Computational Results for Extensive-Form Adversarial Team Games

Andrea Celli, Nicola Gatti

Politecnico di Milano
Piazza Leonardo da Vinci, 32
Milano, Italy
{andrea.celli, nicola.gatti}@polimi.it

Abstract

We provide, to the best of our knowledge, the first computational study of extensive-form adversarial team games. These games are sequential, zero-sum games in which a team of players, sharing the same utility function, faces an adversary. We define three different scenarios according to the communication capabilities of the team. In the first, the teammates can communicate and correlate their actions both *before* and *during* the play. In the second, they can only communicate *before* the play. In the third, *no* communication is possible at all. We define the most suitable solution concepts, and we study the inefficiency caused by partial or null communication, showing that the inefficiency can be arbitrarily large in the size of the game tree. Furthermore, we study the computational complexity of the equilibrium-finding problem in the three scenarios mentioned above, and we provide, for each of the three scenarios, an exact algorithm. Finally, we empirically evaluate the scalability of the algorithms in random games and the inefficiency caused by partial or null communication.

Introduction

The design of algorithms for strategic settings has been a central problem in Artificial Intelligence for several years, with the aim of developing agents capable of behaving optimally when facing strategic opponents. Many efforts have been made for 2-player games, e.g., finding a *Nash* equilibrium (Lemke and Howson, Jr 1964; Gatti et al. 2012) and, more recently, finding a *Stackelberg* equilibrium (Conitzer and Sandholm 2006). The study of this latter problem paved the way to the field of Security Games, which is, nowadays, one of the application fields of non-cooperative game theory with the highest social impact (Tambe 2011).

Fewer results are known, instead, about games with more than 2 players—except for games with particular structures, e.g., congestion and polymatrix games (Nisan et al. 2007). An interesting class of games widely unexplored in the literature is that one of adversarial team games. Here, a team of players with the same utilities faces an adversary (von Stengel and Koller 1997). These games can model many realistic security scenarios and can provide tools to coordinate teammates acting strategically. Basilico *et al.* (2017) study

the inefficiency a team can incur in normal-form games when teammates cannot correlate their strategies w.r.t. when they can. They also provide algorithms to approximate the *Team-maxmin* equilibrium—introduced by von Stengel and Koller (1997)—that is the optimal solution when correlation is not possible. Furthermore, it is known that finding a Team-maxmin equilibrium is FNP-hard and inapproximable in additive sense (Hansen et al. 2008; Borgs et al. 2010).

When extensive-form games enter the picture, adversarial team games become much more intriguing, various forms of correlation being possible. Nevertheless, to the best of our knowledge, this game class is still unexplored in the literature. In the present paper, we focus on three main forms of correlation (Forges 1986). In the first, *preplay* and *intraplay* communication is possible, corresponding to the case in which a *communication device* receives *inputs* from the teammates about the information they observe during the play, and sends them *recommendations* about the action to play at each information set. In the second, *only preplay* communication among the teammates is possible, corresponding to the case in which a *correlation device* communicates a plan of actions to each teammate before playing the game.[1] Finally, in the third, *no communication* is possible[2].

Original contributions. We formally define game models capturing the three aforementioned cases and the most suitable solution concepts (trivially, the Team-maxmin equilibrium in the third setting). Furthermore, we define three

---

[1] With only preplay correlation, three solution concepts are known: Normal-Form, Extensive-Form, and Agent-Form Correlated Equilibrium. The spaces of players' strategies achievable with the three correlation devices are the same, while the players' incentive constraints are different (even if it is not known whether the spaces of the outcomes for the three equilibria in adversarial team games are different). The complexity of computing the equilibrium maximizing the team's utility in adversarial team games with at least 2 teammates is NP-hard for all the three equilibria (von Stengel and Forges 2008). In our paper, we focus on the first one.

[2] This setting is frequent in practice. Consider, for example, a security problem where a set of defensive resources from different security agencies are allocated to protect an environment at risk but, due to organizational constraints, they are not able to share information with each other. The resources have the same goal (i.e., to secure the environment) but cannot coordinate strategy execution. The same happens when a set of resources has to operate in stealth mode.

inefficiency indices for the equilibria, capturing: the inefficiency caused by using a correlation device in place of a communication device, the inefficiency caused by not using any form of communication in place of a communication device, and the inefficiency caused by not using any form of communication in place of a correlation device. We provide lower bounds to the worst-case values of the inefficiency indices, showing that they can be arbitrarily large.

Furthermore, we study the computational complexity of the problems of finding the three equilibria with the different forms of correlation, and we design, for each equilibrium, an exact algorithm. We show that when a communication device is available, an equilibrium can be computed in polynomial time (even in the number of players) by a 2-stage algorithm. In the first stage, the game is cast into an auxiliary 2-player equivalent game, while, in the second stage, a solution is found by linear programming. When a correlation device is available, the problem can be easily shown to be FNP-hard. In this case, we prove that there is always an equilibrium with a small (linear) support, and we design an equilibrium-finding algorithm, based on a classical column-generation approach, that does not need to enumerate an exponential space of actions before its execution. Our algorithm exploits an original hybrid representation of the game combining both normal and sequence forms. The column-generation oracle is shown to deal with an APX-hard problem, with an upper approximation bound decreasing exponentially in the depth of the tree. We also provide an approximation algorithm for the oracle that matches certain approximation guarantees on a subset of instances. When no communication is possible, the equilibrium-finding problem can be easily shown to be FNP-hard. In this case, the problem can be formulated as a non-linear programming problem and solved by resorting to global optimization tools.

Finally, we empirically evaluate the scalability of our algorithms in random game instances. We also evaluate the inefficiency for the team of not adopting a communication device, showing that, differently from the theoretical worst-case bounds, the empirical inefficiency is extremely small.

