
Polynomial-Time Computation of Exact Φ -Equilibria in Polyhedral Games

Gabriele Farina
MIT
gfarina@mit.edu

Charilaos Pipis
MIT
chpipis@mit.edu

Abstract

It is a well-known fact that correlated equilibria can be computed in polynomial time in a large class of concisely represented games using the celebrated Ellipsoid Against Hope algorithm (Papadimitriou and Roughgarden, 2008; Jiang and Leyton-Brown, 2015). However, the landscape of efficiently computable equilibria in sequential (extensive-form) games remains unknown. The Ellipsoid Against Hope does not apply directly to these games, because they do not have the required “polynomial type” property. Despite this barrier, Huang and von Stengel (2008) altered the algorithm to compute exact extensive-form correlated equilibria.

In this paper, we generalize the Ellipsoid Against Hope and develop a simple algorithmic framework for efficiently computing saddle-points in bilinear zero-sum games, even when one of the dimensions is exponentially large. Moreover, the framework only requires a “good-enough-response” oracle, which is a weakened notion of a best-response oracle.

Using this machinery, we develop a general algorithmic framework for computing exact linear Φ -equilibria in any polyhedral game (under mild assumptions), including correlated equilibria in normal-form games, and extensive-form correlated equilibria in extensive-form games. This enables us to give the first polynomial-time algorithm for computing exact linear-deviation correlated equilibria in extensive-form games, thus resolving an open question by Farina and Pipis (2023). Furthermore, even for the cases for which a polynomial time algorithm for exact equilibria was already known, our framework provides a conceptually simpler solution.

1 Introduction

The correlated equilibrium (CE), introduced by Aumann (1974), is one of the most seminal solution concepts in multi-player games. Contrary to the Nash equilibrium, in a correlated equilibrium the players’ strategies are correlated by a fictitious *mediator* that can recommend (but not enforce) behavior. It is then up to this mediator to ensure that the distribution of recommendations does not incentivize any player to *deviate* from their recommended strategy. It is known that this type of equilibrium naturally emerges from the repeated interaction of learning agents (Hart and Mas-Colell, 2000). In practice, this means that one can compute ϵ -approximate CEs in normal-form games by implementing suitable decentralized no-regret dynamics, of which several efficient implementations are known (see, *e.g.*, Blum and Mansour (2007) and Anagnostides et al. (2022)). However, this approach requires $\Omega(\text{poly}(1/\epsilon))$ iterations to compute an ϵ -approximate equilibrium, making it a non-viable choice for high-precision equilibrium computation. In a celebrated result, Papadimitriou and Roughgarden (2008) (with later refinements by Jiang and Leyton-Brown (2015)) devised an algorithm, called *Ellipsoid Against Hope*, that can compute an exact CE in a concisely represented normal-form game in polynomial time in the representation of the game. Their algorithm is an algorithmic version of the clever reduction of Hart and Schmeidler (1989) that casts the computation of CEs as a two-player zero-sum game.

The positive results for normal-form games, however, do not transfer directly to the significantly more involved setting of *extensive-form games*. Extensive-form games are games played on a game tree and can model sequential and simultaneous moves, as well as imperfect information. Despite a significant stream of *positive* results related to learning and equilibrium computation in extensive-form games, the complexity of computing CEs in extensive-form games remains to this day a major open question (Farina and Pipis, 2023; von Stengel and Forges, 2008). Due to its conjectured intractability, researchers have resorted to considering the computation of weaker and generalized notions of correlated equilibrium. A key reference point in this space is given by the framework of Gordon et al. (2008), who define a generalized notion of CE called Φ -equilibria. In a Φ -equilibrium, every player p is endowed with a set Φ_p of *behavior transformation functions*. The goal of the fictitious mediator is then simply to recommend strategies such that no player could unilaterally benefit from deviating by using any of the functions $\phi \in \Phi_p$. In this language, a CE corresponds to the Φ -equilibrium in which each Φ_p is the set of all possible functions from the strategy set of the player to itself. However, by considering appropriate subsets of behavior transformations, weaker supersets of CEs can be efficiently computed and learned through uncoupled learning dynamics. Notable examples of such equilibria in extensive-form games include the extensive-form correlated equilibrium (EFCE) (von Stengel and Forges, 2008), the extensive-form coarse-correlated equilibrium (EFCCE) (Farina et al., 2020), the normal-form coarse-correlated equilibrium (NFCCE) (Moulin and Vial, 1978), the recently-introduced linear-deviation correlated equilibrium (LCE) (Farina and Pipis, 2023), and others (Morrill et al., 2021).

Huang and von Stengel (2008) proposed a specialization of the Ellipsoid Against Hope algorithm to compute exact EFCE in extensive-form games. Later, Farina et al. (2022a) showed efficient no-regret dynamics that converge to the EFCE. More recently, there has been increased interest in understanding what is the Φ -equilibrium that is the closest to CE while still enabling efficient computation and learning. Farina and Pipis (2023) introduced the linear-deviation correlated equilibrium (LCE) that arises from the set Φ_{LIN} of all linear-swap deviations in sequence-form strategies and devise efficient no-linear-swap regret dynamics to approximate it. The LCE captures all notable notions of equilibrium that were previously known to be efficiently computable (including EFCE, EFCCE, and NFCCE). However, Farina and Pipis (2023) left open the key question as to whether LCEs themselves can also be computed exactly in polynomial time, akin to the generalization of the Ellipsoid Against Hope algorithm by Huang and von Stengel (2008), as opposed to just learned via uncoupled learning dynamics. The approach by Huang and von Stengel (2008) relies heavily on the combinatorial structure of the deviation functions that define EFCE, resulting in a rather involved algorithm. This is in stark contrast to the simple framework for constructing Φ -regret minimizers championed by Gordon et al. (2008). This begs the natural question:

*Can we always construct an efficient algorithm for exactly computing Φ -equilibria,
when there exists an efficient no- Φ -regret minimizer?*

In other words, can we create a simple and general framework in the spirit of Gordon et al. (2008) that can enable us to construct algorithms for the exact computation of Φ -equilibria in polyhedral games for any $\Phi \subseteq \Phi_{\text{LIN}}$? We answer this question in the affirmative.

Contributions. In this paper, we propose a framework for computing exact Φ -equilibria in general polyhedral games. Our framework recovers all positive results established by Papadimitriou and Roughgarden (2008), and crucially applies to polyhedral games such as extensive-form games. Using our framework, we develop the first polynomial-time algorithm for computing exact linear-deviation correlated equilibria in extensive-form games, thus resolving an open question by Farina and Pipis (2023). Furthermore, even for the cases for which a polynomial time algorithm for exact equilibria was already known (CEs in normal-form games (Papadimitriou and Roughgarden, 2008; Jiang and Leyton-Brown, 2015) and EFCEs in extensive-form games (Huang and von Stengel, 2008)), our framework provides a conceptually simpler solution.

We show that to compute an exact Φ -equilibrium in a polyhedral game, the following three conditions are sufficient:

1. The game satisfies the “polynomial utility gradient property” (Assumption 4.2) which states that given a product distribution over the joint strategy space of all n players, we can efficiently compute the expectation of the gradient of any player’s utility. This is a natural generalization of the “polynomial expectation property” of Papadimitriou and Roughgarden

(2008), and it is a rather low bar to clear (in fact, it is implicitly assumed in every no-regret learning algorithm).

2. Φ is a set of *linear* transformations (*i.e.*, of the form $\phi(\mathbf{x}) = \mathbf{B}\mathbf{x}$ for some matrix \mathbf{B}) that map the strategy set to itself. This is a technical requirement so that the expectation operator and the application of the deviation function can commute.¹ This condition is satisfied by all notions of Φ -equilibrium mentioned above, including EFCE and LCE in extensive-form games, and CE in normal-form games.
3. The set Φ of transformations is a polytope that admits a polynomial-time separation oracle.

The separation oracle requirement in the third condition is known to be equivalent to efficient linear optimization (Grötschel et al., 1993). In essence, this means that giving a polynomially-sized characterization of a set Φ of linear transformations for a polyhedral set is a sufficient condition to provide both efficient no-regret learning dynamics and an efficient algorithm for computing exact Φ -equilibria. This is exactly what we achieve by applying our result to the set of linear-swap deviations that was recently characterized (Farina and Pipis, 2023; Zhang et al., 2024b) as a polytope of polynomially many constraints and was used to prove efficient no-linear-swap regret dynamics. In light of all these considerations, our framework can be thought of as the counterpart of the Φ -regret minimization framework by Gordon et al. (2008), but for computation of *exact* equilibria rather than regret minimization.

At the heart of our construction, our main technical tool is a generalization of the methodology of the Ellipsoid Against Hope (Papadimitriou and Roughgarden, 2008; Jiang and Leyton-Brown, 2015) to general polyhedral bilinear games. In more detail, we give a new constructive proof of the minimax theorem for players with polyhedral strategy sets, by using only a weakened type of a best-response oracle that we coin Good-Enough-Response (GER) oracle. An interesting property of the GER oracle is that it can be computationally tractable even when the respective best-response oracle is intractable, as we show in Section 4. This algorithmic idea is likely of independent interest and is especially useful when the strategy space of one of the players is very large but there exists an efficient GER oracle that outputs sparse solutions (*e.g.*, vertices of a high-dimensional polytope). This is exactly the type of problem we face when we need to compute exact Φ -equilibria in polyhedral games and we then proceed to apply this machinery to the above question. Interestingly, in order to show the existence of structured good-enough responses in the context of Φ -equilibria, we use an argument based on the existence of an efficient fixed-point oracle for each deviation $\phi \in \Phi$. Such an ingredient was fundamental (albeit used differently; see Hazan and Kale (2007) for a discussion of the role played by fixed-point oracles in the construction of no- Φ -regret algorithms) also in Gordon et al. (2008). In our case, it is one of the technical insights that enable us to sidestep much of the intricacy encountered by Huang and von Stengel (2008).

We defer all proofs of the paper to the appendix.

Related work. We include an extensive discussion of related work in Appendix A.

2 Preliminaries

In this section, we introduce some basic concepts and definitions that will be used in developing our framework.

2.1 Polyhedra, polytopes, and convex sets

Definition 2.1 (Rational polyhedron). *A rational polyhedron $\mathcal{P} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{Ax} \leq \mathbf{b}\}$ is the solution set of a system of linear inequalities with rational coefficients. We say that \mathcal{P} has **facet-complexity** φ if there exists a system of linear inequalities, where each inequality has encoding length² at most φ , and whose solution set is \mathcal{P} . A rational polyhedron that is bounded is called a rational polytope.*

¹Note that going beyond linear transformations can introduce several complications. Most notably, in a recent paper, Zhang et al. (2024a) observe that computing exact fixed-points of non-linear transformations might be PPAD-hard and they instead introduce a new way to perform regret minimization using “approximate expected fixed-points”.

²The encoding length of an inequality is the total amount of bits required to represent all of its coefficients.

One important property of rational polytopes that we will use repeatedly throughout the paper is that they can equivalently be written as the convex hull of a *finite* number of points. We call these points the *vertices* $V(\mathcal{P})$ of polytope \mathcal{P} . Additionally, the vertices of a rational polytope always have rational coordinates and encoding length $\text{poly}(\varphi)$ (Grötschel et al., 1993, Lemma 6.2.4).

