

A Robust Optimization Method for Two-Person Repeated Games with Incomplete Information

Xiaoyi YAN

*School of Mathematics and Systems Science, Wuhan University of Science and Technology, Wuhan
430065, China*

E-mail: 2878774199@qq.com

Yuqiang FENG*

*School of Mathematics and Systems Science, Wuhan University of Science and Technology, Wuhan
430065, China*

E-mail: yqfeng6@126.com

Xianjia WANG

School of Economics and Management, Wuhan University, Wuhan 430072, China

E-mail: wangxj@whu.edu.cn

Jun JIANG

*School of Mathematics and Systems Science, Wuhan University of Science and Technology, Wuhan
430065, China*

E-mail: jiangjun@wust.edu.cn

Abstract In this paper, we discuss two cases of two-player repeated games in an environment of incomplete information with randomness and cognitive uncertainty, where incomplete information refers to the situation in which the transition probabilities and player payoffs are uncertain. We use robust optimization techniques to handle data uncertainty and determine the optimal solution within this framework. Our contributions are threefold: Firstly, we apply Markov processes to the repeated game model. Secondly, we propose an effective robust optimization method that can handle uncertain data in the context of incomplete information and solve the uncertainty problem in different types of repeated games. Finally, our method is more general than previous methods and can adapt to various types of data, providing decision support for risk-averse players.

Keywords incomplete information; two-player repeated games; Markov decision making; robust optimization methods

1 Introduction

Game theory is a theory that examines the interactions between decision-makers in competitive or cooperative settings, analyzing how individuals make choices to maximize their interests. Classical game theory addresses game problems under complete information, where all participants clearly understand the game's rules, strategy space, payoff functions, and the strategic choices of their opponents. These games assume that all relevant input variables (e.g., payoffs) are known and certain. However, this assumption ignores the impact of uncertainties on strategy choice and resource allocation, potentially leading to inaccurate strategies and making it difficult to achieve a Nash equilibrium. In reality, information is often incomplete, and participants usually lack full knowledge of their own or others' payoffs, which complicates the game problem and requires seeking solutions in an uncertain environment^[1,2].

Uncertainties originate from various sources, and their precise quantification is crucial for further analysis. In the context of incomplete information games and decision theory, data uncertainty is typically categorized into two primary types: stochastic uncertainty and cognitive uncertainty^[3]. These two types of uncertainty are conceptually distinct and have different implications for the decision-making process. For handling uncertainties of various forms, the literature commonly identifies three main solution approaches. The first category involves using probabilistic methods to address stochastic payoffs, with a primary emphasis on managing random uncertainties within the game framework. Harsanyi^[4] introduced a Bayesian framework for modeling games with incomplete information, where he postulated that participants maximize their expected payoffs by estimating unknown parameters from prior probability distributions and other available data. Additionally, Cheng et al.^[5] Su and Ji^[6] have also employed probabilistic techniques in their research. However, probabilistic methods require the collection of large amounts of historical data. As noted by Gao^[1], accurate data is often difficult to obtain, especially in gaming systems involving human decisions. To address this, researchers propose a second class of methods using information based on return intervals to manage uncertainty. These approaches address cognitive uncertainty, often assuming gains as fuzzy sets. Fuzzy game theory has attracted the attention of many scholars^[7–9]. Collins and Hu^[10] extended the classical deterministic matrix by defining a fuzzy binary comparison strategy for a fuzzy deterministic interval matrix game. Under the theoretical framework of uncertainty, Li and Yang^[2] proposed a novel mathematical framework designed to address games characterized by incomplete information and uncertain payoffs. They created another new form of game to solve it, providing a theory demonstrating that the equilibrium is the same for both games. The third method is known as the robust optimization method, which also assumes that information about returns can be obtained as intervals. It has the advantage that it does not require information about the specific distribution of benefits. Yu and Liu^[11] proposed a non-distributional model for multi-objective games with incomplete information, using robust optimization methods for each objective to handle payoff uncertainty. Qu and Meng^[12] presented a distributionally robust optimization approach for finite state/action N-player non-zero-sum games with incomplete in-

formation, combining stochastic with robust optimization methods to address game settings. Dey and Zaman^[13] proposed robust optimization methods for solving two-person zero-sum and non-zero-sum incomplete information games considering single or multiple interval inputs (e.g., interval-valued gains). As far as we know, most of the existing methods only deal with single interval data. Nevertheless, in real-world scenarios, obtaining a single interval might be challenging due to the presence of data from various sources, such as expert opinions. Therefore, there is a need for a method that solves two-player games by integrating information from multiple sources. Dey and Zaman introduced a robust optimization approach for addressing two-player games with interval uncertainty, driven by both single and multiple interval data. This method focuses on optimizing the mean while reducing the variance of the game outcomes. They addressed the uncertainty associated with payoffs generated from interval data by using robust optimization methods. Zaman et al.^[14] proposed the moment boundary algorithm to compute the first four moments of multiple intervals of data by aggregating information from multiple sources. The moment boundary algorithm is also used in our proposed robust optimization framework for two-player repeated games.

In recent years, robust optimisation has been widely applied and has achieved notable advances: Maggioni et al.^[15] introduced a sampling–minimax framework that yielded rigorous error bounds and convergence guarantees for multi-stage problems. Beck et al.^[16] presented a systematic survey of the latest models and solution techniques for robust bilevel optimization and min–max Stackelberg games, providing computationally tractable equivalence classes. Li et al.^[17] framed the interaction between demand-perturbing attackers and real-time bus schedulers as a Markov game and solved it with adversarial DDQN, demonstrating robust adaptive scheduling in Hong Kong’s bus network.

Motivated by above results, we study non-cooperative, dynamic, two-player repeated games under incomplete information, the typical case in previous studies involves the presence of Nash equilibria in repeated iterations of the Prisoner’s Dilemma^[18,19]. Researchers have explored the evolution of cooperation and betrayal strategies, equilibrium analysis, and behavioral patterns in long-term interactions through theoretical analysis and experimental validation. Introducing incomplete information to repeated games extends the classical game theory framework, making it more reflective of real-world complexities. Gensbittel and Oliu-Barton^[20] examined optimal strategies for zero-sum games in situations with incomplete information, later Oliu-Barton^[21] studied the optimal strategy of repeated games with small weights under information incompleteness. Other scholars have explored the dynamic analysis of the above games^[22].

In this work, we make three key contributions. Firstly, we introduce an innovative application of Markov processes to model repeated games. Recognizing that the outcome of each round in a repeated game depends not only on the strategies chosen in that round but also on the strategies and outcomes of previous rounds, we utilize Markov processes to capture the dynamics of repeated games and develop a two-player repeated game model under certainty. Secondly, we propose robust optimization methods to address uncertainty in data under incomplete informa-

tion and solve the uncertainty problem across different types of repeated games by optimizing both the mean and minimizing the variance of the objective function, thereby achieving objective robustness. Additionally, feasibility robustness is attained by using a feasible domain approximation method to satisfy constraints. Finally, the proposed method is generalized to handle various types of data, providing decision support for risk-averse players.

The rest of the paper is structured as follows. Section 2 reviews techniques for solving two-person repeated games. Section 3 describes a robust optimization method for handling two-player repeated games in the presence of data uncertainty. Section 4 verifies the feasibility of the proposed method through numerical examples. Finally, section 5 offers conclusions and suggestions for future research.

