrix with every possible combination of actions and the respective payoffs of each strategy.
- Players act simultaneously or without knowing the other player's actions.

	Left (P1)	Right (P1)
Up (P2)	$(4, 3)$	$(-1, -1)$
Down (P2)	$(0, 0)$	$(3, 4)$

Representation of Games

Extensive From

$$G = (N, H, P, A, (u_i)_{i \in N})$$

- N : Set of players, $I = \{1, 2, \ldots, N\}$.
- H is the set of **history**, representing sequences of actions In the game.
- $P : H \to N \cup \{\text{Chance}\}$ is the **player function**, which assigns each decision node to a player or nature (chance).
- $A(h)$ is the **set of available actions** at history h.
- $u_i : Z \to \mathbb{R}$ is the **utility function** for player i, where Z is the set of terminal histories (end nodes).
- **Information Sets:** Indistinguishable decision nodes requiring identical actions.

Representation of games

Extensive Form

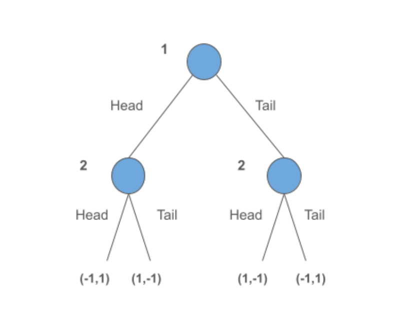
The extensive-form representation provides a detailed account of the game's structure, outlining **who** makes each move, **what players know at the time of their move**, the **available actions** for each player, and the **outcomes** resulting from those actions.

-> This is achieved through a game tree and information sets, alongside essential components like the players involved and the payoffs they receive.

Representation of Games

Extensive Form

- The game is represented by a series of moves on a tree.
- Each node is a choice by a player.
- Lines out of a node represent possible actions.
- Payoffs are specified at the bottom.



Why Information Sets?

Moving from Game Representations to Information Sets

We have seen how games can be represented. However, what happens when players do not have full knowledge of the game's history?

This brings us to the concept of information sets.

Key Question: How do we model situations where players make decisions without knowing all past moves?

What is an Information Set?

Definition: An **information set** is a collection of decision nodes where a player must make a choice but cannot distinguish exactly which node they are at.

Why It Matters:

- Models games with imperfect information.
- Players must make decisions without knowing exactly what their opponent has done.
- Used in many real-world scenarios (e.g., Poker, Auctions).

Complete vs. Perfect Information

Complete Information

- Players know the rules, payoffs, and rationality of others.
- Example: Chess – all pieces and rules are known.

Perfect Information

- There are no simultaneous moves.
- Players see all past actions before making a move.
- Example: Chess and Tic-Tac-Toe.

Example: Information Set in an Extensive Form Game

Scenario:

- Player 1 chooses between A or B.
- Player 2 moves without seeing Player 1's choice.

Key Idea:

- Player 2 is in an **information set** where both choices (A or B) appear identical.
- They must decide between X or Y without knowing what Player 1 chose.

Typology of Games

Dauphine Tunis

Cooperative Games:

Cooperative games, introduced by ***John von Neumann*** and ***Oskar Morgenstern*** in **1944**, are a branch of game theory where players form binding agreements and collaborate to achieve collective payoffs. They focus on coalitions, resource-sharing, and fair reward distribution using concepts like the **Shapley value**, **the core**, and **bargaining solutions.**

John von Neumann

Oskar Morgenstern

Example :

Imagine three companies **(A, B, and C)** want to merge their delivery networks. Each company alone has a weak network, but together they can cover an entire country. **The Shapley value** helps fairly distribute the profits based on each company's contribution.

Example :

Imagine three companies **(A, B, and C)** want to merge their delivery networks. Each company alone has a weak network, but together they can cover an entire country. **The Shapley value** helps fairly distribute the profits based on each company's contribution.

Example :

Imagine three companies **(A, B, and C)** want to merge their delivery networks. Each company alone has a weak network, but together they can cover an entire country. **The Shapley value** helps fairly distribute the profits based on each company's contribution.

→

Example :

Imagine three companies **(A, B, and C)** want to merge their delivery networks. Each company alone has a weak network, but together they can cover an entire country. **The Shapley value** helps fairly distribute the profits based on each company's contribution.