Since we are interested in constructing algorithms that perform exact computations, any discussion of non-rational numbers is not relevant. Thus, from now on, every time we deal with a polytope we will mean a rational polytope.

In our algorithm, we will also make use of the concept of *conic hull*, which is introduced next.

Definition 2.2 (Conic hull). *The conic hull of a convex set \mathcal{X} is $\mathbb{R}_+ \mathcal{X} = \{t \cdot \mathbf{x} \mid t \geq 0, \mathbf{x} \in \mathcal{X}\}$. Furthermore, if \mathcal{X} is a rational polyhedron, its conic hull is also a rational polyhedron.*

2.2 Game theory definitions

We begin by defining polyhedral games, following Gordon et al. (2008). But first, we need to define multi-linear functions.

Definition 2.3 (Multi-linear function). *Let V_1, \dots, V_n be vector spaces. A function $f : V_1 \times \dots \times V_n \rightarrow \mathbb{R}$ is said to be multi-linear if for each $p \in [n]$ and fixed $\mathbf{v}_{-p} \in V_{-p}$ the function $f(\mathbf{v}_p, \mathbf{v}_{-p})$ is linear in $\mathbf{v}_p \in V_p$. In other words, if $\nabla f(\mathbf{v}_{-p})$ is the gradient of $f(\mathbf{v}_p, \mathbf{v}_{-p})$ with respect to \mathbf{v}_p when \mathbf{v}_{-p} is fixed, then $f(\mathbf{v}_p, \mathbf{v}_{-p}) = \mathbf{v}_p \cdot \nabla f(\mathbf{v}_{-p})$.*

Definition 2.4 (Polyhedral game). *In a polyhedral game with n players, every player $p \in [n]$ has a polytope³ strategy set $\mathcal{A}_p \subset \mathbb{R}^{d_p}$ and a multi-linear utility function $u_p : \mathcal{A}_1 \times \dots \times \mathcal{A}_n \rightarrow \mathbb{R}$*

Some notable examples of polyhedral games are: normal-form games, where every player has a probability simplex as their strategy set, and extensive-form games, where the strategy sets of the players are the sets of sequence-form strategies (Romanovskii, 1962; Koller et al., 1996; von Stengel, 1996). We will refer to the encoding length of the game as the *size of the game*. In games of interest this is usually much smaller than holding the full utility function; for example, extensive-form games are encoded using a game tree and different classes of normal-form games can have other succinct descriptions (Papadimitriou and Roughgarden, 2008).

A sub-class of polyhedral games that will be particularly useful in our paper is that of bilinear zero-sum games, which is defined below.

Definition 2.5 (Bilinear zero-sum game). *Let $\mathcal{X} \subset \mathbb{R}^M$, $\mathcal{Y} \subset \mathbb{R}^N$ be two rational polytopes. A bilinear zero-sum game is a game between two players with strategy sets \mathcal{X} and \mathcal{Y} such that the utility of the \mathcal{X} -player is $u_1(\mathbf{x}, \mathbf{y}) = \mathbf{x}^\top \mathbf{A} \mathbf{y}$, for some $\mathbf{A} \in \mathbb{Q}^{M \times N}$, and the utility of the \mathcal{Y} -player is $u_2(\mathbf{x}, \mathbf{y}) = -u_1(\mathbf{x}, \mathbf{y})$*

We can now define the notion of a Φ -equilibrium, which generalizes the correlated equilibrium for arbitrary n -player polyhedral games and sets of strategy transformations Φ . Before we do that, we first need to define the *corner game* $\Gamma(G)$ of a polyhedral game G , following Gordon et al. (2008); Marks (2008). This is the game that arises if we let the action sets of every player p be equal to the vertices $V(\mathcal{A}_p)$ of the polytope strategy set of that player. Note that since \mathcal{A}_p is a polytope, it will have a finite number of vertices. The utilities of this game for a player $p \in [n]$ and pure strategy profile $\mathbf{s} \in V(\mathcal{A}_1) \times \dots \times V(\mathcal{A}_n)$ are simply given by $u_p(\mathbf{s})$. In this paper, we will denote the vertices of every strategy set in a polyhedral game as $\Pi_p = V(\mathcal{A}_p)$. We are now ready to define the Φ -equilibrium.

Definition 2.6 (Φ -equilibrium). *Let G be a polyhedral game of n players and Φ_p be a set of strategy transformations $\phi_p : \mathcal{A}_p \rightarrow \mathcal{A}_p$ for each player $p \in [n]$. A $\{\Phi_p\}$ -equilibrium for G is a joint distribution $\mu \in \Delta(\Pi_1 \times \dots \times \Pi_n)$ on the pure strategy profiles of $\Gamma(G)$, such that for every player $p \in [n]$ and deviation $\phi \in \Phi_p$ it holds*

$$\mathbb{E}_{\mathbf{s} \sim \mu} [u_p(\mathbf{s})] \geq \mathbb{E}_{\mathbf{s} \sim \mu} [u_p(\phi(\mathbf{s}_p), \mathbf{s}_{-p})].$$

That is, no player p has an incentive to unilaterally deviate from the recommended joint strategy \mathbf{s} using any transformation $\phi \in \Phi_p$.

³Despite their name, polyhedral games have strategy sets that are *polytopes*, that is, *bounded* polyhedra.

3 A simple framework for computing equilibria in bilinear zero-sum games using good-enough-response (GER) oracles

We begin by introducing a simple algorithmic framework (Theorem 3.1) for computing min-max equilibria in bilinear zero-sum games. As mentioned before, it relies on the idea of good-enough-responses. The motivation behind this is that sometimes a best-response oracle is not known, or even NP-hard to construct (as we prove in Theorem 4.7). On the contrary, good-enough-responses might be a readily available primitive. For example, we will see in Section 4 that a good-enough-response oracle materializes through the use of fixed-point oracles for transformations $\phi \in \Phi$ and this enables us to devise polynomial time algorithms for computing exact Φ -equilibria in polyhedral games.

Let us assume that we have a bilinear zero-sum game $\mathcal{G}(\mathcal{X}, \mathcal{Y}, \mathbf{A})$, where the strategy sets $\mathcal{X} \subset \mathbb{R}^M, \mathcal{Y} \subset \mathbb{R}^N$ are rational polytopes. We typically assume that $M \gg N$. Additionally, let

$$\text{OPT} = \max_{\mathbf{x} \in \mathcal{X}} \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y},$$

be the value of the game at equilibrium, which is known to us. *In the rest of the paper we assume that $\text{OPT} = 0$.* This is without loss of generality because otherwise, it is possible to create a new game with this property by augmenting the vectors \mathbf{x}, \mathbf{y} with an extra dimension as follows:

$$\begin{bmatrix} \mathbf{x}^\top & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0}^\top & -\text{OPT} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ 1 \end{bmatrix} = \mathbf{x}^\top \mathbf{A} \mathbf{y} - \text{OPT}.$$

Our framework is a formalization of the following observation. The statement

(S1) Given any $\mathbf{y} \in \mathcal{Y}$ we can find some $\mathbf{x} = \mathbf{x}(\mathbf{y}) \in \mathcal{X}$ such that $\mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0$.

implies the following

(S2) There exists $\mathbf{x}^* \in \mathcal{X}$ such that $(\mathbf{x}^*)^\top \mathbf{A} \mathbf{y} \geq 0$ for all $\mathbf{y} \in \mathcal{Y}$.

This follows from the minimax theorem (Neumann, 1928), as the first statement (S1) is equivalent to $\min_{\mathbf{y}} \max_{\mathbf{x}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0$, while the second statement (S2) is equivalent $\max_{\mathbf{x}} \min_{\mathbf{y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0$.

We are interested in the following question: “*Is there an efficient algorithm that when given access to an oracle for (S1), it constructs a solution \mathbf{x}^* for (S2) represented as a mixture of a small number of oracle responses?*”.

3.1 Good-Enough-Response (GER) oracle

We begin by formally defining the oracle we presented previously, which we coin a Good-Enough-Response (GER) oracle. It is defined as follows:

GER(\mathbf{y}):
return $(\mathbf{x}, \mathbf{x}^\top \mathbf{A}) \in \mathcal{X} \times \mathbb{Q}^N$ s.t. $\mathbf{x}^\top \mathbf{A} \mathbf{y} \geq \text{OPT} = 0$

where $\mathbf{y} \in \mathcal{Y} \subset \mathbb{R}^N$, and $\text{OPT} = 0$ as was discussed earlier. Note that this is not a best-response oracle, because it does not return an $\mathbf{x} \in \mathcal{X}$ that maximizes the utility of the max-player. Rather, it suffices to return a “good enough response”, hence the name.

In fact, our algorithms will often need to query a GER oracle for $\mathbf{y}' \in \mathbb{R}_+ \mathcal{Y}$ and not just for vectors in \mathcal{Y} . This however is not a problem because it suffices to find any $\mathbf{y} = \mathbf{y}'/\alpha$ for some $\alpha > 0$ and $\mathbf{y} \in \mathcal{Y}$ and then query GER(\mathbf{y}) instead. To find such a \mathbf{y} efficiently we can again, without loss of generality, assume that all vectors $\mathbf{y} \in \mathcal{Y}$ are augmented with an extra dimension (call it $\mathbf{y}[\emptyset]$) such that $\mathbf{y}[\emptyset] = 1$ for all $\mathbf{y} \in \mathcal{Y}$. Then we can find the desired scaling factor immediately because $\mathbf{y}'[\emptyset] = \alpha$ if and only if $\mathbf{y}' = \alpha \mathbf{y}$ for $\mathbf{y} \in \mathcal{Y}$.

In addition to a good-enough-response oracle, our algorithm also requires a separation oracle $\text{SEP}_{\mathcal{Y}}$ for the polytope \mathcal{Y} , which can be easily converted to a separation oracle for $\mathbb{R}_+ \mathcal{Y}$ by the same “augmenting” argument as before. Combining these two, we can make the final separation oracle (Algorithm 2) that is needed to execute the ellipsoid method on (D) , as presented later. Specifically, if $\mathbf{y} \notin \mathbb{R}_+ \mathcal{Y}$ then we simply return a separating hyperplane via $\text{SEP}_{\mathbb{R}_+ \mathcal{Y}}$, else we return a good-enough-response from GER.

3.2 The framework

Our goal is to compute an $\mathbf{x} \in \mathcal{X}$ that is an optimal (min-max) strategy for the max-player. Equivalently, we seek to find a solution to the following linear program

$$\text{find } \mathbf{x} \in \mathcal{X} \quad \text{s.t. } \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0 \quad (P)$$

This is an LP with M variables which is typically assumed to be much greater (even super-exponentially greater) than N . When faced with this situation, one might want to attempt to directly solve the dual of (P) using the ellipsoid method. However, this would require a proper separation oracle for the dual problem, which corresponds to a linear optimization oracle, or at least a best-response oracle. But as we explained, the oracle access we have is weaker.