2 Two-player Repeated Game

2.1 A Review of Repeated Games Between Two Players

Repeated games are a key area of game theory that examine how players make decisions to maximize payoffs over multiple rounds. As a typical form of dynamic games, repeated games involve playing a single game with a consistent structure multiple times with the same players, strategy choices, and payoff calculations. Unlike single-play games, repeated games consider multiple interactions between players and the effects of previous behaviors and outcomes on future decisions. Considering a typical Prisoner's Dilemma^[23] as an example, as shown in Table 1, players have two choices: cooperation (C) or betrayal (D). In a single game, mutual cooperation yields the highest collective benefit. However, in repeated games, players decide their current strategy at each stage based on historical behaviors. After each round, players may adopt several common strategies: a constant strategy (always cooperate or always betray), a trigger strategy (permanently betray once the other party betrays), a punishment strategy (punish the other party for a period following betrayal, then resume cooperation), or a reciprocal strategy (mirror the other party's previous action). In a one-shot game, based on the minimax principle, both players typically choose betrayal, resulting in a Nash equilibrium. However, in a repeated game, mutual betrayal across rounds leads to lower payoffs compared to mutual cooperation. Therefore, cooperation can become a stable strategy through appropriate tactics, such as the trigger strategy. By making suitable adjustments, if both parties ultimately choose the (C, C) strategy with a stable probability of 1, pareto optimality can be achieved.

Table 1 Payment matrix for the prisoner's dilemma single-stage game^[18]

player 1	player 2	
	cooperation(C)	betrayal(D)
cooperation(C)	(3, 3)	(0, 5)
betrayal(D)	(5, 0)	(1, 1)

Repeated games have extensive applications in economics, political science, biology, and other fields. Previous studies on repeated games mainly focus on the following aspects:

1) Repeatability of the game: Repeated games involve adapt to changes in their opponents' behaviors or the environment.

2) Stability of equilibrium: The equilibrium in repeated games is more complex than in single games. Besides Nash equilibrium, other types of equilibria, such as evolutionary stable strategies, can also occur during the game process, unlike single games which occur only once. This allows players to make decisions at multiple points in time, with previous behaviors and outcomes influencing future decisions.

3) Evolution of strategy: In repeated games, players can gradually adjust their responses based on past experiences and observations. They gradually learn and evolve to exist.

4) Cooperation and betrayal: In repeated games, players may choose between cooperation and betrayal. Long-term cooperation can yield higher returns, but short-term gains might be achieved if one party betrays.

In summary, the study of repeated games encompasses theories and methods from various fields such as game theory, dynamic programming, and reinforcement learning. By analyzing repeated games, we can gain a better understanding of players' decision-making behavior in long-term interactions and propose corresponding strategic suggestions and policy measures.

2.2 Introduction to Markov Decision Process

In this paper, we use Markov decision processes (MDP)^[23] to characterize the decision-making process in two-player repeated games. MDP is a mathematical framework for describing decision-making problems in stochastic environments, widely used in reinforcement learning, operations research, and artificial intelligence. MDP models decision-making problems where choices are made in a stochastic environment, and the current state influences the transition probabilities and immediate rewards of future states.

The main components and characteristics of MDP are as follows: 1) States: States in an MDP are abstract concepts that describe the environment, which are used to describe different scenarios or configurations the system may have, typically denoted by the symbol (S).

2) Actions: In each state, the decision-making agent can take various actions, which are usually represented by the symbol (A).

3) Transition probabilities: Transition probabilities are the magnitude of the likelihood that the system will transfer from one state to another after performing an action. Typically, ($P(s'|s, a)$) denotes the probability of transferring from state (s) to state (s') after performing action (a).

4) Reward function: In an MDP, each state transition is accompanied by an immediate reward. The reward function defines the immediate reward for each action taken in each state. Typically, ($R(s, a, s')$) denotes the reward obtained by transitioning to state (s') after performing action (a) in state (s).

Based on the above elements, we can formally describe the Markov decision process (MDP)

using a four-tuple (S, A, P, R) . The purpose of solving the MDP issue is to find a perfect policy that maximizes the total reward accumulated over the long-term decision process. Common solution methods include value iteration, policy iteration, Q-learning, and deep Q-networks.

2.3 Using Markov Process to Describe Two-person Repeated Game Process (Based on 2-by-2 Matrix Game)

The advantage of using Markov processes to describe repeated game decisions lies in understanding the decision-making process of players in different states and the dynamic evolution of the game. By constructing state transition matrices (including transition probabilities under deterministic or stochastic conditions), we can analyze the long-term behavioral trends of players under different strategies and study the equilibrium solutions of the game.

Considering a game that can be repeated indefinitely with two players, player 1 adopts the strategy e_i , $i = 1, 2$ and player 2 adopts the strategy f_j , $j = 1, 2$. In each round of the decision-making process, players have two strategy choices. For player 1 and player 2, there are four combinations of strategies:

$$O_1 = (e_1, f); O_2 = (e_1, f); O_3 = (e_2, f); O_4 = (e_2, f),$$

Under each strategy combination, player 1 and player 2 receive corresponding payoffs, denoted as E_{ij} and F_{ij} , respectively, representing the payoffs when player 1 and player 2 choose their respective strategies.

The state space of the two players in an infinitely repeated game is denoted as $S_1 = S_2 = S$, where:

$$S = \{s = (i, \Delta) : \Delta \in \{O_1, O_2, O_3, O_4\}, i = 1, 2, \dots, n\}.$$

The Δ denotes the combination of strategies that the two players at each stage, which can be more simply denoted as 1, 2, 3, 4.

Therefore, the set S contains $4n$ possibilities, representing the condition of the player at each step, with n being the number of times the game is repeated. For example, $(1, 1)$ means the player is in state O_1 at stage 1, and $(8, 3)$ means the player is in state O_3 at stage 8.

The action spaces A_1, A_2 for two players in an infinitely repeated game are defined as follows: Each player has two alternative strategies in each round of the game, player 1's action space is $A_1 = \{e_1, e_2\}$ and player 2's action space is $A_2 = \{f_1, f_2\}$. Before selecting a strategy for the next round, players observe the decisions made in the current round, which cause the overall state of the game to shift. For the entire game (not a subgame), the set of possible subsequent states from the current state constitutes the set of available actions: $A(i) = \{j : j \in S\}, \forall i \in S$. For example, if in the second round player 1 and player 2 are in state O_1 , and in the third round they move to state O_2 , this implies that player 1 chooses strategy e_1 and player 2 chooses strategy f_2 . This transition is represented from the state space $(2, 1)$ to $(3, 2)$. Each new round of the game will have four possible state spaces, forming the action space A of the entire game. For instance, the action space for the game in the next round from state $(2, 1)$:

$$A(2, 1) = \{(3, 1), (3, 2), (3, 3), (3, 4)\}.$$

Thus, for the entire repeated game, the collection of possible actions available to each player in the next round can be denoted as:

$$A(i, \Delta) = \{(i + 1, 1), (i + 1, 2), (i + 1, 3), (i + 1, 4)\}.$$

The state transition law ($P(s'|s, \mathbf{a})$) for two players in an infinitely repeated game (where $\mathbf{a}$ is a vector representing the action choices of the two players) can be described as follows: We assume that these choices will be influenced by the outcome of the previous round, without knowing the strategic choices the player will make in the next round. We suppose that player 1 chooses strategy e_1 in state O_i with probabilities (p_1, p_2, p_3, p_4) , and player 2 chooses strategy f_1 in state O_i with probabilities (q_1, q_2, q_3, q_4) . Specifically, at the end of a round, player 1 chooses e_1 with probability p_i and e_2 with probability $1 - p_i$; similarly, player 2 chooses f_1 with probability q_i and f_2 with probability $1 - q_i$. Thus, for the previously constructed state space, the state transition law ($P(s'|s, a)$) at the beginning of each round can be represented by the following matrix $\mathbf{P}$:

$$\mathbf{P} = \begin{pmatrix} p_1 q_1 & p_1(1 - q_1) & (1 - p_1)q_1 & (1 - p_1)(1 - q_1) \\ p_2 q_2 & p_2(1 - q_2) & (1 - p_2)q_2 & (1 - p_2)(1 - q_2) \\ p_3 q_3 & p_3(1 - q_3) & (1 - p_3)q_3 & (1 - p_3)(1 - q_3) \\ p_4 q_4 & p_4(1 - q_4) & (1 - p_4)q_4 & (1 - p_4)(1 - q_4) \end{pmatrix}.$$

Specifically, the third row and third column of the matrix represent $P(O_3|O_3, a) = (1 - p_3)q_3$. This means that if the game was in state O_3 in the previous round, player 1 chooses e_2 with probability $(1 - p_3)$, and player 2 chooses f_1 with probability q_3 . Consequently, the state of the next round remains O_3 .

