Computing Optimal *Ex Ante* Correlated Equilibria in Two-Player Sequential Games

Andrea Celli
Politecnico di Milano
Milan, Italy
andrea.celli@polimi.it

Stefano Coniglio
University of Southampton
Southampton, United Kingdom
s.coniglio@soton.ac.uk

Nicola Gatti
Politecnico di Milano
Milan, Italy
nicola.gatti@polimi.it

ABSTRACT

We investigate the computation of equilibria in extensive-form games when *ex ante* correlation is possible, focusing on correlated equilibria requiring the least amount of communication between the players and the mediator. Motivated by hardness results on *normal-form correlated equilibria*, we investigate whether it is possible to compute *normal-form coarse correlated equilibria* efficiently. We show that an optimal (e.g., social welfare maximizing) *normal-form coarse correlated equilibrium* can be computed in polynomial time in two-player games without chance moves, and that in general multi-player games (including two-player games with chance) the problem is NP-hard. For the two-player case, we provide both a polynomial-time algorithm based on the ellipsoid method and a column generation algorithm based on the simplex method which can be efficiently applied in practice. We also show that the pricing oracle employed in the column generation procedure can be extended to games with two players and chance.

KEYWORDS

Equilibrium computation; correlated equilibrium

ACM Reference Format:
Andrea Celli, Stefano Coniglio, and Nicola Gatti. 2019. Computing Optimal *Ex Ante* Correlated Equilibria in Two-Player Sequential Games. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

1 INTRODUCTION

The computational study of adversarial interactions aiming at finding players' optimal strategies and predicting the most likely outcome of a game is a central problem in Artificial Intelligence. A vast body of literature focuses on the computation of Nash Equilibria (NEs), mainly in two-player zero-sum games [36]. This setting is well understood and, recently, some remarkable results have been achieved by, e.g., Brown and Sandholm [9, 10]. While relevant, this model is rather restrictive, as many practical scenarios are not zero-sum and involve more than two players, and it presents some weaknesses when used as a prescriptive tool, in particular in general-sum games [12, 20]. Indeed, when multiple NEs coexist, the model assumes the lack of communication between the players, preventing them from synchronizing their strategies.

In practical situations where some form of communication is possible, solution concepts different from that of NE are required. The main alternative is the *Correlated Equilibrium* (CE), introduced by Aumann [3]. In a CE, a *device* (i.e., a trusted external mediator) draws strategy profiles from a known joint probability distribution and privately communicates them to each player. The probability distribution induces an equilibrium if each player has no incentive to choose a different strategy from the recommended one, assuming the other players would not deviate either. A variant of the CE is the *Coarse Correlated Equilibrium* (CCE), introduced in [30], which only prevents deviations from happening before knowing the device's recommendation. In normal-form games, CEs and CCEs enjoy some appealing properties that make them plausible solution concepts in many practical scenarios. Specifically, they arises from simple and natural learning dynamics [13, 23], and they can be computed via linear programming on any normal-form game in polynomial time (assuming the number of players is fixed). Moreover, price-of-anarchy analyses show that coarse correlated equilibria characterizing outcomes of no-regret learning dynamics have near-optimal welfare [24, 34]. While *a* CE can be found in polynomial time in some classes of succinctly representable multi-player games, finding an *optimal* CE in these games is, in general, NP-hard [26, 32]. A similar result also holds for the problem of finding an optimal CCE. Barman and Ligett [4] show that for graphical, polymatrix, congestion, and anonymous games the problem is NP-hard.

Sequential games allow for richer forms of interaction among the players than normal-form games, which lead to different forms of correlation whose general understanding is still limited. Most of the works in this area focus on specific classes of games, such as Bayesian games [18, 19] and multi-stage games [17, 31]. In these specific settings, the main solution concepts studied in the literature are the *Normal-Form Correlated Equilibrium* (NFCE), the *Agent-Form Correlated Equilibrium* (AFCE), and the *Communication-Equilibrium.* The first two equilibria only allow for a *unidirectional* communication from the device to the players, while the third equilibrium allows for *bidirectional* communication. The only known results for general extensive-form games are due to von Stengel and Forges [38], who propose the notion of *Extensive-Form Correlated Equilibrium* (EFCE). The complex structure of extensive-form games significantly increases the computational effort required for correlation, as finding an optimal NFCE is NP-hard even with two players [38]. An optimal EFCE can be found efficiently in two-player games without chance moves but, in games with three or more players (including chance), finding an optimal EFCE (or an AFCE) is NP-hard [38]. The only positive result for multi-player games is a polynomial-time algorithm to find *an* EFCE [25].

Correlated equilibria in which recommendations are drawn before the game starts are known as *ex ante* CEs. These equilibria require only unilateral communication from the device to the players. NFCE, AFCE, and EFCE belong to this family and differ in the

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

time at which the recommendations are communicated to players. Specifically, the NFCE requires, for each player, a single interaction with the mediator taking place before the beginning of the game, whereas AFCE and EFCE require a message for each information set reached during the game. As a consequence, AFCE and EFCE are not suited for problems where the agents have limited communication capabilities, a situation which is frequent in practice. This is the case, for instance, of collusion in bidding, where communication during the auction is illegal, and coordinated swindling in public (see also the recent work by Farina et al. [16]). Different forms of correlation have been explored when a team of players faces an adversary [5, 6, 11, 16]. This setting, also known as *ex ante coordination*, is quite different from ours. Our notion of correlation is more flexible as it allows for players with different objectives. Therefore, in our correlation setting individual players have to be incentivized to follow the recommendations of the mediator. In contrast, in the *ex ante coordination* setting there is no need for incentive constraints since team members share their final rewards.

Our Contributions. In this paper, we focus on equilibria involving a low level of communication. A natural question is whether correlation can be reached efficiently when the agents have limited communication capabilities, i.e., when they cannot receive messages during the execution of the game.[1] Motivated by the hardness result for the NFCE, we introduce the notion of *Normal-Form Coarse Correlated Equilibrium* (NFCCE) as the extension of CCE to sequential games. We prove that, unlike the NFCE, the problem of finding an optimal NFCCE admits a polynomial-time algorithm for two-player games without chance moves. In particular, we devise a hybrid formulation for the problem of computing an optimal NFCCE enojoying a polynomial number of constraints and an exponential number of variables. Then, we provide a polynomial-time separation oracle which, together with the ellipsoid algorithm [28], allows us to show that an optimal NFCCE can be computed in polynomial time. This approach cannot be extended to more general settings since, with more than two players (including chance), the problem becomes NP-hard. We also describe a practical algorithm to compute an optimal NFCCE based on column generation, employing different oracles to solve the corresponding pricing problems. In particular, we provide a polynomial-time oracle suitable for the two-player setting, and a MILP oracle which can be adapted to the case of two-player games with nature. The proposed techniques are experimentally evaluated to assess their scalability on game instances from different domains.

