Sum-of-Squares Meets Nash: Lower Bounds for Finding Any Equilibrium

Pravesh K. Kothari
Princeton University and the Institute for Advanced Study
Princeton, NJ, USA

Ruta Mehta
University of Illinois
Urbana-Champaign, IL, USA

ABSTRACT

Computing Nash equilibrium (NE) in two-player game is a central question in algorithmic game theory. The main motivation of this work is to understand the power of sum-of-squares method in computing equilibria, both exact and approximate. Previous works in this context have focused on hardness of approximating "best" equilibria with respect to some natural quality measure on equilibria such as social welfare. Such results, however, do not directly relate to the complexity of the problem of finding *any* equilibrium.

In this work, we propose a framework of *roundings* for the sum-of-squares algorithm (and convex relaxations in general) applicable to finding approximate/exact equilbria in two player bimatrix games. Specifically, we define the notion of *oblivious roundings with verification oracle* (OV). These are algorithms that can access a solution to the degree d SoS relaxation to construct a list of candidate (partial) solutions and invoke a *verification* oracle to check if a candidate in the list gives an (exact or approximate) equilibrium.

This framework captures most known approximation algorithms in combinatorial optimization including the celebrated semi-definite programming based algorithms for Max-Cut, Constraint-Satisfaction Problems, and the recent works on SoS relaxations for Unique Games/Small-Set Expansion, Best Separable State, and many problems in unsupervised machine learning.

Our main results are strong unconditional lower bounds in this framework. Specifically, we show that for $\varepsilon = \Theta(1/poly(n))$, there's no algorithm that uses a $o(n)$-degree SoS relaxation to construct a $2^{o(n)}$-size list of candidates and obtain an ε-approximate NE. For some constant ε, we show a similar result for degree $o(\log(n))$ SoS relaxation and list size $n^{o(\log(n))}$. Our results can be seen as an unconditional confirmation, in our restricted algorithmic framework, of the recent Exponential Time Hypothesis for PPAD.

Our proof strategy involves constructing a family of games that all share a common sum-of-squares solution but every (approximate) equilibrium of any game is far from every equilibrium of any other game in the family (in either player's strategy). Along the way, we strengthen the classical unconditional lower bound against enumerative algorithms for finding approximate equilibria due to Daskalakis-Papadimitriou and the classical hardness of computing equilibria due to Gilbow-Zemel.

CCS CONCEPTS

• **Theory of computation → Exact and approximate computation of equilibria**;

KEYWORDS

lower bounds, PPAD, equilibria, approximate equilibria, sum-of-squares, oblivious rounding

ACM Reference Format:
Pravesh K. Kothari and Ruta Mehta. 2018. Sum-of-Squares Meets Nash: Lower Bounds for Finding Any Equilibrium. In *Proceedings of 50th Annual ACM SIGACT Symposium on the Theory of Computing (STOC'18).* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3188745.3188892

1 INTRODUCTION

The algorithmic complexity of computing Nash equilibria in two-player finite games has been one of the principle focus of algorithmic game theory since its inception, *e.g.,* see [1, 2, 7, 18, 20, 21, 36, 39, 40]. In a series of remarkable papers in 2006, the problem was shown to be PPAD-complete [18, 20]. This motivates the question of relating PPAD to better understood complexity classes such as NP. Such an eventuality, however, appears unlikely: since PPAD is a sub-class of TFNP ("total function NP"), i.e., every two player game has an equilibrium strategy (the "solution"), finding NE cannot be NP-hard unless NP=co-NP [34]. A line of research has focused on understanding the complexity of finding "special" equilibria, like one with good payoffs, in both exact and approximate cases [17, 19, 22, 27]. However, since these "decision formulations" are NP-complete, it is unclear if they relate to the hardness of PPAD. Indeed, as such, these problems could be fundamentally different from the basic question of finding "any" equilibrium [34].

On the flip side, existing efficient algorithms for the problem are known only in specialized situations [1, 16, 21, 29, 32] and barring a few notable exceptions[2, 26, 41], the algorithms involve enumeration over a restricted search space. Perhaps the most well known result of this flavor is the celebrated QPTAS of Lipton-Markakis-Mehta (LMM) [32] to find constant approximate NE by enumerating

STOC'18, June 25–29, 2018, Los Angeles, CA, USA

ACM ISBN 978-1-4503-5559-9/18/06…$15.00
https://doi.org/10.1145/3188745.3188892

over a search space of size $n^{O(\log(n))}$. It was a central open question to improve upon this algorithm or to understand why such an improvement is unlikely. In a recent breakthrough, Rubinstein [39] made a leap in this direction and showed a tantalizingly optimal lower bound assuming a strong conjecture informally known as the Exponential Time Hypothesis (ETH) for PPAD.

This picture of PPAD motivates the two general research goals:

- Can we find a formalization for "enumeration-cost" of an algorithm for computing equilibria and in particular, obtain non-enumerative algorithms for the problem?
- Can we find alternate, preferably unconditional ways to gain evidence of hardness for computing Nash equilibria in two player games?

The central goal of this paper is to present a framework based on rounding of a strong convex relaxation for computing Nash equilibria - the sum of squares semidefinite programming hierarchy. This framework captures the best known algorithms for computing approximate equilibria in two player games and lower bounds against it could point towards inherent hardness of the algorithmic problem of finding equilibria.

In the recent years, sum of squares (SoS) method has yielded a powerful general purpose tool for algorithm design extending spectral methods and linear programming. It captures and yields new state of the art algorithms for a number of problems in worst-case algorithm design [4, 5, 10, 12, 14, 25], breaks integrality gaps for fundamental hard problems in combinatorial optimization [8, 35], and has been remarkably successful in algorithm design for average case problems [11, 13, 33].

As a result, lower bounds against sum of squares method have become credible indicators of computational hardness and have been successfully used as such in cases where evidence based on the standard machinery of NP-hardness seems inadequate. For example, arguably the strongest evidence of hardness ("computational vs statistical complexity gaps") for many fundamental average-case problems such as Planted Clique, Refuting Random CSPs, Maximizing Random Polynomials (equivalently, Tensor Principal Component Analysis) [3, 9, 28, 30, 37] comes from strong lower bounds against the sum of squares algorithm.

