Innovative Mechanism Design for Strategic Decision-Making in AI Environments ⋆

Polina Konovalova[1]

Duke Kunshan University, Kunshan, Jiangsu 215316, China
polina.konovalova@dukekunshan.edu.cn
LinkedIn

Abstract. In this paper, we propose a novel mechanism design approach for government project allocation using game theory principles. We address the challenges posed by integrating artificial intelligence (AI) into strategic decision-making processes, emphasizing dynamic environments with AI agents' adaptive behavior. Our method employs a cake-cutting game framework, where party representatives allocate preferences to projects across multiple fields. Through a Bayesian algorithm, we determine the most preferred project for government funding. By leveraging reinforcement learning, hierarchical learning architectures, and Bayesian game theory, we enhance adaptability and efficacy in AI development. Our approach improves fairness, efficiency, and overall welfare, demonstrated through formal analysis and computational simulations.
Notes: In submission to Final Project for COMPSCI/ECON 206 Computational Microeconomics, 2024 Spring Term (Seven Week - Fourth) instructed by Prof. Luyao Zhang at Duke Kunshan University.

Keywords: computational economics · game theory · innovative education · cake-cutting · artificial intelligence · Bayesian algorithm · political game theory

1 Introduction

In the realm of AI development, navigating the complexities of strategic interactions presents formidable challenges. Traditional game theory, foundational in understanding strategic decision-making, encounters limitations when applied

⋆ **Acknowledgments**: I extend my sincere appreciation to Professor Luyao Zhang for their invaluable guidance and unwavering support throughout this research endeavor, and their mentorship and insightful feedback, which have greatly enhanced the quality and depth of this paper. Their expertise and encouragement have been pivotal in shaping the trajectory of this study. I am equally thankful to my classmates for their engaging dialogues and constructive feedback, which have broadened my perspective and fostered intellectual growth. Lastly, I am deeply grateful to my friends and family for their unwavering encouragement and understanding throughout this academic journey. Their support has been a constant source of motivation and inspiration.

to dynamic environments where learning and adaptation are paramount. As AI agents evolve and adapt their strategies over time through techniques such as reinforcement learning, the static nature of traditional game-theoretic models becomes increasingly inadequate. Moreover, the inherent exploration-exploitation trade-off faced by AI systems further complicates matters, as they must balance between discovering new strategies and exploiting known ones to maximize short-term gains. To address these challenges, innovative solutions are imperative. Integrating reinforcement learning techniques into game-theoretic models enables agents to adapt dynamically, while hierarchical learning architectures offer a framework for balancing exploration and exploitation effectively. Additionally, leveraging Bayesian game theory allows for modeling uncertainty and incorporating learning over time, providing a more nuanced understanding of strategic decision-making in dynamic environments. Through these advancements, a new frontier emerges, promising a more adaptive and effective approach to strategic interactions in AI development.

In this paper, we propose a novel mechanism design approach for government project allocation using game theory principles. Our approach employs a cake-cutting game framework where party representatives, acting as players, allocate preferences to projects across multiple fields. These preferences are then aggregated and processed using a Bayesian algorithm to determine the most preferred project for government funding. We demonstrate the efficacy of our approach through formal analysis and computational simulations, showing how it improves upon existing methods in terms of fairness, efficiency, and overall welfare.

Allocating government funds for various projects is a complex and crucial task that often involves conflicting interests and preferences among different stakeholders. Traditional decision-making processes may suffer from biases, inefficiencies, and lack of transparency. Game theory offers a promising framework for designing mechanisms that can address these challenges by incentivizing strategic behavior and ensuring desirable outcomes. In this paper, we propose a game-theoretic approach to government project allocation, leveraging the principles of cake-cutting games and Bayesian inference.

We model the government project allocation problem as a cake-cutting game, where party representatives act as players with preferences over projects in different fields (economy, social policy, environment, culture). Each player ranks the projects in each field according to their importance, with higher-ranked projects receiving greater weight. These preferences are then aggregated across all players and processed using a Bayesian algorithm to compute the posterior distribution over project allocations.

