SIAM J. COMPUT.
Vol. 40, No. 1, pp. 79–91

HOW HARD IS IT TO APPROXIMATE THE BEST NASH EQUILIBRIUM?*

ELAD HAZAN† AND ROBERT KRAUTHGAMER‡

Abstract. The quest for a polynomial-time approximation scheme (PTAS) for Nash equilibrium in a two-player game, which emerged as a major open question in algorithmic game theory, seeks to circumvent the PPAD-completeness of finding an (exact) Nash equilibrium by finding an approximate equilibrium. The closely related problem of finding an equilibrium maximizing a certain objective, such as social welfare, was shown to be NP-hard [Gilboa and Zemel, *Games Econom. Behav.*, 1 (1989), pp. 80–93]. However, this NP-hardness is unlikely to extend to approximate equilibria, since the latter admits a quasi-polynomial time algorithm [Lipton, Markakis, and Mehta, in *Proceedings of the 4th ACM Conference on Electronic Commerce*, ACM, New York, 2003, pp. 36–41]. We show that this optimization problem, namely, finding in a two-player game an approximate equilibrium achieving a large social welfare, is unlikely to have a polynomial-time algorithm. One interpretation of our results is that a PTAS for Nash equilibrium (if it exists) should not extend to a PTAS for finding the best Nash equilibrium. Technically, our result is a reduction from the notoriously difficult problem in modern combinatorics, of finding a planted (but hidden) clique in a random graph $G(n, 1/2)$. Our reduction starts from an instance with planted clique size $O(\log n)$. For comparison, the currently known algorithms are effective only for a much larger clique size $\Omega(\sqrt{n})$.

Key words. game theory, Nash equilibrium, hidden clique, logarithmically-restricted NP

AMS subject classifications. 91A80, 68R05

DOI. 10.1137/090766991

1. Introduction.

Computational aspects of equilibrium concepts (and in particular of Nash equilibrium) have seen major advances over the last few years, both from the side of algorithms and in terms of computational complexity (namely, completeness and hardness results). Perhaps the most celebrated result in this area [5, 8] (see also [9]) proves that computing a Nash equilibrium in a finite game with two players is PPAD-complete. Consequently, a weaker notion of ε-approximate Nash equilibrium, or in short an ε-equilibrium, was suggested, and the following has emerged as a central open question:

Is there a PTAS for Nash equilibrium?

Recall that the acronym PTAS stands for polynomial-time approximation scheme, meaning that for every fixed $\varepsilon > 0$ there is a polynomial-time algorithm. The question highlighted above thus asks for a polynomial-time algorithm that finds an ε-Nash equilibrium for arbitrarily small but fixed $\varepsilon > 0$. Here and in the rest of this paper, we follow the literature and assume that the game's payoffs are in the interval $[0, 1]$, and approximations are measured additively; see section 2 for precise definitions.

While every game has at least one Nash equilibrium, the game may actually have many equilibria, some more desirable than others. Thus, an attractive solution concept is to find a Nash equilibrium maximizing an objective such as the social welfare (the total utility of all players). For two-player games this problem is known

*Received by the editors August 4, 2009; accepted for publication (in revised form) October 12, 2010; published electronically January 18, 2011. A preliminary version of this paper appeared in Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms [23].

http://www.siam.org/journals/sicomp/40-1/76699.html

†Technion—Israel Institute of Technology, Haifa 32000, Israel (ehazan@ie.technion.ac.il).

‡Weizmann Institute of Science, Rehovot, 76100, Israel (robert.krauthgamer@weizmann.ac.il).
This author's research was supported in part by a grant from the Fusfeld Research Fund. Part of this work was done while this author was at IBM Almaden.

to be NP-hard [21, 6]. But as we shall soon see, this hardness result is unlikely to extend to ε-equilibrium.

