POLITECNICO
MILANO 1863

POLITECNICO DI MILANO
DIPARTIMENTO DI ELETTRONICA, INFORMAZIONE E BIOINGEGNERIA
DOCTORAL PROGRAM IN INFORMATION TECHNOLOGY

COORDINATION AND CORRELATION IN MULTI-PLAYER SEQUENTIAL GAMES

Doctoral Dissertation of:
Andrea Celli

Supervisor:
Prof. Nicola Gatti

Tutor:
Prof. Cesare Alippi

The Chair of the Doctoral Program:
Prof. Barbara Pernici

2019 – Cycle XXXII

Abstract

Computing game-theoretic solution concepts is fundamental to describing the behavior of rational agents taking part in strategic interactions. The large majority of equilibrium-finding techniques are only suited for two-player, zero-sum games, where it is possible to compute strong solutions in theory and in practice. In this thesis, we make a step in the direction of solving more general problems, by focusing on multi-player, general-sum, extensive-form games. In many real-world problems, agents may exploit some form of communication to achieve coordinated behaviors. We mainly focus on the problem of reaching coordination under minimal communication requirements, that is by assuming agents can communicate only before the beginning of the game. We identify three different settings in which communication facilitates coordinated behaviors, and study each of them from a computational perspective.

First, we study problems where teams of agents interact against an opponent. In doing so, we define the appropriate solution concepts, and classify their computational complexity and inefficiencies. We study how to compute an optimal solutions in each of these settings. Then, we focus on the case in which only preplay communication is permitted, and develop the first scalable algorithm to learn an approximate solution for the problem.

The second problem we investigate is computing correlated equilibria and coarse correlated equilibria in multi-player, general-sum, sequential games. We characterize the computational complexity of computing optimal and approximate solutions for these problems, providing both positive and negative results. Then, we describe algorithmic results, and adapt a

state-of-the-art regret minimization technique for the two-player, zero-sum setting to find an approximate coarse correlated equilibrium in our setting.

The last problem we study is how to coordinate agents' behavior through the strategic provision of payoff-relevant information. We study information-structure design problems in which multiple agents can interact in a sequential game. We propose a novel notion of persuasive signaling scheme, and characterize its computational complexity.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

The computational study of game-theoretic solution concepts is fundamental to describe the optimal behavior of rational agents interacting in a strategic setting, and to predict the most likely outcome of a game. Equilibrium computation techniques have been applied to numerous real-world problems. Among other applications, they are the key building block of the best poker-playing AI agents [30, 32, 131], and have been applied to physical and cybersecurity problems (see, *e.g.*, [152, 139, 51, 56, 161, 64, 65]).

This thesis focuses on extensive-form games, which can model sequential moves, imperfect information, and outcome uncertainty. In this setting, a vast body of literature focuses on the computation of Nash equilibria in two-player, zero-sum games (see, *e.g.*, [166, 109, 180, 55, 116]), where recent results demonstrated that it is possible to compute strong solutions in theory and practice. Some works weaken these assumptions by considering two-player, general-sum games [110, 169, 10], or by studying multi-player games with particular structures, such as *compact games* (see, *e.g.*, [15, 57]). Brown and Sandholm [32] recently showed that it is possible to exploit techniques developed for the two-player, zero-sum setting in games with more than two players. However, these results are specific to the poker domain, and lack theoretical guarantees on general multi-player

extensive-form games.

While relevant, two-player, zero-sum games are rather restrictive, as many practical scenarios are not zero-sum and involve more than two players. Moreover, especially in general-sum games, the adoption of a Nash equilibrium may present some difficulties when used as a prescriptive tool. Indeed, when multiple Nash equilibria coexist, the model prevents players from synchronizing their strategies, since communication between players is prohibited. In real-world scenarios, where some form of communication among players is usually possible, different solution concepts are required as communication allows players for coordinated behaviors. This thesis focuses on scenarios where players can exploit *preplay communication* [73, 75], *i.e.*, players have an opportunity to discuss and agree on tactics before the game starts, but will be unable to communicate during the game. Consider, as an illustration, the case of a poker game where multiple players are colluding against an identified target player. Colluders can agree on shared tactics before the beginning of the game, but are not allowed any explicit communication while playing. In other settings, players might be forced to cooperate by the nature of the interaction itself. This is the case, for instance, in Bridge. Preplay players' coordination introduces new challenges with respect to the case in which agents take decisions individually, as understanding how to coordinate before the beginning of the game requires reasoning over the entire game tree. It is easy to see that this causes an exponential blowup in the agents' action space and, therefore, even relatively small game instances are usually deemed intractable in this setting.

When modeling preplay communication, it is instructive to introduce an additional agent, called the *mediator*, that does not take part in the game, but may send signals (usually actions' recommendations) to other players just before the beginning of the game. We explore different forms of preplay coordination in sequential games. The scenarios we consider can be classified through the following questions: i) *who is receiving the mediator's recommendations?* ii) *do players have similar goals?* iii) *is the mediator self-interested? and does the mediator have more information on the state of the game than other players?*

First, we study games comprising a *team* of agents, *i.e.*, agents sharing the same objectives. Team members coordinate their actions against an opponent. Even without communication during the game, the planning phase gives the team members an advantage: for instance, the team members could skew their strategies to use certain actions to signal about their state (for example, in card-playing games, the current hands they're hold-

ing). In other words, by having agreed on each member's planned reaction under any possible circumstance of the game, information can be silently propagated in the clear, by simply observing public information.

Then, we consider the case in which agents receiving mediator's recommendations may not share the same objectives. In this case, the mediator has to incentivize each player to follow moves' recommendations. The mediator is assumed to be *benevolent*, *i.e.*, she aims at maximizing the expected social welfare of the game.

Finally, we study the case in which the mediator is self-interested, and may exploit asymmetries in the availability of information to design a signaling scheme, in order to persuade players to select favorable actions. In this setting, the mediator is looking for a way to coordinate the individual behavior of each player in order to reach a preferred outcome of the game.

This thesis explores the effects of preplay communication in these scenarios from a practical and theoretical perspective. We study the advantages of coordinating strategies before the beginning of the game, the complexity of such problem, and devise algorithms that are theoretically sound as well as applicable in practice.

In the next section, we describe the structure of the thesis and provide a concise description of its main original contributions.

1.1 Structure of the Thesis and Original Contributions

Chapter 2 describes some fundamental notions required to understand the remainder of the thesis, and discuss relevant works related to ours.

Then, the thesis is structured in three main parts, corresponding to the three aforementioned coordination scenarios. We briefly describe the contributions of each of these parts.

1.1.1 Part I: Extensive-Form Team Games

We focus on the setting in which a team of agents faces an adversary in a sequential interaction. In Chapter 3, we formally define our model and show that different forms of intra-team communication result in different models of coordination: i) a mediator that can send and receive *intraplay* signals (*i.e.*, messages are exchanged during the execution of the game); ii) a mediator that only exploits preplay communication, sending recommendations just before the beginning of the game; iii) team members jointly plan their strategies, but have no access to a mediator to synchronize action execution. The thesis mainly focuses on the second scenario, where only preplay communication is possible. Scenarios i) and iii) are instructive to

understand the advantages of different forms of intra-team communication. These different coordination capabilities are compared via the analysis of inefficiency indexes measuring the relative losses in the team's expected utility.

Chapter 4 describes algorithms to compute an exact optimal equilibrium point, whose nature depends on the communication capabilities of the team. Then, we focus on preplay communication, and develop a learning algorithm to compute an approximate solution (Chapter 5). In doing so, we highlight a strong analogy with imperfect-recall games, and propose a new game representation, which can also be applied to this setting. Then, we use the new representation to derive an auxiliary construction that allows us to map the problem of finding an optimal coordinated strategy for the team to the well-understood Nash equilibrium-finding problem in a (larger) two-player zero-sum perfect-recall extensive-form game. By reasoning over the auxiliary game, we devise an anytime algorithm, *fictitious team-play*, that is guaranteed to converge to an optimal coordinated strategy for the team against an optimal opponent.

Finally, Chapter 6 presents an experimental evaluation of the proposed techniques. First, it compares the empirical inefficiencies of the solution concepts defined in Chapter 3, showing that, in practice, preplay communication is often enough to reach near-optimal performances. Then, we demonstrate the scalability of the learning algorithm presented in Chapter 5 on standard imperfect-information test instances.

The results on team games were published as Celli and Gatti [40] at AAAI-2018, and Farina et al. [66] at NeurIPS-2018.

1.1.2 Part II: Correlated Equilibria for Sequential Games

We investigate whether correlation can be reached efficiently even in settings where players have limited communication capabilities (*i.e.*, they can only observe signals before the beginning of the game). Therefore, we focus on sequential games in which only preplay communication is admitted, and study correlated equilibria that allow the mediator to recommend actions just *before* the playing phase of the game (namely, the *correlated equilibrium* (CE) and the *coarse correlated equilibrium* (CCE)).

In Chapter 7, we provide several results characterizing the complexity of computing optimal (*i.e.*, social welfare maximizing) CEs and CCEs, and their approximation complexity. First, we prove that approximating an optimal (*i.e.*, social welfare maximizing) CE is not in Poly-APX even in two player games without chance moves, unless P=NP. Next, we identify the

conditions for which finding an optimal CCE is NP-hard. However, we show that an optimal CCE can be found in polynomial-time in two-player extensive-form games without chance moves. Finally, we complete the picture on the computational complexity of finding social-welfare-maximizing CCEs by showing that the problem is not in Poly-APX, unless P=NP, in games with three or more players (chance included).

Chapter 8 presents algorithms to compute CCEs in general-sum, multi-player, sequential games. First, we provide a column generation framework to compute optimal CCEs in practice, and show how to generalize it to the *hard cases* of the problem. Then, we focus on the problem of computing *an* ε-CCE (*i.e.*, an approximate CCE). We design an enhanced version of CFR [180] which computes an average correlated strategy which is guaranteed to converge to an approximate CCE with a bound on the regret which is sub-linear in the size of the game tree. Finally, Chapter 9 is devoted to an experimental evaluation of the techniques to compute CCEs presented in the previous chapter.

The results of this part of the thesis were published as Celli et al. [42], at AAMAS-2019, and Celli et al. [43], at NeurIPS-2019.

1.1.3 Part III: Private Bayesian Persuasion in Sequential Games

In this part of the thesis, we examine information-structure design problems as a means of forcing coordination towards a certain objective. More precisely, we start from the usual scenario where a mediator can communicate action recommendations to players before the beginning of a sequential game. Suppose that parties (*i.e.*, the mediator and the players) are asymmetrically informed about the current state of the game. Specifically, the mediator is able to observe more information than the other players. We pose the following question: *can the mediator exploit the information asymmetry to coordinate players' behavior toward a favorable outcome?*

This problem can be accurately modeled via the *Bayesian persuasion* framework. We investigate private persuasion problems with multiple receivers interacting in a sequential game, and study the continuous optimization problem of computing a private signaling scheme which maximizes the sender's expected utility. We show how to address sequential, multi-receiver settings algorithmically via the notion of *ex ante* persuasive signaling scheme, where the receivers commit to following the sender's recommendations having observed only the signaling scheme.

Chapter 10 is structured as follows: first, we introduce the notion of *ex ante* persuasive signaling scheme, and formalize its differences from *ex*

interim persuasive schemes. Then, we show that an *ex ante* persuasive signaling scheme can provide the sender with an expected utility that can be arbitrarily larger than the expected utility reached with an *ex interim* persuasive signaling scheme. Motivated by the hardness results for the *ex interim* setting with simultaneous moves, we study the problem of computing optimal *ex ante* signaling schemes. We show that an optimal *ex ante* signaling scheme may be computed in polynomial time in settings with two receivers and independent action types, which makes *ex ante* persuasive signaling schemes a persuasion tool which is applicable in practice. Moreover, we show that this result cannot be extended to settings with more than two receivers, as the problem of computing an optimal *ex ante* signaling scheme becomes NP-hard.

The results of this part of the thesis, as of October 2019, are under review as Celli et al. [41].

CHAPTER 2

Preliminaries

This chapter describes some fundamental notions in Game Theory, the multi-player models central to this thesis, and the basic Bayesian persuasion framework. Then, it presents a discussion of relevant equilibrium-finding techniques. The presentation privileges the concepts that are necessary for understanding our results. The interested reader is referred to [155, 126, 136] for a broader overview of Algorithmic Game Theory and its applications.

2.1 Game and Strategy Representations

In this section, we survey the most widely adopted game models (Subsection 2.1.1) and strategies (Subsection 2.1.2) representations.

2.1.1 Extensive and Normal-Form Games

An *extensive-form game* (EFG) models a sequential interaction among players. An EFG is represented as a game tree, where each node is identified by the ordered sequence of actions (histories) leading to it from the root node. Each node represents a decision point of the game and is associated to a

single player, who has a set of available actions at that node represented by its branches. A payoff for each player is associated to each leaf node (*terminal node*) of the game tree. Finally, exogenous stochasticity is modeled via a virtual player (*a.k.a. nature* or *chance*) that plays non-strategically (*i.e.*, it plays according to a fixed strategy).

In general, a player may not be able to observe all the other players' actions, and players may have information on the state of the game which is not shared (*i.e.*, in poker each player does not know other players' hands). Imperfect information is represented via *information sets* (or *infosets*), which group together decision nodes of a certain player that are indistinguishable to her. An extensive-form game is formally defined as follows.

Definition 2.1 (Extensive-form game)**.** *An extensive-form game Γ is a tuple $(H, Z, \mathcal{P}, A, \pi_c, \mathcal{I}, \{u_i\}_{i\in\mathcal{P}})$ where:*

- *H is the set of nodes of the game, each identified by an ordered sequence of actions.*
- *$Z \subseteq H$ is the set of terminal nodes of the game.*
- *$\mathcal{P} \cup \{c\}$ is the set of players, where c denotes the chance player. For each non-terminal node $h \in H\backslash Z$, $P(h) \in \mathcal{P} \bigcup \{c\}$ is the player who acts at h. Then, let $H_i = \{h \in H | P(h) = i\}$.*
- *$A = \{A_i\}_{i\in\mathcal{P}\cup\{c\}}$ is the set of actions in the game and A_i denotes the set of actions available to player i. The set of actions available at $h \in H$ is denoted by $A(h)$.*
- *π_c is a function defining a probability distribution over $A(h)$, for each $h \in H_c$. $\pi_c(h, a)$ is the probability with which chance plays a at h.*
- *$\mathcal{I} = \{\mathcal{I}_i\}_{i\in\mathcal{P}}$ is the set of all information sets of the game, where $\mathcal{I}_i$ is the set of information sets of player i, defined as a partition over H_i. Each $I \in \mathcal{I}_i$ is such that, for all $h, h' \in I$, $A(h) = A(h')$. Let $A(I)$ be the set of actions available at each decision node in I.*
- *$u_i : Z \to \mathbb{R}$ is the payoff function of player i.*

Given a history h and action a, we denote by $h \cdot a$ the new state reached by taking action a at h. If a sequence of actions leads from h to h', then we write $h \sqsubseteq h'$ (which means that h *precedes* h'), where $h \sqsubset h' \iff h \sqsubseteq h'$ and $h \neq h'$. Then, we can formally define Z as the set $\{h \in H | \nexists h' \in H \text{ s.t. } h \sqsubset h'\}$.

An EFG is *constant-sum* if, for all $z \in Z$, $\sum_{i \in \mathcal{P}} u_i(z) = k$. When $k = 0$ the game is said to be *zero sum*. Games that do not satisfy the constant-sum condition are said *general-sum games*. Finally, we denote by Δ_u the maximum range of payoffs in the game, *i.e.*,

$$\Delta_u = \max_{i \in \mathcal{P}} \left(\max_{z \in Z} u_i(z) - \min_{z \in Z} u_i(z) \right). \tag{2.1}$$

Player i has *perfect recall* if she has perfect memory of her past actions and observations. Otherwise, the player has *imperfect recall*. Formally:

Definition 2.2. *An extensive-form game Γ has perfect recall if for every $i \in \mathcal{P}$, for every $I \in \mathcal{I}_i$, and any pair $h, h' \in I$, the sequences of infosets and actions of player i leading to h, h' is the same.*

As customary, we assume all players have perfect recall.

A *plan* for player i is a tuple σ_i that specifies an action for each infoset of that player. The set of all plans of player i is denoted by $\Sigma_i = \times_{I \in \mathcal{I}_i} A(I)$, whose dimension grows exponentially in the size of the EFG. We denote by $\sigma_i(I)$ the action selected at infoset $I \in \mathcal{I}_i$ by σ_i. Letting $\Sigma \coloneqq \times_{i \in \mathcal{P}} \Sigma_i$, we denote by $\sigma = (\sigma_i)_{i \in \mathcal{P}} \in \Sigma$ the tuple which specifies the plan chosen by each player. Analogously, for each $i \in \mathcal{P}$, we define $\Sigma_{-i} \coloneqq \times_{j \in \mathcal{P} \setminus \{i\}} \Sigma_j$ as the set of tuples σ_{-i} specifying a plan for each player other than i. Finally, $Z(\sigma_i) \subseteq Z$ is the subset of terminal nodes which are (potentially) reachable if player i plays according to $\sigma_i \in \Sigma_i$.

Any EFG can be described via a tabular representation, *i.e.*, its equivalent *normal-form* game (NFG). In a normal-form game players take decisions simultaneously. In order to properly represent an EFG through a simultaneous-moves game, players have to reason, in a single step, about their actions in the entire game tree. Formally, we have:

Definition 2.3 (Equivalent normal-form game)**.** *Given an EFG Γ, its equivalent normal-form representation is a tuple $(\mathcal{P}, \{\Sigma_i\}_{i \in \mathcal{P}}, \{U_i\}_{i \in \mathcal{P}})$, where:*

- *$\mathcal{P}$ is the set of players.*
- *For each $i \in \mathcal{P}$, Σ_i is the set of actions of player i.*
- *Payoffs functions $U_i : \Sigma \to \mathbb{R}$ are such that, for each $i \in \mathcal{P}$, $\sigma \in \Sigma$, $U_i(\sigma)$ is the expected utility observed by i in Γ if all players played according to σ, and chance played according to π_c.*

Despite their conceptual simplicity, normal-form games are not a practical representation of EFGs. This is because of the exponential (in the size of the game tree) growth of the action spaces' dimensions, which renders its applications infeasible even on game instances of small size.

2.1.2 Strategy Representations

A *strategy* is a policy prescribing how to behave to a player. We say that a strategy is *pure* if, for every decision point of the player, it prescribes a single action with certainty. Otherwise, the strategy is said to be *mixed*. Some commonly adopted strategy representations are the following:

- **Normal-form strategies**. A *normal-form strategy* x_i is a probability distribution over Σ_i. We let $\mathcal{X}_i = \Delta(\Sigma_i)$[1] be the normal-form strategy space of player i, and $\mathcal{X} = \Delta(\Sigma)$ be the set of joint probability distributions over Σ. The expected payoff of player i, when she plays x_i and the opponents play $x_{-i} \in \bigtimes_{j \in \mathcal{P} \setminus \{i\}} \mathcal{X}_j$ is denoted, with an overload of notation, by $u_i(x_i, x_{-i})$. A *normal-form strategy profile* $x \in \bigtimes_{i \in \mathcal{P}} \mathcal{X}_i$ is a tuple specifying a normal-form strategy for each player in $\mathcal{P}$.

- **Behavioral strategies**. *Behavioral strategies* are a compact (*i.e.*, with size linear in the dimension of the tree) strategy representation for EFGs. A behavioral strategy for player i is denoted by π_i, which is a vector defining a probability distribution at each player i's infoset. Given π_i, we let $\pi_i(I) \in \Delta(A(I))$ be the (sub)vector representing the probability distribution at $I \in \mathcal{I}_i$, with $\pi_i(I, a)$ denoting the probability of choosing action $a \in A(I)$. A *behavioral strategy profile* is a tuple $\pi = (\pi_i)_{i \in \mathcal{P}}$ specifying a behavioral strategy for each player of the game. The space of behavioral strategy profiles of player i is denoted by Π_i.

- **Sequence-form strategies**. The *sequence form* [110, 166, 147] of an EFG is a compact representation applicable to games with perfect recall, and frequently used in practice. It decomposes strategies into sequences of actions and their realization probabilities. A sequence q_i for player i, defined by a node h, is a tuple specifying player i's actions on the path from the root to h. We denote the set of all sequences for player i by Q_i. A sequence is said *terminal* if, together with some sequences of the other receivers, leads to a terminal node. We let $q_\varnothing$ be the fictitious sequence leading to the root node and qa the extended sequence obtained by appending action a to q. A *sequence-form strategy* (or *realization plan*) for a receiver i is a $|Q_i|$-dimensional column vector $r_i \in [0, 1]^{|Q_1|}$ such that the following two conditions hols: i) $r_i(q_\varnothing) = 1$ and ii) for each $I \in \mathcal{I}_i$ and sequence q leading to I, $-r_i(q) + \sum_{a \in A(I)} r_i(qa) = 0$. These constraints can be compactly

[1] We denote by $\Delta(X)$ the set of probability distributions over a finite set X.

written as $F_i\, r_i = f_i$, where F_i is an $(|\mathcal{I}_i| + 1) \times |Q_i|$ matrix and $f_i^{\mathsf{T}} = (1, 0, \ldots, 0)$ is a vector of dimension $(|\mathcal{I}_i| + 1)$. The first component of f_i corresponds to a fictitious infoset $I_\varnothing$ originating $q_\varnothing$. For each $I \in \mathcal{I}_i$, we denote by $Q(I) \subseteq Q_i$ the set of sequences originating in I. For each $q \in Q_i$, we denote by $I(q) \subseteq \mathcal{I}_i$ the (possibly empty) set of infosets reachable by i after selecting $q \in Q_i$, and without making other intermediate moves. The utility function of player i is represented by a sparse $|\mathcal{P}|$-dimensional matrix, marginalized with respect to π_c, defined only for profiles of terminal sequences leading to a leaf node. With an overload of notation, let $u_i \in \mathbb{R}^{|\times_{i\in\mathcal{P}} Q_i|}$ be one such matrix. The meaning of $u_i(\cdot)$ will be clear from the context. Moreover, by letting $\boldsymbol{q} \in \times_{i\in\mathcal{P}} Q_i$, $u_i(\boldsymbol{q})$ is the utility attained by player i when players choose sequences according to $\boldsymbol{q}$, and chance behaves according to π_c. A sequence-form strategy profile is a tuple $r = (r_i)_{i\in\mathcal{P}}$.

We also introduce the following notation. For any $i \in \mathcal{P} \cup \{c\}$, $h \in H$, and $\pi_i \in \Pi_i$, define

$$\rho^{\pi_i}(h) = \prod_{\substack{(I,a):P(I)=i,\\ h'\in I, h'a\sqsubseteq h}} \pi_i(I, a)$$

to be the probability that player i plays to reach history h under π_i. Then, given a behavioral strategy profile π, the *reach probability* of h under π is $\rho^{\pi}(h) = \prod_{i\in\mathcal{P}\cup\{c\}} \rho^{\pi_i}(h)$. This can be conveniently split into player i's contribution and the other players' (including chance) contribution: $\rho^{\pi}(h) = \rho^{\pi_i}(h)\rho^{\pi_{-i}}(h)$. Notice that, if player i has perfect recall, for any $h, h' \in I$, $I \in \mathcal{I}_i$, $\rho^{\pi_i}(h) = \rho^{\pi_i}(h') := \rho^{\pi_i}(I)$. Analogously, for any $i \in \mathcal{P} \cup \{c\}$, $x_i \in \mathcal{X}_i$, and $h \in H$, we define $\rho^{x_i}(h)$ to be the probability with which player i plays to reach h under x_i. Formally, by letting $\Sigma_i(h) \subseteq \Sigma_i$ be the (possibly empty) set of plans σ_i for which, if $h' \sqsubseteq h$, and $h' \in H_i$, then $h'\sigma_i(I) \sqsubseteq h$, $h' \in I$, we have $\rho^{x_i}(h) = \sum_{\sigma_i\in\Sigma_i(h)} x_i(\sigma_i)$. Finally, given a player $i \in \mathcal{P}$, $h \in H$, and realization plan r_i, by letting $q \in Q_i$ be longest sequence of actions of i preceding h, we have $\rho^{r_i}(h) = r_i(q)$. With definitions analogous to the behavioral case, we have $\rho^{x}(h)$ and $\rho^{r}(h)$. Moreover, with an abuse of notation, $\rho^{(\cdot)}(I)$ denotes the probability of reaching infoset $I \in \mathcal{I}_i$ under a given strategy/strategy profile.

We can now define a notion of equivalence between strategies. Any two strategies of a player are called *realization equivalent* if, for any fixed strategy of the other players, both strategies define the same probabilities of reaching terminal nodes. For example, x_i and x_i' are realization equivalent

if, for any x_{-i} and for any $z \in Z$, $\rho^x(z) = \rho^{x'}(z)$, where $x = (x_i, x_{-i})$, $x' = (x'_i, x_{-i})$. Similarly, two strategies x_i, x'_i are *payoff equivalent* if, for all $j \in \mathcal{P}$ and x_{-i}, $u_j(x_i, x_{-i}) = u_j(x'_i, x_{-i})$.

If player i has perfect recall, the *Kuhn Theorem* establishes that, for any normal-form strategy of player i, there exists an equivalent behavioral strategy [118]. The opposite is also true under milder assumptions: every behavioral strategy of player i has an equivalent normal-form strategy if and only if i is not *absentminded* [126, Th. 6.11].[2] The equivalence between the normal-form strategy space and the behavioral strategy space cease to hold when the player has imperfect recall. For further details on this issue see the comments in Section 5.1.

2.2 Relevant Solution Concepts

We assume players to be *rational*, *i.e.*, expected utility maximizers. Then, given an EFG Γ, $i \in \mathcal{P}$, and a fixed x_{-i}, the set $\mathsf{BR}(x_{-i})$ of *best-response* strategies of i is such that $u_i(\mathsf{BR}(x_{-i}), x_{-i}) = \max_{x_i \in \mathcal{X}_i} u_i(x_i, x_{-i})$. A *Nash equilibrium* (NE) [135] is a strategy profile in which no player can improve her utility by unilaterally deviating from her strategy.

Definition 2.4 (Nash equilibrium)**.** *$x^* = (x^*_i, x^*_{-i}) \in \times_{i \in \mathcal{P}} \mathcal{X}_i$ is a Nash equilibrium iff $u_i(x^*) = u_i(\mathsf{BR}(x^*_{-i}), x^*_{-i})$, for each $i \in \mathcal{P}$.*

Let, for any $x \in \times_{i \in \mathcal{P}} \mathcal{X}_i$, $\delta_i(x) = u_i(\mathsf{BR}(x_{-i}), x_{-i}) - u_i(x)$. If, for each $i \in \mathcal{P}$, $\delta_i(x) = 0$, then x is an NE. Nash equilibria are considered optimal in two-player, zero-sum games, as they guarantee maximal worst-case expected utility against any opponent's strategy. Otherwise, by letting $\varepsilon = \max_{i \in \mathcal{P}} \delta_i(x)$, we say that x is an ε-approximate NE (equivalently, ε-NE). In a ε-NE, no player has exploitability higher than ε. In two-player, zero-sum games, the *exploitability* of a strategy x_i is how much worst it does versus a best response compared to an NE. Formally,

$$e(x_i) = u_i(x^*_i, \mathsf{BR}(x^*_i)) - u_i(x_i, \mathsf{BR}(x_i)).$$

In multi-player settings (*i.e.*, games with $|\mathcal{P}| > 2$), a common metric to evaluate the quality of strategy profile x is $\mathsf{NashConv}(x) = \sum_{i \in \mathcal{P}} \delta_i(x)$ (see, *e.g.*, [96, 120, 158]). In two-player, zero-sum games, where $x = (x_1, x_2)$, $\mathsf{NashConv}(x) = e(x_1) + e(x_2)$.

Multi-player games often allow players for some form of coordination during the execution of the game. This is customarily model via the notion

[2] A player i is *absentminded* if there exists $I \in \mathcal{I}_i$ that can be reached more than once during the game [140, 7].

of *correlated equilibrium* (CE), introduced by Aumann [6]. When dealing with EFGs, a CE is defined as follows:

Definition 2.5 (Correlated equilibrium)**.** *A correlated equilibrium of an EFG is a probability distribution $x^* \in \mathcal{X}$ such that, for each $i \in \mathcal{P}$, and for every $\sigma_i, \sigma_i' \in \Sigma_i$, it holds:*

$$\sum_{\sigma_{-i} \in \Sigma_{-i}} x^*(\sigma_i, \sigma_{-i}) \left(U_i(\sigma_i, \sigma_{-i}) - U_i(\sigma_i', \sigma_{-i})\right) \geqslant 0.$$

A CE can be interpreted in terms of a *mediator* who, *ex ante* the play, draws the joint normal-form plan $\sigma^* \sim x^*$, $\sigma^* \in \Sigma$, and privately communicates each recommendation σ_i^* to the corresponding player.

An alternative notion of correlation is described with the *coarse correlated equilibrium* (CCE)[3] defined by [132], which enforces protection against deviations which are independent from the sampled joint normal-form plan of recommendations.

Definition 2.6 (Coarse correlated equilibrium)**.** *A coarse correlated equilibrium of an EFG is a probability distribution $x^* \in X$ such that, for every $i \in \mathcal{P}$, and every $\sigma_i' \in \Sigma_i$, it holds:*

$$\sum_{\sigma_i \in \Sigma_i} \sum_{\sigma_{-i} \in \Sigma_{-i}} x^*(\sigma_i, \sigma_{-i})(U_i(\sigma_i, \sigma_{-i}) - U_i(\sigma_i', \sigma_{-i}) \geqslant 0.$$

CCEs differ from CEs in that a CCE only requires that following the suggested plan is a best response in expectation, before the recommended plan is actually revealed. Both CEs and CCEs only require preplay communication, *i.e.*, the mediator sends signals to players *before* the beginning of the game.

2.3 Adversarial Team Games in Normal Form

We define a *team* as a set of players with coinciding objectives. Independently from the game representation, we define a team as follows:

Definition 2.7 (Team)**.** *A set of players $\mathcal{T} \subseteq \mathcal{P}$ constitutes a team if $u_i(x) = u_{i'}(x)$, for each $i, i' \in \mathcal{T}$, and for all $x \in \mathcal{X}$.*

We assume $\mathcal{T}$ to be the inclusion-wise maximal subset of players for which the previous condition holds. Moreover, given a team $\mathcal{T}$, let $U_{\mathcal{T}}$ be the normal-form utility function of an arbitrary team member. An important class of team games is the following:

[3] When defined over EFGs, CEs and CCEs are sometimes referred to as NFCEs and NFCCEs, respectively [42]. We follow this slightly different notation to facilitate the reader.

Definition 2.8 (Adversarial team games)**.** *A normal-form* adversarial team game *(ATG) is a game such that $\mathcal{P} = \mathcal{T} \cup \{\mathcal{A}\}$, where $\mathcal{T}$ is a team and $\mathcal{A}$ is the opponent (*a.k.a. *the adversary), and $U_{\mathcal{A}}(\sigma) = -|\mathcal{T}|U_{\mathcal{T}}(\sigma)$, for all $\sigma \in \Sigma$.*

In the normal-form setting, Basilico et al. [16] introduces three main coordination scenarios: i) *no-coordination*, *i.e.*, when each team member takes decisions independently from the others; ii) *non-correlated coordination*, *i.e.*, when team members can jointly select their strategies, but they cannot synchronize actions' execution; iii) *full coordination*, *i.e.*, when team members can synchronize actions' execution.

In the no-coordination scenario, players can neither jointly decide their strategies, nor coordinate actions' execution. Therefore, the appropriate solution concept is the NE, whose computation is PPAD-complete for general (*i.e.*, not two-player, zero-sum) games [53, 50].

In the non-correlated coordination scenario team members can jointly decide each $x_i \in \mathcal{X}_i$, but they are subject to the inability of correlating their actions through exogenous means of coordination (*i.e.*, they cannot synchronize their actions, which must be drawn independently). The appropriate solution concept for this setting is the *Team-maxmin equilibrium* (TME) [168], which is simply defined as the best NE for the team, when each team member $i \in \mathcal{T}$ selects a mixed strategy $x_i \in \mathcal{X}_i$ so that the minimal expected team's payoff over all possible responses of the adversary is maximized. With a slight abuse of notation, let $\mathcal{X}_{\mathcal{T}} = \times_{i \in \mathcal{T}} \mathcal{X}_i$, and $x_{\mathcal{T}} \in \mathcal{X}_{\mathcal{T}}$ be a tuple specifying a normal-form strategy for each team member. Then, $x^* \in \mathcal{X}_{\mathcal{T}} \times \mathcal{X}_{\mathcal{A}}$ is a TME if $u_{\mathcal{T}}(x^*) = \max_{x_{\mathcal{T}} \in \mathcal{X}_{\mathcal{T}}} \min_{x_{\mathcal{A}} \in \mathcal{X}_{\mathcal{A}}} u_{\mathcal{T}}(x_{\mathcal{T}}, x_{\mathcal{A}})$. This is particularly reasonable under a worst-case assumption, since the adversary may even get to know the team's profile. In an ATG, a TME always exists and it is unique, except for degeneracies [168]. Borgs et al. [25] and Hansen et al. [90] study the problem of computing *minmax strategies* in settings with one max-player and multiple min-players, which can be formulated as the problem of computing a TME in an ATG, and *vice versa*. Therefore, the following results hold: the problem of computing a TME is inapproximable in additive sense within $3/m^2$, even in three-player games with m actions per player and binary payoffs [25]. Moreover, even when the number of players is fixed, finding a TME is FNP-hard, and the problem is inapproximable in an additive sense even with binary payoffs [90]. Hansen et al. [90] also provide a quasi-polynomial-time ϵ-approximation (in additive sense) algorithm. Finally, a TME may contain irrational probabilities even with $|\mathcal{T}| = 2$ and 3 different payoff values [90].

In the last scenario (full coordination), team members can synchronize

their actions by means of a *coordination device* defined over $\times_{i\in\mathcal{T}}\Sigma_i$.[4] The team can be modeled as a single player whose action set is $\times_{i\in\mathcal{T}}\Sigma_i$. Then, in an ATG, an optimal strategy profile is a pair of maxmin/minmax strategies (played by the team player, and by the opponent, respectively). One such strategy profile is called *correlated team-maxmin equilibrium* (TMECor), and can be found by means of a linear program (LP) with a number of variables exponential in the team's size.

Basilico et al. [16] study the inefficiencies (in terms of team's expected utility) due to the lack of coordination. The Price of Anarchy (PoA) [111] of a (worst-case) NE w.r.t. the TME (*i.e.*, the best NE for the team) may be PoA $= \infty$ even in three-player games (*i.e.*, $|\mathcal{T}| = 2$), two actions per players, and binary payoffs [16, Th. 1]. Basilico et al. [16] also introduce an index, similar to the *mediation value* by Ashlagi et al. [5] and following the same rationale of the PoA, to measure the inefficiency of the TME w.r.t. the TMECor, which is called *Price of Uncorrelation* (PoU). It is defined as the ratio between team's expected utility at the TMECor and team's expected utility at the TME. A lower bound over the worst-case PoU is PoU $= m^{|\mathcal{P}|-2}$, where m is the number of actions of each player, even in games with binary payoffs [16, Th. 2]. An upper bound over the worst-case PoU is PoU $\leqslant m^{|\mathcal{P}|-2}$ [16, Th. 3]. The former result shows that the upper bound of PoU is at least $m^{|\mathcal{P}|-2}$, while the latter shows that the PoU cannot be larger than $m^{|\mathcal{P}|-2}$. Therefore, PoU is arbitrarily large only asymptotically. The proof of the latter result also provides a polynomial-time algorithm to find, given a correlated strategy, the optimal strategy profile $(x_i)_{i\in\mathcal{T}} \in \mathcal{X}_{\mathcal{T}}$ in terms of worst-case minimization of the PoU.

2.4 Bayesian Persuasion

We provide a concise description of the basic Bayesian persuasion framework, we refer the reader to the surveys by Kamenica [104] and Dughmi [58] for further details.

Bayesian persuasion, introduced by Kamenica and Gentzkow [103], generalizing a prior model by Brocas and Carrillo [28], revolves around influencing the behavior of self-interested agents through the provision of payoff-relevant information. Differently from traditional *mechanism design*, where the designer influences the outcome of the game by providing tangible incentives, in Bayesian persuasion the designer influences the outcome of the game by deciding *who gets to know what* [20].

[4] Following the terminology of Farina et al. [66], a *coordination device* is essentially a correlation device defined on the team's joint actions, and excluding the opponent from coordination.

The classical Bayesian persuasion framework by Kamenica and Gentzkow [103] comprises a single *sender* and a single *receiver*. The sender, who has access to some private information, designs a signaling scheme in order to persuade the receiver to select a favorable action, chosen among a set of A actions.

The model assumes that the sender credibly commits to the selected signaling scheme (a.k.a. the sender's *commitment assumption*). This hypothesis is realistic in many settings where reputation and credibility are a key factor for the long-term utility of the sender [144], as well as whenever an automated signaling scheme either has to abide by a contractual service agreement or it is enforced by a trusted authority [58].

In the basic model, the game is characterized by a *state of nature* $\theta \in \Theta$, which is a parameter determining the payoff function of the game. The realization of the state of nature $\theta \in \Theta$ is drawn from a publicly known prior distribution $\mu_0 \in \text{int}(\Delta(\Theta))$, where $\text{int}(X)$ is the interior set of X. The payoff for the sender and the receiver is determined by the realized state of nature θ and the selected receiver's action $a \in A$. We denote these payoffs by $u_s(\theta, a)$ and $u_r(\theta, a)$, respectively.

A *signaling scheme* (*i.e.*, an information structure) determines players' knowledge about the payoff function of the game. Specifically, a signaling scheme is a (possibly randomized) map between states of nature and *signals*. The sender must commit to a signaling scheme $\varphi : \Theta \to \Delta(\Xi)$, where Ξ denotes the set of signals available.

The interaction between the sender and the receiver goes as follows:

- The sender commits to a signaling scheme φ, which is observed by the receiver.
- The sender observes $\theta \sim \mu_0$, and then computes signal $\xi \sim \varphi_\theta$.
- The receiver observes ξ, updates her beliefs accordingly, and selects an action $a \in A$.
- The sender and the receiver observe $u_r(\theta, a)$ and $u_s(\theta, a)$, respectively.

In this setting, a result similar to the *revelation principle* (see, *e.g.*, Myerson [134]) holds. Specifically, an *optimal signaling scheme* (*i.e.*, a signaling scheme maximizing the sender's expected utility) can always be obtained by restricting the set of signals Ξ to the set of actions A (see Proposition 1 by Kamenica and Gentzkow [103]). In the following, we assume $\Xi = A$ (*i.e.*, the sender makes action recommendations).

A signaling scheme is *persuasive* if each receiver has no incentive in deviating from the recommended action. Then, the sender's optimal persuasive signaling scheme (*i.e.*, the persuasive signaling scheme maximizing sender's expected utility) can be found as the solution to LP (2.2):

$$\max_{\varphi} \sum_{\theta \in \Theta} \mu_0(\theta) \sum_{a \in A} \varphi(\theta, a) u_s(\theta, a) \tag{2.2a}$$

$$\text{s.t.} \sum_{\theta \in \Theta} \mu_0(\theta) \varphi(\theta, a) \left(u_r(\theta, a) - u_r(\theta, a')\right) \geqslant 0 \qquad \forall a, a' \in A \tag{2.2b}$$

$$\sum_{a \in A} \varphi(\theta, a) = 1 \qquad \forall \theta \in \Theta \tag{2.2c}$$

$$\varphi(\theta, a) \geqslant 0 \qquad \forall \theta \in \Theta, a \in A \tag{2.2d}$$

where constraints (2.2b) characterize the signaling scheme's persuasiveness. That is, the sender selects φ so as to maximize her expected utility, subject to φ being persuasive (constraints (2.2b)).

2.5 Equilibrium Finding Techniques

This section introduces two algorithms to compute an ε-NE in two-player, zero-sum games that will be employed in the remainder of the thesis. A further discussion of other related algorithms can be found in Section 2.6.

2.5.1 Fictitious play

Fictitious play (FP) is a game-theoretic model of learning from self-play introduced by Brown [29]. It's a type of *completely uncoupled dynamics*, *i.e.*, players interact in rounds, and each player can choose a (mixed) strategy in each round. At the end of round t, each player observes the expected payoff she would have gotten had she played each of her pure strategies against the opponent's strategy at t. Each player knows her own pure strategies but is not required to know neither the game utility matrices nor the opponent's strategies. For further details, see Fudenberg et al. [77].

The basic version of FP works on normal-form games. To simplify the exposition, we consider the case in which player i, at each t, can observe x_{-i}^t (and not just the realized action sampled according to the strategy at t). Then, player i can track of the opponent's average strategy $\bar{x}_{-i}^t = \frac{1}{t} \sum_{\tau \leqslant t} x_{-i}^\tau$. At each t, player i plays a best response against opponent's empirical distribution of play up to time $t-1$, *i.e.*, $x_i^t \in \mathsf{BR}(\bar{x}_{-i}^{t-1})$. Then, her average strategy is updated as $\bar{x}_i^t = \frac{t-1}{t} \bar{x}_i^{t-1} + \frac{1}{t} x_i^t$.

Average strategies computed via FP converge to an NE in certain classes of games, *e.g.*, two-player zero-sum games [146], and many-player potential games [130]. Specifically, in two-player zero-sum games, by letting $\delta^{\max} = \max_\sigma |U_i(\sigma)|$, it holds that, for all $\varepsilon > 0$, for all $t \geqslant \left(\frac{\delta^{\max}}{\varepsilon}\right)^{\Omega(\sum_{i\in\mathcal{P}} |\Sigma_i|)}$, $(\bar{x}_1^t, \bar{x}_2^t)$ is an ε-approximate NE. The results by Robinson [146] imply a convergence rate of $O(t^{-\frac{1}{\sum_{i\in\mathcal{P}} |\Sigma_i|-2}})$. Karlin [106] conjectured that, in two-player zero-sum games, FP converges at a rate $O(t^{-\frac{1}{2}})$, but this was recently disproved by Daskalakis and Pan [54], who show that on certain instances it may converge as slow as $\Omega(t^{-\frac{1}{|\Sigma_i|}})$ (assuming $|\Sigma_i|$ is equal for all $i \in \mathcal{P}$). In general-sum bimatrix games, FP may not converge to an equilibrium point. Specifically, Goldberg et al. [83] show that, in bimatrix games with payoffs in $[0, 1]$, FP may never get to a solution better than a 0.5-NE. Several other works are devoted to identifying classes of games in which FP converges (or not). A summary of these works can be found in surveys by Fudenberg et al. [77], Hofbauer and Sigmund [99], Krishna and Sjöström [112].

An extension of FP to imperfect-information EFG is due to Heinrich et al. [97], building on the work of Hendon et al. [98]. *Extensive-form fictitious play* (XFP) is implemented in behavioral strategies but inherits the convergence results of *generalized weakened fictitious play* [122, 17] by realization-equivalence [97, Th. 7]. XFP has been extended to work in a sample-based fashion [97], and it is the basis for deep reinforcement learning algorithms that learn ε-NE in a scalable end-to-end fashion [96].

2.5.2 Counterfactual Regret Minimization

As in the FP framework, in *online convex optimization* [179], each player i plays repeatedly against an unknown environment by making a series of decisions $x_i^1, x_i^2, \ldots, x_i^t$. In the basic setting, the decision space of player i is the whole normal-form strategy space $\mathcal{X}_i$. At iteration t, after selecting x_i^t, player i observes a utility $u_i^t(x_i^t)$. The *cumulative external regret* of player i up to iteration T is defined as

$$R_i^T = \max_{\hat{x}_i \in \mathcal{X}_i} \sum_{t=1}^{T} u_i^t(\hat{x}_i) - \sum_{t=1}^{T} u_i^t(x_i^t). \tag{2.3}$$

A *regret minimizer* is a function providing the next player i's strategy x_i^{t+1} on the basis of the past history of play and the observed utilities up to iteration t. A desirable property for regret minimizers is *Hannan consistency* [88], which requires that $\limsup_{T\to\infty} \frac{1}{T} R_i^T \leqslant 0$, *i.e.*, the cumulative regret grows at a sublinear rate in the number of iterations T.

In an EFG, the regret can be defined at each infoset. After T iterations, the cumulative regret for not having selected action $a \in A(I)$ at infoset $I \in \mathcal{I}_i$ (denoted by $R_I^T(a)$) is the cumulative difference in utility that player i would have experienced by selecting a at I instead of following the behavioral strategy π_i^t at each iteration t up to T. Then, the regret for player i at infoset $I \in \mathcal{I}_i$ is defined as $R_I^T = \max_{a \in A(I)} R_I^T(a)$. Moreover, we let $R_I^{T,+}(a) = \max\{R_I^T(a), 0\}$.

Regret matching (RM) [92] is the most widely adopted regret-minimizing scheme when the decision space is a simplex (*e.g.*, $\mathcal{X}_i$ in normal-form games), although alternatives with better theoretical performances exist (see, *e.g.*, *Hedge* [124, 76, 34]). In the context of EFGs, RM is usually applied locally at each infoset, where the player selects a distribution over available actions proportionally to their positive regret. Specifically, at iteration $T+1$ player i selects actions $a \in A(I)$ according to the following probability distribution:

$$\pi_i^{T+1}(I,a) = \begin{cases} \frac{R_I^{T,+}(a)}{\sum_{a' \in A(I)} R_I^{T,+}(a')}, & \text{if } \sum_{a' \in A(I)} R_I^{T,+}(a') > 0 \\ \frac{1}{|A(I)|}, & \text{otherwise} \end{cases}.$$

Playing according to RM at each iteration guarantees, on iteration T, $R_I^T \leqslant \frac{\sqrt{|A(I)|}}{\sqrt{T}} \Delta^{\max}$ [48]. CFR [180] is an anytime algorithm to compute ε-NEs in two-player, zero-sum EFGs. CFR minimizes the external regret R_i^T by employing RM locally at each infoset. In two-player, zero-sum games, if both players have cumulative regrets such that $\frac{1}{T} R_i^T \leqslant \varepsilon$, then their average behavioral strategies are a 2ε-NE [172].

CFR+ is a variation of classical CFR which exhibits better practical performances [162]. However, it uses alternation (*i.e.*, it alternates which player updates her regret at each iteration), which complicates the theoretical analysis to prove convergence [67, 36]. There exist first-order methods that converge at a rate of $O(T^{-1})$ [114, 116], while CFR's worst case convergence rate is in the order of $O(T^{-1/2})$. However, despite their theoretical advantages, these methods perform worse than CFR+ in practice [115]. More recent improvements on vanilla CFR are: *discounted* CFR [31], which exploits a new way of averaging strategies computed at different iterations, and *optimistic* CFR, which employs predictive regret minimizers to improve the theoretical convergence rate up to a rate of $O(T^{-3/4})$ [69].

We remark that CFR has been the foundation for many recent remarkable results in imperfect-information game solving, such as the work by-

Bowling et al. [27], Moravčík et al. [131], Brown and Sandholm [30, 32].

2.6 Discussion

We review further related works connected with ours. First, we survey other applications of models involving teams of agents. Then, we present other notions of correlated equilibria for EFGs, and known algorithmic techniques for these settings. Finally, we examine some extensions of the basic Bayesian persuasion framework by Kamenica and Gentzkow [103].

2.6.1 Previous Results on Games Involving Teams

Our work builds on the results for normal-form games by Basilico et al. [16]. However, Basilico et al. [16]'s techniques cannot be directly applied to EFGs as they directly work on the tabular representation of the game, which grows exponentially in the size of the EFG.

Game-theoretic analysis of teams in adversarial settings are limited to specific classes of games. For example, Lim [123] and Alpern and Lim [2] study the problem's mathematical derivation for rendezvous-evasion games. A number of works deal with team games without any adversarial component. We just cite a few of them for the sake of completeness: Team games were first proposed in [137] as voting games, then studied in repeated and absorbing games to understand the interaction among the players [26, 156], and more recently in Markov games with noisy payoffs [170].