This leads to the concrete questions of interest to us in this work.

Question 1.1. Could the sum of squares method yield faster algorithms for computing equilibria?

A priori, we are faced with the same issue that plagues relating TFNP problems to NP. Standard notions of hardness against SoS method (and more generally, convex relaxations) are integrality gaps. However, the problem of finding an equilibrium using sum of squares naturally involves only a feasibility program and since every game has an equilibrium, the program is always feasible!

In the light of this, we propose a framework that addresses the existence of *rounding* algorithms - i.e. procedures that map solutions to the relaxation to a true solution. Formally, we define the notion of *oblivious rounding with verification oracle* (OV). Using the natural quadratic formulation for (approximate) NE (see (1.1) and (1.2)), OV algorithms can access a solution to its degree d SoS relaxation to construct a list of candidate (partial) solutions, and invoke a *verification* oracle to check if a candidate in the list gives an (exact or approximate) equilibrium. Oblivious rounding algorithms (without verification oracles) were first studied in the context of lower bounds for the problem of maximizing social welfare in [?].

While there are a couple notable exceptions, most known approximation algorithms in combinatorial optimization based on rounding semi-definite/linear programs are captured within our framework. This includes, for instance, the algorithms of Goemans-Williamson for Max-Cut[24], the Arora-Rao-Vazirani[5] algorithm for Sparsest Cut, the Raghavendra-Steurer[38] algorithm for any constraint satisfaction problem along with the recent rounding algorithms developed for the sum-of-squares method for problems such as unique games/small-set-expansion [14], non-negative polynomial optimization[10] and many problems arising in machine learning.

For two-player, n-strategy games, the work of [26] can be reinterpreted as showing a oblivious rounding algorithm for finding approximate equilibria in quasi-polynomial time. In particular, this implies that SoS method combined with oblivious rounding gives a *non-enumerative* algorithm for the probelm. We also provide a different proof of this result in this paper.

The main results of this paper are optimal unconditional lower bounds in our framework for finding approximate/exact equilibria in two player games. Specifically, for $\varepsilon = \Theta(1/n)$, there's no algorithm that uses a $o(n)$-degree SoS relaxation to construct a $2^{o(n)}$-size list of candidates and obtain an ε-approximate NE in two player games with maximum payoff bounded above by 1. For constant ε, we show a similar result for degree $o(\log(n))$-degree SoS relaxation and list size $n^{o(\log(n))}$. We note that, both the lower bounds are against stronger verification oracles – take strategy profile of any one player and checks if the other player has a strategy, which together with the former gives an approximate NE. Our results can be seen as unconditional confirmation, in our restricted algorithmic framework, of the recent Exponential Time Hypothesis for PPAD [6].

To obtain the lower bounds, our proof strategy involves constructing a family of games that all share a common sum-of-squares solution but every (approximate) equilibrium of any game is far from every equilibrium of any other game in the family (in either player's strategy). To get hardness against the stronger oracle mentioned about, we show that even equilibrium strategies of individual players are far apart across games. Along the way, we strengthen the classical unconditional lower bound against enumerative algorithms for finding approximate equilibria due to Daskalakis-Papadimitriou [21]. In addition, we obtain a gap version of the Gilbow-Zemel reduction from k-Clique to high-payoff Nash equilibrium.

In what follows, we will formalize a restricted class of algorithms for computing equilibria using semi-definite programming hierarchies, show that the model is strong enough to capture general, known techniques for computing equilibria, and establish strong lower bounds for the model.

1.1 Games, Equilibria and Quadratic Feasibility Formulation

A game G between Alice and Bob with n strategies for both is described by two payoff matrices R and C in $[-1,1]^{n\times n}$,[1] where Alice plays rows and Bob plays columns. Players may randomize and play a strategy from $\Delta_n = \{x \in [0,1]^n \mid \sum_{i=1}^n x_i = 1\}$. We write $\boldsymbol{e}_i \in \Delta_n$, for the i^{th} pure strategy. When Alice plays $\boldsymbol{x} \in \Delta_n$ and Bob $\boldsymbol{y} \in \Delta_n$, the expected *payoffs* respectively are $\boldsymbol{x}^\top R\boldsymbol{y}$ and $\boldsymbol{x}^\top C\boldsymbol{y}$. Clearly, given an oponent's strategy, there is always a pure strategy giving the maximum payoff. At Nash equilibrium (NE) no player gains by deviating unilaterally. That is,

$$\boldsymbol{x}^\top R\boldsymbol{y} \geqslant \boldsymbol{e}_i^\top R\boldsymbol{y} \text{ for all } i \in [n]; \quad \boldsymbol{x}^\top C\boldsymbol{y} \geqslant \boldsymbol{x}^\top C\boldsymbol{e}_j \text{ for all } j \in [n]. \tag{1.1}$$

At an ε-approximate NE (ε-NE) no player gains by more than ε by deviating unilaterally:

$$\boldsymbol{x}^\top R\boldsymbol{y} \geqslant \boldsymbol{e}_i^\top R\boldsymbol{y} - \varepsilon \text{ for all } i \in [n]; \quad \boldsymbol{x}^\top C\boldsymbol{y} \geqslant \boldsymbol{x}^\top C\boldsymbol{e}_j - \varepsilon \text{ for all } j \in [n]. \tag{1.2}$$

1.2 Sum-of-Squares Method and Pseudo-equilibria

The sum of squares method is a sequence of increasingly tight SDP relaxations for a system of polynomial inequalities. We provide a brief overview of this method specialized to computing (exact/approximate) equilibria here and point the reader to the lecture notes [15] for further details on method and its applications. Central to the sum of squares method is the notion of a pseudo-distribution that generalizes probability distributions.