The reward function for two players under an infinitely repeated game:

In each round of the game, players make decisions and receive their respective payoffs. When player 1 selects strategy i and player 2 selects strategy j , player 1's revenue is E_{ij} , and player 2 is F_{ij} . Considering the states O_1, O_2, O_3, O_4 , the payoffs for player 1 are $E_{11}, E_{12}, E_{21}, E_{22}$, and the payoffs for player 2 are $F_{11}, F_{12}, F_{21}, F_{22}$. In previous discussions, the reward function for two players in an infinitely repeated game was derived by observing the payoff matrix for short-term decisions. However, in realistic conditions, the longer the decision-making process takes, the more time-consuming and less favorable it becomes for the players' payoffs. Thus, long-term decisions are influenced by external factors, such as the introduction of a penalty (discount) factor, which plays a key role in calculating the total payoffs. The penalty factor γ (a number between 0 and 1) indicates the present value weight of future returns. A smaller γ suggests that the player values immediate returns more highly, whereas a larger γ implies a greater value placed on future returns. The revenue function for player i , starting from state s , is defined as:

$$V_i(s) = \max_{\alpha_i \in A_i} \sum_{\alpha_j \in A_j} \sum_{s' \in S} P(s' | s, a_i, a_j) [R_i(s, a_i, a_j) + \gamma V_i(s')].$$

This function denotes the total expected gain of player i in future rounds starting from state s . More simply, the total payoff of a repeated game player with a penalty factor can be expressed as:

$$U^i = \sum_{t=0}^{\infty} \lambda^t u^i(e_t^1, e_t^2).$$

The reward function draws on an important concept from Markov decision processes: A smooth distribution that converges to a stable state when repeated infinitely many times. This is related to the initial decision and the state transition, denoted by π , a row vector representing the probability distribution of the final state that satisfies the equation $\pi = \pi P$, where P is the state transition matrix. Accordingly, player 1's final reward function is:

$$R_1 = \pi_1 E_{11} + \pi_2 E_{12} + \pi_3 E_{21} + \pi_4 E_{22}.$$

Similarly, player 2's final reward function is:

$$R_2 = \pi_1 F_{11} + \pi_2 F_{12} + \pi_3 F_{21} + \pi_4 F_{22}.$$

3 Two-person Repeated Game with Incomplete Information

In Section 2, we outline the state transition probabilities and payoffs for the players. However, in real life, players often lack complete information and may not be aware of their opponents' types, action spaces, preferences, or payoffs. Furthermore, players might only partially observe their opponents' actions, receiving incomplete or noisy information. Environmental uncertainty also impacts decision-making, as players' payoffs may be probabilistic rather than deterministic, and external random factors can alter state transition probabilities. Such uncertainties are often categorized as incomplete information. In essence, in repeated games, both the probabilities and payoffs associated with strategy choices may be unknown. Subsequently, we will explore game strategies under incomplete information from the perspectives of stochastic and cognitive uncertainty. To facilitate understanding, we will first construct a linear programming model to represent a two-player repeated game under deterministic conditions.

3.1 Two-person Repeated Games under Determinism

In Section 2, we redefine the repeated game described by a Markov process for players 1 and 2, aiming to maximize their final payoffs, which are influenced by the stationary distribution in the Markov chain. We assume that the final stationary distribution is $(\pi_1, \pi_2, \pi_3, \pi_4)$, where each π_i is determined by the state transfer probabilities in the system, satisfying $\pi = \pi P$. To facilitate our discussion, we re-express the state transfer matrix P in terms of (a_{ij}) for $i, j = 1, 2, 3, 4$:

$$P = \begin{pmatrix} p_1 q_1 & p_1(1 - q_1) & (1 - p_1)q_1 & (1 - p_1)(1 - q_1) \\ p_2 q_2 & p_2(1 - q_2) & (1 - p_2)q_2 & (1 - p_2)(1 - q_2) \\ p_3 q_3 & p_3(1 - q_3) & (1 - p_3)q_3 & (1 - p_3)(1 - q_3) \\ p_4 q_4 & p_4(1 - q_4) & (1 - p_4)q_4 & (1 - p_4)(1 - q_4) \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}.$$

Next we transform the two-player repeated game with complete information into a linear programming problem to solve for the maximum long-run payoff of the two players, which can be expressed in the following equations:

$$\begin{aligned}
 & \max R_1 = \pi_1 E_{11} + \pi_2 E_{12} + \pi_3 E_{21} + \pi_4 E_{22} \\
 \text{s.t. } & \sum_{i=1}^4 a_{ij} \pi_i = \pi_j, \quad j = 1, 2, 3, 4, \\
 & \sum_{j=1}^4 a_{ij} = 1, \quad i = 1, 2, 3, 4, \\
 & \sum_{i=1}^4 \pi_i = 1,
 \end{aligned} \tag{1}$$

$$\begin{aligned}
 & \max R_2 = \pi_1 F_{11} + \pi_2 F_{12} + \pi_3 F_{21} + \pi_4 F_{22}, \\
 \text{s.t. } & \sum_{i=1}^4 a_{ij} \pi_i = \pi_j, \quad j = 1, 2, 3, 4, \\
 & \sum_{j=1}^4 a_{ij} = 1, \quad i = 1, 2, 3, 4, \\
 & \sum_{i=1}^4 \pi_i = 1.
 \end{aligned} \tag{2}$$

In the above equations, the variables involved are known, and next, we discuss the two cases where the gain and state transfer probabilities are under the stochastic uncertainty and cognitive uncertainty.

3.2 Two-person Repeated Games under Stochastic Uncertainty

We utilize robust optimization techniques to handle the case of incomplete information. Robustness is defined as the reduction of variance while optimizing the mean of the objective function. In a game that lacks complete information, robust optimization considers the uncertainty of input parameters, this reduces the sensitivity of the problem to changes in random returns. Implementing robust optimization requires ensuring both stability of the objective function and viable programs. This is achieved by simultaneously optimizing the mean of the objective function and minimizing its variance measure. When there are unknown factors, stability of viable programs is attained by satisfying constraints. Methods to ensure feasibility robustness include viable area reduction methods, analysis of the most unfavorable scenarios and corner space assessment.

To achieve robustness, the mean and variance of the objective and constraint functions must be estimated. Common estimation methods include Taylor series expansion, sampling-based methods and point estimation methods. Since robust optimization involves optimizing the mean and minimizing the variance of the objective function, multi-objective optimization techniques are essential. Widely discussed methods in the robust optimization literature include

the weighted sum method, ε -constraint method, objective planning and compromise planning. In the proposed robust optimization framework, we use variance as a weigh of the change in the objective function for objective robustness, feasible domain approximation method for stability of viable programs and weighted sum method for aggregating many targets.

