Enhanced Equilibria-Solving via Private Information Pre-Branch Structure in Adversarial Team Games

Chen Qiu[1]   **Haobo Fu**[2]   **Kai Li**[3,4]   **Jiajia Zhang**[1]   **Xuan Wang**[*1]

[1]School of Computer Science and Technology, Harbin Institute of Technology Shenzhen, China
[2]Tencent AI Lab, China
[3]Institute of Automation, Chinese Academy of Sciences, China
[4]School of Artificial Intelligence, University of Chinese Academy of Sciences, China

Abstract

In *ex ante* coordinated adversarial team games (ATGs), a team competes against an adversary, and team members can only coordinate their strategies before the game starts. The team-maxmin equilibrium with correlation (TMECor) is a suitable solution concept for extensive-form sequential ATGs. One class of TMECor-solving methods transforms the problem into solving NE in two-player zero-sum games, leveraging well-established tools for the latter. However, existing methods are fundamentally action-based, resulting in poor generalizability and low solving efficiency due to the exponential growth in the size of the transformed game. To address the above issues, we propose an efficient game transformation method based on private information, where all team members are represented by a single coordinator. We designed a structure called *private information pre-branch*, which makes decisions considering all possible private information from teammates. We prove that the size of the game transformed by our method is exponentially reduced compared to the current state-of-the-art. Moreover, we demonstrate equilibria equivalence. Experimentally, our method achieves a significant speedup of $182.89\times$ to $694.44\times$ in scenarios where the current state-of-the-art method can work, such as small-scale *Kuhn poker* and *Leduc poker*. Furthermore, our method is applicable to larger games and those with dynamically changing private information, such as *Goofspiel*.

1 INTRODUCTION

Games have long served as critical testbeds for exploring how effectively machines can make sophisticated decisions

---

*Corresponding author.

since the early days of computing [Bard et al., 2020, Campbell et al., 2002, Silver et al., 2017]. Finding equilibrium in games has become a significant criterion for evaluating the level of artificial intelligence. In the real world, there have been systems that have achieved superhuman performance, such as *AlphaGo* [Silver et al., 2016], *Libratus* [Brown and Sandholm, 2018], and *DeepStack* [Moravčík et al., 2017]. While many advances [Brown et al., 2019, Zhou et al., 2020, Asmus et al., 2017] have been made in 2-player zero-sum (2p0s) games based on Nash equilibrium (NE) [Nash, 1951] in imperfect information environments, recent research has focused on more complex adversarial team games (ATGs). In ATGs, multiple players with the same utility function form a team to compete against a common adversary [von Stengel and Koller, 1997]. This results in a game where both cooperation and competition coexist.

In this paper, we focus on the ATGs with *ex ante* coordination. More specifically, team members are allowed to coordinate and agree on a common strategy before the game starts. The solution concept for this setting is the team-maxmin equilibrium with correlation (TMECor), which can be thought of as an NE between the team and the adversary in an ATG [Zhang et al., 2021]. TMECor has properties similar to NE, such as exchangeability. However, finding a TMECor is proven to be FNP-hard [Hansen et al., 2008].

Methods for computing TMECor can be roughly divided into four categories. The first involves using linear programming (LP). Hybrid column generation was the first algorithm to compute TMECor in ATGs [Celli and Gatti, 2018]. Its core involves team members adopting joint normal-form strategies, while the adversary uses sequence-form strategies. However, it requires solving an integer LP and suffers from exponential growth in the number of joint actions as the game size increases, making it impractical for large-scale games. The second category involves multi-agent deep reinforcement learning algorithms, such as SIMS [Cacciamani et al., 2021], which learns coordinated strategies from experience. It is limited to games with symmetric observations and requires perfect recall refinement, which is not

feasible when team members have private information. The third approach, based on the team belief directed acyclic graph (TBDAG) [Zhang et al., 2023], captures joint beliefs of team members and constructs a DAG for team strategies, providing a more compact representation of team's strategy space. Unfortunately, applying LP directly to the TB-DAG is inefficient, and subsequent improvements [Zhang et al., 2024, 2022a] suffer from limited interpretability. The last category of methods establishes a connection between ATGs and 2p0s games through game tree transformation. While CFR [Zinkevich et al., 2007] and its variants provide convergence guarantees for finding NE in 2p0s games, no such guarantees exist for multiplayer games [Brown and Sandholm, 2019]. We can leverage these established algorithms by transforming ATGs into equivalent 2p0s games. In this paper, we focus on the game tree transformation-based approach.

The state-of-the-art game tree transformation-based method, TPICA [Carminati et al., 2022], involves the concept of *extensive-form game with visibility*. By introducing the coordinator into an ATG, they convert the task of finding a TMECor into finding an NE in a 2p0s game. However, TPICA suffers from low solving efficiency and limited types of solvable games due to its reliance on action-based transformation. In this method, the coordinator extracts an action from each distinguishable state in the original game to form different recommendations for the team players. Then, a specific action is designated from each recommendation as an available action for the coordinator. To analyze the game size complexity, we consider a setting where the opponent plays first, followed by team members in sequence. We assume that every player has the same number of available actions in every state. Since modifications are made only to team player nodes, we refer to the phase from any specified team player node, which acts first, to all possible next opponent (or terminal) nodes as an episode. For any episode, the size of the transformed game tree is $\mathcal{O}\big((|A|^{|\Omega|})^{|\mathcal{T}|}\big)$, where $|A|$, $|\Omega|$, $|\mathcal{T}|$ denote the number of available actions, the amount of private information, and the number of team players, respectively. Therefore, the size of the transformed game tree grows significantly with the increase in the number of available actions, team players, and private information. In particular, the size growth triggered by a single coordinator node is exponential. Additionally, TPICA cannot be applied to games where players' private information changes.

To address the above issues, we propose a multi-player transformation algorithm (MPTA) based on private information. To mitigate the exponential growth caused by a single coordinator, we designed a new structure called *private information pre-branch* (PIPB), consisting of coordinator and dummy player nodes. Specifically, PIPB allows dummy players to provide the coordinator with all possible private information from teammates. Since the amount of potentially private information in an ATG is fixed, this structure

significantly reduces the size of the transformed game tree compared to the previous state-of-the-art method. This leads to a substantial improvement in the efficiency of equilibrium computation. Furthermore, we demonstrate the equilibrium equivalence before and after the transformation. The private information-based transformation makes our method suitable for games with dynamically changing private information (e.g., *Goofspiel*), expanding the types of solvable games. We show the superior performance of our method through extensive experiments in different game scenarios. Experimental results show that our method computes strategies closer to TMECor compared to the baseline algorithm in the same runtime and significantly reduces runtime within the same number of iterations.

Our contributions can be summarized as follows:

- We proposed MPTA based on private information, which significantly improves equilibria-solving efficiency. For any episode, compared to the previous state-of-the-art, the size growth is reduced. Specifically, our method decreases from $\mathcal{O}\big((|A|^{|\Omega|})^{|\mathcal{T}|}\big)$ to $\mathcal{O}\big(\big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A|\big)^{|\mathcal{T}|}\big)$, where $\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}$ represents the number of ways to arrange $|\mathcal{T}|-1$ elements from a set of $|\Omega|-1$ private information. Additionally, we demonstrated the equilibria equivalence between TMECor in the original game and NE in the transformed game.

- The PIPB structure enhances the generalization capability of our method, allowing it to adapt to a wider variety of games. In particular, it enables our method to effectively address ATGs where the private information available to each player can change dynamically throughout the game.

- We conducted experiments on 14 different configurations across three standard testbeds. The results show a significant improvement in solving efficiency using our method, achieving speedups ranging from $182.89\times$ to $694.44\times$ compared to the baseline. We also compared the sizes of the transformed game trees, showing that our method results in much smaller game trees than the baseline. Furthermore, we experimented with larger-scale games and other types of games that were not supported by the baseline.

All proofs in this paper can be found in Appendix A.