Definition 1.2 (Pseudo-distribution)**.** A degree d pseudo-distribution is a finitely supported signed measure $\tilde{\mu}$ on $\mathbb{R}^n$ such that the associated linear functional (pseudo-expectation) $\tilde{\mathbb{E}}$ that maps any function $f : \mathbb{R}^n \to \mathbb{R}$ to $\tilde{\mathbb{E}}[f] = \sum_{x:\tilde{\mu}(x)\neq 0} \tilde{\mu}(x) f(x)$ satisfies the following properties:

(1) **Normalization:** $\tilde{\mathbb{E}}[1] = 1$ or equivalently, $\sum_{x:\tilde{\mu}(x)\neq 0} \tilde{\mu}(x) = 1$, and
(2) **Positivity:** $\tilde{\mathbb{E}}[q^2] \geqslant 0$ for every degree $\leqslant d/2$ polynomial q on $\mathbb{R}^n$.

A pseudo-distribution $\tilde{\mu}$ is said to satisfy a polynomial inequality constraint $q \geqslant 0$ if for every polynomial p, $\tilde{\mathbb{E}}[p^2 q] \geqslant 0$ whenever $\deg(p^2 q) \leqslant d$.

[1] Normalizing payoffs to lie in $[-1, 1]$ is without loss of generality.

It's not hard to show that any degree ∞ pseudo-distribution is a probability distribution on $\mathbb{R}^n$. We will now use pseudo-distributions in order to define a relaxation of the notion of equilibrium, which we call, *pseudo-equilibrium.*

Definition 1.3 (Degree d Pseudo-equilibrium)**.** Given a two player game (R, C), a degree d pseudo-equilibrium for (R, C) is a degree d pseudo-distribution on strategy profiles $(\boldsymbol{x}, \boldsymbol{y})$ satisfying the degree 2 polynomial inequality constraints in (1.1). A degree d, ε-approximate pseudo-equilibrium is a degree d pseudo-distribution over strategy profiles $(\boldsymbol{x}, \boldsymbol{y})$ satisfying the degree 2 polynomial constraints in (1.2).

1.3 The Model: Oblivious Rounding with Verification

In this section, we present our algorithmic framework based on restricted roundings of sum-of-squares relaxation for polynomial feasibility problems. Let $\Psi(P)$ be a system of polynomial inequalities in $x \in \mathbb{R}^n$ parameterized by a problem instance $P \in \mathbb{R}^m$. For example, the natural quadratic programming formulation of (1.1) for finding exact NE is a system of degree 2 polynomial inequalities over $\Delta_n \times \Delta_n$ parameterized by the payoff matrices R, C that encode instances of two-player games. And similarly, that of (1.2) is for finding ε-NE.

We can now define two algorithmic models that use sum-of-squares relaxations to find solutions to polynomial feasibility program $\Psi(P)$.

Definition 1.4 (Oblivious Rounding)**.** Let $\Psi(P)$ be a polynomial feasibility problem parameterized by instances P. A degree d oblivious rounding algorithm for $\Psi(P)$ takes input a degree d pseudo-expectation satisfying the constraints in $\Psi(P)$ and outputs a point x that satisfies $\Psi(P)$.

In particular, the rounding algorithm is a function only of the degree d pseudo-expectation satisfying $\Psi(P)$, and is independent of the instance P itself. While this notion of rounding might appear restrictive, many of the famous algorithms in combinatorial optimization happen to be oblivious roundings and examples include the Goemans-Williamson algorithm for Max-Cut, the Arora-Rao-Vazirani algorithm for Sparsest-Cut, the Barak-Raghavendra-Steurer and Guruswami-Sinop algorithm for various 2-CSPs among others.

There are also notable exceptions such as the Facility Location algorithm of [31], the generic rounding algorithm for Raghavendra and Steurer for constraint satisfaction [38]. This list in fact can be expanded as most of the recent works on roundings sum-of-squares relaxations for various polynomial optimization problems are *not* oblivious. The CSP rounding algorithm of Raghavendra-Steurer and the recent algorithms based on rounding the SoS hierarchy for various polynomial optimization problem all can be seen as generating a small list of candidate solutions based on the SoS solution and then arguing that one of the candidates is a good solution to the problem at hand.

Inspired by these algorithms, we generalize our framework to allow generating a list of candidates based on the SoS solution to yield the central algorithmic model of interest in this work. This allows us to capture all known rounding algorithms mentioned above with the exception of the facility location algorithm of [31]. For every instance of the underlying problem encoded by $\Psi(P)$, we define the *verification oracle* as follows.

Definition 1.5 (Verification Oracle)**.** Let $\Psi(P)$ be a polynomial feasibility problem over $\mathbb{R}^n$. A verification oracle for P takes input a $y \in \mathbb{R}^n$ and accepts if y satisfies every polynomial inequality in $\Psi(P)$ and rejects otherwise.

In our setting, the above oracle essentially takes a strategy profile $(\boldsymbol{x}, \boldsymbol{y})$ and accepts if it is an NE (ε-*NE*), otherwise rejects. Note that in (1.1) and (1.2), if either $\boldsymbol{x}$ or $\boldsymbol{y}$ are given then the problem becomes linear and can be checked for feasibility in polynomial time. Motivated from this we allow the following stronger oracle, and prove lower bounds against it.

Definition 1.6 (Verification Oracle with Partial Input)**.** Let $\Psi(P)$ be a polynomial feasibility problem over $\mathbb{R}^n$. A verification oracle for P takes input a $y \in \mathbb{R}^k$ for $k \leqslant n$ such that replacing y in $\Psi(P)$ results in a linear system of inequalities. Then the oracle accepts if this linear system is feasible and rejects otherwise.

Definition 1.7 (Oblivious Rounding with Verification Oracle)**.** Let $\Psi(P)$ be a polynomial feasibility problem parameterized by instances P. A degree d, q query oblivious rounding with verification algorithm for $\Psi(P)$ takes input a degree d pseudo-expectation satisfying the constraints in $\Psi(P)$ and outputs a list of q points $x_1, x_2, \ldots, x_q$ is accepted by the Verification Oracle 1.6.

1.4 Our Results

We are now ready to describe our main results.

Our first main result is an exponential lower bound for finding inverse polynomially approximate equilibria in two player games.