Stochastic strategies are introduced when analyzing a two-player repeated game under stochastic uncertainty, which may arise from uncertainty in the state of nature, uncertainty in the players' strategies, or stochastic effects of the external environment. Players make probabilistic decisions in each round of the game, achieved by introducing a probability distribution for the choice of strategies. Additionally, the payoffs for the two players are randomized. For instance, in a two-player repeated game, player 1 chooses strategy p_1 with probability e_1 in the next round, and strategy e_2 with probability $(1 - p_1)$. Similarly, player 2 chooses strategy q_1 with probability f_1 in the next round, and strategy f_2 with probability $(1 - q_1)$. Based on historical data, the state transfer probabilities p_1 and q_1 , as well as the payoffs of both players, follow certain probability distributions. Under stochastic uncertainty, we need to accurately compute the mean and variance of these random variables. The robust optimization formulation considers the stochastic characteristics of the data, aiming to achieve both stability of the objective function and viable programs. The robust optimization formulation for the two players is as follows:

$$\begin{aligned}
& \max_{\pi_i, \mu_{R_1}, \sigma_{R_1}} (\omega \times \mu_{R_1} - (1 - \omega) \sigma_{R_1}) \\
& \text{s.t. } R_1 = \pi_1 E_{11} + \pi_2 E_{12} + \pi_3 E_{21} + \pi_4 E_{22}, \\
& \mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) \geq 0, \quad j = 1, 2, 3, 4, \\
& \sum_{j=1}^4 a_{ij} = 1, \quad i = 1, 2, 3, 4, \\
& \sum_{i=1}^4 \pi_i = 1, \\
& \pi_i \geq 0, \quad i = 1, 2, 3, 4. \tag{3}
\end{aligned}$$

$$\begin{aligned}
& \max_{\pi_i, \mu_{R_2}, \sigma_{R_2}} (\omega \times \mu_{R_2} - (1 - \omega) \sigma_{R_2}) \\
& \text{s.t. } R_2 = \pi_1 F_{11} + \pi_2 F_{12} + \pi_3 F_{21} + \pi_4 F_{22}, \\
& \mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) \geq 0, \quad j = 1, 2, 3, 4, \\
& \sum_{j=1}^4 a_{ij} = 1, \quad i = 1, 2, 3, 4, \\
& \sum_{i=1}^4 \pi_i = 1,
\end{aligned}$$

$$\pi_i \geq 0, i = 1, 2, 3, 4. \quad (4)$$

In this subsection, we analyze the mean $(\mu_{R_1}, \sigma_{R_1})$ and $(\mu_{R_2}, \sigma_{R_2})$, which represent the mean and variance of the strategic values for player 1 and player 2, separately. The symbols $\mu_g(\cdot)$ and $\sigma_g(\cdot)$ denote the mean and variance of the constraints g . The weight coefficient $\omega \geq 0$ indicates the relative importance of the two objectives, balancing the effects of expectation and variance. The constant k correlates with the degree of conservatism of the decision maker. The mean $\mu_g(\cdot)$ and the standard deviation $\sigma_g(\cdot)$ are approximated using the constraint functions in Equations (1) and (2). Constraints are introduced to ensure that the strategy remains feasible under all possible uncertainties. In the context of stochastic uncertainty, actual returns and state transition probabilities may deviate from their expected values (means), with the magnitude of these deviations measured by variance. A higher variance indicates greater uncertainty and larger fluctuations in returns or state transition probabilities. To adjust the original constraints for uncertainty, we introduce the mean-adjustment factor $\mu_g(\cdot)$, which represents the degree to which we expect the original constraints to be satisfied under uncertainty. The variance adjustment factor $k\sigma_g(\cdot)$ reflects our tolerance for uncertainty. The weighting factor k determines our sensitivity to fluctuations. Note that in a complete information game, the expected value does not consider the uncertainty related to the revenue and transfer probabilities. But in Equations (3) and (4), the average game values μ_{R_1} and μ_{R_2} account for parametric uncertainty, particularly the stochastic payoffs E_{ij} and F_{ij} . Similarly, the standard deviations σ_{R_1} and σ_{R_2} are computed with respect to these uncertainties. Additionally, the mean $\mu_g(\cdot)$ and standard deviation $\sigma_g(\cdot)$ are calculated relative to the random variable a_{ij} . In this study, the game's value, influenced by randomized returns and state transfer probabilities, is treated as a random variable and expressed through its mean and standard deviation. To implement Equations (3) and (4), precise knowledge of the mean and variance of the random terms is required. However, this may give rise to undecidability of the input data distribution parameters, especially under cognitive uncertainty, where data might be expressed by sparse point data or interval data. Robust optimization formulations considered the unpredictability of the data. Next, we discuss how to address the robust optimization methods for two-player repeated games under cognitive uncertainty caused by interval data.

3.3 Two-person Repeated Games under Cognitive Uncertainty

Considering our proposed two-player repeated game, where the payoff bimatrix and the state transfer probability matrix are represented by interval data, we apply a robust optimization approach to solve this game under cognitive uncertainty, with the uncertainty represented by interval data. Due to the input parameter uncertainties, the exact value of each element is unknown, so we define the variable values in terms of interval ranges. To address this problem, we use robust optimization techniques. By considering parameter uncertainties, we find a solution that performs well across the range of parameter variations. Firstly, the uncertainty range must be determined, which may involve statistical analysis or expert judgment. Then, the problem is

modeled as a robust optimization problem, with the objective of maximizing the gain over the uncertainty range of the parameters.

Adding cognitive uncertainty to an incomplete information game introduces additional complexity to the computational structure. When finding the mean and standard deviation of the game values, scalar values need to be determined in the corresponding intervals and then solving the optimization problem in Equations (3) and (4). This means that the existence of interval data necessitates evaluating every possible value in these equations to solve the optimization formulation. The selection of different scalar points can lead to varying anticipations of the mean and standard deviation of the game values for the same game. From a theoretical point of view, an infinite number of estimates can be derived from a given interval data. In practice, due to uncertainty, we aim to design a robust strategy that ensures good system performance even in the most unfavorable scenarios. By considering the worst-case scenario, we can ensure that the designed policy is optimal under all possible uncertainty conditions. Given that the decision maker cannot influence the cognitive variables, the robust optimization approach explores the range of possible values for these variables in order to identify the optimal solution. In this study, we begin by framing the issue as a nested optimization problem, The inner optimization aims to identify the worst-case scenario for a given strategy distribution π_i , considering payoffs and state-transfer probabilities within their intervals, to ensure that the objective function (a combination of expected payoffs and standard deviations) is optimized under all uncertainty scenarios. This process ensures that the strategy distribution remains optimal even in the worst case. The outer optimization then determines the optimal strategy distribution π_i that maximizes the objective function, taking into account the worst-case scenario identified by the inner optimization.