2 PRELIMINARIES

2.1 EXTENSIVE-FORM GAMES AND NASH EQUILIBRIUM

An extensive-form game (EFG) is the tree-form model of imperfect-information games with sequential interactions [Kuhn, 1950a, Brown and Sandholm, 2017, Liu et al., 2021]. The set $A = \cup_{i \in N} A_i$ denotes all the possible actions, where

A_i represents a set of available actions of player i. $|A|$ is the number of each player's available actions. H is the set of nodes, and $h \in H$ represents the sequence of all actions from the root to node h. $ha \sqsubseteq h'$ denotes h reaches h' by playing an action a. The set $Z \subseteq H$ contains all the terminal nodes. For each decision node $h \in H$, the result returned by the function $\mathcal{A}(h)$ is all available actions at node h. $\omega_i \in \Omega$ denotes player i's private information (e.g., a card in a poker game), and $|\Omega|$ represents the amount of private information in a game. The player who takes an action at node h is returned by function $P(h)$. The utility function $u_i(z)$ is the player i's payoff mapped from a terminal node $z \in Z$ to the real $\mathbb{R}$. An information set (infoset) I_i represents imperfect information for player i, which means all nodes h, h' are indistinguishable to i in I_i. The set of infosets for player i is denoted by $\mathcal{I}_i$, and the set of all infosets is represented as $\mathcal{I} = \cup_{i \in N} \mathcal{I}_i$.

There are two fundamental paradigms for strategy representation [Carminati et al., 2022, Ma et al., 2018]. A behavioral strategy σ_i of player $i \in N$ is a function that assigns a distribution over all the available actions $\mathcal{A}(I_i)$ to each I_i. Another strategy representation is based on the normal-form plan (also referred to as the pure strategy), $\pi_i = \times_{I \in \mathcal{I}_i} \mathcal{A}(I)$, which is a tuple that specifies one action for each of player i's infosets. A normal-form strategy is the probability distribution of normal-form plans for a player. A reduced normal-form strategy (a.k.a. mixed strategy) $\mu_i \in \Delta(\Pi_i)$ is obtained from a normal-form strategy by consolidating plans that are differentiated via actions taken in unreachable nodes. Henceforth, we focus on reduced normal-form strategies in this paper. For any player $i \in N$, $\mu_i[z]$ (or $\sigma_i[z]$) denotes the probability of reaching terminal nodes $z \in Z$ when i follows strategy μ_i (or σ_i). We represent behavioral strategy profiles with $\boldsymbol{\sigma}$ and normal-form strategy profiles with $\boldsymbol{\mu}$. We define $\boldsymbol{\sigma}_{-i}$ as strategies of players except for i. The expected payoff for player i when he plays strategy σ_i and all the other players follow strategy $\boldsymbol{\sigma}_{-i}$ is denoted by $u_i(\sigma_i, \boldsymbol{\sigma}_{-i})$. Player i's *best response* to strategy σ_{-i}, denoted as $BR_i(\boldsymbol{\sigma}_{-i})$, is a single strategy that maximizes player i's payoff against strategy $\boldsymbol{\sigma}_{-i}$. Formally, $u_i(BR_i(\boldsymbol{\sigma}_{-i}), \boldsymbol{\sigma}_{-i}) = max_{\sigma_i'} u_i(\sigma_i', \boldsymbol{\sigma}_{-i})$. NE is a significant solution concept in 2p0s games in which no player can unilaterally change his strategy to obtain more payoff. It captures the idea of stability under rational behavior, where no player has an incentive to deviate the current strategy. An NE $\boldsymbol{\sigma}$ is a strategy profile where all players play the *best response*. Formally, $\boldsymbol{\sigma}$ is an NE if and only if $\forall i \in N, \sigma_p \in BR_i(\boldsymbol{\sigma}_{-i})$. The *exploitability* $e(\sigma_i)$ of strategy σ_i serves as our measurement metric, which measures how much worse σ_i does versus $BR_{-i}(\sigma_i)$ (i.e., the *best response* of all other players to σ_i) compared to how an equilibrium strategy σ_i^* does against $BR_{-i}(\sigma_i^*)$. This metric provides a quantitative measure of deviation from equilibrium and is widely used in evaluating strategy quality in extensive-form games.

2.2 ADVERSARIAL TEAM GAMES AND TEAM-MAXMIN EQUILIBRIUM WITH CORRELATION

An ATG is an EFG with a set of players N, where a team of players competes against an opponent. That is, $N = \mathcal{T} \cup \{o\} \cup \{c\}$, where $\mathcal{T}$ represents a team, and o is an opponent. The chance player c simulates exogenous randomness in the game, such as dealing a card from a deck. $|\mathcal{T}|$ denotes the number of team members. The team players share payoffs in ATGs. Formally, $\forall i, j \in \mathcal{T}, u_i(z) = u_j(z)$. Following the convention of the relevant literature [Celli and Gatti, 2018, Zhang et al., 2021, Carminati et al., 2022], we assume *perfect recall*, which means each player remembers information acquired in earlier stages of each infoset.

In this work, we focus on the *ex ante* coordinated setting, where TMECor is a significant solution concept. Specifically, a TMECor is an NE that maximizes the team's payoff when team players are allowed to correlate their strategies and agree on tactics before the game begins. A TMECor can be found via a bi-level optimization program formulated over the normal-form strategy profile of team members:

$$
\begin{aligned}
\max_{\mu_\mathcal{T}} \min_{\mu_o} & \sum_{z \in Z} \mu_\mathcal{T}[z] \mu_o[z] u_\mathcal{T}(z) \\
\text{s.t.} \quad & \mu_\mathcal{T} \in \Delta(\times_{i \in \mathcal{T}} \Pi_i) \\
& \mu_o \in \Delta(\Pi_o)
\end{aligned}
\tag{1}
$$

2.3 TEAM-PUBLIC-INFORMATION REPRESENTATION FOR EXTENSIVE-FORM GAMES

Since this subsection involves some additional concepts, we provide a detailed example in Appendix B.2 for a clearer explanation. An action a is classified as *observable* or *unobservable* depending on whether it can be seen by player i when played by another player. If the actions observable by player i at any pair of nodes are the same, these nodes belong to the same infoset. When all infosets are induced, as discussed above, the game is an extensive-form game with visibility (vEFG), where every player has perfect recall. Extending action visibility to a set of player $\mathcal{P}$ (e.g., a team), an action a is called *public* if it is observable by all players in $\mathcal{P}$; *private* if it can be observed by only some player(s) in $\mathcal{P}$; and *hidden* if it is not observable by all players in $\mathcal{P}$ (in this case, a is played by a player not belonging to $\mathcal{P}$). A public infoset for a set of players $\mathcal{P}$ is defined as a public state $S_\mathcal{P} \subset H$, where any two nodes of potentially different players in $\mathcal{P}$ belong to the same public state if the actions that are *public* for $\mathcal{P}$ are the same at these nodes. Clearly, if one node of an infoset I belongs to $S_\mathcal{P}$, then all the other nodes of I also belong to $S_\mathcal{P}$. Let $\mathcal{S}$ denote the set of all public states. $\mathcal{S}_\mathcal{P}(h)$ represents the set of all infosets of players in $\mathcal{P}$ that are in the same public state at node h.
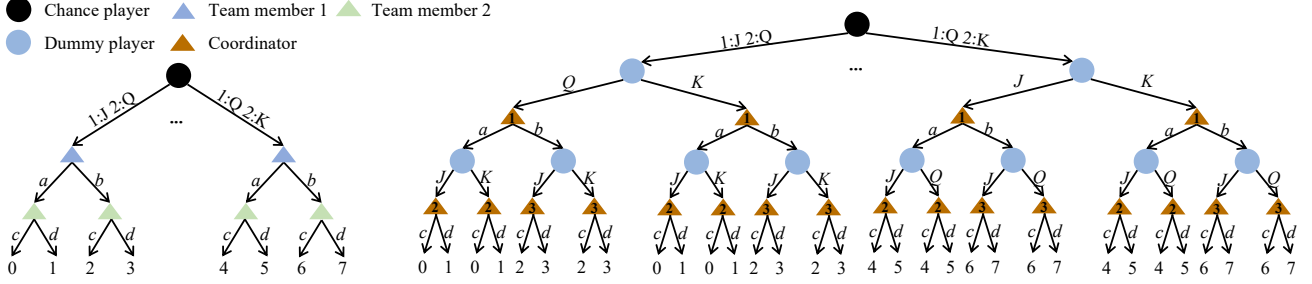
Figure 1: Example of game transformation. "..." indicates omitted branches. The nodes of a player with the same number are in the same infoset. **Left:** Original ATG omitting the opponent nodes. **Right:** Result of transforming the game on the left using MPTA.