Theorem 1.8. *Suppose there exists a degree d, q-query algorithm for computing $\Theta(1/n^4)$-approximate equilibria in two-player games with n strategies. Then, either $d = \Omega(n)$ or $q = 2^{\Omega(n)}$.*

Our second result is a similar statement for constant-approximate equilibria.

Theorem 1.9. *Suppose there exists a degree d, q-query algorithm for computing ε-approximate equilibria in two-player games with n strategies for some small $\varepsilon = \Theta(1)$. Then, either $d = \Omega(\log(n))$ or $q = n^{\Omega(\log(n))}$.*

We note that the SoS hardness obtained in [26] is for "special" NE, namely with *high-payoffs*, and therefore is fundamentally different than our setting of "any" equilibrium precisely because of the PPAD vs NP barrier. Along the way in showing the above results, we improve upon two prior works. First, we prove a "gapped" version of the classical result by Gilbow and Zemel.

Corollary 1.10. *Given a graph G and the k-Clique (or k-independent set) problem, a game (R, C) can be constructed such that: (i) if G has a k-Clique then the game has a NE with payoff $\delta > 0$. (ii) if G has no k-Clique then all ε approximate of G has payoffs at most $(\delta - \varepsilon)$ for $\varepsilon \leqslant O(1/n^2)$.*

Second, we extend the result of Daskalakis and Papadimitriou on lower bounds for oblivious rounding methods. Their construction, in the language of this paper, shows lower bounds on the number of calls to Oracle 1.5 [21]. We extend it to lower bound against the stronger Oracle 1.6.

Corollary 1.11. *For a given game (R, C) with n strategies, if an algorithm queries Oracle 1.6 q times and outputs an $O(1)$-NE of (R, C), then $q = n^{\Omega(\log(n))}$.*

Remark 1.12. A stronger notion of approximation, known as *well-supported* NE (WNE), has also been widely studied. Since every ε-WNE of a game is also its ε-NE, the above lower bounds holds for WNE as well. Digging further into our results, it turns out that we can improve the approximation ratios significantly while keeping the lower bounds as is, *e.g.*, exponential lower bounds for $O(1/n^2)$ approximation. Furthermore, all results can be extended to even more powerful oracle that is only possible for WNE , namely take supports of both players' strategies and output if there is an ε-WNE with the given support. To keep the exposition clean and intuitive, we stick to NE in this paper.

In what follows we discuss main technical ideas behind each of our results. We refer the reader to the publicly available full version of the paper for detailed proofs of the results.

2 LOWER BOUNDS AGAINST OBLIVIOUS ROUNDING WITH VERIFICATION ORACLE

We show two lower bound results matching state the art upper bounds, namely exponential lower bound for $(1/poly(n))$-NE, and quasi-polynomial lower bound for $O(1)$-NE. Both are against the stronger algorithm defined in 1.7 that uses SoS relaxation + Verification Oracle. Recall that, the Verification Oracle 1.6 allows verification with partial solution if the remaining system is linear. This is much stronger than checking if given vector is a solution (Oracle 1.5), and to the best of our knowledge no lower bounds are known against it even without access to SoS relaxations.

At a high-level we want to construct games such that their high-degree pseudo-equilibrium (Definition 1.3) do not give any information about the actual Nash equilibrium. Not only no "rounding" on the pseudo-equilibrium works, but also no enumeration and verification using the pseudo-equilibrium works. To achieve the latter we need to construct a large class of such games. Here large is *exponential* for $\varepsilon = 1/poly(n)$ and *quasi-polynomial* for $\varepsilon = O(1)$, and similarly *high-degree* correspond to degree $\Omega(n)$ and $\Omega(\log(n))$ respectively.

In addition to the notations introduced in Section 1, we will use the following notations.

Notations: We will use $[n]$ to denote set of numbers $\{1,\ldots,n\}$. Given an integer n and a subset $T \subseteq [n]$, define vector $\mathcal{U}_{n,T}$ to be a probability distribution of size n that is uniform on coordinates of T and zero everywhere else, and when $T=[n]$ we use the short-hand notation $\mathcal{U}_n$.

A strategy profile $(\boldsymbol{x},\boldsymbol{y})$ of game (R,C) with n strategies is called its ε-approximate well-supported NE (ε-WNE) iff each player plays a pure strategy with non-zero probability only if it gives maximum payoff upto ε. That is,

$$\text{for every } i \in [n],\ x_i > 0 \Rightarrow e_i^\top Ry \geqslant \max_k e_k^\top Ry - \varepsilon$$
$$\text{and} \quad y_i > 0 \Rightarrow x^\top Ce_i \geqslant \max_k x^\top Ce_k - \varepsilon. \quad (2.1)$$

In the next section we design an abstract construction that turns out to be powerful enough to prove both the lower bounds. This construction may be of independent interest.

2.1 Abstract Game Construction

Our construction of the hard games is based on combining a construction of games with integrality gap against Sum-of-Squares relaxations for computing equilibria that maximize social welfare, and games that are hard for enumeration based algorithms. Main difficulty in combining any two games is to handle new (unwanted) equilibria that may span on both. We abstract out the properties of these two kinds of hard games in this section, and then show how to combine them so that no "unwanted" equilibria are created. Our hardness results in the next two sections will invoke the general combining strategy from this section. Next we discuss main ideas.

- (ε,δ)-**SoSHard game.** A game (R,C) that has a degree t pseudo-equilibrium with high payoffs, say $\delta > 0$, but all its ε-WNE have total payoff at most $2(\delta-\varepsilon) > 0$.
- (ε,τ)-**EnumHard game.** A $K\times K$ game parameterized by a subset $S \subset [K]$ of pure strategies, say (R_S,C_S), such that all its ε-WNE strategies for each player individually are concentrated around (scaled) uniform distribution over S, namely $\mathcal{U}_{K,S}$. By *concentrated* we mean distance at most $(\tau\varepsilon)$ in the l_1 norm.
- **Family of subsets.** For a constant $\beta > 2$, a family $\mathcal{F}$ of $q(\beta)$ many subsets of $[K]$ such that for any two distinct $S,S' \in \mathcal{F}$, (scaled) uniform distributions $\mathcal{U}_{K,S}$ and $\mathcal{U}_{K,S'}$ over S and S' respectively are far apart, *i.e.,* at distance at least $(\beta\tau\varepsilon)$.