2 Background

Re-imagining game theory in the age of AI and complex systems demands a holistic approach that transcends traditional boundaries. No longer confined to human players, this new landscape encompasses a diverse array of agents—from autonomous algorithms to sentient machines — each with their unique rationality and adaptability. To navigate this dynamic arena, game theory must evolve, integrating interdisciplinary insights to probe the complexities of human-AI interaction. It's not just about strategic decision-making; it's a reflection of the intricate dance between human cognition and artificial intelligence. This evolution is crucial as AI becomes increasingly intertwined with societal structures, raising profound questions of agency, responsibility, and fairness. In essence, reimagining game theory in this context is a call to arms — a call to chart a path towards a future where human and artificial intelligence can coexist harmoniously.

3 An Illustration Example

Consider a scenario where three political parties, representing different ideological viewpoints, must decide on the allocation of government funds for four proposed projects: a new economic stimulus package (E), a social welfare program (S), an environmental conservation initiative (EN), and a cultural heritage preservation project (C). Each party has its own preferences over these projects, reflecting its priorities and policy objectives.

Party A, representing a progressive platform, prioritizes social welfare (S) as its top concern, followed by environmental conservation (EN), economic stimulus (E), and cultural heritage preservation (C).

Party B, representing a centrist position, values economic stimulus (E) and environmental conservation (EN) equally, followed by social welfare (S) and cultural heritage preservation (C).

Party C, representing a conservative stance, prioritizes economic stimulus (E) as its main objective, followed by cultural heritage preservation (C), social welfare (S), and environmental conservation (EN).

Using traditional decision-making methods, such as simple voting or weighted averages, it may be challenging to reach a consensus that satisfies all parties and reflects the diverse preferences of stakeholders. However, by employing our Bayesian cake-cutting game framework, we can compute the optimal allocation of funds that maximizes overall welfare while taking into account the preferences of each party.

In this example, our Bayesian algorithm may determine that the social welfare program (S) receives the highest overall score, indicating that it is the most preferred project across all parties. As a result, the government allocates the majority of funds to the social welfare program, reflecting the collective priorities of the electorate.

We implement the Bayesian cake-cutting game framework in Google Colab using Python. This code simulates a negotiation process among three parties to determine the allocation of funds for one of several randomly generated projects across different spheres (economy, social policy, environment, culture) (See 7. Appendix for the code provided in the Google Colab).

3.1 Function Definitions

Definition 1 (reach consensus(preferences)). *This function determines the preference with the highest number of votes (consensus) among the parties. It counts the number of votes for each preference and returns the preferences with the maximum number of votes.*

Definition 2 (update preferences beliefs(preferences, resources)). *This function updates the Bayesian probabilities based on the preferences of the parties. It calculates the posterior beliefs for each preference by counting the number of occurrences and normalizing the counts.*

Definition 3 (negotiate allocation(resources, num parties)). *This function facilitates the negotiation process among the parties to allocate funds for the projects. It prompts each party to input their preferences for each project and determines the final allocation based on the project with the highest preference rank.*

3.2 Generating Random Projects

Random projects are generated from predefined lists for each sphere (economy, social policy, environment, culture). Each project is assigned a random funding amount.

3.3 Outputting Randomly Generated Projects

The code outputs the randomly generated projects along with their respective funding amounts.

3.4 Negotiation Process

Party representatives input their preferences for each project, ranking them from 1 (highest) to 10 (lowest). The negotiation process uses the Bayesian algorithm to update the probabilities based on the preferences of the parties. The final allocation is determined based on the project with the highest probability.

3.5 Bayesian Probabilities Before Final Choice

The code outputs the Bayesian probabilities for each project before the final choice is made. These probabilities are updated based on the preferences of the parties.

3.6 Final Allocation

The code outputs the final allocation decision, i.e., the project with the highest probability.

4 The Pioneers in the History of Game Theory

In the evolution of game theory, prominent theorists have continuously built upon the foundational work of their predecessors, paving the way for groundbreaking advancements and opening new avenues for exploration. From John von Neumann and Oskar Morgenstern's seminal "Theory of Games and Economic Behavior" to John Nash's formulation of the Nash Equilibrium, each contribution has expanded our understanding of strategic interactions. These successive developments not only deepen our comprehension of strategic decision-making but also inspire future scholars to explore novel territories within the discipline, perpetuating its evolution.