Similarly, this paper focuses on public-turn-taking games, where every player knows, at every infoset he plays, the sequence of actions taken by players from the root to that infoset. This indicates that the public states have a specific structure and consist of nodes with histories of the same length for a single player. Carminati et al. [2022] proved that any vEFG with perfect recall and timeability has a strategy-equivalent public-turn-taking vEFG. Completely inflated games refer to situations where every team player knows the exact action taken by another team player at any infoset. This can be achieved for a generic vEFG by modifying the visibility of team players' actions, allowing explicit representation of strategy sharing among teammates before the game starts. In the following, we focus on completely inflated vEFGs for the team.

In the game tree transformed by TPICA, every coordinator who represents the team $\mathcal{T}$ plays a *prescription* among all the combinations of possible actions for each information state I belonging to the public team state. In other words, for a public team state S, the coordinator issues different recommendations to players for every possible information set associated with S. Then, dummy players are used to extract a specific action for each infoset from prescriptions and pass it to the next player. Whenever it is the coordinator's turn to play, all the possible action combinations are listed. Therefore, during the transformation, every added dummy player node corresponds to another node. For any episode, the size of the game tree transformed by TPICA is $\mathcal{O}\big((|A|^{|\Omega|})^{|\mathcal{T}|}\big)$.

3 METHOD

In this section, we provide a comprehensive introduction to our method. We begin by designing a new structure called *private information pre-branch* (PIPB), which provides the coordinator with all possible private information from teammates. This effectively reduces the size of the transformed game since the amount of potentially private information is fixed in a game. Then, we leverage this structure to propose a multi-player transformation algorithm (MPTA) based on

private information. It effectively transforms an ATG into a strategy-equivalent 2p0s game and expands the types of solvable games. Additionally, we provide a proof of equilibrium equivalence. Finally, we demonstrate that the size of the game tree transformed by our method is smaller than that of previous state-of-the-art algorithms.

3.1 THE STRUCTURE OF *PRIVATE INFORMATION PRE-BRANCH*

In the game transformation process, we introduce two types of players: the coordinator, representing the team, and the dummy player, who conveys teammates' possible private information to the coordinator. We consider the general case where players' private information is interdependent. For example, when a team member is dealt J from a deck of three cards J, Q, K, he can safely infer that his teammates' private information cannot include J. We define the PIPB structure as follows.

Definition 1. *Given a completely inflated vEFG $\mathcal{G}$ that satisfies the public-turn-taking property, let h_d denote a dummy player node and $H_t \subseteq H$ denote a set of nodes of the same team member. h_d and H_t form a PIPB iff $\forall h \in H_t, \forall \omega \in \Omega \setminus \{\omega_i\} : h_d \omega \sqsubseteq h$, where $i = P(h)$.*

Intuitively, all team member nodes in a PIPB are connected to a single dummy player node, which serves as their common parent node. The dummy player's available actions represent all possible private information of the teammates. These actions are *unobservable* to all players except those in the next layer. Here, the layer refers to the order of player decisions in the transformed game, which can be visualized in the game tree as hierarchical levels. In other words, only players who act immediately after the dummy player (i.e., those in the next layer) can observe the dummy player's actions, whereas all other players cannot. After the transformation, all team players are replaced by the coordinator. While the visibility of the dummy player's actions remains unchanged, the coordinator's public states will change.

3.2 MULTI-PLAYER TRANSFORMATION ALGORITHM

We propose a multi-player transformation algorithm that utilizes the PIPB structure to transform an ATG into a 2p0s game. This method achieves the equivalent transformation by using potentially private information of teammates. The pseudocode is provided in Algorithm 1. Our method relies on the tree-form structure, so we first construct a complete game tree for the original ATG and then traverse it in a depth-first pre-order manner.

To illustrate more clearly, we provide an example of an ATG, ignoring the opponent, transformed via our method, as shown in Figure 1. The converted result by TPICA can be found in Appendix B.1. In this example, the chance player deals one card to each player from a deck of three cards (J, Q, K) as their private information, and team members' actions are *public* (the opponent's actions are also *public* if he is not ignored). The actions of the chance player are *private* to team members as each team player knows only their own card. This means every team member node in the original game tree is a separate infoset. As shown in the left part of Figure 1, the set of all private information Ω is $\{J, Q, K\}$. When traversing to a team member node, a dummy player node is first added. This dummy player's actions represent all possible private information of the teammates, i.e., $\{Q, K\}$ or $\{J, K\}$. The private information of team members cannot be passed along since they are not publicly observable. The dummy player's actions are based on the private information of the specific team member, so they also cannot be passed along and can only be observed by team members at the next level. Then, we introduce coordinator nodes to make decisions in place of the team member nodes. Terminal nodes are copied according to the sequence of publicly observable actions in the original game. In particular, the payoff of a coordinator is the sum of all team members' payoffs.

Compared to TPICA, the PIPB structure in our method reduces the number of infosets in the transformed game. Coordinator nodes that share the same dummy player node belong to the same information set. Furthermore, the partition of the coordinator nodes' infosets is also based on actions observable to all team members (i.e., actions called *public*). We divide the coordinator's public infosets according to the concept of public state, as shown in the right part of Figure 1, where nodes of a player with the same number belong to the same public infoset. Moreover, since our method also models adversarial team games using vEFGs, as TPICA does, it remains compatible with abstraction and pruning techniques. Theorem 1 further shows that the game tree transformed by our method is theoretically smaller in size than that converted by TPICA.

Theorem 1. *Given an ATG G with visibility that satisfies the public-turn-taking property, and its transformed*

Algorithm 1 Multi-Player Transformation Algorithm

1: **Function** $MPTA(G)$
2: initialize G'
3: $N \leftarrow \mathcal{T} \cup \{o\} \cup \{c\}$
4: $N' \leftarrow \{t\} \cup \{o\} \cup \{c\}$
5: initialize h with the chance player node
6: $h' \leftarrow ProcOfTrans(G, G', h)$
7: **return** G'
8: **Function** $ProcOfTrans(G, G', h)$
9: $\Omega \leftarrow \text{Private}(G)$
10: **if** $P(h) = c$ **then**
11:    $h' \leftarrow h$
12:    $\mathcal{A}'(h') \leftarrow \mathcal{A}(h)$
13:    **for** $a' \in \mathcal{A}'(h')$ **do**
14:       $ProcOfTrans(G, G', ha')$
15:    **end for**
16: **else if** $P(h) = o$ **then**
17:    $h' \leftarrow h$
18:    $\mathcal{A}'(h') \leftarrow \mathcal{A}(h)$
19: **else if** $P(h) \in \mathcal{T}$ **then**
20:    add a dummy player node h_d as h's parent node
21:    $\mathcal{A}'(h_d) \leftarrow \Omega \setminus \{\omega_{P(h)}\}$
22:    **for** $a' \in \mathcal{A}'(h_d)$ **do**
23:       $h' \leftarrow h_d a'$
24:       **for** $a \in \mathcal{A}(h)$ **do**
25:          $ProcOfTrans(G, G', h'a)$
26:       **end for**
27:    **end for**
28: **else**
29:    $z' \leftarrow h$
30:    $u_t(z') \leftarrow \sum_{i \in \mathcal{T}} u_i(h)$
31:    $u'_o(z') \leftarrow -u_t(z')$
32: **end if**
33: **return** h'

---

game $G' = \text{MPTA}(G)$. The size of any episode in G' is $\mathcal{O}\big(\big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A|\big)^{|\mathcal{T}|}\big)$.