The SoSHard game above gives degree t SoS hardness of obtaining high payoff ε-WNE . While for $\beta > 2$, the family of EnumHard games constructed for each $S \in \mathcal{F}$ would give lower bound of $q(\beta)$-queries to the verification oracle to find ε-WNE . But, neither gives hardness for both together, and none for the ε-NE. The question is to combine the two such that we get hardness for ε-NE against SoS + enumeration and verification.

Let (R,C) be $N\times N$ game and (R_S,C_S) be $K\times K$ game, and let all their payoffs be in $[-1,1]$. We combine these two and construct a $(N+K)\times(N+K)$ game (R'_S,C'_S) such that it preserves the degree t pseudo-equilibrium of game (R,C) (completeness), and at the same time all its ε-NE are near the (scaled) uniform distribution over the subset S of $[K]$ (soundness). Again distances are measured in l_1 norm. We first describe the construction.

$$R' = \begin{bmatrix} R & -2 \\ \delta & R_S \end{bmatrix} \quad \text{and} \quad C' = \begin{bmatrix} C & \delta \\ -2 & C_S \end{bmatrix} \quad (2.2)$$

The completeness part is relatively easy. The soundness analysis is involved, and here we sketch a proof: First, we show that every ε-WNE strategy profile is supported only on last K strategies for both the players. And second, that each ε-NE is near the $(\mathbf{0}_N,\mathcal{U}_{K,S})$. Given the first claim, we get that every ε-WNE of game (R'_S,C'_S) is essentially ε-WNE of game (R_S,C_S), which, by property of EnumHard game, is at distance at most $(\tau\varepsilon)$ from $\mathcal{U}_{K,S}$. If an ε-NE of (R'_S,C'_S) is near one of its ε-WNE , then we will get the second claim using the triangle inequality. We can prove the former with quadratic loss in the approximation and linear loss in the distance.

To show the first claim, that every ε-WNE is supported on last K strategies for both the players, note first that either both players play some of first N strategies or neither plays them due to the (-2) block. Next, we show that if they do play first N strategies then both needs to put significant probability mass on them, $O(1)$ to be precise, or else (-2) will dominate. Since for any $i \in [N]$ the payoff from i from the second block of (-2) remains the same, the projection on first N strategies should give $O(\varepsilon)$-WNE of game (R,C). By property of the SoSHard game, the corresponding total payoff from this part of the game can be at most $2(\delta-O(\varepsilon))$. On the other hand, from any of the last K strategies, both players get at least δ payoff from the first block. Combining these insights, we show that at least one of them would want to deviate to playing only last K strategies, and then the other will also deviate to the same.

Putting everything together, we show the following theorem.

Theorem 2.1. *Given parameters ε, $\delta > \varepsilon$, and $\tau > 0$, (i) let (R,C) be a degree t, (ε,δ)-SoSHard game of dimension $N\times N$, and (ii) for an integer K, appropriately chosen β, and subsets $S_1,\ldots,S_{q(\beta)}$ of $[K]$ such that $||\mathcal{U}_{K,S_i} - \mathcal{U}_{K,S_j}||_1 > \beta\tau\varepsilon$, let (R_{S_i},C_{S_i}) be an (ε,τ)-EnumHard game for each $i \in \{1,\ldots,q(\beta)\}$.*

Then, for the family of games $\mathcal{F} = \{\mathcal{G}_i \mid \mathcal{G}_i$ is the game of (2.2) using (R,C) and $(R_{S_i},C_{S_i})\}$,
(1.) All G_i's have a common degree t pseudo-equilibrium.
(2.) For any pair of games $\mathcal{G}_i \neq \mathcal{G}_j \in \mathcal{F}$, their $O(\varepsilon^2)$-NE strategy sets of either players do not intersect.

The above theorem implies that if there exists degree d, q-query algorithm to find ε^2-NE in two player games, then $d = \Omega(t)$ and $q = \Omega(q(\beta))$.

2.2 $O(n^{\log(n)})$ Lower Bound for the Constant Approximation

To get quasi-polynomial lower bound for the constant approximation, namely Theorem 1.9, we prove the following theorem. In this section we discuss the main techniques, and refer the reader to the full version for the formal proof.

Theorem 2.2 (Quasi-polynomial Hardness for $O(1)$-NE). *For every n large enough, there's a family of $\Gamma = n^{\Omega(\log(n))}$ two-player games $\{G_i = (R_i, C_i)\}_{i=1}^{\Gamma}$ with n strategies and all payoffs in $[-1, 1]$ s.t.:*
(1.) ***Completeness:*** *All G_i's share a common degree $\Theta(\log(n))$ pseudo-equilibrium.*
(2.) ***Soundness:*** *There exists an $\varepsilon = O(1)$, such that for any i, j and any pair of ε-NE in G_i and G_j, say $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{x}', \boldsymbol{y}')$ respectively, $\boldsymbol{x} \neq \boldsymbol{x}'$ and $\boldsymbol{y} \neq \boldsymbol{y}'$.*

We will next show construction of SoSHard and EnumHard games for a constant $\varepsilon > 0$, degree $t = \Theta(\log(n))$, where the number of EnumHard games is $n^{\Omega(\log(n))}$. For the SoSHard game, we employ the recent reduction due to [22] to construct (R, C) via reduction from the *free game* to show ETH-hardness for maximizing social welfare in two-player games. We follow through their reduction and observe that it also immediately yields an SoS hard game using the generic method for reduction within SoS framework [42].

We remark that a recent work of [26] also shows SoS hardness for a certain *decision problem* associated with computing equilibria by working through the hardness reduction of [17]. However, their result does not prove SoS-hardness of maximizing social welfare. Since such a result would be of interest (beyond our application in obtaining hardness against rounding algorithms), we state it below in a more direct form. Indeed, it essentially gives SoSHard game construction up to the distinction between ε-WNE and ε-NE. However since ε-WNE are ε-NE the construction is valid.