Instead, we focus on the below linear program. Note that for any $\mathbf{y} \in \mathbb{R}_+ \mathcal{Y}$, $\text{GER}(\mathbf{y})$ should always return an $\mathbf{x} \in \mathcal{X}$ such that $(\mathbf{x}^\top \mathbf{A}) \mathbf{y} \geq 0$, which is a violated constraint of (D) . Thus, we can combine GER and a separation oracle for $\mathbb{R}_+ \mathcal{Y}$ (as in Algorithm 2) to make a separation oracle for this LP.

$$\text{find } \mathbf{y} \in \mathbb{R}_+ \mathcal{Y} \quad \text{s.t. } \max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \leq -1 \quad (D)$$

By the Generalized Farkas lemma (Lemma B.2) and the fact that (P) is feasible, it immediately follows that (D) must be infeasible. Despite the infeasibility, and following the “Against Hope” step of Papadimitriou and Roughgarden (2008), we execute the ellipsoid method on (D) using Algorithm 2 as a separation oracle. The ellipsoid method will run for a number $L = \text{poly}(N)$ of steps and then conclude that (D) is infeasible. Let $\mathbf{x}_1, \dots, \mathbf{x}_L$ be the response vectors returned by GER during this process. We now consider a “compressed” version of the previous LP that only uses vectors \mathbf{x} from the convex hull $\text{co}\{\mathbf{x}_k\}$ of these responses.

$$\text{find } \mathbf{y} \in \mathbb{R}_+ \mathcal{Y} \quad \text{s.t. } \max_{\mathbf{x} \in \text{co}\{\mathbf{x}_k\}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \leq -1 \quad (D')$$

We argue that this LP must also be infeasible; the ellipsoid method is a deterministic algorithm and if we execute it on (D') it will go through the same sequence of candidate points \mathbf{y}_k , to which we can respond with the same sequence of separating hyperplanes as before. These hyperplanes will still be valid for (D') because all of the response vectors we used previously exist in $\text{co}\{\mathbf{x}_k\}$.

Now, using Generalized Farkas lemma (Lemma B.2) again and the fact that (D') is infeasible, it follows that the LP shown below must be feasible.

$$\text{find } \mathbf{x} \in \text{co}\{\mathbf{x}_k\} \quad \text{s.t. } \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0 \quad (P')$$

This is a “compressed” version of (P) , because now every vector $\mathbf{x} \in \text{co}\{\mathbf{x}_k\}$ can be represented as a vector of size L that corresponds to a convex combination of the response vectors $\mathbf{x}_1, \dots, \mathbf{x}_L$. Finally, since (P') is an LP with only a polynomial number of variables, we can solve it in polynomial time using any LP algorithm. This will clearly be a feasible solution for our initial LP (P) , because $\text{co}\{\mathbf{x}_k\} \subset \mathcal{X}$. The full algorithm is shown below, in Algorithm 1. Note that in reality we only use the LPs (D) and (P') . The rest were used as intermediate steps for the presentation of the algorithm.

Algorithm 1: Ellipsoid Against Hope for bilinear zero-sum games

Input: Separation oracle $\text{SEP}_{\mathbb{R}_+ \mathcal{Y}}$ for $\mathbb{R}_+ \mathcal{Y}$, and a good-enough-response oracle GER .

Output: A sparse solution \mathbf{x}^* of (P) represented as a mixture of GER oracle responses.

Execute the ellipsoid method on (D) , using Algorithm 2 as a separation oracle;

Create (P') using the response vectors and compute a feasible solution \mathbf{x}^* ;

Theorem 3.1. *If the following hold*

1. $\mathcal{X} \subset \mathbb{R}^M, \mathcal{Y} \subset \mathbb{R}^N$ are rational polytopes and \mathcal{Y} has facet-complexity at most φ ,
2. we have access to a separation oracle $\text{SEP}_{\mathcal{Y}}$ for \mathcal{Y} and a good-enough-response oracle GER ,
3. the encoding length of $\mathbf{x}^\top \mathbf{A}$ is at most φ for all GER oracle responses and all vertices of \mathcal{X} ,

then Algorithm 1 runs in $\text{poly}(N, \varphi)$ time, performs $L = \text{poly}(N, \varphi)$ oracle calls, and computes an exact solution \mathbf{x}^* of (P) that is a mixture of at most N oracle responses. In particular, the encoding length of \mathbf{x}^* depends polynomially on the encoding length of the GER oracle responses.

Note that since we have assumed that $M \gg N$, it would not make sense for the final solution \mathbf{x}^* to have encoding length $\text{poly}(M)$, as this would invalidate the whole algorithm. In order for the solution to make sense, the GER oracle must only give responses with low encoding length. This is exactly the case in Section 4, where M is a doubly-exponential quantity in the size of the problem, while the GER responses are vectors with only one non-zero entry.

4 Computing linear Φ -equilibria in polynomial time

We have seen in Section 3 how one can compute exact min-max equilibria using good-enough-response (GER) oracles. Now it is time to apply this machinery in the problem of computing *exact* Φ -equilibria in polyhedral games. Crucially, the factor that enables us to utilize the framework of Section 3 is the existence of an efficient GER oracle, which effectively boils down to constructing a product distribution consisting of fixed-points for the strategies of every player of the game.

Let G be any polyhedral game (Definition 2.4) with n players and strategy sets $\mathcal{A}_p \subset \mathbb{R}^{d_p}$ for $p \in [n]$. In this section we apply the framework we developed previously to construct an algorithm that computes an exact Φ -equilibrium of G in polynomial time when Φ is a polytope containing valid linear transformations from polyhedral strategies to polyhedral strategies. Notable examples of sets with these properties are the trigger deviations used for EFCE (Farina et al., 2022a), and the linear-swap deviations used for LCE (Farina and Pipis, 2023) in extensive-form games.

The general idea of our construction is that of the existence proof by Hart and Schmeidler (1989) that casts the problem of Φ -equilibrium computation as one of computing a min-max equilibrium in a two-player zero-sum meta-game between a “Correlator”, who acts upon the simplex of all pure strategy profiles, and a “Deviator”, whose actions correspond to deviations for every player. We call this a *Correlator-Deviator game*.

To make this idea applicable to polyhedral games, we generalize it as follows. We define a bilinear zero-sum meta-game with strategy sets \mathcal{X}, \mathcal{Y} for the two players, where \mathcal{X} is the set of all joint distributions over strategy profiles, $\mathcal{X} = \Delta(\Pi_1 \times \dots \times \Pi_n)$ (hence, a polytope) and \mathcal{Y} is the Cartesian product of Φ_p for all players p , $\mathcal{Y} = \Phi_1 \times \dots \times \Phi_n$, which is a convex set – and in our case, a polytope.

We remark here that linear transformations ϕ_p can be represented using a matrix \mathbf{B}_p such that $\phi_p(\mathbf{x}_p) = \mathbf{B}_p \mathbf{x}_p$. Thus, when we say that Φ_p is a polytope, it means that there exists a system of inequalities that can describe the entries of the corresponding matrix \mathbf{B}_p for every $\phi_p \in \Phi_p$. For notational convenience, we will interchangeably use Φ_p to denote either the set of transformation functions, or a polytope describing the vectors (flattened \mathbf{B}_p matrices) that correspond to transformations. In any event, it should not matter which of the two representations we have, because they are completely equivalent.

The utility matrix \mathbf{U} of the Correlator in the meta-game is shown below. Specifically, it has one row for each pure strategy profile $\mathbf{s} \in \Pi_1 \times \dots \times \Pi_n$, and one column for each tuple $j = (p, a, b)$, where $a, b \in [d_p]$ are used as indices over strategy vectors $\mathbf{s}_p \in \mathcal{A}_p$. Additionally, we always want the final expression to have a quantity $(\sum_p \mathbb{E}_{s \sim \mathbf{x}}[u_p(s)])$ that is independent of the value of \mathbf{y} . To achieve this we can use a trick similar to the one used to make $\text{OPT} = 0$ in Section 3 by augmenting vectors $\mathbf{y} \in \mathcal{Y}$ with an extra dimension \emptyset such that $\mathbf{y}[\emptyset] = 1$ always holds. Then we have⁴

$$\mathbf{U}_{sj} = \begin{cases} \sum_p u_p(s), & j = \emptyset \\ -\mathbf{s}_p[a] u_p(\mathbf{1}_b, \mathbf{s}_{-p}), & \text{otherwise} \end{cases}$$

where $\mathbf{1}_b$ denotes the vector having all 0, apart from index b , which is 1. Note that the number of rows of \mathbf{U} might be doubly-exponential (exponential both in the number of players and the dimension of the polyhedral strategies), which is in contrast to the original Ellipsoid Against Hope algorithm that only allowed a number of rows exponential in the number of players.

Lemma 4.1. *Let G be a polyhedral game with pure strategy set Π_p for every player $p \in [n]$. Additionally, let Φ_p be a set of linear transformations for every $p \in [n]$. If $\mathbf{x} \in \mathcal{X} = \Delta(\Pi_1 \times \dots \times \Pi_n)$ and $\mathbf{y} = (\phi_1, \dots, \phi_n) \in \mathcal{Y} = \Phi_1 \times \dots \times \Phi_n$ then*

$$\mathbf{x}^\top \mathbf{U} \mathbf{y} = \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})].$$

⁴We are slightly abusing the notation here and use $u_p(\mathbf{1}_b, \mathbf{s}_{-p})$ instead of $\mathbf{1}_b \cdot \nabla u_p(\mathbf{s}_{-p})$.

It is now evident that our goal is to compute a joint distribution that is a solution to the following linear program: find $\mathbf{x} \in \mathcal{X}$ s.t. $\min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{U} \mathbf{y} \geq 0$.

Observe that this is slightly different from the required non-negativity in Definition 2.6; there we want the individual (per-player) expectations to be non-negative, while here it suffices for the minimum of the *sum* of expectations to be non-negative. However, we can assume without loss of generality that the identity transformation is always a valid transformation⁵. Then, every LP solution $\mathbf{x} \in \mathcal{X}$ will satisfy $\mathbf{x}^\top \mathbf{U} \mathbf{y} \geq 0$ for all $\mathbf{y} \in \mathcal{Y}$, including $\mathbf{y} = (\phi_1, \mathbf{I}, \dots, \mathbf{I}), (\mathbf{I}, \phi_2, \dots, \mathbf{I}), \dots$ that correspond to the individual expectations.

The previous LP respects exactly the structure of (P) that our min-max framework can handle. The only remaining component to get a polynomial-time algorithm is to have an efficient good-enough-response oracle GER. Specifically, for any valid $\mathbf{y} \in \mathcal{Y}$, we need to respond with an \mathbf{x} such that $\mathbf{x}^\top \mathbf{U} \mathbf{y} \geq 0$. The important insight that allows us to construct an efficient oracle and uncover sparse solutions is that we can always find such an \mathbf{x} that is a product distribution — similar to the original Ellipsoid Against Hope algorithm (Papadimitriou and Roughgarden, 2008) that was based on the observation by Hart and Schmeidler (1989). Note that we can always represent a product distribution by simply specifying its marginals and those, in turn, can always be represented as polyhedral strategies. Thus, representing the product distribution \mathbf{x} only requires linear space in the game size.

Next, we define an important property that a game must have to enable the efficient implementation of the GER oracle we propose (Lemma D.1) for our algorithm.