Traditional decision theory, the predecessor of game theory, delves into individual decision-making under uncertainty, prioritizing optimization of expected utility or minimization of loss. Game theory, as outlined in the seminal work "Theory of Games and Economic Behavior" by John von Neumann and Oskar Morgenstern in 1944, signifies a significant departure from traditional decision theory towards understanding strategic interactions [1]. Their foundational insights set the stage for subsequent advancements in the field. In the 1950s, John Nash's contributions, particularly his formulation of the Nash Equilibrium, established game theory as a separate discipline, linking individual decision-making with strategic interactions [2]. Initially focusing on pure strategies, Nash's equilibrium concept was later expanded in the 1960s by John Harsanyi and Reinhard Selten to incorporate mixed strategies, where players incorporate randomness into their actions [3]. This progression from pure to mixed strategies significantly deepened our understanding of strategic interactions.

While Nash primarily explored non-cooperative games, Lloyd S. Shapley introduced cooperative game theory, allowing players to form coalitions and make binding agreements, thereby expanding the scope of strategic interactions [4]. The transition from static to dynamic games was further advanced by von Neumann's minimax theorem and Nash's extension of static games into dynamic ones, introducing concepts such as subgame perfection [1] [2]. Finally, the shift from games with perfect information, emphasized by von Neumann and Morgenstern, to those with imperfect information, championed by John Harsanyi, enhanced the practical relevance of game theory in real-world situations [5]. These significant milestones in the evolution of game theory not only build upon the groundwork laid by earlier scholars but also open up new avenues for exploration, providing opportunities for future researchers in the field.

Imagining myself as one of the "founding fathers" of game theory, Lloyd Shapley [4], I embarked on a transformative journey aimed at addressing the limitations of non-cooperative game theory, particularly evident in the classic example of the "Prisoner's Dilemma" [6]. The specific challenge I aimed to overcome was the inability of non-cooperative frameworks to capture cooperative interactions and the potential benefits of forming alliances among players.

In the "Prisoner's Dilemma," two individuals are faced with the choice of cooperating or betraying one another to the authorities [6]. In the traditional non-cooperative approach, each player acts independently, often leading to a suboptimal outcome where both players betray each other to minimize personal risk. However, in reality, the players may communicate and agree to cooperate, leading to a mutually beneficial outcome where both refrain from betrayal.

Introducing cooperative game theory, my [Lloyd Shapley's] innovative framework enables players to form coalitions and negotiate binding agreements, thus fostering cooperation and leading to mutually beneficial outcomes. By allowing for the formation of coalitions and equitable distribution of gains among players, cooperative game theory outperforms previous non-cooperative approaches in achieving desired outcomes, fostering better coordination and collaboration among players, as illustrated in the context of the "Prisoner's Dilemma" game.

5 Review Classic Games, Nash Equilibrium and the Analytical Tools

5.1 Exploring Inspirational Games in Strategic or Normal Form

Definition 4 (Strategic or Normal Form). *a representation in game theory that outlines the strategies and associated payoffs for each player in a game, enabling analysis of strategic interactions independently of move sequences. A (finite, n-person) normal-form game is a tuple* (N, A, u)*, where:*

- N *is a finite set of* n *players, indexed by* i*;*
- $A = A_1 \times \ldots \times A_n$*, where* A_i *denotes a finite set of actions available to player* i*. Each vector* $a = (a_1, \ldots, a_n) \in A$ *is called an action profile;*
- $u = (u_1, \ldots, u_n)$*, where* $u_i : A \to \mathbb{R}$ *is a real-valued utility (or payoff) function for player* i *[7].*

The strategic (normal) form stands as the most prevalent depiction of strategic interactions within game theory. In strategic game theory, players make decisions with the understanding that the outcome depends not only on their own actions but also on the actions of others. The Prisoner's Dilemma is one of the central examples of a strategic game.