Non Cooperative Games :

Non-cooperative games, pioneered by ***John Nash*** in the **1950s**, focus on strategic decision-making where players act independently, without binding agreements. Unlike cooperative games, these models analyze competition, strategic interactions, and equilibrium concepts.

Example: The Prisoner's Dilemma

Two criminals **(HSSAN and HSSINE)** are arrested and interrogated separately. They each have two options:

- Cooperate (Stay Silent)
- Defect (Betray the Other)

	HSSINE Stays Silent	**HSSINE Betrays**
HSSAN Stays Silent	**(3 years, 3 years)**	(10 years, 0)
HSSAN Betrays	(0, 10 years)	**(5 years, 5 years)**

Cooperative vs. Non-Cooperative Games

Feature	Cooperative Game	Non-Cooperative Game
Strategy Choice	Collective decisions	Individual decisions
Payoff Distribution	Shared among coalition members	Depends on individual actions

Which Game is Better?

Which Game is Better?

If you want stability and shared success,
cooperative games **are preferable.**

Which Game is Better?

If you want stability and shared success,
cooperative games **are preferable.**

If you thrive in competition and strategic maneuvering,
non-cooperative games **are better.**

Zero-Sum Games :

Imagine a world where every gain comes at an equal loss—like a perfectly balanced seesaw.

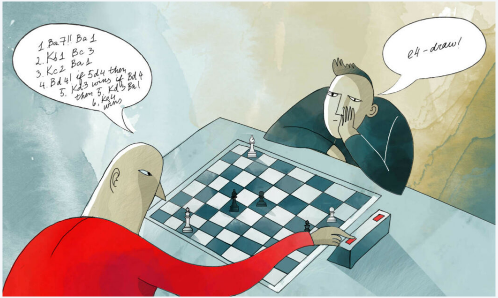
First introduced by ***John von Neumann*** in 1928, zero-sum games describe situations where resources are fixed, and any gain by one participant directly corresponds to a loss by another.

Example: Chess as a Zero-Sum Game

In chess, each player starts with the same number of pieces and aims to checkmate the opponent's king.

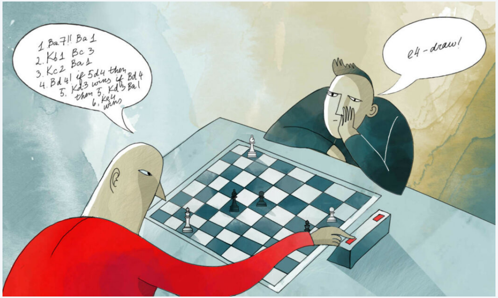
Example: Chess as a Zero-Sum Game

In chess, each player starts with the same number of pieces and aims to checkmate the opponent's king.



Example: Chess as a Zero-Sum Game

In chess, each player starts with the same number of pieces and aims to checkmate the opponent's king.



Mathematical Representation:

- **Payoff function:** Let $u_1(s)$ and $u_2(s)$ be the utility functions of each player for a given strategy s.
- **Zero-Sum Condition:**
$$u_1(s) + u_2(s) = 0, \quad \forall s \in S$$

  where S is the space of all possible game states.

Minimax Theorem in Chess

Optimal Play Strategy: Each player aims to minimize their worst possible loss.

Minimax Theorem in Chess

Optimal Play Strategy: Each player aims to minimize their worst possible loss.

- **Minimax Strategy:**

$$\min_{\sigma_1} \max_{\sigma_2} u_1(\sigma_1, \sigma_2) = \max_{\sigma_2} \min_{\sigma_1} u_2(\sigma_1, \sigma_2)$$

where σ_1, σ_2 are mixed strategies.

Minimax Theorem in Chess

Optimal Play Strategy: Each player aims to minimize their worst possible loss.

- **Minimax Strategy:**

$$\min_{\sigma_1} \max_{\sigma_2} u_1(\sigma_1, \sigma_2) = \max_{\sigma_2} \min_{\sigma_1} u_2(\sigma_1, \sigma_2)$$

  where σ_1, σ_2 are mixed strategies.

- **Implication:**
  - White plays to minimize Black's best possible response.
  - Black plays to maximize their worst-case gain.

Non Zero-Sum Games

-> Not all games are battles : some are delicate negotiations, shifting alliances, and shared victories.