3.3 EQUILIBRIUM EQUIVALENCE

As an important theoretical guarantee for this work, we prove that the TMECor in the original game can be obtained by solving the NE in the transformed game. We call two strategies realization-equivalent if they induce the same probabilities for reaching nodes for all strategies of other players. In other words, the strategies of NE in the transformed game and the strategies of TMECor in the original game are realization-equivalent. We state Lemma 1 on strategy equivalence as follows.

Lemma 1. *Given an ATG G with visibility that satisfies the public-turn-taking property, and the transformed game $G' = \text{MPTA}(G)$. For any joint reduced pure strategy $\pi_\mathcal{T}$ in*

Table 1: Experimental results of the running time of TPICA and our method on several different types and sizes of game instances. Blank cells indicate that the experiment cannot be conducted.

Game instances	Total nodes			Team nodes		Adversary nodes		Runtime		Improvement
	Original	TPICA	**MPTA**	TPICA	**MPTA**	TPICA	**MPTA**	TPICA	**MPTA**	
12**K**3	151	5,395	583	300	144	294	72	139s	**0.76s**	**182.89×**
12**K**4	601	1,337,051	3,097	3,888	768	4,632	384	1,560s	**9.26s**	**168.47×**
12**K**6	3,001	34,191,721	23,161	261,360	5,760	368,760	2,880	>27h	**144s**	**694.44×**
13**K**6	23,401		271,441		75,240		22,680		**562s**	
13**K**8	109,201		1,713,601		475,440		142,800		**5,093s**	
14**K**6	115,921		1,796,401		528,480		105,120		**3,051s**	
12**L**33	13,183	10,777,963	57,799	614,172	14,664	475,566	6,864	56,156s	**240s**	**233.98×**
12**L**43	42,589		251,749		64,008		29,736		**3,006s**	
12**L**63	218,011		1,954,351		497,940		229,620		**9,024s**	
13**L**33	161,491		948,151		262,500		80,220		**4,014s**	
13**L**43	738,241		5,994,241		1,661,760		504,000		**137,817s**	
14**L**33	1,673,311		12,226,231		3,535,320		809,880		**143,475s**	
12**G**	2,509		92,581		29,700		1,464		**241s**	
13**G**	15,307		3,352,669		1,107,162		13,128		**50,107s**	

G, it can be mapped to a corresponding strategy π_t in G', and vice versa.

Leveraging Lemma 1, we present Theorem 2, which demonstrates the equivalence of payoffs.

Theorem 2. *Given a public-turn-taking ATG G with visibility. When transformed into the game $G' = \text{MPTA}(G)$, players following the same strategy profile in both games have equivalent payoffs.*

For brevity, we use the notation '$\mapsto$' to denote the strategy mapping relationship. If we transform each pure plan and sum their probability masses, we obtain the corresponding mixed strategy. Formally, for any $\mu_{\mathcal{T}}$, the corresponding mixed strategy is $\sum_{\pi_{\mathcal{T}}:\pi_t \mapsto \pi_{\mathcal{T}}} \mu_{\mathcal{T}}(\pi_{\mathcal{T}})$. Therefore, Lemma 1 also applies to mixed strategies. Specifically, any $\mu_{\mathcal{T}}$ in G can be mapped to μ_t in G', and vice versa.

Theorem 3. *Given an ATG G with visibility that satisfies the public-turn-taking property, and its transformed game $G' = \text{MPTA}(G)$. If (μ_t^*, μ_o^*) is an NE in G', then strategy $(\mu_T^*, \mu_o^*) : \mu_t^* \mapsto \mu_{\mathcal{T}}^*$ is a TMECor in G.*

4 EXPERIMENTAL EVALUATION

4.1 EXPERIMENTAL SETTING

We conduct experiments on the standard testbed for ATGs. More specifically, we use three different multi-player parametric versions of games: *Kuhn poker* [Farina et al., 2018, Kuhn, 1950b], *Leduc poker* [Farina et al., 2018, Southey et al., 2005] and *Goofspiel* [Farina et al., 2021, Ross, 1971], as they are commonly used for experimental evaluation [Farina et al., 2021]. Specifically, unlike the other two scenarios,

Goofspiel involves changes in the amount of players' private information during the game. The number of players in these games is parameterized for flexibility. The specific rules for these games are provided in Appendix C.

We denote the number of opponents by m and the number of team members by n. For brevity, we use the following symbols to describe the parameters of the experiments:

- $mn\mathbf{K}r$: *Kuhn poker* with r ranks;
- $mn\mathbf{L}rc$: *Leduc poker* with r ranks and c indistinguishable suits. The default maximum number of bets allowed per betting round is 1;
- $mn\mathbf{G}$: *Goofspiel* with three ranks.

In this work, we adopt the state-of-the-art method that can be combined with 2-player game algorithms as our baseline. Specifically, we use the TPICA proposed by Carminati et al. [2022] as the benchmark for comparison. Since TPICA is not open-source, we reproduced it based on the descriptions and details provided in their paper. Note that in our experimental setup, the opponent acts first. To ensure a rigorous and fair comparison, we employ the *counterfactual regret minimization plus* [Tammelin et al., 2015], a well-established algorithm for finding NE in 2p0s games, in both our method and the baseline. All experiments are run on a machine with 18-core 2.7GHz CPU and 250GB memory.

4.2 EXPERIMENTAL RESULTS

In Table 1, we use the number of total nodes to represent the size of different games, where the column 'Original' represents the scale of the original ATG, and the other columns represent the game size after being transformed
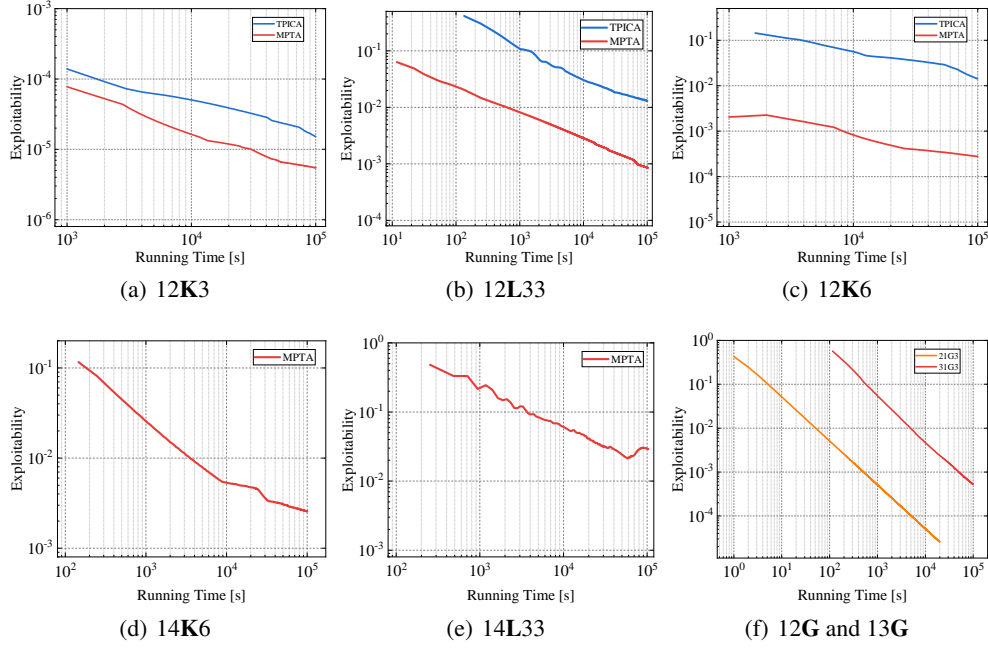
Figure 2: Comparison of exploitability in the same running time. All experiments except 12G run for 100,000 seconds. TPICA fails to work due to out-of-memory in 14**K**6 and 14**L**33 and cannot run on Goofspiel due to changes in private information.