The Prisoner's Dilemma is a game that has a significant impact on game theory. It is known for its simplicity and profound implications [6]. In this classic scenario, two people are arrested and interrogated separately for a crime that they supposedly committed together. Each prisoner faces a dilemma of whether to cooperate by remaining silent or to defect by betraying their partner and confessing. The outcome of the game depends on the decisions made by both prisoners (see Table 1).

- If both prisoners cooperate (remain silent), they each receive a moderate sentence, representing the optimal collective outcome.
- If both prisoners defect (confess), they both receive harsher sentences, although it's individually preferable to cooperating.
- If one prisoner defects while the other cooperates, the defector receives the most favorable outcome (freedom), while the cooperator faces the harshest punishment (longest sentence).

		Player B
	Cooperate	**Defect**
Player A	(3, 3)	(0, 5)
	(5, 0)	(1, 1)

Table 1. Prisoner's Dilemma Payoff Matrix

Nash Equilibrium

Definition 5 (Nash Equilibrium). *a stable equilibrium in the dynamics of a strategic game, wherein each player possesses accurate expectations regarding the actions of others and makes rational decisions accordingly. [2] [8] In a strategic game, a strategy profile $s = (s_1, \ldots, s_n)$ constitutes a Nash equilibrium if, for every player i, the strategy s_i represents the optimal response to the strategies of all other players s_{-i}. [7]*

Nash Equilibrium occurs when every player's strategy remains optimal considering the choices of others, leading to a stable outcome. In the Prisoner's Dilemma, despite mutual cooperation offering a superior outcome, each player's dominant strategy is defection, making reaching equilibrium challenging. Computational determination of Nash equilibrium in strategic games is facilitated through Python-based tools like Nashpy [9], enabling the definition and resolution of equilibria. Nashpy is a Python library tailored for game theory, specifically aimed at discovering Nash equilibria in two-player strategic games.

The following code (see Table 2) allows us to see that the Nash Equilibrium is achieved when both of the players either cooperate or defect:

The implications of the Prisoner's Dilemma extend beyond theoretical realms, permeating various disciplines including economics, politics, and psychology [8]. Serving as a powerful metaphor for real-world dynamics, this game illuminates the intricate interplay between individual incentives and collective outcomes.

Engaging with the Prisoner's Dilemma has enriched my comprehension of strategic interactions within complex systems. It underscores the pivotal roles of trust, cooperation, and communication in navigating scenarios where individual rationality intersects with broader societal interests. Through its nuanced exploration of human behavior and incentive structures, the Prisoner's Dilemma offers valuable insights applicable to diverse contexts and decision-making frameworks.

5.2 Delving into Extensive-Form Games

Definition 6 (Extensive-Form Games). *a detailed description of the sequential structure of the decision problems encountered by the players in a strategic situation. There is perfect information in such a game if each player, when making any decision, is perfectly informed of all the events that have previously occurred. An extensive game with imperfect information differs in that the players may in addition be imperfectly informed about some (or all) of the choices that have already been made.*

An extensive game has the following components:

```
import nashpy as nash
import numpy as np

# Create the game with the payoff matrix
g1 = np.array([[3, 3], [0, 5]])
# g1 is the row player
g2 = np.array([[5, 0], [1, 1]])
# g2 is the column player

# Form the game
game_1 = nash.Game(g1,g2)
game_1

# Find the Nash Equilibrium
equilibria = game_1.support_enumeration()
for eq in equilibria:
    print(eq)
```

Table 2. An example of a Nashpy code finding the Nash Equilibrium. Graciously provided by Professor Luyao Zhang for the students of CS/ECON 206 course.

- *A set N (the set of players).*
- *A set H of sequences (finite or infinite) that satisfies the following three properties:*
  1. *The empty sequence $\emptyset$ is a member of H.*
  2. *If $(a_k)_{k=1,\ldots,K} \in H$ (where K may be infinite) and $L < K$ then $(a_k)_{k=1,\ldots,L} \in H$.*
  3. *If an infinite sequence $(a_k)_{\infty k=1}$ satisfies $(a_k)_{k=1,\ldots,L} \in H$ for every positive integer L then $(a_k)_{\infty k=1} \in H$.*