Assumption 4.2 (Polynomial utility gradient property). *Given a product distribution $\mathbf{x} \in \Delta(\Pi_1 \times \dots \times \Pi_n)$, it is possible to compute the value of*

$$\mathbf{g}_p(\mathbf{x}_{-p}) = \mathbb{E}_{\mathbf{s}_{-p} \sim \mathbf{x}_{-p}} [\nabla u_p(\mathbf{s}_{-p})]$$

for all players $p \in [n]$ in polynomial time in the encoding length of \mathbf{x} and the size of the game.

This assumption generalizes the polynomial expectation property defined in Papadimitriou and Roughgarden (2008) to more general, polyhedral games. In particular, if we have a normal-form game, the polynomial expectation property amounts to computing $\mathbf{g}_p(\mathbf{x}_{-p}) \cdot \mathbf{x}_p$ for a product distribution \mathbf{x} . Moreover, as we stated in the introduction, this assumption is very natural for one more reason; it is implicitly assumed in every no-regret learning algorithm.

Remark 4.3. Papadimitriou and Roughgarden (2008) also defined a second property that is required for efficient computation, called the “polynomial type” property. Even though our algorithm does not require this property, a variant of it is implicit in the fact that the complexity of the algorithm depends polynomially in the number of players and the dimension of every player’s strategy set \mathcal{A}_p . However, this relaxation is what allows our algorithm to handle much broader classes of games, such as the extensive-form games that do not have the polynomial type property.

Theorem 4.4. *Let G be a polyhedral game (Definition 2.4) of n players and $\{\Phi_p\}$ be a collection of polytopes corresponding to sets of linear strategy transformations that map every strategy set \mathcal{A}_p to itself. Additionally, let $N = \sum_p d_p^2$. Assume that*

- there exist polynomial-time separation oracles for \mathcal{A}_p and Φ_p ,
- G satisfies the polynomial utility gradient property (Assumption 4.2),
- ψ is an upper bound on the facet-complexity of every \mathcal{A}_p and Φ_p ,
- $\log u$ is the maximum encoding length of the utilities of G .

Then there exists an algorithm that computes an exact $\{\Phi_p\}$ -equilibrium of G in time $\text{poly}(N, \log u, \psi)$ and performs $\text{poly}(N, \log u, \psi)$ number of calls to all the separation oracles. Additionally, the equilibrium is represented as a convex combination of at most N pure strategy profiles.

As a first application of this framework, we argue that it can be applied to normal-form games that satisfy the polynomial type and the polynomial expectation property, defined in Papadimitriou and Roughgarden (2008).

⁵Otherwise we can replace each Φ_p with $\text{co}\{\Phi_p \cup \{\mathbf{I}\}\}$ which remains a rational polytope and admits a separation oracle when Φ_p has a separation oracle.

Corollary 4.5 (Exact CE in normal-form games). *If a normal-form game G has the polynomial type and the polynomial expectation property, defined in Papadimitriou and Roughgarden (2008), then our algorithm computes an exact correlated equilibrium of G and runs in polynomial time in the size of the game.*

As we have discussed, a very notable example of polyhedral games is that of extensive-form games. Next, we apply Theorem 4.4 to this class of games, and specifically to the set of all linear-swap deviations, recently defined in Farina and Pipis (2023). In particular, this set contains all trigger deviations and thus, our algorithm also produces an extensive-form correlated equilibrium (EFCE) in a conceptually simpler manner than in the early work of Huang and von Stengel (2008).

Corollary 4.6 (Exact LCE computation). *There exists an algorithm that runs in $\text{poly}(N, \log u)$ time and computes an exact linear-deviation correlated equilibrium (LCE) in an extensive-form game.*

Finally, we prove in Theorem 4.7 that, at least in the case of computing Φ -equilibria in polyhedral games, the use of a GER over a best-response oracle is not just more elegant, but it is also necessary because constructing a best-response oracle is NP-hard. At the heart of our hardness result lies a reduction from SAT to equilibrium computation in extensive-form games that has also been used in the past to prove the hardness of equilibrium selection for EFCE and LCE (von Stengel and Forges, 2008; Farina and Pipis, 2023). In a sense, constructing a best-response oracle is as hard as the equilibrium selection problem, while constructing a good-enough-response oracle amounts to computing fixed-points of strategy transformation functions. This further highlights the importance of having a framework akin to the one presented in Section 3 for designing new algorithms; the hardness result rules out solutions that require responses competitive against *any* threshold, but sometimes it is sufficient to only compete with a particular good-enough threshold.

Theorem 4.7 (Hardness of BR oracle). *It is NP-hard to construct a best-response oracle for the Correlator in the Correlator-Deviator game.*

5 Discussion and Future Work

In this paper, we devise a polynomial-time algorithm for computing min-max equilibria in bilinear zero-sum games, by utilizing a good-enough-response oracle. We use this machinery to develop a simple general framework for the *exact* computation of Φ -equilibria in polyhedral games for sets Φ of linear strategy transformations. This framework parallels that of Gordon et al. (2008) on no-regret dynamics, but for exact equilibrium computation. Applying this to extensive-form games, we construct the first polynomial-time algorithm for computing exact linear-deviation correlated equilibria in extensive-form games – a question that had been left open by Farina and Pipis (2023).

We believe that having a simple framework to use as a mental model to guide algorithm design is of paramount importance for the advancement of the field. The Φ -regret minimization framework of Gordon et al. (2008) is indicative of this fact, because it has been key to many interesting results over the years (Morrill et al., 2021; Farina et al., 2022a; Anagnostides et al., 2022; Farina and Pipis, 2023). Compared to no-regret learning, the problem of exact equilibrium computation has been much less studied (basically only in Papadimitriou and Roughgarden (2008); Jiang and Leyton-Brown (2015); Huang and von Stengel (2008)) and we hope that offering a simplified framework will give new insights to advance this front, perhaps aiding in the discovery of new, more practical, algorithms.

Several key questions remain underinvestigated.

- Despite its great theoretical importance, our framework (based on the ellipsoid algorithm) has a polynomial time complexity of rather large degree. Could one devise a more practical alternative while retaining a similar level of generality?
- Can our framework be easily generalized to convex strategy spaces (instead of polytopes)?
- Is there a similar algorithmic framework to compute exact Φ -equilibria in extensive-form games for non-linear transformations Φ ? In recent work, Zhang et al. (2024a) give parameterized algorithms for minimizing Φ -regret when Φ is the set of all degree- k polynomial swap deviations. Can similar guarantees be achieved for high-precision computation of these Φ -equilibria?
- Can ideas similar to those presented in this paper be applied to Markov games?

References

AmirMahdi Ahmadinejad, Sina Dehghani, MohammadTaghi Hajiaghayi, Brendan Lucier, Hamid Mahini, and Saeed Seddighin. 2019. From Duels to Battlefields: Computing Equilibria of Blotto and Other Games. *Mathematics of Operations Research* 44, 4 (2019), 1304–1325.

Ioannis Anagnostides, Constantinos Daskalakis, Gabriele Farina, Maxwell Fishelson, Noah Golowich, and Tuomas Sandholm. 2022. Near-Optimal No-Regret Learning for Correlated Equilibria in Multi-Player General-Sum Games. In *ACM Symposium on Theory of Computing*.

Angelos Assos, Idan Attias, Yuval Dagan, Constantinos Daskalakis, and Maxwell K. Fishelson. 2023. Online Learning and Solving Infinite Games with an ERM Oracle. In *The Thirty Sixth Annual Conference on Learning Theory, COLT 2023, 12-15 July 2023, Bangalore, India (Proceedings of Machine Learning Research, Vol. 195)*, Gergely Neu and Lorenzo Rosasco (Eds.). PMLR, 274–324.

Robert J Aumann. 1974. Subjectivity and correlation in randomized strategies. *Journal of mathematical Economics* 1, 1 (1974), 67–96.

Avrim Blum and Yishay Mansour. 2007. From External to Internal Regret. *J. Mach. Learn. Res.* 8 (2007).

G.W. Brown. 1951. Iterative Solutions of Games by Fictitious Play. In *Activity Analysis of Production and Allocation*, T. C. Koopmans (Ed.). Wiley, New York.

Darshan Chakrabarti, Gabriele Farina, and Christian Kroer. 2024. Efficient Learning in Polyhedral Games via Best Response Oracles. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Yuval Dagan, Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. 2024. From External to Swap Regret 2.0: An Efficient Reduction for Large Action Spaces (*STOC 2024*). Association for Computing Machinery, New York, NY, USA, 1216–1222.

Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. 2009. The complexity of computing a Nash equilibrium. *Commun. ACM* 52, 2 (feb 2009), 89–97.

FAIR, Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, Mojtaba Komeili, Karthik Konath, Minae Kwon, Adam Lerer, Mike Lewis, Alexander H. Miller, Sasha Mitts, Adithya Renduchintala, Stephen Roller, Dirk Rowe, Weiyan Shi, Joe Spisak, Alexander Wei, David Wu, Hugh Zhang, and Markus Zijlstra. 2022. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science* 378, 6624 (2022), 1067–1074.

Gabriele Farina, Tommaso Bianchi, and Tuomas Sandholm. 2020. Coarse Correlation in Extensive-Form Games. In *AAAI Conference on Artificial Intelligence*.

Gabriele Farina, Andrea Celli, Alberto Marchesi, and Nicola Gatti. 2022a. Simple Uncoupled No-regret Learning Dynamics for Extensive-form Correlated Equilibrium. *J. ACM* 69, 6 (2022).

Gabriele Farina, Chung-Wei Lee, Haipeng Luo, and Christian Kroer. 2022b. Kernelized Multiplicative Weights for 0/1-Polyhedral Games: Bridging the Gap Between Learning in Extensive-Form and Normal-Form Games. In *International Conference on Machine Learning*.

Gabriele Farina and Charilaos Pipis. 2023. Polynomial-Time Linear-Swap Regret Minimization in Imperfect-Information Sequential Games. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Ian Gemp, Yoram Bachrach, Marc Lanctot, Roma Patel, Vibhavari Dasagi, Luke Marrs, Georgios Piliouras, Siqi Liu, and Karl Tuyls. 2024. States as Strings as Strategies: Steering Language Models with Game-Theoretic Solvers. arXiv:2402.01704 [cs.CL]

Gauthier Gidel, Tony Jebara, and Simon Lacoste-Julien. 2017. Frank-Wolfe Algorithms for Saddle Point Problems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 54)*, Aarti Singh and Jerry Zhu (Eds.). PMLR, 362–371.

Paul W. Goldberg and Francisco J. Marmolejo-Cossío. 2021. Learning Convex Partitions and Computing Game-theoretic Equilibria from Best-response Queries. *ACM Transactions on Economics and Computation* 9, 1 (2021), 3:1–3:36.

Geoffrey J Gordon, Amy Greenwald, and Casey Marks. 2008. No-regret learning in convex games. In *International Conference on Machine learning*. 360–367.

Martin Grötschel, László Lovász, and Alexander Schrijver. 1993. *Geometric Algorithms and Combinatorial Optimization*. Springer Berlin, Heidelberg.