A fairly simple yet surprisingly powerful technique is *random sampling*, where each player's mixed strategy $\vec{x}$ is replaced by another mixed strategy $\vec{x}'$ that has small support, obtained by sampling a few pure strategies independently from $\vec{x}$ and taking $\vec{x}'$ to be a uniform distribution over the chosen pure strategies. (We allow repetitions, i.e., the support is viewed as a multiset.) This technique leads to a simple algorithm that finds, for a two-player game, an ε-equilibrium in quasi-polynomial time $N^{O(\varepsilon^{-2}\log N)}$ [31], assuming that the game is represented as two $N \times N$ matrices.[1] Indeed, applying random sampling on any Nash equilibrium together with Chernoff-like concentration bounds yields an ε-equilibrium consisting of mixed strategies that are each uniform over a multiset of size $O(\varepsilon^{-2}\log N)$, and such an ε-equilibrium can be found by enumeration (exhaustive search). In fact, this argument applies also to the social welfare maximization problem, and thus the algorithm of [31] finds in time $N^{O(\varepsilon^{-2}\log N)}$ an ε-equilibrium whose social welfare is no more than ε smaller than the maximum social welfare of a Nash equilibrium in the game.

The existence of a quasi-polynomial algorithm may be seen as promising evidence that a polynomial algorithm exists. The latter emerged as a major goal and has drawn intense work with encouraging progress [11, 28, 10, 3, 40], culminating (so far) with a polynomial-time algorithm that computes a 0.3393-equilibrium [40]. All of these algorithms, with the sole exception of [40], rely on the aforementioned approach of proving the existence of a small support ε-equilibrium via sampling, and then finding such an equilibrium using enumeration (exhaustive search) in conjunction with other algorithmic tools (such as linear programming).

While progress on the approximation side has remained steady, the other side of computational lower bounds has resisted attempts to exclude PTAS by extending the known hardness results to approximations, either for any equilibrium or for an objective-maximizing one. For instance, the aforementioned PPAD-hardness results [5] extend to a fully polynomial-time approximation scheme (FPTAS), which is defined similarly to a PTAS except for the additional requirement that the running time is polynomial also in $1/\varepsilon$; they extend also to the smoothed complexity of the problem. These two extensions fall short of excluding a PTAS, and the reason for this difficulty might be the aforementioned quasi-polynomial time algorithms, due to which it is less plausible that we can prove hardness of approximation based on NP-hardness or PPAD-hardness for the corresponding question.

In this paper we use a nonstandard hardness assumption to give the first negative evidence for the existence of a PTAS for the objective-maximizing question. We design a reduction from the well-known problem of finding a hidden (planted) clique in a random graph. The latter choice is nonstandard, as the problem appears to be hard on the average rather than in a worst-case sense. However, in several respects it is an ideal choice. First, it admits a straightforward quasi-polynomial time algorithm. Second, the average-case nature of the problem is particularly suited for constructing games with a highly regular structure, which will be important in our reduction.

The hidden clique problem. Denote by $G_{n,p}$ the distribution over graphs on n vertices generated by placing an edge between every pair of vertices independently with probability p.

In the hidden clique problem, the input is a graph on n vertices drawn at random from the following distribution $G_{n,1/2,k}$: pick $k = k(n)$ arbitrary vertices and place on

[1] Throughout, f is called *quasi-polynomial* if $f(n) \le n^{O(\log n)}$.

them a k-clique, then connect every other pair of vertices by an edge independently with probability 1/2. In other words, the graph is random (like in $G_{n,1/2}$) except for a planted clique whose "location" is hidden (not known to the algorithm) because the clique vertices are chosen by an adversary (but independently of the random graph, e.g., of its degrees). The goal is to recover the planted clique (in polynomial time), with probability at least (say) 1/2 over the input distribution.

In a random graph, the maximum size of a clique is, with high probability, roughly $2\log n$, and when the parameter k is larger than this value, the planted clique will be, with high probability, the unique maximum clique in the graph, and the problem's goal is simply to find the maximum clique in the graph (see Lemma 2.2 for details). The problem was suggested independently by Jerrum [25] and by Kučera [30].