Each member of H represents a history, where each component of a history is an action taken by a player. A history $(a_k)_{k=1,\ldots,K} \in H$ is considered terminal if it is infinite or if there is no a_{K+1} such that $(a_k)_{k=1,\ldots,K+1} \in H$. The set of terminal histories is denoted by Z. [8] [7]

The Centipede game, a classic example of an extensive-form game, captivates me due to its illustration of altruism/personal gain dynamics. In this sequential game, players must decide whether to continue passing or to seize a progressively increasing reward, balancing short-term gains against the risk of losing out on potential future payoffs (see Figure 1). Rosenthal original analysis underscores the strategic depth arising from anticipating opponents' moves and the complexities of timing and commitment. The game has regained popularity recently in the media through variations such as "take something, or double it and give it to the next person." This resurgence underscores its enduring relevance in illustrating strategic tension and psychological dynamics.

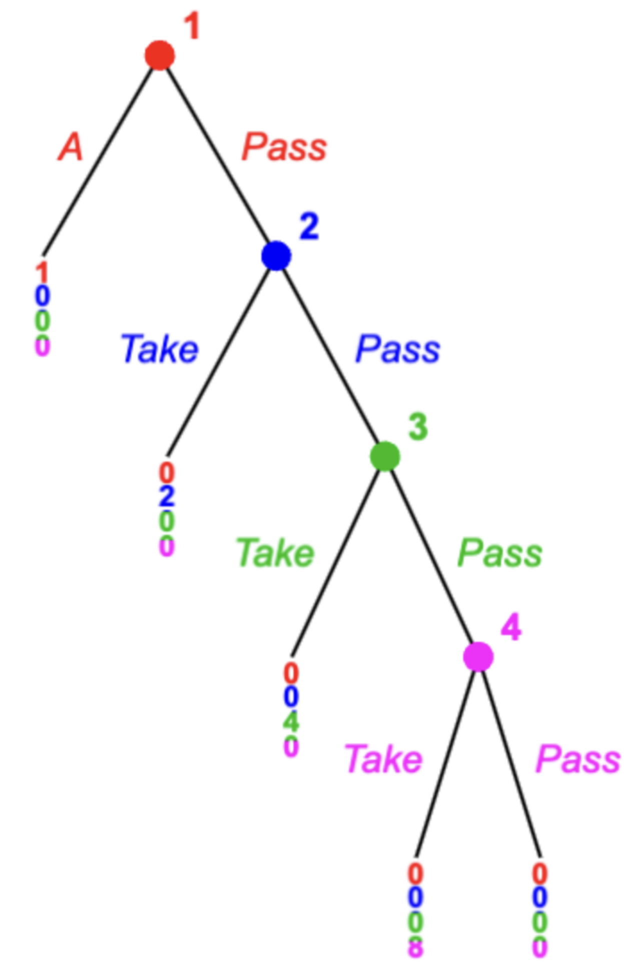

Fig. 1. Game Theory Explorer - Example of a Centipede Game.

5.3 Critiquing Nash Equilibrium and Envisioning Innovations:

The scalability of Nash Equilibrium analysis to games with a higher number of players is a broader challenge within the field of game theory and strategic analysis. Nash Equilibrium itself struggles to provide actionable insights in games with a large number of players due to the exponential increase in complexity. This challenge extends to various analytical tools used in game theory, including Nashpy, as they rely on the foundational concepts of Nash Equilibrium for their analyses. Therefore, while the analytical tools may be useful for studying Nash Equilibria in simpler games or scenarios, the broader limitation of scalability to complex multi-player interactions is inherent to Nash Equilibrium theory itself, impacting all tools and methodologies based on it. The scalability challenge is particularly pronounced in the context of the prisoner's dilemma when multiple iterations involve more than two players. As more individuals are added to the game, the complexity increases exponentially, making it exceedingly difficult to compute Nash Equilibria analytically.

Another limitation of Nash Equilibrium is its assumption of complete information, which may not hold in many real-world scenarios. In practice, players often have imperfect or asymmetric information about their opponents' strategies, payoffs, or even the structure of the game itself. This asymmetry can lead to strategic uncertainty and deviations from equilibrium predictions as players make decisions based on their subjective beliefs or incomplete information. Consequently, traditional Nash Equilibrium analysis may fail to accurately capture the complexities of decision-making in situations of information asymmetry, highlighting the need for alternative modeling approaches that account for uncertainty and incomplete information. For instance, they might be uncertain about the likelihood of their partner defecting or cooperating. This asymmetry in information can lead to deviations from the Nash Equilibrium predictions, as players make decisions based on their subjective beliefs rather than objective knowledge.