by TPICA and our method, respectively. Furthermore, we also provide specific data for the coordinator and adversary nodes. The running time is provided in the column 'Runtime'. The TPICA and MPTA algorithms are run for comparison under the same machine configuration and identical experimental conditions. In the four cases of 12**K**3, 12**K**4, 12**K**6 and 12**L**33 where both MPTA and TPICA are applicable, the total time required by our approach to compute a TMECor is $0.76s$, $9.26s$, $144s$, and $240s$ respectively, which are $182.89\times$, $168.47\times$, $694.44\times$ and $233.98\times$ faster than TPICA. These results show that our method effectively reduces the game size, significantly improving solving speed by several orders of magnitude. It is worth noting that, in certain large-scale game scenarios where TPICA is unable to transform the original game tree, MPTA can still effectively support the computation of TMECor. In particular, 14**K**6 and 14**L**33 are 5-player cases that have never been used as experiments by previous algorithms due to their sheer size. We also observe by the node data in Table 1 that the reason for the speed-up is mainly due to the PIPB structure, which significantly reduces the number of adversary nodes and temporary chance nodes. Furthermore, we conduct detailed analyses on the solving efficiency and execution efficiency of the algorithms.

Solving efficiency. A smaller exploitability value indicates that the current strategy profile is closer to the TMECor. To evaluate the efficiency of the solving process, we conducted a series of tests on the change in exploitability over time within a constrained runtime of 10^5 seconds, applied

to seven different game instances, including *Kuhn poker*, *Leduc poker* and *Goofspiel*, as shown in Figure 2. Figures 2(a), 2(b), and 2(c) show that MPTA consistently outperforms TPICA. This demonstrates that our method delivers higher computational accuracy within the same runtime. As the complexity of the game instances increases, the performance gap between MPTA and TPICA becomes more pronounced, underscoring MPTA's greater capability in handling larger-scale scenarios. For example, TPICA fails due to out-of-memory in the cases of 14**K**6 and 14**L**33, as shown in Figures 2(d) and 2(e). Although MPTA has not yet fully converged to an approximate equilibrium within the given runtime of 10^5 seconds, it has the potential to do so with more computation time. Figure 2(f) highlights MPTA's robust performance in the *Goofspiel* game, which involves dynamic changes in players' private information. TPICA's reliance on fixed *prescriptions*, which specify an action for each infoset, makes it unsuitable for such games, demonstrating the superior generalizability of our approach.

Execution efficiency. The process of finding a TMECor is inherently iterative. To evaluate the efficiency of algorithm execution, we conducted comparative experiments measuring the time taken to complete the same number of iteration rounds across four distinct scenarios: 12**K**3, 12**K**4, 12**K**6, and 12**L**33. As illustrated in Figure 3, MPTA consistently requires less time than TPICA to compute approximate equilibrium strategy profiles. This difference in execution time is particularly evident in Figures 3(a) and 3(b), where our method completes 20,000 iterations in 1,740 seconds for the
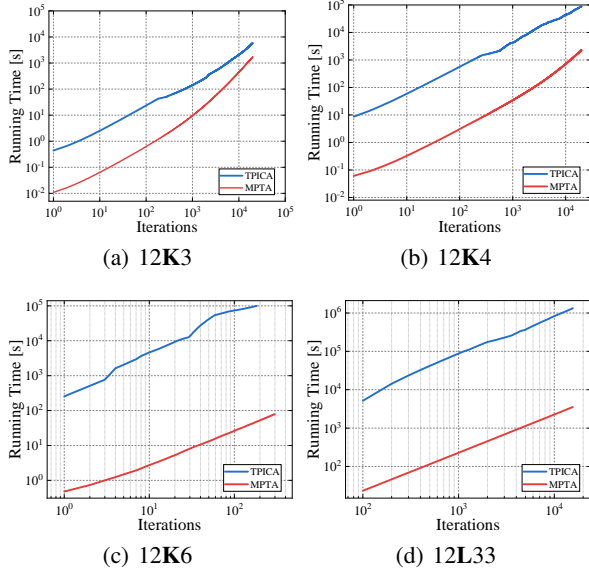
(a) 12**K**3   (b) 12**K**4

(c) 12**K**6   (d) 12**L**33

Figure 3: Comparison of runtime within the same number of iterations. All schemes except for 12**K**6 have been iterated for 20,000 rounds, as the TPICA is too time-consuming to run more rounds.

12**K**3 scenario and 2,293 seconds for the 12**K**4 scenario. In contrast, TPICA requires a much longer 5,878 seconds and 89,337 seconds to complete the same iterations under identical conditions, resulting in substantial speed improvements of 182.89 and 168.47 times, respectively.

The execution efficiency of MPTA becomes even more pronounced as the game scale increases, which is clearly demonstrated in larger, more complex scenarios like 12**K**6 and 12**L**33. Figure 3(c) shows that MPTA completes 300 rounds in just 79 seconds in the 12**K**6 scenario, while TPICA struggles, taking a staggering 99,600 seconds to complete only 184 rounds. Similarly, Figure 3(d) shows that MPTA completes 15,600 iterations in 3,555 seconds in the 12**L**33 scenario, whereas TPICA requires an enormous 1,327,652 seconds to achieve the same number of iterations. This translates to remarkable speed improvements of 694.44 and 233.98 times, respectively. These results emphasize the dramatic enhancement in execution efficiency provided by our method, particularly in scenarios involving larger strategy spaces and greater computational complexity.

5 RELATED WORK

Significant research has been focused on finding suitable solutions for ATGs since the concept of team-maxmin equilibrium was introduced by von Stengel and Koller [1997]. According to the communication capabilities of the team members, Celli and Gatti [2018] defined three different scenarios and corresponding equilibria for the first time in the

extensive-form ATGs.

Basilico et al. [2017] proposed a modified version of the quasi-polynomial time algorithm and a novel anytime approximation algorithm named *IteratedLP*, whose working principle is to maintain the current solution, providing a policy that can be returned at any time for each team member. Farina et al. [2018] adopted a novel realization-form representation that maps the problem of finding an optimal ex-ante-coordinated policy for the team to the problem of finding NE, significantly simplifying the analysis of team coordination in adversarial settings. Zhang and An [2020a] investigated the computational inefficiency resulting from the correlation between team members' strategies and proposed an associated recursive asynchronous multiparametric disaggregation technique to accelerate the computation of TMECor. They accomplished this by reducing the solution space of a mixed integer linear program using an association constraint. Successively, Zhang and An [2020b], Zhang et al. [2021], Farina et al. [2021], Zhang et al. [2022b] proposed more efficient variants of the LP. Although Zhang and Sandholm [2022] used a tree decomposition for constraints and described the team's strategy space by a polytope, finding TMECor still requires solving an LP, which remains computationally expensive in large-scale games. Some researchers have attempted to use multi-agent deep reinforcement learning to handle ATGs. For instance, Cacciamani et al. [2021] added a game-theoretic centralized training regimen and served as a buffer of past experiences. However, this method can only be applied to games where team members have symmetric observations of each other. It cannot be extended to general games with private information and public actions, such as poker.

The idea of using a coordinator to facilitate the coordination between team members can be traced back to the seminal work of decentralized stochastic control [Nayyar et al., 2013]. The TPICA proposed by Carminati et al. [2022] is closely related to our method, providing strong theoretical guarantees for finding equilibrium strategies in ATGs. However, TPICA's action-based game transformation method leads to an exponential growth in game size, which severely limits its scalability in practical applications. In contrast, our method not only offers the same theoretical guarantees but also significantly reduces game size, greatly improving the efficiency of computing TMECor. Moreover, our method expands the types of solvable games, primarily due to the designed PIPB structure.