2 PRELIMINARIES

We briefly introduce several of the basic concepts we use in the rest of the paper. Further details can be found in [36].

2.1 Game Representations

An extensive-form game Γ has a finite set of players N and a finite set of actions A. Exogenous stochasticity is represented through a non-strategic player c (the nature or chance player). V is the set of non-terminal decision nodes, and $V_i \subseteq V$ is the set of decision nodes belonging to player $i \in N \cup \{c\}$. The set of terminal nodes (leaves) is denoted by L. The function $\iota : V \to N \cup \{c\}$ associates each decision node with the player acting at it. The function $\rho : V \to 2^A$ is the *action function*, assigning with each decision node a set of available actions. The successor function is denoted by $\chi : V \times A \to V \cup L$. Let $U_i : L \to \mathbb{R}$ be the utility function of each $i \in N$. Moreover, let $U = \{U_i\}_{i \in N}$. Finally, for each $i \in N \cup \{c\}$ let H_i be an information partition of V_i such that decision nodes within the same information set $h \in H_i$ are not distinguishable by player i. We let $H = \{H_i\}_{i \in N \cup \{c\}}$. The a function π_c is such that $\pi_c(h, a)$ is the fixed probability that chance select a at $h \in H_c$. Moreover, $\rho(h)$ denotes the set of actions available at $h \in H_i$. We remark that, by definition, $\rho(x_1) = \rho(x_2) = \rho(h)$ for any player $i \in N \cup \{c\}$, information set $h \in H_i$, and $x_1, x_2 \in h$. In this paper, we focus on games with *perfect recall*, i.e., games where, at each stage, all the players recall all the information acquired at earlier stages.

An extensive-form game can be equivalently represented in *normal-form*. Let $P_i = \times_{h \in H_i} \rho(h)$ be the set of pure normal-form plans of player $i \in N$. A normal-form plan $p \in P_i$ specifies an action per information set of player i. The *normal-form* of an extensive-form game is characterized by the same set of players N, actions $P = \times_{i \in N} P_i$, and the set of utility functions $U' = \{U'_i\}_{i \in N}$. Function $U'_i : P \to \mathbb{R}$ denotes the expected payoff obtained by marginalizing with respect to π_c. The *reduced* normal form is obtained by deleting duplicated strategies from the normal form.

2.2 Strategy Representation

A normal-form strategy σ_i for $i \in N$ is defined as the function $\sigma_i : P_i \to \Delta^{|P_i|}$. We denote by Σ_i the normal-form strategy space of player i. A *correlated* (joint) normal-form strategy $\sigma \in \Sigma$ is defined as $\sigma : P \to \Delta^{|P|}$. The size of a normal-form strategy is exponential in the size of the extensive-form tree. This shortcoming can be overcome by exploiting the *sequence form* [37], whose size is linear in the size of the game tree.

The sequence form decomposes strategies into sequences of actions and their realization probabilities. A *sequence* for player i, associated with a node x of the game, is the subset of A specifying player i's actions on the path from the root to x. We denote the set of sequences of player i by Q_i. A sequence is said *terminal* if it leads to a terminal node for at least a set of sequences of the other players. The set of terminal sequences of player i is denoted by $\overline{Q}_i$. Moreover, we denote by $q_\emptyset$ the fictitious sequence leading to the root node and, for each action $a \in A$ and sequence $q \in Q_i$, we denote by $qa \in Q_i$ the *extended* sequence obtained by appending action a to q. Let $Q = \bigtimes_{i \in N} Q_i$. When considering a tuple $q = (q_1, \ldots, q_n) \in Q$, we denote by q_i the i-th component of q.

A sequence-form strategy, said *realization plan*, is a function $r_i : Q_i \to \mathbb{R}$ associating each sequence $q \in Q_i$ with its probability of being played. A well-defined sequence-form strategy is such that $r_i(q_\emptyset) = 1$ for each $i \in N$ and, for each h and sequence q leading to h, $-r_i(q) + \sum_{a \in \rho(h)} r_i(qa) = 0$ and $r_i(q) \geq 0$. These constraints are linear in the number of sequences and can be compactly written as $F_i\, r_i = f_i$, where F_i is an $|H_i| \times |Q_i|$ matrix and $f_i^\top = (1, 0, \ldots, 0)$ is a vector of dimension $|H_i|$. The utility function of player i is represented by a sparse n-dimensional matrix defined only for profiles of terminal sequences leading to a leaf node. With a slight abuse of notation, we denote it by $U_i \in \mathbb{R}^{|Q_1| \times \cdots \times |Q_n|}$.

[1]This rules out the possibility of employing an EFCE.

Given a tuple $r_{-i} = (r_1, \dots, r_{i-1}, r_{i+1}, \dots, r_n)$, $\mathrm{BR}(r_{-i})$ denotes the *best response* of player i against a strategy profile r_{-i}. We say that $\hat{r}_i \in \mathrm{BR}(r_{-i})$ if the following holds:

$$\sum_{q \in Q} \hat{r}_i(q_i) \prod_{j \in N \setminus \{i\}} r_j(q_j) U_i(q) = \max_{r_i} \sum_{q \in Q} r_i(q_i) \prod_{j \in N \setminus \{i\}} r_j(q_j) U_i(q),$$

where r_i is constrained to be a valid realization plan.

2.3 Correlation in Normal-Form Games

Let $p_{-i} = (p_1, \dots, p_{i-1}, p_{i+i}, \dots, p_n) \in \times_{j \in N \setminus \{i\}} P_j$. The classical notion of CE [3] for normal-form games is:

Definition 2.1. $\sigma^* \in \Sigma$ is a correlated equilibrium of the normal form game (N, P, U') if, for every $i \in N$ and $p_i, p_i' \in P_i$, the following holds:

$$\sum_{p_{-i} \in P_{-i}} \sigma^*(p_i, p_{-i}) \left(U_i'(p_i, p_{-i}) - U_i'(p_i', p_{-i}) \right) \geq 0.$$

A CE can be interpreted in terms of a mediator who, *ex ante* the play, draws $(p_1, \dots, p_n)$ according to the publicly known σ^* and privately communicates each *recommendation* p_i to the corresponding player.