In principle, many recent multi-agent reinforcement-learning techniques could be adapted to our setting (see, *e.g.*, [72, 87, 125, 159, 143]). However, they present two main issues: i) they mostly deal with fully cooperative scenarios, where there isn't any adversarial component; ii) they lack convergence guarantees to an equilibrium point (experimental evaluations are often measured in terms of *average returns*), which is crucial when considering scenarios where an opponent could know and exploit team's strategy.

2.6.2 Other Notions of Correlation

A first way to classify correlated equilibria for EFGs is by considering when recommendations are computed by the mediator. The class of *preplay correlated equilibria* comprises all solution concepts in which the entire vector of recommendations $\sigma \in \Sigma$ is drawn *before* the beginning of the playing phase of the game. The class of *intraplay correlated equilibria* comprises

those equilibria requiring lotteries to take place at each decision point of the game, *i.e.*, the mediator computes recommendations as the play proceeds, conditioned on possible *intraplay* messages coming from players. The best known intraplay correlated equilibrium is the *communication equilibrium*, which has been studied in the context of *multistage* games by Myerson [133] and Forges [73]. Here, at each stage of the game, each player privately communicates to the central mediator her private information. Then, the mediator computes recommended actions as a function of these inputs.

In our work, we focus on preplay correlated equilibria, whose most widely adopted instantiations are the CE [6, 82], the CCE [132], and the EFCE [167]. In these equilibria communication is *unidirectional* (from the mediator to players), as any *intraplay* communication coming from players would be ignored by the mediator, having already determined the recommendations. We are interested in studying correlation under the most stringent communication requirements, *i.e.*, when the mediator is allowed to exchange messages with the players just before the game starts (preplay communication). This is the case of CEs and CCEs. When considering an EFG, CEs and CCEs are defined over its equivalent reduced normal form.[5] In these solution concepts, the entire vector of recommendations is revealed before the beginning of the game (see Section 2.2). On the other hand, in an EFCE, the mediator does not reveal the whole plan to the players before the game starts, but she incrementally reveals recommendations at each infoset reached by the player. Each recommended move is only revealed when the player reaches the decision point for which the recommendation is relevant (*i.e.*, the infoset where she can make that move). Each player is free to play a move different than the recommended one, but doing so comes at the cost of future recommendations, as the mediator will immediately stop issuing recommendations to players that defect. The *agent-form correlated equilibrium* (AFCE) [74, 75] is a CE of the agent-form game[6] obtained from the given EFG. AFCE and EFCE require communication from the mediatior during game execution, as players are allowed to decide whether to deviate from the recommended action at each decision point they encounter. Recommendations have to be delivered during game execution, at each infoset reached, which makes them more demanding in terms of communication requirements than CEs and CCEs. For each $i \in \mathcal{P}$, the size of the signal (vector of recommendations) for player i that has to be sampled is the same in all preplay solution concepts, and it has a polynomial size of $|\mathcal{I}_i|$

[5] For this reason, in the context of EFGs, CE and CCE are also referred to as normal-form CE (NFCCE) and normal-form CCE (NFCCE) [42, 68].

[6] In the agent form of an EFG, moves are chosen by a different agent per infoset of each player.

(one action for each infoset). The following relation holds between the sets of equilibria described above: CE⊆EFCE⊆CCE⊆AFCE. See von Stengel and Forges [167], Farina et al. [68] for further details.

It is worth mentioning that Farina et al. [68] recently introduced the *extensive-form coarse correlated equilibrium* (EFCCE), which is the coarse equivalent of an EFCE and its set of solution is s.t. EFCE⊆EFCCE⊆CCE.

2.6.3 Computational Results for CE and CCE

The structure of EFGs causes the number of plans $|\Sigma_i|$ to grow exponentially in the size of the original tree. This is the key difficulty in the computation of optimal CEs. Specifically, it is known that finding an *optimal* (*e.g.*, social-welfare maximizing) CE is NP-hard even in two-player EFGs without chance moves [167], and in most classes of succint games [138]. An optimal EFCE can be computed in polynomial time in two-player games without chance moves, but, in games with three or more players (including chance), finding an optimal EFCE (or an optimal AFCE) is NP-hard [167]. The only known results on the complexity of computing an optimal CCE are due to Barman and Ligett [14], who analyse multi-player games of polynomial type (*i.e.*, it is feasible to enumerate pure strategies of all the players). They show that for graphical, polymatrix, congestion, and anonymous games the problem is NP-hard. The polynomial type assumption does not hold for EFGs, where, for each $i \in \mathcal{P}$, Σ_i may have exponential size.

The problem of computing *an* equilibrium point for multi-player EFGs is usually simpler than payoff maximization. An AFCE can be found in polynomial time [167, Proposition 3.13]. Huang and von Stengel [100] apply the *Ellipsoid Against Hope* approach [138, 102] to the EFCE's LP formulation, showing that an EFCE can be computed in polynomial time. The same technical caveat is exploited by Chan et al. [49] to show that, given a *multilinear* game, a CCE can be computed in polynomial time.[7] EFGs are a subclass of multilinear games. However, our complexity results (Chapter 7) cannot be derived from Chan et al. [49]'s framework, since we study the problem of computing optimal equilibrium points. We believe that finding social-welfare maximizing equilibria is fundamental to obtain credible and implementable solutions (particularly in general-sum games). Chan et al. [49]'s results do not provide any guarantee on the quality of the final solution they reach, which may be arbitrarily inefficient (in terms of social welfare) with respect to the optimal one. The complexity of computing a CE is still an open problem.

[7] A game is multilinear if its utility functions are linear in each player's strategy, when fixing other players' strategies. See [49] for a more formal definition.

The aforementioned issue (quality of the final solution) is relevant when approximating CEs/CCEs by simulating no-regret dynamics (*i.e.*, online convex programming). Regret-minimization techniques are known to converge to an ε-CCE in multi-player, general-sum normal-form games [48, 92, 93]. These techniques have been largely employed to solve two-players, zero-sum games, where social optimality is not an issue. Although price-of-anarchy analyses show that, in specific settings, coarse correlated equilibria characterizing outcomes of no-regret learning dynamics achieve near-optimal welfare [149, 94], when applied to general EFGs no guarantee on the quality of the final solution can be provided.

Regret-minimization techniques for EFGs have never been applied, to the best of our knowledge, to compute CEs and CCEs in multi-player, general-sum sequential games. CFR has been used to compute strategies in multi-player games [145, 81, 32] and general-sum games [80]. However, despite good empirical performances on the benchmarks adopted, these works are mainly experimental, since vanilla-CFR has no guarantees of converging to an NE in these settings. The only theoretical guarantee is that in two-player, general-sum EFGs, CFR (in general, any regret-minimization procedure) always eliminates strictly dominated strategies [80]. Farina et al. [70, 71] recently developed a way to approach the set of EFCEs in two-player games with no chance moves via an ad-hoc regret minimizer for the combinatorial space of correlation plans (a notion introduced by von Stengel and Forges [167]).

2.6.4 Extensions of the Bayesian Persuasion Framework

First, we highlight that real-world applications of Bayesian persuasion models are ubiquitous. For instance, this framework has been recently applied to security problems [141, 176, 177], financial-sector stress testing [84], voter coalition formation [1], election manipulation [39], and online advertisement [12, 63].

The basic Bayesian persuasion framework has been extended to take into account the multi-receiver scenario. The multiple receiver case is characterized in terms of the type of information that can be exchanged: in the *private channel* mode, different receivers may receive different information (see, *e.g.*, Kamenica and Gentzkow [103], Arieli and Babichenko [4]), in the *public channel* mode, all receivers observe the same signal (see, *e.g.*, the works by Dughmi et al. [61], Alonso and Câmara [1], Dughmi [59]). Persuasion with private signals has been explored only in very specific settings, such as two-agents two-action games [163], unanimity elections [13], vot-

ing with binary action spaces and binary states of Nature [171], and games with binary action spaces and no inter-agent externalities [11, 4, 62]. As pointed out by Dughmi [58], the problem of computing private signaling schemes in general multi-receiver settings still lacks a general algorithmic framework.

A major drawback of classical Bayesian persuasion models is that they typically assume that the receivers take their actions simultaneously [58, 104]. As most of the real-world economic interactions take place sequentially, modeling sequential decision making would allow for a greater modeling flexibility which could be exploited in the context of, *e.g.*, sequential auctions [121].

Part I

Extensive-Form Team Games

CHAPTER 3

Model and Inefficiency Bounds

In Section 2.3 we surveyed recent results on normal-form team games. In this class of games players take actions simultaneously, and this limits the coordination capabilities of team members. Indeed, when players of an ATG (see Definition 2.8) interact in a sequential game, teams' coordination capabilities are richer (as it happens when transitioning from correlated equilibria in NFGs to correlated equilibria in EFGs [74, 75]). Extensive-form ATGs are, to the best of our knowledge, unexplored in the literature. Part I of this thesis focuses precisely on this class of games.

Team members may adopt one of three coordination mechanisms: i) A *communication device* receiving *inputs* from the teammates (about the information they observe during the play), and sending them *recommendations* about the action to play at each information set. This requires both preplay and intraplay communication. ii) A *coordination device* that only exploits preplay communication by recommending a plan of actions to each team member just before the game starts. iii) Team members jointly plan their strategies, but no communication is possible before or during the game.

We will mainly concentrate on the second scenario, where only preplay communication is possible. We will often refer to it as the *ex ante* co-

ordination setting. Scenarios i) and iii) are instructive to understand the advantages of different forms of intra-team communication.

The chapter is structured as follows: Section 3.1 formally defines game models capturing the three aforementioned cases and the most suitable solution concepts. In Section 3.2, we define three inefficiency indices to compare the team's expected utility for different levels of coordination.

3.1 Model: Extensive-Form ATG

We focus on team games comprising of a single opponent and a single team of agents (Definition 2.7) which are allowed for sequential interactions. von Stengel and Koller [168] analyze zero-sum normal-form games where a single team plays against an adversary. We extend this model to introduce sequential actions. First, we formally define the class of extensive-form ATG we are considering.

Definition 3.1. *An extensive-form adversarial team game is a game Γ such that $\mathcal{P} = \mathcal{T} \cup \{\mathcal{A}, c\}$, where $\mathcal{T}$ is a team and $\mathcal{A}$ is the opponent (*a.k.a. *the adversary), and $u_{\mathcal{A}}(z) = -|\mathcal{T}|u_{\mathcal{T}}(z)$, for all $z \in Z$.*

From here on, we denote by ATG an extensive-form ATG. Moreover, let $\mathcal{I}_{\mathcal{T}} := \bigcup_{i \in \mathcal{T}} \mathcal{I}_i$, and $A_{\mathcal{T}} := \bigcup_{I \in \mathcal{I}_{\mathcal{T}}} A(I)$.

When teammates have no chance of correlating their strategies, the most appropriate solution concept is the Team-maxmin equilibrium (TME) [168]. Formally, the TME for an EFG may be compacly defined in sequence-form strategies as the sequence-form strategy profile satisfying

$$\arg\max_{(r_i)_{i \in \mathcal{T}}} \min_{r_{\mathcal{A}}} \sum_{\boldsymbol{q} \in \times_{i \in \mathcal{P}} Q_i} u_{\mathcal{T}}(\boldsymbol{q}) \prod_{i \in \mathcal{P}} r_i(\boldsymbol{q}_i). \tag{3.1}$$

By using the same arguments used by von Stengel and Forges [167] 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. However, Equation (3.1) yields a non-linear, non-convex optimization problem.

In many scenarios, teammates may exploit greater coordination capabilities. While in normal-form games these capabilities reduce to employing a correlation device à la Aumann [6], in extensive-form games we can distinguish different forms of coordination. The most-accurate level of coordination 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, as a function of their observations. This setting can be modeled by resorting to a *communication device* defined in a similar way to [73]. A weaker form of coordination is achieved when teammates can communicate only before the play (preplay communication). This setting can be modeled by resorting to a *coordination device* which is, in its principle, analogous to a mediator in normal-form CEs. We formally define these two devices as follows.

Definition 3.2. *A communication device is a triple* $(\mathcal{I}_\mathcal{T}, A_\mathcal{T}, R^{\text{Com}})$ *where* $\mathcal{I}_\mathcal{T}$ *is the set of inputs (*i.e.*, infosets) that teammates can communicate to the mediator,* $A_\mathcal{T}$ *is the set of recommendations (*i.e.*, actions), and* $R^{\text{Com}} : \mathcal{I}_\mathcal{T} \times 2^{\mathcal{I}_\mathcal{T}} \times 2^{A_\mathcal{T}} \to [0, 1]$ *is the recommendation function that associates each information set* $I \in \mathcal{I}_\mathcal{T}$ *with a probability distribution over* $A_\mathcal{T}(I)$*, as a function of information sets previously reported by teammates and of the actions recommended by the mediator in the past.*

By letting $\Sigma_\mathcal{T} = \times_{i \in \mathcal{T}} \Sigma_i$, we have:

Definition 3.3. *A coordination device is a pair* $(\Sigma_\mathcal{T}, R^{\text{Cor}})$. $R^{\text{Cor}} : \Sigma_\mathcal{T} \to [0, 1]$ *is a recommendation function such that* $R^{\text{Cor}}(\sigma_\mathcal{T})$*, with* $\sigma_\mathcal{T} \in \Sigma_\mathcal{T}$*, is the probability of recommending* $\sigma_\mathcal{T}$ *to the team.*

Notice that, while a communication device provides its recommendations drawing actions from lotteries during the game, a coordination device issues recommendations only at the beginning of the game. By resorting to these definitions, we introduce the following solution concepts.

Definition 3.4 (Team-maxmin equilibrium variations)**.** *Given a communication device—or a coordination device—for the team, a Team-maxmin equilibrium with communication device (TMECom)—or a Team-maxmin equilibrium with coordination 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 spaces)**.** *The space of lotteries over the outcomes achievable by using a communication device includes the one of the lotteries achievable by using a coordination device, that, in its turn, includes the space of the lotteries achievable without any device.*

3.2 Inefficiency Bounds for Different Coordination Strategies

In this section, we investigate whether there exist scenarios in which the lack of coordination between team members significantly impacts on the final team's expected reward.

Let v_{No}, v_{Com}, v_{Cor} be the expected utility of the team at, respectively, the TME, the TMECom and the TMECor. From Property 1, we can easily derive the following.

Property 2. *For any ATG, it holds:* $v_{\text{Com}} \geqslant v_{\text{Cor}} \geqslant v_{\text{No}}$.

In order to evaluate the inefficiencies due to the impossibility of adopting a communication or a coordination device, we resort to the concept of Price of Uncorrelation (PoU), previously introduced by Basilico et al. [16] 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 team's expected utility at the TMECor, and the team's expected utility at the TME, once all the team's payoffs are normalized in $[0, 1]$. We propose the following variations of the PoU to describe inefficiencies in EFGs.

Definition 3.5 (Inefficiency indices)**.**

- $PoU_{\text{Com/No}} = \frac{v_{\text{Com}}}{v_{\text{No}}}$,
- $PoU_{\text{Cor/No}} = \frac{v_{\text{Cor}}}{v_{\text{No}}}$,
- $PoU_{\text{Com/Cor}} = \frac{v_{\text{Com}}}{v_{\text{Cor}}}$.

In perfect-information games all the indices assume a value of 1 (*i.e.*, team members have no benefits in exchanging messages), the solution being unique, except in the case of degeneracy, as shown by backward induction. In imperfect-information games, revealing information to other team members may yield indices strictly greater than 1.

In normal-form games, the tight upper bound to PoU is $m^{|\mathcal{P}|-2}$, where m is the number of actions of each player [16]. Defining bounds as a function of m is not practical in EFGs, where each player may have a different set of actions at each infoset. Thus, the bounds are given as a function of $|Z|$ (*i.e.*, the number of terminal nodes).

We define a particular kind of player, called *spy*, which we employ to derive some of our bounds.

Definition 3.6 (Spy player)**.** *Player $i \in \mathcal{P}$ is said to be a spy if, for each $I \in \mathcal{I}_i$, $|A_i(I)| = 1$ and I is a singleton.*

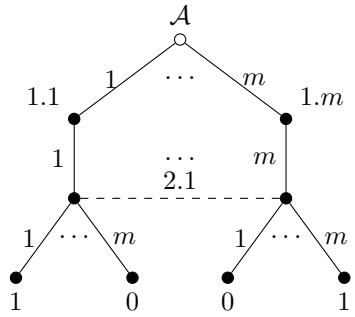

Figure 3.1: *A game with a spy used in Example 1.*

At its core, a spy is just an *observer* of the game. A spy can positively contribute to the team's final expected utility as long as she has a way to communicate and share her observations.

Suppose we have an ATG to which a team's spy (*i.e.*, the spy belongs to $\mathcal{T}$) is added so that each adversary's action is followed by a spy's infoset. Therefore, the spy can perfectly observe the choices of the adversary. The introduction of the spy does not affect team's utility in a TMECor (since $\Sigma_{\mathcal{T}}$ does not change) or in a TME, but improves the team's capabilities, and final utility, in a TMECom (where she can communicate relevant information on the actions taken by $\mathcal{A}$).

The following three examples provide lower bounds to the worst-case values of the indices, showing that the inefficiency may be arbitrarily large in $|Z|$. Table 3.1 summarizes the bounds and also highlights the worst-case bounds w.r.t. the number of players, obtained once $|Z|$ is fixed.

Inefficiency index	$n = \lvert\mathcal{P}\rvert$ is given	$n = \lvert\mathcal{P}\rvert$ is free
$PoU_{\text{Com/ No}}$	$\lvert Z\rvert^{(1-\frac{1}{n-1})}$	$\lvert Z\rvert/2$
$PoU_{\text{Cor/No}}$	$\lvert Z\rvert^{(1-\frac{2}{n})}$	$\lvert Z\rvert/4$
$PoU_{\text{Com/Cor}}$	$\lvert Z\rvert^{\frac{1}{n-1}}$	$\sqrt{\lvert Z\rvert}$

Table 3.1: *Lower bounds to the worst-case PoUs.*

Example 1 (Lower bound for worst-case $PoU_{\text{Com/No}}$)**.** *Consider an ATG with $|\mathcal{P}| = n$, and $m \geqslant 2$ actions for each player at every decision node, except for team player 1, who is a spy. The game tree is structured as follows (see Figure 3.1 for a graphic depiction with $n = 3$):*

- *The adversary $\mathcal{A}$ 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 (*e.g.*, infoset* 2.1 *in Figure 3.1);*
- $u_{\mathcal{T}}(\cdot) = 1$ *iff, for each* $i \in \mathcal{T}\backslash\{1\}$*, and for each* $I \in \mathcal{I}_i$*, the action chosen at* I *is equal to the one selected by* $\mathcal{A}$*.*

Since the spy observes $\mathcal{A}$*'s moves, we have* $v_{\text{Com}} = 1$*. On the other hand, in absence of coordination, the best strategy for each player in* $\mathcal{T}\backslash\{1\}$ *is to select actions according to a uniform distribution at each infoset, yielding* $v_{\text{No}} = m^{2-n}$*. Then,* $PoU_{\text{Com/No}} = m^{n-2}$*. Since the tree structure is such that* $|Z| = m^{n-1}$*, we obtain* $PoU_{\text{Com/No}} = |Z|^{(1-\frac{1}{n-1})}$*. Once* $|Z|$ *is fixed, the inefficiency is monotonically increasing in* n*, but* n *is upper bounded by* $n = \log_2(|Z|) + 1$ *(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*,* $PoU_{\text{Com/No}} = |Z|/2$*.*

Example 2 (Lower bound for worst-case $PoU_{\text{Cor/No}}$)**.** *Consider an ATG with* $|\mathcal{P}| = n$ *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.* $u_{\mathcal{T}}(\cdot) = 1$ *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* $|Z|$*. 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,* $|Z| = m^n$ *and thus* $PoU_{\text{Cor/No}} = |Z|^{(1-\frac{2}{n})}$*. The worst case w.r.t.* n *is reached when* $m = 2$ *and* $n = \log_2(|Z|)$*. Therefore,* $PoU_{\text{Cor/No}} = |Z|/4$*.*

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* $|Z| = m^{n-1}$ *and thus* $PoU_{\text{Com/Cor}} = |Z|^{\frac{1}{n-1}}$*. 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*,* $PoU_{\text{Com/Cor}} = \sqrt{|Z|}$*.*

CHAPTER 4

Algorithms for Extensive-Form ATGs

We study the problem of computing TME, TMECor, and TMECom in extensive-form adversarial team games. In Section 4.1 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 two-stage algorithm. In the first stage, the game is cast into an equivalent two-player game, while, in the second stage, a solution is found by linear programming. Section 4.2 focuses on the problem of computing a TMECor. When a coordination 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. Our algorithm exploits an original hybrid representation of the game combining both normal and sequence forms. The column-generation oracle's problem is shown to be APX-hard, with an upper approximation bound decreasing exponentially in the depth of the tree. We also provide an approximation algorithm for the oracle's problem, that matches certain approximation guarantees on a subset of instances. Finally, 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 via global

optimization tools (see Section 4.3)

4.1 Computing a TMECom

We show that there exists a polynomial-time TMECom-finding algorithm. Indeed, we prove that the problem of finding a TMECom is equivalent to finding a two-player maxmin strategy in an auxiliary two-player zero-sum 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 employ. Let Γ be an extensive-form game and $Q_{\text{tot}} = \times_{i\in\mathcal{P}} Q_i$. We define the following functions. Function $\mathsf{lead} : H \to 2^{Q_{\text{tot}}}$ returns the sequence profile constituting the path from the root to a given node of the tree. Function $\mathsf{path} : H \times 2^{\mathcal{P}} \to 2^{Q_{\text{tot}}}$ is s.t., for each $h \in H$ and each set of players $G \subseteq \mathcal{P}$,

$$\mathsf{path}(h|G) = \Big\{ q \subset \bigtimes_{i\in G} Q_i \Big| \exists q' \subset \bigtimes_{i\in\mathcal{P}\setminus G} Q_i \text{ and } (q,q') = \mathsf{lead}(h) \Big\}.$$

Intuitively, $\mathsf{path}(h|G)$ returns the unique profile of sequences of players in G leading to h when combined with some sequences of the players in $\mathcal{P}\setminus G$.

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

Definition 4.1 (G-observable game)**.** *For any EFG game Γ and any set of players $G \subseteq \mathcal{P}$, the G-observable game $\hat{\Gamma}$ is obtained from Γ via a modification of the information structure $\mathcal{I}$. Specifically, $\hat{\mathcal{I}} = \{\hat{\mathcal{I}}_i\}_{i\in G} \cup \{\mathcal{I}_i\}_{i\in\mathcal{P}\setminus G}$ is such that:*

1. *for each decision node $h \in H\setminus Z$, there exists one and only one $\hat{I} \in \hat{\mathcal{I}}$ s.t. $h \in \hat{I}$ and $P(I) = P(\hat{I})$, where I denotes the information set containing h in Γ;*
2. *for each player $i \in G$, $\hat{\mathcal{I}}_i$ is the set with the lowest possible cardinality s.t. for each $\hat{I} \in \hat{\mathcal{I}}_i$ and for each pair of decision nodes $h, h' \in \hat{I}$, it holds: $\mathsf{path}(h|G) = \mathsf{path}(h'|G)$ and there exists $I \in \mathcal{I}_i$ s.t. $h \in I$ and $h' \in I$.*

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 $\mathcal{P}\setminus G$. Intuitively, if an arbitrary subset

of players in a perfect-recall EFG is treated as a single player, the resulting game may end up having imperfect recall. However, in a G-observable perfect recall game, if the players in G are treated as a single player, the resulting game still has perfect recall.

We focus on the setting in which $G = \mathcal{T}$, and show that a $\mathcal{T}$-observable auxiliary game can be built in polynomial time in the size of the input.

Lemma 4.1 ($\mathcal{T}$-observable game construction)**.**
The $\mathcal{T}$-observable game $\hat{\Gamma}$ of a generic ATG Γ can be computed in polynomial time.

Proof. We provide the sketch of an algorithm to build a $\mathcal{T}$-observable game (*i.e.*, a G-observable game with $G = \mathcal{T}$) 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 values the information set id to be used in $\hat{\Gamma}$. Γ is traversed in depth-first order while keeping track of the sequence leading to the current node. For each $h \in H \backslash Z$ s.t. $P(h) \in \mathcal{T}$, a search/insertion over the first hash-table is performed by hashing $\mathtt{path}(h|\mathcal{T})$. 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 $O(|H|^2)$. □

The following result shows that an (optimal) TMECom may be found in polynomial-time.

Theorem 4.1 (TMECom computation)**.** *Given an extensive-form ATG and a communication device for $\mathcal{T}$, a TMECom can be found in polynomial time.*

Proof. Given an ATG Γ, the use of a communication device for the team $\mathcal{T}$ changes the information structure of the game inducing a $\mathcal{T}$-observable game $\hat{\Gamma}$, which can be computed in polynomial time (Lemma 4.1). A TMECom can be computed over $\hat{\Gamma}$ as follows. Given a communication device $(I_{\mathcal{T}}, A_{\mathcal{T}}, R^{\text{Com}})$, R^{Com} enforces a probability distribution γ over the set of feedback rules. The distribution γ is chosen in order to maximize the team's expected utility. In this setting, no incentive constraints are required because teammates share the same utility function and therefore, under the hypothesis that γ maximizes it in expectation, 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 $\hat{\mathcal{I}}_{\mathcal{T}} = \bigcup_{i \in \mathcal{T}} \hat{\mathcal{I}}_i$, γ reduces to a distribution over rules of type $\{\beta = (\beta^I)_{I \in \hat{\mathcal{I}}_{\mathcal{T}}} | \beta^I : \mathsf{path}(I|\mathcal{T}) \rightarrow \Delta(A(I)), \forall I \in \hat{\mathcal{I}}_{\mathcal{T}}\}$. 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 two-player maxmin problem over $\hat{\Gamma}$ between $\mathcal{A}$ and a single team player with sequences $\bigcup_{i \in \mathcal{T}} Q_i$. By construction, the team player has perfect recall. Thus, the problem can be solved by determining the maxmin solution to a two-player, zero-sum game. This can be done via a compact LP with a polynomial number of variables and constraints, by resorting to the sequence form [166]. □

4.2 Computing a TMECor

In this section, we investigate the *ex ante* coordination setting, in which only preplay communication is allowed. We initially focus on the computational complexity of the problem of computing a TMECor.

Theorem 4.2 (TMECor complexity)**.** *Computing a TMECor is FNP-hard when there are two teammates, each with an arbitrary number of information sets, and when there is an arbitrary number of teammates, each with one information set.*

The first result directly follows from the reduction presented in [167, Theorem 1.3] since the game instances used in the reduction are exactly extensive-form ATGs with $|\mathcal{T}| = 2$. The second result can be proved by adapting the reduction described in [109, 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 computing a Team-maxmin equilibrium with coordinated strategies (see Section 2.3). This latter equilibrium can be found in polynomial time in the size of the normal form, which, however, is given by $\times_{i \in \mathcal{P}} \Sigma_i$, where each Σ_i is exponentially large in the size of the tree. Here, we provide a more efficient method that can also be used in an *anytime fashion*, without requiring any exponential enumeration before the algorithm execution. In our method, we exploit a hybrid representation that, to the best of our knowledge, has not been used in previous works.

4.2.1 Hybrid Representation

In our representation, the adversary's strategies are described in sequence form, while the team plays over *jointly-reduced normal-form plans*, as for-

mally defined in the following. Given an arbitrary ATG Γ, let us denote by $\Sigma_r = \{\Sigma_{r,i}\}_{i \in \mathcal{P}}$ the set of actions of the reduced normal-form of Γ (see, *e.g.*, [108, 160, 165]), where $\Sigma_{r,i}$ is the set of reduced plans for player i. Therefore, $\times_{i \in \mathcal{T}} \Sigma_{r,i}$ is the set of joint reduced plans of the team. Let the function $\mathsf{terminal} : Q_{\mathcal{A}} \times \{\times_{i \in \mathcal{T}} \Sigma_{r,i}\} \to 2^Z$ be s.t. it returns, for a given pair $(q_{\mathcal{A}}, \sigma_{\mathcal{T}})$, the set of terminal nodes that may be reached when the adversary plays $q_{\mathcal{A}}$ and the team members, at each of their information sets, play according to $\sigma_{\mathcal{T}}$. If no terminal node is reached, $\varnothing$ is returned. In the absence of moves of Nature, the set returned by $\mathsf{terminal}$ is always a singleton. Then, we define some equivalence classes over $\times_{i \in \mathcal{T}} \Sigma_{r,i}$ by the relation $\sim$:

Definition 4.2. *The equivalence relation $\sim$ over $\times_{i \in \mathcal{T}} \Sigma_{r,i}$ is s.t., given $\sigma, \sigma' \in \times_{i \in \mathcal{T}} \Sigma_{r,i}$, $\sigma \sim \sigma'$ iff, for each $q_{\mathcal{A}} \in Q_{\mathcal{A}}$, $\mathsf{terminal}(q_{\mathcal{A}}, \sigma) = \mathsf{terminal}(q_{\mathcal{A}}, \sigma')$.*

Definition 4.3 (Jointly-reduced plans)**.** *The set of jointly-reduced plans $\Sigma_{\text{jr}} \subseteq \times_{i \in \mathcal{T}} \Sigma_{r,i}$ is obtained by picking exactly one representative from each equivalence class of $\sim$.*

The team's utility function is represented by the sparse $|Q_{\mathcal{A}}| \times |\Sigma_{\text{jr}}|$ matrix U_h. Given a pair $(q_{\mathcal{A}}, \sigma_{\text{jr}}) \in Q_{\mathcal{A}} \times \Sigma_{\text{jr}}$, a value is stored in U_h iff $\mathsf{terminal}(q_{\mathcal{A}}, \sigma_{\text{jr}}) \neq \varnothing$. When pair $(q_{\mathcal{A}}, \sigma_{\text{jr}})$ may lead to more than one terminal node, due to the presence of chance, $U_h(q_{\mathcal{A}}, \sigma_{\text{jr}})$ is obtained as the sum of such outcomes, each weighted by its reach probability $\rho^{\pi_c}(z)$. Formally, $U_h(q_{\mathcal{A}}, \sigma_{\text{jr}}) = \sum_{z \in \mathsf{terminal}(q_{\mathcal{A}}, \sigma_{\text{jr}})} \rho^{\pi_c}(z) u_{\mathcal{T}}(z)$.

Let $x_{\mathcal{T}} \in \Delta(\Sigma_{\text{jr}})$ be the team strategy over Σ_{jr}. The problem of finding a TMECor in our hybrid representation can be formulated as the following LP, which we name HYBRID-MAXMIN:

$$\begin{aligned} \underset{x_{\mathcal{T}} \in [0,1]^{|\Sigma_{\text{jr}}|}, v}{\arg\max} \quad & \sum_{I \in \mathcal{I}_{\mathcal{A}} \cup \{I_{\varnothing}\}} f_{\mathcal{A}}(I) v(I) \\ \text{s.t.} \quad & \sum_{I \in \mathcal{I}_{\mathcal{A}} \cup \{I_{\varnothing}\}} F_{\mathcal{A}}(I, q_{\mathcal{A}}) v(I) - \sum_{\sigma \in \Sigma_{\text{jr}}} U_h(q_{\mathcal{A}}, \sigma) x_{\mathcal{T}}(\sigma) \leqslant 0 \quad \forall q_{\mathcal{A}} \in Q_{\mathcal{A}} \\ & \sum_{\sigma \in \Sigma_{\text{jr}}} x_{\mathcal{T}}(\sigma) = 1 \end{aligned} \tag{4.1}$$

Where $f_{\mathcal{A}}$ and $F_{\mathcal{A}}$ are the opponent's sequence form matrices (see Section 2.1.2). The above LP is composed of $|Q_{\mathcal{A}}| + 1$ constraints (except for $x_{\mathcal{T}}(\cdot) \geqslant 0$ constraints) and an exponential number of variables $x_{\mathcal{T}}$.

Algorithm 4.1 `Hybrid Column Generation` (HCG)

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

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 LP (4.1) admits a basic optimal solution with at most $|Q_{\mathcal{A}}| + 1$ variables with strictly positive values [153]. 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), there are $|Q_{\mathcal{A}}|$ joint plans in the basis. □

4.2.2 Column Generation Algorithm

Proposition 1 shows that the NP-hardness of the problem is merely due to guessing the correct jointly-reduced plans played with strictly positive probability at the TMECor. Thus, we can avoid enumerating the entire action space Σ_{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 et al. [128]).

The hybrid column-generation approach (HCG) is described by the pseudocode in Algorithm 4.1. Intuitively, it works by progressively adding joint normal-form plans from Σ_{jr} to Σ_{cur}, and by building the hybrid utility matrix U_h along with Σ_{cur}. The algorithm receives in input the game tree Γ 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 Σ_{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. The algorithm progressively populates U_h by adding a $Q_{\mathcal{A}}$-dimensional column vector for each new plan added to Σ_{cur}. The realization plan $\overline{r}_{\mathcal{A}}$ is initialized as a realization plan equivalent to a uniform behavioral mixed strategy, *i.e.*, the adversary, at each $I \in \mathcal{I}_{\mathcal{A}}$, randomizes uniformly over $A(I)$ (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 7-10 are repeated until an optimal solution is found. Initially, σ_{br} is added to Σ_{cur} (Line 7) and players' utilities at nodes reached by $(q_{\mathcal{A}}, \sigma_{\text{br}})$ for every $q_{\mathcal{A}}$ are added to U_h. Then, the algorithm solves the maxmin (HYBRID-MAXMIN), which has $|Q_{\mathcal{A}}| + 1$ constraints and $|\Sigma_{\text{cur}}| + |\mathcal{I}_{\mathcal{A}}|$ variables, and minmax (HYBRID-MINMAX) problems restricted to Σ_{cur} (Lines 8 and 9), where the HYBRID-MINMAX is obtained via strong duality and reads:

$$\begin{aligned} \underset{r_{\mathcal{A}} \in [0,1]^{|Q_{\mathcal{A}}|}, v}{\arg\min} \quad & v \\ \text{s.t. } v - \sum_{q \in Q_{\mathcal{A}}} & U_h(q, \sigma_{\text{jr}}) r_{\mathcal{A}}(q) \geqslant 0 \quad \forall \sigma_{\text{jr}} \in \Sigma_{\text{cur}} \\ \sum_{q \in Q_{\mathcal{A}}} & F_{\mathcal{A}}(I, q) r_{\mathcal{A}}(q) = f_{\mathcal{A}}(I) \qquad \forall I \in \mathcal{I}_{\mathcal{A}} \end{aligned} \tag{4.2}$$

Finally, the algorithm calls BR-ORACLE to find the best response to $\overline{r}_{\mathcal{A}}$ (Line 10). Our algorithm repeatedly solves LP (4.1) and (4.2). Given the solution of HYBRID-MINMAX at a certain iteration, the algorithm employs an oracle to generate a new jointly-reduced plan of the team, as a best response to $\overline{r}_{\mathcal{A}}$, to be added to the LP, which is subsequently solved again. The algorithm terminates when the jointly-reduced plan generated by the oracle is already in Σ_{cur}, and therefore the objective function cannot be further improved.

4.2.3 Best-Response Oracle

Given an ATG 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 [167, 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 by showing that finding a joint normal-form plan of the team in best response to a given sequence-form strategy of the opponent is APX-hard (*i.e.*, it does not admit a PTAS, unless P=NP). For an introduction to approximation-

preserving reductions see Ausiello et al. [9].

Theorem 4.3. *BR-T is APX-hard.*

Proof. We prove that MAX-SAT is AP-reducible to BR-T (MAX-SAT $\leqslant_{AP}$ BR-T). Given a boolean formula ϕ in conjunctive normal form, MAX-SAT is the problem of determining the maximum number of clauses that can be made true by a truth assignment to variables of ϕ. For any ϕ with c clauses, we build, with a construction similar to [167, Theorem 1.3], an extensive-form ATG Γ_ϕ as follows:

- $\mathcal{P} = \{\mathcal{A}\} \cup \mathcal{T}$, and $\mathcal{T} = \{1, 2\}$;
- $\mathcal{A}$ plays first and has a unique decision node (*i.e.*, the root of the tree) with c available actions;
- Player 1 plays on the second level of the tree and has a singleton infoset for each clause in ϕ. Each infoset has, as its actions, the variables that appear in the clause it identifies;
- Player 2 plays on the third level of the tree. She has one information set for each literal of ϕ. At each of her information sets, Player 2 chooses whether the literal has to be positive or negative;
- $u_{\mathcal{T}}(\cdot) = 1$ if the literal chosen by Player 1 is true in the assignment made by Player 2.

Consider $\mathcal{A}$ to be randomizing uniformly over her actions. With this construction, Γ_ϕ admits a team's pure strategy profile leading to payoff 1 iff ϕ is satisfiable. Denote with σ_{br} the solution to BR-T for Γ_ϕ. Function $g(\phi, \Gamma_\phi, \sigma_{\text{br}})$ maps the best-response result back to a feasible assignment for the MAX-SAT problem.

Once fixed $\overline{r}_{\mathcal{A}}$ so that each terminal sequence of $\mathcal{A}$ is selected with probability $1/c$, the objective functions of MAX-SAT and BR-T are equivalent since maximizing the utility of the team implies finding the maximum number of satisfiable instances in ϕ. Denote by $\mathsf{OBJ}_{BR}(\Gamma_\phi)$ and $\mathsf{OBJ}_{MS}(\phi)$ the value of the two objective functions of BR-T and MAX-SAT, respectively. It holds $\frac{1}{c}\mathsf{OBJ}_{MS}(\phi) = \mathsf{OBJ}_{BR}(\Gamma_\phi)$. For this reason, the AP-condition holds. Specifically, for any ϕ, for any rational $\alpha > 1$, for any feasible solution σ_{br} to BR-T over Γ_ϕ, it holds:

$$\frac{\mathsf{OPT}_{BR}(\Gamma_\phi)}{\mathsf{OBJ}_{BR}(\Gamma_\phi)} \leqslant \alpha \implies \frac{\mathsf{OPT}_{MS}(\phi)}{\mathsf{OBJ}_{MS}(g(\phi, \Gamma_\phi, \sigma_{\text{br}}))} \leqslant 1 + \beta(\alpha - 1)$$

where $\mathsf{OPT}_{BR}(\cdot)$ and $\mathsf{OPT}_{MS}(\cdot)$ are, respectively, the optimal solutions to a given instance of the two problems, and $\beta = 1$. Therefore, since MAX-SAT is an APX-complete problem (see [8]) and it is AP-reducible to BR-T, BR-T is APX-hard. □

Moreover, by letting $\alpha_{(\cdot)} \in [0, 1]$ be the best approximation bound of the maximization problem $(\cdot)$, we prove the following result.

Theorem 4.4. *Denote with BR-T-t the problem BR-T over ATG instances of fixed maximum depth $3t$ and branching factor variable at each decision-node, it holds: $\alpha_{\text{BT-T-t}} \leqslant (\alpha_{\text{MAX-SAT}})^t$.*

Proof. We recall that $\alpha_{(\cdot)} \in [0, 1]$ denotes the best upper-bound for the efficient approximation of maximization problem $(\cdot)$.

Let ϕ be a boolean formula in conjunctive normal form. Build an ATG Γ_ϕ following the construction explained in the proof of Theorem 4.3. We build a new ATG starting from Γ_ϕ. Fix the maximum depth of the new ATG's tree to an arbitrary constant value $3t \geqslant 1$. At this point, for each terminal node $z_j \in Z$ of Γ_ϕ s.t. $u_{\mathcal{T}}(z_j) = 1$, replicate Γ_ϕ by substituting z_j with the root of a new subtree $\Gamma_\phi^{z_j} = \Gamma_\phi$. Repeat this procedure on the terminal nodes of the newly added subtrees until the longest path from the root of Γ_ϕ to one of the new leafs traverses t copies of the original tree. Denote the full tree obtained through this process with Γ'_ϕ. The maximum depth of Γ'_ϕ is $3t$, and it contains the set of $\{\Gamma_\phi^{z_1}, \ldots, \Gamma_\phi^{z_k}\}$ replicas of Γ_ϕ.

Suppose, by contradiction, there exists a polynomial-time approximation algorithm for BR-T-t guaranteeing a constant approximation factor $\alpha'_{\text{BR-T-t}} > (\alpha_{\text{MAX-SAT}})^t$. Apply this algorithm to find an approximate solution to BR-T-t over Γ'_ϕ. For at least one of the sub-trees in $\{\Gamma_\phi, \Gamma_\phi^{z_1}, \ldots, \Gamma_\phi^{z_k}\}$, it has to hold: $\alpha_{\text{BR-T-t}}^{z_j} > \alpha_{\text{MAX-SAT}}$, where $\alpha_{\text{BR-T-t}}^{z_j}$ is the approximation ratio obtained by the algorithm for the problem BR-T-t over $\Gamma_\phi^{z_j}$. As shown in the proof of Theorem 4.3, a solution to BR-T over a tree obtained with our construction can be mapped back to obtain an approximate solution to MAX-SAT. The same reasoning holds for BR-T-t. Therefore, if $\alpha_{\text{BR-T-t}}^{z_j} > \alpha_{\text{MAX-SAT}}$, then $\alpha_{\text{MAX-SAT}}^{z_j} > \alpha_{\text{MAX-SAT}}$, where $\alpha_{\text{MAX-SAT}}^{z_j}$ is the approximation ratio obtained approximating the MAX-SAT instance by mapping the solution of BR-T-t over $\Gamma_\phi^{z_j}$. Therefore, the approximation algorithm guarantees a constant approximation factor for MAX-SAT which is strictly greater that its theoretical upper bound. Then, we reach a contradiction. □

Theorem 4.4 implies that the upper bound on the reachable approximation factor decreases exponentially as the depth of the tree increases.[1]

[1] Notice that $\alpha_{\text{MAX-SAT}} \leqslant 7/8$, see Håstad [95].

The column-generation oracle solving BR-T can be formulated as the following mixed-integer linear program (MILP), which employs a binary variable for each terminal node of the game.

$$\begin{aligned}
&\arg\max_{(r_i)_{i\in\mathcal{T}},w\in\{0,1\}^{|Z|}} \sum_{z\in Z} \rho^{\pi_c}(z)u_{\mathcal{T}}(z)w(z)\overline{r}_{\mathcal{A}}(\mathsf{path}(z|\{\mathcal{A}\})) \\
&\qquad \text{s.t.} \sum_{q_i\in Q_i} F_i(I,q_i)r_i(q_i) = f_i(I) && \substack{\forall i\in\mathcal{T},\\ \forall I\in\mathcal{I}_i\cup\{I_\varnothing\}} \qquad (4.3)\\
&\qquad\quad w(z) \leqslant r_i(q_i) && \substack{\forall i\in\mathcal{T},\forall z\in Z,\\ \forall q_i\in\mathsf{path}(z|\{i\})}
\end{aligned}$$

MILP (4.3) produces a pure sequence-form strategy for each team member by forcing, for each $z \in Z$, the corresponding binary variable to be equal to 1 iff all team's sequences on the path to z are selected with probability 1. Specifically, $w(z)$ is a binary variable which is equal to 1 iff, for all $i \in \mathcal{T}$, and all the sequences $q_i \in Q_i$ necessary to reach z, it holds $r_i(q_i) = 1$. 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^Z the set of sequences played with probability one by i that are not subsets of any other sequence played with positive probability. Let σ'_i be the reduced normal-form plan of player i specifying all and only actions played in the sequences belonging to Q_i^Z. The joint plan $\sigma' = (\sigma'_i)_{i\in\mathcal{T}}$ is s.t. $\sigma' \in \Sigma_{\text{jr}}$.

A simple approximation algorithm can be obtained by a continuous relaxation of the binary constraints $w(z) \in \{0, 1\}$. The resulting mathematical program is linear and therefore solvable in polynomial time. An approximated solution can be obtained by randomized rounding [142]. We show that when considering game trees encoding MAX-SAT instances (see the proof of Theorems 4.3), the approximation algorithm matches the ratio guaranteed by randomized-rounding for MAX-SAT. The linear program relaxation of the MILP oracle reads:

$$\begin{aligned}
&\arg\max_{(r_i)_{i\in\mathcal{T}},w} \sum_{z\in Z} \rho^{\pi_c}(z)u_{\mathcal{T}}(z)w(z)\overline{r}_{\mathcal{A}}(\mathsf{path}(z|\{\mathcal{A}\})) \\
&\qquad \text{s.t.} \sum_{q_i\in Q_i} F_i(I,q_i)r_i(q_i) = f_i(I) && \substack{\forall i\in\mathcal{T},\\ \forall I\in\mathcal{I}_i\cup\{I_\varnothing\}} \qquad (4.4)\\
&\qquad\quad w(z) \leqslant r_i(q_i) && \substack{\forall i\in\mathcal{T},\forall z\in Z,\\ \forall q_i\in\mathsf{path}(z|\{i\})}\\
&\qquad\quad 0 \leqslant w(z) \leqslant 1 && \forall z \in Z
\end{aligned}$$

Let $((r_i^*)_{i\in\mathcal{T}}, w^*)$ be an optimal solution to the LP relaxation (4.4). We select the approximate best-response, which has to be a pure realization

plan for each player, by selecting actions according to probabilities specified by $(r_i^*)_{i \in \mathcal{T}}$. Notice that, once an action has been selected, probability values at the next decision-node of the team have to be rescaled so that they sum to one (therefore, the rounding process starts from the root and proceeds downwards).

Let us focus on games encoding MAX-SAT instances. Specifically, denote with ϕ a generic boolean formula in conjunctive normal form and with Γ_ϕ the ATG built as specified in the proof of Theorem 4.3. It is interesting to notice that, for any ϕ, the application of our approximation algorithm to Γ_ϕ guarantees the same approximation ratio of randomized rounding applied to the relaxation of the ILP formulation for MAX-SAT. Specifically, let $\mathtt{A}^R_{\text{BR-T}}$ and $\mathtt{A}^R_{\text{MAX-SAT}}$ denote the approximate algorithms based on randomized rounding for BR-T and MAX-SAT, respectively. The following result holds:

Proposition 2. *For any ϕ, the approximation ratio for MAX-SAT over ϕ obtained by the solution of BR-T over Γ_ϕ via $\mathtt{A}^R_{\text{BR-T}}$ is guaranteed to be at least $(1 - 1/e)$,* i.e.*, the ratio guaranteed by $\mathtt{A}^R_{\text{MAX-SAT}}$.*

Proof. The relaxation of the MAX-SAT ILP ($\mathtt{A}^R_{\text{MAX-SAT}}$) is the following linear formulation:

$$
\begin{aligned}
&\max \sum_{c \in C_\phi} v_c \\
&\text{s.t. } v_c \leqslant \sum_{i \in \mathcal{P}_c} y_i + \sum_{i \in \mathcal{N}_c} (1 - y_i) && c \in C_\phi \\
&\qquad 0 \leqslant y_i \leqslant 1 && \forall i \in L_\phi \\
&\qquad 0 \leqslant z_c \leqslant 1 && \forall c \in C_\phi
\end{aligned}
$$

where C_ϕ is set of clauses of ϕ, L_ϕ is the set of literals of ϕ, $\mathcal{P}_c$ and $\mathcal{N}_c$ are the sets of literals appearing in clause c non-negated or negated, respectively, and y_i is the probability of setting literal i to true.

Consider a game Γ_ϕ encoding a generic ϕ. If we apply the relaxation of the best-response oracle to Γ_ϕ, $\mathtt{A}^R_{\text{BR-T}}$ and $\mathtt{A}^R_{\text{MAX-SAT}}$ are equivalent. To see this, first let Player 2 determine her realization plan r_2^*. Once r_2^* has been fixed, Player 1 has, at each of her information sets $I \in \mathcal{I}_1$, a fixed expected utility v_a associated with each of her available action $a \in A(I)$. Let $\{a_1, \ldots, a_k\}$ be the set of available actions at one of the information sets of Player 1. There are three possible cases:

- $\sum_{a \in \{a_1, \ldots, a_k\}} v_a < 1$. In this case Player 1 selects, for each a, a probability $p(a) \geqslant v_a$.