In **Non-Zero-Sum Games**, one player's gain doesn't mean another's loss, and outcomes can be mutually beneficial or harmful, reshaping how we think about strategy.

Example: Subsidies on Essential Products in Tunisia

- In Tunisia, the government provides subsidies on essential products like bread and sugar to support families with limited income.
- While this policy ensures food security and access to basic products, it has long-term health consequences.
- **Choices for the State:**
  - Maintain subsidies
  - Remove subsidies
- **Choices for Society:**
  - Continue consuming subsidized products
  - Change consumption habits

PAYOFF MATRIX OF THE SUBSIDY DECISION

State's Decision	Society Continues Consumption	Society Changes Habits
State Maintains Subsidies	**Short-term benefit**: Immediate affordability, but long-term health problems increase healthcare costs (High cost for State, health burden for society)	**Short-term loss**: Reduced healthcare costs, but immediate negative reaction from consumers (State benefits from fewer health issues, Society sacrifices some short-term comfort)
State Removes Subsidies	**Immediate benefit**: Reduced healthcare burden, but some social unrest and economic inequality (Cost to State, short-term loss for Society)	**Long-term benefit**: Potential health benefits, reduced healthcare burden, but economic difficulties in the short term (Benefit for State, long-term gain for Society)

Zero-Sum Games vs. Non-Zero-Sum Games

Aspect	Zero-Sum Games	Non-Zero-Sum Games
Players' Goals	Maximize their own payoff at the expense of the other	Maximize mutual benefit, cooperation is often possible
Payoff Structure	The total payoff to all players sums to zero	The total payoff can be positive or negative
Cooperation	Cooperation is usually not possible, as it results in one player's loss	Cooperation can lead to better outcomes for both players

Which One is Better?

Which One is Better?

- **If the goal is collaboration or mutual benefit**, Non-Zero-Sum Games are the better choice, leading to optimal outcomes for all parties involved.

Which One is Better?

- **If the goal is collaboration or mutual benefit**, Non-Zero-Sum Games are the better choice, leading to optimal outcomes for all parties involved.

- **If the scenario is purely competitive**, where only one party can win, Zero-Sum Games are preferable, as they focus on competition without any opportunity for mutual gain.

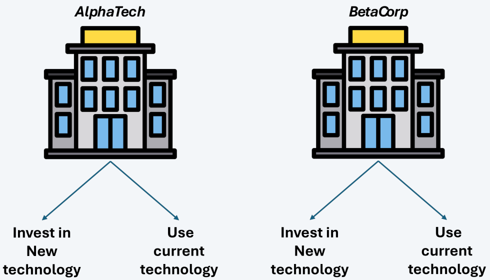
Static Games:

Static games in game theory refer to situations where:

- Players make their decisions simultaneously or without knowing the actions of others in advance.
- There is no sequence of actions or repeated interactions over time.
- They are called ”static” because they do not evolve dynamically.
- Often modeled using payoff matrices or payoff functions.

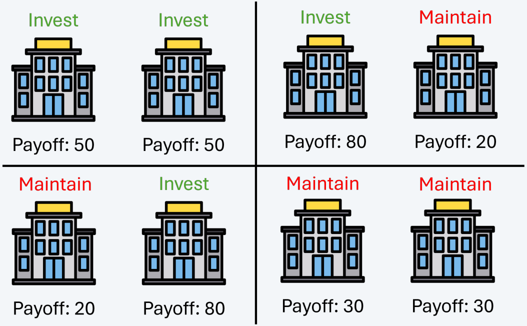
Example :

Imagine two tech companies, AlphaTech and BetaCorp, must simultaneously decide whether to invest in a costly innovation or stick to their current strategy.



Example :

Explanation of Payoffs for each possibility:



Dynamic Games:

Dynamic games in game theory involve:

- Multiple stages or periods where players make decisions over time.
- Past actions influencing future decisions, adding a strategic layer.
- A temporal and strategic dimension, making them more complex than static games.
- Unlike static games, players do not act simultaneously but rather in a structured sequence.

Quiz: Identify the Game Type

Match the following scenarios to the correct type of game (Static Simultaneous, Static Sequential, Dynamic Simultaneous, or Dynamic Sequential).