Another possibility is enforcing protection against deviations of players which are independent from the sampled outcome. This can be done though the notion of coarse correlated equilibrium [30]:

Definition 2.2. $\sigma^* \in \Sigma$ is a coarse correlated equilibrium of a normal-form game (N, P, U') if, for every $i \in N$ and $p_i' \in P_i$, the following holds:

$$\sum_{p_i \in P_i} \sum_{p_{-i} \in P_{-i}} \sigma^*(p_i, p_{-i}) \left(U_i'(p_i, p_{-i}) - U_i'(p_i', p_{-i}) \right) \geq 0.$$

CCEs differ from CEs in that a CCE only requires that following the suggested action be a best response in expectation before the recommended action is actually revealed. Moreover, we recall that every CE is also a CCE while the converse is, in general, not true.

2.4 Correlation in Extensive-Form Games

We review the main notions of correlation for general extensive-form games. In this general setting, it is customary to consider *ex ante* CEs, i.e., correlated equilibria in which an action profile is sampled before the game is played. In this paper, we focus on the following solution concepts:

Definition 2.3. A normal-form correlated equilibrium (respectively, normal-form coarse correlated equilibrium) of an extensive-form game Γ is a correlated equilibrium (respectively, coarse correlated equilibrium) of the reduced normal-form game equivalent to Γ.

In these two solution concepts, the entire vector of recommendations specifying one action per information set is revealed to the players before the game starts. Thus, once the recommendation is received each player commits to playing a pure strategy.

Informally, an AFCE [18] is a CE of the agent-form game equivalent to the given extensive-form game. In the agent form of the game, moves are chosen by a different agent per information set of the player. In an EFCE [38], each recommendation is assumed to be in a *sealed envelope* and is revealed only when the player reaches the relevant information set (i.e., the information set where she can make that move). The main difference between EFCE and NFCE/NFCCE is that the former requires recommendations to be delivered during the game execution, thus being more demanding in terms of communication requirements. It is crucial to notice that the size of the signal that has to be sampled is the same, and it is polynomial (one action per information set).

Letting $S_\circ$ be the set of equilibria of type $\circ$ of a given game, we have: $S_{NFCE} \subseteq S_{EFCE} \subseteq S_{NFCCE} \subseteq S_{AFCE}$. See von Stengel and Forges [38] for further details.

3 COMPARISON BETWEEN NFCE AND NFCCE

In the next sections, we study the problems of computing an NFCCE maximizing the social welfare (i.e., the cumulative utility of the players). We refer to it as NFCCE-SW. Moreover, we denote by NFCE-SW the problem of computing a social welfare maximizing NFCE. The generalization of our results to the case in which one searches for an equilibrium maximizing a linear combination of the players' utility, omitted here for reasons of space, is straightforward.

We believe that the motivation for studying the computation of NFCCEs is twofold. First, it is known that finding a socially optimal NFCE is NP-hard even for two-player extensive-form games without chance moves [38]. Second, as already mentioned, an NFCCE may not be an NFCE. In particular, an optimal NFCCE may lead to a social welfare arbitrarily larger than the social welfare provided by an NFCE which is optimal for the same game. This is shown via the following example:

Example 3.1. Consider the extensive-form game in Figure 1 as an example. The game is parametric in $k > 1$, and it is played between two players. Each player has a unique information set (I_1 for Player 1, and I_2 for Player 2).

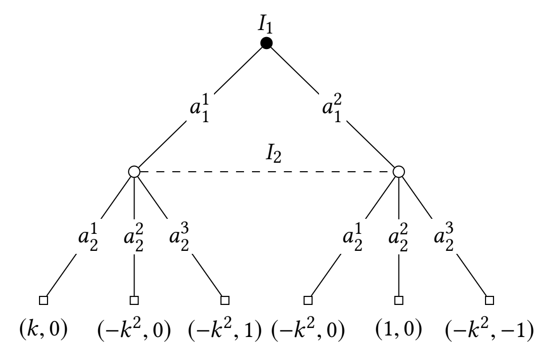

Figure 1: Example illustrating the difference between CE and CCE.

The joint strategy profile assigning probability 1/2 to (a_1^1, a_2^1) and (a_1^2, a_2^2) is the NFCCE maximizing the social welfare of the players, which is $(k+1)/2$. The unique optimal NFCE 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 NFCCE allows the players to reach a social welfare which is arbitrarily larger than the social welfare reached through the optimal NFCE.

4 COMPLEXITY OF COMPUTING AN OPTIMAL NFCCE

We show that there exists a polynomial-time algorithm for solving the NFCCE-SW problem with two players. First, we provide a compact formulation for the problem. Then, we describe a polynomial-time algorithm for solving it.

4.1 Problem Formulation

Given an extensive-form game Γ, a direct application of Definition 2.3 yields a Linear Programming problem (LP) with an exponential number of variables and an exponential number of constraints. We provide the following result:

LEMMA 4.1. *The NFCCE-SW problem for an extensive-form game Γ can be formulated as an LP with an exponential number of variables but only a polynomial number of constraints.*

To prove the lemma, we provide a hybrid representation which exploits the tree structure of the problem combining both the normal form and the sequence form.

First, we say that 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 the terminal nodes of the game tree. Let $r_{p_i} \in \{0,1\}^{|Q_i|}$ be a $|Q_i|$-dimensional column vector representing the pure realization plan for player $i \in N$ that is *realization equivalent* to $p_i \in P_i$. We recall that every plan of the reduced normal form is realization equivalent to exactly one pure realization plan, see von Stengel [37]. We say that the normal-form plan $p_i \in P_i$ is a best response against profile r_{-i} if $r_{p_i} \in \text{BR}(r_{-i})$. In the following, and when not differently specified, U_i denotes the sequence-form utility matrix of player i.

According to Definition 2.2, the constraints describing an NFCCE for Player 1 can be written as follows (for Player 2, the constraints are analogous):

$$\sum_{p_1 \in P_1} \sum_{p_2 \in P_2} \sigma(p_1, p_2)\, U_1'(p_1, p_2) - \sum_{p_1 \in P_1} \sum_{p_2 \in P_2} \sigma(p_1, p_2)\, U_1'(p_1', p_2) \geq 0 \qquad \forall p_1' \in P_1.$$

The first term is the expected utility of Player 1 at the equilibrium. Let v_1 be the $|H_1|$-dimensional vector of variables of the dual of the best-response problem in sequence form. The second term is the expected utility obtained by Player 1 when she deviates to $p_1' \in P_1$, assuming Player 2 follows the recommendation. Therefore, it is enough to enforce the constraint corresponding to plan p_1' constituting the best-response of Player 1 against the fixed behavior of Player 2. To compactly represent such constraint, we decompose the best-response problem locally at each information set.