- $\exists a \in \{a_1, \ldots, a_k\}, \upsilon_a = 1$. In this case, playing a with probability 1 guarantees Player 1 to satisfy the corresponding clause.
- $\sum_{a \in \{a_1,\ldots,a_k\}} \upsilon_a \geqslant 1$ and $\forall a \in \{a_1, \ldots, a_k\}, \upsilon_a < 1$. In this case, a_1 is selected with probability $p(a_1) = \upsilon_{a_1}$, a_2 is selected with probability $p(a_2) = \min\{1 - p(a_1), \upsilon_{a_2}\}$ and so on. The resulting strategy profile guarantees expected utility one for the corresponding clause.

Therefore, the value reachable in each clause is determined only by the choice of Player 2, *i.e.*, the final utility of the team depends only on r_2^*. Being the objective functions of the two formulations equivalent, the relaxed oracle enforces the same probability distribution over literals' truth assignments. That is, the optimal values of r_2^* and y_i^* are equivalent. Notice that, in these game instances, Player 2 plays only on one level and we can sample a solution to MAX-SAT according to r_2^* as if it was y_i^*. Therefore, randomized rounding of r_2^* leads to the same approximation guarantee of $\mathsf{A}^R_{\text{MAX-SAT}}$, *i.e.*, $(1 - 1/e)$ [175]. □

4.3 Computing a TME

Finding a TME in a normal-form ATGs is NP-hard [90, 16]. Then, it is easy to see that the following result holds.

Theorem 4.5 (TME complexity)**.** *Computing a TME of an extensive-form ATG is FNP-hard, and its value is inapproximable in additive sense even with binary payoffs.*

The problem of finding a TME can be compactly formulated as the following non-linear mathematical programming problem, which exploits the sequence-form of the game:

$$
\begin{aligned}
\max_{(r_i)_{i\in\mathcal{T}}} \quad & v(I_\varnothing) \\
\text{s.t.} \quad & \sum_{I \in \mathcal{I}_\mathcal{A} \cup \{I_\varnothing\}} F_\mathcal{A}(I, q_\mathcal{A}) v(I) \leqslant \\
& \qquad \leqslant \sum_{q_\mathcal{T} \in Q_\mathcal{T}} \left(u_\mathcal{T}(q_\mathcal{T}, q_\mathcal{A}) \prod_{i \in \mathcal{T}} r_i(q_{\mathcal{T},i}) \right) \qquad \forall q_\mathcal{A} \in Q_\mathcal{A} \qquad (4.6) \\
& \sum_{q_i \in Q_i} F_i(I, q_i) r_i(q_i) = f_i(I) \qquad \substack{\forall i \in \mathcal{T}, \\ \forall I \in \mathcal{I}_i \cup \{I_\varnothing\}} \\
& r_i(q_i) \geqslant 0 \qquad \substack{\forall i \in \mathcal{T}, \\ \forall q_i \in Q_i}
\end{aligned}
$$

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

CHAPTER 5

Learning a TMECor

We focus on the setting in which a team can exploit a coordination device via preplay communication, when playing against an opponent. Consider, as an illustration, the case of a poker game with three or more players, where all but one of them collude against an identified target player and will share the winnings after the game. In other settings, players might be forced to cooperate by the nature of the interaction itself. This is the case, for instance, in the card-playing phase of Bridge, where a team of two players, called the *defenders*, plays against a third player, the *declarer*. Situations of a team ganging up on a player are, of course, ubiquitous in many non-recreational applications as well, such as war where the colluders do not have time or means of communicating during battle, collusion in bidding where communication during the auction is illegal, coordinated swindling in public, and so on.

Even without communication *during* the game, the planning phase gives the team members an advantage: for instance, the team members could skew their strategies to use certain actions to signal about their state (for example, that they have particular cards). *Ex ante* coordination (*i.e.*, playing a TMECor) can enable the team members to obtain significantly higher utility (up to a factor linear in the number of the game-tree leaves) than the

utility they would obtain by abstaining from coordination (*i.e.*, adopting a TME) [16, 40].

The possibility for the team members to communicate before game play—that is, coordinate their strategies *ex ante* —makes the use of behavioral strategies unsatisfactory, as we remark in Section 5.1. The reasons for this are closely related to the fact that the team can be represented as a single meta-player, which typically has imperfect recall. This is because team members observe different aspects of the play (opponent's moves, each others' moves, and chance's moves) and cannot communicate during the game. Then, solving the game amounts to computing an NE in normal-form strategies in a two-player zero-sum imperfect-recall game. The focus on normal-form strategies is crucial. Indeed, it is known that behavioral strategies, that provide a compact representation of the players' strategies, cannot be employed in imperfect-recall games without incurring a loss of expressiveness [140]. Some imperfect-recall games do not even have any NE in behavioral strategies [174]. Even when an NE in behavioral strategies exists, its value can be up to a linear factor (in the number of the game-tree leaves) worse than that of an NE in normal-form strategies. For these reasons, recent efficient techniques for approximating maxmin behavioral strategy profiles in imperfect-recall games [46, 47] are not applicable to our domain.

Structure of the Chapter. We propose a new game representation (Section 5.2), which we call the *realization form*. In perfect-recall games it essentially coincides with the sequence form, but, unlike the sequence form, it can also be used in imperfect-recall games. Then, we use it to derive an *auxiliary game* that is equivalent to the original one (Section 5.3). It provides a sound way to map the problem of finding an optimal ex-ante-coordinated strategy for the team to the well-understood Nash equilibrium-finding problem in a (larger) two-player zero-sum perfect-recall game. Finally, in Section 5.4, by reasoning over the auxiliary game, we devise an anytime algorithm, *fictitious team-play*, that is guaranteed to converge to an optimal coordinated strategy for the team against an optimal opponent.

5.1 Remarks on the TMECor

In the setting of *ex ante coordination*, team members have the opportunity to discuss tactics before the game begins, but are otherwise unable to communicate during the game, except via publicly-observed actions (*i.e.*, only preplay communication is allowed). A powerful, game-theoretic way to think about *ex ante* coordination is through a *coordination device* (Defini-

tion 3.3). We briefly recall here its core idea: in the planning phase before the game starts, the team members identify a set of joint pure normal-form plans. Then, just before the play, the coordination device will randomly draw one of the normal-form plans from a given probability distribution, and the team members will all act as specified in the selected plan. A TMECor is an NE where team members play *ex ante* normal-form strategies. In an approximate version, ε-TMECor, neither the team nor the opponent can gain more than ε by deviating from their strategy, assuming that the other does not deviate.

In the following, to simplify the exposition, we assume $\mathcal{P} = \{\mathcal{A}, 1, 2\}$, and $\mathcal{T} = \{1, 2\}$. By sampling a recommendation from a joint probability distribution over $\Sigma_1 \times \Sigma_2$, the coordination device introduces a correlation between the strategies of the team members that is otherwise impossible to capture using behavioral strategies. In other words, in general there may be no behavioral strategy for the team player that is realization-equivalent to the normal-form strategy induced by the coordination device, as the following example further illustrates.

Example 4. *Consider the zero-sum game in Figure 10.3. Two team members (Players 1 and 2) play against an adversary $\mathcal{A}$. The team obtains a cumulative payoff of 2 when the game ends at ① or ⑧, and a payoff of 0 otherwise. A valid* ex ante *coordination device is as follows: the team members toss an unbiased coin; if heads comes up, Player 1 will play action A and Player 2 will play action C; otherwise, Player 1 will play action B and Player 2 will play action D. The realization induced on the leaves is such that $\rho(①) = \rho(⑧) = 1/2$ and $\rho(ⓘ) = 0$ for $i \notin \{1, 8\}$. No behavioral strategy for the team members is able to induce the same realization. This coordination device is enough to overcome the imperfect information of Player 2 about Player 1's move, as Player 2 knows what action will be played by Player 1 even though Player 2 will not observe it during the game.*

One might wonder whether there is value in forcing the coordination device to only induce normal-form strategies for which a realization equivalent tuple of behavioral strategies (one for each team member) exists. Indeed, under such a restriction, the problem of constructing an optimal coordination device would amount to finding the optimal tuple of behavioral strategies (one for each team member) that maximizes the team's utility. This solution concept is precisely the *team-maxmin equilibrium (TME)*, as defined in Section 3.1. TME offers conceptual simplicity that unfortunately comes at a high cost. First, finding the best tuple of behavioral strategies

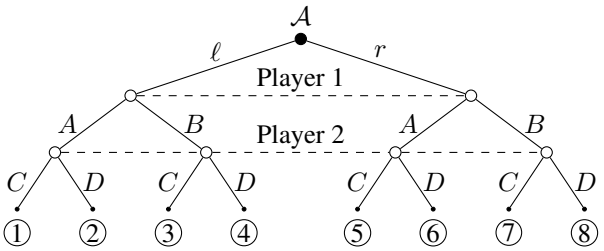

Figure 5.1: *Example of extensive-form ATG. The uppercase letters denote the action names. The circled numbers uniquely identify the terminal nodes.*

is a non-linear, non-convex optimization problem. Moreover, restricting to TMEs is also undesirable in terms of final utility for the team, since it may incur in an arbitrarily large loss compared to a TMECor (see Example 2). Interestingly, as we will prove in Section 5.2.2, there is a strong connection between TME and TMECor. The latter solution concept can be seen as the natural "convexification" of the former, in a sense that we will make precise in Theorem 5.1.

5.2 The *Realization Form*

In this section, we introduce the *realization form* of a game, which enables one to represent the strategy space of a player by a number of variables that is linear in the game size (as opposed to exponential as in the normal form), even in games with imperfect recall. For each player i, a *realization-form strategy* is a vector that specifies the probability with which i plays to reach the different terminal nodes. The mapping from normal-form strategies to realization-form strategies allows us to compress the action space from $\mathcal{X}_i$, which has as many coordinates as the number of normal-form plans (*i.e.*, $|\Sigma_i|$)—usually exponential in the size of the tree—to a space that has one coordinate for each terminal node Z. This mapping is many-to-one because of the redundancies in the normal-form representation. Given a realization-form strategy, all the normal-form strategies that induce it are payoff equivalent (see Section 2.1.2).

The construction of the realization form relies on the following observation.

Observation 1. *Let Γ be a game and $z \in Z$ be a terminal node. Given a normal-form strategy profile $x = (x_i)_{i \in \mathcal{P}} \in \times_{i \in \mathcal{P}} \mathcal{X}_i$, the probability of reaching z can be uniquely decomposed as the product of the contributions*

of each individual player, plus chance's contribution. Formally,

$$\rho^x(z) = \rho^{x_c}(z) \prod_{i \in \mathcal{P}} \rho^{x_i}(z).$$

Definition 5.1 (Realization function)**.** *Let Γ be an EFG. The* realization function *of player $i \in \mathcal{P}$ is the function $f_i^\Gamma : \mathcal{X}_i \to [0,1]^{|Z|}$ that maps every normal-form strategy for player i to the corresponding vector of realizations for each terminal node: $f_i^\Gamma : \mathcal{X}_i \ni x_i \mapsto (\rho^{x_i}(z_1), \dots, \rho^{x_i}(z_{|Z|}))$.*

We are interested in the range of f_i^Γ, called the *realization polytope* of player i.

Definition 5.2 (Realization polytope and strategies)**.** *Player i's* realization polytope Ω_i^Γ *in game Γ is the range of f_i^Γ, that is the set of all possible realization vectors for player i: $\Omega_i^\Gamma := f_i^\Gamma(\mathcal{X}_i)$. We call an element $\omega_i \in \Omega_i^\Gamma$ a* realization-form strategy *(or, simply, realization) of player i.*

The function that maps a tuple of realization-form strategies, one for each player, to the payoff of each player, is multilinear. This is by construction and follows from Observation 1. Moreover, the realization function has the following strong property.

Lemma 5.1. *f_i^Γ is a linear function and Ω_i^Γ is a convex polytope.*

Proof. We start by proving that f_i is linear. Fix a terminal node $z \in Z$. Given a normal-form strategy $x_i \in \mathcal{X}_i$, the contribution of player i to the probability of the game ending in z is computed as

$$\rho^{x_i}(z) = \sum_{\sigma_i \in \Sigma_i(z)} x_i(\sigma_i),$$

which is linear in x_i.

Now, we show that Ω_i^Γ is a convex polytope. By definition, $\Omega_i^\Gamma = f_i(\mathcal{X}_i)$ is the image of a convex polytope under a linear function, and therefore it is a convex polytope itself. □

For players with perfect recall, the realization form is the projection of the sequence form, where variables related to non-terminal sequences are dropped. In other words, when the perfect-recall property is satisfied, it is possible to move between the sequence-form and the realization-form representations by means of a simple linear transformation. Therefore, the realization polytope of perfect-recall games can be described with a linear number (in the game size) of linear constraints. Conversely, in games with

imperfect recall the number of constraints required to describe the realization polytope may be exponential. A key feature of the realization form is that it can be applied to both settings without any modification. For example, an optimal NE in a two-player zero-sum game, with or without perfect recall and/or information, can be computed through the bilinear saddle-point problem:

$$\max_{\omega_1 \in \Omega_1^\Gamma} \min_{\omega_2 \in \Omega_2^\Gamma} \omega_1^\top U \omega_2,$$

where U is a (diagonal) $|Z| \times |Z|$ *payoff* matrix.

Finally, the realization form of a game is formally defined as follows.

Definition 5.3 (Realization form)**.** *Given an extensive-form game Γ, its realization form is a tuple $(\mathcal{P}, Z, U, \Omega^\Gamma)$, where Ω^Γ specifies a realization polytope for each $i \in \mathcal{P}$.*

5.2.1 Two Examples of Realization Polytopes

To illustrate the realization-form construction, we consider two three-player zero-sum extensive-form games with perfect recall, where a team composed of two players plays against the third player. As already observed, since the team members have the same incentives, the team as a whole behaves as a single meta-player with (potentially) imperfect recall. As we show in Example 5, ex-ante coordination allows team members to behave as a single player with perfect recall. In contrast, in Example 6, the signaling power of *ex ante* coordinated strategies is not enough to fully reveal team members' private information.

Example 5. *Consider the extensive-form game depicted in Figure 10.3. $\mathcal{X}_\mathcal{T}$ is the 4-dimensional simplex corresponding to the space of probability distributions over the set of pure normal-form plans*

$$\Sigma_\mathcal{T} = \{AC, AD, BC, BD\}.$$

*Given $x \in \mathcal{X}_\mathcal{T}$, the probability with which $\mathcal{T}$ plays to reach a certain outcome is the sum of every $x(\sigma)$ such that plan $\sigma \in \Sigma_\mathcal{T}$ is consistent with the outcome (*i.e.*, the outcome is reachable if $\mathcal{T}$ plays σ). In the example, we have:*

$$f_\mathcal{T}(x) = \Big(x(A,C),\, x(A,D),\, x(B,C),\, x(B,D),\\ x(A,C),\, x(A,D),\, x(B,C),\, x(B,D) \Big),$$

$$\begin{cases} \omega(⑤) + \omega(⑥) + \omega(⑦) + \omega(⑧) = 1, \\ \omega(②) = \omega(⑥), \quad \omega(④) = \omega(⑧), \\ \omega(①) = \omega(⑤), \quad \omega(③) = \omega(⑦), \\ \omega(ⓘ) \geqslant 0 \quad i \in \{1,2,3,4,5,6,7,8\}. \end{cases}$$

Polytope 5.1: *Description of the realization polytope for the game of Figure 10.3.*

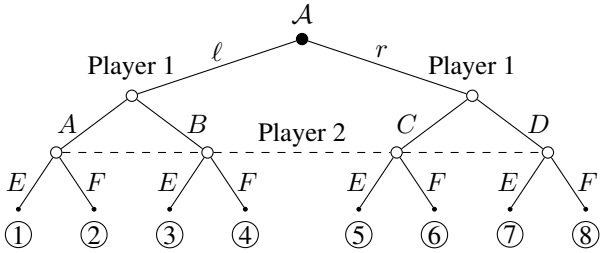

Figure 5.2: *A game where coordinated strategies have a weak signaling power. The uppercase letters denote the action names. The circled numbers uniquely identify the terminal nodes.*

where outcomes are ordered from left to right in the tree. Then, the realization polytope is described by Polytope 5.1. These constraints show that Player $\mathcal{T}$ has perfect recall when employing coordinated strategies. Indeed, the constraints coincides with the sequence-form constraints obtained when splitting Player 2's information set into two information sets, one for each action $\{A, B\}$.

Example 6. *In the game in Figure 5.2, the team Player $\mathcal{T}$ has imperfect recall even when coordination is allowed. In this case, the signaling power of* ex ante *coordinated strategies is not enough for Player 1 to propagate the information observed (that is, $\mathcal{A}$'s move) to Player 2. Table 5.1 shows the value of the realization function, evaluated in each pure normal-form plans of Player $\mathcal{T}$. Each row is a realization vector, and the realization polytope $\Omega^{\Gamma}_{\mathcal{T}}$ is the convex hull of all these vectors. Therefore, the realization polytope $\Omega^{\Gamma}_{\mathcal{T}}$ is characterized by the set of constraints in Polytope 5.2. As one might expect, this polytope contains Polytope 5.1.*

5.2.2 Relationship between TMECor and TME

In this subsection we study the relationship between the TMECor and the TME, and prove a fact of potential independent interest. Specifically, we

	①	②	③	④	⑤	⑥	⑦	⑧
ACE	1	0	0	0	1	0	0	0
ACF	0	1	0	0	0	1	0	0
ADE	1	0	0	0	0	0	1	0
ADF	0	1	0	0	0	0	0	1
BCE	0	0	1	0	1	0	0	0
BCF	0	0	0	1	0	1	0	0
BDE	0	0	1	0	0	0	1	0
BDF	0	0	0	1	0	0	0	1

Table 5.1: *Example 6: Mapping between pure normal-form plans and their images under the realization function.*

$$\begin{cases} \omega(⑤) + \omega(⑥) + \omega(⑦) + \omega(⑧) = 1, \\ \omega(②) + \omega(④) = \omega(⑥) + \omega(⑧), \\ \omega(①) + \omega(③) = \omega(⑤) + \omega(⑦), \\ \omega(ⓘ) \geqslant 0 \quad i \in \{1,2,3,4,5,6,7,8\}. \end{cases}$$

Polytope 5.2: *Description of the realization polytope for the game of Figure 5.2.*

prove that the realization polytope of a *non-absentminded* player (see Piccione and Rubinstein [140]) is the convex hull of the set of realizations that are reachable starting from behavioral strategies. This gives a precise meaning to our claim that *the TMECor concept is the convexification of the TME concept*.

Definition 5.4. *Let Γ be an EFG. The* behavioral-realization function *of player i is the function $\tilde{f}_i^\Gamma : \Pi_i \ni \pi_i \mapsto (\rho^{\pi_i}(z_1), \ldots, \rho^{\pi_i}(z_{|Z|})) \in [0,1]^{|Z|}$. Accordingly, the* behavioral-realization set *of player i is the range of $\tilde{f}_i^\Gamma$, that is $\tilde{\Omega}_i^\Gamma := \tilde{f}_i^\Gamma(\Pi_i)$. This set is generally non-convex.*

Denoting by $\mathrm{co}(\cdot)$ the convex hull of a set, we have the following:

Theorem 5.1. *Consider a game Γ. If player i is not absent-minded, then $\Omega_i^\Gamma = \mathrm{co}\big(\tilde{\Omega}_i^\Gamma\big)$.*

Proof.

($\subseteq$) We know as a direct consequence of Lemma 5.1 that $\Omega_i^\Gamma = \mathrm{co}\{f_i(\sigma) : \sigma \in \Sigma_i\}$. Since every pure normal-form plan is also a behavioral strategy, $f_i(\sigma) \in \tilde{\Omega}_i^\Gamma$ for all $\sigma \in \Sigma_i$. Hence, $\Omega_i^\Gamma = \mathrm{co}\{f_i(\sigma) : \sigma \in \Sigma_i\} \subseteq \mathrm{co}(\tilde{\Omega}_i^\Gamma)$.

($\supseteq$) Finally, we prove that that $\Omega_i^\Gamma \supseteq \mathrm{co}(\tilde{\Omega}_i^\Gamma)$. Since Ω_i^Γ is convex, it is enough to show that $\Omega_i^\Gamma \supseteq \tilde{\Omega}_i^\Gamma$. In other words, it is enough to prove

that every behavioral-realization is also a realization in the sense of Definition 5.2, provided that player i is not absent-minded. This is a well-known fact, and we refer the reader to Theorem 6.11 in the book by Maschler et al. [126].

□

5.3 *Auxiliary Game:* an Equivalent Game that Enables the Use of Behavioral Strategies

We describe our construction for three-player zero-sum extensive-form games with perfect recall, and we will model the interaction of a team composed of two players playing against the third player. The theory developed also applies to settings with teams with an arbitrary number of players.

We prove that it is possible to construct an *auxiliary game* with the following properties:

- it is a two-player perfect-recall game between the adversary $\mathcal{A}$ and a *team-player* $\mathcal{T}$;
- for both players, the set of behavioral strategies is as "expressive" as the set of the normal-form strategies in the original game (*i.e.*, in the case of the team, the set of strategies that team members can achieve through a coordination device).

To accomplish this, we introduce a root node ϕ, whose branches correspond to the normal-form strategies of the first player of the team. This representation enables the team to express any probability distribution over the ensuing subtrees, and leads to an equivalence between the behavioral strategies in this new perfect-recall game (the auxiliary game) and the normal-form strategies of the original two-player imperfect-recall game between the team player $\mathcal{T}$ and the opponent. The auxiliary game is a perfect-recall representation of the original imperfect-recall game such that the expressiveness of behavioral (and sequence-form) strategies is increased to match the expressiveness of normal-form strategies in the original game.

Consider an ATG Γ with $\mathcal{P} = \{1, 2, \mathcal{A}\}$, where 1 and 2 are team members. We will refer to Player 1 as the *pivot player*. For any $\sigma_1 \in \Sigma_1$, we define Γ_{σ_1} as the two-player game with $\mathcal{P}' = \{2, \mathcal{A}\}$ that we obtain from Γ by fixing the choices of Player 1 as follows: $\forall I \in \mathcal{I}_1$ and $\forall a \in A(I)$, if $a = \sigma_1(I)$, then $\pi_{1,\sigma_1}(I, a) = 1$; otherwise, $\pi_{1,\sigma_1}(I, a) = 0$. Once π_{1,σ_1} has been fixed in Γ_{σ_1}, decision nodes belonging to Player 1 can be considered as if they were chance nodes. The auxiliary game of Γ, denoted with Γ^*, is defined as follows.

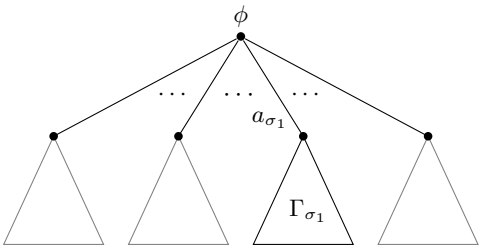

Figure 5.3: *Structure of the auxiliary game* Γ^*.

Definition 5.5 (Auxiliary Game)**.** *The auxiliary game* Γ^* *is a two-player game obtained from* Γ *in the following way:*

- $\mathcal{P} = \{\mathcal{T}, \mathcal{A}\}$*;*
- *the root* ϕ *is a decision node of Player* $\mathcal{T}$ *with* $A(\phi) = \{a_{\sigma_1}\}_{\sigma_1 \in \Sigma_1}$*;*
- *each* a_{σ_1} *is followed by a subtree* Γ_{σ_1}*;*
- $\mathcal{A}$ *does not observe the action chosen by* $\mathcal{T}$ *at* ϕ*.*

By construction, all the decision nodes of any information set of team $\mathcal{T}$ are part of the same subtree Γ_{σ_1}. Intuitively, this is because, in the original game, team members jointly pick an action from their joint probability distribution and, therefore, every team member knows what the other member is going to play. The opponent has the same number of information sets both in Γ and Γ^*. This is because she does not observe the choice at ϕ and, therefore, her information sets span across all subtrees Γ_{σ_1}. The basic structure of the auxiliary game tree is depicted in Figure 5.3 (information sets of $\mathcal{A}$ are omitted for clarity). Games with more than two team members can be represented via a Γ^* which has a number of subtrees equal to the Cartesian product of the normal-form plans of all team members except one.

The next lemma is fundamental to understand the equivalence between behavioral strategies of Γ^* and normal-form strategies of Γ. Intuitively, it justifies the introduction of the root node ϕ, whose branches correspond to the normal-form strategies of the pivot player. This representation enables the team $\mathcal{T}$ to express any convex combination of realizations in the Γ_{σ_1} subtrees.

Lemma 5.2. *For any* Γ, $\Omega_{\mathcal{T}}^{\Gamma} = \mathrm{co}\left(\bigcup_{\sigma\in\Sigma_1} \Omega_{\mathcal{T}}^{\Gamma_\sigma}\right)$.

Proof.

($\supseteq$) We start by proving that, for all $\sigma_1 \in \Sigma_1$, $\Omega_{\mathcal{T}}^{\Gamma_{\sigma_1}} \subseteq \Omega_{\mathcal{T}}^{\Gamma}$. Indeed, as a direct consequence of Lemma 5.1,

$$\begin{aligned}\Omega_{\mathcal{T}}^{\Gamma_{\sigma_1}} &= \mathrm{co}\left(\{f_{\mathcal{T}}^{\Gamma}(\sigma_1,\sigma_2) : \sigma_2 \in \Sigma_2\}\right)\\ &\subseteq \mathrm{co}\left(\{f_{\mathcal{T}}^{\Gamma}(\sigma_1',\sigma_2) : \sigma_1' \in \Sigma_1, \sigma_2 \in \Sigma_2\}\right)\\ &= \Omega_{\mathcal{T}}^{\Gamma}.\end{aligned}$$

Thus,

$$\bigcup_{\sigma_1\in\Sigma_1} \Omega_{\mathcal{T}}^{\Gamma_{\sigma_1}} \subseteq \Omega_{\mathcal{T}}^{\Gamma},$$

and therefore, using the monotonicity of the convex hull function,

$$\mathrm{co}\left(\bigcup_{\sigma_1\in\Sigma_1} \Omega_{\mathcal{T}}^{\Gamma_{\sigma_1}}\right) \subseteq \mathrm{co}(\Omega_{\mathcal{T}}^{\Gamma}) = \Omega_{\mathcal{T}}^{\Gamma},$$

where the last equality holds by convexity of $\Omega_{\mathcal{T}}^{\Gamma}$ (Lemma 5.1).

($\subseteq$) Take $\omega \in \Omega_{\mathcal{T}}^{\Gamma}$; we will show that $\omega \in \mathrm{co}\left(\bigcup_{\sigma\in\Sigma_1} \Omega_{\mathcal{T}}^{\Gamma_\sigma}\right)$ by exhibiting a convex combination of points in the polytopes $\{\Omega_{\mathcal{T}}^{\Gamma_\sigma} : \sigma \in \Sigma_1\}$ that equals ω. By definition of realization function (Definition 5.1), ω is the image of a normal-form strategy $\alpha \in \Delta(\Sigma_1 \times \Sigma_2)$ for the team. Hence, by linearity of the realization function $f_{\mathcal{T}}$ (Lemma 5.1),

$$\omega = \sum_{\substack{\sigma_1\in\Sigma_1\\ \sigma_2\in\Sigma_2}} \alpha(\sigma_1,\sigma_2)\, f_{\mathcal{T}}^{\Gamma}(\sigma_1,\sigma_2). \tag{5.1}$$

Now, define

$$\nu_{\sigma_1} := \sum_{\sigma_2\in\Sigma_2} \alpha(\sigma_1,\sigma_2)$$

for each $\sigma_1 \in \Sigma_1$. Clearly, each ν_{σ_1} is non-negative, and the sum of all ν_{σ_1}'s is 1. Hence, from (5.1) we find that

$$\omega = \sum_{\substack{\sigma_1\in\Sigma_1\\ \nu_{\sigma_1}>0}} \nu_{\sigma_1}\xi_{\sigma_1}, \quad \text{where} \quad \xi_{\sigma_1} := \sum_{\sigma_2\in\Sigma_2} \frac{\alpha(\sigma_1,\sigma_2)}{\nu_{\sigma_1}} f_{\mathcal{T}}^{\Gamma}(\sigma_1,\sigma_2).$$

Consequently, if we can show that $\xi_{\sigma_1} \in \Omega_{\mathcal{T}}^{\Gamma_{\sigma_1}}$ for all $\sigma_1 \in \Sigma_1 : \nu_{\sigma_1} > 0$, the proof is complete. Note that for all relevant σ_1, ξ_{σ_1} is a convex

combination of points in the set $\{f^{\Gamma}_{\mathcal{T}}(\sigma_1, \sigma_2) : \sigma_2 \in \Sigma_2\} \subseteq \Omega^{\Gamma_{\sigma_1}}_{\mathcal{T}}$. Finally, using the fact that $\Omega^{\Gamma_{\sigma_1}}_{\mathcal{T}}$ is convex (Lemma 5.1), we find $\xi_{\sigma_1} \in \Omega^{\Gamma_{\sigma_1}}_{\mathcal{T}}$, concluding the proof.

□

The following theorem follows from Lemma 5.2 and characterizes the relationship between Γ and Γ^*. It shows that there is a strong connection between the strategies of Player $\mathcal{T}$ in the auxiliary game and the *ex ante* coordinated strategies for the team members in the original game Γ. To simplify the notation, we drop the player's index in normal-form plans, when not strictly necessary (*i.e.*, we write $\sigma \in \Sigma_1$).

Theorem 5.2. *Games Γ and Γ^* are realization-form equivalent in the following sense:*

(i) ***Team****. Given any distribution over the actions at the game tree root ϕ (i.e., a choice $\Sigma_1 \ni \sigma \mapsto \lambda_\sigma \geqslant 0$ such that $\sum_\sigma \lambda_\sigma = 1$) and any choice of realizations $\{\omega_\sigma \in \Omega^{\Gamma_\sigma}_{\mathcal{T}}\}_{\sigma\in\Sigma_1}$, we have that $\sum_{\sigma\in\Sigma_1} \lambda_\sigma \omega_\sigma \in \Omega^{\Gamma}_{\mathcal{T}}$. The converse is also true: given any $\omega \in \Omega^{\Gamma}_{\mathcal{T}}$, there exists a choice of $\{\lambda_\sigma\}_{\sigma\in\Sigma_1}$ and realizations $\{\omega_\sigma \in \Omega^{\Gamma_\sigma}_{\mathcal{T}}\}_{\sigma\in\Sigma_1}$ such that $\omega = \sum_{\sigma\in\Sigma_1} \lambda_\sigma \omega_\sigma$.*

(ii) ***Adversary****. The adversary's realization polytope satisfies $\Omega^{\Gamma}_{\mathcal{A}} = \Omega^{\Gamma^*}_{\mathcal{A}}$.*

The following is then a direct consequence of Theorem 5.2

Corollary 1. *The set of payoffs reachable in Γ coincides with the set of payoffs reachable in Γ^*. Specifically, any strategy $\{\lambda_\sigma\}_{\sigma\in\Sigma_1}$, $\{\omega_\sigma\}_{\sigma\in\Sigma_1}$ over Γ^* is payoff-equivalent to the realization-form strategy $\omega = \sum_{\sigma\in\Sigma_1} \lambda_\sigma \omega_\sigma$ in Γ.*

Proof. The payoff for the team in Γ is equal to $\langle \sum_{\sigma\in\Sigma_1} \lambda_\sigma \omega_\sigma, y\rangle$, where $y \in \mathbb{R}^{|\cdot|}$ is a generic loss vector.

On the other hand, in Γ^*, $\mathcal{A}$ does not observe the initial move in Γ^*, and therefore the loss vector y remains valid in each Γ_σ. Therefore, the team's payoff in Γ^* is $\sum_{\sigma\in\Sigma_1} \lambda_\sigma \langle \omega_\sigma, y\rangle$. The two payoffs clearly coincide. □

Remark 1. *Since Γ_σ has perfect recall, every realization $\omega_\sigma \in \Omega^{\Gamma_\sigma}$ can be induced by $\mathcal{T}$ via behavioral strategies.*

The above shows that for every *ex ante* coordinated strategy for the team in Γ, there exists a corresponding (payoff-equivalent) behavioral strategy for $\mathcal{T}$ in Γ^*, and *vice versa*. Hence, due to realization-form equivalence

between Γ and Γ^*, finding a TMECor in Γ (employing *ex ante* coordinated normal-form strategies), is equivalent to finding an NE in Γ^* (with behavioral strategies).

5.4 *Fictitious Team-Play:* an Anytime Algorithm for TMECor

This section introduces an anytime algorithm, *fictitious team-play*, for finding a TMECor. It follows from the previous section that in order to find a TMECor in Γ, it suffices to find a two-player NE in the auxiliary game Γ^* (and vice versa, although we do not use this second direction). Furthermore, since Γ^* is a two-player perfect-recall zero-sum game, the *fictitious play (FP)* algorithm can be applied with its theoretical guarantee of converging to an NE. As described in Section 2.5, Fictitious play [29, 146] is an iterative algorithm originally described for normal-form games. It keeps track of average normal-form strategies $\bar{x}_i$, which are output in the end, and they converge to an NE. At iteration t, player i computes the best response against the opponent's empirical distribution of play up to time $t-1$, that is, $x_i^t = \mathsf{BR}(\bar{x}_{-i}^{t-1})$. Then her average strategy is updated as $\bar{x}_i^t = \frac{t-1}{t}\bar{x}_i^{t-1} + \frac{1}{t}x_i^t$. Conceptually, our fictitious team-play algorithm coincides with FP applied to the auxiliary game Γ^*. However, in order to avoid the exponential size of Γ^*, our fictitious team-play algorithm does not explicitly work on the auxiliary game. Rather, it encodes the best-response problems by means of mixed integer linear programs (MILPs) on the original game Γ.

5.4.1 The Main Algorithm

The pseudocode of the main algorithm is given as Algorithm 5.1, where $\mathsf{BR}_{\mathcal{A}}(\cdot)$ and $\mathsf{BR}_{\mathcal{T}}(\cdot)$ are the subroutines for solving the team's and adversary's best-response problems, respectively.

Our algorithm employs realization-form strategies. This allows for a significantly more intuitive way of performing averaging (Steps 7, 8, 10) than what is done in full-width extensive-form fictitious play [97], which employs behavioral strategies.

Our algorithm maintains an average realization $\bar{\omega}_{\mathcal{A}}$ for the adversary. Moreover, the $|\Sigma_1|$-dimensional vector $\bar{\lambda}$ keeps the empirical frequencies of actions at node ϕ in auxiliary game Γ^* (see Figure 5.3). Finally, for each $\sigma \in \Sigma_1$, $\bar{\omega}_{\mathcal{T},\sigma} \in \Omega_{\mathcal{T}}^{\Gamma_\sigma}$ is the average realization of the team in the subtree Γ_σ. After t iterations of the algorithm, only t pairs of strategies are generated. Hence, an optimized implementation of the algorithm can employ a lazy data structure to keep track of the changes to $\bar{\lambda}$ and $\bar{\omega}_{\mathcal{T},\sigma}$.

Algorithm 5.1 proceeds as follows. Initially (Step 2), the average realization $\bar{\omega}_{\mathcal{A}}$ of the adversary is set to the realization-form strategy equivalent to a uniform behavioral strategy profile. At each iteration the algorithm first computes a team's best-response against $\bar{\omega}_{\mathcal{A}}$ (Line 6). We require that the chosen best response assigns probability 1 to one of the available actions (say, a_{σ^t}) at node ϕ. (A pure—that is, non-randomized—best response always exists and, therefore, there always exists at least one best response selecting a single action at the root with probability one.) Then, the average frequencies and team's realizations are updated on the basis of the observed $(\sigma^t, \omega^t_{\mathcal{T}})$ (Line 7 and 8). Finally, the adversary's best response $\omega^t_{\mathcal{A}}$ against the updated average strategy of the team is computed (Line 9), and the empirical distribution of play of the adversary is updated (Line 10).

Algorithm 5.1 Fictitious team-play

1: **function** FICTITIOUSTEAMPLAY(Γ)
2: Initialize $\bar{\omega}_{\mathcal{A}}$
3: $\bar{\lambda} \leftarrow (0, \dots, 0)$, $t \leftarrow 1$
4: $\bar{\omega}_{\mathcal{T},\sigma} \leftarrow (0, \dots, 0) \quad \forall \sigma \in \Sigma_1$
5: **while** within computational budget **do**
6: $(\sigma^t, \omega^t_{\mathcal{T}}) \leftarrow \mathsf{BR}_{\mathcal{T}}(\bar{\omega}_{\mathcal{A}})$
7: $\bar{\lambda} \leftarrow (1 - \frac{1}{t})\bar{\lambda} + \frac{1}{t}\mathbb{1}_{\sigma^t}$
8: $\bar{\omega}_{\mathcal{T},\sigma^t} \leftarrow (1 - \frac{1}{t})\bar{\omega}_{\mathcal{T},\sigma^t} + \frac{1}{t}\omega^t_{\mathcal{T}}$
9: $\omega^t_{\mathcal{A}} \leftarrow \mathsf{BR}_{\mathcal{A}}(\bar{\lambda}, \{\bar{\omega}_{\mathcal{T},\sigma}\}_{\sigma})$
10: $\bar{\omega}_{\mathcal{A}} \leftarrow (1 - \frac{1}{t})\bar{\omega}_{\mathcal{A}} + \frac{1}{t}\omega^t_{\mathcal{A}}$
11: $t \leftarrow t + 1$
12: **end while**
13: **return** $(\bar{\lambda}, (\bar{\omega}_{\mathcal{T},\sigma})_{\sigma \in \Sigma_1})$
14: **end function**

The *ex ante* coordinated strategy profile for the team (*i.e.*, team's strategy at the TMECor) is implicitly represented by the pair $(\bar{\lambda}, \bar{\omega}_{\mathcal{T},\sigma})$. In particular, that pair encodes a coordination device that operates as follows:

- At the beginning of the game, a pure normal-form plan $\tilde{\sigma} \in \Sigma_1$ is sampled according to the discrete probability distribution encoded by $\bar{\lambda}$. Player 1 will play the game according to the sampled plan.
- Player 2 will play according to any normal-form strategy in $f_2^{-1}(\bar{\omega}_{\mathcal{T},\tilde{\sigma}})$, that is, any normal-form strategy whose realization is $\bar{\omega}_{\mathcal{T},\tilde{\sigma}}$.

Then, the correctness of the algorithm is a direct consequence of realization-equivalence between Γ and Γ^*, which was shown in Theorem 5.2. In particular, the strategy of the team converges to a profile that is part of a normal-form NE in the original game Γ (*i.e.*, a TMECor of the original ATG).

5.4.2 Best-Response Subroutines

The problem of finding the adversary's best response to a pair of strategies of the team, namely $\mathsf{BR}_{\mathcal{A}}(\bar{\lambda}, \{\bar{\omega}_{\mathcal{T},\sigma}\}_{\sigma})$, can be efficiently tackled by working on Γ (second point of Theorem 5.2). In contrast, the problem of computing $\mathsf{BR}_{\mathcal{T}}(\bar{\omega}_{\mathcal{A}})$ is NP-hard [167], and inapproximable (Theorem 4.3).

An oracle for the team's best-response problem was already described as the MILP Formulation 4.3. We ameliorate the performances of the best-response procedure by employing a *meta-oracle* that uses simultaneously, as parallel processes, MILP 4.3 and a new team's best-response oracle (described in the following). The meta-oracle stops both subroutines as soon as one of the two has found a solution or, in the case a time-limit is reached, it stops both subroutines and it returns the best solution (in terms of team's utility). This circumvents the need to prove optimality in the MILP, which often takes most of the MILP-solving time, and opens the doors to heuristic MILP-solving techniques. One of the key features of the meta-oracle is that its performances are not impacted by the size of Γ^*, which is never *explicitly* employed in the best-responses computation.

The second best-response oracle employed by the meta-oracle is a MILP (Formulation 5.2) in which the number of binary variables is polynomial in Γ and proportional to the number of sequences of the *pivot* player. The subroutine looks for a pair $(\sigma^t, \omega^t_{\mathcal{T}})$, with $\sigma \in \Sigma_1$, and $\omega^t_{\mathcal{T}} \in \Omega^{\Gamma_\sigma}_{\mathcal{T}}$, in best-response to a given $\bar{\omega}_{\mathcal{A}}$. In order to compute $(\sigma^t, \omega^t_{\mathcal{T}})$, we employ the sequence-form strategies of the team defined over Γ. Specifically, the pure sequence-form strategy r_1 corresponds to selecting a $\sigma \in \Sigma_1$ at ϕ in Γ^*. Determining the (potentially mixed) sequence-form strategy for the other team member (r_2) is equivalent to computing $\omega^t_{\mathcal{T}}$ in the subtree selected by r_1. Without loss of generality, we assume all payoffs of the team to be non-negative; indeed, payoffs can always be shifted by a constant without affecting the BR problem. In the following, sequence form constraints (see Section 2.1.2) are written, as customary, in matrix form as $F_i r_i = f_i$.

Let $u^{\bar{\omega}_{\mathcal{A}}}$ is the $|Q_1| \times |Q_2|$ be the utility matrix of the team obtained by marginalizing with respect to the given realization of the opponent $\bar{\omega}_{\mathcal{A}}$. Moreover, r_1 is a $|Q_1|$-dimensional vector of binary variables. Then, the $\mathsf{BR}_{\mathcal{T}}(\bar{\omega}_{\mathcal{A}})$ subroutine consists of the following MILP:

$$\arg\max_{w,r_1,r_2} \sum_{q_1 \in Q_1} w(q_1) \tag{5.2a}$$

$$\text{s.t. } w(q_1) \leqslant \sum_{q_2 \in Q_2} u^{\bar{\omega}_{\mathcal{A}}}_{q_1,q_2} r_2(q_2) \qquad \forall q_1 \in Q_1 \tag{5.2b}$$

$$w(q_1) \leqslant M r_1(q_1) \qquad \forall q_1 \in Q_1 \tag{5.2c}$$

$$F_1 r_1 = f_1 \tag{5.2d}$$

$$F_2 r_2 = f_2 \tag{5.2e}$$

$$r_2(q_2) \geqslant 0 \qquad \forall q_2 \in Q_2 \tag{5.2f}$$

$$r_1 \in \{0,1\}^{|Q_1|} \tag{5.2g}$$

The formulation can be derived starting from the problem of maximizing $r_1^\top u r_2$ under constraints (5.2d)–(5.2g). Let $a_{q_1} = \sum_{q_2} u^{\bar{\omega}_{\mathcal{A}}}_{q_1,q_2} r_2(q_2)$, and $w(q_1) = r_1(q_1) a_{q_1}$. Then, the objective function becomes $\sum_{q_1 \in Q_1} w(q_1)$. In order to ensure that, whenever $r_1(q_1) = 0$, $w(q_1) = 0$, the following constraints are necessary: $w(q_1) \leqslant M r_1(q_1)$ and $w(q_1) \geqslant 0$, where M is the maximum payoff of the team. Moreover, in order to ensure that $w(q_1) = a_{q_1}$ holds whenever $r_1(q_1) = 1$, we introduce $w(q_1) \leqslant a_{q_1}$ and $w(q_1) \geqslant a_{q_1} - M(1 - r_1(q_1))$. It is enough to enforce upper bounds on w's values (Constraints (5.2b) and (5.2c)) because of the objective function that we are maximizing and since we assume a positive utility for each terminal node.

In settings with more than two team members, our formulation enables one to pick any one team player's strategy and represent it using continuous variables instead of having binary variables for her in the best-response oracle MILP.

An experimental evaluation of fictitious team-play is presented in Chapter 6, where we also compare it to the column-generation technique (HCG) devised in Section 4.2.

CHAPTER 6

Experimental Evaluation

This chapter first presents an evaluation of the empirical inefficiencies due to the lack of coordination among team members. Specifically, Section 6.1 describes an experimental evaluation of the inefficiency indices presented in Section 3.2, where equilibria are computed by exploiting the techniques of Chapter 4. These techniques are also evaluated in terms of the required time to compute an exact equilibrium point in the three coordination settings we identified. Then, we focus on the computation of a TMECor, and compare the column-generation technique (Section 4.2) with fictitious team-play (Section 5.4).

6.1 Empirical Inefficiencies

We start by describing the experimental setting employed in the first set of experiments.

6.1.1 Experimental Setting and Preliminary Observations

The experimental setting for the inefficiencies' evaluation is based on randomly generated extensive-form ATGs. The random game generator takes

as inputs: the number of players $n = |\mathcal{P}|$, 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 according to Section 6.1).

The algorithms are implemented in Python 2.7.6, adopting GUROBI 7.0 for LPs and MILPs, AMPL 20170207 and global optimization solver BARON 17.1.2 [164, 151]. 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.

TMECom are computed according to Section 4.1 and TME is computed via MILP (4.6). TMECor is always computed via Algorithm 4.1 (employing best-response oracle (4.3)), as we observe that employing the reduced normal form (see, *e.g.*, Kohlberg and Mertens [108], Swinkels et al. [160], Vermeulen and Jansen [165]) of the game is largely impractical, even for instances of small size. As an example, Figure 6.1 shows that, in three-player games with $\nu = 0.5$, the algorithm does not reach termination within the deadline even for instances of depth 6. Moreover, the amount of memory required by the algorithm would make the computation unfeasible even with higher time thresholds. Instances of three-player ATGs with depth 6 required, at least, around $20Gb$ of memory each, with the most demanding instances requiring more than $70Gb$. Since the reduced normal form grows exponentially with the size of the tree, this approach is not feasible for larger game instances.

6.1.2 Empirical PoUs

Figure 6.3 describes the empirical $PoU_{Com/No}$, Figure 6.4 describes the empirical $PoU_{Cor/No}$, and Figure 6.5 describes the empirical $PoU_{Com/Cor}$ (on all of the test instances). As an example, Figure 6.2 shows the average empirical inefficiency indices in the specific case of ATGs with 3 players, for different values of ν. By observing Figure 6.2 we observe that, despite

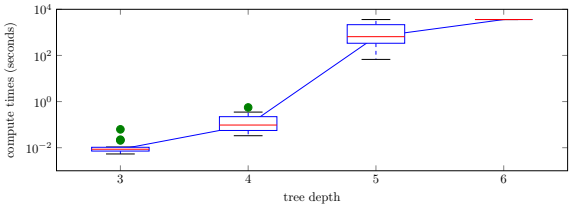

Figure 6.1: *Average compute times and box plots of the computation of TMECor through the reduced normal form. (Three-player instances with $\nu = 0.5$.)*

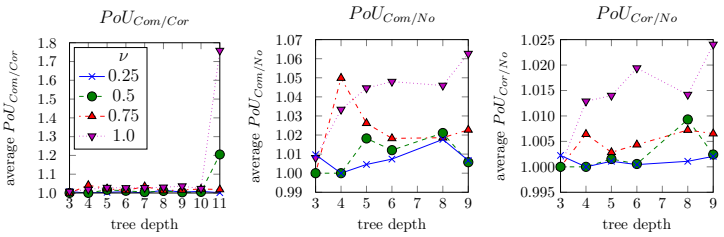

Figure 6.2: *Average empirical inefficiency indices with 3 players and some values of ν.*

the theoretical worst-case value increases in Z, the empirical increase, if any, is negligible. For instance, the worst-case value of $PoU_{Com/Cor}$ with $n = 3$ and $|Z| = 2^{11}$ is > 45, while the average empirical value is < 2. We also observe that the inefficiency increases in ν, suggesting that it may be maximized in normal-form games. As expected, in perfect-information games (*i.e.*, $\nu = 0$), and in games with a small number of infosets (*i.e.*, $\nu = 0.25$) coordination (achieved either via a communication or a coordination device) has a negligible role in improving team's expected utility. The box plots 6.3, 6.4, 6.5 (where data are displayed for a number of actions up to the biggest instances solvable within the time threshold by both the equilibrium-finding algorithms involved in the ratio) confirm these trends.