We develop a robust optimization model for two-player repeated games under cognitive uncertainty, described as the following nested optimization problem:

$$\begin{aligned}
& \max_{\pi_i, \mu_{R_1}, \sigma_{R_1}} \left(\min_{\mu_{\alpha_{ij}}, \mu_{E_{ij}}} (\omega \times \mu_{R_1} - (1 - \omega)\sigma_{R_1}) \right), \\
& \text{s.t. } R_1 = \pi_1 E_{11} + \pi_2 E_{12} + \pi_3 E_{21} + \pi_4 E_{22}, \\
& \mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) \geq 0, \quad i = 1, 2, 3, 4, \\
& lb_{ij} \leq a_{ij} \leq ub_{ij}, \\
& lb_{ij} \leq E_{ij} \leq ub_{ij}, \\
& \sum_{i=1}^4 \pi_i = 1, \\
& \pi_i \geq 0, \quad i = 1, 2, 3, 4, \\
& \max_{\pi_i, \mu_{R_2}, \sigma_{R_2}} \left(\min_{\mu_{\alpha_{ij}}, \mu_{F_{ij}}} (\omega \times \mu_{R_2} - (1 - \omega)\sigma_{R_2}) \right), \\
& \text{s.t. } R_2 = \pi_1 F_{11} + \pi_2 F_{12} + \pi_3 F_{21} + \pi_4 F_{22},
\end{aligned} \tag{5}$$

$$\begin{aligned}
\mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i - \pi_j \right) &\geq 0, \quad i = 1, 2, 3, 4, \\
lb_{ij} &\leq a_{ij} \leq ub_{ij}, \\
lb_{ij} &\leq F_{ij} \leq ub_{ij}, \\
\sum_{i=1}^4 \pi_i &= 1, \\
\pi_i &\geq 0, \quad i = 1, 2, 3, 4.
\end{aligned} \tag{6}$$

The implementation of Equations (5) and (6) necessitates knowledge of the range for the mean and standard deviation of the input variables. To compute the mean and variance of interval data, Zaman et al.^[14] proposed a specialized moment bounds algorithm, as shown in Table 2. Once the bounds for the mean and variance of the interval data are estimated, we apply the upper limits of the variance to address a robust two-player zero-sum game model in the context of interval uncertainty. Using the upper bound on variance is justified by its ability to make the solution less sensitive to fluctuations in the unknown input parameters.

Table 2 Moment boundary algorithm for interval data^[14]

Moment	Interval	
	Single interval	Multiple interval
	$M_1 = \sum_{i=1}^2 x_i p(x_i)$	
First moment	$\underline{M}$: PMF = 1 at lower end point	$[\underline{M}, \overline{M}] = \left[\frac{1}{n} \sum_{i=1}^n l_i, \frac{1}{n} \sum_{i=1}^n u_i \right]$
	= 0 elsewhere	
	$\overline{M}$: PMF = 1 at upper end point	
Secondary moment	= 0 elsewhere.	$\max / \min_{x_1, x_2, \dots, x_n} \frac{1}{n} \sum_{i=1}^n \left(x_i - \frac{1}{n} \sum_{j=1}^n x_j \right)^2$
	$M_2 = \sum_{i=1}^2 x_i^2 p(x_i) - \left(\sum_{i=1}^2 x_i p(x_i) \right)^2$	
	$\underline{M}$: PMF = 1 at any end point	
	= 0 elsewhere	
	$\overline{M}$: PMF = 0.5 at each end point.	s.t. $l_i \leq x_i \leq u_i, \quad i = 1, 2, \dots, n$

We describe the problem as a nested optimization problem with the following solution procedure:

Therefore, through Equations (5) and (6), we can generate intervals for players 1 and 2. Our method employs multi-objective optimization, which not only optimizes the mean of the game value under interval uncertainty but also minimizes its variance. Consequently, the final estimated interval of the game value depends on the weight parameters ω applied during the optimization process.

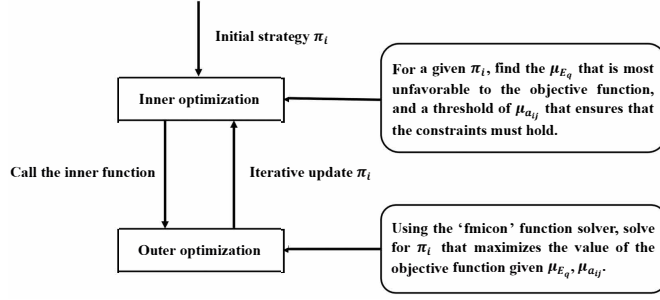


Figure 1 Solution process for nested optimization problems

Given that repeatedly performing robust optimization under cognitive uncertainty can lead to convergence issues and high computational costs, we propose a decoupling strategy to separate the robust optimization problem from the cognitive analysis. This separation aims to improve computational efficiency. We can decouple the nested formulas in Equations (5) and (6) as follows:

$$\begin{aligned}
 & \pi_i^*, \mu_{R_1}^*, \sigma_{R_1}^* = \arg \max_{\pi_i, \mu_{R_1}, \sigma_{R_1}} (\omega \times \mu_{R_1} - (1 - \omega) \sigma_{R_1}) \\
 \text{s.t. } & R_1 = \pi_1 E_{11}^* + \pi_2 E_{12}^* + \pi_3 E_{21}^* + \pi_4 E_{22}^*, \\
 & \mu_g \left(\sum_{i=1}^4 a_{ij}^* \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij}^* \pi_i - \pi_j \right) \geq 0, \quad i = 1, 2, 3, 4, \\
 & \sum_{i=1}^4 \pi_i = 1, \\
 & \pi_i \geq 0, \quad i = 1, 2, 3, 4, \\
 & \mu_{a_{ij}}^*, \mu_{E_{ij}}^* = \arg \min_{\mu_{a_{ij}}, \mu_{E_{ij}}} (\omega \times \mu_{R_1} - (1 - \omega) \sigma_{R_1}), \\
 \text{s.t. } & R_1 = \pi_1^* E_{11} + \pi_2^* E_{12} + \pi_3^* E_{21} + \pi_4^* E_{22}, \\
 & \mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i^* - \pi_j^* \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i^* - \pi_j^* \right) \geq 0, \quad i = 1, 2, 3, 4, \\
 & lb_{ij} \leq a_{ij} \leq ub_{ij}, \\
 & lb_{ij} \leq E_{ij} \leq ub_{ij}, \\
 & \sum_{i=1}^4 \pi_i^* = 1, \\
 & \pi_i^* \geq 0, \quad i = 1, 2, 3, 4, \\
 & \pi_i^*, \mu_{R_2}^*, \sigma_{R_2}^* = \arg \max_{\pi_i, \mu_{R_2}, \sigma_{R_2}} (\omega \times \mu_{R_2} - (1 - \omega) \sigma_{R_2}), \\
 \text{s.t. } & R_2 = \pi_1 F_{11}^* + \pi_2 F_{12}^* + \pi_3 F_{21}^* + \pi_4 F_{22}^*, \\
 & \mu_g \left(\sum_{i=1}^4 a_{ij}^* \pi_i - \pi_j \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij}^* \pi_i - \pi_j \right) \geq 0, \quad i = 1, 2, 3, 4,
 \end{aligned} \tag{7}$$

$$\begin{aligned}
& \sum_{i=1}^4 \pi_i = 1, \\
& \pi_i \geq 0, \quad i = 1, 2, 3, 4, \\
& \mu_{a_{ij}}^*, \mu_{E_{ij}}^* = \arg \min_{\mu_{a_{ij}}, \mu_{E_{ij}}} (\omega \times \mu_{R_2} - (1 - \omega) \sigma_{R_2}), \\
& \text{s.t. } R_2 = \pi_1^* F_{11} + \pi_2^* F_{12} + \pi_3^* F_{21} + \pi_4^* F_{22}, \\
& \mu_g \left(\sum_{i=1}^4 a_{ij} \pi_i^* - \pi_j^* \right) + k \sigma_g \left(\sum_{i=1}^4 a_{ij} \pi_i^* - \pi_j^* \right) \geq 0, \quad i = 1, 2, 3, 4, \\
& lb_{ij} \leq a_{ij} \leq ub_{ij}, \\
& lb_{ij} \leq E_{ij} \leq ub_{ij}, \\
& \sum_{i=1}^4 \pi_i^* = 1, \\
& \pi_i^* \geq 0, \quad i = 1, 2, 3, 4.
\end{aligned} \tag{8}$$

4 Numerical Examples

In this subsection, three numerical examples are used to illustrate the performance of two-player repeated games under different types of uncertainty: data certainty, stochastic uncertainty, and cognitive uncertainty. The first example is a classical two-player repeated prisoner's dilemma with complete information. Here, the final strategy choices and expected payoffs of the players are obtained by solving a set of linear programming problems. The second example extends the two-player repeated prisoner's dilemma by introducing stochastic uncertainty, where the input data are influenced by random variables due to external factors. The third example involves a company's product development and marketing problem, where differing opinions lead to the evaluation of multiple data intervals for the marketing team's revenues and the shifting market state.