By definition of sequence form, $f_1^\top v_1$ is equal to the first component of v_1, whose value corresponds to the utility of Player 1 at the equilibrium. Then:

$$\begin{cases} \sum_{p_1 \in P_1} \sum_{p_2 \in P_2} \sigma(p_1, p_2)\, U_1'(p_1, p_2) = f_1^\top v_1 \\ f_1^\top v_1 - \sum_{p_1 \in P_1} \sum_{p_2 \in P_2} \sigma(p_1, p_2)\, U_1'(p_1', p_2) \geq 0 \quad \forall p_1' \in P_1. \end{cases}$$

The double summation in the two inequalities above can be written as:

$$\sum_{p_2 \in P_2} \left(\sum_{p_1 \in P_1} \sigma(p_1, p_2) \right) U_1'(p_1', p_2).$$

Letting $\bar{\sigma}_2(p_2) = \sum_{p_1 \in P_1} \sigma(p_1, p_2)$, $\bar{\sigma}_2 \in \Delta^{|P_2|}$ can be interpreted as the prior probability that plan p_2 be played by Player 2. The mixed strategy $\bar{\sigma}_2$ can be written as the following realization-equivalent sequence-form strategy: $\bar{r}_2 = \sum_{p_2 \in P_2} \bar{\sigma}(p_2) r_{p_2}$, which is a valid realization plan due to convexity. Now, we only need to show that $f_1^\top v_1$ is not strictly smaller than the value of the best response of Player 1 given the strategy $\bar{r}_2$ of Player 2. Formally, given $r_1' \in \text{BR}(\bar{r}_2)$, the constraint reads $f_1^\top v_1 \geq r_1' U_i \bar{r}_2$. By exploiting the dual of the best-response problem in sequence form, this is equivalent to showing $F_1^\top v_1 - U_1 \bar{r}_2 \geq 0$. Thus, expanding $\bar{r}_2$ and deriving the equilibrium constraints for Player 2 we obtain the following mathematical program:

$$\max_{\sigma, v_1, v_2} \sum_{(p_1,p_2) \in P_1 \times P_2} \sigma(p_1, p_2)\, r_{p_1}^\top (U_1 + U_2)\, r_{p_2} \tag{1}$$

$$\sum_{(p_1,p_2) \in P_1 \times P_2} \sigma(p_1, p_2)\, r_{p_1}^\top U_i\, r_{p_2} = f_i^\top v_i \quad \forall i \in N \tag{2}$$

$$F_1^\top v_1 - U_1 \left(\sum_{p_2 \in P_2} \left(\sum_{p_1 \in P_1} \sigma(p_1, p_2) \right) r_{p_2} \right) \geq 0 \tag{3}$$

$$F_2^\top v_2 - U_2^\top \left(\sum_{p_1 \in P_1} \left(\sum_{p_2 \in P_2} \sigma(p_1, p_2) \right) r_{p_1} \right) \geq 0 \tag{4}$$

$$\sum_{(p_1,p_2) \in P_1 \times P_2} \sigma(p_1, p_2) = 1 \tag{5}$$

$$\sigma(p_1, p_2) \geq 0 \qquad \forall (p_1, p_2) \in P_1 \times P_2. \tag{6}$$

We make the following observations on the above LP:

- Variables $\sigma \in \Delta^{|P_1| \times |P_2|}$ constitute the correlated strategy for Players 1 and 2.
- This formulation constitutes a proof of Lemma 4.1 as it employs a polynomial number of constraints (namely, $|Q_1| + |Q_2| + 3$) and an exponential number of variables (i.e., one for each pair of plans in $P_1 \times P_2$).

4.2 Efficient Algorithm

The following lemma will be employed to prove our central result. It shows that a player can reason in a *best-response fashion* to minimize the utility of the other player weighted by some arbitrary coefficients, while also guaranteeing the reachability of a given terminal node.

LEMMA 4.2. *Given a generic two-player extensive-form game Γ, an outcome $\ell \in L$, and a vector $\zeta \in \mathbb{R}^{|Q_1|}$, the problem of finding $p_2 \in P_2$ under the constraints that*

- *there exists some $p_1 \in P_1$ s.t. (p_1, p_2) leads to outcome ℓ and*
- *$\zeta^\top U_1 r_{p_2}$ is minimized*

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

Proof. Let us focus on the case in which we look for $p_2 \in P_2$. First, define $\bar{U}_1$ such that

$$\bar{U}_1(q_1, q_2) := \zeta(q_1) U_1(q_1, q_2) \quad \forall (q_1, q_2) \in Q_1 \times Q_2.$$

Then, let $\bar{\Gamma}$ be the extensive-form game obtained from Γ by substituting Player 1's utility function with $\bar{U}_1$. Given $\bar{\Gamma}$, denote by (q_1^ℓ, q_2^ℓ) the pair of sequences identifying ℓ, and by H_i^ℓ the set of information sets of player i encountered in sequence q_i^ℓ. Algorithm 1 returns the set of actions (A'_i) forming a plan of the normal-form game (not reduced) equivalent to $\bar{\Gamma}$.

Algorithm 1 `Constrained-plan-search`

function C-PLAN-SEARCH($x, \bar{\Gamma}, i, q_i^\ell, H_i^\ell, A'_i$) ▷ i is the player for which we want to find a plan, A'_i is the temporary set (initially empty) of actions of i selected
  $v \leftarrow K$ ▷ K is a sufficiently large constant
  $a' \leftarrow$null
  if x is terminal **then**
    return $(\bar{U}_{-i}(x), A'_i)$
  else
    if $x \in V_{-i}$ **then**
      for $y \in x$.child **do**
        v+ = C-PLAN-SEARCH($y, \bar{\Gamma}, i, q_i^\ell, H_i^\ell, A'_p$).val
      return (v, A'_i)
    else
      if $\exists h \in H_i^\ell : x \in h$ **then**
        $a' \leftarrow$ action specified by q_i^ℓ
        $y_{a'} \leftarrow$ child of x reached through a'
        v = C-PLAN-SEARCH($y_{a'}, \Gamma', i, q_i^\ell, H_i^\ell, A'_i$).val
      else
        for $y \in x$.child **do**
          $temp \leftarrow$ C-PLAN-SEARCH($y, \bar{\Gamma}, i, q_i^\ell, H_i^\ell, A'_i$)
          if $temp$.val $< v$ **then**
            $v \leftarrow temp$.val
            $a' \leftarrow a \in \rho(x) : \chi(x, a) = y$
      return$(v, A'_i \cup \{a'\})$