Lemma 2.3 (SoSHard game for $O(1)$-NE). *There exists a game (R, C) for $R, C \in [-1, 1]^{N \times N}$ such that:* (1.) ***Completeness:*** *There's a degree $\Omega(\log(N))$, pseudo-equilibrium with payoffs $\geqslant 1$ for both players.* (2.) ***Soundness:*** *For every ε-NE $(\boldsymbol{x}, \boldsymbol{y})$ payoffs of both players is at most $(1 - \varepsilon)$ for $\varepsilon < 1/1200$.*

Construction of an EnumHard game is a bit tricky. A family of games with disjoint set of constant approximate NE was constructed by Daskalakis and Papadimitriou [21] to show hardness against oblivious rounding using Oracle 1.5. However, in all of their games, the second player's NE strategy set is the same, and therefore the hardness against the stronger Oracle 1.6 does not follow. We extend their construction to obtain an EnumHard game, and thereby strengthen their result to get hardness against oblivious rounding using Oracle 1.6.

In the construction of [21], m is taken to be $\binom{l}{l/2}$ for some even integer l. For every l sized subset $S \subset [m]$ an $m \times m$ game (A_S, B_S) is constructed. The guarantee is that at every ε-WNE , the row player plays strategies only from S, and in fact her strategy is at most (8ε) distance away from the $\mathcal{U}_{m,S}$ in the l_1 norm. Furthermore, for a given $\beta > 0$ there exists a family of at least $n^{(0.8-2\beta\varepsilon)\log(m)}$ many such subsets S, such that for any pair of subsets S, S', $||\mathcal{U}_{m,S} - \mathcal{U}_{m,S'}||_1 > \beta\varepsilon$.

Note the guarantees above are only for the row-player's NE strategies. To construct an EnumHard game, we need such guarantees for both the players. There does not seem any way to directly modify [21] construction to achieve this. Instead, is it possible to use two of the above games where in one of them the roles of row and column players are switched? The challenge here turns out to be to combine them in such a way that both players (almost) uniformly randomize between the two games at all constant approximate equilibrium. We show that the following $2m \times 2m$ game achieves the goal:

$$R_S = \begin{bmatrix} 2 + A_S & -2 \\ -2 & 2 + B_S^\top \end{bmatrix} \quad C_S = \begin{bmatrix} -2 + B_S & 2 \\ 2 & -2 + A_S^\top \end{bmatrix}.$$

Let $n = 2m$, $\varepsilon = O(1)$, $x_L = \sum_{i \leqslant m} x_i$, $y_L = \sum_{i \leqslant m} y_i$, $x_R = 1 - x_L$ and $y_R = 1 - y_R$. We first show that $(\frac{9}{19} - \frac{\varepsilon}{9}) \leqslant x_L, x_R, y_L, y_R \leqslant (\frac{10}{19} + \frac{\varepsilon}{9})$ at any ε-WNE $(\boldsymbol{x}, \boldsymbol{y})$ of the above game (R_S, C_S). Thus, both players significantly randomizes among the two games. Using this we show that both $((x_i)_{i \in [m]}, (y_i)_{i \in [m]})$ and $((y_i)_{i=(m+1)}^n, (x_i)_{i=(m+1)}^n)$ profiles after normalization are $O(\varepsilon)$-WNE of game (A_S, B_S). Therefore, from the result of [21] the normalized version of $\boldsymbol{x}^L = (x_i)_{i \in [m]}$ and $\boldsymbol{y}^R = (y_i)_{i=(m+1)}^n$ should be at distance $O(\varepsilon)$ from $\mathcal{U}_{m,S}$ in l_1 norm.

If we get such a guarantee for non-normalized versions of $\boldsymbol{x}^L$ and $\boldsymbol{y}^R$ w.r.t. $c\mathcal{U}_{m,S}$ for some fixed constant c, then we are done. However, due to at least $(2/19)$ gap between lower and upper bound on x_L and y_R, it is not possible to get $O(\varepsilon)$ gap between $\boldsymbol{x}^L$ or $\boldsymbol{y}^R$ and $c\mathcal{U}_{m,S}$ for any fixed constant c. To achieve this we show even tighter bounds on x_L and y_R, namely $(1/2 - \varepsilon) \leqslant x_L, y_R \leqslant (1/2 + \varepsilon)$. Now indeed the difference between upper and lower bound is $O(\varepsilon)$ as needed. Putting everything together we prove the following, and thereby construct a family of $n^{\Omega(\log(n))}$ many EnumHard games for an $\varepsilon = O(1)$.

Lemma 2.4. *For an $\varepsilon = O(1)$, a family $\mathcal{F}$ of $n^{0.75\log(n)}$ many subsets $S \subset [m]$ can be obtained so that: (i) for every $S \in \mathcal{F}$, for all ε-WNE $(\boldsymbol{x}, \boldsymbol{y})$ of game (R_S, C_S), $||\boldsymbol{x}^L - c\mathcal{U}_{m,S}||_1, ||\boldsymbol{y}^R - c\mathcal{U}_{m,S}||_1 \leqslant \tau\varepsilon$, where $\tau = O(1)$ and $c = 1/2$. (ii) for any pair of subsets $S, S' \in \mathcal{F}$, $||\mathcal{U}_{m,S} - \mathcal{U}_{m,S'}||_1 > \frac{18}{c}(\tau + 1)\tau\varepsilon$.*

Putting together the SoSHard game of Lemma 2.3 and EnumHard games of Lemma 2.4 using the construction of Section 2.1 we get Theorem 2.2.

2.3 Exponential Lower Bound for Inverse-polynomial Approximation

To get the exponential lower bound of Theorem 1.8, we prove the following theorem. We discuss the main techniques next, and refer the reader to the full version for the formal proof.