Preliminaries

A *perfect-information extensive-form game* (Shoham and Leyton-Brown 2009) is a tuple $(N, A, V, L, \iota, \rho, \chi, U)$, where: N is a set of n players, A is a set of actions, V is the set of nonterminal decision nodes, L is the set of terminal (leaf) nodes, $\iota : V \to N$ is a function returning the player acting at a given decision node, $\rho : V \to 2^A$ is the action function—assigning to each choice node a set of available actions—, $\chi : V \times A \to V \cup L$ is the successor function, and $U = \{U_1, U_2, \ldots, U_n\}$ is the set of utility functions in which $U_i : L \to \mathbb{R}$ specifies utilities over terminal nodes for player i. Let V_i be the inclusion-wise maximal set of decision nodes such that, for all $x \in V_i$, $i = \iota(x)$. Then, an *imperfect-information extensive-form game* is a tuple $(N, A, V, L, \iota, \rho, \chi, U, H)$, where $(N, A, V, L, \iota, \rho, \chi, U)$ is an extensive-form game with perfect information and $H = \{H_1, H_2, \ldots, H_n\}$ is the set of information sets, in which H_i is a partition of V_i such that, for any $x_1, x_2 \in V_i$, $\rho(x_1) = \rho(x_2)$ whenever there exists a $h \in H_i$ where $x_1 \in h$

and $x_2 \in h$. As usual in game theory, we assume, for each $a \in A$, there is only one h s.t. $a \in \rho(h)$. We focus on games with *perfect recall* where, for each player i and each $h \in H_i$, decision nodes belonging to h share the same sequence of moves of player i on their paths from the root.

The study of extensive-form games is commonly conducted under other equivalent representations. The *normal form* is a tabular representation in which player i's actions are plans $p \in P_i$, specifying a move at each information set in H_i, and player i's utility is $U_i' : P_1 \times \ldots \times P_n \to \mathbb{R}$ s.t. $U_i'(p_1, \ldots, p_n) = U_i(l)$, where $l \in L$ is the terminal node reached when playing plan profile $(p_1, \ldots, p_n)$. Basically, in the normal-form representation, players decide their behavior in the whole game *ex ante* the play. The *reduced* normal form is obtained by deleting replicated strategies from the normal form. However, the size of the reduced normal form is exponential in the number of information sets. A *mixed strategy* σ_i of player $i \in N$ is a probability distribution on her set of pure strategies P_i. In the *agent form*—whose definition is omitted due to reasons of space, see (Selten 1975)—, players play *behavioral strategies*, denoted by $\pi_i(h, a)$, each specifying a probability distribution over the actions $\rho(h)$ available at information set h of player i. Two strategies, even of different representations, are *realization equivalent* if, for any strategy profile of the opponents, they induce the same probability distribution over the outcomes. In a finite perfect-recall game, any mixed strategy can be replaced by an equivalent behavioral one (Kuhn 1953).

Both normal and agent forms suffer from computational issues that can be overcome by using the *sequence form* (von Stengel 1996), whose size is linear in the size of the game tree. A sequence for player i, defined by a node x of the game tree, is the subset of A specifying player i's actions on the path from the root to x. We denote the set of sequences of player i by Q_i, these are the sequence-form actions of player i. A sequence is said *terminal* if, together with some sequences of the other players, leads to a terminal node. Moreover, we denote by $q_\varnothing$ the fictitious sequence leading to the root node and with $qa \in Q_i$ the *extended* sequence obtained by appending $a \in A$ to sequence $q \in Q_i$. The sequence-form strategy, said *realization plan*, is a function $r_i : Q_i \to \mathbb{R}$ associating each sequence $q \in Q_i$ with its probability of being played. A well-defined sequence-form strategy is such that, for each $i \in N$, $r_i(q_\varnothing) = 1$, for each h and sequence q leading to h, $-r_i(q) + \sum_{a \in \rho(h)} r_i(qa) = 0$ and $r_i(q) \geqslant 0$. Constraints are linear in the number of sequences and can be written as $F_i r_i = f_i$, where F_i is an opportune matrix and f_i is an opportune vector. The utility function of player i is represented as an n-dimensional matrix defined only for profiles of terminal sequences leading to a leaf. With a slight abuse of notation, we denote it by U_i.

A Nash equilibrium (NE), whose definition does not depend on the representation of the game, is a strategy profile in which no player can improve her utility by deviating from her strategy once fixed the strategies of all the other players.

Extensive-Form Adversarial Team Games, Equilibria, and Inefficiency

We initially provide the formal definition of a team.

Definition 1 (Team) *Given an extensive-form game with imperfect information* $(N, A, V, L, \iota, \rho, \chi, U, H)$*, a team* $\mathcal{T}$ *is an inclusion-wise maximal subset of players* $T \subseteq N$ *such that, for any* $i, j \in T$*, for all* $l \in L$*,* $U_i(l) = U_j(l)$*.*

We denote by $H_\mathcal{T}$ the set $\bigcup_{i \in T} H_i$ and by $A_\mathcal{T}$ the set of actions available at the information sets in $H_\mathcal{T}$. An extensive-form team game (EF-TG) is a generic extensive-form game where at least one team is present. Von Stengel and Koller (1997) analyze zero-sum normal-form games where a single team plays against an adversary. We extend this game model to the scenario of extensive-form games.

Definition 2 (STSA-EF-TG) *A zero-sum single-team single-adversary extensive-form team game (STSA-EF-TG) is a game* $(N, A, V, L, \iota, \rho, \chi, U, H)$ *in which:*

- $N = T \cup \{n\}$*, where set* T *defines a team* $\mathcal{T}$ *(as in Definition 1) and player* n *is the adversary* ($\mathcal{A}$)*;*
- *for each* $l \in L$ *it holds:* $U_\mathcal{A}(l) = -(n-1)U_\mathcal{T}(l)$*, where* $U_\mathcal{T}$ *denotes the utility of teammates and* $U_\mathcal{A}$ *that one of the adversary.*