To retrieve A'_i, Algorithm 1 performs a depth-first traversal of the tree while keeping track of the value to be minimized at each decision node (v) and selecting actions while moving backwards. Then, p_2 can be computed by traversing the tree from the root, and selecting actions according to those specified in A'_2. □

Let us focus on the dual $\mathcal{D}$ of LP (1)–(6):

Lemma 4.3. *$\mathcal{D}$ admits a polynomial-time separation oracle.*

Proof. Let $\alpha_i \in \mathbb{R}$, for all $i \in N$, be the dual variables of constraints (2), $\beta_1 \in \mathbb{R}^{|Q_1|}$ the dual variables of constraints (3), $\beta_2 \in \mathbb{R}^{|Q_2|}$ the dual variables of constraints (4), and $\gamma \in \mathbb{R}$ the dual variable of constraint (5). With $n = 2$, $\mathcal{D}$ is an LP with a number of variables ($|Q_1| + |Q_2| + 3$) polynomial in the size of the tree and an exponential ($|P_1 \times P_2| + |H_1| + |H_2|$) number of constraints. We show that, given a vector $\bar{z} = (\bar{\alpha}_1, \bar{\alpha}_2, \bar{\beta}_1, \bar{\beta}_2, \bar{\gamma})$, the problem of either finding a hyperplane separating $\bar{z}$ from the set of feasible solutions to $\mathcal{D}$ or proving that no such hyperplane exists can be solved in polynomial time. Since the number of dual constraints corresponding to the primal variables v_i is linear, these constraints can be checked efficiently for violation. We are left with the problem of determining whether any of the following constraints, defined for all $(p_1, p_2) \in P_1 \times P_2$, is violated:

$$r_{p_1}^\top U_1 r_{p_2} \bar{\alpha}_1 + r_{p_1}^\top U_2 r_{p_2} \bar{\alpha}_2 + \bar{\beta}_1^\top U_1 r_{p_2} + r_{p_1}^\top U_2 \bar{\beta}_2 + \bar{\gamma} \geq r_{p_1}^\top (U_1 + U_2) r_{p_2}.$$

Let us consider the *separation problem* of finding an inequality of $\mathcal{D}$ which is maximally violated at $\bar{z}$. The problem reads:

$$\min_{(p_1,p_2) \in P_1 \times P_2} \left\{ r_{p_1}^\top ((\bar{\alpha}_1 - 1) U_1 + (\bar{\alpha}_2 - 1) U_2) r_{p_2} + \bar{\beta}_1^\top U_1 r_{p_2} + r_{p_1}^\top U_2 \bar{\beta}_2 \right\}.$$

A pair p_1, p_2 yielding a violated inequality exists if and only if the separation problem admits an optimal solution of value $< -\bar{\gamma}$.

One such pair (if any) can be found in polynomial time by enumerating over the (polynomially many) possible outcomes $\ell \in L$ of the game. For each of them, we look for a pair (p_1^ℓ, p_2^ℓ) minimizing the objective function of the separation problem, halting as soon as a pair (p'_1, p'_2) 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. First, notice that $r_{p_1}^\top((\bar{\alpha}_1 - 1)U_1 + (\bar{\alpha}_2 - 1)U_2) r_{p_2}$ is constant for the family of pairs identifying $\ell \in L$. Therefore, we can consider an individual subproblem for each player (i.e., we can find p_1^ℓ and p_2^ℓ independently). Hence, for each outcome ℓ and for each player i the corresponding p_i^ℓ can be found in polynomial time due to Lemma 4.2. □

The following theorem shows that, in certain cases, the NFCCE-SW problem can be solved efficiently:

Theorem 4.4. *Given an extensive-form game Γ with $n = 2$ players and without chance moves, an NFCCE maximizing the social welfare can be computed in time polynomial in the size of the game tree.*

Proof. Lemma 4.3 shows that there exists a polynomial-time separation oracle for $\mathcal{D}$. Then, $\mathcal{D}$ can be solved in polynomial time via the ellipsoid method due to the equivalence between optimization and separation [21, 28]. As the method solves, in polynomial time, a primal-dual system encompassing not just $\mathcal{D}$ but also its primal problem NFCCE-SW, it also produces, simultaneously, an optimal solution to the latter. □

4.3 Negative Result

The approach that we presented here cannot be extended to games with two players and the chance player as, upon introducing the latter, the problem transitions from polynomially solvabile to NP-hard. Interestingly, other problems in which this transition takes place are, for example, the problem of computing a socially optimal EFCE [38] and the problem of deciding if a two-player zero-sum extensive-form game with perfect recall admits a pure strategy equilibrium [8, 22]. In our setting, the following holds.

Theorem 4.5. *Computing an NFCCE maximizing the social welfare is NP-hard even in extensive-form games with two players, chance moves, and binary outcomes.*

Proof. The construction introduced in [38, Theorem 1.3] can be employed. We sketch its basic structure and apply it to the problem of solving the NFCCE-SW problem. The reduction is from SAT, whose generic instance is a Boolean formula ϕ in conjunctive normal form with η clauses and ν variables. Given ϕ, we build an auxiliary game Γ_ϕ with size proportional to that of the boolean formula. The root of Γ_ϕ is a chance node, with one action for each clause of ϕ. Then, on the *second level* of the tree, there are η decision nodes of Player 1, each one belonging to a singleton information set and identifying a single clause of ϕ. At each of this decision nodes, Player 1 has an action for each literal (negated or non-negated variable) 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 ν information sets, one for each variable appearing in ϕ. Each of these information sets has two available actions, corresponding to the truth assignment of the variable, and leading to a terminal node. Both players have the same payoffs, which is 0 if the literal (chosen by player 1 from the clause) is false and 1 if it is true. Γ_ϕ admits a pure strategy guaranteeing social welfare 2 if and only if ϕ is satisfiable. Otherwise, the maximum expected social welfare cannot be more than $2(1 - 1/\eta)$. Notice that a pure strategy maximizing the social welfare is also an NFCCE, since no *ex-ante* deviation would result in an increase in the player's utility, being it already maximal. Then, finding a solution to NFCCE-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. This concludes the proof. □

Notice that, when considering the separation problem of $\mathcal{D}$, working with chance is hard because the first term of the objective function of the separation problem is no longer constant when the outcome is fixed. In the case with $n > 2$ and no chance moves, one would have to determine the joint best response of two players at a time (to maximize the terms of the objective function of the separation problem following the first one), which is NP-hard [38].