4.1 Deterministic Case

Suppose there are two players involved in a repeated Prisoner's Dilemma, where player 1 and player 2 each has two strategies, and for different combinations of strategies, players have different payoffs. We assume that the players' payoffs for each round satisfy a double payoff matrix:

$$\begin{pmatrix} (3, 2) & (3, 1) \\ (4, 2) & (2, 1) \end{pmatrix}.$$

At the end of a round of the game player 1 and player 2 makes a strategic choice for the next round based on the state he is currently in, can be expressed as:

	e_1	e_2
O_1	p_1	$1 - p_1$
O_2	p_2	$1 - p_2$
O_3	p_3	$1 - p_3$
O_4	p_4	$1 - p_4$

	f_1	f_2
O_1	q_1	$1 - q_1$
O_2	q_2	$1 - q_2$
O_3	q_3	$1 - q_3$
O_4	q_4	$1 - q_4$

Here we take $p_1 = 0.5$, $p_2 = 0.6$, $p_3 = 0.7$, $p_4 = 0.8$; $q_1 = 0.4$, $q_2 = 0.5$, $q_3 = 0.6$, $q_4 = 0.7$, the corresponding state transfer probability matrix is:

$$P = \begin{pmatrix} 0.2 & 0.3 & 0.3 & 0.2 \\ 0.3 & 0.3 & 0.2 & 0.2 \\ 0.42 & 0.28 & 0.18 & 0.12 \\ 0.56 & 0.24 & 0.14 & 0.0 \end{pmatrix}.$$

In this case, we use Eqs. (1), (2) to solve it, which is a typical linear programming problem, and get the result, as shown in Table 3:

Table 3 Results under deterministic data

Stationary distribution π	0.3345; 0.2860; 0.2195; 0.1600
Long-term average reward for player1	2.4874
Long-term average reward for player2	2.3866

The results show that the two players have their respective optimal payoffs when they are in state O_1, O_2, O_3, O_4 with probabilities of (0.3345; 0.2860; 0.2195; 0.1600), which are 2.4874 and 2.3866, respectively.

4.2 Stochastic Uncertainty Case

In game theory, particularly in repeated game scenarios, players' payoffs and state transition probabilities can be influenced by uncertainties. For instance, the exact probability of an opponent's strategic choices or changes in environmental variables may be unknown. To address this uncertainty, robust optimization methods are employed. In this context, we treat the players' payoff matrices and state transition probabilities as random variables, assuming that the payoffs for both players follow a normal distribution $E, F \sim N(\mu, \sigma^2)$. Consequently, the overall state transition matrices of the repeated game system are also random. In practical scenarios, other probability distributions, such as the uniform distribution, might be observed based on different empirical results. Here, we illustrate this with a numerical example: After an investigation, the

benefits for player 1 and player 2 when choosing different strategies i, j are presented in Table 4 as below:

Table 4 The payment matrix of two players

player 1	player 2	
	e_1	e_2
f_1	$((10, 1), (8, 1.2))$	$((15, 1.5), (12, 1.7))$
f_2	$((20, 2), (18, 2.2))$	$((25, 2.5), (22, 2.7))$

The payoffs follow a normal distribution, and the transfer probabilities a_{ij} are characterized by $\mu_{a_{ij}} = 0.5$, $\sigma_{a_{ij}} = 0.1$. In our robust optimization model, we aim to optimize both the mean of the expected return and the variance of the minimum return. We use a weight coefficient of 0.5 to balance the return expectation and risk, and a risk preference parameter of 1.96 to represent the player's sensitivity to uncertainty. According to the stochastic uncertainty Equations (3) and (4) presented in Section 3, the optimization problem is solved using MATLAB's 'fmincon' function, which iteratively determines the optimal solution based on the initial guess and optimization options. During the optimization process, we may face issues with convergence to a feasible point due to the inherent randomness of input variables, which can lead to results that are locally optimal rather than globally optimal. To address this, we can refine initial conditions, adjust parameters, set random seeds, or increase the number of iterations to enhance the prediction of optimization results and better analyze strategy selection.

The results are shown in Table 5, with both players focusing on choosing the last strategy:

Table 5 Results under randomized uncertainty data

	Optimal pi					Optimal objective value
Player 1	0.0401	0.2458	0.1233	0.6597		10.8496
Player 2	0.0257	0.2097	0.1115	0.6532		8.8829

We performed several parameter adjustments during the optimization process and conducted 12 sets of tests with about 10 iterations per set, which resulted in a relatively fast optimization.

In a robust optimization model, ω is a weighting parameter used to balance the expected return and risk in the objective function, thus affecting the final strategy and outcome. The change in expectation and risk is analyzed for different values of the weights, as shown in Figure 2, Figure 3:

Obviously, as ω approaches 1, the objective function increasingly prioritizes expected returns over risk. Consequently, the optimization process will favor strategies with higher expected returns, even if they involve greater risk. Conversely, as ω approaches 0, the objective function shifts its focus towards risk avoidance, placing less emphasis on expected returns. This leads the optimization process to favor strategies with lower risk, even if they yield lower expected returns. When ω is approximately 0.5, the objective function maintains a balanced emphasis on both

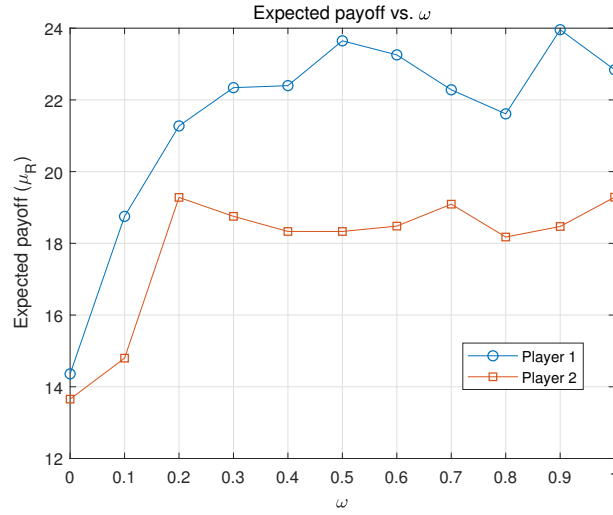


Figure 2 Changes in expectations with different weights

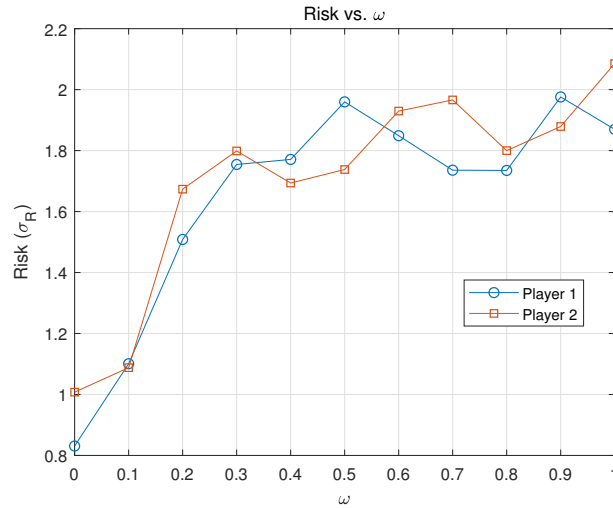


Figure 3 Changes in risk with different weights

expected return and risk, resulting in strategies that represent a compromise between the two. By varying ω , one can observe how changes in expected return and risk are affected. Typically, an increase in ω leads to higher expected returns and greater risk, while a decrease in ω results in lower expected returns and reduced risk. These observations align with the expectations set by the robust optimization model.