When the teammates have no chance to correlate their strategies, the most appropriate solution concept is the Team-maxmin equilibrium (TME). Formally, the TME is defined as $\arg\max_{r_1,\ldots,r_{n-1}} \min_{r_n} U_\mathcal{T} \prod_{i=1}^{n} r_i$. By using the same arguments used by von Stengel and Koller (1997) for the case of normal-form games, it follows that also in extensive-form games a TME is unique except for degeneracy and it is the NE maximizing team's expected utility. Nevertheless, in many scenarios, teammates may exploit higher correlation capabilities. While in normal-form games these capabilities reduce to employing a correlation device as proposed by (Aumann 1974), in extensive-form games we can distinguish different forms of correlation. More precisely, the strongest correlation is achieved when teammates can communicate both before and during the execution of the game (preplay and intraplay communication), exchanging their private information by exploiting a *mediator* that recommends actions to them. This setting can be modeled by resorting to a *communication device* defined in a similar way to (Forges 1986). A weaker correlation is achieved when teammates can communicate only before the play (preplay communication). This setting can be modeled by resorting to a *correlation device* analogous to that one for normal-form games. We formally define these two devices as follows (as customary, $\Delta(\cdot)$ denotes the simplex over $\cdot$).

Definition 3 (Communication device) *A communication device is a triple* $(H_\mathcal{T}, A_\mathcal{T}, R^{\mathrm{Com}})$ *where* $H_\mathcal{T}$ *is the set of inputs (i.e., information sets) that teammates can communicate to the mediator,* $A_\mathcal{T}$ *is the set of outputs (i.e., actions) that the mediator can recommend to the teammates, and* $R^{\mathrm{Com}} : 2^{H_\mathcal{T}} \times 2^{A_\mathcal{T}} \to \Delta(A_\mathcal{T})$ *is the recommendation function that associates each information set* $h \in H_\mathcal{T}$ *with a probability distribution over* $\rho(h)$*, as a function of information sets previously reported by teammates and of the actions recommended by the mediator in the past.*

Definition 4 (Correlation device) *A correlation device is a pair* $(\{P_i\}_{i \in T}, R^{\mathrm{Cor}})$*.* $R^{\mathrm{Cor}} : \times_{i \in T} P_i \to \Delta(\times_{i \in T} P_i)$ *is the recommendation function which returns a probability distribution over the reduced joint plans of the teammates.*

Notice that, while a communication device provides its recommendations drawing actions from probability distributions during the game, a correlation device does that only before the beginning of the game. Resorting to these definitions, we introduce the following solution concepts.

Definition 5 (Team-maxmin equilibrium variations) *Given a communication device—or a correlation device—for the team, a Team-maxmin equilibrium with communication device (TMECom)—or a Team-maxmin equilibrium with correlation device (TMECor)—is a Nash equilibrium in which all teammates follow their recommendations and, only for TMECom, report truthfully their information.*

Notice that in our setting (i.e., zero-sum games), both TMECom and TMECor maximize team's utility. We state the following, whose proof is straightforward.

Property 1 (Strategy space) *The space of lotteries over the outcomes achievable by using a communication device includes that one of the lotteries achievable by using a correlation device, that, in its turn, includes the space of the lotteries achievable without any device.*

Let v_{No}, v_{Com}, v_{Cor} be the utility of the team at, respectively, the TME, the TMECom and the TMECor. From the property above, we can easily derive the following.

Property 2 (Equilibria utility) *The game values obtained with the different solution concepts introduced above are such that* $v_{\mathrm{Com}} \geqslant v_{\mathrm{Cor}} \geqslant v_{\mathrm{No}}$*.*

In order to evaluate the inefficiency due to the impossibility of adopting a communication or correlation device, we resort to the concept of Price of Uncorrelation (PoU), previously introduced in (Basilico et al. 2017) as a measure of the inefficiency of the TME w.r.t. the TMECor in normal-form games. In these games, the PoU is defined as the ratio between the utility given by the TMECor and the utility given by the TME, once all the team's payoffs are normalized in $[0, 1]$. For extensive-form games, we propose the following variations of the PoU to capture all the possible combinations of different forms of correlation.

Definition 6 (Inefficiency indices) $PoU_{\mathrm{Com/No}} = \frac{v_{\mathrm{Com}}}{v_{\mathrm{No}}}$, $PoU_{\mathrm{Cor/No}} = \frac{v_{\mathrm{Cor}}}{v_{\mathrm{No}}}$, $PoU_{\mathrm{Com/Cor}} = \frac{v_{\mathrm{Com}}}{v_{\mathrm{Cor}}}$.

In perfect-information games all these indices assume a value of 1, the solution being unique unless degeneracy by backward induction. With imperfect information the indices can be larger than 1. In normal-form games, the tight upper bound to PoU is m^{n-2}, where m is the number of actions of each player and n is the number of players (Basilico et al. 2017). Using a definition based on m is not suitable for extensive-form games, where each player may have a different number of actions per node. Thus, we state the bounds in terms of $|L|$ (i.e., the number of terminal nodes). The following three examples provide lower bounds to the worst-case values of the indices, showing that the inefficiency may be arbitrarily large in L.
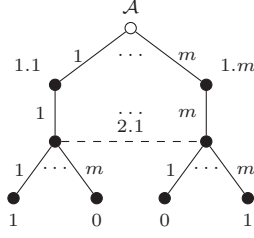
Figure 1: A game with a spy used in Example 1.

Initially, to ease the presentation, we define a specific type of team player that we call *spy*.

Definition 7 (Spy player) *Player $i \in T$ is said to be a spy if, for each $h \in H_i$, $|\rho(h)| = 1$ and h is a singleton.*

A spy just observes the actual state of the game and her contribution to the play is only due to her communication capabilities. Notice that the introduction of a spy after decision nodes of the adversary does not affect the team's utility in a TMECor (the team's joint plans are the same) but improves the team's capabilities, and final utility, in a TMECom.

Example 1 (Lower bound for worst-case $PoU_{\text{Com/No}}$)
Consider a STSA-EF-TG with n players and $m \geq 2$ actions for each player at every decision node except for the first team player, who is a spy. The game tree is structured as follows (see Figure 1 for the case with $n = 3$).