Sergiu Hart and Andreu Mas-Colell. 2000. A Simple Adaptive Procedure Leading to Correlated Equilibrium. *Econometrica* 68, 5 (2000), 1127–1150.

Sergiu Hart and David Schmeidler. 1989. Existence of Correlated Equilibria. *Mathematics of Operations Research* 14, 1 (1989), 18–25.

Elad Hazan and Satyen Kale. 2007. Computational Equivalence of Fixed Points and No Regret Algorithms, and Convergence to Equilibria. In *Advances in Neural Information Processing Systems*, J. Platt, D. Koller, Y. Singer, and S. Roweis (Eds.), Vol. 20. Curran Associates, Inc.

Wan Huang and Bernhard von Stengel. 2008. Computing an extensive-form correlated equilibrium in polynomial time. In *International Workshop on Internet and Network Economics*. Springer, 506–513.

Albert Xin Jiang and Kevin Leyton-Brown. 2015. Polynomial-time computation of exact correlated equilibrium in compact games. *Games and Economic Behavior* 91 (2015), 347–359.

Christopher Kiekintveld, Manish Jain, Jason Tsai, James Pita, Fernando Ordóñez, and Milind Tambe. 2009. Computing optimal randomized resource allocations for massive security games. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1* (Budapest, Hungary) (AAMAS '09). International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 689–696.

Daphne Koller, Nimrod Megiddo, and Bernhard von Stengel. 1996. Efficient Computation of Equilibria for Extensive Two-Person Games. *Games and Economic Behavior* 14, 2 (1996), 247–259.

Wouter M Koolen, Manfred K Warmuth, Jyrki Kivinen, et al. 2010. Hedging Structured Concepts. In *COLT*. Citeseer, 93–105.

Marc Lanctot, Vinícius Flores Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. 2017. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 4190–4203.

C. Marks. 2008. *No-regret learning and game-theoretic equilibria*. Ph.D. Dissertation. Brown University, Providence, RI.

Dustin Morrill, Ryan D’Orazio, Marc Lanctot, James R. Wright, Michael Bowling, and Amy R. Greenwald. 2021. Efficient Deviation Types and Learning for Hindsight Rationality in Extensive-Form Games. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*. PMLR, 7818–7828.

H. Moulin and J. P. Vial. 1978. Strategically zero-sum games: The class of games whose completely mixed equilibria cannot be improved upon. *Int. J. Game Theory* 7, 3–4 (sep 1978), 201–221.

J. von Neumann. 1928. Zur Theorie der Gesellschaftsspiele. *Math. Ann.* 100, 1 (1928), 295–320.

Christos H. Papadimitriou and Tim Roughgarden. 2008. Computing Correlated Equilibria in Multi-Player Games. *J. ACM* 55, 3 (2008).

Binghui Peng and Aviad Rubinstein. 2024. Fast Swap Regret Minimization and Applications to Approximate Correlated Equilibria. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing* (Vancouver, BC, Canada) (STOC 2024). Association for Computing Machinery, New York, NY, USA, 1223–1234.

I. Romanovskii. 1962. Reduction of a Game with Complete Memory to a Matrix Game. *Soviet Mathematics* 3 (1962).

Aviad Rubinstein. 2015. Inapproximability of Nash Equilibrium. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing* (Portland, Oregon, USA) (STOC '15). Association for Computing Machinery, New York, NY, USA, 409–418.

Aviad Rubinstein. 2016. Settling the Complexity of Computing Approximate Two-Player Nash Equilibria. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*. 258–265.

Eiji Takimoto and Manfred K Warmuth. 2003. Path kernels and multiplicative updates. *The Journal of Machine Learning Research* 4 (2003), 773–818.

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

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

Haifeng Xu. 2016. The Mysteries of Security Games: Equilibrium Computation Becomes Combinatorial Algorithm Design. In *Proceedings of the 2016 ACM Conference on Economics and Computation* (Maastricht, The Netherlands) (EC '16). Association for Computing Machinery, New York, NY, USA, 497–514.

Haifeng Xu, Fei Fang, Albert Jiang, Vincent Conitzer, Shaddin Dughmi, and Milind Tambe. 2014. Solving Zero-Sum Security Games in Discretized Spatio-Temporal Domains. *Proceedings of the AAAI Conference on Artificial Intelligence* 28, 1 (Jun. 2014).

Brian Hu Zhang, Ioannis Anagnostides, Gabriele Farina, and Tuomas Sandholm. 2024a. Efficient Φ -Regret Minimization with Low-Degree Swap Deviations in Extensive-Form Games. arXiv:2402.09670 [cs.GT]

Brian H. Zhang, Gabriele Farina, and Tuomas Sandholm. 2024b. Mediator Interpretation and Faster Learning Algorithms for Linear Correlated Equilibria in General Sequential Games. In *The Twelfth International Conference on Learning Representations*.

A Related work

Algorithms for computing equilibria can be classified broadly into three categories:

- *Polynomial-time algorithms* compute an *exact* equilibrium in time polynomial in the input game size. Note that exact equilibria only make sense when the game has rational utilities, otherwise we settle for ϵ -approximate equilibria in time polynomial in $\log(1/\epsilon)$ and the size of the game.
- *Fully polynomial-time approximation schemes (FPTASs)* compute ϵ -approximate equilibria in time polynomial in $1/\epsilon$ and the size of the input game.
- *Polynomial-time approximation schemes (PTASs)* compute ϵ -approximate equilibria in time that is polynomial in the size of the input game, for every fixed $\epsilon > 0$ —however, they might in general have an exponential dependence on $1/\epsilon$.

Nash equilibria are known to be PPAD-complete, thus ruling out any polynomial-time algorithm for them (Daskalakis et al., 2009). Additionally, approximating a Nash equilibrium, even for a constant ϵ is known to also be PPAD-complete for n -player games (Rubinstein, 2015) and to require quasi-polynomial time for 2-player games, assuming ETH for PPAD (Rubinstein, 2016), thus ruling out any PTAS or FPTAS algorithm.

The complexity landscape is significantly more favorable for the case of correlated equilibrium (CE). Specifically, Hart and Mas-Colell (2000); Blum and Mansour (2007) gave efficient no-regret

dynamics (minimizing the so-called *internal regret*) that, if used by all players in a game, can be used to compute an ϵ -approximate CE in normal-form games in time polynomial in the size of the game and $1/\epsilon$. This constitutes the first FPTAS for the computation of CEs. Finally, Papadimitriou and Roughgarden (2008) gave a centralized algorithm that exactly computes a CE in a concisely represented normal-form game in polynomial time in the size of the game. This was the first polynomial-time algorithm for CEs.

The complexity of CE is not settled for extensive-form games (EFGs), and its determination remains a major unresolved question in the field (von Stengel and Forges, 2008; Farina and Pipis, 2023). An advance in this direction was provided very recently by the breakthrough results from two concurrent works of Dagan et al. (2024) and Peng and Rubinstein (2024), which provide a PTAS for CE in extensive-form games, though it remains unclear whether a polynomial-time algorithm or even an FPTAS exist.

Due to the conjectured intractability of computing a normal-form CE in EFGs, researchers have come up with other notions of equilibrium (von Stengel and Forges, 2008; Morrill et al., 2021; Farina and Pipis, 2023; Zhang et al., 2024b) that lie on a spectrum of Φ -equilibria. Φ -equilibria provide a generalization of CE that ranges from least hindsight-rational (coarse correlated equilibria), to maximum hindsight rational (CE), depending on the size of the set of behavior transformation Φ considered by the players. One of the most notable and natural notions of sequential rationality is that of the extensive-form correlated equilibrium (EFCE) (von Stengel and Forges, 2008). The EFCE was shown to be efficiently computable exactly (Huang and von Stengel, 2008) using a method similar to the Ellipsoid Against Hope of Papadimitriou and Roughgarden (2008). Additionally, it was shown that there exist efficient regret dynamics (minimizing the so called trigger regret) that can be used to compute an ϵ -approximate equilibrium by Farina et al. (2022a). This also gives an FPTAS for EFCE.

Currently, the highest notion of rationality that admits an FPTAS is the recently defined linear-deviation correlated equilibrium (LCE) (Farina and Pipis, 2023), which subsumes previous equilibrium notions such as EFCE. Farina and Pipis (2023) proved that there exist efficient regret dynamics (minimizing linear-swap regret) that can converge to an LCE; the polynomial complexity was then later improved by Zhang et al. (2024b). They however left open the question as to whether there exists a polynomial time algorithm that can compute an exact LCE. We resolve this open question in this paper, showing that the exact computability of equilibria in EFGs extends up to the linear-deviation correlated equilibrium.

Relationship with work on combinatorially-structured games. Our algorithmic framework in Section 3 can be used to compute exact min-max equilibria in bilinear games when one of the players has an exponentially large action space and we can only use a good-enough-response oracle—a weaker notion than the best-response oracle.

In light of applications related to Machine Learning and Deep Learning, there has recently been increased renewed interest in games having exponential (or even infinite) action spaces. For example, Assos et al. (2023) propose regret dynamics that can converge to approximate coarse correlated equilibria in infinite (nonparametric) games when a suitable notion of dimension of the game (Littlestone and fat-threshold) is bounded. Dagan et al. (2024) generalize this result even further proving that there also exist regret-dynamics in these cases that can converge in approximate correlated equilibria. Interestingly, recent breakthroughs in Large Language Models have inspired work on “language-based” games that typically have an enormous number of strategies (FAIR et al., 2022; Gemp et al., 2024).

However, the interest in games involving large strategy spaces is by no means a recent phenomenon. For instance, the community of security games has traditionally been interested in the problem of computing Stackelberg equilibria for games where one player (the leader) can possibly have exponentially many strategies (Kiekintveld et al., 2009; Xu, 2016). Another notable example is that of the Colonel Blotto game, which involves exponentially large strategy sets in both players (Ahmadinejad et al., 2019). Finally, work on *learning* in combinatorially-structured games has found applications to online optimization on combinatorial domains such as EFG strategy spaces and flow polytopes (Farina et al., 2022b; Koolen et al., 2010; Takimoto and Warmuth, 2003).

Even though our framework for computing min-max equilibria might have some resemblance to some of the methods in these papers, to the best of our knowledge, in all of the past cases the algorithms cleverly exploit the special combinatorial structure inherent in security games and usually involve

reducing the dimensionality of the strategy space as a first step. Our framework on the other hand, might allow for applications where the large decision set of a player is not amenable to some kind of smaller-dimensional representation.

Finally, we remark that an interesting recurring theme in games with large strategy spaces is that they often assume some kind of best-response oracle access.

Relationship with work based on best-response oracle access to games Using best-response oracles is a ubiquitous technique for learning in games or equilibrium computation, starting from the foundational method of fictitious play (Brown, 1951) where players apply a best-response oracle at every round to respond to the empirical frequency of play of their opponent. Best-response oracles (or variants thereof) have additionally been used in security games (Ahmadinejad et al., 2019; Xu et al., 2014), in bilinear games (Gidel et al., 2017), in efficient learning on polytopes (Chakrabarti et al., 2024), in the computation of well-supported equilibria in bilinear games (Goldberg and Marmolejo-Cossío, 2021), in the PSRO for Reinforcement Learning (Lanctot et al., 2017), in infinite games (Assos et al., 2023; Dagan et al., 2024). We remark however that in this paper we do *not* use a best-response oracle in our algorithms. Rather, we use a weaker notion that we coin “Good-Enough-Response” (GER) oracle. In certain cases (such as the computation of Φ -equilibria in Section 4) it is critical to relax the requirement for a best-response oracle because no such oracle can be constructed unless $P = NP$ (Theorem 4.7).