It is not difficult to see that the hidden clique problem becomes easier only as k gets larger, and the best polynomial-time algorithm to date, due to Alon, Krivelevich, and Sudakov [2], solves the problem whenever $k = \Omega(\sqrt{n})$ (see also [18]). Improving over this bound is a well-known open problem, and certain algorithmic techniques provably fail this task, namely the Metropolis process [25] and the Lovász–Schrijver hierarchy of relaxations [19]. Recent results [20, 4] based on r-dimensional tensors (the generalization of matrices to dimension $r \geq 3$) suggest an algorithmic approach capable of finding a hidden clique of size $O(n^{1/r})$, but currently this tensor-based approach is not known to yield a polynomial-time algorithm.

The hidden clique problem can be easily solved in quasi-polynomial time $n^{O(\log n)}$; for the most difficult regime $k = O(\log n)$, this is obviously true (via exhaustive search) even for worst-case instances of the maximum clique problem.

1.1. Our results. We relate the worst-case hardness of finding an approximate equilibrium to that of solving the hidden clique problem, formally stated as follows.

THEOREM 1.1. *There are constants $\hat{\varepsilon}, \hat{c} > 0$ such that the following holds. If there is a polynomial-time algorithm that finds in an input two-player game an $\hat{\varepsilon}$-equilibrium whose social welfare is no more than $\hat{\varepsilon}$ smaller than the maximum social welfare of an equilibrium in this game, then there is a (randomized) polynomial-time algorithm that solves the hidden clique problem for $k = \hat{c}\log n$ with high probability.*

We remark that our proof is actually shown for the special case of symmetric two-player games (see section 2 for definitions), which only makes the result stronger (since this is a hardness result). We make no attempt to optimize various constants in the proofs.

Subsequent work. Recently, Minder and Vilenchik [33] improved the constants in our main theorem, showing that a polynomial-time approximation scheme for the best Nash equilibrium implies a polynomial-time algorithm for finding a hidden clique of size $(3+\delta)\log n$ in a random graph, or alternatively for the decision version of detecting a hidden clique of size $(2+\delta)\log n$, for an arbitrarily small constant $\delta > 0$. The leading constant 2 is a natural barrier for the problem, since a clique of size roughly $2\log n$ exists in a random graph with very high probability.

1.2. Related work. There are complexity classes that attempt to capture problems requiring running time $n^{O(\log n)}$; see [37] and the references in section 5 therein. It is plausible that our approach of relying on the hidden clique problem may be used to prove hardness of approximation for problems mentioned in [37], such as the VC dimension of a 0-1 matrix and the minimum dominating set in a tournament graph.

Average-case hardness. The hidden clique problem is related to the assumption that refuting 3-satisfiability (3SAT) is hard on average (for low-density formulas), which was used by Feige [15] to derive constant factor hardness of approximation for

several problems, such as minimum bisection, dense k-subgraph, and maximum bipartite clique. His results may be interpreted as evidence that approximation within the same (or similar) factor is actually NP-hard, which is a plausible possibility but not known to date. In fact, the random 3SAT refutation conjecture may be viewed [16] as an analogue of the hidden clique problem, in the following sense. An important distinction between these two average-case problems appears to be that straightforward algorithms based on enumeration require different running times, exponential for one problem and quasi-polynomial for the other. However, they are based on different yet related combinatorial optimization problems, and it is perhaps not surprising that some of the inapproximability results in [15] can be shown (more or less) also under the assumption that the hidden clique problem cannot be solved in polynomial time. Consider for example the dense k-subgraph problem; the hidden clique graph itself obviously contains a k-vertex subgraph of full density, while any algorithm that is likely to find in it a sufficiently dense k-vertex subgraph can be used to find the planted clique; see, e.g., Lemma 5.3. The argument for the maximum bipartite clique problem is similar.