To address these limitations, a new tool or conceptual framework could integrate advanced computational techniques with game theory principles. Reinforcement Learning within Game Theory offers a dynamic solution for modeling strategic interactions. Through allowing agents to learn and adapt their strategies over repeated interactions, we can grasp the evolving nature of decision-making in scenarios such as the prisoner's dilemma. Agents can employ strategies like tit-for-tat or forgiveness, leading to deeper analyses of long-term cooperation dynamics.

Moreover, Hierarchical Learning Architectures introduce an innovative framework for modeling multiplayer games. By organizing agents into hierarchical layers, we can capture strategic interactions at various levels of abstraction, accommodating the complexity of real-world interactions and enabling a more comprehensive understanding of strategic dynamics.

Additionally, Bayesian Game Theory provides a principled framework for addressing uncertainty and incomplete information. By modeling players' beliefs as probability distributions and updating them based on observed actions and outcomes, we can accurately capture strategic reasoning. Integrating Bayesian techniques into the analysis of the prisoner's dilemma allows for more realistic predictions of behavior and more informed decision-making.

These advancements not only improve scalability and adaptability but also capture the intricacies of real-world interactions, empowering decision-makers to navigate strategic environments more effectively. I am excited about the potential of these approaches to push the boundaries of game theory and contribute to more effective decision-making in strategic contexts, as shown in Figure 2.

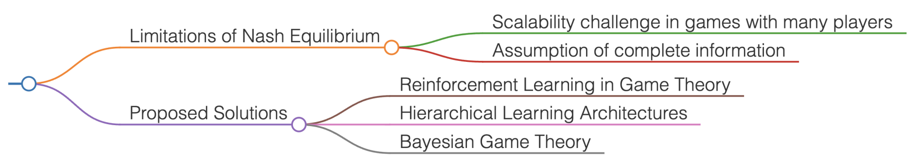

Fig. 2. Nash Equilibrium limitations and proposed solutions. Created with Markmap.

Bayesian (Subgame Perfect) Nash Equilibrium

Definition 7 (Subgame). *In the context of a perfect-information extensive-form game, the subgame of G located at node h refers to the limited portion of G that includes only the descendants of h. The collection of subgames of G encompasses all such limited portions of the game rooted at various nodes within G. [7]*

Definition 8 (Subgame Perfect Nash Equilibrium). *According to Shoham and Leyton-Brown, Subgame-perfect equilibria (SPE) of a game G are defined as strategy profiles s such that, for any subgame G' of G, the strategy profile s, when applied within G', constitutes a Nash equilibrium of G'. According to Osborne and Rubinstein, a subgame-perfect equilibrium is characterized by a strategy profile s^* in Γ, where for any history h, the strategy profile $s^*_{|h}$ forms a Nash equilibrium of the subgame $\Gamma(h)$.*

Theorem 1. *Single-deviation Principle: In a multistage game with continuous play extending to infinity, a strategy profile qualifies as a subgame-perfect Nash equilibrium if, and only if, it successfully withstands the single-deviation test for each player at every stage. [11]*

Proof. According to MIT OpenCourseWare, the single-deviation test assesses a strategy profile σ^*. At any stage, with fixed moves for other players as per σ^*, can a player improve their payoff by deviating from σ^*? If so, σ^* fails the single-deviation test for that player at that stage. Failing this test for any player implies σ^* cannot be a subgame-perfect equilibrium, as it doesn't induce a Nash equilibrium in the subsequent subgame. Conversely, in a multistage game continuous at infinity, if σ^* passes the single-deviation test for every player at every stage, it qualifies as a subgame-perfect equilibrium.

Example 1. In the Centipede game, players take turns deciding whether to continue or stop, with each turn representing a stage. The single-deviation test assesses whether deviating from the equilibrium strategy at any stage would yield a higher payoff for a player. If a deviation leads to a higher payoff, the equilibrium strategy fails the single-deviation test, indicating it's not subgame-perfect.