1. Two firms fixing their pricing simultaneously without knowing the other’s choice.
2. A chess game where each player moves one after the other.
3. Two players negotiating a contract over multiple rounds.
4. A penalty kick in soccer.
5. Repeated Prisoner’s Dilemma

Example of applications of Dynamic Games:

Dynamic games are used to model real-world situations where decisions are made over time and past actions influence future outcomes. Examples include:


Game Theory Strategies : Solutions concepts

Dauphine Tunis

Pure Strategy

Definition: A strategy with a determinated choice.
Mathematical Representation:

$$\sigma_i(s_i) = 1 \quad \text{for some } s_i \in S_i, \quad \sigma_i(s) = 0 \text{ for all } s \neq s_i \tag{1}$$

Example: Battle of the Sexes

- Two players: Husband and Wife
- Choices: Ballet or Football
- Both prefer going together over going alone

	Football	**Ballet**
Football	(2,1)	(0,0)
Ballet	(0,0)	(1,2)

Mixed Strategy

Definition: A strategy where a player randomizes over different probabilistic actions.
Mathematical Representation:

$$\sigma_i = (p_1, p_2, ..., p_n), \quad \sum_j p_j = 1, \quad 0 \leq p_j \leq 1 \tag{2}$$

Example: Penalty Kick in Soccer

- The kicker chooses to aim left or right.
- The goalkeeper chooses to dive left or right.

	Dive Left	Dive Right
Kick Left	(0,1)	(1,0)
Kick Right	(1,0)	(0,1)

Nash equilibrium

Dauphine Tunis

Nash equilibrium

- **What Is Nash Equilibrium ?**

  The Nash equilibrium, a fundamental concept in game theory, refers to a state where players, even knowing their opponents' strategies, have no incentive to change their own strategies because doing so would not improve their outcomes.

  It represents an optimal outcome where each player's strategy is the best response to the others, ensuring no participant benefits from unilateral deviation.

Nash Equilibrium

Definition: A Nash equilibrium is a situation where no player can improve their payoff by unilaterally changing their strategy.

Mathematical Formulation: A strategy profile $(s_1^*, s_2^*, \ldots, s_n^*)$ is a Nash equilibrium if, for every player i:

$$u_i(s_1^*, \ldots, s_i^*, \ldots, s_n^*) \geq u_i(s_1^*, \ldots, s_i, \ldots, s_n^*) \quad \forall s_i \in S_i$$

Key Properties:

- Each player is playing their **best response** to others' strategies.
- Nash equilibria are stable and self-reinforcing.
- They may be unique, multiple, or non-existent depending on the game.

Nash Equilibrium vs. Dominant Strategy

Nash equilibrium is often compared to dominant strategy, both being strategies of game theory unless that both terms are similar but slightly different :

The Nash equilibrium:

states that nothing is gained if any of the players change their strategy while all of the other players maintain their strategy.

Dominant strategy:

asserts that a player will choose a strategy that will lead to the best outcome regardless of the strategies that the other players have chosen.

Nash equilibrium

- **How Do You Find Nash Equilibrium?**

  To find the Nash equilibrium in a game, one would have to model out each of the possible scenarios to determine the results and then choose what the optimal strategy would be.

Methods to Compute Nash Equilibria

(1) Best Response Analysis

- Identify the best response for each player.
- Solve for the point where best responses intersect.

(2) Iterated Elimination of Strictly Dominated Strategies

- Remove strategies that are never optimal.
- If a unique strategy remains, it is the Nash equilibrium.

(3) Optimization Approach (Continuous Games):

- Formulate utility functions $u_i(s_i, s_{-i})$.
- Compute derivatives to find first-order conditions (FOCs).
- Solve the system of equations.

Iterated Elimination: Prisoner's Dilemma

Scenario: Two suspects, **(HSSAN and HSSINE)**, must decide whether to Defect (Betray the Other) or cooperate (Stay Silent).

Payoff Matrix:

	HSSINE Stays Silent	HSSINE Betrays
HSSAN Stays Silent	**(3 years, 3 years)**	(10 years, 0)
HSSAN Betrays	(0, 10 years)	**(5 years, 5 years)**

Iterated Elimination: Prisoner's Dilemma

- HSSAN's Best Response:
  - If HSSINE chooses to stay silent, HSSAN compares 3 years (Silent) vs. 0 years (Betray). $\Rightarrow$ Betray is better.
  - If HSSINE chooses to betray, HSSAN compares 10 years (Silent) vs. 5 years (Betray). $\Rightarrow$ Betray is better.
- HSSINE's Best Response:
  - If HSSAN chooses to stay silent, HSSINE compares 3 years (Silent) vs. 10 years (Betray). $\Rightarrow$ Betray is better.
  - If HSSAN chooses to betray, HSSINE compares 0 years (Silent) vs. 5 years (Betray). $\Rightarrow$ Betray is better.