It is interesting to notice that $PoU_{Com/Cor}$, despite some outliers, mostly remains close to 0. This suggests that, in many settings, *ex ante* coordination may be enough to achieve near-optimal performances for the team. Indeed, *intraplay* communication may enable negligible improvements in team's expected utility. In settings where maintaining communication channels during the playing phase of the game comes at a cost, agents should carefully analyze their possible gains from playing a TMECom, instead of a ("cheaper") TMECor.

Computing Times. In Figure 6.6 we report the average compute times of the algorithms and their box plots with 3 players and $\nu = 0.5$. Figure 6.7 summarizes the results for all time test instances. As expected, the

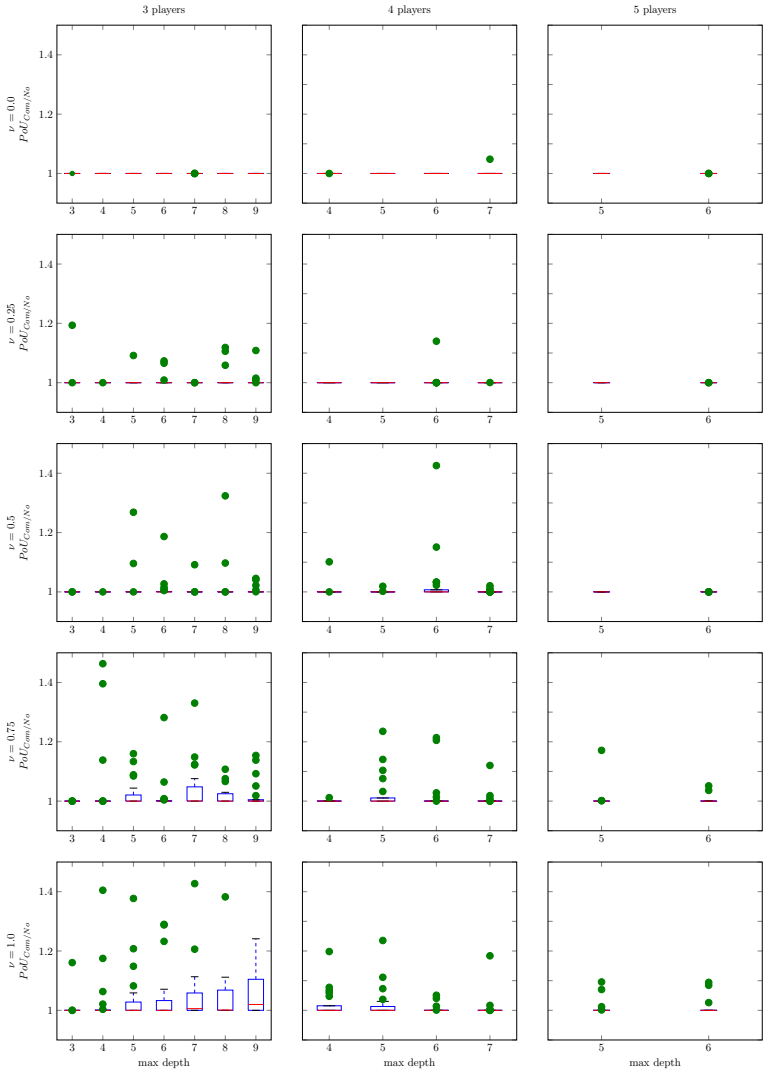

Figure 6.3: *Box plots of the $PoU_{Com/No}$ inefficiency index.*

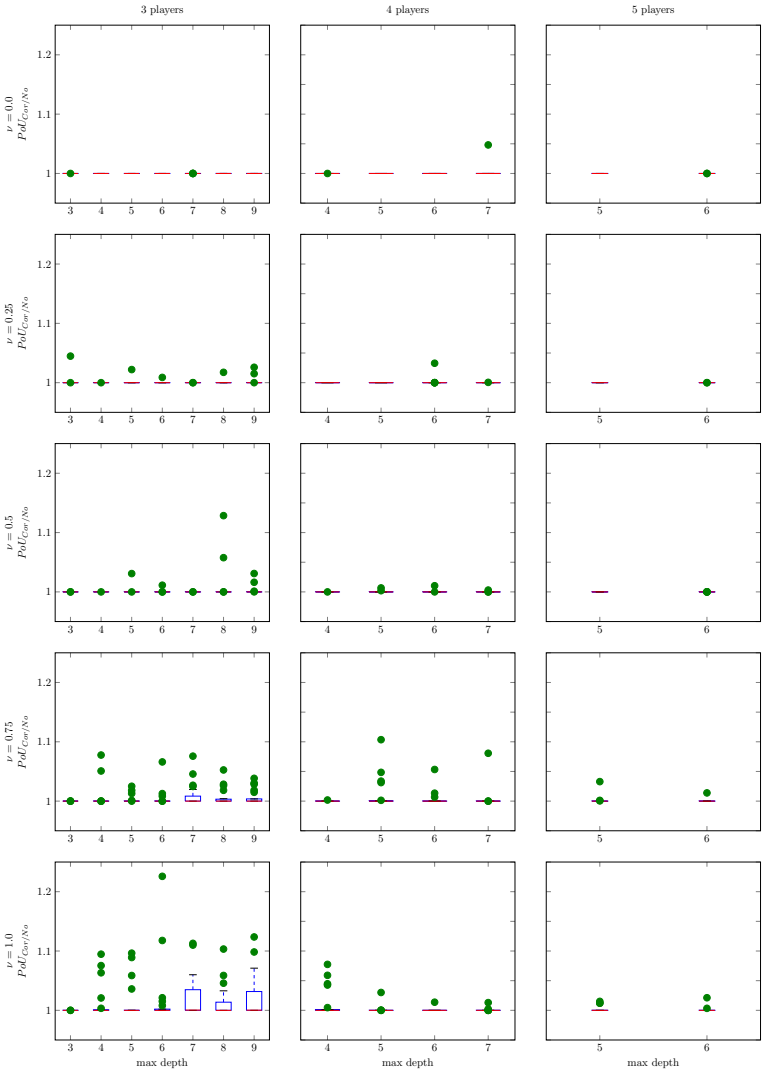

Figure 6.4: *Box plots of the $PoU_{Cor/No}$ inefficiency index.*

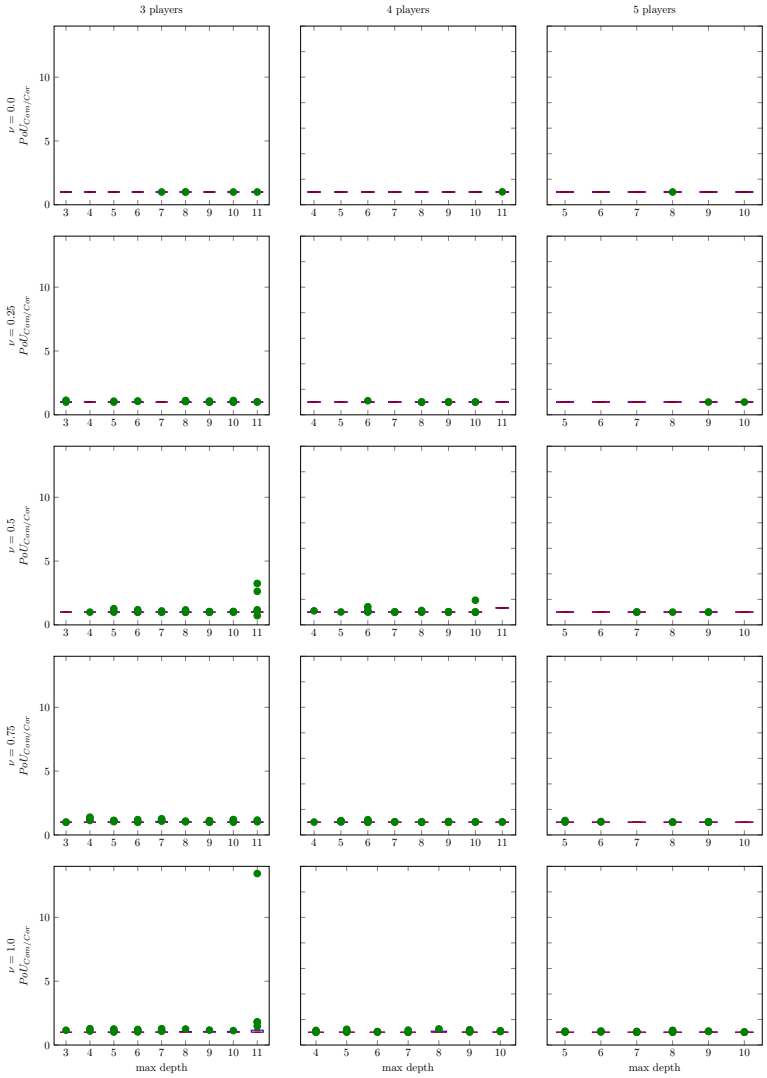

Figure 6.5: *Boxplots of the $PoU_{Com/Cor}$ inefficiency index.*

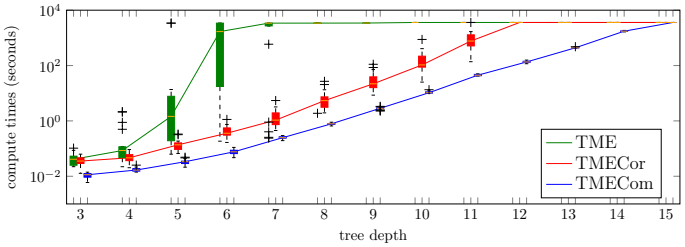

Figure 6.6: *Average compute times of the algorithms and their box plots with 3 players and* $\nu = 0.5$.

TMECom computation scales well, allowing one to solve games with more than 16,000 terminal nodes in the time limit. The performances of Algorithm 4.1 (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.

6.2 Comparison of TMECor-Finding Algorithms

In this section, we focus on the setting with only preplay communication, and present a comparison between different TMECor-finding algorithms.

6.2.1 Experimental Setting

We conducted experiments on three-player Kuhn poker games [117] and three-player Leduc hold'em poker games [157]. These are standard games in the computational game theory literature. Our instances are parametric in the number of ranks in the deck. The instances adopted are listed in Tables 6.1 and 6.2, where Kr and Lr denote, respectively, a Kuhn instance with r ranks and a Leduc instance with r ranks (*i.e.*, $3r$ total cards). Table 6.1 also displays the instances' dimensions in terms of the number of information sets per player and the number of sequences (*i.e.*, number of information set–action pairs) per player, as well as the payoff dispersion Δ_u—that is, the difference between the maximum and minimum attainable team utility.

We present a brief description of the three-player variants of the benchmark games we employ.

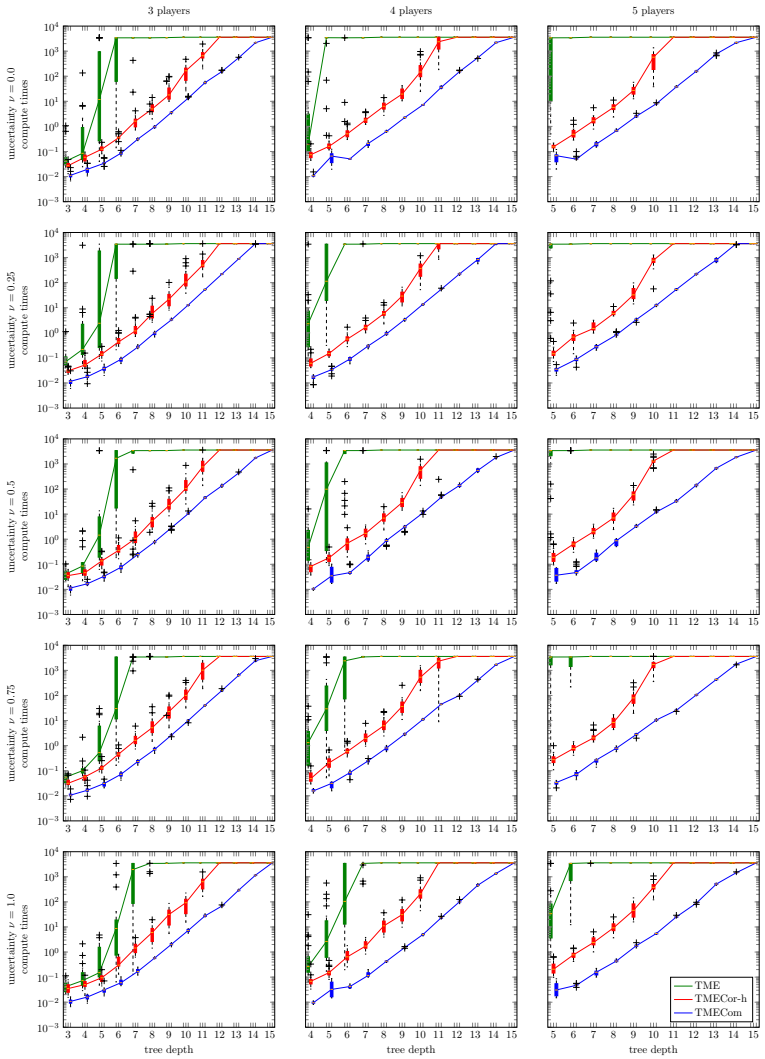

Figure 6.7: *Average compute times of the algorithms and their box plots with every game configuration.*

- **Kuhn3-k.** In Kuhn3-k there are three players and k possible cards. Each player initially pays one chip to the pot, and is dealt a single private card. Then, players act in turns. The first player may check or bet—put one additional chip in the pot. The second player either decides whether to check or bet after first player's check, or whether to fold/call the bet. If no bet was previously made, the third player decides between checking or betting. Otherwise, she has to fold or call. If the second player bet, the first player still has to decide between fold/call. If the third player bet, then both the first and the second have to choose between folding or calling the bet. At the showdown, the player with the highest card who has not folded wins all the chips in the pot.

- **Leduc3-k.** Leduc hold'em poker is a widely-used benchmark in the imperfect-information game-solving community. In order to better evaluate the scalability of our technique, we employ a larger three-player variant of the game. In our enlarged variant, the deck contains three suits and $k \geqslant 3$ card ranks, that is, it consists of triples of cards $1, \ldots, k$ for a total of $3k$ cards.

  Each player initially pays one chip to the pot, and is dealt a single private card. After a first round of betting (with betting parameter $p = 2$, see below), a community card is dealt face up. Then, a second round of betting is played (with betting parameter $p = 4$, see below). Finally, a showdown occurs and players that did not fold reveal their private cards. If a player pairs her card with the community card, she wins the pot. Otherwise, the player with the highest private card wins. In the event that all players have the same private card, they draw and split the pot.

  Each round of betting with betting parameter p goes as follows:

  (1) Player 1 can check or bet p. If Player 1 checks, the betting round continues with Step (2); otherwise, the betting round continues with Step (8).

  (2) Player 2 can check or bet p. If Player 2 checks, the betting round continues with Step (3); otherwise, the betting round continues with Step (6).

  (3) Player 3 can check or bet p. If Player 3 checks, the betting round ends; otherwise, the betting round continues with Step (4).

  (4) Player 1 can fold or call. If Player 1 folds, the betting round continues with Step (5); otherwise, Player 1 adds p to the pot and

the betting round continues with Step (5).

(5) Player 2 can fold or call. In either case the betting round ends. If Player 2 calls, she adds p to the pot.

(6) Player 3 can fold or call. If Player 3 folds, the betting round continues with Step (7); otherwise, Player 3 adds p to the pot and the betting round continues with Step (7).

(7) Player 1 can fold or call. If Player 1 calls, she adds p to the pot. After Player 1's choice the betting round ends.

(8) Player 2 can fold or call. If Player 2 folds, the betting round continues with Step (9); otherwise, Player 2 adds p to the pot and the betting round continues with Step (9).

(9) Player 3 can fold or call. If Player 3 calls, she adds p to the pot. The betting round terminates after her choice.

We compared fictitious team-play against the hybrid column generation (HCG) algorithm (Algorithm 4.1). To make the comparison fair, we instantiate HCG with the same meta-oracle discussed in the previous section. We again use Gurobi 8.0 MILP solver to solve the best response problem for the team. However, in the case of HCG, no time limit can be set on Gurobi without invalidating the theoretical convergence guarantee of the algorithm. This is a drawback, as it prevents HCG from running in an *anytime* fashion, despite column generation otherwise being an anytime algorithm. In the Leduc poker instances, HCG exceeded the memory budget (40 GB).

Game	**Tree size**		Δ_u	**Fictitious team-play**						**HCG**
	Inf.	Seq.		10%	5%	2%	1.5%	1%	0.5%	
K3	25	13	6	0s	0s	0s	1s	1s	1s	0s
K4	33	17	6	1s	1s	4s	4s	30s	1m 12s	9s
K5	41	21	6	1s	2s	44s	1m	4m 15s	8m 57s	1m 58s
K6	49	25	6	1s	12s	43s	5m 15s	8m 30s	23m 32s	25m 26s
K7	57	29	6	4s	17s	2m 15s	5m 46s	6m 31s	23m 49s	2h 50m
L3	457	229	21	15s	1m	14m 05s	30m 40s	1h 34m 30s	$>$ 24h	oom
L4	801	401	21	1s	1m 31s	11m 8s	51m 5s	6h 51m	$>$ 24h	oom

Table 6.1: *Comparison between the run times of fictitious team-play (for various levels of accuracy) and the hybrid column generation (HCG) algorithm.*
*(*oom*: out of memory.)*

6.2.2 Comparison

We instantiated fictitious team-play with the meta-oracle previously discussed (Section 5.4.2), which returns the best solution found by the MILP

Game	Team Utility		
	Adv 1	Adv 2	Adv 3
K3	0.0000	0.0000	0.0003
K4	0.0405	0.0259	-0.0446
K5	0.0434	0.0156	-0.0282
K6	0.0514	0.0271	-0.0253
K7	0.0592	0.0285	-0.0259
L3	0.2332	0.2089	0.1475
L4	0.1991	0.1419	-0.0223

Table 6.2: *Values of the average strategy profile for different choices of adversary.*

Game	Team Utility		
	Adv 1	Adv 2	Adv 3
K3	-0.0002	-0.0002	-0.0001
K4	0.0369	0.0215	-0.0474
K5	0.0405	0.0137	-0.0274
K6	0.0499	0.0262	-0.0267
K7	0.0569	0.0271	-0.0254
L3	0.1533	0.0529	-0.0412
L4	0.0829	-0.029	-0.1901

Table 6.3: *Worst case utilities for the team.*

oracles within the time limit. We let each best-response formulation run on the Gurobi 8.0 MILP solver, with a time limit of 15 seconds and 5000 maximum iterations. Our algorithm is an anytime algorithm, so it does not require a target accuracy ε for an ε-TMECor to be specified in advance. Table 6.1 shows the anytime performance, that is, the time it took to reach an $\alpha\Delta_u$-TMECor for different accuracies $\alpha \in \{10\%, 5\%, 2\%, 1.5\%, 1\%, 0.5\%\}$. Results in Table 6.1 assume that the team consists of the first and third mover in the game; the opponent is the second mover. Table 6.2 shows the value of the average strategy computed by fictitious team-play for different choices of the opponent player. This value corresponds to the expected utility of the team for the average strategy profile $(\bar{\lambda}, \bar{\omega}_{\mathcal{T},\sigma})$ at iteration 1000. Table 6.3 shows the utility that the team is guaranteed to achieve in each game instance, with varying position of the opponent. These values are the worst case utilities, obtained when the opponent is best responding against the average team strategy. Specifically, let $\bar{\omega}'_{\mathcal{T}} \in \Omega^{\Gamma}_{\mathcal{T}}$ be the team realization over Γ induced by the average team strategy $(\bar{\lambda}, (\bar{\omega}_{\mathcal{T},\sigma})_{\sigma\in\Sigma_1})$ (computed through Algorithm 5.1), and let $\omega^*_{\mathcal{A}} = \mathsf{BR}(\bar{\lambda}, (\bar{\omega}_{\mathcal{T},\sigma})_{\sigma\in\Sigma_1})$. Then, the values are computed as, $\bar{\omega}'^{\top}_{\mathcal{T}} U \omega^*_{\mathcal{A}}$, where U is a suitably defined (diagonal) $|Z| \times |Z|$ *payoff* matrix.

Our experiments show that fictitious team-play scales to significantly larger games than HCG. Interestingly, in almost all the games, the value of the team was non-negative: by colluding, the team was able to achieve victory.

Moreover, we evaluate also on these benchmark games the $PoU_{Cor/No}$, confirming that a TMECor provides to the team a substantial payoff increase over the setting where team members play in behavioral strategies. Here, We employ fictitious team-play with 5000 iterations and a time limit of 15 seconds on the oracles' compute times. A TME is computed by

Game	TME	TMECor
K3	$-6.03 \cdot 10^{-8}$	0.0004
K4	0.0237	0.0335
K5	0.0116	0.0205
K6	0.0207	0.0329
K7	0.0198	0.0333

Table 6.4: *Comparison between the utility of the team at the TME and at the TMECor.*

solving the non-linear, non-convex optimization problem described in Formulation 4.6. We employ AMPL 20181005, with the global optimization solver BARON 18.8.23, and we set a time threshold of 15 hours. Table 6.4 describes the results obtained in games where the opponent plays as the second player. Column TME displays the utility obtained when the opponent best-responds to the incumbent team strategies computed by the solver (BARON never reaches an optimal solution within the time limit). It is easy to see that *ex ante* coordination (TMECor) always makes team members better off with respect to playing behavioral strategies (TME).

Part II

Correlated Equilibria for Sequential Games

CHAPTER 7

Complexity Results for Correlated Equilibria in EFGs

We empirically observed (Section 6.1) that in many adversarial settings communication before the beginning of the game is sufficient to achieve near optimal performances for a team of agents sharing the same objectives. Now, we investigate whether coordination via preplay communication is effective (*i.e.*, efficient and easily computable) in general EFGs where players have arbitrary utilities. This amounts to relaxing the hypothesis that players subject to the coordination device have the same utility functions, which calls for incentive constraints to keep recommendations credible for every player. We investigate whether correlation can be reached efficiently even in settings where players with arbitrary utilities (and the mediator) have limited communication capabilities. We focus on EFGs in which only preplay communication is allowed, and study correlated equilibria that allow the mediator to recommend actions just *before* the playing phase of the game (namely, the *correlated equilibrium* (CE) and the *coarse correlated equilibrium* (CCE)), drawing a parallel with the notion of TMECor developed for ATGs.

First, we prove that approximating an optimal (*i.e.*, social welfare max-

imizing) CE is not in Poly-APX even in two player games, unless P=NP (Section 7.1). Next, in Section 7.2, we identify conditions for which finding an optimal CCE is NP-hard. However, we show that an optimal CCE can be found in polynomial-time in two-player extensive-form games without chance moves. Finally, in Section 7.3, we complete the picture on the computational complexity of finding social-welfare-maximizing CCEs by showing that the problem is not in Poly-APX, unless P=NP, in games with three or more players (chance included).

We denote the problems of computing a social-welfare-maximizing CE and CCE as CE-SW and CCE-SW, respectively.

7.1 The Complexity of Approximating an Optimal CE

In this section, we show that CE-SW is inapproximable (*i.e.*, not in Poly-APX), even in two-player games without chance moves. CE-SW is known to be NP-hard [167]. One could ask herself whether a *good* approximate solution can be computed efficiently. We show that the answer is negative, proving that there is no polynomial-time approximation algorithm that finds a CCE whose value approximates that of a solution to CCE-SW up to within any polynomial factor in the input size, unless P=NP. Formally, Poly-APX is the class of optimization problems that admit a polynomial time $\mathsf{poly}(\alpha)$-approximation algorithm,[1] where $\mathsf{poly}(\alpha)$ is a polynomial function of the input size α (see Ausiello et al. [9] for further details).

Theorem 7.1. *CE-SW is not in Poly-APX even in two-player EFGs without chance moves, unless P=NP.*

Proof. Given a boolean formula in conjunctive normal form ϕ with n clauses and m variables, MAX-SAT is the problem of determining the maximum number of clauses that can be made true by a truth assignment to the variables of the formula. Denote by $C := \{c_j\}_{j=1}^{n}$ the set of clauses belonging to ϕ, and by c^* the value of the optimal assignment for MAX-SAT over ϕ. Clearly, $1 \leqslant c^* \leqslant n$. The set of variables appearing in ϕ is denoted by $V := \{v_j\}_{j=1}^{m}$.

First, given an instance ϕ and a constant $\bar{c} \in \{1, \ldots, n\}$, we build a parametric (in the constant) EFG $\Gamma_{\bar{c}}$ with $\mathcal{P} = \{1, 2\}$. In the following, subscripts specify the relevant player. For example, we write I_i to denote a generic element of $\mathcal{I}_i$. The structure of $\Gamma_{\bar{c}}$ is the following.

[1] An r-approximation algorithm for an optimization problem is such that $\frac{OPT}{APX} \leqslant r$, where OPT is the value of the optimal solution to the problem, and APX the value of the solution returned by the approximation algorithm.

- The root of the tree is h_2^0, and $h_2^0 \in I_2^0$, $|I_2^0| = 1$. At h_2^0, Player 2 has one action per clause, that is $A(h_2^0) = \{a_2^{c_j}\}_{j=1}^n$.
- Each $a_2^{c_j}$ leads to a decision node of Player 1. Specifically, for each $c_j \in C$, $h_2^0 \cdot a_2^{c_j} = h_1^{c_j}$. For each $c_j \in C$, $h_1^{c_j} \in I_1^0$, and $|I_1^0| = n$. The set of actions available at this infoset is $A(I_1^0) = \{a_1^{c_j}\}_{j=1}^n \cup \{a_1^{\mathsf{T}}\}$.
- Each $a_1^{c_j}$ leads to a terminal node. Formally, for each $c_i, c_j \in C$, $x_1^{c_i} \cdot a_1^{c_j} = z^{i,j} \in Z$. Payoffs at each $z^{i,j}$ are as follows:

$$u_1(z^{i,j}) = \begin{cases} -1 & \text{if } i = j \\ \frac{n\bar{c}+1}{n-1} & \text{otherwise} \end{cases}, \quad u_2(z^{i,j}) = -u_1(z^{i,j}),$$

  where $\bar{c}$ is the given parameter.
- For each $c_i \in C$, $x_1^{c_i} \cdot a_1^{\mathsf{T}} = h_2^i$. Each h_2^i belongs to a singleton infoset I_2^i, and has one action for each literal appearing in c_i. For any $v_j \in V$, $\pm v_j$ represent, respectively, a positive and a negative literal. Then, actions available at each h_2^i are denoted by $a_2^{i,\pm v_j}$, depending on how variable v_j appears in clause c_i.
- For each h_2^i, and for each $a_2^{i,\pm v_j} \in A(h_2^i)$, $h_2^i \cdot a_2^{i,\pm v_j} = x_1^{i,\pm v_j}$. These nodes are grouped into variable-specific infosets, that is $x_1^{i,\pm v_j} \in I_1^{v_j}$. There is an infoset $I_1^{v_j}$ for each $v_j \in V$. At $I_1^{v_j}$, Player 1 selects the truth assignment for v_j. Specifically, for each $I_1^{v_j}$, $A(I_1^{v_j}) = \{\mathsf{T}, \mathsf{F}\}$, where choosing T (F) means Player 1 sets v_j to true (false).
- For each $i \in \{1, \ldots, n\}$, $j \in \{1, \ldots, m\}$, and $a \in \{\mathsf{T}, \mathsf{F}\}$, $x_1^{i,\pm v_j} \cdot a = z^{i,\pm v_j, a} \in Z$. The utilities in these terminal nodes are defined as $u_2(z^{i,\pm v_j,a}) = 0$ for each $a \in \{\mathsf{T}, \mathsf{F}\}$, and:

$$u_1(z^{i,+v,a}) = \begin{cases} n & \text{if } a = \mathsf{T} \\ 0 & \text{if } a = \mathsf{F} \end{cases}, \quad u_1(z^{i,-v_j,a}) = \begin{cases} 0 & \text{if } a = \mathsf{T} \\ n & \text{if } a = \mathsf{F} \end{cases}.$$

The resulting $\Gamma_{\bar{c}}$ has perfect recall. Indeed, infosets of type $I_1^{v_j}$ are reached only from I_1^0 through action a_1^{T}.

Let $\hat{x} \in \mathcal{X}$ be a probability distribution with the following properties:

Property 3. *There exists $\bar{\sigma}_1 \in \Sigma_1$ such that:*

$$\sum_{\sigma_2 \in \Sigma_2} \hat{x}(\bar{\sigma}_1, \sigma_2) = 1, \quad \textit{and} \quad \bar{\sigma}_1 \in \{a_1^{\mathsf{T}}\} \bigtimes \left(\bigtimes_{v_j \in V} A(I_1^{v_j}) \right).$$

Property 4. *Player 2 selects actions at $A(I_2^0)$ with a uniform probability and, for each infoset I_2^i, she plays a single action with probability 1. Formally, for each $c_i \in C$, there exists $\bar{\sigma}_2^i \in \Sigma_2$ such that:*

$$\sum_{\sigma_1 \in \Sigma_1} \hat{x}(\sigma_1, \bar{\sigma}_2^i) = \frac{1}{n}, \quad \textit{and} \quad \bar{\sigma}_2^i \in \{a_2^{c_i}\} \bigtimes \left(\bigtimes_{v_j \in V} A(I_2^j) \right).$$

Property 5. *The distribution maximizes the social welfare while satisfying Properties 3, 4.*

The joint distribution $\hat{x}$ provides a social welfare equal to $\frac{nc^*}{n} = c^*$.

Let x^* be a solution to CE-SW (*i.e.*, a CE maximizing the social welfare) over $\Gamma_{\bar{c}}$. We provide some guarantees over the social welfare provided by x^* for varying values of the parameter $\bar{c}$. In a second step, we will employ this information to prove our result. There are three cases, depending on the relation between $\bar{c}$ and the optimal value of MAX-SAT for the instance ϕ (c^*).

1. $(\bar{c} < c^*)$. In this setting, $\hat{x}$ is a CE of $\Gamma_{\hat{c}}$. In order to check Player 1's incentive constraints, it is enough to compare $\bar{\sigma}_1$ with a generic plan $\sigma_1' \in \{a_1^{\mathsf{T}}\} \times \left(\times_{v_j \in V} A(I_1^{v_j}) \right)$, $\sigma_1' \neq \bar{\sigma}_1$, and to a generic plan $\sigma_1'' \in \{a_1^{c_i}\}_{c_i \in C} \times \left(\times_{v_j \in V} A(I_1^{v_j}) \right)$. In the first case, CE constraints are not violated due to Property 5. In the second case, assuming Player 2 follows a plan drawn according to $\hat{x}$, Player 1's expected utility for choosing σ_1'' in place of $\bar{\sigma}_1$ is: $-\frac{1}{n} + \frac{n-1}{n}\frac{n\bar{c}+1}{n-1} = \bar{c}$. The expected utility when playing $\bar{\sigma}_1$ is c^*. Since $\bar{c} < c^*$, Player 1 has no interest in deviating from $\bar{\sigma}_1$. The same holds for Player 2 since $\hat{x}$ assigns strictly positive probability only to pairs of plans of type $(\bar{\sigma}_1, \sigma_2)$, and $u_2(\bar{\sigma}_1, \sigma_2) = 0$ for all $\sigma_2 \in \Sigma_2$. Then, $\hat{x}$ being a CE of $\Gamma_{\bar{c}}$, x^* guarantees a social welfare $\geqslant c^*$.

2. $(\bar{c} = c^*)$. We claim that, in this setting, $\Gamma_{\bar{c}}$ admits only CEs that have either social welfare 0 or c^*. Therefore, x^* provides social welfare c^*. To show this, we first note that the only case in which a strictly positive social welfare is reached is when Player 1 plays a_1^{T} at I_1^0. We show that it is rational for Player 1 to play a_1^{T} only when Property 4 holds. Specifically, Player 2 has to play each $a_2^{c_i} \in A(I_2^0)$ with a cumulative probability of $1/n$. To see this, let $x \in \mathcal{X}$ and assume $a_2^{c_i} \in A(I_2^0)$ is

played with probability

$$\gamma = \sum_{\substack{\sigma \in \Sigma: \\ \sigma(I_2^0) = a_2^{c_i}}} x(\sigma) < \frac{1}{n}.$$

Then, by playing $a_1^{c_i}$ at I_1^0, Player 1 obtains $-\gamma + (1-\gamma)\frac{n\bar{c}+1}{n-1} > \bar{c}$. Therefore, she would select $a_1^{c_i}$ rather than a_1^{T}. Now, since Property 4 has to hold to reach a strictly positive social welfare, strategy $\hat{x}$ still guarantees an optimal social welfare. It is still a CE of $\Gamma_{\bar{c}}$ since Player 1's constraints hold with equality. Any other CE that may be reached when a_1^{T} is played would lead to a social welfare of at most c^*. Indeed, Player 1 could play multiple truth assignments with strictly positive probability. However, their convex combination would result in a social welfare of at most c^*. Moreover, any probability distribution in which Player 1 does not play a_1^{T} at I_1^0 provides social welfare equal to 0, due to the payoff structure of $\Gamma_{\bar{c}}$.

3. $(\bar{c} > c^*)$. In this setting, $\Gamma_{\bar{c}}$ admits only CEs with social welfare equal to 0. Strategy $\hat{x}$ is no longer a CE of $\Gamma_{\bar{c}}$. Indeed, even if Property 4 is still enforced, Player 1 is better off when selecting an action of type $a_1^{c_i}$ at I_1^0, which leads to payoff $\bar{c}$ (against c^* obtained when playing a_i^{T}). The same consideration holds for any other joint strategy in which a_1^{T} is recommended with strictly positive probability to Player 1. Moreover, there always exists at least one CE with social welfare 0. To see this, it is enough to consider a simpler game where plans of Player 1 are restricted to $\Sigma_1' = \{\sigma_1 \in \Sigma_1 | \sigma_1(I_1^0) \neq a_1^{\mathsf{T}}\}$, and Player 2 only plays at I_2^0. In this game, a probability distribution such that actions in $A(I_2^0)$ and $A(I_1^0)$ are recommended uniformly is an NE. This solution is also an NE, and therefore a CE, of $\Gamma_{\bar{c}}$, and provides social welfare equal to 0.

Assume, by contradiction, that there exists a polynomial-time approximation algorithm $\mathtt{A}_{\mathsf{CE}}$ providing an approximate solution to CE-SW with approximation factor $r = \frac{1}{f(n)}$, and let $f(n) = 2^n$ (the polynomiality of the process is preserved as $f(n)$ can be codified with n bits). Let us proceed iteratively by setting $\bar{c} = 1$, and increasing its value by 1 at each step, until we reach $\bar{c} = n$. For each value of $\bar{c}$, we run $\mathtt{A}_{\mathsf{CE}}$ over $\Gamma_{\bar{c}}$. If $\bar{c} < c^*$ (case 1), we obtain a solution which is greater than or equal to $\frac{c^*}{f(n)}$ and, therefore, it is > 0. If $\bar{c} = c^*$ (case 2), $\mathtt{A}_{\mathsf{CE}}$ returns $\frac{c^*}{f(n)}$, which is, again, > 0. If $\bar{c} > c^*$ (case 3), $\mathtt{A}_{\mathsf{CE}}$ returns 0. If we denote by $\tilde{c}$ the first value of $\bar{c}$ for which $\mathtt{A}_{\mathsf{CE}}$

returns 0, then we have (in polynomial time in the size of ϕ) that $c^* = \tilde{c} - 1$. This leads to a contradiction as, if $\mathtt{A}_{\text{CE}}$ existed, MAX-SAT—which is known to be NP-hard—would be solvable in polynomial time. □

7.2 Complexity of Computing an Optimal CCE

In this section, we first make a few useful remarks on the nature of CCEs. Then, we study in which cases CCE-SW is computationally hard (Section 7.2.2), and prove that the problem is poly-time solvable in two-player EFGs with no chance (Section 7.2.3).

7.2.1 Remarks on CCE

Before proceeding with the computational complexity results regarding CCEs, let us underline some of the factors that make CCE an appealing solution concept from a practical perspective:

- An optimal (*i.e.*, social-welfare-maximizing) CCE may lead to an arbitrarily larger welfare than an optimal CE (see Example 7), which, in its turn, may be arbitrarily better than a socially optimal Nash equilibrium. This is particularly important in the context of multi-player, general-sum games, where we believe a credible equilibrium point should be one maximizing the overall social welfare.

- CCE requires players to adhere to the mechanism *ex ante* the game, which calls for credible players' commitments before the recommendations are revealed. This is not unrealistic as, on a general level, players will observe their *ex ante* commitment every time they reason with a long term horizon, where a reputation for credibility positively affects their utility. Moreover, they could be forced to stick to their ex ante commitment by contractual agreements.

- It is known that finding a socially optimal CE is NP-hard [167] (and not in Poly-APX, as shown in Theorem 7.1) even in the simple case of two-player extensive-form games without chance moves.

The following example highlights that an optimal CCE may guarantee a better social welfare than an optimal CE.

Example 7. *Consider the extensive-form game in Figure 7.1. The game is parametric in $k > 1$, and it has two players. Each player has a unique information set (I_1 for Player 1, and I_2 for Player 2).*

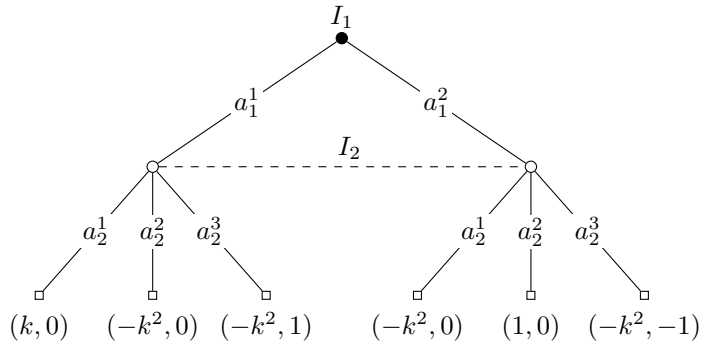

Figure 7.1: *Example illustrating the difference between CE and CCE.*

The joint strategy profile assigning probability $1/2$ *to* (a_1^1, a_2^1) *and* (a_1^2, a_2^2) *is the CCE maximizing the social welfare of the players, which is* $(k+1)/2$*. The unique optimal CE is the probability distribution assigning probability 1 to* (a_1^2, a_2^2)*, providing a social welfare of 1 independently of* k*. Therefore, for increasing values of* k *an optimal CCE allows the players to reach a social welfare which is arbitrarily larger than the social welfare reached through the optimal CE.*

7.2.2 Negative Result

We study the complexity of CCE-SW. The problem is shown to be NP-hard in EFGs with three or more players, including chance. The setting in which $|\mathcal{P}| = 2$ is addressed in Section 7.2.3.

Theorem 7.2. *CCE-SW is NP-hard even in EFGs with two-players, chance moves, and binary outcomes.*

Proof. The construction introduced in [167, Theorem 1.3] can be employed. We describe its basic structure and how it can be adapted to our setting.

The reduction is from SAT, whose generic instance is a Boolean formula ϕ in conjunctive normal form with a set C of clauses, $|C| = n$, and a set V of variables, $|V| = m$. Given ϕ, we build an auxiliary game Γ_ϕ, which has size proportional to that of ϕ. The root of Γ_ϕ is a chance node, with one action for each $c_i \in C$. Then, on the *second level* of the tree, there are n decision nodes of Player 1, each one belonging to a singleton infoset and identifying a single clause of ϕ. At each of this decision nodes, Player 1 has an action for each literal appearing in the clause identified by the decision node. Player 2 plays on the *third level* of the game and has a decision node for each literal appearing in ϕ. An action of Player 1 leads to

the decision node of Player 2 corresponding to the same literal. Decision nodes of Player 2 are grouped in m infosets, one for each variable in V. Each of these information sets has two available actions, corresponding to the truth assignment of the variable, which lead to a terminal node. Both players have the same payoffs, which is 0 if the literal (chosen by Player 1) is false and 1 if it is true, when evaluated under the truth assignment selected by Player 2.

Players' expected payoffs are equal to the number of literals selected by Player 1, evaluating to true under the truth assignment of Player 2, divided by the number of clauses n. As a result, if ϕ is satisfiable, then Γ_ϕ admits a pure joint strategy (*i.e.*, a joint plan $\sigma \in \Sigma$) guaranteeing social welfare 2. Otherwise, the maximum expected social welfare cannot be more than $2(1 - 1/n)$. A pure strategy maximizing the social welfare is also a CCE, since no *ex-ante* deviation would result in an increase in each player's utility, being it already maximal. A CCE may recommend with strictly positive probability more than a single plan. In this case, if the guaranteed social welfare is 2, any truth assignment played with strictly positive probability satisfies ϕ. Then, finding a solution to CCE-SW in polynomial time would imply the existence of a polynomial time algorithm for the SAT problem, which leads to a contradiction, unless P=NP. □

In Section 7.2.3, we show that CCE-SW can be found in polynomial time when $|\mathcal{P}| = 2$ and there are no chance moves. Therefore, Theorem 7.2 states that, by introducing chance, the problem transitions from polynomially solvable to NP-hard, and this happens despite chance being non-strategic. Other problems in which this transition takes place are: the problem of computing socially optimal EFCE [167], and the problem of deciding if a two-player, zero-sum EFG admits a pure strategy equilibrium [23, 89].

The following corollary is an immediate consequence of Theorem 7.2.

Corollary 2. *CCE-SW is NP-hard even in EFGs with $|\mathcal{P}| = 3$, and no chance moves.*

Proof. A simple variation of construction for the proof of Theorem 7.2 can be employed. Let ϕ be a given instance of SAT. We substitute the chance node at the root of Γ_ϕ with a decision node of Player 3, who receives the negative of the identical payoffs to Player 1 and 2. Therefore, Player 3 has an incentive to randomize. The maximum social welfare achievable at a CCE is 1 if and only if the SAT formula is satisfiable. □

7.2.3 Positive Result

In this section, we address the problem of solving CCE-SW in EFGs with $|\mathcal{P}| = 2$, and no chance moves. We show that there exists a polynomial-time algorithm for this setting. This is in contrast with the hardness and inapproximanility results for CE-SW, even in the two-player setting (Theorem 7.1).

First, we provide a compact formulation of the problem. Then, we describe a polynomial-time algorithm for solving it.

Problem Formulation. A direct application of Definition 2.6 yields an LP with an exponential number of variables and an exponential number of constraints. However, the structure of the problem allows us to provide a (more) compact formulation. The following result holds for EFGs with an arbitrary number of players, including chance.

Lemma 7.1. *Given an EFG, CCE-SW can be formulated as an LP with an exponential number of variables and a polynomial number of constraints.*

To ease the notation, we describe the formulation for the setting we are addressing in this section (*i.e.*, two player games with no chance moves). The formulation can be easily extended to settings with more than two players and chance.

To prove the lemma, we provide a hybrid representation which exploits the tree structure of the problem by combining the normal and the sequence form. We say that a realization plan r_i is *pure* if is satisfies the sequence-form constraints, and it is such that $r_i \in \{0,1\}^{|Q_i|}$. A realization plan is *realization equivalent* to a normal-form plan if, for any strategy profile of the other players, they enforce the same probability distribution over terminal nodes Z. We write r_{σ_i} to denote the $|Q_i|$-dimensional column vector representing the pure realization plan of player i realization equivalent to $\sigma_i \in \Sigma_i$. Any plan is realization equivalent to exactly one pure realization plan (see, *e.g.*, von Stengel 1996).

Following Definition 2.6, the incentive constraints describing a CCE can be written as follows (we report only Player 1's contraints, the same reasoning can be applied for the other player): for each $\sigma_1' \in \Sigma_1$ it has to hold that

$$\sum_{\sigma_1 \in \Sigma_1} \sum_{\sigma_2 \in \Sigma_2} x(\sigma_1, \sigma_2) U_1(\sigma_1, \sigma_2) - \sum_{\sigma_1 \in \Sigma_1} \sum_{\sigma_2 \in \Sigma_2} x(\sigma_1, \sigma_2) U_1(\sigma_1', \sigma_2) \geqslant 0. \tag{7.1}$$

The first term of the constraint is Player 1's expected utility when both players follow recommendations drawn from $x \in \mathcal{X}$.

In a two-player EFG, the problem of determining Player 1's best response against a fixed sequence form strategy $\bar{r}_2$ of Player 2 can be compactly written as the following LP:

$$\max_{r_1} \quad r_1^\top u_1 \bar{r}_2 \tag{7.2}$$

$$\text{s.t. } F_1 r_1 = f_1 \tag{7.3}$$

$$r_1 \geqslant 0 \tag{7.4}$$

where $u_i \in \mathbb{R}^{|\times_{i\in\mathcal{P}} Q_i|}$ is the sparse sequence-form utility matrix of player i. Let $v_1 \in \mathbb{R}^{(|\mathcal{I}_1|+1)}$ be the column vector of dual variables corresponding to constraints (7.3). The dual LP is the following:

$$\min_{v_1} \quad f_1^\top v_1 \tag{7.5}$$

$$\text{s.t. } F_1^\top v_1 - u_1 \bar{r}_2 \geqslant 0. \tag{7.6}$$

For each $I \in \mathcal{I}_1$, $v_1(I)$ is the portion of expected utility that Player 1 achieves by playing her best response in the subgame rooted in I, given Player 2's strategy $\bar{r}_2$. Variable $v_1(I_\varnothing)$ (*i.e.*, $f_1^\top v_1$) represents Player 1's expected utility under the realization plan she chooses to play, given $\bar{r}_2$. In the optimal solution, for LP duality, $f_1^\top v_1$ is Player 1's expected utility at the best response.

Then, by exploiting dual variables v_1, we can rewrite Constraints (7.1) as:

$$\begin{cases} \displaystyle\sum_{\sigma_1\in\Sigma_1}\sum_{\sigma_2\in\Sigma_2} x(\sigma_1,\sigma_2)U_1(\sigma_1,\sigma_2) = f_1^\top v_1 \\ \displaystyle f_1^\top v_1 - \sum_{\sigma_1\in\Sigma_1}\sum_{\sigma_2\in\Sigma_2} x(\sigma_1,\sigma_2)U_1(\sigma_1',\sigma_2) \geqslant 0 \quad \forall \sigma_1' \in \Sigma_1 \end{cases}.$$

By rewriting the double summation in the above inequalities we obtain:

$$\sum_{\sigma_2\in\Sigma_2}\left(\sum_{\sigma_1\in\Sigma_1} x(\sigma_1,\sigma_2)\right) U_1(\sigma_1',\sigma_2),$$

where $\bar{x}_2(\sigma_2) = \sum_{\sigma_1\in\Sigma_1} x(\sigma_1,\sigma_2)$, $\bar{x}_2 \in \Delta(\Sigma_2)$, can be interpreted as the prior probability that Player 2 will play plan σ_2. Since Player 2 has perfect recall, any normal-form strategy can be written as a realization-equivalent sequence-form strategy (*Kuhn's theorem*, see [118]). Specifically, we obtain $\bar{r}_2 = \sum_{\sigma_2\in\Sigma_2} \bar{x}_2(\sigma_2) r_{\sigma_2}$, which is a valid realization plan due to convexity. Now, for $x \in \mathcal{X}$ to be a CCE, $f_1^\top v_1$ needs to be greater than or

equal to the value of the best response of Player 1 given Player 2's strategy $\bar{r}_2$. Notice that evaluating $f_1^\top v_1$ with respect to $\mathsf{BR}(\bar{r}_2)$ corresponds to incentive compatibility *ex ante* the recommendations by the mediator. Then, by exploiting the dual of the best-response problem in sequence form (LP (7.5)-(7.6)), the constraints read:

$$F_1^\top v_1 - u_1 \bar{r}_2 \geqslant 0.$$

Finally, by expanding $\bar{r}_2$ and adding Player 2's constraints, we obtain the following LP:

$$\max_{x \geqslant 0, v_1, v_2} \quad \sum_{(\sigma_1,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) r_{\sigma_1}^\top (u_1 + u_2)\, r_{\sigma_2} \tag{7.7}$$

$$\text{s.t.} \quad \sum_{(\sigma_1,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) r_{\sigma_1}^\top u_i r_{\sigma_2} = f_i^\top v_i \qquad \forall i \in \mathcal{P} \tag{7.8}$$

$$F_1^\top v_1 - u_1 \left(\sum_{\sigma_2\in\Sigma_2} \left(\sum_{\sigma_1\in\Sigma_1} x(\sigma_1,\sigma_2) \right) r_{\sigma_2} \right) \geqslant 0 \tag{7.9}$$

$$F_2^\top v_2 - u_2^\top \left(\sum_{\sigma_1\in\Sigma_1} \left(\sum_{\sigma_2\in\Sigma_2} x(\sigma_1,\sigma_2) \right) r_{\sigma_1} \right) \geqslant 0 \tag{7.10}$$

$$\sum_{(\sigma_2,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) = 1, \tag{7.11}$$

where objective function (7.7) is equivalent to

$$\sum_{(\sigma_1,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) \left(\sum_{i\in\mathcal{P}} U_i(\sigma_1,\sigma_2) \right).$$

This formulation constitutes a proof to Lemma 7.1 as it employs a polynomial number of constraints (namely, $|Q_1| + |Q_2| + 3$), and an exponential number of variables (*i.e.*, $|\Sigma| + |\mathcal{I}_1| + |\mathcal{I}_2| + 2$).