6 CONCLUSION AND DISCUSSION

In this paper, we propose a multi-player transformation algorithm that establishes a connection between 2p0s games and ATGs. Our method effectively restricts the exponential growth of the transformed game's action space by leveraging the PIPB structure, allowing it to efficiently handle situa-

tions where private information changes during the game. Furthermore, we prove the equilibrium equivalence between the original and transformed games, providing a solid theoretical foundation for the validity of our approach. Through 14 experiments conducted on multiple standard testbeds, we consistently observed exceptional performance, further demonstrating the practical effectiveness and robustness of our method.

Our approach has potential real-world applications, such as modeling team members' inability to communicate in environmental protection efforts across different regions. In future work, we plan to expand on our method by exploring equilibrium refinement techniques.

Acknowledgements

References

Tiago C. Asmus, Graçaliz P. Dimuro, and Benjamín Bedregal. On two-player interval-valued fuzzy bayesian games. *Int J Intell Syst*, 32(6):557–596, 2017.

Nolan Bard, Jakob N. Foerster, Sarath Chandar, Neil Burch, Marc Lanctot, H. Francis Song, Emilio Parisotto, Vincent Dumoulin, Subhodeep Moitra, Edward Hughes, Iain Dunning, Shibl Mourad, Hugo Larochelle, Marc G. Bellemare, and Michael Bowling. The hanabi challenge: A new frontier for AI research. *Artificial Intelligence*, 280: 103216, 2020.

Nicola Basilico, Andrea Celli, Giuseppe De Nittis, and Nicola Gatti. Computing the team–maxmin equilibrium in single–team single–adversary team games. *Intelligenza Artificiale*, 11(1):67–79, 2017.

Noam Brown and Tuomas Sandholm. Safe and nested subgame solving for imperfect-information games. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 689–699, 2017.

Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. *Science*, 365(6456):885–890, 2019.

Noam Brown, Adam Lerer, Sam Gross, and Tuomas Sandholm. Deep counterfactual regret minimization. In *Proceedings of the 36th International Conference on Machine Learning*, ICML'19, pages 793–802. PMLR, 2019.

Federico Cacciamani, Andrea Celli, Marco Ciccone, and Nicola Gatti. Multi-agent coordination in adversarial environments through signal mediated strategies. *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, pages 269–278, May 2021.

Murray Campbell, A Joseph Hoane Jr, and Feng-hsiung Hsu. Deep blue. *Artificial Intelligence*, 134(1-2):57–83, 2002.

Luca Carminati, Federico Cacciamani, Marco Ciccone, and Nicola Gatti. A marriage between adversarial team games and 2-player games: Enabling abstractions, no-regret learning, and subgame solving. *Proceedings of the 39th International Conference on Machine Learning*, 162: 2638–2657, July 2022.

Andrea Celli and Nicola Gatti. Computational results for extensive-form adversarial team games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018.

Gabriele Farina, Andrea Celli, Nicola Gatti, and Tuomas Sandholm. Ex ante coordination and collusion in zero-sum multi-player extensive-form games. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 9661–9671, 2018.

Gabriele Farina, Andrea Celli, Nicola Gatti, and Tuomas Sandholm. Connecting optimal ex-ante collusion in teams to extensive-form correlation: Faster algorithms and positive complexity results. In *Proceedings of the 38th International Conference on Machine Learning*, pages 3164–3173, 18–24 Jul 2021.

Kristoffer Arnsfelt Hansen, Thomas Dueholm Hansen, Peter Bro Miltersen, and Troels Bjerre Sørensen. Approximability and parameterized complexity of minmax values. In *Internet and Network Economics: 4th International Workshop*, pages 684–695, 2008.

Harold W Kuhn. Extensive games. *Proceedings of the National Academy of Sciences*, 36(10):570–576, 1950a.

Harold W Kuhn. A simplified two-person poker. *Contributions to the Theory of Games*, 1:97–103, 1950b.

Chanjuan Liu, Enqiang Zhu, Qiang Zhang, and Xiaopeng Wei. Exploring the effects of computational costs in extensive games via modeling and simulation. *Int J Intell Syst*, 36(8):4065–4087, 2021.

Wenjun Ma, Xudong Luo, and Yuncheng Jiang. Matrix games with missing, interval, and ambiguous lottery payoffs of pure strategy profiles and compound strategy profiles. *Int J Intell Syst*, 33(3):529–559, 2018.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

John Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.

Ashutosh Nayyar, Aditya Mahajan, and Demosthenis Teneketzis. Decentralized stochastic control with partial history sharing: A common information approach. *IEEE Transactions on Automatic Control*, 58(7):1644–1658, 2013.

Sheldon M Ross. Goofspiel—the game of pure strategy. *Journal of Applied Probability*, 8(3):621–625, 1971.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Finnegan Southey, Michael H. Bowling, Bryce Larson, Carmelo Piccione, Neil Burch, Darse Billings, and D. Chris Rayner. Bayes' bluff: Opponent modelling in poker. *Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence*, pages 550–558, 2005.

Oskari Tammelin, Neil Burch, Michael Johanson, and Michael Bowling. Solving heads-up limit texas hold'em. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 645–652. AAAI Press, 2015.

Bernhard von Stengel and Daphne Koller. Team-maxmin equilibria. *Games and Economic Behavior*, 21(1-2):309–321, 1997.

Brian Zhang, Luca Carminati, Federico Cacciamani, Gabriele Farina, Pierriccardo Olivieri, Nicola Gatti, and Tuomas Sandholm. Subgame solving in adversarial team games. In *Advances in Neural Information Processing Systems*, volume 35, pages 26686–26697. Curran Associates, Inc., 2022a.

Brian Hu Zhang and Tuomas Sandholm. Team correlated equilibria in zero-sum extensive-form games via tree decompositions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5252–5259, Jun. 2022.

Brian Hu Zhang, Luca Carminati, Federico Cacciamani, Gabriele Farina, Pierriccardo Olivieri, Nicola Gatti, and Tuomas Sandholm. Subgame solving in adversarial team games. In *Advances in Neural Information Processing Systems*, volume 35, pages 26686–26697, 2022b.

Brian Hu Zhang, Gabriele Farina, and Tuomas Sandholm. Team belief DAG: Generalizing the sequence form to team games for fast computation of correlated team maxmin equilibria via regret minimization. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pages 40996–41018, 23–29 Jul 2023.

Youzhi Zhang and Bo An. Computing team-maxmin equilibria in zero-sum multiplayer extensive-form games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(02):2318–2325, Apr. 2020a.

Youzhi Zhang and Bo An. Converging to team-maxmin equilibria in zero-sum multiplayer games. *Proceedings of the 37th International Conference on Machine Learning*, pages 11033–11043, 2020b.

Youzhi Zhang, Bo An, and Jakub Černý. Computing ex ante coordinated team-maxmin equilibria in zero-sum multiplayer extensive-form games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(6):5813–5821, May 2021.

Youzhi Zhang, Bo An, and Daniel Dajun Zeng. DAG-based column generation for adversarial team games. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pages 58563–58578, 21–27 Jul 2024.

Yichi Zhou, Tongzheng Ren, Jialian Li, Dong Yan, and Jun Zhu. Lazy-cfr: fast and near-optimal regret minimization for extensive games with imperfect information. In *Proceedings of the 8th International Conference on Learning Representations, ICLR*, 2020.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, volume 20, pages 1729–1736, 2007.