Equilibrium: After eliminating strictly dominated strategies, the Nash equilibrium is:

$$(\text{Betray, Betray}) : \text{Both defect, resulting in } (5 \text{ years}, 5 \text{ years}).$$

Iterated Elimination: Weighted Rock-Paper-Scissors

Scenario: Players A and B play a weighted version of Rock-Paper-Scissors:

- Rock wins against Scissors but loses to Paper.
- Paper wins against Rock but loses to Scissors.
- Scissors win against Paper but lose to Rock.

Weighted Payoff Matrix:

	Player B: Rock	**Player B: Paper**	**Player B: Scissors**
Player A: Rock	$(0, 0)$	$(-1, 2)$	$(2, -1)$
Player A: Paper	$(2, -1)$	$(0, 0)$	$(-1, 2)$
Player A: Scissors	$(-1, 2)$	$(2, -1)$	$(0, 0)$

Iterated Elimination: Weighted Rock-Paper-Scissors

Iterative Elimination:

- No pure strategy strictly dominates another.
- Mixed Strategies: Let Player A mix with probabilities (p_R, p_P, p_S), and Player B mix with (q_R, q_P, q_S).

Expected Payoffs: Player A's expected payoff is:

$$U_A = p_R(0q_R + (-1)q_P + 2q_S) + p_P(2q_R + 0q_P + (-1)q_S) + p_S((-1)q_R + 2q_P + 0q_S)$$

Equilibrium:

- Both players mix with equal probabilities: $(1/3, 1/3, 1/3)$.
- Payoff: $(0, 0)$.

Key Lesson: Even in competitive games, balance emerges as players seek to neutralize one another's strategies.

Best Response Analysis: Battle of the Sexes

Scenario: A and B must decide where to go: a Bach concert or a Stravinsky concert. A prefers Bach, while B prefers Stravinsky. Both value being together over going alone.

Payoff Matrix:

	B: Bach	**B**
A: Bach	$(2, 1)$	$(0, 0)$
A: Stravinsky	$(0, 0)$	$(1, 2)$

Best Response Analysis: Battle of the Sexes

Best Responses:

- If B chooses Bach, A's best response is Bach ($2 > 0$).
- If B chooses Stravinsky, A's best response is Stravinsky ($1 > 0$).
- Similarly, B's best responses are aligned with A's choices.

Equilibria:

- $(\text{Bach}, \text{Bach})$: A and B go to Bach.
- $(\text{Stravinsky}, \text{Stravinsky})$: A and B go to Stravinsky.

Optimization Approach: Public Goods Game

Scenario: A group of n individuals decides how much to contribute to a public good. Each individual i can contribute c_i to the public pool.

The public good provides total benefit $G = \sum c_i$, shared equally. Contributing has a personal cost.

Payoff Function: The utility of Player i:

$$u_i(c_1, ..., c_n) = \frac{G}{n} - c_i = \frac{\sum_{j=1}^{n} c_j}{n} - c_i$$

Optimization Approach: Public Goods Game

Optimization:

- Player i maximizes $u_i(c_1, ..., c_n)$:

$$\max_{c_i} u_i(c_1, ..., c_n)$$

- Take the derivative:

$$\frac{\partial u_i}{\partial c_i} = \frac{1}{n} - 1$$

- Set to 0 to find the best response:

$$\frac{1}{n} - 1 = 0 \quad \Rightarrow \quad c_i^* = 0$$

Optimization Approach: Public Goods Game

Nash Equilibrium: In equilibrium, all players free-ride and contribute nothing:

$$c_1 = c_2 = \ldots = c_n = 0$$

Key Lesson: The Nash equilibrium illustrates ***the free-rider problem***: when benefits are shared but costs are individual, no one contributes, and the public good fails.

The perfect Nash equilibrium:

- **Explanation**

Dynamic games are used to model real-world situations where decisions are made over time and past actions influence future outcomes.