A Polynomial-Time Algorithm. In order to provide a polynomial-time algorithm for CCE-SW when $|\mathcal{P}| = 2$, we first have to show that LP (7.7)-(7.11) admits an efficient separation oracle.

The following lemma is a step in this direction. It shows that a player i can reason in a best-response fashion to minimize the product of a certain $|Q_i|$-dimensional vector by her realization plan, while guaranteeing the reachability of a given terminal node.

Lemma 7.2. *Given a two-player EFG Γ, an outcome $z \in Z$, and a vector $\xi \in \mathbb{R}^{|Q_1|}$, the problem of finding $\sigma_2^* \in \Sigma_2$ such that:*

Algorithm 7.1 `Constrained-plan search for` (z, ξ)

function F(I_{cur},Q^*) ▷ $I_{\mathsf{cur}} \in \mathcal{I}_2$ is the current infoset
  $\hat{Q} \leftarrow \varnothing$, $w(q_2) \leftarrow -\infty \quad \forall q_2 \in Q_2$
  if $I \in \mathcal{I}_2^z$ **then**
    $\hat{Q} \leftarrow \{q_2 \in Q_2 | q_2 \in Q(I) \text{ and } q_2 \in Q_2^z\}$
  else
    $\hat{Q} \leftarrow Q(I)$
  end if
  for $q_2 \in \hat{Q}$ **do**
    $w(q_2) \leftarrow \xi^\top u_1 \mathbf{e}_{q_2} + \sum_{I \in I(q_2)} F(I, Q^*)$
  end for
  $q_2^* = \arg\min_{q_2 \in Q_2} w(q_2)$
  $Q^* \leftarrow Q^* \cup \{q_2^*\}$
  return $w(q_2^*)$
end function

1. *there exists a* $\sigma_1 \in \Sigma_1$ *such that* (σ_1, σ_2^*) *leads to* z
2. $\xi^\top u_1 r_{\sigma_2^*} = \min_{\sigma_2} \xi^\top u_1 r_{\sigma_2}$

can be solved in polynomial time. The same holds when the two players are interchanged.

Proof. Let us focus on the case in which, given $z \in Z$ and $\xi \in \mathbb{R}^{|Q_1|}$, we look for $\sigma_2^* \in \Sigma_2$. This problem can be solved in polynomial time as shown in Algorithm 7.1, where $\mathcal{I}_i^z$ and Q_i^z are, respectively, the set of infosets and sequences of i encountered on the path from the root to z, and $\mathbf{e}_{q_2} \in \{0,1\}^{|Q_2|}$ is the canonical vector with a single 1 corresponding to q_2.

The algorithm propagates values backward starting from terminal nodes, and selects the sequence minimizing such values locally at each infoset. When it encounters infosets in $\mathcal{I}_2^z$, the relevant sequence in Q_2^z is selected. The running time is polynomial in $|\mathcal{I}_2|$. Once Q^* has been determined by visiting each $I \in \mathcal{I}_2$, the corresponding optimal σ_2^* can be built directly. □

Let us focus on the dual $\mathcal{D}$ of LP (7.7)-(7.11):

Lemma 7.3. *$\mathcal{D}$ admits a polynomial-time separation oracle.*

Proof. Let $\alpha_i \in \mathbb{R}$, for all $i \in \mathcal{P}$, be the dual variables of constraints (7.8), $\beta_1 \in \mathbb{R}^{|Q_1|}$ the dual variables of constraints (7.9), $\beta_2 \in \mathbb{R}^{|Q_2|}$ the dual variables of constraints (7.10), and $\gamma \in \mathbb{R}$ the dual variable of constraint (7.11). When $|\mathcal{P}| = 2$, $\mathcal{D}$ is an LP with a number of variables $(|Q_1| + |Q_2| + 3)$ polynomial in the size of the tree, and an exponential $(|\Sigma| + |\mathcal{I}_1| + |\mathcal{I}_2| + 2)$ number of constraints.

We show that, given a vector $\psi = (\bar{\alpha}_1, \bar{\alpha}_2, \bar{\beta}_1, \bar{\beta}_2, \bar{\gamma})$, the problem of either finding a hyperplane separating ψ from the seat of feasible solutions to $\mathcal{D}$ or proving that no such hyperplane exists can be solved in polynomial time. As the number of dual constraints corresponding to the primal variables v_1, v_2 is linear, all these constraints can be checked efficiently for violation. Besides those, the problem features the following constraint for each $(\sigma_1, \sigma_2) \in \Sigma$:

$$r_{\sigma_1}^\top u_1 r_{\sigma_2} \bar{\alpha}_1 + r_{\sigma_1}^\top u_2 r_{\sigma_2} \bar{\alpha}_2 + \bar{\beta}_1^\top u_1 r_{\sigma_2} + r_{\sigma_1}^\top u_2 \bar{\beta}_2 + \bar{\gamma} \geqslant r_{\sigma_1}^\top (u_1 + u_2) r_{\sigma_2},$$

where we recall that $u_i \in \mathbb{R}^{|\times_{i \in \mathcal{P}} Q_i|}$. Given ψ, the *separation problem* of finding a maximally violated inequality of $\mathcal{D}$ reads:

$$\min_{(\sigma_1,\sigma_2)\in\Sigma} \left\{ r_{\sigma_1}^\top \left((\bar{\alpha}_1 - 1)u_1 + (\bar{\alpha}_2 - 1)u_2\right) r_{\sigma_2} + \bar{\beta}_1^\top u_1 r_{\sigma_2} + r_{\sigma_1}^\top u_2 \bar{\beta}_2 \right\}. \tag{7.12}$$

A pair (σ_1, σ_2) yielding violated inequality exists if and only if the separation problem admits an optimal solution of value $< -\bar{\gamma}$. If such a pair (σ_1, σ_2) exists, it can be found in polynomial time by enumerating over the (polynomially many) outcomes $z \in Z$. For each z, we look for (σ_1, σ_2) minimizing the objective function of the separation problem, halting as soon as a pair yielding a violated constraint is found. If the procedure terminates without finding any suitable pair, we deduce that no violated inequalities exist and $\mathcal{D}$ has been solved. Given $z \in Z$, we first notice that the term $r_{\sigma_1}^\top((\bar{\alpha}_1 - 1)u_1 + (\bar{\alpha}_2 - 1)u_2)r_{\sigma_2}$ is completely determined. The remaining terms can be minimized independently for each player. Hence, for each outcome z and for each player i, the corresponding optimal plan can be found in polynomial time due to Lemma 7.2. □

Then, we can state the following key result:

Theorem 7.3. *Given an EFG with $|\mathcal{P}| = 2$ and without chance moves, CCE-SW can be computed in polynomial time in the size of the game.*

Proof. Due to the equivalence between optimization and separation [86], and since the separation problem for $\mathcal{D}$ can be solved in polynomial time (Lemma 7.3), one can solve $\mathcal{D}$ in polynomial time via the ellipsoid method [107]. As the ellipsoid method solves a primal-dual system encompassing both $\mathcal{D}$ and the primal (LP (7.7)-(7.11)), it also produces a solution to the latter. □

7.3 Complexity of Approximating an Optimal CCE

Theorem 7.2 showed in two-player EFGs with chance moves, or in EFGs where $|\mathcal{P}| > 2$, CCE-SW is NP-hard. Here, we provide an even stronger

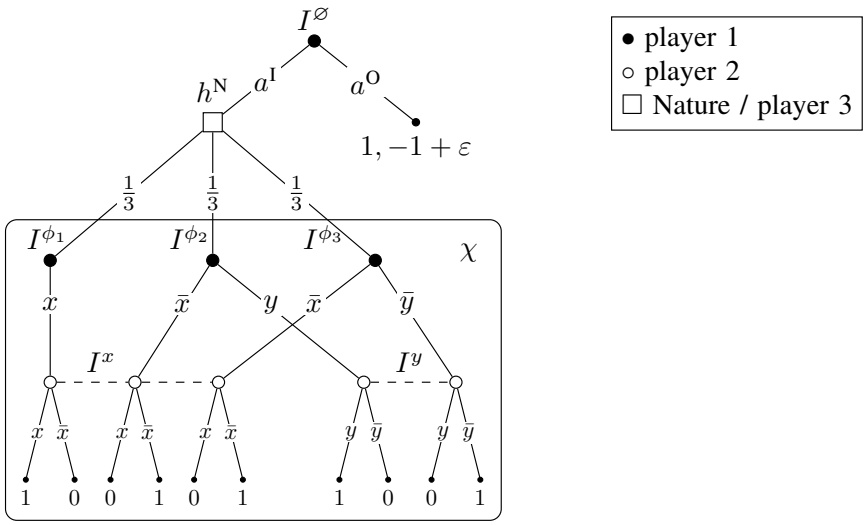

Figure 7.2: *An example of game for the reduction of Theorem 7.4, where* $V = \{x, y, z\}$, $C = \{\phi_1, \phi_2, \phi_3\}$, $\phi_1 = x$, $\phi_2 = \bar{x} \vee y$, *and* $\phi_3 = \bar{x} \vee y$.

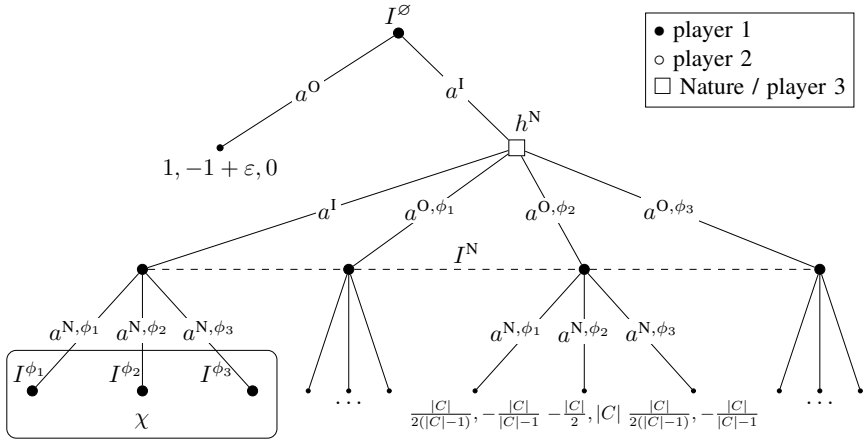

Figure 7.3: *An example of game for the reduction of Theorem 7.5, with* V *and* C *as in Figure 7.2.*

negative result: there is no polynomial-time approximation algorithm which finds a CCE whose value approximates that of a social-welfare-maximizing CCE up to any polynomial factor in the input size unless P = NP. We prove our results by means of a reduction from SAT, a well known NP-complete problem [79]. We denote by C the set of clauses and by V the finite set of variables.

For clarity, Figure 7.2 and 7.3 show concrete examples of the EFGs employed for the reductions of Theorems 7.4 and 7.5.

Theorem 7.4. *Given a two-player EFG with Nature, the problem of computing a social-welfare-maximizing CCE is not in Poly-APX unless P = NP.*

Proof. An example of our reduction from SAT is provided in figure 7.2. We provide a reduction from SAT. Given a SAT instance (C, V), we build a two-player EFG with Nature $\Gamma_\varepsilon(C, V)$ with the following structure:

- The game starts in $h^\varnothing \in I^\varnothing$, where Player 1 chooses an action between a^{I} and a^{O}. In the first case, the game goes on with $h^\varnothing \cdot a^{\mathrm{I}} = h^{\mathrm{N}}$. Otherwise, the game ends with $u_1(z) = 1$ and $u_2(z) = -1 + \varepsilon$.
- At state h^{N}, Nature selects an action among $\{a^\phi \mid \phi \in C\}$ uniformly at random, with $h^{\mathrm{N}} \cdot a^\phi = h^\phi$.
- Each state h^ϕ constitutes a Player 1's infoset I^ϕ. At I^ϕ, Player 1 chooses an action in $\{a^{\phi,l} \mid l \in \phi\}$, where l denotes a literal in ϕ. Then, $h^\phi \cdot a^{\phi,l} = h^{\phi,l}$.
- All states $h^{\phi,l}$ such that $l = v$ or $l = \bar{v}$ for some $v \in V$ belong to the same Player 2's infoset I^v. At I^v, Player 2 has two actions available, namely a^v and $a^{\bar{v}}$.
- Then, the game ends and players' payoffs $u_1(z) = u_2(z)$ are equal to 1 if and only if $z = h^{\phi,l} \cdot a^l$, while they are 0 otherwise.

Intuitively, each of the $2^{|V|}$ Player 2's plans corresponds to a truth assignment τ where variable $v \in V$ is set to TRUE (resp., FALSE) if a^v (resp., $a^{\bar{v}}$) is played at I^v. Moreover, a Player 1's plan determines whether the game is played (a^{I}) or not (a^{O}) and, in the first case, it selects one literal for each clause $\phi \in C$ (corresponding to the action played at infoset I^ϕ). If Player 1 plays a^{I}, Nature chooses a clause $\phi \in C$ uniformly at random, and, then, the players' payoffs are 1 if and only if Player 1 selected a literal of ϕ evaluating to TRUE under τ. Thus, in this case, players' expected payoffs are equal to the number of literals selected by Player 1 evaluating

to TRUE under τ, divided by the number of clauses $|C|$. As a result, if SAT is satisfiable, then there exists a joint plan where players' expected payoffs are equal to 1. It is sufficient that Player 2 plays the plan associated to a satisfying truth assignment τ, while Player 2 selects a literal evaluating to TRUE under τ for each clause. This is also a CCE with maximum social welfare equal to 2, as it provides the players with their maximum expected payoffs. Instead, if SAT is not satisfiable, then any CCE must recommend Player 1 to play a^{O} at $I^{\varnothing}$, otherwise her expected payoff would be strictly less than 1 and she would have an incentive to deviate to action a^{O}, reaching a payoff of 1. Hence, in this case, any CCE has social welfare ε. Now, let $\varepsilon = \frac{1}{2^{\eta}}$, where η is the size of the SAT instance (ε can be encoded with a number of bits polynomial in $|C|$ and $|V|$). Assume there is a polynomial-time $\mathsf{poly}(\eta)$-approximation algorithm. If SAT is satisfiable, the algorithm applied to $\Gamma_{\varepsilon}(C, V)$ would return a CCE with social welfare at least $\frac{2}{\mathsf{poly}(\eta)}$. Since, for η sufficiently large, $\frac{2}{\mathsf{poly}(\eta)} > \frac{1}{2^{\eta}}$, the algorithm would allow us to decide in polynomial time whether SAT is satisfiable, a contradiction unless $\mathsf{P} = \mathsf{NP}$. □

Theorem 7.5. *Given a three-player EFG without Nature, the problem of computing a social-welfare-maximizing CCE is not in Poly-APX unless P = NP.*

Proof. We use a reduction similar to that in Theorem 7.4. We build a three-player EFG $\hat{\Gamma}_{\varepsilon}(C, V)$ such that:

- The game starts in $h^{\varnothing} \in I^{\varnothing}$, as $\Gamma(C, V)$.
- At state h^{N}, Player 3 plays an action $\{a^{\mathrm{O},\phi} \mid \phi \in C\} \cup \{a^{\mathrm{I}}\}$, with $h^{\mathrm{N}} \cdot a^{\mathrm{I}} = h^{\mathrm{I}}$, $h^{\mathrm{N}} \cdot a^{\mathrm{O},\phi} = h^{\mathrm{O},\phi}$.
- All states $h^{\mathrm{O},\phi}$ and h^{I} belong to a Player 1's infoset I^{N}, where she selects an action among $\{a^{\mathrm{N},\phi} \mid \phi \in C\}$.
- Then, if Player 3 played a^{I}, $h^{\mathrm{I}} \cdot a^{\mathrm{N},\phi} = h^{\phi}$ and the games goes on as $\Gamma(C, V)$ (with Player 3's payoffs set to zero). Instead, if Player 3 played an action $a^{\mathrm{O},\phi}$, the game ends with $2u_1(z) = 2u_2(z) = -u_3(z) = \frac{|C|}{|C|-1}$ if $z = h^{\mathrm{O},\phi} \cdot a^{\mathrm{N},\phi'}$) and $\phi \neq \phi'$, while $2u_1(z) = 2u_2(z) = -u_3(z) = -|C|$ if $\phi = \phi'$.

Intuitively, the introduction of a third player allows us to simulate the random move of Nature in $\Gamma(C, V)$, since, in any CCE of $\hat{\Gamma}(C, V)$, Player 1 is recommended to play a uniform distribution at infoset I^{N} and Player 3 is always told to play action a^{I}. First, if Player 3 is recommended an action $a^{\mathrm{O},\phi}$

with positive probability, then Player 2 would have an incentive to switch to action a^{O} at $I^{\varnothing}$. Moreover, assuming Player 3 is told to play a^{I}, if Player 1 is recommended to play some action $a^{\mathrm{N},\phi}$ with probability $p > \frac{1}{|C|}$, then Player 3 would have an incentive to switch to action $a^{\mathrm{O},\phi}$, as she would get $p|C| - (1-p)\frac{|C|}{|C|-1} > 0$, while she gets 0 by playing a^{I}. Finally, a reasoning similar to that for Theorem 7.4 concludes the proof. □

CHAPTER 8

Computing Coarse Correlated Equilibria in EFGs

We address the following natural question: *is it possible to compute social-welfare-maximizing CCEs in large multi-player, general-sum EFGs?*

In Section 8.1 we provide a column generation framework to compute optimal CCEs in practice, and show how to generalize it to the *hard cases* of the problem (Section 8.1.4).

Then, in Section 8.2, we relax the optimality constraint, and look for *an* ε-CCE. We design an enhanced version of CFR which computes an averaged correlated strategy which is guaranteed to converge to an approximate CCE with a bound on the regret which is sub-linear in the size of the game tree. Solutions determined via the latter algorithm will empirically prove to be nearly-socially-optimal (see the experimental evaluation in Chapter 9).

8.1 Computing a Social-Welfare-Maximizing CCE

The algorithm described in the proof of Theorem 7.3 guarantees a polynomial running time but is not appealing from a practical perspective. This is because it is based on the ellipsoid method, which is a powerful theoretical

tool, but is well-known to be inefficient in practice. In this section, we devise a practical algorithm to solve CCE-SW that may be applied to realistic settings. It is based on the simplex method and computes an optimal CCE via *column generation*. First, we describe the framework for the two-player setting without chance moves. In Sections 8.1.4 we show how to adapt it to the case of EFGs with $|\mathcal{P}| > 2$, potentially with chance moves.

8.1.1 A Practical Algorithm

Starting from LP (7.7)-(7.11), we rewrite its constraints in standard form as $M\lambda = b$, where $b^\top = (1, 0, \dots, 0)$ is a vector of dimension $(|Q_1|+|Q_2|+3)$, and λ is the variables vector

$$\lambda^\top = (\underbrace{x(\sigma_1, \sigma_2), \dots}_{|\Sigma|}, v_1^\top, v_2^\top, s_1^\top, s_2^\top),$$

in which s_i is a $|Q_i|$-dimensional column vector of slack variables. The cost vector associated with λ is:

$$c^\top = (\underbrace{\underbrace{[u_1(\sigma_1, \sigma_2) + u_2(\sigma_1, \sigma_2)]}_{x(\sigma_1,\sigma_2)}, \dots,}_{|\Sigma|} \underbrace{0, \dots, 0}_{|\mathcal{I}_1|+|\mathcal{I}_2|+|Q_1|+|Q_2|+2}).$$

Then, we can compactly rewrite LP (7.7)-(7.11) as:

$$\max_{\lambda} \; c^\top \lambda \quad \text{s.t. } M\lambda = b. \tag{8.1}$$

Let $M_{(\cdot,j)}$ denote the j-th column of M. Letting c_j be the cost associated with the j-th component of λ, and letting c_B be the vector of costs of the basic variables, the j-th *reduced cost* is:

$$\bar{c}_j = c_j - c_\mathsf{B}^\top B^{-1} M_{(\cdot,j)}, \tag{8.2}$$

where $B := [M_{(\cdot,j')}, M_{(\cdot,j'')}, \dots]$ for indices $j', j'', \dots$ corresponding to basic variables (see Bertsimas and Tsitsiklis 1997 for further details).

A crucial subproblem is determining, given a basic feasible solution to LP (8.1), a variable with the largest reduced cost (a.k.a. primal *pricing* problem). We refer to this problem as LRC. Lemma 7.3 already implies the tractability of LRC, since it is equivalent to finding a maximally violated contraint in the dual $\mathcal{D}$. We can state the following (whose proof directly follows from Lemma 7.3).

Corollary 3. *LRC can be solved in polynomial time.*

We present two pricing oracles to solve LRC in Sections 8.1.2, 8.1.3.

To describe the overall structure of the column generation algorithm (denoted by CG), we assume to have an efficient pricer solving LRC. The algorithm works in two phases: first, it determines a basic feasible solution to LP (8.1) and, then, it iteratively improves the solution until an optimal one is found. The two phases proceed as follows:

- **Phase 1: finding a feasible point**. A basic feasible solution to CCE-SW is determined by solving an auxiliary problem (depicted in LP (8.3)-(8.7)) with artificial variables, where a new variable a_i is introduced for each equality constraint and their sum is minimized in the objective function. We have:

$$\min_{\substack{x\geqslant 0, v_1, v_2 \\ a_{1,i}\geqslant 0, a_2 \geqslant 0 \\ s_i \geqslant 0}} \quad \sum_{i\in\mathcal{P}} a_{1,i} + a_2 \tag{8.3}$$

$$\text{s.t.} \quad \sum_{(\sigma_1,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) r_{\sigma_1}^\top u_i r_{\sigma_2} + a_{1,i} = f_i^\top v_i \qquad \forall i \in \mathcal{P} \tag{8.4}$$

$$F_1^\top v_1 - s_1 - u_1 \left(\sum_{\sigma_2\in\Sigma_2} \left(\sum_{\sigma_1\in\Sigma_1} x(\sigma_1,\sigma_2) \right) r_{\sigma_2} \right) = 0 \tag{8.5}$$

$$F_2^\top v_2 - s_2 - u_2^\top \left(\sum_{\sigma_1\in\Sigma_1} \left(\sum_{\sigma_2\in\Sigma_2} x(\sigma_1,\sigma_2) \right) r_{\sigma_1} \right) = 0 \tag{8.6}$$

$$a_2 + \sum_{(\sigma_2,\sigma_2)\in\Sigma} x(\sigma_1,\sigma_2) = 1, \tag{8.7}$$

  where $u_i \in \mathbb{R}^{|\times_{i\in\mathcal{P}} Q_i|}$ is the sequence-form utility matrix of player i. We refer to LP (8.3)-(8.7) as the auxiliary master problem (AUX-MASTER). When the above LP is restricted to a subset of plans $\hat{\Sigma} = \hat{\Sigma}_1 \times \hat{\Sigma}_2$, with $\hat{\Sigma}_i \subseteq \Sigma_i$ for each $i \in \mathcal{P}$, we write AUX-MASTER($\hat{\Sigma}$).

  If some artificial variable with index j is found in the optimal basis of the auxiliary problem, we can compute in polynomial time a variable j' of the original problem to replace it. It is enough to either maximize or minimize $\mathbf{e}_j^\top B^{-1} M_{(\cdot,j)}$, and this problem can be solved via Algorithm 7.1.

- **Phase 2: finding an optimal solution**. Starting from a basic feasible

Algorithm 8.1 CCE-SW via column-generation (CG) for Γ

function CG(Γ)
  ph $\leftarrow 1$ ▷ Phase of the algorithm
  $\hat{\Sigma} \leftarrow \varnothing$ ▷ Set of currently available joint plans
  violation $\leftarrow \infty$ ▷ Highest constraints' violation
  while (violation $> \varepsilon$ **or** ph $\neq 2$) **do**
    if ph $= 1$ **then** Solve AUX-MASTER($\hat{\Sigma}$)
    else $x^* \leftarrow$ Solve MASTER($\hat{\Sigma}$)
    end if
    $\psi \leftarrow$ compute dual variables
    if ph $= 1$ **then** $(\hat{\sigma}, \text{violation}) \leftarrow$ AUX-PRICER(Γ, ψ)
    else $(\hat{\sigma}, \text{violation}) \leftarrow$ PRICER(Γ, ψ)
    end if
    if violation $> \varepsilon$ **then**
      $\hat{\Sigma} \leftarrow \hat{\Sigma} \bigcup \{\hat{\sigma}\}$
    else
      if ph $= 1$ **then** ph $\leftarrow 2$
      end if
    end if
  end while
  return x^*
end function

solution, the algorithm iteratively improves it until an optimal solution is found. At each iteration, LP (8.1) (a.k.a. the master problem) is solved on a subset of plans $\hat{\Sigma} \subseteq \Sigma$ (we write MASTER($\hat{\Sigma}$)). Then, a new variable (*i.e.*, a new column of M) to enter $\hat{\Sigma}$ has to be determined as the variable with the highest reduced cost. We can solve this problem efficiently through the pricing oracles (see Sections 8.1.2 and 8.1.3).

Notice that if we were to solve CCE-SW with a standard implementation of the simplex method, we would have to compute the reduced cost of all the nonbasic variables to find a new one to enter the basis (which would require exponential time in the size of the game).

The fundamental steps of the algorithm are highlighted in Algorithm 8.1, where ε is a small (close to 0) cut-off value needed for numerical reasons, and *violation* is the highest constraints' violation at the current iteration. The algorithm solves master problems restricted to $\hat{\Sigma} \subseteq \Sigma$, which is built iteratively. Two possible implementation of PRICER and AUX-PRICER oracles are described in the following two subsections.

Algorithm 8.2 `Poly-time pricing oracle for` LRC

1: **function** P-LRC(Γ)
2: $V \leftarrow \varnothing$ ▷ Set containing variables' indices
3: $\forall j, \bar{c} \leftarrow -\infty$
4: $\psi \leftarrow c_{\mathsf{B}}^{\top} B^{-1}$ ▷ $\psi = (\bar{\beta}_1, \bar{\beta}_2, \bar{\alpha}_1, \bar{\alpha}_2, \bar{\gamma})$
5: **for** $j \in \{|\Sigma| + 1, \ldots, |c|\}$ **do** ▷ Explicit computation for poly-sized subset of var.
6: $\bar{c}_j \leftarrow c_j - \psi M_{(\cdot,j)}$
7: $V \leftarrow V \bigcup \{j\}$
8: **end for**
9: **for** $z \in Z$ **do**
10: $\hat{\sigma}_1 \leftarrow$ `constrained-plan search` for $(z, \bar{\beta}_1)$ (Alg.7.1)
11: $\hat{\sigma}_2 \leftarrow$ `constrained-plan search` for $(z, \bar{\beta}_2)$ (Alg.7.1)
12: $j \leftarrow$ index of $x(\hat{\sigma}_1, \hat{\sigma}_2)$ in c
13: $\bar{c}_j \leftarrow c_j - \psi M_{(\cdot,j)}$
14: $V \leftarrow V \bigcup \{j\}$
15: **end for**
16: $\hat{\sigma}^* \leftarrow$ joint plan corresponding to $\arg\max_{j \in V} \bar{c}_j$
17: **return** $(\hat{\sigma}^*, \max_j \bar{c}_j)$
18: **end function**

8.1.2 Polynomial-Time Pricing Oracle

Corollary 3 already states that there exists a polynomial-time oracle for LRC. We devise a poly-time oracle (P-LRC) based on Algorithm 7.1, that can be employed in both phases of Algorithm 8.1 (as `PRICER` and `AUX-PRICER`).

Let us focus on Phase 2 (Phase 1 can be treated analogously). Algorithm 8.2 describes P-LRC. Given a basic feasible solution to LP (8.1), $c_{\mathsf{B}}^{\top} B^{-1}$ is a row vector of dual variables ψ, which can be computed in polynomial time (Line 4) given it has dimension dimension $(|Q_1| + |Q_2| + 3)$. By following the notation of Lemma 7.3, let $\psi = (\bar{\beta}_1, \bar{\beta}_2, \bar{\alpha}_1, \bar{\alpha}_2, \bar{\gamma})$.

The reduced costs of variables v_i and s_i, for each $i \in \mathcal{P}$, can be computed explicitly via the definition, since their number is polynomial in the size of the EFG (Line 5). Then, Algorithm 8.2 has to evaluate the reduced costs of x variables. P-LRC enumerates the outcomes of the game (Line 9). Each $(\sigma_1, \sigma_2) \in \Sigma$ leading to z has the same c_j and, therefore, we are left with the problem of minimizing $\psi M_{(\cdot,j)}$, which amounts to minimize $(\bar{\beta}_1^{\top} u_1 r_{\sigma_2} + r_{\sigma_1}^{\top} u_2 \bar{\beta}_2)$. The problem can be split into a subproblem per player, and solved independently through Algorithm 7.1 (Line 10 and 11). By applying this procedure for each of the outcomes and selecting, among the resulting pairs, one with the largest reduced cost, P-LRC determines the new variable entering the basis in polynomial time.

P-LRC guarantees each phase of Algorithm 8.1 to run in polynomial time. However, the bottleneck of this approach is that, at each iteration, P-LRC has to traverse the EFG twice for each $z \in Z$, which is not practical for large EFGs. To circumvent this issue, the next subsection describes a second oracle based on mixed-integer linear programming (see Chapter 9 for an experimental comparison between the two approaches).

8.1.3 A MILP Pricing Oracle

We describe an oracle (MI-LRC) for computing a solution to LRC by solving a Mixed-Integer Linear Program (MILP). Differently from P-LRC, MI-LRC does not need an explicit enumeration of the terminal nodes, and it can be extended to EFGs with chance and more than two players (Section 8.1.4).

The crucial difference between MI-LRC and P-LRC is in the way they handle the inspection of the reduced costs associated with the x variables. In MI-LRC, lines 9-14 of Algorithm 8.2 are substituted with an MILP, which is described next.

Let R_i be a $(|Q_i| \times |Z|)$ matrix such that $R_i(q_i, z) = 1$ if $q_i \in Q_i^z$, and $R_i(q_i, z) = 0$ otherwise, and let t be a $|Z|$-dimensional vector of binary variables. Let $\psi = (\bar{\beta}_1, \bar{\beta}_2, \bar{\alpha}_1, \bar{\alpha}_2, \bar{\gamma})$ and $u_i \in \mathbb{R}^{|\times_{i \in \mathcal{P}} Q_i|}$. The variable with the largest reduces cost is determined with the following MILP:

$$\max_{\substack{t \in \{0,1\}^{|Z|} \\ r_i \in \mathbb{R}^n_+}} \quad \left((1-\bar{\alpha}_1) r_1^\top - \bar{\beta}_1^\top\right) u_1 r_2 + r_1^\top u_2 \left((1-\bar{\alpha}_2) r_2 - \bar{\beta}_2\right) \tag{8.8a}$$

$$\text{s.t.} \quad F_i r_i = f_i \qquad \forall i \in \mathcal{P} \tag{8.8b}$$

$$r_i \geqslant R_i t \qquad \forall i \in \mathcal{P} \tag{8.8c}$$

$$\mathbb{1}^\top t = 1 \tag{8.8d}$$

The objective function (8.8a) follows from the definition of the reduced costs (we are looking for a variable whose dual constraint is maximally violated). Constraints (8.8c) force the realization plans to select with probability 1 the sequences on the path to the selected outcome. While the objective function contains quadratic terms, they only involve binary variables. Therefore, it can be restated as a linear function after introducing a new variable and four linear constraints per bilinear term according to the standard *McCormick envelope* reformulation [127].

An optimal realization plan r_i^*, solution to (8.8a)-(8.8d), may not be pure.[1] Nevertheless, there always exists a pair of pure realization plans

[1] That is, there may exists $q_i \in Q_i$ such that $r_i^*(q_i) \in (0, 1)$.

leading to the same z, and granting the same objective value, which can be determined following a similar reasoning to Algorithm 7.1 while considering only sequences played with strictly positive probability in r_i^*.

Once an optimal pair of pure realization plans has been determined, MI-LRC computes the reduced cost associated with it (Equation (8.2)) and compares it other variables' costs (Line 17 of Algorithm 8.2). When considering Phase 1, the only difference in `AUX-PRICER` is that objective function (8.8a) does not contain the term $r_1^\top (u_1 + u_2) r_2$.

8.1.4 Generalizing The Framework

In this section, we highlight some extensions of the algorithmic framework presented in Section 8.1.1. First, let us remark that the previous sections focus on solving CCE-SW but our results apply even when searching for a CCE maximizing any linear combination of players' utilities.

The column generation approach can be generalized to the *hard cases* of the problem (namely, when $|\mathcal{P}| > 2$ or chance is present). The two master problems (*i.e.*, `MASTER` and `AUX-MASTER`) remain the same, while pricers have to be suitably modified. As one could expect from Theorem 7.2 and Corollary 2, P-LRC cannot be extended to these settings (since Algorithm 7.1 cannot be applied). Therefore, we start from MI-LRC and show how to adapt it, first to the setting of two-players EFGs with chance moves, and then to EFGs with $|\mathcal{P}| > 2$.

Two-Player EFGs with Chance Moves. The reasoning behind P-LRC cannot be extended to this setting since the first term of the objective function of the separation problem (Equation (7.12)) is no longer constant once an outcome is fixed.[2] Let Q_c be the set of sequences of the chance player, and $(q_1^z, q_2^z, q_c^z) \in Q_1 \times Q_2 \times Q_c$ be the unique tuple of the terminal sequences leading to outcome z. The crucial issue is that, given $z \in Z$, there may exist some $z' \in Z$, $z' \neq z$, reachable through $(q_1^z, q_2^z, q_c^{z'})$, with $q_c^{z'} \neq q_c^z$.

We can circumvent this issue by adapting MI-LRC as follows. First, for each $i \in \mathcal{P}$ we compute the utility matrices $u_{i,\pi_c} \in \mathbb{R}^{|Q_1| \times |Q_2|}$, obtained by marginalizing u_i with respect to π_c. Formally, denoting by $r_c \in [0,1]^{|Q_c|}$ the (fixed) realization plan of the chance player realization-equivalent to π_c, for each $(q_1, q_2) \in Q_1 \times Q_2$ we have $u_{i,\pi_c}(q_1, q_2) = \sum_{q_c \in Q_c} r_c(q_c) u_i(q_1, q_2, q_c)$. Objective function (8.8a) is then modified by substituting each u_i with u_{i,π_c}. Moreover, upon denoting by $R_c \in \mathbb{R}^{|Q_c| \times |Z|}$ the matrix defined analogously

[2] A similar procedure cannot be adapted even when marginalizing utility matrices with respect to π_c. Enumerating over pairs of terminal sequences is not feasible either, leaving the first term of (7.12) underspecified (too see this, it is enough to consider an instance of the construction employed for Theorem 7.2).

to R_i, it suffices to substitute constraint (8.8d), with a new family of constraints, which reads:

$$R_{c,(q_c,\cdot)}t = 1 \qquad \forall q_c \in Q_c,$$

where $R_{c,(q_c,\cdot)}$ denotes row q_c of R_c. Intuitively, t now selects a subset of terminal nodes, which are those that may be reached with the selected pair of plans (σ_1, σ_2). The new family of constraints forces Player 1 and 2's plans to be completely specified.

EFGs with $|\mathcal{P}| > 2$. In the case with $|\mathcal{P}| > 2$ and no chance moves, Lemma 7.3 does not hold as one would have to determine the joint best response of (at least) two players at a time, which is known to be NP-hard [167].

Let us focus on the case of EFGs with $|\mathcal{P}| = 3$, and no chance moves (the oracle can be easily adapted to take chance moves into account). If the EFG has players $i, j, l \in \mathcal{P}$, we denote by $u_{i,r_j} \in \mathbb{R}^{|Q_i|\times|Q_l|}$ the sequence-form utility matrix of player i marginalized with respect to r_j, that is, for each pair of terminal sequences $(q_i, q_l) \in Q_i \times Q_l$: $u_{i,r_j}(q_i, q_l) = \sum_{q\in Q_j} r_j(q)u_i(q_i, q_l, q)$. The objective function that needs to be maximized (building on (8.8a)) is:

$$r_1^\top \Big((1-\bar{\alpha}_1)u_{1,r_3} + (1-\bar{\alpha}_2)u_{2,r_3} + (1-\bar{\alpha}_3)u_{3,r_3} \Big) r_2 + \\ - \bar{\beta}_1^\top u_{1,r_3} r_2 - r_1^\top u_{2,r_3}\bar{\beta}_2 - r_2^\top u_{3,r_1}\bar{\beta}_3. \quad (8.9)$$

The first term of (8.9) is constant once a terminal node has been fixed. The remaining terms amount to computing joint best responses. For example, the term $-\bar{\beta}_1^\top u_{1,r_3} r_2$ consists of Player 2's and 3's best responses given the gradient $\bar{\beta}_1^\top u_1 \in \mathbb{R}^{|Q_2|\times|Q_3|}$. Notice that this is a *joint* best-response problem in the sense that the term to be minimized/maximized depends on the choice of both players. However, each best-responding player i follows a marginal strategy from $\Delta(\Sigma_i)$, and not a correlated strategy from $\Delta(\Sigma_2 \times \Sigma_3)$.

We obtain the following formulation:

$$\max_{\substack{\forall i\in\mathcal{P}, r_i\geqslant 0\\ t\in\{0,1\}^{|Z|}\\ \forall i\in\mathcal{P}, t_i\in\{0,1\}^{|Z|}\\ \forall i\in\mathcal{P}, \delta_i\in\{0,1\}^{|Q_i|}}} \Bigg(r_1^\top \left((1-\bar{\alpha}_1)u_{1,r_3} + (1-\bar{\alpha}_2)u_{2,r_3} + (1-\bar{\alpha}_3)u_{3,r_3}\right) r_2 + \\ - \bar{\beta}_1^\top u_{1,r_3} r_2 - r_1^\top u_{2,r_3}\bar{\beta}_2 - r_2^\top u_{3,r_1}\bar{\beta}_3 \Bigg) \quad (8.10a)$$

$$F_i r_i = f_i \qquad \forall i \in \mathcal{P} \qquad (8.10b)$$

$$\mathbb{1}^\top t = 1 \qquad (8.10c)$$

$$t_i \geqslant t \qquad \forall i \in \mathcal{P} \qquad (8.10d)$$

$$R_i t_i = \mathbb{1} \qquad \forall i \in \mathcal{P} \qquad (8.10e)$$

$$\delta_i \geqslant \frac{1}{|Z|} R_i t_j \qquad \forall i \in \mathcal{P}, \forall j \in \mathcal{P}\backslash\{i\} \qquad (8.10f)$$

$$r_i \geqslant \delta_i \qquad \forall i \in \mathcal{P} \qquad (8.10g)$$

The oracle employs $(|\mathcal{P}|+1)\ |Z|$-dimensional vectors of binary variables. Vector $t \in \{0,1\}^{|Z|}$ selects a single outcome (constraint (8.10c)), which determines the value of the first term of the objective function. Each best-response term is associated with a vector of variables $t_i \in \{0,1\}^{|Z|}$. As an example, t_1 determines the set of outcomes that may be reached via a feasible (constraint (8.10b)) choice of r_2 and r_3. A reachable outcome for each of Player 1's sequences is chosen (constraint (8.10e)), and these choices have to be compatible with the outcome selected via t (constraint (8.10d)). Finally, constraints (8.10e) and (8.10g) force $r_i(q) = 1$ if q lies on the path to at least one outcome selected bt t_j, $\forall j \in \mathcal{P}, j \neq i$. That is: $R_{i,(q,\cdot)} t_j \geqslant 1 \implies r_i(q) \geqslant 1$, for each $i, j \in \mathcal{P}, i \neq j$.

Objective function (8.10a) may be rewritten with a slightly different notation to get an LP. Let $u'_i \in \mathbb{R}^{\times_{j\in\mathcal{P}}|Q_j|}$ be a matrix such that, for each $(q_1,q_2,q_3) \in \times_{i\in\mathcal{P}} Q_i$, $u'_i(q_1,q_2,q_3) = \bar{\beta}_i(q_i)u_i(q_1,q_2,q_3)$, with $i \in \mathcal{P}$. The utility function encoded by u'_i can be equivalently represented with a vector associating to each z the value $u'_i(q_1^z, q_2^z, q_3^z)$, which contains all relevant information since u'_i contains an entry only for tuple of sequences identifying a terminal node. With an abuse of notation, we write $u'_i(z) = u'_i(q_1^z, q_2^z, q_3^z)$. Then, by employing the binary variables already defined, we obtain:

$$\max \left(\sum_{z\in Z} \left(t(z) \left(\sum_{i\in\mathcal{P}} (1-\bar{\alpha}_i) u_i(z) \right) - \sum_{i\in\mathcal{P}} t_i(z) u'_i(z) \right) \right),$$

which is the linear objective function we exploit in practice.

8.2 Approaching the Set of CCEs

This section focuses on the problem of computing an approximate CCE in multi-player (*i.e.*, $|\mathcal{P}| > 2$), general-sum EFGs. In the normal-form setting, any *Hannan consistent* regret-minimizing procedure for simplex decision spaces may be employed to approach the set of CCEs [24, 48]—the most common of such techniques is *regret matching* (RM) [22, 92]. However, approaching the set of CCEs in sequential games is more demanding. One could represent the sequential game with its equivalent normal form and apply RM to it, but this would result in a guarantee on the cumulative regret which would be exponential in the size of the game tree. Thus, reaching a good approximation of a CCE could require an exponential number of iterations.

The problem of designing learning algorithms avoiding the construction of the normal form has been successfully addressed in sequential games for the two-player, zero-sum setting (see Section 2.5). This is done by decomposing the overall regret locally at the information sets of the game [67]. However, these kind of algorithms (*e.g.*, CFR/CFR+) work with players' behavioral strategies rather than with correlated strategies, and, thus, they are not guaranteed to converge to a CCE in general-sum games, even with two players.

We start by pointing out simple examples where CFR-like algorithms available in the literature cannot be directly employed for our purpose, as they only provide players' average strategies whose product is not guaranteed to converge to an approximate CCE (Section 8.2.1). Then, in Section 8.2.2, we show how CFR can be easily adapted to approach the set of CCEs in multi-player, general-sum sequential games by resorting to sampling procedures (we call the resulting, naive algorithm CFR-S). Finally, we design an enhanced version of CFR (called CFR-Jr) which computes an average correlated strategy which is guaranteed the convergence to an approximate CCE with a bound on the regret that is sub-linear in the size of the game tree (Section 8.2.3). An experimental comparison between CFR-S and CFR-Jr is postponed to Chapter 9.

8.2.1 When CFR is not Enough

When players follow strategies recommended by a regret minimizer, the *empirical frequency of play* approaches the set of CCEs [48]. Suppose that, at time t, the players play a joint normal-form plan $\sigma^t \in \Sigma$ drawn according to their current strategies. Then, the empirical frequency of play after T iterations is defined as the joint probability distribution $\bar{x}^T \in \mathcal{X}$

	σ_L	σ_R
σ_L	$1, 1$	$1, 0$
σ_R	$0, 1$	$1, 1$

1,0	0,1	0,0
0,0	2,0	0,1
0,1	0,0	1,0

Figure 8.1: Left*: Game where $\bar{x}_1^T \otimes \bar{x}_2^T$ does not converge to a CCE.* Right*: A simple variation of the Shapley game where the outer product of the average strategies $\bar{x}_1^T \otimes \bar{x}_2^T$ obtained by RM does not converge to a CCE as $T \to \infty$ (see Figure 8.2).*

such that $\bar{x}^T(\sigma) := \frac{|t \leqslant T : \sigma^t = \sigma|}{T}$ for every $\sigma \in \Sigma$. However, vanilla CFR and its most popular variations (such as, *e.g.*, CFR+ [162] and DCFR [31]) do not keep track of the empirical frequency of play, as they only keep track of the players' average behavioral strategies. This ensures that the strategies are compactly represented, but it is not sufficient to recover a CCE in multi-player, general-sum games. Indeed, it is possible to show that, even in normal-form games, if the players play according to some regret-minimizing strategies, then the product distribution $x \in \mathcal{X}$ resulting from players' (marginal) average strategies may not converge to a CCE. In order to see this, we provide the following simple example.

Example 8. *Consider the two-player normal-form game depicted on the left in Figure 8.1. At iteration t, let players' strategies x_1^t, x_2^t be such that $x_1^t(\sigma_L) = x_2^t(\sigma_L) = (t+1) \mod 2$. Clearly, $u_1^t(x^t) = u_2^t(x^t) = 1$ for any t. For both players, at iteration t, the regret of not having played σ_L is 0, while the regret of σ_R is -1 if and only if t is even, otherwise it is 0. As a result, after T iterations, $R_1^T = R_2^T = 0$, and, thus, x_1^t and x_2^t minimize the cumulative external regret. Players' average strategies $\bar{x}_1^T = \frac{1}{T}\sum_{t=1}^T x_1^t$ and $\bar{x}_2^T = \frac{1}{T}\sum_{t=1}^T x_2^t$ converge to $(\frac{1}{2}, \frac{1}{2})$ as $T \to \infty$. However, $x \in \mathcal{X}$ such that $x(\sigma) = \frac{1}{4}$ for every $\sigma \in \Sigma$ is not a CCE of the game. Indeed, a player is always better off playing σ_L, obtaining a utility of 1, while she only gets $\frac{3}{4}$ if she chooses to stick to x. We remark that $\bar{x}^T$ converges, as $T \to \infty$, to $x \in \mathcal{X} : x(\sigma_L, \sigma_L) = x(\sigma_R, \sigma_R) = \frac{1}{2}$, which is a CCE.*

The example above employs handpicked regret-minimizing strategies, but similar examples can be easily found when applying common regret minimizers. As an illustrative case, Figure 8.1 shows, on the right, a simple variation of the Shapley game [154]. We show that, in this simple two-player game, the outer product of the average strategies $\bar{x}_1^T \otimes \bar{x}_2^T$ obtained via RM does not converge to a CCE as $T \to \infty$ (Figure 8.2). Specifically, when evaluating $\bar{x}_1^T \otimes \bar{x}_2^T$, ε of the ε-CCE has a cyclic behavior and does not converge to zero. It is clear that the same issue may happen when directly applying CFR to general-sum EFGs (see Chapter 9).