5 A PRACTICAL ALGORITHM

Due to being based on the ellipsoid method (which, while being a powerful theoretical tool, is well-known to be inefficient in practice), the algorithm that we used in the proof of Theorem 4.4 is not appealing from a practical perspective. We propose, here, a computationally more efficient method based on the simplex method (we refer the unfamiliar reader to Bertsimas and Tsitsiklis [7] for a comprehensive introduction) to compute optimal NFCCEs via a *column generation* technique. The focus on two-player games is motivated by the negative result in the previous section.

Let x be a vector containing the variables of LP (1)–(6):

$$x^\top = (\underbrace{\sigma(p'_1, p'_2), \dots, \sigma(p''_1, p'_2), \dots}_{|P_1 \times P_2|}, v_1^\top, v_2^\top, s_1^\top, s_2^\top),$$

where, for each $i = 1, 2$, v_i is defined as in the proof of Lemma 4.1 and s_i is a $|Q_i|$-dimensional column vector of slack variables. The cost vector c associated with the variables is:

$$c^\top = (\underbrace{[U'_1(p'_1, p'_2) + U'_2(p'_1, p'_2)]}_{\sigma(p'_1, p'_2)}, \dots, \underbrace{0, \dots, 0}_{|H_1|+|H_2|+|Q_1|+|Q_2|}),$$

where U'_i is the utility matrix of the reduced normal-form game. We compactly rewrite the constraints of LP (1)–(6) in standard form as $M x = b$, where $b^\top = (1, 0, \dots, 0)$ is a vector of dimension $(|Q_1| + |Q_2| + 3)$. We denote the j-th column of M by $M_{(\cdot, j)}$.

The algorithm works in two phases, determining, first, a basic feasible solution and, then, iteratively improving it until an optimal one is found. The crucial component of the algorithm is an oracle for solving, given a basic feasible solution to LP (1)–(6), the problem (we refer to it as LRC) of finding a variable with the largest reduced cost—the so-called (primal) *pricing* problem. Notice that the tractability of such problem is already implied by Theorem 4.4 as the problem is equivalent to that of finding a maximally violated constraint in the dual $\mathcal{D}$. Hence:

Corollary 5.1. *LRC can be solved in polynomial-time.*

Letting c_j be the cost associated with the j-th component of x and letting c_B be the vector of costs of the basic variables, the j-th reduced cost is:

$$\bar{c}_j = c_j - c_B^\top B^{-1} M_{(\cdot, j)}, \tag{7}$$

where $B = [M_{(\cdot, j')}, M_{(\cdot, j'')}, \dots]$ and $j', j'', \dots$ are the indices of the basic variables. We rely on the following polynomial-time oracle, P-LRC, described in Algorithm 2 (another oracle is presented in the next section).

First, notice that, given a basic feasible solution, $c_B^\top B^{-1}$ is equal to a vector (call it ζ) of dimension $(|Q_1| + |Q_2| + 3)$, computable in polynomial time (Line 4). By employing the same notation as the one adopted for the dual variables in the proof of Lemma 4.3, let $\zeta^\top = (\bar{\beta}_1, \bar{\beta}_2, \bar{\alpha}_1, \bar{\alpha}_2, \bar{\gamma})$, where $\bar{\beta}_i$ is the vector of dual variables of constraints (3) and (4), $\bar{\alpha}_i$ are the dual variables of constraints (2), and $\bar{\gamma}$ is that of constraint (5).

Algorithm 2 P-LRC

```
1:  function P-LRC(Γ, M, c, B)
2:      J ← ∅
3:      ∀j, c̄_j ← ∞
4:      ζ ← c_B^T B^{-1}
5:      for j ∈ {|P_1 × P_2| + 1, . . . , |c|} do
6:          c̄_j ← c_j − ζ M_(·,j)
7:          J ← J ∪ {j}
8:      for ℓ ∈ L do
9:          p̂_i ← C-PLAN-SEARCH(ℓ, β̄_i), ∀i ∈ N
10:         ĵ ← index of σ(p̂_1, p̂_2) in c
11:         c̄_ĵ ← c_ĵ − ζ M_(·,ĵ)
12:         J ← J ∪ {ĵ}
13:     j* = arg max_{j∈J} c̄_j
14:     return j*
```

The reduced costs of the variables v_i and s_i can be computed directly by definition since their number is polynomial in the size of the tree (Lines 5 to 7). We are left with the problem of evaluating the reduced costs of the $\sigma(\cdot,\cdot)$ variables. P-LRC enumerates the outcomes of the game (Line 8). Since all the pairs of plans identifying ℓ have the same c_j, the problem of minimizing $\zeta^\top M_{(\cdot,j)}$ amounts to finding a pair (p_1, p_2) minimizing $(\bar{\beta}_1 U_1 r_{p_2} + \bar{\beta}_2 U_2^\top r_{p_1})$. The problem can be split into a subproblem per player, and solved through Algorithm 1, which we presented in the proof of Lemma 4.2 (Line 9, where we simplified the signature of C-PLAN-SEARCH for ease of notation). By applying this procedure for each of the outcomes and selecting, among the resulting pairs, one with the largest reduced cost (Line 13), we are able to determine the new variable entering the basis in polynomial time.

The two phases of the overall algorithm are the following ones, and both adopt P-LRC:

Phase 1: finding a feasible point. A basic feasible solution to NFCCE-SW is determined by solving an auxiliary problem with artificial variables, where a new variable is introduced for each equality constraint, and their sum is minimized in the objective function. If some artificial variable with index $\bar{j}$ is found in the optimal basis of the auxiliary problem, we can find, in polynomial-time, a variable j of the original problem to replace it by either maximizing or minimizing $e_{\bar{j}} B^{-1} M_{(\cdot,j)}$, where $e_{\bar{j}}$ is a vector of zeros with suitable dimension and equal to 1 in position j (the problem can be solved with Algorithm 1).

Phase 2: finding an optimal solution. Starting from a basic feasible solution, the algorithm iteratively improves it until an optimal solution is found. We remark that, while, if we were to solve the problem with a standard implementation of the simplex method, we would have to compute the reduced cost of all the nonbasic variables to find one to enter the basis (which would require exponential time in the size of the game), by employing P-LRC the next variable to enter the basis can be found in polynomial time. This follows from the same reasoning that led to Corollary 5.1.

We remark that, while the two phases require polynomial time, the bottleneck of the approach is that, at each iteration, P-LRC has to traverse the game tree twice for each $\ell \in L$. This, as we will better assess further in the paper from a computational perspective, can be very time consuming in practice. To circumvent this issue, we present a second oracle based on mixed-integer linear programming (see the experimental evaluation for a comparison between the two approaches).