- *The adversary plays first;*
- *then the spy observes her move;*
- *each one of the other teammates is assigned one of the following levels of the game tree and all her decision nodes are part of the same information set;*
- *$U_{\mathcal{T}} = 1$ iff, for each $i \in T\backslash\{1\}$ and for each $h \in H_i$, the action chosen at h is equal to the one selected by $\mathcal{A}$.*

We have $v_{\text{Com}} = 1$, $v_{\text{No}} = m^{2-n}$ and thus $PoU_{\text{Com/No}} = m^{n-2}$. Since the tree structure is such that $|L| = m^{n-1}$ we obtain $PoU_{\text{Com/No}} = |L|^{(1-\frac{1}{n-1})}$. Once $|L|$ is fixed, the inefficiency is monotonically increasing in n, but n is upper bounded by $n = \log_2(|L|) + 1$ (corresponding to the case in which each team player except the spy has the minimum number of actions, i.e., 2). It follows that, in the worst case w.r.t. n, $PoU_{\text{Com/No}} = \frac{|L|}{2}$.

Example 2 (Lower bound for worst-case $PoU_{\text{Cor/No}}$)
Consider a STSA-EF-TG with n players and m actions at each of their decision nodes, in which each level of the game tree is associated with one player and forms a unique information set. $U_{\mathcal{T}} = 1$ iff all the teammates choose the same action of the adversary, who plays first. This case corresponds to the worst case for PoU in normal-form games. Here we formulate the bound in terms of $|L|$. We have $v_{\text{No}} = m^{1-n}$ and $v_{\text{Cor}} = 1/m$. It follows that $PoU_{\text{Cor/No}} = m^{n-2}$. This time, $|L| = m^n$ and thus $PoU_{\text{Cor/No}} = |L|^{(1-\frac{2}{n})}$. The worst case w.r.t. n is reached when $m = 2$ and $n = \log_2(|L|)$. Therefore, $PoU_{\text{Cor/No}} = \frac{|L|}{4}$.

Example 3 (Lower bound for worst-case $PoU_{\text{Com/Cor}}$)
Consider the game presented in Example 1. Since $v_{\text{Com}} = 1$ and $v_{\text{Cor}} = 1/m$, it follows $PoU_{\text{Com/Cor}} = m$. The structure of the game tree is such that $|L| = m^{n-1}$ and thus $PoU_{\text{Com/Cor}} = |L|^{\frac{1}{n-1}}$. Notice that, in this case, the inefficiency is maximized when $n = 3$, which corresponds to having a team of two members. Thus, in the worst case w.r.t. n, $PoU_{\text{Com/Cor}} = \sqrt{|L|}$.

Finding a TMECom

We show that there is a polynomial-time TMECom-finding algorithm. Indeed, we prove that the problem of finding a TMECom is equivalent to finding a 2-player maxmin strategy in an auxiliary 2-player game with perfect recall and that the auxiliary game can be built in polynomial time.

First, we define the structure of the auxiliary game we use. Let Γ be an extensive-form game and $Q = \bigcup_{i \in N} Q_i$. We define the following functions. Function $\text{lead} : V \cup L \to 2^Q$ returns the sequence profile constituting the path from the root to a given node of the tree. Function $\text{path} : V \times 2^N \to 2^Q$ is s.t., for each $x \in V$ and each set of players $G \subseteq N$,

$$\text{path}(x|G) = \left\{ q \subset \bigcup_{i \in G} Q_i \middle| \exists q' \subset \bigcup_{i \in N\backslash G} Q_i \land q \cup q' = \text{lead}(x) \right\}.$$

Intuitively, $\text{path}(x|G)$ returns the unique profile of sequences of players in G leading to x when combined with some sequences of the players in $N\backslash G$.

The following definition describes the information structure of the auxiliary extensive-form game.

Definition 8 (G-observable game) *For any game $\Gamma = (N, A, V, L, \iota, \rho, \chi, H)$ and any set of players $G \subseteq N$, the G-observable game $\hat{\Gamma}$ is a tuple $(N, A, V, L, \iota, \rho, \chi, \hat{H})$, where $\hat{H} = \left(\bigcup_{i \in G} \hat{H}_i \right) \cup \left(\bigcup_{i \in N\backslash G} H_i \right)$ is such that:*

1. *for each decision node $x \in V$, there exists one and only one $\hat{h} \in \hat{H}$ s.t. $x \in \hat{h}$ and $\iota(h) = \iota(\hat{h})$ where h denotes the information set containing x in Γ;*

2. *for each player $i \in G$, $\hat{H}_i$ is the set with the lowest possible cardinality s.t. for each $\hat{h} \in \hat{H}_i$ and for each pair of decision nodes $x, x' \in \hat{h}$, it holds:*

$$\left(\text{path}(x|G) = \text{path}(x'|G) \right) \land \left(\exists h \in H_i | x \in h \land x' \in h \right).$$

In a G-observable extensive-form game, players belonging to G are fully aware of the moves of other players in G and share the same information on the moves taken by players in $N\backslash G$. We show that we can build $\hat{\Gamma}$ in polynomial time.

Lemma 1 (T-observable game construction)
The T-observable game $\hat{\Gamma}$ of a generic STSA-EF-TG Γ can be computed in polynomial time.

Proof. We provide the sketch of an algorithm to build a T-observable game (i.e., a G-observable game with $G = T$) in time and space polynomial in the size of the game tree. The algorithm employs nested hash-tables. The first hash-table associates each joint sequence of the team with another

hash-table, which is indexed over information sets and has as value the information set id to be used in $\hat{\Gamma}$. Γ is traversed in a depth-first manner while keeping track of the sequence leading to the current node. For each $x \in V$ s.t. $\iota(x) \in T$, a search/insertion over the first hash-table is performed by hashing $\mathsf{path}(x|T)$. Then, once the sequence-specific hash-table is found, the information set is assigned a new id if it is not already present as a key. $\hat{\Gamma}$ is built by associating to each decision node of the team a new information set as specified in the hash-table. The worst-case running time is $O(|V|^2)$.□

Theorem 2 (TMECom computation) *Given a STSA-EF-TG and a communication device for $\mathcal{T}$, the unique (unless degeneracy) TMECom can be found in polynomial time.*