Enhanced Equilibria-Solving via Private Information Pre-Branch Structure in Adversarial Team Games
(Appendix)

Chen Qiu[1] Haobo Fu[2] Kai Li[3,4] Jiajia Zhang[1] Xuan Wang[*1]

[1]School of Computer Science and Technology, Harbin Institute of Technology Shenzhen, China
[2]Tencent AI Lab, China
[3]Institute of Automation, Chinese Academy of Sciences, China
[4]School of Artificial Intelligence, University of Chinese Academy of Sciences, China

A PROOFS

A.1 THE PROOF OF THEOREM 1

Theorem 1. *Given an ATG G with visibility that satisfies the public-turn-taking property, and its transformed game $G' = \mathrm{MPTA}(G)$. The size of any episode in G' is $\mathcal{O}\big(\big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A|\big)^{|\mathcal{T}|}\big)$.*

Proof. We assume that each player has $|A|$ available actions at every state in the original game. During the traversal of the original game tree, the dummy player nodes will provide all possible private information from the teammates. Therefore, the number of available actions at the dummy player nodes is $\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}$. The coordinator inherits the team members' actions since they are *public*. When the team in the game consists of two players, the size of any episode in G' is given by:

$$\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!} + \frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A| + \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^2|A|$$
$$+ \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^2|A|^2$$

When the team in the game consists of three players, the size of any episode in G' is given by:

$$\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!} + \frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A| + \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^2|A|$$
$$+ \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^2|A|^2 + \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^3|A|^2$$
$$+ \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^3|A|^3$$

Thus, extending to the general case where the team consists of $|\mathcal{T}|$ players, the size of any episode in G' is given by:

$$\sum_{n=1}^{|\mathcal{T}|} \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^n (|A|^{n-1} + |A|^n).$$

Let $S_1 = \sum_{n=1}^{|\mathcal{T}|} (\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!})^n |A|^{n-1}$, and $S_2 = \sum_{n=1}^{|\mathcal{T}|} (\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!})^n |A|^n$. Then, we have:

$$\sum_{n=1}^{|\mathcal{T}|} \big(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}\big)^n (|A|^{n-1} + |A|^n) = S_1 + S_2.$$

---

[*]Corresponding author.

First, consider S_1:

$$S_1 = \frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!} + \left(\frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}\right)^2 |A| + \dots$$
$$+ \left(\frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}\right)^{|\mathcal{T}|} |A|^{|\mathcal{T}| - 1}.$$

S_1 meets the criteria for a finite geometric series. Using the geometric series sum formula, we have:

$$S_1 = \frac{(|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!})^{|\mathcal{T}|} - 1}{|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!} - 1} \frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}$$

Similarly, consider S_2:

$$S_2 = \frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}|A| + \left(\frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}\right)^2 |A|^2 + \dots$$
$$+ \left(\frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}\right)^{|\mathcal{T}|} + |A|^{|\mathcal{T}|}.$$

Thus,

$$S_2 = \frac{(|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!})^{|\mathcal{T}|} - 1}{|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!} - 1} \frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!}|A|.$$

Adding S_1 and S_2, we have:

$$S_1 + S_2 = (1 + |A|)\frac{(|\Omega| - 1)!}{(|\Omega| - |\mathcal{T}|)!} \frac{(|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!})^{|\mathcal{T}|} - 1}{|A|\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!} - 1}.$$

Therefore, the size of any episode in G' is $\mathcal{O}\left(\left(\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A|\right)^{|\mathcal{T}|}\right)$.

This concludes the proof. $\qquad\square$

Let the game transformed by TPICA be denoted as $\hat{G}$. In $\hat{G}$, each team member node, except for the team member node who first acts, corresponds to an additional dummy player node. We recognize that, according to the *prescription* property in TPICA, every coordinator node except for the first one will be matched with a dummy player node. By applying a derivation process similar to Theorem 1, the size of any episode in $\hat{G}$ is $2\sum_{n=1}^{|\mathcal{T}|}(|A|^{|\Omega|})^n$. Using the geometric series sum formula, the above equation is equal to $2|A|^{|\Omega|}\frac{(|A|^{|\Omega|})^{|\mathcal{T}|} - 1}{|A|^{|\Omega|} - 1}$ (i.e., $\mathcal{O}((|A|^{|\Omega|})^{|\mathcal{T}|})$). Then, we compare the bases of the two results (i.e., $\frac{(|\Omega|-1)!}{(|\Omega|-|\mathcal{T}|)!}|A|$ and $|A|^{|\Omega|}$). Clearly, the exponential growth rate of the latter far exceeds the polynomial growth rate of the former.

A.2   THE PROOF OF LEMMA 1

Lemma 1. *Given an ATG G with visibility that satisfies the public-turn-taking property, and the transformed game $G' = \mathrm{MPTA}(G)$. For any joint reduced pure strategy $\pi_{\mathcal{T}}$ in G, it can be mapped to a corresponding strategy π_t in G', and vice versa.*

Proof. We can prove Lemma 1 by recursively traversing G and G' in a depth-first pre-order manner.

Case 1: from $\pi_{\mathcal{T}}$ to π_t.   Let h and h' denote the current nodes reached in G and G', respectively. G satisfies the public-turn-taking, ensuring h and h' either represent the same player or are both terminal nodes. Initializing h with the chance player node. Ω is a set of all private information in G. When constructing π_t:

1) For the chance node: $\pi_c = \pi_{c'}$ always holds as our algorithm does not modify the chance node. This ensures that the actions specified by the chance player's strategy in the original game are the same as those in the transformed game.

2) For opponent nodes: The proof is identical to 1).

3) For team member nodes: Let $\pi_{\mathcal{T}}[I(h)]$ represent the joint reduced pure strategy of $\pi_{\mathcal{T}}$ at infoset $I(h)$. During the traversal, our algorithm expands h based on all the possibly private information of teammates into $|\Omega| - 1$ nodes. These nodes belong to the same infoset, denoted as I'. Let $\pi_t[I']$ represent the reduced pure strategy of π_t at I', where $I' \in S_t(h)$. When t is in a public state, $\pi_{\mathcal{T}}[I(h)] = \pi_t(I')$.

4) For terminal nodes: When reaching the terminal node through the above process, our algorithm ensures the following holds: $u'_t = \sum_{i \in \mathcal{T}} u_i(h)$ and $u'_o(h') = u_o(h) = -\sum_{i \in \mathcal{T}} u_i(h)$.

Case 2: from π_t to $\pi_{\mathcal{T}}$. The proof follows the same points as the previous case. $\square$

A.3 THE PROOF OF THEOREM 2

Theorem 2. *Given a public-turn-taking ATG G with visibility, and its transformed game $G' = \mathrm{MPTA}(G)$, they have equivalent payoffs.*

Proof. The proof directly relies on Lemma 1. Specifically, for any strategy $\pi_{\mathcal{T}}$ in G, we can find a corresponding strategy π_t in G' that yields the same payoff. Similarly, for any strategy π_t in G', there exists a payoff-equivalent strategy $\pi_{\mathcal{T}}$ in G. This ensures that the payoff for the players remains unchanged whether they choose π_t in G' or $\pi_{\mathcal{T}}$ in G. $\square$

A.4 THE PROOF OF THEOREM 3

Theorem 3. *Given an ATG G with visibility that satisfies the public-turn-taking property, and its transformed game $G' = \mathrm{MPTA}(G)$. If (μ_t^*, μ_o^*) is an NE in G', then strategy $(\mu_{\mathcal{T}}^*, \mu_o^*) : \mu_t^* \mapsto \mu_{\mathcal{T}}^*$ is a TMECor in G.*

Proof. For brevity, let u_t and $u_{\mathcal{T}}$ represent $u_t(\pi_c, \pi_o, \pi_t)$ and $u_{\mathcal{T}}(\pi_c, \pi_o, \pi_{\mathcal{T}})$, respectively. If μ_t^* is an NE in G', then by the definition of NE, the following holds:

$$\mu_t^* \in \arg\max_{\mu_t} \min_{\mu_o} \sum_{\substack{\pi_c \in \Pi_c \\ \pi_o \in \Pi_o \\ \pi_t \in \Pi_t}} \mu_c(\pi_c)\mu_o(\pi_o)\mu_t(\pi_t)u_t.$$

If $\mu_{\mathcal{T}}^*$ is a TMECor, it satisfies:

$$\mu_{\mathcal{T}}^* \in \arg\max_{\mu_{\mathcal{T}}} \min_{\mu_o} \sum_{\substack{\pi_c \in \Pi_c \\ \pi_o \in \Pi_o \\ \pi_{\mathcal{T}} \in \Pi_{\mathcal{T}}}} \mu_c(\pi_c)\mu_o(\pi_o)\mu_{\mathcal{T}}(\pi_{\mathcal{T}})u_{\mathcal{T}}.$$

Let $\min_{TMECor}(\mu_{\mathcal{T}})$ and $\min_{NE}(\mu_t)$ denote the inner minimization problems for TMECor and NE, respectively.