6 GENERAL MIXED-INTEGER ORACLE

In this section, 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 nee for explicitly enumerating the terminal nodes of the game, and, furthermore, it can be extended to games with chance. We provide, here, a description of the oracle for the case of a two-player game with and without chance moves. MI-LRC can also be extended to games with $n > 2$, but we omit the description of this setting due to space constraints.

The crucial difference between MI-LRC and P-LRC is in the way they handle the inspection of the reduced costs associated with the $\sigma(\cdot,\cdot)$ variables. In MI-LRC, lines 8–12 of Algorithm 2 are substituted with an MILP.

6.1 Two-Player Games

Let us first focus on the case of a two-player game without chance moves. Let R_i be a $|Q_i| \times |L|$ matrix such that:

$$R_i(q_i, \ell) = \begin{cases} 1 & \text{if } q_i \text{ is on the path from the root to } \ell \\ 0 & \text{otherwise.} \end{cases}$$

Let also z be an $|L|$-dimensional vector of binary variables. MI-LRC solves the following problem:

$$\max_{\substack{z \in \{0,1\}^{|L|} \\ r_i \in \mathbb{R}_+^n}} \left((1-\bar{\alpha}_1) r_1^\top - \bar{\beta}_1^\top\right) U_1 r_2 + r_1^\top U_2 \left((1-\bar{\alpha}_2) r_2 - \bar{\beta}_2\right) \tag{8}$$

$$F_i r_i = f_i \qquad \forall i \in N \tag{9}$$

$$r_i \geq R_i z \qquad \forall i \in N \tag{10}$$

$$\sum_{\ell \in L} z(\ell) = 1. \tag{11}$$

The objective function (8) follows from the definition of the reduced costs (we are looking for a variable whose dual constraint is maximally violated). Constraints (10) force the realization plans to select with probability 1 the sequences on the path to the selected outcome ℓ. Notice that, 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 formulation proposed in [29].

Notice that an optimal realization plan r_i^* solving MI-LRC to optimality may not be *pure* (i.e., there may exist some $q \in Q_i$ s.t. $r_i^*(q) \in (0,1)$). Nevertheless, there always exists a pair of pure realization plans leading to the same terminal node and granting the same value $\bar{\beta}_1^\top U_1 r_2^* + r_1^{*\top} U_2 \bar{\beta}_2$. Once a pair of pure realization plans has been determined, the reduced cost associated with it has to be computed according to equation (7) and compared to the reduced costs of the remaining variables (Line 13 of Algorithm 2).[2]

6.2 Two-Player Games with Nature

We denote by $(q_1^\ell, q_2^\ell, q_c^\ell)$ the unique tuple of the sequences leading to ℓ, where q_c^ℓ is a sequence of the chance player. The crucial point is that, given $\ell \in L$, there may exist some $\ell' \in L \setminus \{\ell\}$, reachable through $(q_1^\ell, q_2^\ell, q_c^{\ell'})$, satisfying $q_c^{\ell'} \neq q_c^\ell$. MI-LRC can be adapted to this scenario as follows. First, for each $i \in N$ we compute the utility matrices U_{i,π_c} (with dimension $|Q_1| \times |Q_2|$) obtained by marginalizing each U_i with respect to π_c. Denoting by r_c the realization plan defined over the sequences of the chance player which are 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) is then modified by substituting each U_i with U_{i,π_c}. Moreover, upon denoting by R_c the $|\bar{Q}_c| \times |L|$ matrix defined analogously to R_i, it suffices to substitute each of constraints (11), one per $\bar{q} \in \bar{Q}_c$, with the constraint $R_{c,(q_c,\cdot)} z = 1$, where $R_{c,(q_c,\cdot)}$ denotes row q_c of R_c. This way, MI-LRC can be extended to the

[2] It is enough to traverse the tree depth-first, and select sequences, among those played with strictly positive probability in r_i^*, following the same reasoning of Algorithm 1.

Table 1: Comparison of the performance with the two different oracles.

	Game size				P-LRC				MI-LRC			
Game	$\|Q_1\|$	$\|Q_2\|$	$\|H_1\|$	$\|H_2\|$	Phase1 (steps)	Phase2 (steps)	Time (sec)	Solved (in 12h)	Phase1 (steps)	Phase2 (steps)	Time (sec)	Solved (in 12h)
R5-2	20	26	10	14	4.6	6.8	0.3	20	4.7	2.25	0.02	20
R5-3	126	102	43	35 5.5	5.5	8.9	9.2	20	5.5	3.15	0.32	20
R5-4	400	404	100	102	8.2	12.1	439.8	20	7.5	4.8	8.1	20
R10-2	664	680	333	340	5.5	11.6	1121.8	20	5.8	6.2	23.4	20
R12-2	2649	2697	1325	1349	5	7	41421.3	1	6.2	6.3	391.7	20

Table 2: Performance of MI-LRC on large two-player games and games with Nature.

Game	$\|Q_1\|$	$\|Q_2\|$	$\|H_1\|$	$\|H_2\|$	Phase1	Phase2	Time
R13-2	5364	5316	2682	2659	6.1	8.9	3368.2
G3R	58	58	47	47	7	1	0.1
G3S	334	334	274	274	64	1	100.6
G3D	334	334	274	274	6	1	1.7

more demanding setting of games with two-players and chance moves.

7 EXPERIMENTAL EVALUATION

We compare the performance of our column generation method with the two different oracles P-LRC and MI-LRC on 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. We also experiment on instances of two-player games with chance. We employ *Goofspiel* game instances [33, 35], 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 high 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 our 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 in which the order of the prize cards is known.

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.

We remark that the use of an LP defined directly on the normal form of a game (Definition 2.2) is impractical for every instance of our experimental 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 1 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 2.

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 NFCCE even on instances with more than 5000 cumulative sequences and 2500 information sets in less than one hour.

8 DISCUSSION

In this paper, we have studied *ex ante* correlated equilibria in extensive-form games with low communication requirements. First, we have showed that an optimal NFCCE can be computed in polynomial time in two-player games without chance moves. This complexity result cannot be extended to more general settings (i.e., games with Nature, or games with three or more players), as the problem is shown to become NP-hard. Moreover, we have devised a scalable column generation method based on the simplex algorithm which allows for computing optimal NFCCE efficiently in practice. We have also devised two pricing oracles for the problem of finding a column with the largest reduced cost. The first one is guaranteed to have polynomial running time, while the second is an MILP with less appealing theoretical properties but a largely better performance in practice. We have experimentally evaluated our column generation technique to demonstrate its scalability, and assessed its performance when employing different pricing oracles. Our results show that correlation in sequential games is possible in practice, even when requiring a minimal communication effort.