B Further preliminaries on linear programs

Here, we introduce some more specialized Lemmas that will be useful for proving the main results of Section 3. The first one concerns feasibility sets of the form we use in our main LPs and shows that they are, in fact, polytopes and that their facet-complexity is properly bounded.

Lemma B.1. *Let \mathbf{A} be a matrix and*

$$\mathcal{P} = \left\{ \mathbf{u} \in \mathcal{U} \mid \max_{\mathbf{q} \in \mathcal{Q}} \mathbf{q}^\top \mathbf{A} \mathbf{u} \leq c \right\}.$$

If the following conditions hold

- \mathcal{U} be a rational polyhedron with facet-complexity at most φ ,
- \mathcal{Q} be the convex hull of a set of finitely many points $V(\mathcal{Q}) = \{\hat{\mathbf{q}}_1, \dots, \hat{\mathbf{q}}_K\}$,
- the inequality $(\hat{\mathbf{q}}^\top \mathbf{A}) \mathbf{u} \leq c$ has encoding length at most φ for all vertices $\hat{\mathbf{q}} \in V(\mathcal{Q})$.

Then the set \mathcal{P} is a rational polytope with facet-complexity at most φ .

Proof. First we show that $\mathcal{P} = \mathcal{P}'$, where

$$\mathcal{P}' := \left\{ \mathbf{u} \in \mathcal{U} \mid (\hat{\mathbf{q}}^\top \mathbf{A}) \mathbf{u} \leq c \ \forall \hat{\mathbf{q}} \in V(\mathcal{Q}) \right\},$$

is the polytope defined by finitely many inequality constraints, corresponding to the vertices of \mathcal{Q} . Combining this with the assumption that each such inequality has encoding length at most φ , the result follows immediately. It remains to prove the desired set equality:

- Case $\mathcal{P} \subseteq \mathcal{P}'$:

$$\mathbf{u} \in \mathcal{P} \implies \mathbf{u} \in \mathcal{U}, \max_{\mathbf{q} \in \mathcal{Q}} \mathbf{q}^\top \mathbf{A} \mathbf{u} \leq c \implies \mathbf{u} \in \mathcal{U}, (\hat{\mathbf{q}}^\top \mathbf{A}) \mathbf{u} \leq c \ \forall \hat{\mathbf{q}} \in V(\mathcal{Q}),$$

where the last implication follows from the fact that for all $\hat{\mathbf{q}} \in V(\mathcal{Q}) \subseteq \mathcal{Q}$,

$$(\hat{\mathbf{q}}^\top \mathbf{A}) \mathbf{u} \leq \max_{\mathbf{q} \in \mathcal{Q}} \mathbf{q}^\top \mathbf{A} \mathbf{u} \leq c.$$

- Case $\mathcal{P} \supseteq \mathcal{P}'$: By definition, any point $\mathbf{q} \in \mathcal{Q}$ can be written as the convex combination of all vertices $\mathbf{q} = \sum_i^K \lambda_i \hat{\mathbf{q}}_i$. Thus, we have

$$\mathbf{u} \in \mathcal{P}' \implies \mathbf{u} \in \mathcal{U}, (\hat{\mathbf{q}}^\top \mathbf{A}) \mathbf{u} \leq c \ \forall \hat{\mathbf{q}} \in V(\mathcal{Q}) \implies \mathbf{u} \in \mathcal{U}, \max_{\mathbf{q} \in \mathcal{Q}} \mathbf{q}^\top \mathbf{A} \mathbf{u} \leq c,$$

where the last implication holds because for any $\mathbf{q} \in \mathcal{Q}$,

$$\mathbf{q}^\top \mathbf{A} \mathbf{u} = \sum_i^K \lambda_i \hat{\mathbf{q}}_i^\top \mathbf{A} \mathbf{u} \leq \sum_i^K \lambda_i c.$$

This concludes the proof. \square

We also mention a result in the spirit of the classical Farkas lemma. We use this result in to prove our main Theorem of Section 3.

Lemma B.2 (Generalized Farkas lemma). *Let $\mathcal{X} \subset \mathbb{R}^M, \mathcal{Y} \subset \mathbb{R}^N$ be convex compact sets. Then exactly one of the following two statements is true.*

1. *There exists $\mathbf{x} \in \mathcal{X}$ such that $\min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0$.*
2. *There exists $\mathbf{y} \in \mathbb{R}_+ \mathcal{Y}$ such that $\max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \leq -1$.*

Proof. I) We first show that (1) and (2) cannot be true simultaneously. Assume otherwise and let $\hat{\mathbf{x}} \in \mathcal{X}, \hat{\mathbf{y}} \in \mathbb{R}_+ \mathcal{Y}$ be values that satisfy (1) and (2) respectively. Since $\hat{\mathbf{y}}$ belongs to the conic hull of \mathcal{Y} it must be $\hat{\mathbf{y}} = k\mathbf{y}'$ for some $k > 0$ and $\mathbf{y}' \in \mathcal{Y}$. Thus,

$$\max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \hat{\mathbf{y}} \leq -1 \implies \max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y}' \leq -\frac{1}{k}.$$

Additionally, it holds

$$0 \leq \min_{\mathbf{y} \in \mathcal{Y}} \hat{\mathbf{x}}^\top \mathbf{A} \mathbf{y} \leq \hat{\mathbf{x}}^\top \mathbf{A} \mathbf{y}' \leq \max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y}' \leq -\frac{1}{k},$$

which is a contradiction. Thus, the statements (1) and (2) cannot be true simultaneously.

II) We now proceed to prove that when (2) is false then (1) must be true. We begin by showing that (2) being false implies that for any $\gamma > 0$ there does not exist any $\mathbf{y} \in \mathbb{R}_+ \mathcal{Y}$ such that $\max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \leq -\gamma$. Suppose otherwise; then $\mathbf{y}' = \mathbf{y}/\gamma$ is a multiple of an element of $\mathbb{R}_+ \mathcal{Y}$ and thus $\mathbf{y}' \in \mathbb{R}_+ \mathcal{Y}$. Furthermore,

$$\max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y}' = \frac{1}{\gamma} \max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \leq -1,$$

which is a contradiction because we have assumed that (2) is false. It directly follows that

$$\min_{\mathbf{y} \in \mathcal{Y}} \max_{\mathbf{x} \in \mathcal{X}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0.$$

By the minimax theorem it also holds

$$\max_{\mathbf{x} \in \mathcal{X}} \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0 \implies \exists_{\mathbf{x} \in \mathcal{X}} : \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{x}^\top \mathbf{A} \mathbf{y} \geq 0$$

and thus, statement (1) is true.

III) Finally, we need to prove the inverse direction; when (1) is false then (2) must be true. This is trivial because if (2) was false, then by II) we would have that (1) is true, which contradicts the assumption. \square

C Omitted proofs from Section 3

Theorem 3.1. *If the following hold*

1. $\mathcal{X} \subset \mathbb{R}^M, \mathcal{Y} \subset \mathbb{R}^N$ are rational polytopes and \mathcal{Y} has facet-complexity at most φ ,
2. we have access to a separation oracle $SEP_{\mathcal{Y}}$ for \mathcal{Y} and a good-enough-response oracle GER ,
3. the encoding length of $\mathbf{x}^\top \mathbf{A}$ is at most φ for all GER oracle responses and all vertices of \mathcal{X} ,

Algorithm 2: Separation oracle for the ellipsoid method on (D)

Input: Separation oracle $\text{SEP}_{\mathbb{R}_+\mathcal{Y}}$ for $\mathbb{R}_+\mathcal{Y}$, and a good-enough-response oracle GER .

Output: A separating hyperplane \mathbf{c} for \mathbf{y} in (D) , and a corresponding vector \mathbf{x} from GER , if it exists.

```
if  $\text{SEP}_{\mathbb{R}_+\mathcal{Y}}$  deems that  $\mathbf{y}$  is in  $\mathbb{R}_+\mathcal{Y}$  then
  | Set  $(\mathbf{x}, \mathbf{c})$  to the output  $(\mathbf{x}, \mathbf{A}^\top \mathbf{x})$  of  $\text{GER}(\mathbf{y})$ ;
else
  | Set  $\mathbf{c}$  to the separating hyperplane output by  $\text{SEP}_{\mathbb{R}_+\mathcal{Y}}$ ;
  |  $\mathbf{x} = \emptyset$ ;
end
```

then Algorithm 1 runs in $\text{poly}(N, \varphi)$ time, performs $L = \text{poly}(N, \varphi)$ oracle calls, and computes an exact solution \mathbf{x}^* of (P) that is a mixture of at most N oracle responses. In particular, the encoding length of \mathbf{x}^* depends polynomially on the encoding length of the GER oracle responses.

Proof. First, we can assume without loss of generality that there exists a dimension \emptyset in all $\mathbf{y} \in \mathcal{Y}$ such that $\mathbf{y}[\emptyset] = 1$. Otherwise, it is always possible to augment these vectors with an extra dimension before applying the next steps of the algorithm. This allows us to convert the given separation oracle $\text{SEP}_{\mathcal{Y}}$ into a new separation oracle $\text{SEP}_{\mathbb{R}_+\mathcal{Y}}$ for $\mathbb{R}_+\mathcal{Y}$ that we can then use to construct the general oracle of Algorithm 2.

The first step of the algorithm is to execute the ellipsoid method on (D) . Using the assumptions of the theorem in Lemma B.1 with $\mathcal{U} = \mathbb{R}_+\mathcal{Y}$ and $\mathcal{Q} = \mathcal{X}$, it follows that (D) is a polytope and has facet-complexity at most φ . Additionally, by the fact that (P) is feasible (it has an equilibrium with $\text{OPT} = 0$) and by Lemma B.2 it follows that (D) must be infeasible.

To execute the ellipsoid method on (D) would then mean, in the language of Grötschel et al. (1993), to solve the Strong Nonemptiness Problem for (D) using the strong separation oracle of Algorithm 2. To this end, we use the algorithm from Theorem 6.4.1 of Grötschel et al. (1993). This algorithm works for any polyhedron, even if it is not bounded or full-dimensional, as might be the case here. To do that, it might execute the central-cut ellipsoid method more than once, but never more than N times. In our case, we already know that (D) is infeasible and thus, the algorithm terminates after N executions of the central-cut ellipsoid and concludes that (D) is infeasible.

Since the central-cut ellipsoid method is an oracle-polynomial algorithm that is executed N times in polyhedra of facet-complexity at most φ , the whole process runs in polynomial time and performs a polynomial number of separation oracle calls. To calculate the exact number L of oracle calls, we note that the algorithm in Grötschel et al. (1993, Theorem 6.4.1) initializes the central-cut ellipsoid method with

$$R = 2^{O(N^2\varphi)} \text{ and } \epsilon = 2^{-O(N^5\varphi)},$$

while the central-cut method terminates in $O(N \log(1/\epsilon) + N^2 \log R)$ iterations (Grötschel et al., 1993, Theorem 3.2.1). Combining these with the fact that the central-cut ellipsoid method is repeated N times, we get that the number of oracle calls is $L = O(N^7\varphi)$.