Assume there exists a strategy $\mu_{\mathcal{T}}'$ such that its value under the definition of TMECor is greater than that of $\mu_{\mathcal{T}}^*$. That is, $\min_{TMECor}(\mu_{\mathcal{T}}') > \min_{TMECor}(\mu_{\mathcal{T}}^*)$. According to Lemma 1, there exists a strategy μ_t' such that $\mu_{\mathcal{T}}' \mapsto \mu_t'$. From Theorem 2, we then have:

$$\min_{TMECor}(\mu_{\mathcal{T}}') = \min_{NE}(\mu_t') > \min_{NE}(\mu_t^*).$$

This results in a contradiction, as it implies that μ_t^* is not an NE in G'. Therefore, $\mu_{\mathcal{T}}^*$ must be a TMECor in G. $\square$

B MORE DETAILS OF CONCEPTUAL EXPLANATION AND THE CONVERTED RESULT

B.1 CONVERTED RESULT BY TPICA

Figure 4 is the result of the TPICA transformation. Due to the large number of nodes, we only display a portion of the transformed game tree.
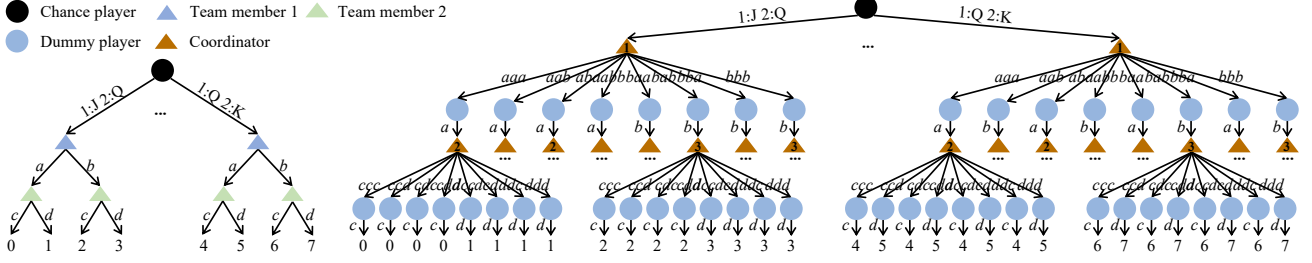
Figure 4: Example of game transformation. "..." indicates omitted branches. The nodes of a player with the same number are in the same infoset. **Left:** Original ATG omitting the opponent. **Right:** Result of transforming the game on the left using TPICA.

B.2 CONCEPTUAL EXPLANATION OF TEAM-PUBLIC-INFORMATION REPRESENTATION

The left side of Figure 4 shows the original ATG tree, ignoring the opponent nodes, with all nodes forming the set H. Since every player can only observe the cards dealt to himself by the chance player, the actions of the chance player are *private* to the coordinator. The actions of all other players are observable, so their actions are *public* to the coordinator. In the left side of Figure 4, the two nodes belonging to team member 1 are in two infosets due to the different private information at each node. For team member 2, the actions observed under the same hand are different, making each of the four nodes belonging to team member 2 a separate information set as well. The set of private information in this game is $\Omega = \{J, Q, K\}$. All leaf nodes form the set of terminal nodes Z.

The right side of Figure 4 shows the result of the game transformation using TPICA. The coordinator represents a team consisting of team members 1 and 2. The coordinator's public state is divided only by actions called *public*. Therefore, the two nodes of team member 1 in the original game tree belong to the same public state. Since we omitted the branches of the chance nodes, marked by "...", the actual possible deals are: $[1 : J, 2 : Q]$, $[1 : J, 2 : K]$, $[1 : Q, 2 : J]$, $[1 : Q, 2 : K]$, $[1 : K, 2 : J]$, and $[1 : K, 2 : Q]$. Nodes with the same private information are in the same infoset. That is, there are three infosets for team member 1 at this level, each consisting of two nodes. According to the concept of *prescription*, every *prescription* should select an action from the three infosets to form recommendations. Thus, there are 2^3 *prescriptions*, i.e., aaa, aab, aba, abb, baa, bab, bba, and bbb. The dummy player will select a specific action from these recommendations as the coordinator's available action. For instance, in the left subtree, the dummy player chooses the first action from each *prescription*; in the right subtree, the dummy player chooses the last action from each *prescription*. The same process applies to the coordinator when traversing to team member 2.

C RULES OF VARIOUS GAME INSTANCES

In our work, the number of players in each game scenario is parameterized for flexibility. To clearly articulate each instance's rules, we will illustrate using the 3-player version as an example.

- **The rule of *Kuhn poker*:** In 3-player Kuhn poker, there are three players and k possible cards. Players take turns acting in sequence. Before the game starts, each player pays one chip to the pot and is dealt a private card. The game proceeds with the following steps:

  1) Player 1 can choose to check or bet. If checking, the betting round continues with step 2); otherwise, the betting round proceeds to step 3).

  2) Player 2 can choose to check or bet. It is important to note that if Player 2 chooses to bet, then Player 1 must decide between folding or calling after Player 3's action. If Player 2 also chooses to check, the betting round continues with step 4).

  3) Player 2 can choose to fold or call.

  4) Player 3 chooses to check or bet. When Player 3 checks, the betting round ends; otherwise, Player 1 and Player 2 must decide between folding or calling.

  5) Player 3 chooses to fold or call. The betting round concludes after her decision.

  We assume that Player 1 is the adversary, while Player 2 and Player 3 are team members. In the event of the opponent's victory, Players 2 and 3 share the loss. If the team wins, Player 2 and Player 3 share the team's rewards. The n-player

Kuhn poker adopted in our work is an extended version based on the 3-player Kuhn poker.

- **The rule of *Leduc poker*:** In the 3-player version of adversarial team Leduc poker, the deck contains three suits and $k \geq 3$ card ranks. Each player starts by contributing one chip to the pot and receiving a private card. There are two betting rounds in total. After the first betting round, the community card is revealed. Then, players who have not folded proceed to the second betting round. After the conclusion of the second betting round, players remaining in the game will reveal their private cards. If a player pairs her card with the community card, she will win the pot.

  If a player's single private card forms a pair with the community card, she will win the pot. Otherwise, the player with the highest private card wins. We assume that Player 1 is the opponent, while Player 2 and Player 3 are team members. In the adversarial team games, there are some modifications to the payoff structure. If Player 1 wins, she takes all the chips from the pot. If Player 2 or Player 3 wins, the chips contributed by the team members are returned to them, and the chips bet by Player 1 are evenly distributed among each team member.

- **The rule of *Goofspiel*:** *Goofspiel* is a bidding game. We adopt a variant version with three cards. Every player has a hand of cards with values $\{1, 2, 3\}$. A third stack of cards, also with values $\{1, 2, 3\}$, is shuffled and placed on the table. At the onset of each round, a neutral referee places a card on the table as the reward for that round. Players bid by selecting a card from their hand, and the player with the highest bid claims the reward. In the event of a tie, the reward is equitably shared among the tying players. After three rounds, all rewards are distributed among the players, contributing to their scores. Assuming that Player 2 and Player 3 form a team, the final score of each team member is calculated by summing and averaging the rewards obtained by Player 2 and Player 3.