Subgame-perfect Nash equilibrium is vital in understanding the limitations of Nash Equilibrium, as it offers a more stringent criterion that accounts for dynamic decision-making and strategic credibility over time. By requiring strategies to withstand deviations at each stage, subgame-perfect equilibrium reveals vulnerabilities in traditional Nash Equilibrium, particularly in sequential and repeated interactions. This distinction sheds light on the complexities of strategic behavior and equilibrium outcomes, offering insights into more realistic models and predictions in game theory.

6 Game Theory Glossary Tables

Terminology	**Definition**	**Author**
Strategic or Normal Form Games	A type of game representation that lists all players, their possible strategies, and the resulting payoffs in a matrix or table format. In this form, players make decisions simultaneously without knowing the actions chosen by other players.	Nash
Extensive-Form Games	A type of game representation that captures the sequential nature of decision-making by depicting the possible actions of players as nodes in a tree-like structure. It includes information about the order of play, possible choices at each decision point, and payoffs associated with different outcomes.	Selten
Nash Equilibrium	A stable state in a game where each player's strategy is optimal given the strategies chosen by the other players, and no player has an incentive to unilaterally deviate from their chosen strategy.	Nash
Sequential Games with Perfect Information	Games where players have complete information about previous actions and outcomes, allowing for the analysis of optimal strategies.	Rosenthal
Bayesian Games	Games in which players have private information that affects their payoffs or beliefs about the game.	Harsanyi
Utility Theory	A theory that models individual preferences or choices in terms of utility functions, which assign numerical values to outcomes or alternatives based on their desirability or satisfaction.	Neumann and Morgenstern
Zero-Sum Games	Games where the total utility or payoff is constant, meaning one player's gain is exactly balanced by another player's loss.	Neumann and Morgenstern
Minimax Theorem	A fundamental result in game theory that states that in zero-sum games with perfect information, there exists a value known as the minimax value, and optimal strategies for both players that achieve this value.	Neumann and Morgenstern
Incomplete Information	A situation in game theory where players have imperfect or incomplete knowledge about the game environment, including the strategies, payoffs, or preferences of other players.	Harsanyi