Next adjustments k , observing changes in players' strategies under different risk preferences:

As shown in Figures 4 and 5, it is evident that players consistently favor the last strategy, regardless of the value of k . The proportion of π_4 is consistently the largest, which is consistent with our numerical case. This indicates that players prefer state O_4 , which offers both high payoffs and high risk, it is optimal for player 1 and player 2 to choose (e_2, f_2) .

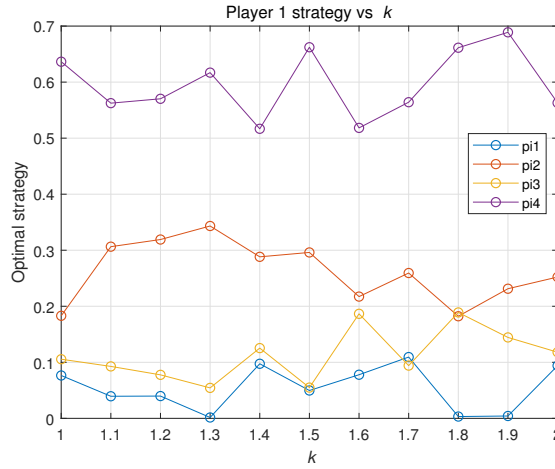


Figure 4 Changes in player 1's strategy under different risks

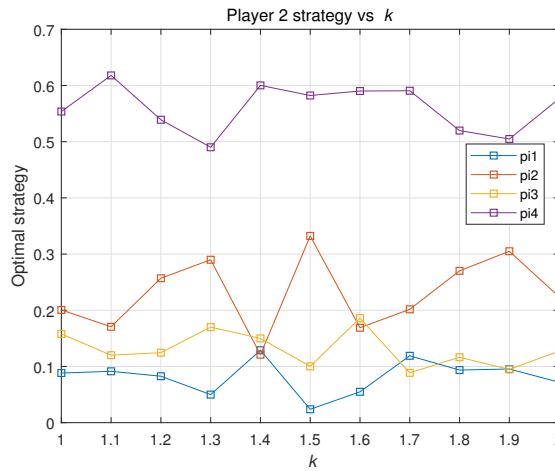


Figure 5 Changes in player 2's strategy under different risks

In summary, these graphs illustrate how the risk preference parameter k impacts a player's expected return and risk within a robust optimization model. This understanding is crucial for developing rational strategies that effectively balance returns and risks.

4.3 Cognitive Uncertainty Case

A company is developing a new app, with two teams: the product development team (player 1) and the marketing team (player 2), collaborating to ensure a successful launch. However, the market environment and internal decisions are fraught with uncertainties. Each team must make critical decisions every quarter, with their benefits influenced by market demand, competitor behavior, and various other factors. These decisions impact not only the current quarter's revenue but also the market state in subsequent quarters. The revenue matrices $\mathbf{E}$ for the product development team and $\mathbf{F}$ for the marketing team represent their potential returns under different strategy combinations. The state shift probability matrix $\mathbf{A}$ shows the likelihood of

market state transitions under these strategies. The diverse opinions within the teams result in evaluating our matrices with multiple data intervals, as shown below. We aim to find the optimal outcome using Equations (5) and (6).

Player 1's payoff matrix $\mathbf{E}$:

$$\mathbf{E} = \begin{bmatrix} \{[1, 2], [2, 3]\} & \{[2, 3], [3, 4]\} \\ \{[3, 4], [4, 5]\} & \{[4, 5], [5, 6]\} \end{bmatrix}.$$

Player 2's payoff matrix $\mathbf{F}$:

$$\mathbf{F} = \begin{bmatrix} \{[1, 3], [2, 4]\} & \{[2, 4], [3, 5]\} \\ \{[3, 5], [4, 6]\} & \{[4, 6], [5, 7]\} \end{bmatrix}.$$

The state transfer probability matrix $\mathbf{A}$:

$$\mathbf{A} = \begin{bmatrix} \{[0.1, 0.3], [0.2, 0.4]\} & \{[0.2, 0.4], [0.3, 0.5]\} & \{[0.3, 0.5], [0.4, 0.6]\} & \{[0.4, 0.6], [0.5, 0.7]\} \\ \{[0.2, 0.4], [0.3, 0.5]\} & \{[0.1, 0.3], [0.2, 0.4]\} & \{[0.4, 0.6], [0.5, 0.7]\} & \{[0.3, 0.5], [0.4, 0.6]\} \\ \{[0.3, 0.5], [0.4, 0.6]\} & \{[0.4, 0.6], [0.5, 0.7]\} & \{[0.1, 0.3], [0.2, 0.4]\} & \{[0.2, 0.4], [0.3, 0.5]\} \\ \{[0.4, 0.6], [0.5, 0.7]\} & \{[0.3, 0.5], [0.4, 0.6]\} & \{[0.2, 0.4], [0.3, 0.5]\} & \{[0.1, 0.3], [0.2, 0.4]\} \end{bmatrix}.$$

Results are shown in Table 6:

Table 6 Results under randomized uncertainty data	
Optimal strategy pi	0, 0.5, 0, 0.5
Optimal objective function value of 1	0.625
Optimal objective function value of 2	0.375

In this case, before performing robust optimization according to Equations (5) and (6), it is necessary to calculate the mean and variance of the multi-interval data involved. This process utilizes the methodology proposed by Zaman et al., i.e:

$$\bar{E} = \left[\frac{1}{n} \sum_{i=1}^n lb_{ij}, \frac{1}{n} \sum_{i=1}^n ub_{ij} \right],$$

$$\text{Var}(E) = \max_{x_1, x_2, \dots, x_n} \left\{ \frac{1}{n} \sum_{i=1}^n \left(x_i - \frac{1}{n} \sum_{j=1}^n x_j \right)^2 \right\} \Big| lb_{ij} \leq x_i \leq ub_{ij}.$$

where n is the number of elements. From the optimization results, it is evident that the optimal value for player 1's objective function is achieved at 0.625, with player 2's objective function peaking at 0.375 when the chosen strategy is (0, 0.5, 0, 0.5). The methodology of the solution, as delineated in Section 3.3, entails addressing the inner-layer optimization problem to identify the most adverse scenario for the objective function, given the initial strategy. Subsequently, the strategy is revised based on the results of the inner optimization, and the outer optimization is reinitiated to enhance the objective function. This iterative approach is employed to pinpoint

the scalar value that meets the criteria of the objective function, persisting until the convergence is attained.