Theorem 2.5 (Exponential Lower Bound for NE). *For every n large enough, there's a family of $\Gamma = 2^{\Theta(n)}$ games $\{G_i = (A_i, B_i)\}_{i=1}^{\Gamma}$ with n pure strategies for both players and all payoffs in $[-1, 1]$ such that:*
(1.) ***Completeness:*** *There exists a degree $\Theta(n)$, shared pseudo-equilibrium for every G_i simultaneously.*
(2.) ***Soundness:*** *For any $i \neq j$ if $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{x}', \boldsymbol{y}')$ are $\Omega(1/n^4)$-NE of games G_i and G_j respectively then $\boldsymbol{x} \neq \boldsymbol{x}'$ and $\boldsymbol{y} \neq \boldsymbol{y}'$.*

In order to use the abstract game to show this, we construct an SoSHard game with degree $\Omega(n)$ pseudo-equilibrium with high payoff, while all its $O(1/n^2)$-WNE have low payoff. And a family of exponentially many EnumHard games with disjoint set of $O(1/n^2)$-WNE .

For the SoSHard game construction, we reduce the independent-set problem to a gap version of approximate Nash equilibrium. Given a graph $G = (V, E)$ on n vertices, and the problem of finding k-sized independent set (k-IS) in it, we construct a $(2n+1) \times (2n+1)$ game (R, C) as follows (inspired by Gilbow and Zemel [23]): Let $\gamma = 1/2$, $A = \mathbf{1}_{n \times n} - E + I_n$ where I is an identity matrix, $B = (k + \gamma) * I_n$, and $B' = (-4) * \mathbf{1}_{n \times n}$.

$$R = \begin{bmatrix} A & B & \multirow{2}{*}{$(-1)\mathbf{1}_{2n\times 1}$} \\ B' & \mathbf{0}_{n\times n} & \\ \multicolumn{2}{c}{(1+\frac{\gamma}{k})\mathbf{1}_{1\times 2n}} & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} A & B' & \multirow{2}{*}{$(1+\frac{\gamma}{k})\mathbf{1}_{2n\times 1}$} \\ B & \mathbf{0}_{n\times n} & \\ \multicolumn{2}{c}{(-1)\mathbf{1}_{2n\times 1}} & 1 \end{bmatrix}$$

We show that: (i) if G has a k sized independent set then game (R, C) has an equilibrium with payoff $(1 + \gamma/k)$, (ii) if G does not have an independent set of size k then all $(1/5k)$-WNE of game (R, C) gives payoff at most 1. For a fixed independent-set size k, the map from IS to NE in claim (i) is linear, *i.e.,* low degree. Therefore, using the generic methodology for reductions within SoS framework [42], degree t pseudo distribution of the independent set formulation maps to degree $\Omega(t)$ pseudo equilibrium of game (R, C). Known SoS hardness for independent set with a fixed $O(n)$ size [42] gives us $\Omega(n)$ pseudo-equilibrium even though no $O(n)$-IS exists.

An independent set can be formed or destroyed by deletion or addition of one edge. The main difficulty is to create the gap in payoffs for the bigger set of approximate NE. That is, to go between exact NE with good payoffs and *all* $(1/poly(n))$-NE having bad payoffs; note that a NE is also ε-NE.

The above construction of SoSHard game, and a family of $2^{\Omega(n)}$ many EnumHard games, together with Theorem 2.1 gives Theorem 2.5.

ACKNOWLEDGEMENT

P.K. thanks Avi Wigderson for helpful discussions, suggestions and comments during various stages of this work.

REFERENCES

[1] Tim Abbott, Daniel Kane, and Paul Valiant. 2005. On the complexity of two-player win-lose games. In *IEEE Annual Symposium on Foundations of Computer Science*. 113–122.

[2] Bharat Adsul, Jugal Garg, Ruta Mehta, and Milind Sohoni. 2011. Rank-1 Bimatrix Games: A Homeomorphism and a Polynomial Time Algorithm. In *ACM Symposium on the Theory of Computing (STOC)*. 195–204.

[3] Sarah R. Allen, Ryan O'Donnell, and David Witmer. 2015. How to refute a random CSP. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science—FOCS 2015*. IEEE Computer Soc., Los Alamitos, CA, 689–708.

[4] Sanjeev Arora, Boaz Barak, and David Steurer. 2015. Subexponential algorithms for unique games and related problems. *J. ACM* 62, 5 (2015), Art. 42, 25. https://doi.org/10.1145/2775105

[5] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2010. $O(\sqrt{\log n})$ approximation to sparsest cut in $\tilde{O}(n^2)$ time. *SIAM J. Comput.* 39, 5 (2010), 1748–1771. https://doi.org/10.1137/080731049

[6] Yakov Babichenko, Christos Papadimitriou, and Aviad Rubinstein. 2016. Can Almost Everybody be Almost Happy?. In *Innovations in Theoretical Computer Science*. 1–9.

[7] Yakov Babichenko and Aviad Rubinstein. 2017. Communication complexity of approximate Nash equilibria. In *ACM Symposium on the Theory of Computing (STOC)*.

[8] Boaz Barak, Fernando G. S. L. Brandão, Aram W. Harrow, Jonathan Kelner, David Steurer, and Yuan Zhou. 2012. Hypercontractivity, sum-of-squares proofs, and their applications [extended abstract]. In *STOC'12—Proceedings of the 2012 ACM Symposium on Theory of Computing*. ACM, New York, 307–326. https://doi.org/10.1145/2213977.2214006

[9] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin. 2016. A Nearly Tight Sum-of-Squares Lower Bound for the Planted Clique Problem. *CoRR* abs/1604.03084 (2016).

[10] Boaz Barak, Jonathan A. Kelner, and David Steurer. 2014. Rounding sum-of-squares relaxations. In *STOC*. ACM, 31–40.

[11] Boaz Barak, Jonathan A. Kelner, and David Steurer. 2015. Dictionary learning and tensor decomposition via the sum-of-squares method [extended abstract]. In *STOC'15—Proceedings of the 2015 ACM Symposium on Theory of Computing*. ACM, New York, 143–151.

[12] Boaz Barak, Pravesh K. Kothari, and David Steurer. 2017. Quantum Entanglement, Sum-of-Squares and the Log-Rank Conjecture. *Symposium on Theory of Computing, STOC* (2017).

[13] Boaz Barak and Ankur Moitra. 2016. Noisy Tensor Completion via the Sum-of-Squares Hierarchy. In *COLT (JMLR Workshop and Conference Proceedings)*, Vol. 49. JMLR.org, 417–445.