Next, note that (D') is comprised of constraints coming from GER oracle responses, which by Lemma B.1 gives that the facet-complexity of (D') must also be at most φ . By going through the same process as before, the algorithm will reach the same conclusion after executing the central-cut ellipsoid method N times; (D') is infeasible.

Finally, by the infeasibility of (D') and Lemma B.2, it follows that (P') must be feasible. An equivalent way to express (P') is

$$\begin{aligned} & \text{find } \mathbf{a} \\ & \text{s.t. } \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{a}^\top (\mathbf{X}^\top \mathbf{A}) \mathbf{y} \geq 0 \\ & \quad \mathbf{a} \in \Delta^L, \end{aligned}$$

where Δ^L is the L -dimensional simplex and $\mathbf{X} = [\mathbf{x}_1 \mid \dots \mid \mathbf{x}_L]$ is a matrix with the GER oracle responses as its columns.

Applying Lemma B.1 for $\mathcal{U} = \Delta^L$ and $\mathcal{Q} = \mathcal{Y}$ we conclude that (P') describes a polytope

$$\mathcal{P} = \left\{ \mathbf{a} \in \Delta^L \mid \min_{\mathbf{y} \in \mathcal{Y}} \mathbf{a}^\top (\mathbf{X}^\top \mathbf{A}) \mathbf{y} \geq 0 \right\}$$

of encoding length at most $L \text{poly}(\varphi)$. This is because, for any vertex $\hat{\mathbf{y}} \in V(\mathcal{Y})$, the inequality $\mathbf{a}^\top (\mathbf{X}^\top \mathbf{A}) \hat{\mathbf{y}} \geq 0$ has L coefficients, each of which having encoding length $\text{poly}(\varphi)$. This can be solved in polynomial time by any known linear programming method. Even better, it is possible to compute a basic feasible solution of this LP, which will have at most N non-zero entries and thus the final solution $\mathbf{x}^* = \mathbf{X}\mathbf{a}$ will be a mixture of at most N oracle responses. \square

D Omitted proofs from Section 4

Lemma 4.1. *Let G be a polyhedral game with pure strategy set Π_p for every player $p \in [n]$. Additionally, let Φ_p be a set of linear transformations for every $p \in [n]$. If $\mathbf{x} \in \mathcal{X} = \Delta(\Pi_1 \times \dots \times \Pi_n)$ and $\mathbf{y} = (\phi_1, \dots, \phi_n) \in \mathcal{Y} = \Phi_1 \times \dots \times \Phi_n$ then*

$$\mathbf{x}^\top \mathbf{U} \mathbf{y} = \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})].$$

Proof. As we discussed, each linear transformation ϕ_p can be viewed as a $d_p \times d_p$ transformation matrix \mathbf{B}_p . We denote the matrix entries with $\mathbf{B}_p[b, a]$. In particular, if $\mathbf{s}'_p = \phi_p(\mathbf{s}_p)$, we have $\mathbf{s}'_p[b] = \sum_a \mathbf{B}_p[b, a] \mathbf{s}_p[a]$. Then for any $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} = (\phi_1, \dots, \phi_n) \in \mathcal{Y}$ we have

$$\begin{aligned} \mathbf{x}^\top \mathbf{U} \mathbf{y} &= \sum_s x_s \sum_p \left(u_p(s) - \sum_{a \in [d_p]} \sum_{b \in [d_p]} \mathbf{B}_p[b, a] \mathbf{s}_p[a] u_p(\mathbf{1}_b, \mathbf{s}_{-p}) \right) \\ &= \sum_p \sum_s x_s \left(u_p(s) - u_p \left(\sum_{b \in [d_p]} \mathbf{1}_b \sum_{a \in [d_p]} \mathbf{B}_p[b, a] \mathbf{s}_p[a], \mathbf{s}_{-p} \right) \right) \\ &= \sum_p \sum_s x_s \left(u_p(s) - u_p \left(\sum_{b \in [d_p]} \mathbf{1}_b \mathbf{s}'_p[b], \mathbf{s}_{-p} \right) \right) \\ &= \sum_p \sum_s x_s (u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})) \\ &= \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})], \end{aligned}$$

where in the second equality we have used the multi-linearity of the utilities. \square

Now we are ready to present the good-enough-response oracle that will allow us to develop an efficient algorithm for computing exact Φ -equilibria in polyhedral games. As a general backbone, this Lemma follows the constructive proof that Papadimitriou and Roughgarden (2008) did for the CE existence result of Hart and Schmeidler (1989) and, crucially, it produces pure strategies using the idea of Jiang and Leyton-Brown (2015).

Lemma D.1 (GER oracle for Φ -equilibria). *For every $\mathbf{y} \in \mathcal{Y} = \Phi_1 \times \dots \times \Phi_n$ there exists a pure strategy profile $\mathbf{s} \in \Pi_1 \times \dots \times \Pi_n$ such that $\mathbf{1}_s^\top \mathbf{U} \mathbf{y} \geq 0$. Furthermore, such a strategy profile alongside with the vector $\mathbf{1}_s^\top \mathbf{U}$ can be computed efficiently, provided that the game satisfies the polynomial utility gradient property (Assumption 4.2) and there exists a polynomial-time separation oracle for every \mathcal{A}_p .*

Proof. As we have discussed, we can denote all linear transformations ϕ_p using a matrix \mathbf{B}_p such that $\phi_p(\mathbf{x}_p) = \mathbf{B}_p \mathbf{x}_p$.

First note that there always exists a fixed-point of any linear strategy transformation ϕ_p ; this follows from Brouwer's fixed-point theorem and the fact that these transformations are continuous maps of a

Algorithm 3: Purified GER oracle

Input: Polyhedral game G of n players, $\mathbf{y} \in \mathcal{Y} = \Phi_1 \times \cdots \times \Phi_n$, separation oracles $\text{SEP}_{\mathcal{A}_p}$ for all $p \in [n]$.

Output: A pure strategy profile $s \in \Pi_1 \times \cdots \times \Pi_n$ such that $\mathbf{1}_s^\top \mathbf{U} \mathbf{y} \geq 0$.

Compute a product distribution \mathbf{x} s.t. $\mathbf{x}^\top \mathbf{U} \mathbf{y} = 0$ by finding fixed points $\mathbf{x}_p = \phi_p(\mathbf{x}_p)$ for all $p \in [n]$;

for $p \in [n]$ **do**

 Find a set of $k \leq d_p + 1$ vertices $\{\hat{\mathbf{s}}_p^{(1)}, \dots, \hat{\mathbf{s}}_p^{(k)}\} \subset V(\mathcal{A}_p)$ s.t. $\mathbf{x}_p = \sum_{i=1}^k \lambda_i \hat{\mathbf{s}}_p^{(i)}$;

 Set \mathbf{s}_p^* to the vertex $\hat{\mathbf{s}}_p^{(i)}$ that satisfies $\mathbf{x}_{(p \rightarrow \mathbf{s}_p^*)}^\top \mathbf{U} \mathbf{y} \geq 0$;

 Set \mathbf{x} to be $\mathbf{x}_{(p \rightarrow \mathbf{s}_p^*)}$;

end

Finally, \mathbf{x} must correspond to a pure strategy profile s ;

compact convex set \mathcal{A}_p to itself. Additionally, since the transformations are linear we can always efficiently compute a fixed-point of any transformation by solving the following LP:

$$\begin{aligned} & \text{find } \mathbf{x}_p \\ & \text{s.t. } \mathbf{B}_p \mathbf{x}_p = \mathbf{x}_p \\ & \mathbf{x}_p \in \mathcal{A}_p \end{aligned}$$

that can be solved in polynomial time using the ellipsoid method with the given separation oracle for \mathcal{A}_p .

Next, let us restrict our attention only to product distributions $\mathbf{x} \in \Delta(\Pi_1 \times \cdots \times \Pi_n)$. In this case it will be $\mathbf{x}_s = \mathbf{x}_{-p}(\mathbf{s}_{-p}) \mathbf{x}_p(\mathbf{s}_p)$ for all pure strategy profiles s , which gives

$$\begin{aligned} \mathbf{x}^\top \mathbf{U} \mathbf{y} &= \sum_p \sum_{\mathbf{s}_{-p}} \sum_{\mathbf{s}_p \in \Pi_p} x_{-p}(\mathbf{s}_{-p}) x_p(\mathbf{s}_p) [u_p(\mathbf{s}_p, \mathbf{s}_{-p}) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})] \\ &= \sum_p \sum_{\mathbf{s}_{-p}} x_{-p}(\mathbf{s}_{-p}) u_p([\mathbf{x}_p - \phi_p(\mathbf{x}_p)], \mathbf{s}_{-p}) \\ &= \sum_p \mathbf{g}_p(\mathbf{x}_{-p}) \cdot [\mathbf{x}_p - \phi_p(\mathbf{x}_p)], \end{aligned} \tag{1}$$

where $\mathbf{x}_p \in \mathcal{A}_p$ is the marginal distribution for player p represented as a point of the polyhedral strategy set. In the second equality we have used the multi-linearity of $u_p(\cdot, \mathbf{s}_{-p})$ and the linearity of the transformations; $\sum_{\mathbf{s}_p} x_p(\mathbf{s}_p) \phi_p(\mathbf{s}_p) = \phi_p(\mathbf{x}_p)$. It directly follows from the last equality that if we set each marginal distribution equal to the corresponding fixed-point $\mathbf{x}_p = \phi_p(\mathbf{x}_p)$, we get a product distribution \mathbf{x} such that $\mathbf{x}^\top \mathbf{U} \mathbf{y} = 0$.

Now, it remains to find a way to extract the desired pure strategy profile s from this product distribution. We follow a similar procedure to the purification technique used by Jiang and Leyton-Brown (2015). Similar to their algorithm, we define $\mathbf{x}_{(p \rightarrow \mathbf{s}_p)}$ for a product distribution \mathbf{x} to be the product distribution in which player p plays pure action $\mathbf{s}_p \in \mathcal{A}_p$ and all other players act according to \mathbf{x}_{-p} . Additionally, note that since \mathcal{A}_p is a polytope, it must hold

$$\mathbf{x}_p = \sum_{\mathbf{s}_p \in V(\mathcal{A}_p)} \lambda_{\mathbf{s}_p} \mathbf{s}_p$$

for some convex combination $\{\lambda_{\mathbf{s}_p} \geq 0 \mid \sum_{\mathbf{s}_p} \lambda_{\mathbf{s}_p} = 1\}$. By the product distribution structure, it is easy to see that for every player $p \in [n]$,

$$\mathbf{x}^\top \mathbf{U} \mathbf{y} = \sum_{\mathbf{s}_p \in V(\mathcal{A}_p)} [\mathbf{x}_{(p \rightarrow \mathbf{s}_p)}^\top \mathbf{U} \mathbf{y}] \lambda_{\mathbf{s}_p} \tag{2}$$

The algorithm of Jiang and Leyton-Brown (2015) iterates over all players and for each player p they search over all its pure strategies and find one, \mathbf{s}_p^* , for which $\mathbf{x}_{(p \rightarrow \mathbf{s}_p^*)}^\top \mathbf{U} \mathbf{y} \geq 0$. Such a pure

strategy must always exist because (2) represents a convex combination over all vertices (a.k.a. pure strategies). However, in our case we cannot iterate over all pure strategies for a player because they might be exponentially many.