Different values of ω have a significant impact on the player's expectation μ_{R_1}, μ_{R_2} because ω reflects the player's trade-off between benefit and risk. Specifically, the value of ω determines the weights of return and risk in the objective function, which affects the choice of the optimal strategy π_i , and ultimately μ_{R_1}, μ_{R_2} . In order to analyze the effect of different ω on the player's expectation (the variance is uniformly taken as the upper bound), we can set different values of ω , run the optimization algorithm, and observe the corresponding μ_{R_1}, μ_{R_2} values for different values of ω :

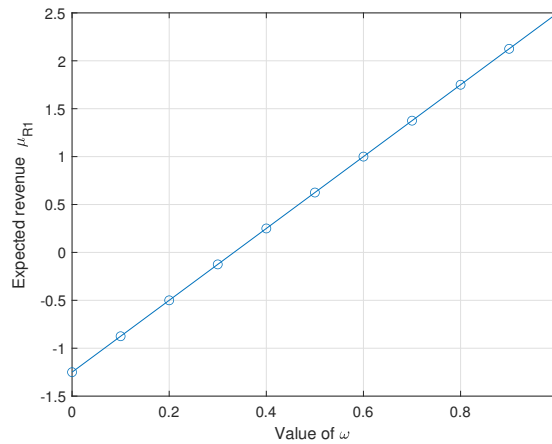


Figure 6 Expectations of player 1 with different weights

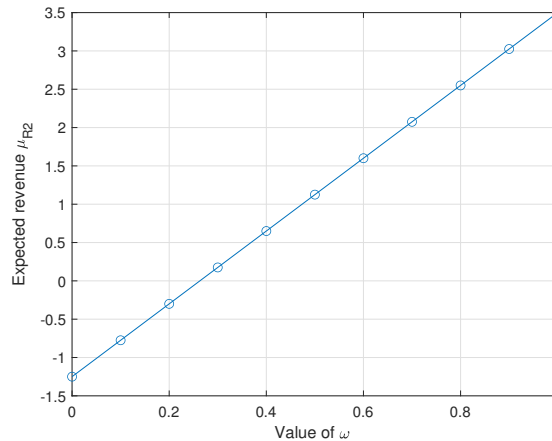


Figure 7 Expectations of player 2 with different weights

Obviously, as ω increases, the dependence of the objective function on the player's expectation increases, which leads to the optimal μ_{R_1}, μ_{R_2} increasing linearly with ω .

The example illustrates that uncertainty is a common problem in business decisions, such as fluctuations in market demand, competitor behavior, etc. In the actual decision-making process,

interval data often appear due to incomplete information or differences in evaluation methods. Handling it with the help of robust optimization methods is a good choice to provide a more robust basis for decision making. In the subsequent research, for such practical problems, we should cell phone relevant data as much as possible and conduct detailed analysis before making decisions to reduce uncertainty. Clarify the risk preferences and set the weighting parameters in a joint effort to balance the risk and return. Decision support systems can also be developed or used to help decision makers analyze data, build models and optimize more effectively.

5 Conclusion and Recommendation

In this paper, we discuss two-person repeated games under both complete and incomplete information. We construct a linear programming formulation for the certain case using a Markov decision process. For uncertain data under incomplete information, we employ a robust optimization approach that accounts for parameter uncertainties, seeking an optimal solution within these uncertainties. Robustness is defined as optimizing the mean and minimizing the variance of the objective function. We develop robust optimization formulations under both stochastic and cognitive uncertainty, addressing scenarios where uncertain data are represented as either random variables or interval data. Our proposed method is more general than previous approaches, capable of handling various types of data, including deterministic (point-valued), selective (described by exact probability distributions), and cognitive (represented by single or multiple interval data). Additionally, given the uncertainty of input variables, we compute the mean and standard deviation of the game values to facilitate risk-averse decision-making when selecting strategies. In our numerical experiments, we observe that the solution of the model requires continuous iteration from an initial value, suggesting potential for significant improvements in algorithm selection and solution speed. Future work can explore a wider range of game types and address the combined cases of point-valued, random-valued, and interval-valued data simultaneously.

Acknowledgements The authors gratefully acknowledge the editors and anonymous referees for their insightful comments and helpful suggestions that led to a marked improvement of the article.

References

- [1] Gao J. Uncertain bimatrix game with applications. *Fuzzy Optimization and Decision Making*, 2013, 12: 65–78.
- [2] Li Y, Yang Z. Games with incomplete information and uncertain payoff: From the perspective of uncertainty theory. *Soft Computing*, 2019, 23: 13669–13678.
- [3] Zaman K, McDonald M, Mahadevan S, et al. Robustness-based design optimization under data uncertainty. *Structural and Multidisciplinary Optimization*, 2011, 44: 183–197.
- [4] Harsanyi J. Games with incomplete information played by “Bayesian” players, I–III Part I. The basic model. *Management Science*, 1967, 14(3): 159–182.

- [5] Cheng J, Leung J, Lisser A. Random-payoff two-person zero-sum game with joint chance constraints. *European Journal of Operational Research*, 2016, 252(1): 213–219.
- [6] Su Q, Ji L. Research on coordination mechanism of medical data sharing based on stochastic evolutionary game. *Information Sciences*, 2023, 41(9): 37–47.
- [7] Chavoshlou A, Khamseh A, Naderi B. An optimization model of three-player payoff based on fuzzy game theory in green supply chain. *Computers & Industrial Engineering*, 2019, 128: 782–794.
- [8] Sangeetha K, Parimala M. On solving a fuzzy game problem using hexagonal fuzzy numbers. *Materials Today: Proceedings*, 2021, 47: 2102–2106.
- [9] Li S. Research on environmental behavior based on game theory in a fuzzy environment. Nanchang: Nanchang University, 2023.
- [10] Dwayne C W, Hu C. Studying interval valued matrix games with fuzzy logic. *Soft Computing*, 2008, 12(2): 147–155.
- [11] Yu H, Liu H. Robust multiple objective game theory. *Journal of Optimization Theory and Applications*, 2013, 159: 272–280.
- [12] Qu S, Meng D, Zhou Y, et al. Distributionally robust games with an application to supply chain. *Journal of Intelligent & Fuzzy Systems*, 2017, 33(5): 2749–2762.
- [13] Dey A, Zaman K. A robust optimization approach for solving two-person games under interval uncertainty. *Computers & Operations Research*, 2020, 119: 104937.
- [14] Zaman K, Rangavajhala S, McDonald M P, et al. A probabilistic approach for representation of interval uncertainty. *Reliability Engineering & System Safety*, 2011, 96(1): 117–130.
- [15] Maggioni F, Dabbene F, Pflug G C. Sampling methods for multi-stage robust optimization problems. *Annals of Operations Research*, 2025, 347(3): 1385–1423.
- [16] Beck Y, Ljubić I, Schmidt M. A survey on bilevel optimization under uncertainty. *European Journal of Operational Research*, 2023, 311(2): 401–426.
- [17] Li G, Chow A H F, Ying C. Robust optimization for adaptive bus service scheduling with adversarial reinforcement learning under demand uncertainties. *Transportation Research Part C: Emerging Technologies*, 2025, 178: 105222.
- [18] Rapoport A. Optimal policies for the Prisoner’s Dilemma. *Psychological Review*, 1967, 74(2): 136–148.
- [19] Ueda M. Memory-two strategies forming symmetric mutual reinforcement learning equilibrium in repeated prisoners’ dilemma game. *Applied Mathematics and Computation*, 2023, 444: 127819.
- [20] Gensbittel F, Oliu-Barton M. Optimal strategies in zero-sum repeated games with incomplete information: The dependent case. *Dynamic Games and Applications*, 2020, 10(4): 1–17.
- [21] Oliu-Barton M. Asymptotically optimal strategies in repeated games with incomplete information and vanishing weights. *Journal of Dynamics & Games*, 2019, 6(4): 259–275.
- [22] Bressaud X, Quas A. Dynamical analysis of a repeated game with incomplete information. *Mathematics of Operations Research*, 2017, 42(4): 1085–1105.
- [23] Liu A, Liu K. Modeling of scheduling problems based on Markov decision processes[C]. *Proceedings of the 9th Annual Conference on Uncertain Systems, 5th Annual Conference on Intelligent Computing, and 13th Annual Conference on Young Scholars in Information and Management*, 2011: 321–330.