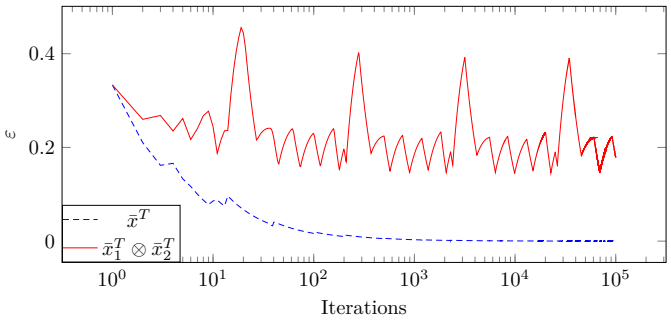

Figure 8.2: *Quality of the ε-CCEs attained by $\bar{x}^T$ and $\bar{x}_1^T \otimes \bar{x}_2^T$ on the game depicted on the right of Figure 8.1.*

8.2.2 CFR with Sampling (CFR-S)

Motivated by the previous examples, we describe a simple variation of CFR guaranteeing approachability to the set of CCEs even in multi-player, general-sum EFGs.

Vanilla CFR proceeds as follows (see Section 2.5 for the details): for each iteration t, and for each infoset $I \in \mathcal{I}_i$, player i observes the realized utility for each action $a \in A(I)$, and then computes $\pi_i^t(I) \in \Delta(A(I))$ according to standard RM. Once $\pi_i^t(I)$ has been computed, it is used by the regret minimizers of infosets on the path from the root to I so as to compute observed utilities.

We propose CFR *with sampling* (CFR-S) as a simple way to keep track of the empirical frequency of play. The basic idea is letting each player i, at each t, draw σ_i^t according to her current strategy. Algorithm 8.3 describes the structure of CFR-S, where function RECOMMEND builds a normal-form plan σ_i^t by sampling, at each $I \in \mathcal{I}_i$, an action in $A(I)$ according to π_i^t computed via RM, and UPDATE updates the average regrets local to each regret minimizer by propagating utilities according to σ_i^t. Each player i experiences utilities depending, at each t, on the sampled plans σ_{-i}^t (Line 6). Joint normal form plans $\sigma^t := (\sigma_i^t, \sigma_{-i}^t)$ can be easily stored to compute the empirical frequency of play. We state the following (see Appendix A for detailed proofs):

Theorem 8.1. *The empirical frequency of play $\bar{x}^T$ obtained with CFR-S converges to a CCE almost surely, for $T \to \infty$.*

Moreover, the cumulative regret grows as $O(T^{-1/2})$. This result is in line with the approach of Hart and Mas-Colell [92] for normal-form games. Despite its simplicity, we show (see Chapter 9 for an experimental evaluation)

Algorithm 8.3 `CFR-S for player` i

```
1: function CFR-S(Γ,i)
2:     Initialize a regret minimizer for each I ∈ 𝓘_i
3:     t ← 1
4:     while t < T do
5:         σ_i^t ← RECOMMEND(I_∅)
6:         Observe u_i^t(σ_i) := u_i(σ_i, σ_{-i}^t)
7:         UPDATE(I_∅, σ_i^t, u_i^t)
8:         t ← t + 1
9:     end while
10: end function
```

that it is possible to achieve better perfomances via a smarter reconstruction technique that keeps CFR deterministic, avoiding any sampling step.

8.2.3 CFR with Joint Distribution Reconstruction (CFR-Jr)

We design a new method—called *CFR with joint distribution reconstruction* (CFR-Jr)—to enhance CFR so as to approach the set of CCEs in multi-player, general-sum EFGs. Differently from the naive CFR-S algorithm, CFR-Jr does not sample normal-form plans, avoiding any stochasticity.

The main idea behind CFR-Jr is to keep track of the average joint probability distribution $\bar{x}^T \in \mathcal{X}$ arising from the regret-minimizing strategies built with CFR. The key component of CFR-Jr is a polynomial-time algorithm which constructs, at each iteration, the players' normal-form strategies by working on the game tree, avoiding to build the (exponential-sized) normal-form representation.

Formally, $\bar{x}^T = \frac{1}{T}\sum_{t=1}^{T} x^t$, where $x^t \in \mathcal{X}$ is the joint probability distribution defined as the product of the players' normal-form strategies at iteration t. At each t and for each $i \in \mathcal{P}$, CFR-Jr computes π_i^t with CFR's update rules, and then constructs a strategy $x_i^t \in \mathcal{X}_i$ which is realization equivalent (*i.e.*, it induces the same probability distribution on the terminal nodes) to π_i^t. We do this efficiently by directly working on the game tree, without resorting to the normal-form representation. Strategies x_i^t are then employed to compute x^t.

CFR-Jr. In Algorithm 8.4 we provide a sketch of the CFR-Jr algorithm, which uses an implementation of the vanilla CFR algorithm as a subroutine (denoted by CFR).

CFR-Jr maintains a variable $\bar{x}$ which stores the sum of the joint probability distributions x^t. CFR-Jr executes, at each t, an iteration of the CFR algorithm (Line 5). In particular, the CFR subroutine executes a step of

Algorithm 8.4 `CFR-Jr`

1: **function** CFR-JR(Γ)
2: $\bar{x} \leftarrow \mathbf{0}$, $t \leftarrow 0$
3: **while** $t < T$ **do**
4: **for all** $i \in \mathcal{P}$ **do**
5: $\pi_i^t \leftarrow \text{CFR}(\Gamma, i)$
6: $x_i^t \leftarrow$ NF-STRATEGY-RECONSTRUCTION(π_i^t)
7: **end for**
8: $\bar{x} \leftarrow \bar{x} + \bigotimes_{i \in \mathcal{P}} x_i^t$ ▷ $\bigotimes_{i \in \mathcal{P}} x_i^t$ is the joint distribution x^t defined as the product of the players' normal-form strategies
9: $t \leftarrow t + 1$
10: **end while**
11: **return** $\bar{x}^T = \bar{x}/T$
12: **end function**

Algorithm 8.5 `Reconstruct` x_i `from` π_i

1: **function** NF-STRATEGY-RECONSTRUCTION(π_i)
2: $\mathbf{X} \leftarrow \varnothing$ ▷ $\mathbf{X}$ is a dictionary defining x_i
3: $\omega(z) \leftarrow \rho^{\pi_i}(z) \quad \forall z \in Z$
4: **while** $\omega > 0$ **do**
5: $\bar{\sigma}_i \leftarrow \arg\max_{\sigma_i \in \Sigma_i} \min_{z \in Z(\sigma_i)} \omega(z)$
6: $\bar{\omega} \leftarrow \min_{z \in Z(\bar{\sigma}_i)} \omega_i(z)$
7: $\omega \leftarrow \omega - \bar{\omega}\, \rho^{\bar{\sigma}_i}$
8: $\mathbf{X} \leftarrow \mathbf{X} \cup (\bar{\sigma}_i, \bar{\omega})$
9: **end while**
return x_i built from $\mathbf{X}$'s pairs
10: **end function**

vanilla CFR, including the update of regrets and behavioral strategies. In addition, at each iteration t, CFR-Jr constructs normal-form strategies x_i^t (one for each player $i \in \mathcal{P}$) which are realization equivalent to the behavioral strategies π_i^t obtained with CFR (Line 6). Then the product x^t of the players' normal-strategies is computed and added to $\bar{x}$ (Line 8). Notice that $\bar{x}$ is not used by the CFR subroutine to update the players' strategies and regrets. Finally, CFR-Jr returns $\bar{x}$ divided by T, which represents the average $\bar{x}^T$.

Strategy Reconstruction. Algorithm 8.5 shows a polynomial-time procedure to compute a normal-form strategy $x_i \in \mathcal{X}_i$ realization equivalent to a given behavioral strategy π_i. The algorithm maintains a vector ω (*i.e.*, a realization-form strategy as in Definition 5.2) which is initialized with the probabilities of reaching each terminal node by playing π_i (Line 3), and it works by iteratively assigning probability to normal-form plans so as to

induce the same distribution of ω over Z. In order for this to work, at each iteration, the algorithm must pick a normal-form plan $\bar{\sigma}_i \in \Sigma_i$ which maximizes the minimum (remaining) probability $\omega(z)$ over the terminal nodes $z \in Z(\bar{\sigma}_i)$ reachable when playing $\bar{\sigma}_i$ (Line 5). Then, for each $z \in Z(\bar{\sigma}_i)$, $\omega(z)$ is decreased by $\bar{\omega}$, (Line 7), and $\bar{\sigma}_i$ is assigned probability $\bar{\omega}$ in x_i (Line 8). The algorithm terminates when the vector ω is zeroed, returning a normal-form strategy x_i realization equivalent to π_i.

This is formally stated by the following result, which also provides a polynomial upper bound on the running time of the algorithm, and on the support size of the returned normal-form strategy x_i. [3]

Theorem 8.2. *Algorithm 8.5 outputs a normal-form strategy $x_i \in \mathcal{X}_i$ realization equivalent to the given behavioral strategy π_i. It runs in time $O(|Z|^2)$, and x_i has support size of at most $|Z|$.*

Intuitively, the result in Theorem 8.2 relies on the crucial observation that, at each iteration t, there is at least one terminal node $z \in Z$ whose corresponding reach probability $\omega(z)$ is zeroed at t (see Appendix B for a complete proof). The algorithm is guaranteed to terminate since each $\omega(z)$ is guaranteed to remain $\geqslant 0$ at every iteration, which is the case given how the normal-form plans are selected (Line 5), and since the game has perfect recall. This guarantees that the algorithm eventually terminates in at most $|Z|$ iterations. Moreover, the size of a reconstructed normal-form strategy profile is upper bounded by the number of leaves reached with strictly positive probability when following the original behavioral profile, which is, in its turn, upper bounded by the size of the support of the latter strategy (*i.e.*, the number of non-zero action probabilities). Then, the reconstructed normal-form strategy for a single player has size which is also upper bounded by the size of the original behavioral strategy.

Finally, the following theorem (whose full proof can be found in Appendix B) proves that the average distribution $\bar{x}^T$ obtained with CFR-Jr approaches the set of CCEs. Formally:

Theorem 8.3. *If $\frac{1}{T} R_i^T \leqslant \varepsilon$ for each player $i \in \mathcal{P}$, then $\bar{x}^T$ obtained with CFR-Jr is an ε-CCE.*

This is a direct consequence of the connection between regret-minimizing procedures and CCEs, and of the fact that $\bar{x}^T$ is obtained by averaging the products of normal-form strategies which are realization equivalent to regret-minimizing behavioral strategies obtained via CFR. Moreover, we remark that the regret experienced by CFR-Jr is bounded by $\Delta|\mathcal{I}_i|\sqrt{|A_i|}/\sqrt{T}$,

[3] Given a normal-form strategy $x_i \in \mathcal{X}_i$, its *support* is defined as the set of $\sigma_i \in \Sigma_i$ such that $x_i(\sigma_i) > 0$.

as in CFR [180], because our reconstruction procedure does not alter the way in which regret is minimized.

CHAPTER 9

Experimental Evaluation

This chapter is devoted to an experimental evaluation of the techniques to compute CCEs presented in the previous chapter. In Section 9.1, we evaluate the performance of techniques to compute social-welfare-maximizing CCEs. We study the behavior of the column-generation Algorithm 8.1 with different pricing oracles. In Section 9.2 we relax the optimality requirement, and focus on the problem of computing an approximate CCE via the techniques of Section 8.2. We evaluate the scalability of CFR-S and CFR-Jr on standard testbeds. While both algorithms solve instances which are orders of magnitude larger than those solved by the column generation algorithm, CFR-Jr dramatically outperforms CFR-S. Moreover, CFR-Jr proves to be a good heuristic to compute optimal CCEs, returning nearly-socially-optimal solutions in all the instances of our testbeds.

9.1 Computing an Optimal CCE

We compare the performance of our column generation method with the two different oracles P-LRC (Section 8.1.2) and MI-LRC (Section 8.1.3) on instances of two-player, general-sum games with and without chance moves.

	Game size			
Game	$\lvert Q_1\rvert$	$\lvert Q_2\rvert$	$\lvert H_1\rvert$	$\lvert H_2\rvert$
R5-2	20	26	10	14
R5-3	126	102	43	35
R5-4	400	404	100	102
R10-2	664	680	333	340
R12-2	2649	2697	1325	1349
R13-2	5364	5316	2682	2659
G3R	58	58	47	47
G3S	334	334	274	274
G3D	334	334	274	274

Table 9.1: *Number of infosets and sequences of the test instances.*

9.1.1 Experimental Setting

We employ the following game instances:

- Random two-player general-sum games with utilities in $(-1, 1)$. Denoting by Rd-b games of depth d and branching factor b, we generate 20 instances for each of the following configurations: R5-2, R5-3, R5-4, R10-2, R12-2, R13-2.
- *Goofspiel* game instances [148, 150]. Goofspiel is a bidding game where each player has a hand of cards numbered from 1 to K. A third stack of K cards is shuffled and used as prizes. Each turn a prize card is revealed, and each player chooses a private card to bid, with the highest card winning the current prize. After K turns, all the prizes have been dealt out and the payoff of each player is the sum of the prize cards that they have won. In these experiments, we use $K = 3$ (3 card ranks), with two different tie-breaking rules, namely, the players splitting the value of the card on the table equally (G3S) or discarding it (G3D). G3R is the variant of the game in which the order of the prize cards is known.

The instances' dimensions are summarized in Figure 9.1.

For the experiments, we employ the state-of-the-art MILP solver GUROBI (version 8.0). The computations are run on a multi-processor system equipped with 16 dual 2.6 GHz Intel Sandybridge processors and 64 GBs of RAM.

9.1.2 Results

First, we remark that the use of an LP defined directly on the normal form of a game (Definition 2.6) is impractical for every instance of our experimental

	P-LRC				MI-LRC			
Game	Phase1 (steps)	Phase2 (steps)	Time (sec)	Solved (in 12h)	Phase1 (steps)	Phase2 (steps)	Time (sec)	Solved (in 12h)
R5-2	4.6	6.8	0.3	20	4.7	2.25	0.02	20
R5-3	5.5	8.9	9.2	20	5.5	3.15	0.32	20
R5-4	8.2	12.1	439.8	20	7.5	4.8	8.1	20
R10-2	5.5	11.6	1121.8	20	5.8	6.2	23.4	20
R12-2	5	7	41421.3	1	6.2	6.3	391.7	20

Table 9.2: *Comparison of the performance with the two different oracles.*

Game	Phase1	Phase2	Time
R13-2	6.1	8.9	3368.2
G3R	7	1	0.1
G3S	64	1	100.6
G3D	6	1	1.7

Table 9.3: *Performance of MI-LRC on large two-player games and games with Nature.*

setting due to its exponential size. For instance, games like G3S and G3D contain more than $5 \cdot 10^{13}$ variables. For problems of this size, even building their LP formulation in memory is almost impossible (let alone solving it). The column generation techniques we propose completely circumvent this issue.

Table 9.2 reports the average results that we obtained on the two-player instances of class R5-2, R5-3, R5-4, R10-2, and R12-2, with both P-LRC and MI-LRC. The results obtained on the R13-2 instances, together with those for two-player games with chance, which are too large to be handled with P-LRC, are reported in Table 9.3.

First, we notice that the number of columns generated before reaching optimality is always quite small. This justifies even more the adoption of a column generation approach, since the algorithm requires only a few iterations to reach an optimal solution once a basic feasible solution is found. Moreover, the results clearly illustrate that the MI-LRC oracle allows for a dramatic improvement in the performance of the algorithm. Overall, our column generation method employing MI-LRC is able to compute a socially optimal CCE even on instances with more than 5000 cumulative sequences and 2500 information sets in less than one hour.

9.2 Learning an ε-CCE

We experimentally evaluate CFR-Jr, comparing its performance with that of CFR-S, CFR, and CG (*i.e.*, the column generation algorithm described in Algorithm 8.1) with the MILP pricing oracle. Another known algorithm to compute a CCE is by Huang and von Stengel [100] (see also Jiang and Leyton-Brown [102] for an amended version).[1] However, this algorithm relies on the ellipsoid method, which is known to be inefficient in practice [85]. Moreover, we remark that directly applying RM on the normal form is not feasible, as $|\Sigma| > 10^{20}$ even for the smallest instances of our test bed.

9.2.1 Experimental Setting

We conduct experiments on parametric instances of three-player Kuhn poker games [117], three-player Leduc hold'em poker games [157], two and three-player Goofspiel games [148], some randomly generated general-sum EFGs, and a variation of the Shapley game. The two-player zero-sum versions of these games are standard benchmarks for imperfect-information game solving. Each instance is identified by parameters p and r, which denote, respectively, the number of players and the number of ranks in the deck of cards. For example, a three-player Kuhn game with rank 4 is denoted by Kuhn3-4, or K3-4. Rp-d denotes a random game with p players, and depth of the game tree d.

A description of three-player Kuhn and Leduc poker can be found in Section 6.2. Here, we briefly describe the three-player Goofspiel variant and the different tie-breaking rules we employ for this evaluation (denoted by A, DA, DH, AL), and our variation of the Shapley game.

Goofspiel. In this game, cards rank A (low), 2, . . ., 10, J, Q, K (high). When scoring points, the Ace is worth 1 point, cards 2-10 their face value, Jack 11, Queen 12, and King 13. Goofspiel$p$-r (p is the number of players) employs $p + 1$ suits, each containing cards A, . . .,r. One suit is singled out as the prizes. The prizes are shuffled and placed between the players, with the top card turned face up. Each of the remaining suits becomes the hand of one of the players. The game proceeds in rounds. Each player selects a card from her hand, keeping her choice secret from the opponent. Once all players have selected a card, they are simultaneously revealed, and the player with the highest bid wins the prize card. We employ the following tie breaking rules to obtain different kinds of instances. Some of them

[1] The algorithm by Huang and von Stengel [100] computes an EFCE in polynomial time. Since EFCE⊆CCE, the algorithm's solution is also a CCE of the game.

(*e.g.*, *Accumulate*) are *almost* constant-sum games (*i.e.*, constant sum for all but few outcomes), while others (*e.g.*, *Discard always*) present larger differences in the sum of payoffs attainable at different terminal nodes:

- *Accumulate* (A): the prize card goes to the player that selected the highest unique card. If more players selected the same card, the prize card is taken aside and the game continues unveiling the next one: the winner (if any) of the new round will take both prize cards. The process is repeated until the tie is broken or the game ends, in which case all prize cards that have been taken aside are discarded.
- *Discard-if-all* (DA): the prize card goes to the player that selected the highest *unique* card; if all players selected the same card, the prize card is discarded.
- *Discard-if-high* (DH): if the tie is on the highest-valued card, then the prize card is discarded; otherwise, the prize card goes to the player that selected the highest unique card.
- *Discard always* (AL): the prize card is discarded on any tie, and the game goes on with the next round.

The game ends when the players terminate their cards. Players calculate their final utility by summing up the value of the prize cards they won.

Asymmetric Shapley Game. We tested CFR-Jr also on some extensive-form variants of the Shapley game, a normal-form general-sum 3x3 game introduced by Shapley [154] that has been shown to induce cyclic non-convergent behaviors in iterative algorithms such as Fictitious Play [101]. Our extensive-form asymmetric variation of the game reads as follows:

- At each stage of the game, a player has to select a number in the set $\{0, 1, 2\}$.
- Player 1 selects a number and publicly discloses it. Then, Player 2 chooses a number and writes it down, without disclosing it to the other player. Finally, Player 1 selects another number, without knowing the previous choice of Player 2.
- Let s be the sum of the three numbers that have been selected. The players' utilities are computed as follows:
  - if $s \bmod 3 = 0$, then the utility is $(0, 0)$;
  - if $s \bmod 3 = 1$, then the utility is $(1, 0)$;
  - if $s \bmod 3 = 2$, then the utility is $(0, 1)$;

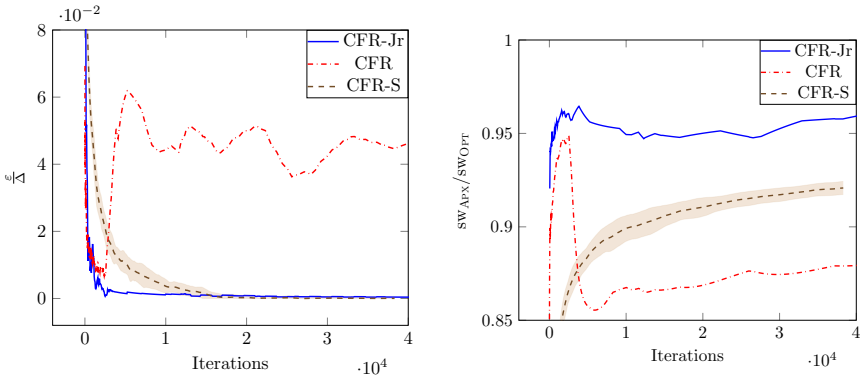

Figure 9.1: Left*: Convergence rate attained in G2-4-DA*. Right*: Social welfare attained in G2-4-DA*.

– if the first number selected by Player 1, and the number selected by Player 2 are equal in value, then the utility gained by each player is doubled.

The last step is fundamental to introduce some asymmetries in the game and ensure that a uniform joint strategy (*i.e.*, $x(\sigma) = 1/|\Sigma|$, for each σ) is not a CCE. This is a problem in the standard Shapley game as many regret minimizers employ a uniform strategy as initialization, and therefore would converge *instantly* to a CCE.

CFR, CFR-S and CFR-Jr algorithms have all been implemented in the Python 3 language, while the column generation algorithm employs the GUROBI 8.0 MILP solver to solve the pricing problems. All the experiments are run with a 24 hours time-limit on a UNIX machine with a total of 32 cores working at 2.3 GHz, equipped with 128 GB of RAM.

9.2.2 Comparison with vanilla CFR

As already discussed in Section 8.2.1, vanilla CFR [180] can converge to a CCE only in specific settings (*e.g.*, two-player, zero-sum games). In general (*i.e.*, multi-player, general-sum) games tracking each player's average strategy is not enough to compute equilibria that may not be the result of a product of marginal strategies. While this is well understood in theory (there is no guarantee of convergence for CFR in general-sum games, even if they have two-player and no chance), we provide some practical examples where this behavior is evident.

Figure 9.1 shows the results for an instance of Goofspiel (with the DA tie-breaking rule), and Figure 9.2 shows the results for our asymmetric

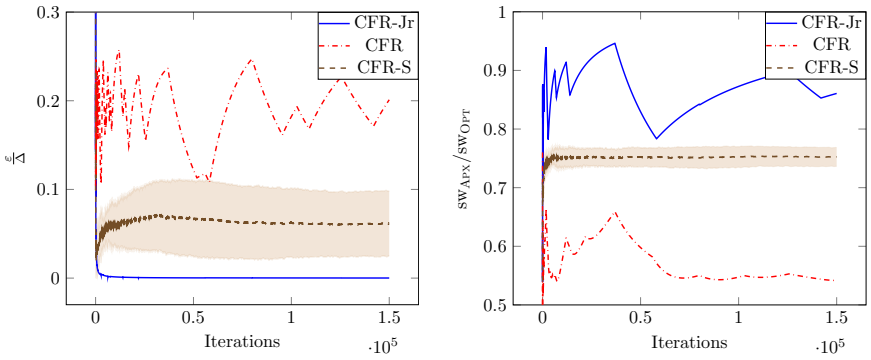

Figure 9.2: Left*: Convergence rate attained in the Asymmetric Extensive-Form Shapley game.* Right*: Social welfare attained with respect to the optimal one.*

Shapley game. We compare CFR with CFR-Jr and CFR-S, and the latter is averaged over multiple runs to account for its stochasticity (the plot shows the mean plus/minus the standard deviation).

In both examples, CFR has a non-converging, cyclic behavior, which also prevents it from reaching high values in social-welfare. On the other hand, CFR-Jr is able to consistently reach a social welfare higher than 80% of the maximum achievable. We also highlight that, in general, CFR-S performs worse than CFR-Jr, as it needs much more iterations to converge (this is partly due to the sampling procedure).

It is interesting to notice that, in Figure 9.2 (Right), CFR-Jr's attained social-welfare has a cyclic behavior, even if the ε-CCE computed has ε in the order of 10^{-4}. This behavior can be understood by considering the difference between converging in a punctual sense to an equilibrium and approaching a convex, closed set of equilibria [22] (which is what is achieved via no-regret learning techniques [48, 92]). In the former case, the distance between the convergence point and the sequence's points goes to zero. In the latter, the distance from the points in the sequence and the set goes to zero (where the distance is defined as the distance between the point and its projection onto the set). Therefore, we do not have any guarantee that, at different timesteps, CFR-Jr will provide approximations of the same equilibrium point.

9.2.3 CFR-S and CFR-Jr: Convergence Rate and Social-Welfare

In this section, we compare the performances of the algorithms guaranteed to compute an ε-CCE: CFR-S, CFR-Jr, and our column generation tech-

nique with MILP pricing oracle (denoted in the following by CG). To make the comparison fair, the column generation technique is suitably adapted to stop once a feasible solution is found (*i.e.*, once a CCE, even if not optimal, is found).

We evaluate the run time required by the algorithms to find an approximate CCE. The results are provided in Tables 9.4 and 9.5, which report the run time needed by CFR-S, CFR-Jr to achieve solutions with different levels of accuracy, and the time needed by CG for reaching an equilibrium. The accuracy α of the ε-CCEs reached is defined as $\alpha = \frac{\varepsilon}{\Delta}$, so that it is a measure of the percentage gap between the exact solution and the approximate one. In the tables, games are indicated with the shorthand notations previously defined, and their size (*i.e.*, number of infosets) and delta utility Δ are also reported. We generated 20 instances for each Rp-d family

CFR-Jr consistently outperforms CG and is significantly faster than CFR-S on the larger game instances. The data are consistent with the behavior displayed in Figure 9.1 (Left) and Figure 9.2 (Left).

The tables show, for the general-sum games, the social welfare approximation ratio between the social welfare of the solutions returned by the algorithms (sw_{APX}) and the optimal social welfare (sw_{OPT}). In practice, we employed the optimal payoff (the maximum sum of players' utilities), which is not guaranteed to be achievable by a CCE, as an upper bound on the optimal social welfare. Employing an upper bound was necessary as CG could not scale to the larger instances of our test bed.

The social welfare guaranteed by CFR-Jr is always nearly optimal, which makes it a good heuristic to compute optimal CCEs. Reaching a *socially good* equilibrium is crucial, in practice, to make correlation credible. Moreover, this is important as one of the reasons for adopting CCEs is that an optimal CCE provides a social-welfare which may be arbitrarily larger than that provided by CEs or NEs. Finding solutions that have social-welfare close to the optimal one implies that we can fully exploit CCE's advantages w.r.t. other alternative solution concepts.

9.2.4 Support Size of CFR-Jr's Joint Strategies

Theorem 8.2 provides a worst-case bound of $|Z|$ on the size of the realization equivalent normal-form strategies x_i^t generated at each iteration t. However, at each iteration t, CFR-Jr needs to compute x^t as the outer product of strategies x_i^t. The problem may be that, at each iteration t, CFR-Jr could generate a previously unseen x^t. This would result, in the worst case, in a final joint strategy $\bar{x}$ with support $T|Z|^{|\mathcal{P}|}$, which can easily become in-

Game	Tree size #infosets	Δ	CFR-S $\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.01$	$\alpha = 0.005$	$\alpha = 0.001$	$\alpha = 0.0005$	sw_{APX}/sw_{OPT}	CG
K3-6	72	6	0.22s	1.41s	24m20s	9h15m	$>$ 24h	$>$ 24h	-	3h47m
K3-7	84	6	0.62s	4.22s	1h3m	17h11m	$>$ 24h	$>$ 24h	-	14h37m
K3-10	120	6	1.89s	22.69s	11h19m	$>$ 24h	$>$ 24h	$>$ 24h	-	$>$ 24h
L3-4	1200	21	4.0s	10m33s	$>$ 24h	$>$ 24h	$>$ 24h	$>$ 24h	-	$>$ 24h
L3-6	2664	21	21.54s	2h5m	$>$ 24h	$>$ 24h	$>$ 24h	$>$ 24h	-	$>$ 24h
L3-8	4704	21	35.3s	13h55m	$>$ 24h	$>$ 24h	$>$ 24h	$>$ 24h	-	$>$ 24h
G2-4-A*	4856	10	1m11s	10m31s	27h3m	$>$ 24h	$>$ 24h	$>$ 24h	0.979	$>$ 24h
G2-4-DA*	4856	10	12.83s	2m1s	53m28s	3h28m	4h48m	4h17m	0.918	$>$ 24h
G2-4-DH*	4856	10	11.56s	1m19s	42m18s	2h7m	3h19m	3h28m	0.918	$>$ 24h
G2-4-AL*	4856	10	15.01s	2m3s	43m18s	1h33m	4h4m	4h20m	0.919	$>$ 24h
G3-4-A*	98508	10	6m19s	1h33m	$>$ 24h	$>$ 24h	$>$ 24h	$>$ 24h	0.995	$>$ 24h
G3-4-DA*	98508	10	9m17s	1h13m	17h12m	$>$ 24h	$>$ 24h	$>$ 24h	0.986	$>$ 24h
G3-4-DH*	98508	10	5m24s	47m33s	11h51m	19h40m	22h11m	$>$ 24h	0.886	$>$ 24h
G3-4-AL*	98508	10	2m23s	32m34s	10h25m	15h32m	14h36m	17h30m	0.692	$>$ 24h
R3-12*	3071	1	45.0s	1m44s	13m10s	35m38s	10h8m	3h8m	0.906	$>$ 24h
R3-15*	24542	1	10m5s	21m30s	2h5m	4h28m	3h25m	7h50m	0.924	$>$ 24h

Table 9.4: *Running times and social welfare obtained by the CFR-S algorithm (for various levels of accuracy), and the CG algorithm. General-sum instances are marked with *. Results of CFR-S are averaged over 50 runs.*

Game	Tree size #infosets	Δ	CFR-Jr $\alpha = 0.1$	$\alpha = 0.05$	$\alpha = 0.01$	$\alpha = 0.005$	$\alpha = 0.001$	$\alpha = 0.0005$	sw_{APX}/sw_{OPT}	CG
K3-6	72	6	1.03s	1.03s	4.55s	13.41s	1m7s	11m21s	-	3h47m
K3-7	84	6	2.35s	2.35s	7.92s	14.33s	10m49s	51m27s	-	14h37m
K3-10	120	6	7.21s	7.21s	17.2s	72.78s	31m41s	4h11m	-	$>$ 24h
L3-4	1200	21	1.72s	1m15s	1h1m	6h10m	$>$ 24h	$>$ 24h	-	$>$ 24h
L3-6	2664	21	8.2s	2m40s	2h35m	1h19m	$>$ 24h	$>$ 24h	-	$>$ 24h
L3-8	4704	21	7m44s	20m22s	17h32m	$>$ 24h	$>$ 24h	$>$ 24h	-	$>$ 24h
G2-4-A*	4856	10	5m28s	20m23s	4h3m	11h4m	$>$ 24h	$>$ 24h	0.994	$>$ 24h
G2-4-DA*	4856	10	1m3s	1m36s	14m31s	56m6s	$>$ 24h	$>$ 24h	0.976	$>$ 24h
G2-4-DH*	4856	10	1m10s	1m51s	16m27s	1h5m	$>$ 24h	$>$ 24h	0.976	$>$ 24h
G2-4-AL*	4856	10	1m10s	1m48s	15m2s	55m43s	$>$ 24h	$>$ 24h	0.976	$>$ 24h
G3-4-A*	98508	10	1h21s	1h3m	3h3m	4h13m	5h4m	$>$ 24h	0.999	$>$ 24h
G3-4-DA*	98508	10	9m25s	12m18s	1h1m	1h50m	$>$ 24h	$>$ 24h	1.000	$>$ 24h
G3-4-DH*	98508	10	13m59s	16m38s	2h21m	4h8m	8h50m	15h27m	1.000	$>$ 24h
G3-4-AL*	98508	10	13m55s	1h21m	1h38m	5m2s	$>$ 24h	$>$ 24h	0.730	$>$ 24h
R3-12*	3052	1	7.67s	16.94s	1m37s	3m19s	17m1s	24m6s	0.897	$>$ 24h
R3-15*	24588	1	3m10s	3m34s	9m1s	14m53s	1h19m	3h3m	0.931	$>$ 24h

Table 9.5: *Running times and social welfare obtained by the CFR-Jr algorithm (for various levels of accuracy), and the CG algorithm. General-sum instances are marked with *.*

Algorithm 9.1 `CFR-Jr-`k

function CFR-JR(Γ)
  $\bar{x} \leftarrow \mathbf{0}$, $t \leftarrow 0$
  while $t < T$ **do**
    for all $i \in \mathcal{P}$ **do**
      $\pi_i^t \leftarrow \text{CFR}(\Gamma, i)$
      if $t \bmod k = 0$ **then**
        $x_i^t \leftarrow$ NF-STRATEGY-RECONSTRUCTION(π_i^t)
      end if
    end for
    if $t \bmod k = 0$ **then**
      $\bar{x} \leftarrow \bar{x} + \bigotimes_{i \in \mathcal{P}} x_i^t$
    end if
    $t \leftarrow t + 1$
  end whilereturn $\bar{x}^T = \frac{\bar{x}}{\lfloor \frac{T}{k} \rfloor}$
end function

tractable in games with many players and large numbers of terminal nodes.

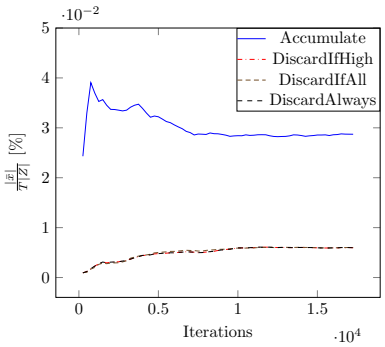

Figure 9.3: *Support size of $\bar{x}$ produced by CFR-Jr at different iterations.*

Fortunately, in typical scenarios, the support of the joint strategies will often be the same among different iterations, so that the final solution size will be significantly smaller than the worst case. To show this, we consider two-player instances of Goofspiel (G2-4) with the four tie-breaking rules. Figure 9.3 shows the ratio of $|\bar{x}|$ against $T|Z|$ (*i.e.*, the worst case $\sum_{t=1}^{T} |x_i^t|$). The ratio against the real worst case would be even smaller as the denominator would be $T|Z|^2$. For all four tie solvers, the CFR-Jr algorithm builds joint strategies $\bar{x}$ that are significantly more compact than what is required in the worst case, having a support size that is orders of magnitude smaller.

9.2.5 CFR-Jr with Different Reconstruction Rates

From a theoretical standpoint, CFR-Jr requires a joint distribution reconstruction step to be carried out at every iteration t, to ensure that the resulting normal-form joint strategy approaches the set of CCEs (see Theorem 8.3). We investigate whether it is possible to trade some accuracy to reduce the computational time of the algorithm, by performing the joint distribution reconstruction at a subset of the iterations. This could also allow the algorithm to store smaller normal-form strategies, by skipping the reconstruction during the first iterations. Indeed, during the first iterations, CFR (and, therefore, CFR-Jr) produces behavioral strategies that tend to be fairly uniformly distributed over all the possible actions, leading to ω assigning some probability to all terminals. This implies that the reconstructed normal-form strategies, in the first iterations, have considerably large supports.

These considerations suggest that a slight modification of the CFR-Jr algorithm, that we call CFR-Jr-k (see Algorithm 9.1), may perform better in some settings. The idea behind CFR-Jr-k is that the reconstruction procedure is carried out only every k iterations. We have evaluated CFR-Jr-k for different values of k. In all the tests we performed, the CFR-Jr-k algorithm always showed good convergence. In Figures 9.4–9.6, we report the experimental results related to instances of Kuhn3-6. The plots show both the convergence speed in terms of number of iterations and in terms of run time, as well as the size of the support of the average joint strategy that was stored by the algorithm (which is always monotonically increasing by construction). In Figures 9.7–9.9, we report the experimental results related to instances of Kuhn3-10.

In regard to the performances, larger reconstruction rates let the algorithm complete the same amount of iterations in a shorter time. On the other hand, smaller reconstruction rates can lead earlier to a good joint strategy, and hence to reach lower values of ε. There is a trade-off between iteration speed and reconstruction accuracy, which can be exploited to tackle different problems with the most suited level of precision. For what regards the size of the support of the joint average strategy, we can clearly see that lower reconstruction rates, running more times the reconstruction algorithm in the same amount of time, and being more susceptible to high-frequency variations in the behavioral strategies built by CFR, require up to ten time more space to store their joint strategies.

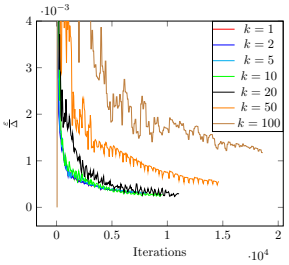

Figure 9.4: *K3-6. Convergence in number of iterations for CFR-Jr with different reconstruction rates*

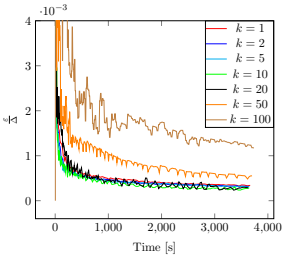

Figure 9.5: *K3-6. Convergence in run time (seconds) for CFR-Jr with different reconstruction rates*

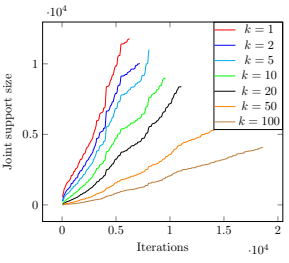

Figure 9.6: *K3-6. Size of the support of the joint strategy obtained from CFR-Jr with different reconstruction rates*

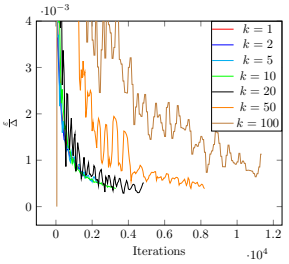

Figure 9.7: *K3-10. Convergence in number of iterations for CFR-Jr with different reconstruction rates*

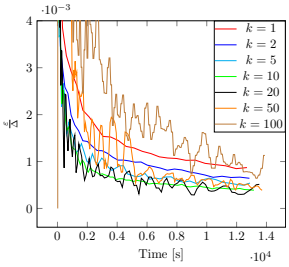

Figure 9.8: *K3-10. Convergence in run time (seconds) for CFR-Jr with different reconstruction rates*

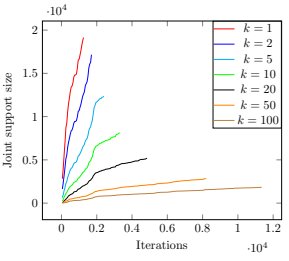

Figure 9.9: *K3-10. Size of the support of the joint strategy obtained from CFR-Jr with different reconstruction rates*

Part III

Private Bayesian Persuasion in Sequential Games

CHAPTER 10

Bayesian Persuasion with Sequential Receivers' Interactions

In this part of the thesis, we examine information-structure design problems as a means of forcing players' coordination towards a certain objective. More precisely, we start from the usual scenario where a mediator can communicate action recommendations to players before the beginning of an EFG. Suppose that parties (*i.e.*, the mediator and the players) are asymmetrically informed about the current state of the game. We pose the following question: *can an informed mediator exploit the information asymmetry to coordinate players' behavior toward a favorable outcome?*

This problem can be accurately modeled via the *Bayesian persuasion* framework. However, we are interested in studying cases where multiple players may interact in a sequential game. As we shall see, this setting requires the development of new techniques, and we will exploit analogies with the problems studied in Part II to examine Bayesian persuasion with sequential players' interactions through the lens of algorithms and complexity.

We investigate private persuasion games with multiple receivers interacting in a sequential game, and study the continuous optimization problem

of computing a private signaling scheme which maximizes the sender's expected utility. We focus on the framework with independent action types, similarly to what is previously done by Dughmi and Xu [60]. In doing so, we make a step in the direction of providing a general private multi-receiver Bayesian persuasion framework through the introduction of a model with the following key features:

1. it has an arbitrary number of receivers' actions and states of Nature;
2. it allows inter-agent externalities; [1]
3. it models sequential interactions among receivers.

As we highlighted in Section 2.6, these features mitigate some of the usual stringent hypothesis of previous works on the private multi-receiver setting. In particular, the last point constitutes the major difference from classical Bayesian persuasion models, which typically assume that the receivers take their actions simultaneously [58, 104] as, to the best of our knowledge, we address here the multi-receiver case with sequential interactions among receivers for the first time in the literature.

We show how to address sequential, multi-receiver settings algorithmically via the notion of *ex ante* persuasive signaling scheme, where the receivers commit to following the sender's recommendations having observed only the signaling scheme (and not the actual signal realization). This is motivated by the fact that the classical notion of persuasiveness (*ex interim* persuasiveness), which allows the receivers to deviate after observing the sender's signal, renders most of the associated design problems (with the exception of very narrow settings) computationally intractable even when the interaction is simultaneous [60], ultimately making its adoption impractical in real-world applications where the receivers act sequentially.

In parallel with our work, Xu [178] introduced a notion of *ex ante* persuasion similar to ours, but studies it in a significantly more restrictive setting: public signaling, simultaneous moves, binary actions, and no inter-agent externalities. Therefore, our work is more general.

Ex ante persuasive signaling schemes may be employed every time the environment allows for a credible receivers' commitment before the recommendations are revealed. As argued by Kamenica and Gentzkow [103], this is *not* unrealistic. On a general level, the receivers will uphold their *ex ante* commitment every time they reason with a long-term horizon where a reputation for credibility positively affects their utility [144]. In some

[1] When there are inter-agent externalities, the utility of a receiver is determined by the state of Nature, her action, and (crucially) the actions selected by all the other receivers.

cases, they could also be forced to stick to their *ex ante* commitment by contractual agreements or penalties.

The extension of the basic framework (see Section 2.4) to the case with multiple receivers is of major interest, see, *e.g.*, its applications to private-value auctions studied by Kaplan and Zamir [105]. Moreover, we highlight that many real-world problems involve *ex ante* commitments. This happens, for example, when the signaling schemes are implemented as software (*e.g.*, recommender systems) and receivers can decide whether to adopt it or not. This is the case in sequential auctions in online advertising, where a (trusted) third party service (*e.g.*, programmatic advertising platforms) could allow bidders for coordinated behaviors during the sequential auction, leading to better outcomes in terms of bidders' payoffs, and to more efficient allocations of the ads. Another example may come from traffic flow control systems. A company managing a road network has private information on the current traffic and road conditions, and it is interested in minimizing the congestion over its infrastructure. The drivers could get suggestions on the route to follow through an app with known criteria for computing the recommendations. The drivers could be forced to commit to following the road manager's suggestion by, *e.g.*, their insurance policy via a black-box/telematic insurance program.

The chapter is structured as follows: in Section 10.1 we introduce the notion of *ex ante* persuasive signaling scheme, and formalize its differences from *ex interim* persuasive schemes. Then, we show that *ex ante* persuasiveness can provide the sender with a utility that can be arbitrarily larger than that provided by *ex interim* persuasiveness (Section 10.1.3). Motivated by the hardness results for the *ex interim* setting with simultaneous moves provided by Dughmi and Xu [60], we study the problem of computing optimal *ex ante* signaling schemes. First, we prove a result of independent interest that plays a crucial role in the following proofs. More precisely, in Section 10.2.1, we show that, given a multi-player game and a behavioral strategy of a perfect-recall player, it is possible to find, in polynomial time, a realization-equivalent mixed strategy (defined on the normal form) with a polynomially-sized support (this result is complementary to Algorithm 8.5). In Section 10.2.2, we show that an optimal *ex ante* signaling scheme may be computed in polynomial time in settings with two receivers and independent action types, which makes *ex ante* persuasive signaling schemes a persuasion tool which is applicable in practice. Moreover, we show that this result cannot be extended to settings with more than two receivers, as the problem of computing an optimal *ex ante* signaling scheme becomes NP-hard (Section 10.3).

10.1 Notions of Persuasion

In our model, we assume to have a *sender* denoted by S and a set of *receivers* $\mathcal{R} = \{1, \dots, n\}$. Each receiver $i \in \mathcal{R}$ is faced with the problem of selecting actions from a set A_i with *a priori* uncertain payoffs. We adopt the perspective of the sender, whose goal is persuading the receivers to take actions which are favorable for her. The fundamental feature of our model is that receivers confront themselves in a *sequential* decision problem, which we describe as an extensive-form game with imperfect information and perfect recall in which $\mathcal{P} = \mathcal{R}$.

Payoffs are a function of the actions taken by the receivers and of an unknown *state of nature* θ, drawn from a set of potential realizations Θ. We follow the standard framework of Dughmi and Xu [60], where each action a has a set of possible types Θ_a and in which a state of nature θ is a vector specifying the realized type of each action of the receivers, *i.e.*, $\theta \in \Theta = \bigtimes_{i \in \mathcal{R}} \bigtimes_{a \in A_i} \Theta_a$.[2] Furthermore, as also done by Dughmi and Xu [60], we assume action types which are drawn independently from action-specific marginal probability distributions denoted by $\tilde{\pi}_a \in \text{int}(\Delta(\Theta_a))$, where $\tilde{\pi}_a(t)$ is the probability of a having type $t \in \Theta_a$.[3] These marginal probability distributions form a common prior over the states of nature which we assume to be known explicitly to both sender and receivers. This common knowledge can be equivalently represented by the distribution $\mu_0 \in \Delta^{|\Theta|}$, where $\mu_0(\theta) = \prod_{i \in \mathcal{R}} \prod_{a \in A_i} \tilde{\pi}_a(\theta_a)$. Notice that receivers' sequential interaction is assumed to be an extensive-form game of no chance. The model describes exogenous stochasticity via the prior over Θ.

In the following, we describe the two models of persuasion (*ex interim* and *ex ante*), and highlight their differences with some examples.

10.1.1 *Ex interim* Persuasiveness

Let $u_S : \Sigma \times \Theta \to \mathbb{R}$ and $u_i : \Sigma \times \Theta \to \mathbb{R}$ be the payoff functions of the sender and receiver $i \in \mathcal{R}$. We assume that the sender is allowed to tailor signals to individual receivers through private communications. Let Ξ_i be the set of signals available to receiver i, and let $\Xi = \bigtimes_{i \in \mathcal{R}} \Xi_i$. We assume that the sender has access to private information and her goal is designing a *signaling scheme* $\varphi : \Theta \to \Delta(\Xi)$ to persuade the receivers to select actions which are favorable for her. We denote by φ_θ the probability distribution over Ξ having observed θ. In the classical Bayesian persuasion framework

[2] Standard (*i.e.*, non Bayesian) EFGs can be represented by assigning to each Θ_a a singleton. Note that this model also encompasses Bayesian games *á la* Harsanyi [91].

[3] $\text{int}(X)$ is the interior of set X.

by Kamenica and Gentzkow [103], the receivers decide their behavior after having observed the sender's signal and after having updated their prior over Θ accordingly. The sender-receivers interaction goes as follows:

- The sender chooses φ and publicly discloses it.
- Nature draws a state $\theta \sim \mu_0$, observed by the sender.
- The sender draws a tuple $\xi \sim \varphi_\theta$ and privately sends signal ξ_i to each receiver $i \in \mathcal{R}$.
- Each receiver i updates her posterior distribution knowing φ and having observed ξ_i. Then, each of the receivers selects a plan $\sigma_i \in \Sigma_i$. Together, their joint choices form the tuple $\sigma = (\sigma_1, \ldots, \sigma_n)$.
- Sender and receivers get, respectively, payoffs $u_S(\theta, \sigma)$ and $u_i(\theta, \sigma)$, for all $i \in \mathcal{R}$.