Proof. Given a STSA-EF-TG Γ, the use of a communication device for the team $\mathcal{T}$ changes the information structure of the game inducing a T-observable game $\hat{\Gamma}$. A TMECom can be computed over $\hat{\Gamma}$ as follows. Given a communication device $(H_{\mathcal{T}}, A_{\mathcal{T}}, R^{\mathrm{Com}})$, R^{Com} enforces a probability distribution γ over the set of feedback rules. γ is chosen in order to maximize the expected utility of the team. In this setting, no incentive constraints are required because teammates share the same utility function and therefore, under the hypothesis that γ maximizes it, it is in their best interest to follow the recommendations sent by the device and to report truthfully their information. Thus, considering the function path to be defined over information sets and $\hat{H}_{\mathcal{T}} = \bigcup_{i \in T} \hat{H}_i$, γ reduces to a distribution over rules of type $\{\beta = (\beta^h)_{h \in \hat{H}_{\mathcal{T}}} | \beta^h : \mathsf{path}(h|T) \to \rho(h), \forall h \in \hat{H}_{\mathcal{T}}\}$.

We are left with an optimization problem in which we have to choose γ s.t. the worst-case utility of the team is maximized. This is equivalent to a 2-player maxmin problem over $\hat{\Gamma}$ between $\mathcal{A}$ and a player playing over team's joint sequences. By construction, the team player has perfect recall and thus the maxmin problem can be formulated as an LP in sequence form, requiring polynomial time.   □

Finding a TMECor

We initially focus on the computational complexity of the problem of searching for a TMECor.

Theorem 3 (TMECor complexity) *Finding a TMECor is FNP-hard when there are two teammates, each with an arbitrary number of information sets, or when there is an arbitrary number of teammates, each with one information set.*

The first result directly follows from the reduction presented in (von Stengel and Forges 2008, Theorem 1.3) since the game instances used in the reduction are exactly STSA-EF-TGs with 2 teammates. The second result can be proved by adapting the reduction described in (Koller and Megiddo 1992, Proposition 2.6), assigning each information set of the game instances to a different teammate.

In principle, a TMECor can be found by casting the game in normal form and then by searching for a Team-maxmin equilibrium with correlated strategies. This latter equilibrium can be found in polynomial time in the size of the normal form, which, however, is given by $P_1 \times \ldots \times P_n$, where

each P_i is exponentially large in the size of the tree. We provide here a more efficient method that can also be used in an anytime fashion, without requiring any exponential enumeration before the execution of the algorithm. In our method, we use a hybrid representation that, to the best of our knowledge, has not been used in previous works.

Hybrid representation. In our representation, $\mathcal{A}$'s strategy is represented in sequence form, while the team plays over *jointly-reduced plans*, as formally defined below. Given a generic STSA-EF-TG Γ, let us denote with $P_r = \{P_{r,1}, \ldots, P_{r,n}\}$ the set of actions of the reduced normal-form of Γ, where $P_{r,i}$ is the set of reduced plans for player i. Therefore, $\times_{i \in T} P_{r,i}$ is the set of joint reduced plans of the team. Let function terminal : $Q_{\mathcal{A}} \times \{\times_{i \in T} P_{r,i}\} \to L \cup \{\varnothing\}$ be s.t. it returns, for a given pair $(q_{\mathcal{A}}, p)$, the terminal node reached when the adversary plays $q_{\mathcal{A}}$ and the team members, at each of their information set, play according to p. If no terminal node is reached, $\varnothing$ is returned. We define some equivalence classes over $\times_{i \in T} P_{r,i}$ by the relation $\sim$:

Definition 9 *The equivalence relation $\sim$ over $\times_{i \in T} P_{r,i}$ is s.t., given $p_1, p_2 \in \times_{i \in T} P_{r,i}$, $p_1 \sim p_2$ iff, for each $q_{\mathcal{A}} \in Q_{\mathcal{A}}$, terminal$(q_{\mathcal{A}}, p_1) =$ terminal$(q_{\mathcal{A}}, p_2)$.*

Definition 10 (Jointly-reduced plans) *The set of jointly-reduced plans $P_{jr} \subseteq \times_{i \in T} P_{r,i}$ is obtained by picking exactly one representative from each equivalence class of $\sim$.*

The team's utility function is represented by the sparse $|Q_{\mathcal{A}}| \times |P_{jr}|$ matrix U_h. Given a pair $(q_{\mathcal{A}}, p_{jr}) \in Q_{\mathcal{A}} \times P_{jr}$, $U_{\mathcal{T}}(\mathsf{terminal}(q_{\mathcal{A}}, p_{jr}))$ is stored in U_h iff terminal$(q_{\mathcal{A}}, p_{jr}) \neq \varnothing$. Notice that U_h is well defined since each pair $(q_{\mathcal{A}}, p_{jr})$ leads to at most one terminal-node.

Let $\sigma_{\mathcal{T}}$ denote the team's strategy over P_{jr}. The problem of finding a TMECor in our hybrid representation can be formulated as the following LP named HYBRID-MAXMIN:

$$\arg\max_{\sigma_{\mathcal{T}}, v} \sum_{h \in H_{\mathcal{A}} \cup \{h_{\varnothing}\}} f_{\mathcal{A}}(h)v(h) \quad \text{s.t.}$$

$$\sum_{h \in H_{\mathcal{A}} \cup \{h_{\varnothing}\}} F_{\mathcal{A}}(h, q_{\mathcal{A}})v(h) - \sum_{p \in P_{jr}} U_h(q_{\mathcal{A}}, p)\sigma_{\mathcal{T}}(p) \leqslant 0 \quad \forall q_{\mathcal{A}} \in Q_{\mathcal{A}}$$

$$\sum_{p \in P_{jr}} \sigma_{\mathcal{T}}(p) = 1$$

$$\sigma_{\mathcal{T}}(p) \geqslant 0 \quad \forall p \in P_{jr}$$

composed of $|Q_{\mathcal{A}}| + 1$ constraints (except $\sigma_{\mathcal{T}}(p) \geqslant 0$ constraints) and an exponential number of variables $\sigma_{\mathcal{T}}$. Thus, we can state the following proposition.