To make this procedure general for all polyhedral games, we observe that by Carathéodory's theorem there must always exist a subset $\{\hat{s}_p^{(1)}, \dots, \hat{s}_p^{(k)}\} \subset V(\mathcal{A}_p)$ of at most $k \leq d_p + 1$ vertices of \mathcal{A}_p that satisfy

$$\mathbf{x}_p = \sum_{i=1}^k \lambda_i \hat{s}_p^{(i)}$$

for some convex combination represented with $\lambda_1, \dots, \lambda_k$. Thus, we can follow the same procedure as before but this time only search over k vertices instead of all (possibly exponentially many) vertices of \mathcal{A}_p . The complete algorithm is shown in Algorithm 3.

This can be implemented in polynomial time because: (a) there exists an algorithmic version of Carathéodory's theorem Grötschel et al. (1993, Theorem 6.5.11) that only requires access to a separation oracle for \mathcal{A}_p , and (b) Assumption 4.2 allows us to compute $\mathbf{x}_{(p \rightarrow s_p^*)}^\top \mathbf{U} \mathbf{y}$ in polynomial time for any product distribution $\mathbf{x}_{(p \rightarrow s_p^*)}$, as is evident from (1). \square

Theorem 4.4. *Let G be a polyhedral game (Definition 2.4) of n players and $\{\Phi_p\}$ be a collection of polytopes corresponding to sets of linear strategy transformations that map every strategy set \mathcal{A}_p to itself. Additionally, let $N = \sum_p d_p^2$. Assume that*

- there exist polynomial-time separation oracles for \mathcal{A}_p and Φ_p ,
- G satisfies the polynomial utility gradient property (Assumption 4.2),
- ψ is an upper bound on the facet-complexity of every \mathcal{A}_p and Φ_p ,
- $\log u$ is the maximum encoding length of the utilities of G .

Then there exists an algorithm that computes an exact $\{\Phi_p\}$ -equilibrium of G in time $\text{poly}(N, \log u, \psi)$ and performs $\text{poly}(N, \log u, \psi)$ number of calls to all the separation oracles. Additionally, the equilibrium is represented as a convex combination of at most N pure strategy profiles.

Proof. The set $\mathcal{X} = \Delta(\Pi_1 \times \dots \times \Pi_n)$ is trivially a rational polytope and the set $\mathcal{Y} = \Phi_1 \times \dots \times \Phi_n \subset \mathbb{R}^N$ is the Cartesian product of rational polytopes, hence a rational polytope. Furthermore, we can directly construct a polynomial-time separation oracle for \mathcal{Y} by calling the separation oracles for each one of the sets Φ_p . Additionally, every row of the \mathbf{U} matrix has N entries, each with encoding length at most $2 \log u$. Using the good-enough-response oracle from Lemma D.1, each response $(\mathbf{x}_k, \mathbf{x}_k^\top \mathbf{U}) \in \mathcal{X} \times \mathbb{Q}^N$ corresponds to a pair of a pure strategy profile (vertex of \mathcal{X}) and a row of \mathbf{U} . Thus, each response has encoding length at most $2N \log u$. Set $\varphi = \max(2N \log u, \psi)$.

Now, we can apply Theorem 3.1, which gives us an algorithm running in $\text{poly}(N, \varphi)$ time and performing $\text{poly}(N, \varphi)$ oracle calls. Combining this with Lemma D.1, it follows that the total time complexity is $\text{poly}(N, \varphi) = \text{poly}(N, \log u, \psi)$. Finally, the optimal solution \mathbf{x}^* will be comprised of a mixture of N oracle responses. In other words, \mathbf{x}^* will be an exact $\{\Phi_p\}$ -equilibrium for the game G with probability mass on at most N pure strategy profiles. \square

Corollary 4.5 (Exact CE in normal-form games). *If a normal-form game G has the polynomial type and the polynomial expectation property, defined in Papadimitriou and Roughgarden (2008), then our algorithm computes an exact correlated equilibrium of G and runs in polynomial time in the size of the game.*

Proof. A normal-form game is a polyhedral game where every strategy set is a probability simplex. Additionally, the set Φ of all linear transformations in normal-form games is that of swap-deviations which is equivalent to the set of all stochastic matrices (Gordon et al., 2008). Both the sets of strategies and the sets of stochastic matrices can easily be represented as polytopes of bounded facet-complexity having polynomially many constraints. Finally, the polynomial utility gradient property

(Assumption 4.2) reduces to the polynomial expectation property in succinct normal-form games and the polynomial type property is implicitly satisfied (see Remark 4.3). Thus, all requirements of Theorem 4.4 are satisfied and we conclude that there exists a polynomial time algorithm for computing CEs in normal-form games. \square

Corollary 4.6 (Exact LCE computation). *There exists an algorithm that runs in $\text{poly}(N, \log u)$ time and computes an exact linear-deviation correlated equilibrium (LCE) in an extensive-form game.*

Proof. We apply Theorem 4.4 for the set of linear-swap deviations. Specifically, in Farina and Pipis (2023, Theorem 3.1) it is proved that:

- The set Φ_{LIN} of linear-swap deviations for a player p is a rational polytope.
- This polytope can be described using a polynomial number of equality constraints, which immediately implies the existence of an efficient separation oracle.
- Every constraint of the characterization has at most $|\Sigma_p|^2$ coefficients, each belonging to $\{0, 1, -1\}$. Thus, the facet-complexity of Φ_{LIN} must be $\psi = |\Sigma_p|^2$.

Finally, since the number of non-zero utilities are at most equal to the game tree size, it trivially follows that extensive-form games satisfy the polynomial utility gradient property (Assumption 4.2). It follows that there exists a polynomial time algorithm for computing LCEs. \square

Theorem 4.7 (Hardness of BR oracle). *It is NP-hard to construct a best-response oracle for the Correlator in the Correlator-Deviator game.*

Proof. A best-response oracle for the Correlator in the Correlator-Deviator game must respond with the optimal $\mathbf{x} \in \mathcal{X} = \Delta(\Pi_1 \times \dots \times \Pi_n)$ for any given $\mathbf{y} \in \mathcal{Y} = \Phi_1 \times \dots \times \Phi_n$. More precisely, we have to be able to compute

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}} \{ \mathbf{x}^\top \mathbf{U} \mathbf{y} \} = \arg \max_{\mathbf{x} \in \mathcal{X}} \left\{ \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})] \right\}$$

for all $\mathbf{y} \in \mathcal{Y}$.

To prove that this process is intractable it suffices to find a game and an equilibrium concept such that it is NP-hard to compute \mathbf{x}^* for at least one $\mathbf{y} \in \mathcal{Y}$. For the solution concept we choose the coarse-correlated equilibrium, in which the sets Φ_p consist of all constant (or external) deviations that output a fixed strategy $\phi_p(\mathbf{x}_p) = \mathbf{s}_p \in \Pi_p$ no matter the input strategy \mathbf{x}_p . For the game, we choose the SAT-game that was also used to prove the hardness of equilibrium selection for EFCE and LCE in extensive-form games (von Stengel and Forges, 2008; Farina and Pipis, 2023). The exact details are not important for our purposes, but we will only use the fact that any SAT instance can be encoded in a 2-player extensive-form game using a polynomial-time reduction. In this game, any pure strategy profile $s \in \Pi_1 \times \dots \times \Pi_n$ with social welfare (sum of players' utilities) equal to 2 corresponds to a satisfying assignment for the SAT instance, while any other strategy profile has social welfare at most $2(1 - 1/n)$. Thus, there exists a Nash equilibrium (and hence, a CCE) corresponding to a pure strategy profile that has maximum social welfare.

Before we proceed, we augment the SAT-game by adding an extra decision point for both players at the beginning of the game that asks whether they want to play. If both players respond “Yes” then the game continues as normal, otherwise –if at least one player responds “No” – the game ends and both players get a 0 payoff. We denote the pure strategy of the “No” answer from player 1 with s_1^N and the “No” answer from player 2 with s_2^N .

Now, to prove the desired result consider a Correlator-Deviator game applied to the computation of a CCE in the above augmented SAT-game. Assume that there exists a polynomial-time best-response oracle for this game that returns a solution of polynomial size in the representation of the game (and hence, the size of the SAT instance). Then, we can use it to best-respond to $\mathbf{y} = (\phi_1, \phi_2)$ for

$\phi_1(\mathbf{x}_1) = \mathbf{s}_1^N$ and $\phi_2(\mathbf{x}_2) = \mathbf{s}_2^N$. Specifically, we have

$$\begin{aligned}
\mathbf{x}^* &= \arg \max_{\mathbf{x} \in \mathcal{X}} \left\{ \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s) - u_p(\phi_p(\mathbf{s}_p), \mathbf{s}_{-p})] \right\} \\
&= \arg \max_{\mathbf{x} \in \mathcal{X}} \left\{ \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s)] - \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(\mathbf{s}_p^N, \mathbf{s}_{-p})] \right\} \\
&= \arg \max_{\mathbf{x} \in \mathcal{X}} \left\{ \sum_p \mathbb{E}_{s \sim \mathbf{x}} [u_p(s)] \right\}.
\end{aligned}$$

In other words, the BR oracle returns in this case a distribution \mathbf{x}^* over pure strategy profiles with maximum social welfare. Since the BR oracle computes a polynomially-sized \mathbf{x}^* in polynomial-time, we can uncover a pure strategy profile of maximum social welfare that corresponds, by construction of the SAT-game, to a satisfying assignment of the SAT instance. We conclude that constructing a best-response oracle in the Correlator-Deviator game corresponding to the computation of a CCE in extensive-form games is NP-hard. \square

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: **[Yes]**

Justification: We include our contributions in the abstract as well as in the "Contributions" paragraph of the introduction.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: **[Yes]**

Justification: We include a discussion of the limitations in the final section of the paper, alongside with a list of related open problems.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: All theorems, lemmas and corollaries of the paper explicitly mention the set of assumptions needed. Additionally, all complete proofs are included in the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [NA]

Justification: Since this is a purely theoretical paper, exploring the tractability of computing high-precision equilibria in games (and giving a polynomial-time algorithm of rather high complexity), it does not include any experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: The paper does not include experiments requiring code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [NA]

Justification: The paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [NA]

Justification: The paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer “Yes” if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [NA]

Justification: The paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: Our research conforms with the NeurIPS Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: Our paper is a theoretical investigation of the tractability of computing high-precision equilibria in polyhedral games and does not suggest a way to deploy new technologies. Thus, it does not have any immediate societal impact.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: Our paper concerns theoretical results of equilibrium computation in games and poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification: Our paper does not use existing assets.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We do not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.