- **Why is it necessary?**

In some dynamic games, a Nash equilibrium can rely on non-credible threats or irrational behaviors in situations that may not necessarily occur. The perfect Nash equilibrium eliminates these equilibria by requiring that the strategies be rational at every stage of the game, even in subgames that are not reached.

Example :

- **The Ultimatum Game:**

1- Player 1 proposes a split of a sum of money
(e.g., 80% for themselves, 20% for Player 2).

2- Player 2 accepts or rejects the proposal.

2-a If Player 2 accepts, the money is split as proposed.
2-b If Player 2 rejects, both players receive 0.

How to find perfect Nash equilibria while ensuring that strategies are optimal at each stage of the game?

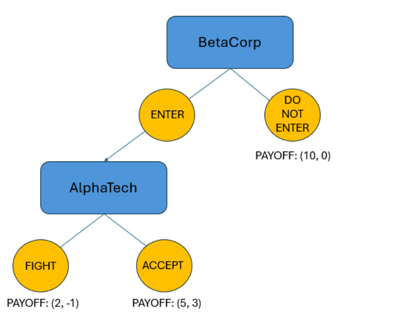
- **Backward Induction Method**

To solve a sequential game using backward induction, follow these steps:

1. Start from the last decision nodes of the game.
2. Determine the best action for the player at each node.
3. Move backward to the beginning of the game, using the decisions already made.

Example: The game of entering a market

A monopoly (AlphaTech) is already present in a market. A new company (BetaCorp) is deciding whether or not to enter the market.If Betacorp decides to enter, AlphaTech must choose between competing or accepting.



The Expansion Towards Artificial Intelligence

Dauphine Tunis

How can game theory be applied to solve real-world problems today?

Game Theory & AI

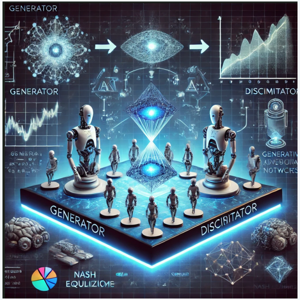
- ***Game Theory to AI***: Offers structured models for competitive environments, equilibrium analysis, and predictions, helping AI agents understand and plan for strategic interactions.

- ***AI to Game Theory***: Enhances game theory by providing adaptability, learning capabilities, and real-time decision-making, addressing imperfections in information and large strategy spaces where classical game theory is limited.

Mutual Contributions of Game Theory and AI

ASPECT	GAME THEORY -> AI	AI -> GAME THEORY
STRATEGIC DECISION-MAKING	PROVIDES FRAMEWORKS FOR MODELING INTERACTIONS AMONG RATIONAL AGENTS, ENHANCING AI'S ABILITY TO PREDICT AND STRATEGIZE IN MULTI-AGENT ENVIRONMENTS.	OFFERS COMPUTATIONAL TOOLS TO SOLVE COMPLEX GAME-THEORETIC MODELS THAT ARE ANALYTICALLY INTRACTABLE.
LEARNING MECHANISMS	INTRODUCES CONCEPTS LIKE NASH EQUILIBRIUM TO GUIDE THE DEVELOPMENT OF LEARNING ALGORITHMS IN AI.	IMPLEMENTS REINFORCEMENT LEARNING TO SIMULATE AND ANALYZE STRATEGIC INTERACTIONS, ENRICHING GAME THEORY WITH EMPIRICAL DATA.
SCALABILITY	OFFERS THEORETICAL MODELS FOR STRATEGIC INTERACTIONS BUT MAY STRUGGLE WITH SCALABILITY IN COMPLEX SCENARIOS.	UTILIZES COMPUTATIONAL POWER TO HANDLE LARGE-SCALE PROBLEMS, EXTENDING THE APPLICABILITY OF GAME-THEORETIC CONCEPTS.

Applications of Game Theory in AI



Generative Adversarial Networks (GANs)

Why AI Adopts Game Theory?

- **Generative Adversarial Networks (GANs)**:

What Are GANs?

- GANs use two neural networks–the **Generator** and the **Discriminator**–to create realistic synthetic data.
- These two networks engage in a **zero-sum game**, where:
  - The **Generator** tries to produce fake data that looks real.
  - The **Discriminator** tries to distinguish between real and fake data.
  - Over time, the competition improves the generated data.