Proposition 1 *There exists at least one TMECor in which the number of joint plans played with strictly positive probability by the team is at most $|Q_{\mathcal{A}}|$.*

Proof. The above LP admits a basic optimal solution with at most $|Q_{\mathcal{A}}| + 1$ variables with strictly positive values (Shapley and Snow 1950). Since v is always in the basis (indeed, we can add a constant to make the team's utility in each terminal node strictly positive without affecting equilibrium strategies), the joint plans in the basis are $|Q_{\mathcal{A}}|$.   □

Proposition 1 shows that the NP-hardness of the problem is merely due to guessing the jointly-reduced plans played

with strictly positive probability in a TMECor. Thus, we can avoid enumerating entirely P_{jr} before executing the algorithm by working with a subset of jointly-reduced plans built progressively, in a classical column-generation fashion (see, e.g., (McMahan, Gordon, and Blum 2003)).

Column-generation algorithm. The pseudocode is given in Algorithm 1. It receives in input the game tree and the sequence-form constraint matrices F_i of all the players (Line 1). Then, the algorithm is initialized, assigning a matrix of zeros to U_h, an empty set to P_{cur}, and 0 to v (Line 2). Notice that U_h is sparse and therefore its representation requires a space equal to the number of non-null entries. $\overline{r}_{\mathcal{A}}$ is initialized as a realization plan equivalent to a uniform behavioral mixed strategy, i.e., the adversary, at each information set, randomizes uniformly over all the available actions (Line 3). Then, the algorithm calls the BR-ORACLE (defined below) to find the best response of the team given the adversary's strategy $\overline{r}_{\mathcal{A}}$ (Line 4). Lines 6-10 are repeated until an optimal solution is found. Initially, br is added to P_{cur} (Line 6) and players' utilities at nodes reached by $(q_{\mathcal{A}}, br)$ for every $q_{\mathcal{A}}$ are added to U_h. Then, the algorithm solves the maxmin (HYBRID-MAXMIN) and minmax (HYBRID-MINMAX) problems restricted to P_{cur} (Lines 8 and 9), where the HYBRID-MINMAX problem is defined as:

$$\underset{r_{\mathcal{A}}, v}{\arg\min} \quad v \quad \text{s.t.}$$

$$v - \sum_{q \in Q_{\mathcal{A}}} U_h(q, p_{jr}) r_{\mathcal{A}}(q) \geqslant 0 \qquad \forall p_{jr} \in P_{jr}$$

$$\sum_{q \in Q_{\mathcal{A}}} F_{\mathcal{A}}(h, q) = f_{\mathcal{A}}(h) \qquad \forall h \in H_{\mathcal{A}}$$

$$r_{\mathcal{A}}(q) \geqslant 0 \qquad \forall q \in Q_{\mathcal{A}}$$

Finally, the algorithm calls BR-ORACLE to find the best response to $\overline{r}_{\mathcal{A}}$ (Line 10).

Best-response oracle. Given a generic STSA-EF-TG Γ, we denote the problem of finding the best response of the team against a given a fixed realization plan $\overline{r}_{\mathcal{A}}$ of the adversary over Γ as BR-T. This problem is shown to be NP-hard in the reduction used for (von Stengel and Forges 2008, Theorem 1.3), where we can interpret the initial chance move as the fixed strategy of the adversary. We can strengthen such a hardness result as follows (the proofs are provided in the full version of the paper, available on the arXiv website):

Theorem 4 *BR-T is APX-hard.*

Let $\alpha_{(\cdot)} \in [0, 1]$ be the best approximation bound of the maximization problem $(\cdot)$.

Theorem 5 *Denote with BR-T-h the problem BR-T over STSA-EF-TG instances of fixed maximum depth $3h$ and branching factor variable at each decision-node, it holds:*

$$\alpha_{BT\text{-}T\text{-}h} \leqslant (\alpha_{MAX\text{-}SAT})^h.$$

This means that the upper bound on the approximation factor decreases exponentially as the depth of the tree increases[3].

---

[3]Notice that $\alpha_{MAX\text{-}SAT} \leqslant 7/8$, see (Håstad 2001).

Algorithm 1 Hybrid Column Generation

1: **function** HYBRID-COL-GEN($\Gamma, F_1, \ldots, F_{n-1}, F_{\mathcal{A}}$)   ▷ Γ is a generic STSA-EF-TG and F_i are sequence-form constraint matrices
2:    $U_h = \mathbf{0}, P_{cur} = \{\}, v \leftarrow 0$ ▷ initialization
3:    $\overline{r}_{\mathcal{A}} \leftarrow$ realization plan equivalent to a uniform behavioral mixed strategy
4:    $br \leftarrow$ BR-ORACLE($\Gamma, \{F_1, \ldots, F_{n-1}\}, \overline{r}_{\mathcal{A}}$)   ▷ call to the oracle
5:    **while** $br \notin P_{cur}$ **do**
6:      $P_{cur} \leftarrow P_{cur} \cup br$
7:      players' utilities in $(q_{\mathcal{A}}, br)$ for every $q_{\mathcal{A}}$ are added to U_h
8:      $\sigma_{\mathcal{T}} \leftarrow$ solve HYBRID-MAXMIN problem with $(U_h, P_{cur}, F_{\mathcal{A}})$
9:      $\overline{r}_{\mathcal{A}} \leftarrow$ solve HYBRID-MINMAX problem with $(U_h, P_{cur}, F_{\mathcal{A}})$
10:      $br \leftarrow$ BR-ORACLE($\Gamma, \{F_1, \ldots, F_{n-1}\}, \overline{r}_{\mathcal{A}}$)
11:    **return** $(\overline{r}_{\mathcal{A}}, \sigma_{\mathcal{T}})$