Even in the multi-receiver setting, a result similar to the *revelation principle* (see, *e.g.*, Myerson [134]) holds [58]. Specifically, an *optimal signaling scheme* (*i.e.*, a signaling scheme maximizing the sender's expected utility) can always be obtained by restricting the set of signals Ξ to the set of plans Σ (see Proposition 1 by Kamenica and Gentzkow [103]). In the following, we assume $\Omega = \Xi$ (*i.e.*, the sender recommends a plan to be followed by each receiver). The receivers have an incentive to follow the sender's recommendation $\hat{\sigma}_i$ if the recommended plan is preferred to any other action, conditional on the knowledge of $\hat{\sigma}_i$. We call this condition *ex interim persuasiveness*, which is precisely the kind of constraint characterizing a *Bayes Correlated Equilibrium* (BCE) [18, 19]. We remark that, according to the definition of BCE, the signaling scheme with sequential receivers' interactions must necessarily be defined on plans and cannot be compactly represented by using sequences or actions.

Definition 10.1 (*Ex interim* persuasiveness)**.** *A signaling scheme $\varphi : \Theta \to \Delta(\Sigma)$ is* ex interim persuasive *if the following holds for all $i \in \mathcal{R}$ and $\sigma_i, \sigma_i' \in \Sigma_i$:*

$$\sum_{\theta \in \Theta, \sigma_{-i} \in \Sigma_{-i}} \mu_0(\theta)\varphi_\theta(\sigma_i, \sigma_{-i})\Big(u_i(\theta, (\sigma_i, \sigma_{-i})) \; - \; u_i(\theta, (\sigma_i', \sigma_{-i}))\Big) \;\geqslant\; 0.$$

Definition 10.2. *A signaling scheme $\varphi : \Theta \to \Delta(\Sigma)$ is a BCE if it is* ex interim *persuasive.*

10.1.2 *Ex ante* Persuasiveness

We introduce the setting in which the receivers have to decide whether to follow the sender's recommendations before actually observing them, basing their decision only on the knowledge of μ_0 and φ.[4] The interaction between sender and receivers goes as follows.

- The sender computes φ, and publicly discloses it.
- The receivers decide whether to adhere to the recommendations drawn according to φ or not.
- Nature draws a state $\theta \sim \mu_0$, observed by the sender.
- If $i \in \mathcal{R}$ decided to opt-in to the signaling scheme:
  - the sender draws $\hat{\sigma}_i \sim \varphi_\theta$ and privately communicates it to receiver i;
  - receiver i acts according to the recommended $\hat{\sigma}_i$.
- Sender and receivers get, respectively, payoffs $u_S(\theta, \sigma)$ and $u_i(\theta, \sigma)$, $\forall i \in \mathcal{R}$, where $\sigma_i = \hat{\sigma}_i$ if i adhered to the signaling scheme.

In this setting, the receivers adhere to the signaling scheme (*i.e.*, $\sigma_i = \hat{\sigma}_i$) if φ is *ex ante* persuasive:

Definition 10.3 (*Ex ante* persuasiveness)**.** *The signaling scheme $\varphi : \Theta \to \Delta(\Sigma)$ is* ex ante *persuasive if, for all $i \in \mathcal{R}$ and $\sigma_i \in \Sigma_i$, the following holds:*

$$\sum_{\substack{\theta\in\Theta,\sigma_i'\in\Sigma_i \\ \sigma_{-i}\in\Sigma_{-i}}} \mu_0(\theta)\varphi_\theta(\sigma_i', \sigma_{-i})\Big(u_i(\theta, (\sigma_i', \sigma_{-i})) \;-\; u_i(\theta, (\sigma_i, \sigma_{-i}))\Big) \quad \geqslant \quad 0.$$

Such constraints define *Bayes Coarse Correlated Equilibria* (BCCE), *i.e.*, the generalization of coarse correlated equilibria to incomplete-information games, see Forges [74], Cai and Papadimitriou [37], Hartline et al. [94], and Caragiannis et al. [38].[5]

Definition 10.4. *A signaling scheme $\varphi : \Theta \to \Delta(\Sigma)$ is a BCCE if it is* ex ante *persuasive.*

[4] As discussed in the beginning of the chapter, the receivers' commitment to follow a certain signaling scheme is not an unrealistic assumption for the same reason why it is realistic to assume the sender's commitment power.

[5] The set of (non Bayesian) coarse correlated equilibria is characterized by the constraints of Definition 10.3, with $|\Theta_a| = 1\ \forall a \in A$.

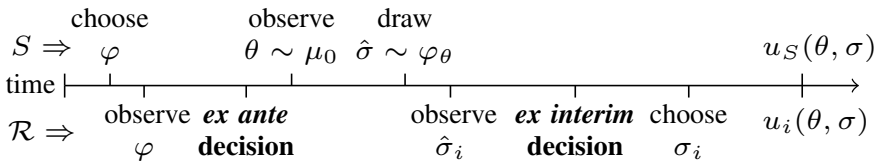

Figure 10.1: *Interaction between sender and receivers in the* ex ante *and* ex interim *settings.*

	In	Out	P
E	$(-1, 1)$	$(1, 0)$	$(0, 1/2)$
H	$(-1, -1)$	$(1, 0)$	$(0, 0)$

Figure 10.2: *A game where* ex ante *persuasion guarantees the sender a higher expected utility with respect to* ex interim *persuasiveness.*

10.1.3 Comparison between Notions of Persuasiveness

Figure 10.1 summarizes the interaction flow between sender and receivers in the two aforementioned settings. The key difference is the time at which the receivers decide whether to adhere to the signaling scheme or not.

We also propose the following illustrative example (in the basic single-receiver setting) to further illustrate the main differences between the two notions of persuasiveness.

Example 9. *The incumbent of an industry wants to persuade a potential new entrant to the market. The market can be either easy (E), with probability 0.3, or hard (H), with the remaining probability. The incumbent knows the state of the market. The entrant has three possible actions available: entering the market (In), staying out of the market (Out), or proposing a partnership to the incumbent (P). Figure 10.2 depicts the utility matrix for the game (the first values are the incunbent's payoffs).*

The incumbent wants the entrant to stay out of the market, values its entrance negatively, and is indifferent towards a partnership. The entrant values entering the new market positively only when it has favorable conditions. A partnership in a hard market gives the entrant 0 (rather than a negative score) as no fixed costs have to be sustained. In this setting, forcing the entrant (contractually) to commit to following the incumbent's recommendations ex ante *is strictly better (in terms of expected utility) for the incumbent.*

An optimal ex ante *signaling scheme (e.g., $\varphi_E(In) = \varphi_E(Out) = \frac{1}{2}$,*

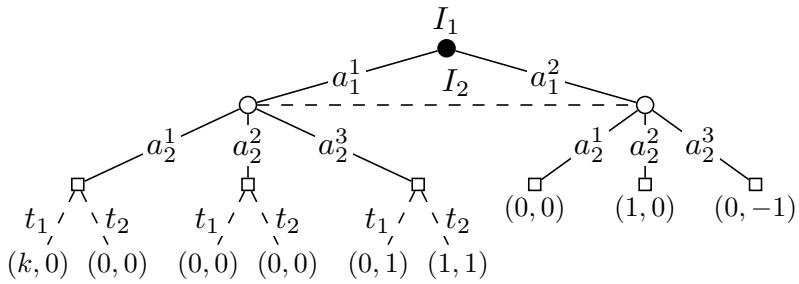

Figure 10.3: *A game with two receivers in which action a_1^1 has two possible types t_1 and t_2. Terminal nodes report receivers' utilities.*

$\varphi_H(Out) = 1$*) guarantees the sender an expected utility of* 0.7*. An optimal* ex interim *signaling scheme (e.g.,* $\varphi_E(P) = 1$*,* $\varphi_H(Out) = \frac{11}{14}$*,* $\varphi_H(P) = \frac{3}{14}$*,) guarantees a sender's expected utility of 0.55. Therefore,* ex ante *persuasion provides a 27% increase in utility for the incumbent w.r.t. ex interim persuasion.*

We remark that the set of *ex ante* persuasive signaling schemes strictly includes the set of *ex interim* signaling schemes. In particular, an optimal *ex ante* persuasive signaling scheme may lead to an expected utility for the sender that is arbitrarily larger than the one she would obtain with an optimal *ex interim* scheme. This is shown by means of the following example.

Example 10. *Consider the game in Figure 10.3, with two receivers with one information set each (I_1 for receiver 1 and I_2 for receiver 2), and parametric in $k \gg 1$. Action $a_1^1 \in A_1$ is such that $\Theta_{a_1^1} = \{t_1, t_2\}$ and $\tilde{\pi}_{a_1^1}(t_1) = \tilde{\pi}_{a_1^1}(t_2) = 1/2$. The figure only reports the receivers' utilities, as we assume $u_S(\theta, \sigma) = u_1(\theta, \sigma) + u_2(\theta, \sigma)$, $\forall(\theta, \sigma)$. The signaling scheme with $\varphi'_{t_1}(a_1^1, a_2^1) = 1/2$, $\varphi'_{t_1}(a_1^2, a_2^2) = 1/2$, and $\varphi'_{t_2}(a_1^1, a_2^3) = 1$ is* ex ante persuasive*, but it is not* ex interim *persuasive. The optimal* ex interim *persuasive signaling scheme is such that $\varphi''_{t_1}(a_1^2, a_2^2) = 1$, and $\varphi''_{t_2}(a_1^1, a_2^3) = 1$. Signaling scheme φ' guarantees the sender an expected utility of $(k+5)/4$, while signaling scheme φ'' guarantees $3/2$. Therefore, for increasing values of k, an optimal* ex ante *signaling scheme provides an arbitrarily larger utility than what can be obtained by* ex interim *persuasion.*

10.2 Positive Result

In the independent-actions setting, Dughmi and Xu [60] show that computing an optimal *ex interim* signaling scheme is #P-hard even with a single

receiver. Motivated by this negative result, we study the problem of computing an optimal (for the sender) *ex ante* persuasive signaling scheme. We denote this problem by OPT-EA. It amounts to computing a Coarse Correlated Equilibrium (CCE) for the game of complete information obtained by treating Nature as a player with a *trivial* (*i.e.*, constant everywhere) payoff function and subject to having marginal strategies constrained to be μ_0.[6]

In contrast with the known hardness results for the *ex interim* setting, and following a similar reasoning to that of Theorem 7.3, we show that OPT-EA with $|\mathcal{R}| = 2$ can be solved in polynomial time (see Theorem 10.3 below). To prove our main theorem, we first show how to build, in polynomial time, a *small* (*i.e.*, with a support of size upper bounded by a polynomial) mixed strategy which is realization-equivalent to a given behavioral strategy (this is an alternative approach to Algorithm 8.5, and presents the problem from a complementary perspective).

10.2.1 Small–Supported Mixed Strategy

Given a behavioral strategy profile π_i^* for a generic perfect-recall player i, we show (see Theorem 10.2 below) that it is always possible to compute in polynomial time some $x_i^* \in \mathcal{X}_i$ such that (i) it is realization-equivalent to π_i^* and (ii) it has a support of polynomial size.

For each $\sigma_i \in \Sigma_i$, let

$$\zeta(\sigma_i) \coloneqq \{q \in Q_i | \exists I \in \mathcal{I}_i, \sigma_i(I) = q\}$$

(*i.e.*, the set of sequences selected with probability 1 in a realization plan equivalent to σ_i). Analogously, $\forall \sigma = (\sigma_1, \sigma_2) \in \Sigma$ we denote by $\zeta(\sigma)$ the set of tuples (q_1, q_2) such that $q_1 \in \zeta(\sigma_1)$ and $q_2 \in \zeta(\sigma_2)$. In the remainder of the section, we drop the dependency on i when it is not strictly necessary. We denote by M an $|Q_i| \times |\Sigma_i|$ matrix where $M(q, \sigma) = 1$ iff $q \in \zeta(\sigma)$ and $M(q, \sigma) = 0$ otherwise. We denote by M_q the row of M specifying the plans containing q in their support. Let r^* be the $|Q_i|$-dimensional vector representing the realization plan of player i which is realization-equivalent to π^*. In order to compute x^*, it is enough to find an optimal solution to the LP

$$\max_{x \in \mathbb{R}_{\geqslant 0}^{|\Sigma_i|}} \quad \{\mathbb{1}^\top x \quad \text{s.t.} \quad Mx \leqslant r^*\},$$

which we denote by (A), which has a polynomial number of constraints and an exponential number of variables.

[6] Notice that finding an optimal CCE with two players and Nature is hard in the worst-case, while our problem is a variation.

By relying on the assumption of perfect recall and proceeding by contradiction, we establish the following lemma:

Lemma 10.1. *An optimal solution x^* to (A) satisfies $Mx^* = r^*$.*

Proof. Consider a behavioral strategy π^* whose realization-equivalent realization plan is denoted by r^*. Since player i has perfect recall, there always exists at least a mixed strategy $\hat{x} \in \mathcal{X}_i$ realization equivalent to π^* [126, Th. 6.11]. Therefore, the optimal value of (A) is 1 (since $\mathbb{1}^\top \hat{x} = 1$). Given $\hat{x} \in \mathcal{X}_i$, a distribution assigning to each sequence $q \in Q_i$ value $\sum_{\sigma \in \Sigma_i : q \in \zeta(\sigma)} \hat{x}_\sigma$ is a valid realization plan. Therefore, if $x \in \Delta(\Sigma_i)$ then Mx is a well defined realization plan for i. Now, by contradiction, assume that x^* is an optimal solution to (A) and that there exists $q' \in Q_i$ such that $M_{q'} x^* < r^*(q')$. Optimality implies $\mathbb{1}^\top x^* = 1$ and, therefore, $x^* \in \Delta(\Sigma_i)$. Let $Mx^* = r$. We have $r(q') < r^*(q')$. Since the sequence-form constraints hold, there must exist some $q'' \in Q_i$ such that $r(q'') > r^*(q'')$. This leads to a contradiction since x^* would not be a feasible solution. □

We now characterize an optimal solution to (A) by two properties which are proven by relying on Lemma 10.1 and on the fact that, as LP (A) contains a polynomial number of constraints, it admits an optimal basic solution with only a polynomial number of nonzero variables The support of strategy $x \in X_i$ is denoted by $\text{supp}(x)$. Then, we have:

Theorem 10.1. *These two properties hold:*

- *(i) An optimal solution x^* to (A) is a normal-form strategy ($x^* \in \mathcal{X}_i$) realization equivalent to r^*,*
- *(ii) there exists an optimal solution x^* with $\text{supp}(x^*)$ of polynomial size.*

Proof. Since $Mx^* = r^*$ (Lemma 10.1), we have $M_{q_\varnothing} x^* = r^*(q_\varnothing)$, that is $\mathbb{1}^\top x^* = 1$. Therefore, $x^* \in \mathcal{X}_i$. Realization equivalence follows from Lemma 10.1 and from the fact that Mx^* defines a valid realization plan. Moreover, LP (A) admits a basic optimal solution with at most $|Q_i|$ variables with strictly positive values [153]. Then, there exists an optimal x^* with support of polynomial size. □

We introduce some auxiliary notation to simplify the presentation of the following results. Specifically, for each $q \in Q_i$, we denote by $I_\downarrow(q) \subseteq \mathcal{I}_i$ the set of infosets reachable by i after selecting q without making other

intermediate moves, whereas $I_{\uparrow}(q) \in \mathcal{I}_i$ denotes the unique infoset where the last action of q was taken. [7]

Let $\mathcal{D}$ be the dual of problem (A). By showing that an optimal plan corresponding to a violated dual constraint can be found in polynomial time by backward induction, we can establish the following:

Lemma 10.2. *$\mathcal{D}$ admits a polynomial-time separation oracle.*

Proof. Let $\alpha \in \mathbb{R}^{|Q_i|}$ be the vector of dual variables (corresponding to constraints $Mx \leqslant r^*$). $\mathcal{D}$ is an LP with a polynomial number of variables ($|Q_i|$) and an exponential number of constraints ($|\Sigma_i|$).

We show that, given $\bar{\alpha} \in \mathbb{R}^{|Q_i|}$, the problem of finding a hyperplane separating $\bar{\alpha}$ from the set of feasible solutions to $\mathcal{D}$ or proving that no such hyperplane exists can be solved in polynomial time. The problem amounts to determining whether there exists a violated (dual) constraint $M_\sigma^\top \bar{\alpha} \geqslant 1$ for some $\sigma \in \Sigma_i$. Given $\bar{\alpha}$, the *separation problem* of finding one such constraint of maximum violation reads: $\min_{\sigma \in \Sigma_i}\{M_\sigma^\top \bar{\alpha}\}$. A plan σ yielding a violated constraint exists iff the separation problem admits an optimal solution of value < 1. One such plan (if any) can be found in polynomial time by reasoning in a backward induction fashion, starting from information sets of i originating only terminal sequences, and proceeding backwards. At each $I \in \mathcal{I}_i$, player i selects $\hat{q} \in Q(I)$ such that

$$\hat{q} \in \arg\min_{q \in Q(I)}\{\sum_{I' \in I_{\downarrow}(q)} w_{I'} + \bar{\alpha}(q)\},$$

and subsequently sets

$$w_I := \sum_{I' \in I_{\downarrow}(\hat{q})} w_{I'} + \bar{\alpha}(\hat{q}).$$

This procedure requires a computing time linear in $|\mathcal{I}_i|$. Then, a maximally violated inequality can be found by building a plan according to the sequences determined at the previous step. □

Next, by relying on Lemma 10.2 and on the *ellipsoid method* for solving LPs we prove a result which is the basis for our main theorem, Theorem 10.3 (whose statement and proof are given in full in the next subsection):

[7] When the context requires disambiguation between different games, we write $I_{\downarrow}^{\Gamma}(q)$ to denote the result for EFG Γ.

Theorem 10.2. *Given an EFG, a perfect-recall player i, and a behavioral strategy profile π^* for i (with the realization-equivalent realization plan r^*), a solution to LP (A) can be found in polynomial time.*

Proof. Due to the equivalence between optimization and separation [86], since the separation problem for $\mathcal{D}$ can be solved in polynomial time one can solve $\mathcal{D}$ in polynomial time via the ellipsoid method [107]. As the ellipsoid method solves a primal-dual system encompassing both $\mathcal{D}$ and (A), it also produces a solution to (A). □

Finally, we show that we can efficiently compute a solution with support size of at most $|Q_i|$ by applying the ellipsoid method for at most a polynomial number of iterations:

Corollary 4. *A* basic feasible solution *to (A) can be computed in polynomial time.*

Proof. First, the ellipsoid method returns an optimal solution x^* with polynomial support size (say $|\text{supp}(x^*)| = m$). This is because the number of iterations is polynomial and, by adding a new inequality to the dual per iteration, we also add a new variable to the primal per each iteration. If x^* is a basic feasible solution, its support is necessarily the smallest possible and we can halt the procedure. If not, x^* belongs to the relative interior of a face of the polytope defined by (A).

Suppose to have $\{e_j x\}_{j=1}^m = \text{supp}(x)$, where e_j is the canonical vector with a single 1 in position j. To obtain an optimal basic-feasible solution, we proceed as follows. Let $j := 1$. First, we restrict (A) to the optimal face by adding the constraint $\mathbb{1}^\top x = \mathbb{1}^\top x^*$. Then, we reoptimize the LP maximizing the objective function $e_j x$. If we do not obtain a basic-feasible solution, we iterate the procedure adding the constraint $e_j x = e_j x^*$ and letting $j := j + 1$. The dimension of the LP is reduced by 1 at each iteration (and the number of steps is polynomial, as we have one for each of the polynomially-many variables in $\text{supp}(x^*)$). This leads to optimizing over faces of (A) of increasingly smaller dimension. When the dimension reaches 1, the corresponding solution is necessarily a basic one. □

10.2.2 Optimal *Ex Ante* Persuasive Schemes

Computing an *ex ante* persuasive signaling scheme is equivalent to computing a CCE for an EFG of complete information where Nature is treated as a player with constant utility and marginal strategies constrained to be

equal to μ_0. We focus on the setting where $|\mathcal{R}| = 2$ and show that OPT-EA can be solved in polynomial time. We reason over an auxiliary game where each action of the receivers is followed by one of Nature's nodes, determining its type. Marginal probabilities $\tilde{\pi}$ determining action types are treated as behavioral strategies of the Nature player, which we denote by N. Formally:

Definition 10.5. *Given an EFG Γ describing the interaction between receivers and a set of marginal distributions $\{\tilde{\pi}_a \in \text{int}(\Delta(\Theta_a))\}_{a \in A}$, the auxiliary game $\hat{\Gamma}$ is an EFG such that:*

- *It has a set of players $\mathcal{R} \cup \{\mathsf{N}\}$, with N denoting the chance player.*
- *For each receiver $i \in \mathcal{R}$, her utility function is the same as in Γ,* i.e.*, $\forall(\theta, \sigma) \in \Theta \times \Sigma$, $u_i(\theta, \sigma) = \hat{u}_i(\theta, \sigma)$. Nature has $\hat{u}_{\mathsf{N}}(\cdot) = k \in \mathbb{R}$ constant everywhere.*
- *The receivers have the same information structures as in Γ,* i.e.*, $\forall i \in \mathcal{R}$, $\mathcal{I}_i = \hat{\mathcal{I}}_i$, and $\forall q \in Q_i$, $I^{\Gamma}_{\downarrow}(q) = I^{\hat{\Gamma}}_{\downarrow}(q)$.*
- *$\forall i \in \mathcal{R}$, each $a \in A_i$ is immediately followed by a singleton infoset $I \in \mathcal{I}_{\mathsf{N}}$ such that $A(I) = \Theta_a$.*
- *$\forall I \in \mathcal{I}_{\mathsf{N}}$, with I following $a \in A$, N selects actions (types) at I according to the marginal distribution $\tilde{\pi}_a$.*

The first step is devising an LP to compute a BCCE with a polynomial number of constraints and an exponential number of variables. We do so by computing an optimal CCE over $\hat{\Gamma}$ via a slight variation of the LP described in Section 7.2.3.

First, notice that θ is a plan of player N in $\hat{\Gamma}$. A distribution in $\Delta(\Theta)$ is a mixed strategy of N. Denote by μ^* the mixed strategy realization equivalent to $\tilde{\pi}$ computed (in poly-time) as in the proof of Theorem 10.2. Let $\Theta^* := \text{supp}(\mu^*)$. Due to Corollary 4, the set Θ^* has polynomial size. Then, we write the problem as a function of $\gamma \in \Delta(\Sigma \times \Theta^*)$ (*i.e.*, we look for a correlated distribution in $\hat{\Gamma}$, encompassing the Nature player).

Let v_i be the $|\mathcal{I}_i|$-dimensional vector of variables of the dual of the best-response problem for receiver i in sequence form. Moreover, we employ sparse $(|\mathcal{R}| + 1)$-dimensional matrices describing the utility function of sender and receivers for the profiles (θ, q_1, q_2) leading to terminal nodes of $\hat{\Gamma}$. We denote them by $U_i \in \mathbb{R}^{|\Theta^*| \times |Q_1| \times |Q_2|}$, with $i \in \mathcal{R} \cup \{S\}$.[8] In the following, we let $q = (q_1, q_2)$.

[8] U_i employs both the sequence form (for receivers), and plans of N. However, polynomiality of Θ^* implies polynomiality of U_i.

The problem of computing a CCE over $\hat{\Gamma}$ reads:

$$\max_{\substack{\gamma \geqslant 0, \\ v_1, v_2}} \sum_{\substack{\theta \in \Theta^* \\ \sigma \in \Sigma}} \gamma(\theta, \sigma) \sum_{q \in \zeta(\sigma)} U_S(\theta, q) \tag{10.1a}$$

$$\text{s.t.} \sum_{\substack{\theta \in \Theta^* \\ \sigma \in \Sigma}} \gamma(\theta, \sigma) \sum_{q \in \zeta(\sigma)} U_i(\theta, q) \geqslant \sum_{\substack{I' \in \mathcal{I}_i: \\ I' \in I_\downarrow(q_\varnothing)}} v_i(I') \qquad \forall i \in \mathcal{R} \tag{10.1b}$$

$$v_1(I_\uparrow(q_1)) - \sum_{I' \in I_\downarrow(q_1)} v_1(I') + \\ - \sum_{\substack{\theta \in \Theta^* \\ \sigma \in \Sigma}} \gamma(\theta, \sigma) \sum_{q_2 \in \zeta(\sigma_2)} U_1(\theta, q_1, q_2) \geqslant 0 \qquad \forall q_1 \in Q_1 \tag{10.1c}$$

$$v_2(I_\uparrow(q_2)) - \sum_{I' \in I_\downarrow(q_2)} v_2(I') + \\ - \sum_{\substack{\theta \in \Theta^* \\ \sigma \in \Sigma}} \gamma(\theta, \sigma) \sum_{q_1 \in \zeta(\sigma_1)} U_2(\theta, q_1, q_2) \geqslant 0 \qquad \forall q_2 \in Q_2 \tag{10.1d}$$

$$\sum_{\sigma \in \Sigma} \gamma(\theta, \sigma) = \mu^*(\theta) \qquad \forall \theta \in \Theta^*. \tag{10.1e}$$

We make the following observations on the above LP, which we denote by (B):

- The left term of constr. (10.1b) is the expected utility of i at the equilibrium. Incentive constraints (10.1c) and (10.1d) are compactly encoded by exploiting the sequence form. Intuitively, we decompose the best-response problem locally at each infoset. The constraints impose that the utility at the equilibrium be no smaller than the value achieved when playing the plan obtained by letting the receiver best respond in each infoset.
- Constraint (10.1e) forces Nature's marginal distribution to be equal to the prior μ^*.
- Once a solution γ^* to (B) has been computed, an optimal solution to OPT-EA is the signaling scheme which, having observed θ, recommends σ with probability $\gamma^*(\theta, \sigma)/\mu^*(\theta)$.

The following key positive result holds:

Theorem 10.3. *OPT-EA can be solved in polynomial time when* $|\mathcal{R}| \leqslant 2$.

Proof. Let $\mathcal{D}_B$ be the dual of (B). Let α_1, α_2 be the dual variables of constraints (10.1b), $\beta_1 \in \mathbb{R}^{|Q_1|}$ and $\beta_2 \in \mathbb{R}^{|Q_2|}$ the dual variables of (10.1c) and (10.1d), and $\delta \in \mathbb{R}^{|\Theta^*|}$ the dual variables of (10.1e). We show that, given $(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\beta}_1, \bar{\beta}_2, \bar{\delta})$, the problem of finding either a hyperplane separating the solution from the feasible set of $\mathcal{D}_B$ or proving that no such hyperplane exists can be solved in polynomial time. Along the lines of Theorem 10.2, this implies that (B) is solvable in polynomial time by the ellipsoid method. As the number of dual constraints corresponding to variables v_i is linear, all these constraints can be checked efficiently for violation. Besides those, the dual problem $\mathcal{D}_B$ features the following constraint for each $(\theta, \sigma) \in \Theta^* \times \Sigma$:

$$\sum_{i \in \mathcal{R}} \sum_{q \in \zeta(\sigma)} U_i(\theta, q)\bar{\alpha}_i + \frac{\bar{\delta}(\theta)}{\mu^*(\theta)} - \sum_{q \in \zeta(\sigma)} U_S(\theta, q) + \\ - \sum_{q \in Q_1 \times \zeta(\sigma_2)} U_1(\theta, q)\bar{\beta}_1(q_1) - \sum_{q \in \zeta(\sigma_1) \times Q_2} U_2(\theta, q)\bar{\beta}_2(q_2) \geqslant 0.$$

Given $(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\beta}_1, \bar{\beta}_2, \bar{\delta})$, the *separation problem* of finding a maximally violated inequality of $\mathcal{D}_B$ reads:

$$\min_{\theta, \sigma} \left\{ \sum_{q \in \zeta(\sigma)} \left[\sum_{i \in \mathcal{R}} U_i(\theta, q) - U_S(\theta, q) \right] + \frac{\bar{\delta}(\theta)}{\mu^*(\theta)} \right. \\ \left. - \sum_{q \in Q_1 \times \zeta(\sigma_2)} U_1(\theta, q)\bar{\beta}_1(q_1) - \sum_{q \in \zeta(\sigma_1) \times Q_2} U_2(\theta, q)\bar{\beta}_2(q_2) \right\}.$$

A pair (θ, σ) yielding a violated inequality exists iff the separation problem admits an optimal solution of value < 0. If such a (θ, σ) exists, it can be determined in polynomial time by enumerating all the (polynomially many) $(\theta, z) \in \Theta^* \times \hat{Z}$, where $\hat{Z}$ is the outcome set of $\hat{\Gamma}$. For each pair (θ, z), we look for a $\sigma \in \Sigma$ which, together with some actions of N, minimizes the objective function of the separation problem and could lead to z. The procedure halts as soon as a plan σ such that (θ, σ) yielding a violated inequality is found; if it terminates without finding any, $\mathcal{D}_B$ has been solved. First, by fixing a pair (θ, z) the first two terms of the objective function are completely determined. The remaining terms can be minimized independently for each receiver. Let us consider the problem of

Algorithm 10.1 `Separation plan search for` (θ, z)

function F(I,Q^*) ▷ $I \in \mathcal{I}_2$ is the current infoset
  $\hat{Q} \leftarrow \varnothing$, $w(q_2) \leftarrow -\infty \quad \forall q_2 \in Q_2$
  if $I \in \mathcal{I}_2^z$ **then**
    $\hat{Q} \leftarrow \{q_2 \in Q_2 | q_2 \in Q(I) \text{ and } q_2 \in Q_2^z\}$
  else
    $\hat{Q} \leftarrow Q(I)$
  end if
  for $q_2 \in \hat{Q}$ **do**
    $w(q_2) \leftarrow \sum_{q_1 \in Q_1} U_1(\theta, q_1, q_2)\bar{\beta}_1(q_1) + \sum_{I' \in I_{\downarrow}(q_2)} \mathrm{F}(I', Q^*)$
  end for
  $q_2^* = \arg\max_{q_2 \in Q_2} w(q_2)$
  $Q^* \leftarrow Q^* \cup \{q_2^*\}$
  return $w(q_2^*)$
end function

finding $\sigma_2 \in \Sigma_2$ (the other one is solved analogously). It reads

$$\max_{\sigma_2 \in \Sigma_2} \left\{ \sum_{q_1 \in Q_1} \sum_{q_2 \in \zeta(\sigma_2)} U_1(\theta, q_1, q_2)\bar{\beta}_1(q_1) \right\},$$

subject to the constraint that σ_2 be an admissible plan for the given z (*i.e.*, given the solution plan, it has to be possible to reach z together with some actions of the other players).

This problem can be solved in poly-time as shown in Algorithm 10.1 (analogous to Algorithm 7.1), where $\mathcal{I}_i^z$ and Q_i^z are, respectively, the set of infosets and sequences of i encountered on the path from the root to z.

Once Q^* has been determined by visiting each $I \in \mathcal{I}_2$, the corresponding optimal σ_2 can be built directly. As in Corollary 4, an optimal solution to (B) has polynomial support size. Then, it is used to determine an optimal solution to OPT-EA in poly-time. □

10.3 Negative Results

We conclude by showing that the previous approach cannot be extended to settings where $|\mathcal{R}| > 2$ and that, in particular, the border between easy and hard cases coincides with $|\mathcal{R}| = 2$. Indeed, the fact that computing an optimal CCE for a three-player EFG is NP-hard (see Corollary 2) directly implies the following:

Theorem 10.4. *OPT-EA is NP-hard when* $|\mathcal{R}| > 2$.

Proof. Let $|\mathcal{R}| = 3$ and, $\forall a \in A$, $|\Theta_a| = 1$. Then, the problem amounts to computing an optimal CCE for a three player EFG, which is NP-hard (Corollary 2). □

For completeness, we present an alternative proof of Theorem 10.4 based on solving the dual $\mathcal{D}_B$ with the ellipsoid algorithm. As a consequence of the equivalence between optimization and separation [86], the following holds:

Theorem 10.5. *Computing an optimal solution to $\mathcal{D}_B$ is NP-hard when $|\mathcal{R}| > 2$.*

Proof. Consider the case in which $\mathcal{R} = 3$ and $\mathcal{D}_B$ is adapted accordingly. Let $q = (q_1, q_2, q_3)$. Given the dual variables $(\bar{\alpha}_1, \bar{\alpha}_2, \bar{\alpha}_3, \bar{\beta}_1, \bar{\beta}_2, \bar{\beta}_3, \bar{\delta})$, the separation problem reads:

$$\min_{\theta,\sigma} \left\{ \sum_{q\in\zeta(\sigma)} \left[\sum_{i\in\mathcal{R}} U_i(\theta,q) - U_S(\theta,q) \right] + \frac{\bar{\delta}(\theta)}{\mu^*(\theta)} - \sum_{q\in Q_1\times\zeta(\sigma_2)\times\zeta(\sigma_3)} U_1(\theta,q)\bar{\beta}_1(q_1) \right.$$
$$\left. - \sum_{q\in\zeta(\sigma_1)\times Q_2\times\zeta(\sigma_3)} U_2(\theta,q)\bar{\beta}_2(q_2) - \sum_{q\in\zeta(\sigma_1)\times\zeta(\sigma_2)\times Q_3} U_3(\theta,q)\bar{\beta}_3(q_3) \right\}.$$

Consider a setting with the following features: $\forall\theta \in \Theta^*$, $\bar{\delta}(\theta) = 0$ (a valid assumption since $\delta \in \mathbb{R}^{|\Theta^*|}$); $\forall(\theta,q) \in \Theta^* \times (\bigtimes_{i\in\mathcal{R}} Q_i)$, $U_S(\theta,q) = U_1(\theta,q)$; $\forall(\theta,q) \in \Theta^* \times (\times_{i\in\mathcal{R}} Q_i)$, $U_2(\theta,q) = U_3(\theta,q) = 0$. Then, finding a maximally violated inequality corresponds to solving:

$$\arg\max_{\theta,\sigma_2,\sigma_3} \left\{ \sum_{q\in Q_1\times\zeta(\sigma_2)\times\zeta(\sigma_3)} U_1(\theta,q)\bar{\beta}_1(q_1) \right\}.$$

Let $U_1' \in \mathbb{R}^{|\Theta^*|\times|Q_2|\times|Q_3|}$ be such that, for each (θ, q_2, q_3), $U_1'(\theta,q_2,q_3) = \sum_{q_1\in Q_1} U_1(\theta,q_1,q_2,q_3)\bar{\beta}(q_1)$. If Θ^* is a singleton, the problem becomes

$$\arg\max_{\sigma_2,\sigma_3} \left\{ \sum_{(q_2,q_3)\in\zeta(\sigma_2)\times\zeta(\sigma_3)} U_1'(q_1,q_2) \right\}.$$

This is a joint best-response problem between receivers 2 and 3, which is known to be NP-hard [167]. Therefore, the separation problem for the constraints corresponding to the primal variables γ is NP-hard, which implies that it is NP-hard to solve $\mathcal{D}_B$. □

As the optimal solution values of OPT-EA and $\mathcal{D}_B$ coincide as a consequence of strong duality, Theorem 10.5 implies that computing the optimal solution value of OPT-EA is also NP-hard.

CHAPTER 11

Conclusions and Future Research

The research on equilibrium computation in general-sum, multi-player, sequential games has not yet reached the level of maturity reached in the two-player, zero-sum setting, where it is possible to compute strong solutions in theory and practice. In these settings, equilibrium selection problems may render the choice of the appropriate solution concept not obvious, since the Nash equilibrium may not be the appropriate one. Many practical scenarios allow for some form of communication, mitigating the equilibrium selection issue. In this thesis, we studied multi-player problems where players can reach some form of coordination via preplay communication.

First, we focused on problems where a team of agents faces an adversary in a sequential interaction. We defined different solution concepts for this setting, in order to take into account different communication capabilities of team members. The proposed solution concepts were compared via the analysis of inefficiency indexes which measure the effectiveness of different forms of communication in terms of expected utility for the team. These indices were also analyzed empirically, and, interestingly, preplay communication emerged as a good trade-off between communication requirements and attained expected utility. We described how to find an optimal solution in each team's coordination setting, and tested these techniques on

imperfect-information multiplayer test instances. Then, we focused on the computation of an approximate solution in the case in which team members are allowed to exchange messages only before the beginning of the game. In doing so, we highlighted a strong analogy with imperfect-recall games, and propose a new game representation, which can also be applied to this setting. Then, we used the new representation to derive an auxiliary construction that allows us to map the problem of finding an optimal coordinated strategy for the team to the well-understood Nash equilibrium-finding problem in a (larger) two-player zero-sum perfect-recall game. By reasoning over the auxiliary game, we devised an anytime algorithm, *fictitious team-play*, that is guaranteed to converge to an optimal coordinated strategy for the team against an optimal opponent.

In the future, it would be interesting to develop a scalable end-to-end approach to learning an ε-TMECor without prior domain knowledge. Adapting FTP for this purpose presents a number of challenges, the main one being the problem of estimating team's joint best-responses. There exist sample-based versions of fictitious play for the two-player, zero-sum setting [97, 96]. However, Heinrich and Silver [96] employed standard reinforcement learning techniques (*i.e.*, DQN [129]) to compute approximate best responses which are not appropriate for our setting. Adapting existing techniques originally developed for the two-player, zero-sum setting would possibly result in considerable improvements in the dimension of the problems that can be solved. For example, we are interested in understanding which is the best way to combine CFR and our auxiliary game structure, and how to exploit depth-limited solving techniques (see, *e.g.*, [35]) in games involving teams. Finally, as pointed out by Celli et al. [44], the study of algorithms for team games could shed further light on how to deal with imperfect-recall games, that are receiving increasing attention in the community due to the application of imperfect-recall abstractions to the computation of strategies for large sequential games [173, 113, 33, 78, 119, 45].

In the second part of the thesis, we focused on the problem of coordination when players may have discording objectives. We studied the problem of computing social-welfare-maximizing correlated equilibria when players have limited communication capabilities, and can observe signals only before the beginning of the game. We showed that approximating an optimal CE is not in Poly-APX even in two player games (without chance moves), unless P=NP. Motivated by this hardness result, we studied the complexity of computing CCEs. We showed that an optimal CCE can be found in polynomial time in two-player (general-sum) sequential games without chance moves, and identified the conditions for which finding an

optimal CCE is NP-hard. We characterized the approximation complexity of finding social-welfare-maximizing CCEs by showing that the problem is not in Poly-APX, unless P=NP, in games with three or more players (chance included). Then, we described algorithms to compute CCEs in general-sum, multi-player, sequential games. First, we provided a column generation framework to compute optimal CCEs, and showed how to generalize it to the *hard cases* of the problem. Then, we focused on the problem of computing *an* ε-CCE. We devised an enhanced version of CFR which computes an averaged correlated strategy which is guaranteed to converge to an approximate CCE.

In the future, it would be interesting to further improve the scalability of our methods to compute optimal CCEs to tackle games of even larger size. Among the possible techniques to achieve this, we mention the adoption of heuristics for solving the pricing oracle, the use of stabilization techniques as well as techniques for achieving a speedup in cutting plane and column generation methods [3, 52]. Moreover, a CCE strategy profile could be employed as a starting point to approximate tighter solution concepts that admit some form of correlation. This could be the case, *e.g.*, of the TMECor, which we used to model collusive behaviors and interactions involving teams. Finally, it would be interesting to further investigate whether it is possible to define regret-minimizing procedures for general EFGs leading to refinements of the CCEs, such as CEs and EFCEs.[1] This begets new challenging problems in the study of how to minimize regret in structured games.

The last part of the thesis was devoted to the study of coordination in information-structure design problems. We extended the Bayesian persuasion framework to model problems where receivers (with externalities) can interact in a sequential game. We showed how to address the problem via the notion of *ex ante* persuasive signaling scheme, where the receivers commit to following the sender's recommendations having observed only the signaling scheme. *Ex ante* persuasive signaling schemes can provide the sender with a utility that can be arbitrarily larger than that provided by *ex interim* persuasive signaling schemes. Motivated by the hardness results for the *ex interim* setting with simultaneous moves, we studied the problem of computing optimal *ex ante* signaling schemes. We showed that an optimal *ex ante* signaling scheme may be computed in polynomial time in settings with two receivers and independent action types. Moreover, we showed that this result cannot be extended to settings with more than two receivers,

[1] Farina et al. [71] develop a regret minimization algorithm for computing an ε-EFCE. However, they focus on the two-player, no-chance-moves setting.

as the problem of computing an optimal *ex ante* signaling scheme becomes NP-hard.

In the future, we are interested both in combining other forms of correlation with the Bayesian persuasion framework, *e.g.*, extensive-form correlation as defined by von Stengel and Forges [167], and in investigating forms of perfection in sequential information-design problems. Moreover, the Bayesian persuasion framework still lacks a rigorous experimental evaluation assessing the practical performances of the known algorithms. We suspect new techniques will have to be developed to compute optimal signaling schemes for large game instances.

APPENDIX $\mathcal{A}$

CFR-S: Omitted Proofs

The theoretical guarantees of CFR-S (Section 8.2.2) can be derived via the (almost) direct application of the framework by Farina et al. [67]. For the sake of completeness, we explicitly prove CFR-S's guarantees in the following.

At each iteration t, let $\sigma_i^t \in \Sigma_i$ be the normal-form plan sampled by player i and $\sigma_{-i}^t \in \Sigma_{-i}$ be the plans drawn by the other players. The utility experienced by player i at stage t is denoted by $u_i^t(\sigma_i^t) \coloneqq u_i(\sigma_i^t, \sigma_{-i}^t)$. The players' observations in CFR-S call for a slight variation in the definition of cumulative regret. After T iterations, we define the cumulative regret experienced by player i as

$$\tilde{R}_i^T \coloneqq \max_{\hat{\sigma}_i \in \Sigma_i} \sum_{t=1}^{T} \left(u_i^t(\hat{\sigma}_i) - u_i^t(\sigma_i^t) \right). \tag{A.1}$$

The connection between the cumulative regret and the set of CCEs remains unchanged when the regret is defined as in Equation (A.1), as shown by the following result (whose proof is similar to that by Hart and Mas-Colell [92, Proposition in Section 3]).

Theorem A.1. *If* $\limsup_{T\to\infty} \frac{1}{T}\tilde{R}_i^T \leqslant 0$ *almost surely for each player* $i \in \mathcal{P}$*, then the empirical frequency of play* $\bar{x}^T$ *converges almost surely as* $T \to \infty$ *to the set of CCEs.*

Proof. By definition of cumulative regret, and by taking its average, we have

$$\limsup_{T\to\infty} \frac{1}{T} \max_{\hat{\sigma}_i \in \Sigma_i} \sum_{t=1}^{T} \left(u_i^t(\hat{\sigma}_i) - u_i^t(\sigma_i^t)\right) \leqslant 0,$$

which holds almost surely. Let $\sigma^t := (\sigma_i^t, \sigma_{-i}^t)$. It follows that, for each normal-form plan $\hat{\sigma}_i \in \Sigma_i$ we have

$$\frac{1}{T}\sum_{t=1}^{T} \left(u_i(\hat{\sigma}_i, \sigma_{-i}^t) - u_i(\sigma^t)\right) = \sum_{\sigma\in\Sigma} \bar{x}^T(\sigma) \left(u_i(\hat{\sigma}_i, \sigma_{-i}) - u_i(\sigma)\right).$$

Where $\bar{x}^T(\sigma)$ is the empirical frequency of σ after T iterations. On any subsequence where $\bar{x}^T$ converges, that is $\bar{x}^T \to x^* \in \mathcal{X}$, it holds almost surely, for each $\hat{\sigma}_i \in \Sigma_i$ that

$$\sum_{\sigma\in\Sigma} \bar{x}^T(\sigma) \left(u_i(\hat{\sigma}_i, \sigma_{-i}) - u_i(\sigma)\right) \to \sum_{\sigma\in\Sigma} x^*(\sigma) \left(u_i(\hat{\sigma}_i, \sigma_{-i}) - u_i(\sigma)\right).$$

The result immediately holds for Definition 2.6. □

In the following, we follow the approach of Farina et al. [67] to show how to decompose $\tilde{R}_i^T$ into regret terms which are computed locally at player i's infosets. This allows us to avoid working with the (exponential-sized) normal form of an EFG even if $\tilde{R}_i^T$ is defined over player i's normal-form plans. $\tilde{R}_i^T$ can be minimized via the minimization of other suitably defined regrets computed locally at player i's infosets. In order to do this, we use the idea of *laminar regret decomposition* [67], but reasoning only on vertices of $\mathcal{X}_i$.

Given $\sigma_i \in \Sigma_i$, we denote by $\sigma_i(I)$ the action selected in σ_i at infoset $I \in \mathcal{I}_i$. Moreover, $\sigma_{i\downarrow I}$ is the (sub)vector containing the actions selected in σ_i at $I \in \mathcal{I}_i$ and all its descendant infosets.

First, we denote with $u_{i,I}^t : A(I) \to \mathbb{R}$ the *immediate utility* observed by player i at infoset $I \in \mathcal{I}_i$, during iteration t. For every $a \in A(I)$, $u_{i,I}^t(a)$ is the utility experienced by player i if the game ends after playing a at I, without passing through another player i's infoset.

Then, the following is player i's utility attainable at infoset $I \in \mathcal{I}_i$ when a normal-form plan $\hat{\sigma}_i \in \Sigma_i$ is selected:

$$\hat{V}_I^t(\hat{\sigma}_{i\downarrow I}) := u_{i,I}^t(\hat{\sigma}_{i\downarrow I}(I)) + \sum_{I' \in \mathcal{C}_{I,\hat{\sigma}_{i\downarrow I}(I)}} \hat{V}_{I'}^t(\hat{\sigma}_{i\downarrow I'}), \tag{A.2}$$

where $\mathcal{C}_{I,a} \subseteq \mathcal{I}_i$ is the set of possible next player i's infosets, given that she played action $a \in A(I)$ at infoset $I \in \mathcal{I}_i$. We introduce a parameterized utility function, which is used to define regrets locally at each infoset, and reads as follows:

$$\hat{u}^t_{i,I} : a \in A(I) \mapsto u^t_{i,I}(a) + \sum_{I' \in \mathcal{C}_{I,a}} \hat{V}^t_{I'}(\sigma^t_{i\downarrow I'}). \tag{A.3}$$

The utility function $\hat{u}^t_{i,I}$ preserves convexity of u^t_i. Finally, we modify the notion of *laminar regret*, as

$$\hat{R}^t_I := \max_{a \in A(I)} \sum_{t=1}^{T} \hat{u}^t_{i,I}(a) - \sum_{t=1}^{T} \hat{u}^t_{i,I}(\sigma^t_i(I)). \tag{A.4}$$

Let $V^t_I := \hat{V}^t_I(\sigma^t_{i,\downarrow I})$. Then, we introduce the cumulative regret at infoset $I \in \mathcal{I}_i$, defined as

$$R^T_{\downarrow I} := \max_{\hat{\sigma}_{i\downarrow I}} \sum_{t=1}^{T} \hat{V}^t_I(\hat{\sigma}_{i\downarrow I}) - \sum_{t=1}^{T} V^t_I. \tag{A.5}$$

Lemma A.1. *The cumulative regret at each infoset $I \in \mathcal{I}_i$ can be decomposed as*

$$R^T_{\downarrow I} = \max_{a \in A(I)} \left(\sum_{t=1}^{T} \hat{u}^t_{i,I}(a) + \sum_{I' \in \mathcal{C}_{I,a}} R^T_{\downarrow I'} \right) - \sum_{t=1}^{T} \hat{u}^t_{i,I}(\sigma^t_i(I)).$$

Proof. By definition of cumulative regret at $I \in \mathcal{I}_i$ we have that:

$$\begin{aligned} R^T_{\downarrow I} &:= \max_{\hat{\sigma}_{i\downarrow I}} \sum_{t=1}^{T} \hat{V}^t_I(\hat{\sigma}_{i\downarrow I}) - \sum_{t=1}^{T} V^t_I = \\ &= \max_{\hat{\sigma}_{i\downarrow I}} \sum_{t=1}^{T} \left(u^t_{i,I}(\hat{\sigma}_{i\downarrow I}(I)) + \sum_{I' \in \mathcal{C}_{I,\hat{\sigma}^t_{i\downarrow I}(I)}} \hat{V}^t_{I'}(\hat{\sigma}_{i\downarrow I'}) \right) - \sum_{t=1}^{T} V^t_I = \\ &= \max_{a \in A(I)} \left(\sum_{t=1}^{T} u^t_{i,I}(a) + \sum_{I' \in \mathcal{C}_{I,a}} \max_{\hat{\sigma}_{i\downarrow I'}} \sum_{t=1}^{T} \hat{V}^t_{I'}(\hat{\sigma}_{i\downarrow I'}) \right) - \sum_{t=1}^{T} V^t_I. \end{aligned}$$

Then, by employing Equation (A.5), we get

$$R^T_{\downarrow I} = \max_{a \in A(I)} \left(\sum_{t=1}^{T} u^t_{i,I}(a) + \sum_{I' \in \mathcal{C}_{I,a}} \left(R^T_{\downarrow I'} + \sum_{t=1}^{T} V^t_{I'} \right) \right) - \sum_{t=1}^{T} V^t_I.$$

Finally, we obtain the result by rewriting terms according to Equation (A.3). □

The following theorem shows that, in order to minimize $\tilde{R}_i^T$, it is enough to minimize the laminar regret locally at each $I \in \mathcal{I}_i$ as defined in Equation (A.4).