In the future, it would be interesting to further improve the scalability of our methods 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 [1, 2, 14], and the introduction of dominance relationships among the columns to reduce their generation to a subset which is more effective in terms of bound improvement. Moreover, our techniques for *ex ante* correlated equilibria could also be employed in the *Bayesian persuasion* framework [15, 27]. Investigating whether it is possible to adapt our approach to compute optimal signaling schemes in this setting would be an interesting avenue for future research.

REFERENCES

[1] E. Amaldi, S. Coniglio, and S. Gualandi. 2010. Improving Cutting Plane Generation with 0-1 Inequalities by Bi-criteria Separation. In *Experimental Algorithms*, Paola Festa (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 266–275.

[2] E. Amaldi, S. Coniglio, and S. Gualandi. 2014. Coordinated cutting plane generation via multi-objective separation. *Mathematical Programming* 143, 1 (2014), 87–110.

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

[4] S. Barman and K. Ligett. 2015. Finding Any Nontrivial Coarse Correlated Equilibrium Is Hard. In *Proceedings of the ACM Conference on Economics and Computation (EC)*. 815–816.

[5] B. Basilico, A. Celli, G. De Nittis, and N. Gatti. 2017. Coordinating Multiple Defensive Resources in Patrolling Games with Alarm Systems. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*. 678–686.

[6] N. Basilico, A. Celli, G. De Nittis, and N. Gatti. 2017. Team-maxmin equilibrium: efficiency bounds and algorithms. In *AAAI Conference on Artificial Intelligence (AAAI)*.

[7] D. Bertsimas and J. N. Tsitsiklis. 1997. *Introduction to linear optimization*. Vol. 6. Athena Scientific Belmont.

[8] J. R. S. Blair, D. Mutchler, and M. Lent. 1996. Perfect recall and pruning in games with imperfect information. *Computational Intelligence* 12, 1 (1996), 131–154.

[9] N. Brown and T. Sandholm. 2017. Safe and nested subgame solving for imperfect-information games. In *Advances in Neural Information Processing Systems (NeurIPS)*. 689–699.

[10] N. Brown and T. Sandholm. 2017. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science* (2017), eaao1733.

[11] A. Celli and N. Gatti. 2018. Computational Results for Extensive-Form Adversarial Team Games. In *AAAI Conference on Artificial Intelligence (AAAI)*.

[12] S. Ceppi, N. Gatti, G. Patrini, and M. Rocco. 2010. Local search techniques for computing equilibria in two-player general-sum strategic-form games. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. 1469–1470.

[13] N. Cesa-Bianchi and G. Lugosi. 2006. *Prediction, learning, and games*. Cambridge university press.

[14] S. Coniglio and M. Tieves. 2015. On the Generation of Cutting Planes which Maximize the Bound Improvement. In *Experimental Algorithms*, Evripidis Bampis (Ed.). Springer International Publishing, 97–109.

[15] S. Dughmi and H. Xu. 2016. Algorithmic Bayesian persuasion. In *ACM STOC*. ACM, 412–425.

[16] G. Farina, A. Celli, N. Gatti, and T. Sandholm. 2018. Ex ante coordination and collusion in zero-sum multi-player extensive-form games. In *Advances in Neural Information Processing Systems (NeurIPS)*.

[17] F. Forges. 1986. An approach to communication equilibria. *Econometrica* (1986), 1375–1385.

[18] F. Forges. 1993. Five legitimate definitions of correlated equilibrium in games with incomplete information. *Theory and Decision* 35, 3 (01 Nov 1993), 277–310.

[19] F. Forges. 2006. Correlated Equilibrium in Games with Incomplete Information Revisited. *Theory and Decision* 61, 4 (01 Dec 2006), 329–344.

[20] N. Gatti, G. Patrini, M. Rocco, and T. Sandholm. 2012. Combining local search techniques and path following for bimatrix games. *arXiv preprint arXiv:1210.4858* (2012).

[21] M. Grötschel, L. Lovász, and A. Schrijver. 1981. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica* 1, 2 (01 Jun 1981), 169–197.

[22] K. A. Hansen, P. B. Miltersen, and T. B. Sørensen. 2007. Finding equilibria in games of no chance. In *International Computing and Combinatorics Conference*. Springer, 274–284.

[23] S. Hart and A. Mas-Colell. 2000. A simple adaptive procedure leading to correlated equilibrium. *Econometrica* 68, 5 (2000), 1127–1150.

[24] J. Hartline, V. Syrgkanis, and E. Tardos. 2015. No-regret learning in Bayesian games. In *Advances in Neural Information Processing Systems (NeurIPS)*. 3061–3069.

[25] W. Huang and B. von Stengel. 2008. Computing an extensive-form correlated equilibrium in polynomial time. *Internet and Network Economics* (2008), 506–513.

[26] A. X. Jiang and K. Leyton-Brown. 2015. Polynomial-time computation of exact correlated equilibrium in compact games. *Games and Economic Behavior* 91 (2015), 347–359.

[27] E. Kamenica and M. Gentzkow. 2011. Bayesian persuasion. *AM ECON REV* 101, 6 (2011), 2590–2615.

[28] L. G. Khachiyan. 1980. Polynomial algorithms in linear programming. *U. S. S. R. Comput. Math. and Math. Phys.* 20, 1 (1980), 53–72.

[29] G. P. McCormick. 1976. Computability of global solutions to factorable nonconvex programs: Part I-Convex underestimating problems. *Mathematical programming* 10, 1 (1976), 147–175.

[30] H. Moulin and J-P Vial. 1978. Strategically zero-sum games: the class of games whose completely mixed equilibria cannot be improved upon. *International Journal of Game Theory* 7, 3 (1978), 201–221.

[31] R. B. Myerson. 1986. Multistage Games with Communication. *Econometrica* 54, 2 (1986), 323–358.

[32] C. H. Papadimitriou and T. Roughgarden. 2008. Computing correlated equilibria in multi-player games. *Journal of the ACM (JACM)* 55, 3 (2008), 14.

[33] S. M. Ross. 1971. Goofspiel–the game of pure strategy. *Journal of Applied Probability* 8, 3 (1971), 621–625.

[34] T. Roughgarden. 2009. Intrinsic robustness of the price of anarchy. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*. ACM, 513–522.

[35] A. Saffidine, H. Finnsson, and M. Buro. 2012. Alpha-Beta Pruning for Games with Simultaneous Moves.. In *AAAI Conference on Artificial Intelligence (AAAI)*.

[36] Y. Shoham and K. Leyton-Brown. 2009. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.

[37] B. von Stengel. 1996. Efficient Computation of Behavior Strategies. *Games and Economic Behavior* 14, 2 (1996), 220 – 246.

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