The column-generation oracle solving BR-T can be formulated as the following integer linear program (ILP):

$$\underset{r_1, \ldots, r_{(n-1)}, x}{\arg\max} \quad \sum_{l \in L} U_{\mathcal{T}}(l) x(l) \overline{r}_{\mathcal{A}}(\mathsf{path}(l|\{n\})) \quad \text{s.t.}$$

$$\sum_{q_i \in Q_i} F_i(h, q_i) r_i(q_i) = f_i(h) \qquad \substack{\forall i \in T, \\ \forall h \in H_i \cup \{h_{\varnothing}\}}$$

$$x(l) \leqslant r_i(q_i) \qquad \substack{\forall i \in T, \forall l \in L, \\ \forall q_i \in \mathsf{path}(l|\{i\})}$$

$$x(l) \in \{0, 1\} \qquad \forall l \in L$$

where $x(l)$ is a binary variable which is equal to 1 iff, for all the sequences $q_i \in Q$ necessary to reach l, it holds $r_i(q_i) = 1$. Notice that the oracle returns a pure realization plan for each of the teammates. Team's best-response is a jointly-reduced realization plan that can be derived as follows. Denote with Q_i^L the set of sequences played with probability one by i that are not subsets of any other sequence played with positive probability. Let p_i' be the reduced normal-form plan of player i specifying all and only actions played in the sequences belonging to Q_i^L. The joint plan $p' = (p_1', \ldots, p_{n-1}')$ is s.t. $p' \in P_{jr}$.

A simple approximation algorithm can be obtained by a continuous relaxation of the binary constraints $x(l) \in \{0, 1\}$. The resulting mathematical program is linear and therefore solvable in polynomial time. An approximated solution can be obtained by randomized rounding (Raghavan and Tompson 1987). When considering game trees encoding MAX-SAT instances (see the proof of Theorems 4), the approximation algorithm matches the ratio guaranteed by randomized-rounding for MAX-SAT (details are given in the full version of the paper).

Finding a TME

We recall that finding a TME is hard, since it is hard even with normal-form games (Hansen et al. 2008).

Theorem 6 (TME complexity) *Finding a TME is FNP-hard and its value is inapproximable in additive sense even with binary payoffs.*

The problem of finding a TME can be formulated as the following non-linear mathematical programming problem:
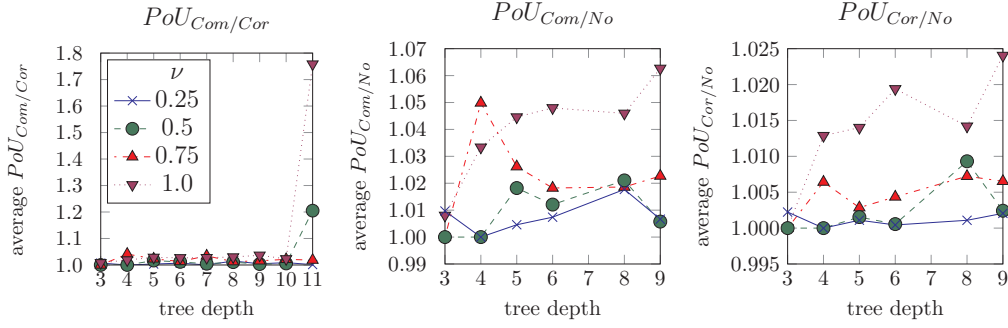
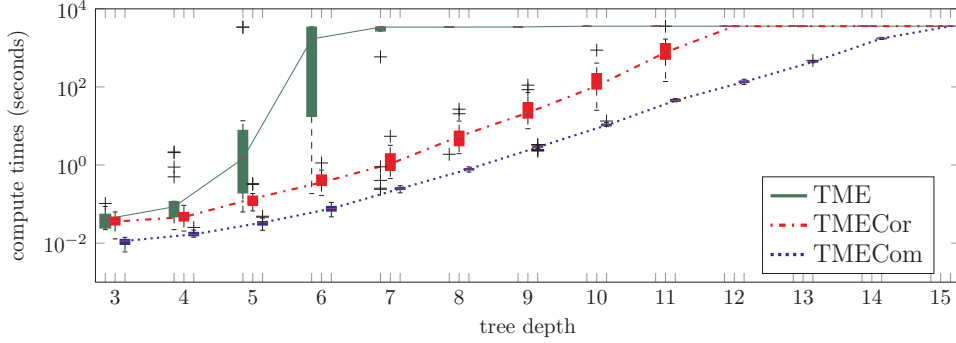Figure 2: Average empiric inefficiency indices with 3 players and some values of ν.

Figure 3: Average compute times of the algorithms and their box plots with 3 players and $\nu = 0.5$.

$$\max_{r_1, \ldots, r_{(n-1)}} \quad v(h_\varnothing) \qquad \text{s.t.}$$

$$\sum_{h \in H_\mathcal{A} \cup \{h_\varnothing\}} F_\mathcal{A}(h, q_\mathcal{A}) v(h) \leqslant$$

$$\leqslant \sum_{q_\mathcal{T} \in Q_\mathcal{T}} \left(U_\mathcal{T}(q_\mathcal{T}, q_\mathcal{A}) \prod_{i \in T} r_i(q_\mathcal{T}(i)) \right) \qquad \forall q_\mathcal{A} \in Q_\mathcal{A}$$

$$\sum_{q_i \in Q_i} F_i(h, q_i) r_i(q_i) = f_i(h) \qquad \substack{\forall i \in T, \\ \forall h \in H_i \cup \{h_\varnothing\}}$$

$$r_i(q_i) \geqslant 0 \qquad \substack{\forall i \in T, \\ \forall q_i \in Q_i}$$

where $Q_\mathcal{T}$ is the set of team's joint sequences and $q_\mathcal{T}(i)$ identifies the sequence of player i in $q_\mathcal{T}$. This program can be solved exactly, within a given numerical accuracy, by means of global optimization tools in exponential time.