Lemma A.2. *The cumulative regret $\tilde{R}_i^T$ satisfies the following:*

$$\tilde{R}_i^T \leqslant \max_{\hat{\sigma}_i \in \Sigma_i} \sum_{I \in \mathcal{I}_i} \rho_I^{\hat{\sigma}_i} \hat{R}_I^T.$$

Proof. Consider a generic infoset $I \in \mathcal{I}_i$. By exploiting Lemma A.1 and Definition A.4, we can write:

$$\begin{aligned} R_{\downarrow I}^T &= \max_{a \in A(I)} \left(\sum_{t=1}^T \hat{u}_{i,I}^t(a) + \sum_{I' \in \mathcal{C}_{I,a}} R_{\downarrow I'}^T \right) - \sum_{t=1}^T \hat{u}_{i,I}^t(\sigma_i^t(I)) \leqslant \\ &\leqslant \max_{a \in A(I)} \sum_{t=1}^T \hat{u}_{i,I}^t(a) + \max_{a \in A(I)} \sum_{I' \in \mathcal{C}_{I,a}} R_{\downarrow I'}^T - \sum_{t=1}^T \hat{u}_{i,I}^t(\sigma_i^t(I)) = \\ &= \hat{R}_I^T + \max_{a \in A(I)} \sum_{I' \in \mathcal{C}_{I,a}} R_{\downarrow I'}^T. \end{aligned}$$

By starting from the root of the game and applying the above equation inductively, we obtain our result. □

The last result provides an immediate proof of the following.

Theorem 8.1. *The empirical frequency of play $\bar{x}^T$ obtained with CFR-S converges to a CCE almost surely, for $T \to \infty$.*

Proof. CFR-S minimizes each laminar regret $\hat{R}_I^T$, as defined in Equation (A.4), through standard RM, which guarantees that $\limsup_{T \to \infty} \frac{1}{T} \hat{R}_I^T \leqslant 0$ almost surely. Therefore, $\limsup_{T \to \infty} \frac{1}{T} \tilde{R}_i^T \leqslant 0$ almost surely (Lemma A.2), which implies that the empirical frequency of play converges almost surely to a CCE for $T \to \infty$ (Theorem A.1). □

Finally, we observe that, at each iteration t and infoset $I \in \mathcal{I}_i$, $\sigma_i^t(I)$ is selected according to the strategy $\pi_i^t(I, \cdot)$ recommended by the regret minimizer at infoset I. Thus, σ_i^t is drawn with probability $\prod_{I \in \mathcal{I}_i} \pi_i^t(I, \sigma_i^t(I))$, which is equal to $x_i^t(\sigma_i^t)$, where $x_i^t \in \mathcal{X}_i$ is the normal-form strategy realization equivalent to the behavioral strategy π_i^t.

APPENDIX $\mathcal{B}$

CFR-Jr: Omitted Proofs

In order to give the full proof of Theorem 8.2, we first need to prove two lemmas concerning the existence of a normal-form plan $\bar{\sigma}_i$ such that $\bar{\omega} = \min_{z \in Z(\bar{\sigma}_i)} \omega(z) > 0$ whenever the vector ω has at least a strictly positive component.

To simplify the presentation, we introduce some additional notation. Extending the definition of $Z(\sigma_i)$, let $Z(I, a)$ be the set of terminal nodes potentially reachable from infoset $I \in \mathcal{I}_i$ when player i selects $a \in A(I)$. Moreover, we denote by $Z(\sigma_i, I, a)$ the set of terminal nodes potentially reachable from I after playing action a at I, and then following the actions prescribed by $\sigma_i \in \Sigma_i$. $Z(I)$ and $Z(\sigma_i, I)$ are defined analogously.

Observe that, in Line 5 of Algorithm 8.5, the normal-form plan $\bar{\sigma}_i \in \arg\max_{\sigma_i \in \Sigma_i} \min_{z \in Z(\sigma_i)} \omega(z)$ can be recursively built, while traversing the game tree. Let σ_i^ω be a subset of Σ_i recursively defined as follows:

$$\Sigma_i^\omega := \left\{ \bar{\sigma}_i \in \Sigma_i | \forall I \in \mathcal{I}_i, \bar{\sigma}_i(I) \in \arg\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) \right\}. \tag{B.1}$$

Any $\bar{\sigma}_i \in \Sigma_i^\omega$ is a feasible result for Line 5 in Algorithm 8.5.

Lemma B.1. *Given $\bar{\sigma}_i \in \Sigma_i^\omega$, it holds that*

$$\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = 0 \qquad \forall I \in \mathcal{I}_i,$$

if and only if $\omega = \mathbf{0}$.

Proof. The proof is by induction on the depth of the game tree. Let $\mathcal{C}_{I,a}$ be the set of player i's infosets immediately reachable by playing action $a \in A(I)$ at infoset $I \in \mathcal{I}_i$.

As for the base case of the induction, let us consider $I \in \mathcal{I}_i$ such that $\mathcal{C}_{I,a} = \varnothing$ for all $a \in A(I)$. By the definition of ω we have:

$$\omega(z) = \omega(z') = \rho^{\pi_i}(I)\pi_i(I, a),$$

for each $a \in A(I)$, and each pair $z, z' \in Z(I, a)$. This implies

$$\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = \max_{a \in A(I)} \rho^{\pi_i}(I)\pi_i(I, a) = \max_{z \in Z(I)} \omega(z).$$

Clearly the max of a non-negative function over a set is zero iff the function is zero for all the elements of the set. Then,

$$\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = \max_{z \in Z(I)} \omega_z = 0$$

iff $\omega(z) = 0$ for all $z \in Z(I)$.

As for the inductive step, let us consider a generic infoset $I \in \mathcal{I}_i$. It holds that $Z(\bar{\sigma}_i, I, \bar{a}) = Z(\bar{\sigma}_i, I)$ if

$$\bar{a} \in \arg\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z). \tag{B.2}$$

By reasoning as above, we can conclude that $\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = 0$ iff $\min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = 0$ for all $a \in A(I)$. Now, take any pair $(a', I') \in A(I) \times \mathcal{C}_{I,a'}$, by applying the above observation it follows:

$$\max_{a \in A(I')} \min_{z \in Z(\bar{\sigma}_i, I', a)} \omega(z) = \min_{z \in Z(\bar{\sigma}_i, I', \bar{a})} \omega(z) = \min_{z \in Z(\bar{\sigma}_i, I')} \omega(z),$$

where $\bar{a}$ is computed as in (B.2). Being I' a descendant of I, we have that $Z(I') \subseteq Z(I, a')$ and, in particular, $Z(\bar{\sigma}_i, I') \subseteq Z(\bar{\sigma}_i, I, a')$. Thus, $\min_{z \in Z(\bar{\sigma}_i, I, a')} \omega_i(z) = 0$ implies that $\min_{z \in Z(\bar{\sigma}_i, I')} \omega_i(z) = 0$. By the induction hypothesis, we have that for $I' \in \mathcal{I}_i$ following $I \in \mathcal{I}_i$, it holds

$$\max_{a \in A(I')} \min_{z \in Z(\bar{\sigma}_i, I', a)} \omega_i(z) = 0$$

iff $\omega(z) = 0$ for every $z \in Z(I')$. Then,

$$\max_{a \in A(I)} \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = 0 \quad \Leftrightarrow \quad \min_{z \in Z(\bar{\sigma}_i, I, a)} \omega(z) = 0 \quad \forall a \in A(I),$$

where the right term is true iff

$$\max_{a \in A(I')} \min_{z \in Z(\bar{\sigma}_i, I', a)} \omega(z) = 0 \quad \forall a' \in A(I), I' \in \mathcal{C}_{I,a'}.$$

This holds, by inductive hypothesis, iff

$$\omega(z) = 0 \quad \forall z \in Z(I'), \quad \forall a' \in A(I), I' \in \mathcal{C}_{I,a'}.$$

Given that $Z(I) = \bigcup_{a' \in A(I), I' \in \mathcal{C}_{I,a'}} Z(I')$, the last condition holds iff $\omega(z) = 0$ for all $z \in Z(I)$, which concludes the proof. □

Lemma B.2. *If $\omega > \mathbf{0}$, then a normal-form plan $\bar{\sigma}_i \in \Sigma_i^\omega$ is such that $\min_{z \in Z(\bar{\sigma}_i)} \omega_z > 0$.*

Proof. Let $\varnothing$ be the root infoset of player i. Observe that $Z(\bar{\sigma}_i) = Z(\bar{\sigma}_i, \varnothing, \bar{a})$ if $\bar{a} \in \arg\max_{a \in A(\varnothing)} \min_{z \in Z(\bar{\sigma}_i, \varnothing, a)} \omega(z)$. Applying Lemma B.1 to infoset $\varnothing$, we have that

$$\max_{a \in A(\varnothing)} \min_{z \in Z(\bar{\sigma}_i, \varnothing, a)} \omega(z) = \min_{z \in Z(\bar{\sigma}_i)} \omega(z) = 0$$

iff $\omega(z) = 0$ for every $z \in Z(\varnothing) = Z$. Since $\omega(z) \geqslant 0$ for all $z \in Z$, we have $\min_{z \in Z(\bar{\sigma}_i)} \omega(z) > 0$ iff $\omega > \mathbf{0}$. Being this last condition always verified within the main loop of Algorithm 8.5, we have that a normal-form plan $\bar{\sigma}_i \in \arg\max_{\sigma_i \in \Sigma_i} \min_{z \in Z(\sigma_i)} \omega_z$ is such that $\min_{z \in Z(\bar{\sigma}_i)} \omega_z > 0$. □

Theorem 8.2. *Algorithm 8.5 outputs a normal-form strategy $x_i \in \mathcal{X}_i$ realization equivalent to the given behavioral strategy π_i. It runs in time $O(|Z|^2)$, and x_i has support size of at most $|Z|$.*

Proof. **Time complexity**. We have that $\bar{\omega}\, \rho^{\bar{\sigma}_i}(z)$ is equal to

$$\bar{\omega} = \min_{z \in Z(\bar{\sigma}_i)} \omega(z)$$

for all $z \in Z(\bar{\sigma}_i)$, and equal to zero for all $z \notin Z(\bar{\sigma}_i)$. Then, after each iteration, $\omega \geqslant \mathbf{0}$. Moreover it holds $\omega(\bar{z}) = 0$, for each $\bar{z} \in \arg\min_{z \in Z(\bar{\sigma}_i)} \omega(z)$. Then, at each iteration, at least one component of ω goes from being > 0, to 0. Given that ω is always non-negative, we have that the vector ω is zeroed in at most $|Z|$ iterations.

Each iteration runs in $O(\max\{|\mathcal{I}_i|, |Z|\})$, as $\bar{\sigma}_i$ can be recursively computed by iterating on the infosets in a bottom-up fashion, while each ω update needs to consider each terminal node at most once. Given that for each non-degenerate game tree (*i.e.*, $A(I) > 1$ for all $I \in \mathcal{I}_i$) we have $|\mathcal{I}_i| \leqslant |Z|$, the overall complexity of the algorithm is $O(|Z|^2)$.

Support size. No normal-plan can be selected more than once because: i) after $\bar{\sigma}_i$ is selected, at least one component of ω in $Z(\bar{\sigma}_i)$ is zeroed; ii) $\bar{\sigma}_i$ is selected so that $\min_{z \in Z(\bar{\sigma}_i)} \omega(z) > 0$. Then, the support of x_i has size equal to the number of normal-form plans $\bar{\sigma}_i$ selected at each iteration of Algorithm 8.5, which is at most $|Z|$.

Realization equivalence. Let $\bar{\sigma}_i^k$ be the normal-form plan selected at the k-th iteration. By recursively expanding

$$\omega(z) \leftarrow \omega(z) - \bar{\omega}\, \rho^{\bar{\sigma}_i}(z)$$

we obtain the following (for clarity, we add apices indicating the iteration):

$$\begin{aligned}
\omega^k(z) &= \omega^{k-1}(z) - \rho^{\bar{\sigma}_i^k}(z) \min_{z' \in Z(\bar{\sigma}_i^{k-1})} \omega^{k-1}(z') = \\
&= \omega^{k-2}(z) - \rho^{\bar{\sigma}_i^{k-1}}(z) \min_{z' \in Z(\bar{\sigma}_i^{k-2})} \omega^{k-2}(z') - \rho^{\bar{\sigma}_i^k}(z) \min_{z' \in Z(\bar{\sigma}_i^{k-1})} \omega^{k-1}(z') = \\
&= \ldots = \omega^0(z) - \sum_{k'=1}^{k} \rho^{\bar{\sigma}_i^{k'}}(z) \min_{z' \in Z(\bar{\sigma}_i^{k'-1})} \omega^{k'-1}(z').
\end{aligned}$$

Suppose that the algorithm alts at iteration k. Then $\omega^k = \mathbf{0}$, which gives:

$$\omega^0(z) = \sum_{k'=1}^{k} \rho^{\bar{\sigma}_i^{k'}}(z) \min_{z' \in Z(\bar{\sigma}_i^{k'-1})} \omega^{k'-1}(z').$$

Finally, we show that x_i and π_i are realization equivalent by checking that they force the same distribution over Z. We have, for each $z \in Z$:

$$\begin{aligned}
\rho^{x_i}(z) &= \sum_{\sigma_i \in \Sigma_i} \rho^{\sigma_i}(z) x_i(\sigma_i) = \\
&= \sum_{\sigma_i \in \{\bar{\sigma}_i^{k'}\}_1^k} \rho^{\sigma_i}(z) x_i(\sigma_i) = \sum_{k'=1}^{k} \rho^{\bar{\sigma}_i^{k'}}(z) x_i(\bar{\sigma}_i^{k'}) = \\
&= \sum_{k'=1}^{k} \rho^{\bar{\sigma}_i^{k'}}(z) \min_{z' \in Z(\bar{\sigma}_i^{k'-1})} \omega^{k'-1}(z') = \rho^{\pi_i}(z),
\end{aligned}$$

where π_i is the behavioral strategy given in input. This concludes the proof. □

Theorem 8.3. *If $\frac{1}{T}R_i^T \leqslant \varepsilon$ for each player $i \in \mathcal{P}$, then $\bar{x}^T$ obtained with CFR-Jr is an ε-CCE.*

Proof. First, let us recall that $x^t \in \mathcal{X}$ is defined in such a way that $x^t(\sigma) = \prod_{i\in\mathcal{P}} x_i^t(\sigma_i)$ for every joint normal-form plan $\sigma \in \Sigma$, with $\sigma = (\sigma_i)_{i\in\mathcal{P}}$. By assumption, $\frac{1}{T}R_i^T \leqslant \varepsilon$ implies the following:

$$\max_{\hat{\sigma}_i\in\Sigma_i}\left(\sum_{t=1}^{T}\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\hat{\sigma}_i,\sigma_{-i})\prod_{j\neq i\in\mathcal{P}} x_j^t(\sigma_j) + \right.$$
$$\left. - \sum_{t=1}^{T}\sum_{\sigma_i\in\Sigma_i}\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\sigma_i,\sigma_{-i})\prod_{j\in\mathcal{P}} x_j^t(\sigma_j)\right) \leqslant \varepsilon T.$$

Moreover, since the condition holds for every $i \in \mathcal{P}$, by re-writing the max operator we get, $\forall i \in \mathcal{P}, \hat{\sigma}_i \in \Sigma_i$:

$$\sum_{t=1}^{T}\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\hat{\sigma}_i,\sigma_{-i})\prod_{j\neq i\in\mathcal{P}} x_j^t(\sigma_j) +$$
$$- \sum_{t=1}^{T}\sum_{\sigma_i\in\Sigma_i}\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\sigma_i,\sigma_{-i})\prod_{j\in\mathcal{P}} x_j^t(\sigma_j) \leqslant \varepsilon T.$$

Since $\sum_{\sigma_i\in\Sigma_i} x_i^t(\sigma_i) = 1$, it follows that $\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\hat{\sigma}_i,\sigma_{-i})\prod_{j\neq i\in\mathcal{P}} x_j^t(\sigma_j)$ is equal to $\sum_{\sigma_i\in\Sigma_i}\sum_{\sigma_{-i}\in\Sigma_{-i}} u_i(\hat{\sigma}_i,\sigma_{-i})\prod_{j\in\mathcal{P}} x_j^t(\sigma_j)$. Thus,

$$\sum_{t=1}^{T}\sum_{\sigma_i\in\Sigma_i}\sum_{\sigma_{-i}\in\Sigma_{-i}}\prod_{j\in\mathcal{P}} x_j^t(\sigma_j)\left(u_i(\hat{\sigma}_i,\sigma_{-i}) - u_i(\sigma_i,\sigma_{-i})\right) \leqslant \varepsilon T \quad \forall i \in \mathcal{P}, \hat{\sigma}_i \in \Sigma_i.$$

Using the definition of $\bar{x}^T$, we obtain

$$\sum_{t=1}^{T}\sum_{\sigma_i\in\Sigma_i}\sum_{\sigma_{-i}\in\Sigma_{-i}} \bar{x}^T(\sigma_i,\sigma_{-i})\left(u_i(\hat{\sigma}_i,\sigma_{-i}) - u_i(\sigma_i,\sigma_{-i})\right) \leqslant \varepsilon \quad \forall i \in \mathcal{P}, \hat{\sigma}_i \in \Sigma_i,$$

which proves that $\bar{x}^T$ is an ε-CCE. □

Bibliography

[1] R. Alonso and O. Câmara, "Persuading voters," *AM ECON REV*, vol. 106, no. 11, pp. 3590–3605, 2016.

[2] S. Alpern and W. S. Lim, "The symmetric rendezvous-evasion game," *SIAM Journal on Control and Optimization*, vol. 36, no. 3, pp. 948–959, 1998.

[3] E. Amaldi, S. Coniglio, and S. Gualandi, "Coordinated cutting plane generation via multi-objective separation," *Mathematical Programming*, vol. 143, no. 1, pp. 87–110, Feb 2014.

[4] I. Arieli and Y. Babichenko, "Private Bayesian persuasion," *Available at SSRN 2721307*, 2016.

[5] I. Ashlagi, D. Monderer, and M. Tennenholtz, "On the value of correlation," *Journal of Artificial Intelligence Research*, vol. 33, pp. 575–613, 2008.

[6] R. Aumann, "Subjectivity and correlation in randomized strategies," *Journal of Mathematical Economics*, vol. 1, no. 1, pp. 67–96, 1974.

[7] R. J. Aumann, S. Hart, and M. Perry, "The absent-minded driver," *Games and Economic Behavior*, vol. 20, no. 1, pp. 102–116, 1997.

[8] G. Ausiello, P. Crescenzi, and M. Protasi, "Approximate solution of NP optimization problems," *Theoretical Computer Science*, vol. 150, no. 1, pp. 1–55, 1995.

[9] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media, 2012.

[10] D. Avis, G. D. Rosenberg, R. Savani, and B. Von Stengel, "Enumeration of Nash equilibria for two-player games," *Economic theory*, vol. 42, no. 1, pp. 9–37, 2010.

[11] Y. Babichenko and S. Barman, "Computational aspects of private Bayesian persuasion," *arXiv:1603.01444*, 2016.

[12] A. Badanidiyuru, K. Bhawalkar, and H. Xu, "Targeting and signaling in ad auctions," in *ACM-SIAM SODA*, 2018, pp. 2545–2563.

[13] A. Bardhi and Y. Guo, "Modes of persuasion toward unanimous consent," *THEOR ECON*, vol. 13, no. 3, pp. 1111–1149, 2018.

[14] S. Barman and K. Ligett, "Finding any nontrivial coarse correlated equilibrium is hard," in *Proceedings of the ACM Conference on Economics and Computation (EC)*, 2015, pp. 815–816.

[15] S. Barman, K. Ligett, and G. Piliouras, "Approximating Nash equilibria in tree polymatrix games," in *International Symposium on Algorithmic Game Theory*. Springer, 2015, pp. 285–296.

[16] N. Basilico, A. Celli, G. De Nittis, and N. Gatti, "Team-maxmin equilibrium: efficiency bounds and algorithms," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2017.

[17] M. Benaïm, J. Hofbauer, and S. Sorin, "Stochastic approximations and differential inclusions," *SIAM Journal on Control and Optimization*, vol. 44, no. 1, pp. 328–348, 2005.

[18] D. Bergemann and S. Morris, "Robust predictions in games with incomplete information," *ECONOMETRICA*, vol. 81, no. 4, pp. 1251–1308, 2013.

[19] ——, "Bayes correlated equilibrium and the comparison of information structures in games," *THEOR ECON*, vol. 11, no. 2, pp. 487–522, 2016.

[20] ——, "Information design, Bayesian persuasion, and Bayes correlated equilibrium," *AM ECON REV*, vol. 106, no. 5, pp. 586–91, 2016.

[21] D. Bertsimas and J. N. Tsitsiklis, *Introduction to linear optimization*. Athena Scientific Belmont, 1997, vol. 6.

[22] D. Blackwell, "An analog of the minmax theorem for vector payoffs," *Pacific Journal of Mathematics*, vol. 6, pp. 1–8, 1956.

[23] J. R. Blair, D. Mutchler, and M. Lent, "Perfect recall and pruning in games with imperfect information," *Computational Intelligence*, vol. 12, no. 1, pp. 131–154, 1996.

[24] A. Blum and Y. Mansour, "Learning, regret minimization, and equilibria," *Algorithmic game theory*, pp. 79–102, 2007.

[25] C. Borgs, J. Chayes, N. Immorlica, A. T. Kalai, V. Mirrokni, and C. Papadimitriou, "The myth of the folk theorem," *Games and Economic Behavior*, vol. 70, no. 1, pp. 34–43, 2010.

[26] G. Bornstein, I. Erev, and H. Goren, "The effect of repeated play in the IPG and IPD team games," *Journal of Conflict resolution*, vol. 38, no. 4, pp. 690–707, 1994.

[27] M. Bowling, N. Burch, M. Johanson, and O. Tammelin, "Heads-up limit hold'em poker is solved," *Science*, vol. 347, no. 6218, Jan. 2015.

[28] I. Brocas and J. D. Carrillo, "Influence through ignorance," *The RAND Journal of Economics*, vol. 38, no. 4, pp. 931–947, 2007.

[29] G. W. Brown, "Some notes on computation of games solutions," RAND CORP SANTA MONICA CA, Tech. Rep., 1949.

[30] N. Brown and T. Sandholm, "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals," *Science*, p. eaao1733, 2017.

[31] ——, "Solving imperfect-information games via discounted regret minimization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1829–1836.

[32] ——, "Superhuman ai for multiplayer poker," *Science*, vol. 365, no. 6456, pp. 885–890, 2019.

[33] N. Brown, S. Ganzfried, and T. Sandholm, "Hierarchical abstraction, distributed equilibrium computation, and post-processing, with application to a champion no-limit texas hold'em agent," in *Workshops at the Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[34] N. Brown, C. Kroer, and T. Sandholm, "Dynamic thresholding and pruning for regret minimization," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[35] N. Brown, T. Sandholm, and B. Amos, "Depth-limited solving for imperfect-information games," in *Advances in Neural Information Processing Systems*, 2018, pp. 7663–7674.

[36] N. Burch, M. Moravcik, and M. Schmid, "Revisiting CFR+ and alternating updates," *Journal of Artificial Intelligence Research*, vol. 64, pp. 429–443, 2019.

[37] Y. Cai and C. Papadimitriou, "Simultaneous bayesian auctions and computational complexity," in *ACM EC*, 2014, pp. 895–910.

[38] I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, M. Kyropoulou, B. Lucier, P. Renato, E. Tardos *et al.*, "Bounding the inefficiency of outcomes in generalized second price auctions," *J ECON THEORY*, vol. 156, no. C, pp. 343–388, 2015.

[39] M. Castiglioni, A. Celli, and N. Gatti, "Persuading voters: It's easy to whisper, it's hard to speak loud," *arXiv preprint arXiv:1908.10620*, 2019.

[40] A. Celli and N. Gatti, "Computational results for extensive-form adversarial team games," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[41] A. Celli, S. Coniglio, and N. Gatti, "Bayesian persuasion with sequential games," *arXiv preprint arXiv:1908.00877*, 2019.

[42] ——, "Computing optimal ex ante correlated equilibria in two-player sequential games," in *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*, 2019, pp. 909–917.

[43] A. Celli, A. Marchesi, T. Bianchi, and N. Gatti, "Learning to correlate in multi-player general-sum sequential games," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[44] A. Celli, G. Romano, and N. Gatti, "Personality-based representations of imperfect-recall games," in *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*, ser. AAMAS, 2019, pp. 1868–1870.

[45] J. Čermák, B. Bošansky, and V. Lisy, "An algorithm for constructing and solving imperfect recall abstractions of large extensive-form games," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. AAAI Press, 2017, pp. 936–942.

[46] J. Čermák, B. Bošanskỳ, and M. Pěchouček, "Combining incremental strategy generation and branch and bound search for computing maxmin strategies in imperfect recall games," in *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2017, pp. 902–910.

[47] J. Čermák, B. Bošanskỳ, K. Horák, V. Lisỳ, and M. Pěchouček, "Approximating maxmin strategies in imperfect recall games using a-loss recall property," *International Journal of Approximate Reasoning*, vol. 93, pp. 290–326, 2018.

[48] N. Cesa-Bianchi and G. Lugosi, *Prediction, learning, and games*. Cambridge university press, 2006.

[49] H. Chan, A. X. Jiang, K. Leyton-Brown, and R. Mehta, "Multilinear games," in *International Conference on Web and Internet Economics*. Springer, 2016, pp. 44–58.

[50] X. Chen, X. Deng, and S.-H. Teng, "Settling the complexity of computing two-player Nash equilibria," *Journal of the ACM (JACM)*, vol. 56, no. 3, p. 14, 2009.

[51] X. Chen, Z. Han, H. Zhang, G. Xue, Y. Xiao, and M. Bennis, "Wireless resource scheduling in virtualized radio access networks using stochastic learning," *IEEE Transactions on Mobile Computing*, vol. 17, no. 4, pp. 961–974, 2017.

[52] S. Coniglio and M. Tieves, "On the generation of cutting planes which maximize the bound improvement," in *Experimental Algorithms*. Springer International Publishing, 2015, pp. 97–109.

[53] C. Daskalakis, P. Goldberg, and C. Papadimitriou, "The complexity of computing a Nash equilibrium," *SIAM Journal on Computing*, vol. 39, no. 1, pp. 195–259, 2009.

[54] C. Daskalakis and Q. Pan, "A counter-example to karlin's strong conjecture for fictitious play," in *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 2014, pp. 11–20.

[55] C. Daskalakis, A. Deckelbaum, and A. Kim, "Near-optimal no-regret algorithms for zero-sum games," *Games and Economic Behavior*, vol. 92, pp. 327–348, 2015.

[56] B. DeBruhl, C. Kroer, A. Datta, T. Sandholm, and P. Tague, "Power napping with loud neighbors: optimal energy-constrained jamming and anti-jamming," in *Proceedings of the 2014 ACM conference on Security and privacy in wireless & mobile networks*. ACM, 2014, pp. 117–128.

[57] A. Deligkas, J. Fearnley, T. P. Igwe, and R. Savani, "An empirical study on computing equilibria in polymatrix games," in *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2016, pp. 186–195.

[58] S. Dughmi, "Algorithmic information structure design: a survey," *ACM SIGEC EX*, vol. 15, no. 2, pp. 2–24, 2017.

[59] ——, "On the hardness of designing public signals," *GAME ECON BEHAV*, 2018.

[60] S. Dughmi and H. Xu, "Algorithmic Bayesian persuasion," in *ACM STOC*. ACM, 2016, pp. 412–425.

[61] S. Dughmi, N. Immorlica, and A. Roth, "Constrained signaling in auction design," in *ACM-SIAM SODA*, 2014, pp. 1341–1357.

[62] S. Dughmi and H. Xu, "Algorithmic persuasion with no externalities," in *ACM EC*, 2017, pp. 351–368.

[63] Y. Emek, M. Feldman, I. Gamzu, R. P. Leme, and M. Tennenholtz, "Signaling schemes for revenue maximization," in *ACM EC*, 2012, pp. 514–531.

[64] F. Fang, P. Stone, and M. Tambe, "When security games go green: Designing defender strategies to prevent poaching and illegal fishing," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[65] F. Fang, T. H. Nguyen, R. Pickles, W. Y. Lam, G. R. Clements, B. An, A. Singh, M. Tambe, and A. Lemieux, "Deploying paws: Field optimization of the protection assistant for wildlife security," in *Twenty-Eighth IAAI Conference*, 2016.

[66] G. Farina, A. Celli, N. Gatti, and T. Sandholm, "Ex ante coordination and collusion in zero-sum multi-player extensive-form games," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[67] G. Farina, C. Kroer, and T. Sandholm, "Online convex optimization for sequential decision processes and extensive-form games," *arXiv preprint arXiv:1809.03075*, 2018.

[68] G. Farina, T. Bianchi, and T. Sandholm, "Coarse correlation in extensive-form games," *arXiv preprint arXiv:1908.09893*, 2019.

[69] G. Farina, C. Kroer, N. Brown, and T. Sandholm, "Stable-predictive optimistic counterfactual regret minimization," *arXiv preprint arXiv:1902.04982*, 2019.

[70] G. Farina, C. K. Ling, F. Fang, and T. Sandholm, "Correlation in extensive-form games: Saddle-point formulation and benchmarks," *arXiv preprint arXiv:1905.12564*, 2019.

[71] ——, "Efficient regret minimization algorithm for extensive-form correlated equilibrium," *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[72] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, "Counterfactual multi-agent policy gradients," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[73] F. Forges, "An approach to communication equilibria," *Econometrica*, pp. 1375–1385, 1986.

[74] ——, "Five legitimate definitions of correlated equilibrium in games with incomplete information," *Theory and Decision*, vol. 35, no. 3, pp. 277–310, Nov 1993.

[75] ——, "Correlated equilibrium in games with incomplete information revisited," *Theory and Decision*, vol. 61, no. 4, pp. 329–344, Dec 2006.

[76] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[77] D. Fudenberg, F. Drew, D. K. Levine, and D. K. Levine, *The theory of learning in games*. MIT press, 1998, vol. 2.

[78] S. Ganzfried and T. Sandholm, "Potential-aware imperfect-recall abstraction with earth mover's distance in imperfect-information games," in *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[79] M. R. Garey and D. S. Johnson, *Computers and intractability*. wh freeman New York, 2002, vol. 29.

[80] R. G. Gibson, "Regret minimization in games and the development of champion multiplayer computer poker-playing agents," 2014.

[81] R. G. Gibson and D. Szafron, "Regret minimization in multiplayer extensive games," in *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[82] I. Gilboa and E. Zemel, "Nash and correlated equilibria: Some complexity considerations," *Games and Economic Behavior*, vol. 1, no. 1, pp. 80–93, 1989.

[83] P. W. Goldberg, R. Savani, T. B. Sørensen, and C. Ventre, "On the approximation performance of fictitious play in finite games," *International Journal of Game Theory*, vol. 42, no. 4, pp. 1059–1083, 2013.

[84] I. Goldstein and Y. Leitner, "Stress tests and information disclosure," *J ECON THEORY*, vol. 177, pp. 34–69, 2018.

[85] M. Grötschel, L. Lovász, and A. Schrijver, "The ellipsoid method and its consequences in combinatorial optimization," *Combinatorica*, vol. 1, pp. 169–197, 1981.

[86] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric algorithms and combinatorial optimization*. Springer Science & Business Media, 2012, vol. 2.

[87] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *International Conference on Autonomous Agents and Multiagent Systems*. Springer, 2017, pp. 66–83.

[88] J. Hannan, "Approximation to bayes risk in repeated play," *Contributions to the Theory of Games*, vol. 3, pp. 97–139, 1957.

[89] K. A. Hansen, P. B. Miltersen, and T. B. Sørensen, "Finding equilibria in games of no chance," in *International Computing and Combinatorics Conference*. Springer, 2007, pp. 274–284.

[90] K. A. Hansen, T. D. Hansen, P. B. Miltersen, and T. B. Sørensen, "Approximability and parameterized complexity of minmax values," in *International Workshop on Internet and Network Economics*. Springer, 2008, pp. 684–695.

[91] J. C. Harsanyi, "Games with incomplete information played by bayesian players," *MANAGE SCI*, vol. 14, no. 3, pp. 159–182, 320–334, 486–502, 1967.

[92] S. Hart and A. Mas-Colell, "A simple adaptive procedure leading to correlated equilibrium," *Econometrica*, vol. 68, no. 5, pp. 1127–1150, 2000.

[93] ——, "A general class of adaptive strategies," *Journal of Economic Theory*, vol. 98, no. 1, pp. 26–54, 2001.

[94] J. Hartline, V. Syrgkanis, and E. Tardos, "No-regret learning in bayesian games," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2015, pp. 3061–3069.

[95] J. Håstad, "Some optimal inapproximability results," *Journal of the ACM (JACM)*, vol. 48, no. 4, pp. 798–859, 2001.

[96] J. Heinrich and D. Silver, "Deep reinforcement learning from self-play in imperfect-information games," *arXiv preprint arXiv:1603.01121*, 2016.

[97] J. Heinrich, M. Lanctot, and D. Silver, "Fictitious self-play in extensive-form games," in *International Conference on Machine Learning*, 2015, pp. 805–813.

[98] E. Hendon, H. J. Jacobsen, and B. Sloth, "Fictitious play in extensive form games," *Games and Economic Behavior*, vol. 15, no. 2, pp. 177–202, 1996.

[99] J. Hofbauer and K. Sigmund, "Evolutionary game dynamics," *Bulletin of the American Mathematical Society*, vol. 40, no. 4, pp. 479–519, 2003.

[100] W. Huang and B. von Stengel, "Computing an extensive-form correlated equilibrium in polynomial time," in *International Workshop on Internet and Network Economics*. Springer, 2008, pp. 506–513.

[101] A. Jafari, A. Greenwald, D. Gondek, and G. Ercal, "On no-regret learning, fictitious play, and Nash equilibrium," in *ICML*, vol. 1, 2001, pp. 226–233.

[102] A. X. Jiang and K. Leyton-Brown, "Polynomial-time computation of exact correlated equilibrium in compact games," *Games and Economic Behavior*, vol. 91, pp. 347–359, 2015.

[103] E. Kamenica and M. Gentzkow, "Bayesian persuasion," *AM ECON REV*, vol. 101, no. 6, pp. 2590–2615, 2011.

[104] E. Kamenica, "Bayesian persuasion and information design," *ANNU REV ECON*, vol. 11, 2018.

[105] T. R. Kaplan and S. Zamir, "The strategic use of seller information in private-value auctions," *Hebrew University, Center For Rationality Working Paper*, no. 221, 2000.

[106] S. Karlin, *Mathematical methods and theory in games, programming, and economics*. Addison-Wesley, 1959.

[107] L. G. Khachiyan, "Polynomial algorithms in linear programming," *USSR Computational Mathematics and Mathematical Physics*, vol. 20, no. 1, pp. 53–72, 1980.

[108] E. Kohlberg and J.-F. Mertens, "On the strategic stability of equilibria," *Econometrica: Journal of the Econometric Society*, pp. 1003–1037, 1986.

[109] D. Koller and N. Megiddo, "The complexity of two-person zero-sum games in extensive form," *Games and economic behavior*, vol. 4, no. 4, pp. 528–552, 1992.

[110] D. Koller, N. Megiddo, and B. Von Stengel, "Efficient computation of equilibria for extensive two-person games," *Games and economic behavior*, vol. 14, no. 2, pp. 247–259, 1996.

[111] E. Koutsoupias and C. Papadimitriou, "Worst-case equilibria," in *Annual Symposium on Theoretical Aspects of Computer Science*. Springer, 1999, pp. 404–413.

[112] V. Krishna and T. Sjöström, "Learning in games: Fictitious play dynamics," in *Cooperation: Game-Theoretic Approaches*. Springer, 1997, pp. 257–273.

[113] C. Kroer and T. Sandholm, "Imperfect-recall abstractions with bounds in games," in *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 2016, pp. 459–476.

[114] C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm, "Faster first-order methods for extensive-form game solving," in *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. ACM, 2015, pp. 817–834.

[115] C. Kroer, G. Farina, and T. Sandholm, "Solving large sequential games with the excessive gap technique," in *Advances in Neural Information Processing Systems*, 2018, pp. 864–874.

[116] C. Kroer, K. Waugh, F. Kılınç-Karzan, and T. Sandholm, "Faster algorithms for extensive-form game solving via improved smoothing functions," *Mathematical Programming*, pp. 1–33, 2018.

[117] H. W. Kuhn, "A simplified two-person poker," in *Contributions to the Theory of Games*, ser. Annals of Mathematics Studies, 24, H. W. Kuhn and A. W. Tucker, Eds. Princeton, New Jersey: Princeton University Press, 1950, vol. 1, pp. 97–103.

[118] ——, "Extensive games and the problem of information," in *Contributions to the Theory of Games*, ser. Annals of Mathematics Studies, 28, H. W. Kuhn and A. W. Tucker, Eds. Princeton, NJ: Princeton University Press, 1953, vol. 2, pp. 193–216.

[119] M. Lanctot, R. Gibson, N. Burch, M. Zinkevich, and M. Bowling, "No-regret learning in extensive-form games with imperfect recall," *arXiv preprint arXiv:1205.0622*, 2012.

[120] M. Lanctot, V. Zambaldi, A. Gruslys, A. Lazaridou, K. Tuyls, J. Pérolat, D. Silver, and T. Graepel, "A unified game-theoretic approach to multiagent reinforcement learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 4190–4203.

[121] R. P. Leme, V. Syrgkanis, and É. Tardos, "Sequential auctions and externalities," in *ACM-SIAM SODA*. SIAM, 2012, pp. 869–886.

[122] D. S. Leslie and E. Collins, "Generalised weakened fictitious play," *Games and Economic Behavior*, vol. 56, no. 2, pp. 285 – 298, 2006.

[123] W. S. Lim, "A rendezvous-evasion game on discrete locations with joint randomization," *Advances in Applied Probability*, vol. 29, no. 4, pp. 1004–1017, 1997.

[124] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Information and computation*, vol. 108, no. 2, pp. 212–261, 1994.

[125] R. Lowe, Y. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in Neural Information Processing Systems*, 2017, pp. 6379–6390.

[126] M. Maschler, E. Solan, and S. Zamir, *Game Theory*. Cambridge University Press, 2013.

[127] G. P. McCormick, "Computability of global solutions to factorable nonconvex programs: Part i-convex underestimating problems," *Mathematical programming*, vol. 10, no. 1, pp. 147–175, 1976.

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

[129] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[130] D. Monderer and L. S. Shapley, "Fictitious play property for games with identical interests," *Journal of economic theory*, vol. 68, no. 1, pp. 258–265, 1996.

[131] M. Moravčík, M. Schmid, N. Burch, V. Lisý, D. Morrill, N. Bard, T. Davis, K. Waugh, M. Johanson, and M. Bowling, "Deepstack: Expert-level artificial intelligence in heads-up no-limit poker," *Science*, vol. 356, no. 6337, 2017.

[132] H. Moulin and J.-P. Vial, "Strategically zero-sum games: the class of games whose completely mixed equilibria cannot be improved upon," *International Journal of Game Theory*, vol. 7, no. 3, pp. 201–221, 1978.

[133] R. B. Myerson, "Multistage games with communication," *Econometrica*, vol. 54, no. 2, pp. 323–358, 1986.

[134] R. Myerson, "Incentive compatibility and the bargaining problem," *Econometrica*, vol. 47, no. 1, pp. 61–73, 1979.

[135] J. Nash, "Non-cooperative games," *Annals of mathematics*, pp. 286–295, 1951.

[136] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic game theory*. Cambridge University Press, 2007, vol. 1.

[137] T. R. Palfrey and H. Rosenthal, "A strategic calculus of voting," *Public choice*, vol. 41, no. 1, pp. 7–53, 1983.

[138] C. H. Papadimitriou and T. Roughgarden, "Computing correlated equilibria in multi-player games," *Journal of the ACM (JACM)*, vol. 55, no. 3, p. 14, 2008.

[139] R. Píbil, V. Lisỳ, C. Kiekintveld, B. Bošanskỳ, and M. Pěchouček, "Game theoretic model of strategic honeypot selection in computer networks," in *International Conference on Decision and Game Theory for Security*. Springer, 2012, pp. 201–220.

[140] M. Piccione and A. Rubinstein, "On the interpretation of decision problems with imperfect recall," *Games and Economic Behavior*, vol. 20, no. 1, pp. 3–24, 1997.

[141] Z. Rabinovich, A. Jiang, M. Jain, and H. Xu, "Information disclosure as a means to security," in *AAMAS*, 2015, pp. 645–653.

[142] P. Raghavan and C. D. Tompson, "Randomized rounding: a technique for provably good algorithms and algorithmic proofs," *Combinatorica*, vol. 7, no. 4, pp. 365–374, 1987.

[143] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Qmix: monotonic value function factorisation for deep multi-agent reinforcement learning," *arXiv preprint arXiv:1803.11485*, 2018.

[144] L. Rayo and I. Segal, "Optimal information disclosure," *J POLIT ECON*, vol. 118, no. 5, pp. 949–987, 2010.

[145] N. A. Risk and D. Szafron, "Using counterfactual regret minimization to create competitive multiplayer poker agents," in *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 2010, pp. 159–166.

[146] J. Robinson, "An iterative method of solving a game," *Annals of mathematics*, pp. 296–301, 1951.

[147] I. Romanovskii, "Reduction of a game with complete memory to a matrix game," *Soviet Mathematics*, vol. 3, 1962.

[148] S. M. Ross, "Goofspiel–the game of pure strategy," *Journal of Applied Probability*, vol. 8, no. 3, pp. 621–625, 1971.

[149] T. Roughgarden, "Intrinsic robustness of the price of anarchy," in *Proceedings of the forty-first annual ACM symposium on Theory of computing*. ACM, 2009, pp. 513–522.

[150] A. Saffidine, H. Finnsson, and M. Buro, "Alpha-beta pruning for games with simultaneous moves." in *AAAI Conference on Artificial Intelligence (AAAI)*, 2012.

[151] N. V. Sahinidis, *BARON 17.8.9: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2017.

[152] A. Schlenker, H. Xu, M. Guirguis, C. Kiekintveld, A. Sinha, M. Tambe, S. Y. Sonya, D. Balderas, and N. Dunstatter, "Don't bury your head in warnings: A game-theoretic approach for intelligent allocation of cyber-security alerts." in *IJCAI*, 2017, pp. 381–387.

[153] L. S. Shapley and R. N. Snow, "Basic solutions of discrete games," *Annals of Mathematics Studies*, vol. 24, pp. 27–35, 1950.

[154] L. Shapley, "Some topics in two-person games," *Advances in game theory*, vol. 52, pp. 1–29, 1964.

[155] Y. Shoham and K. Leyton-Brown, *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2009.

[156] E. Solan, "Absorbing team games," *Games and Economic Behavior*, vol. 31, no. 2, pp. 245–261, 2000.

[157] F. Southey, M. Bowling, B. Larson, C. Piccione, N. Burch, D. Billings, and C. Rayner, "Bayes' bluff: opponent modelling in poker," in *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence (UAI)*, 2005, pp. 550–558.

[158] S. Srinivasan, M. Lanctot, V. Zambaldi, J. Pérolat, K. Tuyls, R. Munos, and M. Bowling, "Actor-critic policy optimization in partially observable multiagent environments," in *Advances in Neural Information Processing Systems*, 2018, pp. 3422–3435.

[159] P. Sunehag, G. Lever, A. Gruslys, W. M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Z. Leibo, K. Tuyls *et al.*, "Value-decomposition networks for cooperative multi-agent learning based on team reward," in *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*, 2018, pp. 2085–2087.

[160] J. Swinkels *et al.*, *Subgames and the reduced normal form*. Econometric Research Program, Princeton University, 1989.

[161] M. Tambe, *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge university press, 2011.

[162] O. Tammelin, N. Burch, M. Johanson, and M. Bowling, "Solving heads-up limit Texas hold'em," in *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[163] I. Taneva, "Information design," *AM ECON J-MICROECON*, 2019, forthcoming.

[164] M. Tawarmalani and N. V. Sahinidis, "A polyhedral branch-and-cut approach to global optimization," *Mathematical Programming*, vol. 103, pp. 225–249, 2005.

[165] D. Vermeulen and M. Jansen, "The reduced form of a game," *European Journal of Operational Research*, vol. 106, no. 1, pp. 204–211, 1998.

[166] B. von Stengel, "Efficient computation of behavior strategies," *Games and Economic Behavior*, vol. 14, no. 2, pp. 220 – 246, 1996.

[167] B. von Stengel and F. Forges, "Extensive-form correlated equilibrium: Definition and computational complexity," *Mathematics of Operations Research*, vol. 33, no. 4, pp. 1002–1022, 2008.

[168] B. von Stengel and D. Koller, "Team-maxmin equilibria," *Games and Economic Behavior*, vol. 21, no. 1, pp. 309 – 321, 1997.

[169] B. Von Stengel, "Computing equilibria for two-person games," *Handbook of game theory with economic applications*, vol. 3, pp. 1723–1759, 2002.

[170] X. Wang and T. Sandholm, "Reinforcement learning to play an optimal Nash equilibrium in team markov games," in *Advances in neural information processing systems*, 2003, pp. 1603–1610.

[171] Y. Wang, "Bayesian persuasion with multiple receivers," *Available at SSRN 2625399*, 2013.

[172] K. Waugh, "Abstraction in large extensive games," Master's thesis, University of Alberta, 2009.

[173] K. Waugh, M. Zinkevich, M. Johanson, M. Kan, D. Schnizlein, and M. Bowling, "A practical use of imperfect recall," in *Eighth Symposium on Abstraction, Reformulation, and Approximation*, 2009.

[174] P. C. Wichardt, "Existence of Nash equilibria in finite extensive form games with imperfect recall: A counterexample," *Games and Economic Behavior*, vol. 63, no. 1, pp. 366–369, 2008.

[175] D. P. Williamson and D. B. Shmoys, *The design of approximation algorithms*. Cambridge university press, 2011.

[176] H. Xu, Z. Rabinovich, S. Dughmi, and M. Tambe, "Exploring information asymmetry in two-stage security games," in *AAAI*, 2015, pp. 1057–1063.

[177] H. Xu, R. Freeman, V. Conitzer, S. Dughmi, and M. Tambe, "Signaling in Bayesian Stackelberg games," in *AAMAS*, 2016, pp. 150–158.

[178] H. Xu, "On the tractability of public persuasion with no externalities," *CoRR*, vol. abs/1906.07359, 2019.

[179] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *International Conference on Machine Learning (ICML)*, Washington, DC, USA, 2003, pp. 928–936.

[180] M. Zinkevich, M. Bowling, M. Johanson, and C. Piccione, "Regret minimization in games with incomplete information," in *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, 2007.