[14] Boaz Barak, Prasad Raghavendra, and David Steurer. 2011. Rounding semidefinite programming hierarchies via global correlation. In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science—FOCS 2011*. IEEE Computer Soc., Los Alamitos, CA, 472–481. https://doi.org/10.1109/FOCS.2011.95

[15] Boaz Barak and David Steurer. [n. d.]. Proofs, beliefs, and algorithms through the lens of sum-of-squares. ([n. d.]). Note for the course co-taught by Boaz Barak and Pablo Parrilo at Harvard-MIT, and David Steurer and Pravesh Kothari at Princeton.

[16] Siddharth Barman. 2015. Approximating Nash equilibria and dense bipartite subgraphs via an approximate version of Caratheodory's theorem. In *ACM Symposium on the Theory of Computing (STOC)*. 361–369.

[17] Mark Braverman, Young Kun Ko, and Omri Weinstein. 2015. Approximating the best Nash equilibrium in $nO(logn)$-time breaks the exponential time hypothesis. In *ACM-SIAM Annual Symposium on Discrete Algorithms (SODA)*. 970–982.

[18] X. Chen, X. Deng, and S.-H. Teng. 2009. Settling the complexity of computing two-player Nash equilibria. *Journal of the ACM* 56(3) (2009).

[19] Vincent Conitzer and Tuomas Sandholm. 2008. New complexity results about Nash equilibria. *Games and Economic Behavior* 63, 2 (2008), 621–641.

[20] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou. 2009. The Complexity of Computing a Nash Equilibrium. *SIAM J. Comput.* 39(1) (2009), 195–259.

[21] Constantinos Daskalakis and Christos H. Papadimitriou. 2009. On oblivious PTAS's for Nash equilibrium. In *ACM Symposium on the Theory of Computing (STOC)*. 75–84.

[22] Argyrios Deligkas, John Fearnley, and Rahul Savani. 2016. Inapproximability Results for Approximate Nash Equilibria. In *12th International Conference Web and Internet Economics (WINE)*. 29–43.

[23] I. Gilboa and E. Zemel. 1989. Nash and correlated equilibria: Some complexity considerations. *Games Econ. Behav.* 1 (1989), 80–93.

[24] Michel X. Goemans and David P. Williamson. 2001. Approximation algorithms for MAX-3-CUT and other problems via complex semidefinite programming. In *STOC*. ACM, 443–452.

[25] Venkatesan Guruswami and Ali Kemal Sinop. 2011. Lasserre Hierarchy, Higher Eigenvalues, and Approximation Schemes for Quadratic Integer Programming with PSD Objectives. *Electronic Colloquium on Computational Complexity (ECCC)* 18 (2011), 66.

[26] Aram W. Harrow, Anand V. Natarajan, and Xiaodi Wu. 2016. Tight SoS-degree bounds for approximate Nash equilibria. In *Computational Complexity Conference (CCC)*.

[27] Elad Hazan and Robert Krauthgamer. 2011. How hard is it to approximate the best Nash equilibrium? *SIAM J. Comput.* 40, 1 (2011), 79–91.

[28] Samuel B. Hopkins, Jonathan Shi, and David Steurer. 2015. Tensor principal component analysis via sum-of-square proofs. In *COLT (JMLR Workshop and Conference Proceedings)*, Vol. 40. JMLR.org, 956–1006.

[29] Ravi Kannan and Thorsten Theobald. 2010. Games of Fixed Rank: A Hierarchy of Bimatrix Games. *Economic Theory* 42, 1 (2010), 157–174. Preliminary version appeared in SODA'07, and available at arXiv:cs/0511021 since 2005.

[30] Pravesh K. Kothari, Ryuhei Mori, Ryan O'Donnell, and David Witmer. 2017. Sum of squares lower bounds for refuting any CSP. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*. 132–145. https://doi.org/10.1145/3055399.3055485

[31] Shi Li. 2013. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Inf. Comput.* 222 (2013), 45–58. https://doi.org/10.1016/j.ic.2012.01.007

[32] Richard J. Lipton, Evangelos Markakis, and Aranyak Mehta. 2003. Playing Large Games Using Simple Strategies. In *ACM Conference on Electronic Commerce*. 36–41.

[33] Tengyu Ma, Jonathan Shi, and David Steurer. 2016. Polynomial-time Tensor Decompositions with Sum-of-Squares. *CoRR* abs/1610.01980 (2016).

[34] N. Megiddo and C. H. Papadimitriou. 1991. On Total Functions, Existence Theorems and Computational Complexity. *Theor. Comput. Sci.* 81, 2 (1991), 317–324.

[35] Ryan O'Donnell and Yuan Zhou. 2012. Approximability and proof complexity. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, Philadelphia, PA, 1537–1556.

[36] C. Papadimitriou. 1994. On the complexity of the parity argument and other inefficient proofs of existence. *J. Comput. System Sci.* 48(3) (1994), 498–532.

[37] Prasad Raghavendra, Satish Rao, and Tselil Schramm. 2016. Strongly Refuting Random CSPs Below the Spectral Threshold. *CoRR* abs/1605.00058 (2016).

[38] Prasad Raghavendra and David Steurer. 2009. How to round any CSP. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2009)*. IEEE Computer Soc., Los Alamitos, CA, 586–594. https://doi.org/10.1109/FOCS.2009.74

[39] Aviad Rubinstein. 2016. Settling the complexity of computing approximate two-player Nash equilibria. In *IEEE Annual Symposium on Foundations of Computer Science*. 258–265.

[40] Rahul Savani and Bernhard von Stengel. 2006. HardâĂŘtoâĂŘSolve Bimatrix Games. *Econometrica* 74, 2 (2006), 397–429.

[41] Haralampos Tsaknakis and Paul Spirakis. 2007. An optimization approach for approximate Nash equilibria. In *Web and Internet Economics (WINE)*. 42–56.

[42] Madhur Tulsiani. 2009. CSP gaps and reductions in the lasserre hierarchy. In *STOC*. ACM, 303–312.