Experimental Evaluation

Experimental setting. Our experimental setting is based on randomly generated STSA-EF-TGs. The random game generator takes as inputs: the number n of players, a probability distribution over the number of actions available at each information set, the maximum depth d of the tree, and a parameter ν for tuning the information structure of the tree. Specifically, this parameter encodes the probability with which a newly created decision-node, once it has been randomly assigned to a player, is assigned to an existing information-set

(thus, when it is equal to 0 the game is with perfect information), while guaranteeing perfect recall for every player. Finally, payoffs associated with terminal nodes are randomly drawn from a uniform distribution over $[0, 1]$. We generate 20 game instances for each combination of the following parameters' values: $n \in \{3, 4, 5\}$, $d \in \{n, \ldots, 15\}$ with step size 1 (i.e., for games with 5 players, $d \in \{5, 6, \ldots, 15\}$), $\nu \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$. For simplicity, we fix the branching factor to 2 (this value allows us to maximize d and it is also the worst case for the inefficiency indices).

The algorithms are implemented in Python 2.7.6, adopting GUROBI 7.0 for LPs and ILPs, AMPL 20170207 and global optimization solver BARON 17.1.2 (Tawarmalani and Sahinidis 2005). We set a time limit to the algorithms of 60 minutes. All the algorithms are executed on a UNIX computer with 2.33GHz CPU and 128 GB RAM. We discuss the main experimental results with 3 players below, while the results with more players are provided in the full version of the paper. Since the computation of the TMECor from the reduced normal form is impractical for $d \geqslant 5$, we use only Algorithm 1 employing the exact oracle (this demonstrated very fast on every instance).

Empirical PoUs. We report in Fig. 2 the average empiric inefficiency indices with 3 players for some values of ν. We observe that, despite the theoretical worst-case value increases in L, the empiric increase, if any, is negligible. For instance, the worst-case value of $PoU_{Com/Cor}$ with $n = 3$ and $L = 2^{11}$ is > 45, while the average empiric value is less

than 2. We also observe that the inefficiency increases in ν, suggesting that it may be maximized in normal-form games.

Compute time. We report in Fig. 3 the average compute times of the algorithms and their box plots with 3 players and $\nu = 0.5$ (the plot includes instances reaching the time limit as this not affects results presentation). As expected, the TMECom computation scales well, allowing one to solve games with more than 16,000 terminal nodes in the time limit. The performances of Algorithm 1 (TMECor) are remarkable since it solves games with more than 2,000 terminals in the time limit, and presents a narrow boxplot, meaning that the variance in the compute time is small. Notice that, with $d \leqslant 10$, the compute times of TMECom and TMECor are comparable, even if the former is computationally hard while the latter is solvable in polynomial-time. As expected, the TME computation does not scale well and its compute time is extremely variable among different instances.

Conclusions

In this paper, we focus on extensive-form team games with a single adversary. Our main contributions include the definition of game models employing different correlation devices and their suitable solution concepts. We study the inefficiency a team incurs employing various forms of correlation, providing lower bounds to the worst-case values of the inefficiency indices that are arbitrarily large in the game tree. Furthermore, we study the complexity of finding the equilibria, and we provide exact algorithms. Finally, we experimentally evaluate the scalability of our algorithms and the empirical equilibrium inefficiency in random games. In the future, it would be interesting to study approximate equilibrium-finding algorithms in order to reach an improved scalability in all the three correlation scenarios.

References

Aumann, R. 1974. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics* 1(1):67–96.

Basilico, N.; Celli, A.; Nittis, G. D.; and Gatti, N. 2017. Team-maxmin equilibrium: efficiency bounds and algorithms. In *AAAI*.

Borgs, C.; Chayes, J. T.; Immorlica, N.; Kalai, A. T.; Mirrokni, V. S.; and Papadimitriou, C. H. 2010. The myth of the folk theorem. *Games and Economic Behavior* 70(1):34–43.

Conitzer, V., and Sandholm, T. 2006. Computing the optimal strategy to commit to. In *ACM EC*, 82–90.

Forges, F. 1986. An approach to communication equilibria. *Econometrica* 1375–1385.

Gatti, N.; Patrini, G.; Rocco, M.; and Sandholm, T. 2012. Combining local search techniques and path following for bimatrix games. In *UAI*, 286–295.

Hansen, K. A.; Hansen, T. D.; Miltersen, P. B.; and Sørensen, T. B. 2008. Approximability and parameterized complexity of minmax values. In *WINE*, 684–695.

Håstad, J. 2001. Some optimal inapproximability results. *Journal of the ACM (JACM)* 48(4):798–859.

Koller, D., and Megiddo, N. 1992. The complexity of two-person zero-sum games in extensive form. *Games and economic behavior* 4(4):528–552.

Kuhn, H. W. 1953. *Extensive Games and the Problem of Information*. Princeton University Press. 193–216.

Lemke, C. E., and Howson, Jr. 1964. Equilibrium Points of Bimatrix Games. *Journal of the Society for Industrial and Applied Mathematics* 12(2):413–423.

McMahan, H. B.; Gordon, G. J.; and Blum, A. 2003. Planning in the presence of cost functions controlled by an adversary. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 536–543.

Nisan, N.; Roughgarden, T.; Tardos, E.; and Vazirani, V. 2007. *Algorithmic game theory*, volume 1. Cambridge University Press.

Raghavan, P., and Tompson, C. D. 1987. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica* 7(4):365–374.

Selten, R. 1975. Reexamination of the perfectness concept for equilibrium points in extensive games. *International journal of game theory* 4(1):25–55.

Shapley, L. S., and Snow, R. N. 1950. Basic solutions of discrete games. *Annals of Mathematics Studies* 24:27–35.

Shoham, Y., and Leyton-Brown, K. 2009. Multiagent systems: Algorithmic, game-theoretic, and logical foundations.

Tambe, M. 2011. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press.

Tawarmalani, M., and Sahinidis, N. V. 2005. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming* 103:225–249.

von Stengel, B., and Forges, F. 2008. Extensive-form correlated equilibrium: Definition and computational complexity. *Mathematics of Operations Research* 33(4):1002–1022.

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

von Stengel, B. 1996. Efficient computation of behavior strategies. *Games and Economic Behavior* 14(2):220 – 246.