Example: Generating Handwritten Digits

```python
import numpy as np
import tensorflow as tf  # For neural network operations

# Define the Generator network
def build_generator():
    model = tf.keras.Sequential()
    model.add(tf.keras.layers.Dense(128, input_dim=100))  # Input: Random noise vector (100 dimensions)
    model.add(tf.keras.layers.LeakyReLU(alpha=0.2))
    model.add(tf.keras.layers.Dense(784, activation='tanh'))  # Output: Fake image (28x28 pixels flattened)
    return model

# Define the Discriminator network
def build_discriminator():
    model = tf.keras.Sequential()
    model.add(tf.keras.layers.Dense(128, input_dim=784))  # Input: Image (28x28 pixels flattened)
    model.add(tf.keras.layers.LeakyReLU(alpha=0.2))
    model.add(tf.keras.layers.Dense(1, activation='sigmoid'))  # Output: Probability (real or fake)
    return model

# Build the networks
generator = build_generator()
discriminator = build_discriminator()

# Loss functions
cross_entropy = tf.keras.losses.BinaryCrossentropy()

# Discriminator loss
def discriminator_loss(real_output, fake_output):
    real_loss = cross_entropy(tf.ones_like(real_output), real_output)  # Real data should be labeled as 1
    fake_loss = cross_entropy(tf.zeros_like(fake_output), fake_output)  # Fake data should be labeled as 0
    return real_loss + fake_loss

# Generator loss
def generator_loss(fake_output):
    return cross_entropy(tf.ones_like(fake_output), fake_output)  # Generator wants Discriminator to label fake data as 1
```

```python
for epoch in range(epochs):
    for batch in range(len(real_data) // batch_size):
        # Sample real data
        real_images = get_real_data_batch(batch_size)  # Real images from the dataset

        # Generate fake data
        noise = np.random.normal(0, 1, (batch_size, 100))  # Random noise vector
        fake_images = generator(noise)

        # Train the Discriminator
        with tf.GradientTape() as disc_tape:
            real_output = discriminator(real_images)
            fake_output = discriminator(fake_images)
            disc_loss = discriminator_loss(real_output, fake_output)

        gradients_of_discriminator = disc_tape.gradient(disc_loss, discriminator.trainable_variables)
        discriminator_optimizer.apply_gradients(zip(gradients_of_discriminator, discriminator.trainable_variables))

        # Train the Generator
        with tf.GradientTape() as gen_tape:
            fake_images = generator(noise)
            fake_output = discriminator(fake_images)
            gen_loss = generator_loss(fake_output)

        gradients_of_generator = gen_tape.gradient(gen_loss, generator.trainable_variables)
        generator_optimizer.apply_gradients(zip(gradients_of_generator, generator.trainable_variables))

    print(f"Epoch {epoch+1}, Discriminator Loss: {disc_loss}, Generator Loss: {gen_loss}")

# After training, the Generator can produce realistic images
```

Key Takeaways

Zero-Sum Game

- The Generator and Discriminator are engaged in a **zero-sum game**.
- **Dynamic:**
  - If the Generator improves, the Discriminator suffers (and vice versa).
  - Their competition drives both networks to improve iteratively.

Strategic Interaction

- Each player adapts its strategy based on the other's actions:
  - The Generator learns to exploit weaknesses in the Discriminator.
  - The Discriminator learns to detect flaws in the Generator's output.

Payoff Functions

- The loss functions act as **payoff functions**, guiding the behavior of both players:
  - The Generator minimizes its loss by maximizing the Discriminator's error.
  - The Discriminator minimizes its loss by correctly classifying real and fake data.

Key Takeaways

Equilibrium Outcome

- At convergence, the system reaches a **Nash equilibrium**:
  - The Generator produces indistinguishable fake data.
  - The Discriminator outputs a 50% confidence score for both real and fake data.

Conclusion

Game theory offers a powerful lens to enhance AI coherence, turning inconsistency into a structured optimization challenge. By leveraging strategic interactions, models can refine responses dynamically, boosting reliability without retraining.

Looking ahead, deeper integration of game-theoretic mechanisms could lead to self-correcting AI-systems that debate, validate, and refine their own outputs in real time. This shift could bring us closer to truly autonomous, trustworthy, and context-aware generative models.

THANK YOU !

Team: Yassine Dridi, Ryhem Mekni, Islem Fatma Gamha, Youssef Bougriba

Dauphine Tunis

March 15, 2025