7 Appendix

7.1 Python Code: Bayesian Cake-Cutting Game Framework

```python
import random
from collections import Counter

def reach_consensus(preferences):
    vote_counts = Counter(preferences)
    max_votes = max(vote_counts.values())
    consensus = [preference for preference, count in
        vote_counts.items() if count == max_votes]
    return consensus

def update_preferences_beliefs(preferences, resources):
    posterior_beliefs = {preference: 0 for preference in
        resources}
    for preference in preferences:
        posterior_beliefs[preference] += 1
    total_counts = sum(posterior_beliefs.values())
    for preference in posterior_beliefs:
        posterior_beliefs[preference] /= total_counts
    return posterior_beliefs

def negotiate_allocation(resources, num_parties):
    all_preferences = {}
    for i in range(num_parties):
        preferences = {}
        for resource in resources:
            preference_rank = int(input(f"Party {i+1}, rank
                your preference for {resource} (1 being
                highest): "))
            preferences[resource] = preference_rank
        all_preferences[f"Party {i+1}"] = preferences

    final_preferences = {}
    for resource in resources:
        preferences_for_resource = {party: preferences[
            resource] for party, preferences in
            all_preferences.items()}
        consensus = max(preferences_for_resource, key=
            preferences_for_resource.get)
        final_preferences[consensus] = resource

    return final_preferences

def generate_random_projects(spheres):
    random_projects = {}
    for sphere, projects in spheres.items():
        random_project = random.choice(projects)
```

```
        random_funding = random.randint(10, 250) * 1000000  #
            Funding in millions
        random_projects[sphere] = {"project": random_project,
            "funding": random_funding}
    return random_projects

# Define spheres and projects
spheres = {
    "Economy": [
        "TradeNet Initiative", "BusinessBoost Project", "
            MarketEase Platform",
        "ExportPro Program", "CommerceConnect", "TradeAI
            Network",
        "GlobalGrowth Initiative", "MarketLink Project", "
            ExportEdge Program",
        "TradeTech Solutions"
    ],
    "Social Policy": [
        "CommunityCare Initiative", "HealthBridge Project", "
            EmpowerAll Program",
        "CommunityConnect Initiative", "WellnessHub Project",
            "CareCompass Program",
        "AidAccess Initiative", "HopeHarbor Project", "
            CommunitySupport Program",
        "EqualOpportunity Initiative"
    ],
    "Environment": [
        "EcoSmart Initiative", "GreenGrid Project", "
            EarthGuard Program",
        "EcoLife Initiative", "RenewaTech Project", "
            ClimateCare Program",
        "EcoBalance Initiative", "SustainaCity Project", "
            GreenPulse Program",
        "EcoCycle Initiative"
    ],
    "Culture": [
        "CulturEase Initiative", "ArtFusion Project", "
            HeritageLink Initiative",
        "CultureBridge Program", "UnityCanvas Initiative", "
            ArtisanCraft Project",
        "GlobalCultural Initiative", "TraditionsAlive Program"
            , "CulturAware Initiative",
        "ArtisanConnect Program"
    ]
}

# Generate random projects
random_projects = generate_random_projects(spheres)

# Output the randomly generated projects
```

```
print("Good time of the day, dear party representatives.")
print("Our today's agenda is to determine where we are going
    to allocate funds for one of the next innovative projects
    .\n")
print("Here are the projects on our today's agenda:\n")
for sphere, project_info in random_projects.items():
    print(f"{sphere}:\n- Project: {project_info['project']}\n-
        Funding: (${project_info['funding'] // 1000000}
        Millions)\n")

# Use randomly generated projects in the negotiation process
resources = [project_info["project"] for project_info in
    random_projects.values()]
final_preferences = negotiate_allocation(resources, 3)
print("Bayesian probabilities before final choice:")
posterior_beliefs = update_preferences_beliefs(list(
    final_preferences.values()), resources)
for preference, probability in posterior_beliefs.items():
    print(f"{preference}: {probability:.2f}")
print("\nFinal allocation:", list(final_preferences.values())
    [0])
```

Bibliography

[1] V. Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton University Press, 1947. [Online]. Available: https://www.jstor.org/stable/j.ctt1r2gkx

[2] J. Nash, "Equilibrium points in n-person games," *Proceedings of the National Academy of Sciences*, vol. 36, no. 1, pp. 48–49, 1950. [Online]. Available: https://www.pnas.org/content/36/1/48

[3] J. C. Harsanyi, "Bargaining in ignorance of the opponent's utility function," *Journal of Conflict Resolution*, vol. 6, no. 1, pp. 29–38, 1962.

[4] L. S. Shapley, "Stochastic games," *Proceedings of the National Academy of Sciences*, vol. 39, no. 10, pp. 1095–1100, 1953. [Online]. Available: https://www.pnas.org/content/39/10/1095

[5] J. C. Harsanyi, "Games with incomplete information played by "bayesian" players. I–III," *Management Science*, vol. 14, no. 3, pp. 159–182, 1967.

[6] R. Axelrod, "The evolution of cooperation," *Science*, vol. 211, no. 4489, pp. 1390–1396, 1984.

[7] Y. Shoham and K. Leyton-Brown, *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.

[8] M. J. Osborne and A. Rubinstein, *A course in game theory*. MIT Press, 1994.

[9] M. Stephens and V. Knight, "Nashpy: Library for the computation of nash equilibria," https://github.com/drvinceknight/Nashpy, 2018.

[10] R. W. Rosenthal, "Sequential games with perfect information," *Econometrica*, vol. 49, no. 1, pp. 17–25, 1981.

[11] MIT OpenCourseWare, "Chapter 11 subgame-perfect nash equilibrium," 2012. [Online]. Available: https://ocw.mit.edu/courses/14-12-economic-applications-of-game-theory-fall-2012/resources/mit14_12f12_chapter11/

[12] R. Selten, "Reexamination of the perfectness concept for equilibrium points in extensive games," *International Journal of Game Theory*, vol. 4, no. 1, 1975.