It is worth noting that the assumption that the hidden clique problem is hard was used in a few other contexts, including for cryptographic applications [26] and for hardness of testing almost k-wise independence [1]. The decision version of the hidden clique problem, namely, to distinguish between the distributions $G_{n,1/2,k}$ and $G_{n,1/2}$, is attributed to Saks in [29, section 5].

Computing equilibria. The last decade has seen a vast amount of literature on computational aspects of equilibria in various scenarios, including, e.g., succinct games (where the payoffs can be represented succinctly, e.g., due to strong symmetries or because direct interaction between players is limited to a certain graph structure) or markets (where sellers and buyers interact via prices). For more details, we refer the reader to the excellent and timely surveys [36, 38] and the many references therein. More concretely for Nash equilibrium in bimatrix games, see the recent surveys [35, 39].

In general, the problems of finding any equilibrium and that of finding an equilibrium that maximizes some objective need not have the same (runtime) complexity, although certain algorithmic techniques may be effective to both. As mentioned earlier, this indeed happens for ε-equilibrium in two-player games when employing random sampling combined with quasi-polynomial enumeration. Another example is the use of the discretization method [27], which was recently used in [12] to find an ε-equilibrium in anonymous games with fixed number of strategies, but actually extends to the value-maximization version [7]. Yet another example are the algorithms of [27, 13] for graphical games on bounded degree trees.

2. Preliminaries. Let $[n] = \{1, 2, \ldots, n\}$. An event $\mathcal{E}$ is said to occur with high probability if $\Pr[\mathcal{E}] \geq 1 - 1/n^{\Omega(1)}$; the value of n will be clear from the context. An algorithm is called *efficient* if it runs in polynomial time, $n^{O(1)}$. Throughout, we ignore rounding issues, assuming, e.g., that $\log n$ is integral. All logarithms are to base 2, unless stated explicitly.

2.1. Nash equilibria in games. In what follows, we restrict our attention to symmetric games, and hence our definition assumes square matrices for the payoff. A (square) *two-player bimatrix game* is defined by two payoff matrices $R, C \in \mathbb{R}^{n \times n}$, such that if the row and column players choose pure strategies $i, j \in [n]$, respectively, the payoffs to the row and column players are $R(i, j)$ and $C(i, j)$, respectively. The game is called *symmetric* if $C = R^\top$.

A *mixed strategy* for a player is a distribution over pure strategies (i.e., rows/columns), and for brevity we may refer to it simply as a strategy. An ε-approximate Nash equilibrium is a pair of mixed strategies (x, y) such that

$$\forall i \in [n], \quad e_i^\top Ry \le x^\top Ry + \varepsilon,$$
$$\forall j \in [n], \quad x^\top Ce_j \le x^\top Cy + \varepsilon.$$

Here and throughout, e_i is the ith standard basis vector, i.e., 1 in ith coordinate, and 0 in all other coordinates. If $\varepsilon = 0$, the strategy pair is called a *Nash equilibrium* (NE). The definition immediately implies the following.

PROPOSITION 2.1. *For an ε-equilibrium (x, y), it holds that for all mixed strategies $\tilde{x}, \tilde{y}$,*

$$\tilde{x}^\top Ry \le x^\top Ry + \varepsilon,$$
$$x^\top R\tilde{y} \le x^\top Ry + \varepsilon.$$

As we are concerned with an additive notion of approximation, we assume that the entries of the matrices are in the range $[0, M]$, for M which is a constant independent of all the other parameters. It will be technically convenient to use $M \neq 1$, but the results easily translate to the case $M = 1$ by scaling all payoffs.

Consider a pair of strategies (x, y). We call $x^\top Ry$ the *payoff* of the row player (this is actually the expected payoff), and similarly for the column player. The *value* of an (approximate) equilibrium for the game is the average of the payoffs of the two players. Recall that *social welfare* is the total payoff of the two players, and thus equals twice the value.

2.2. The hidden clique problem. Recall that in this problem the input is drawn at random from the distribution $G_{n,1/2,k}$. Intuitively, the problem only becomes easier as k gets larger, at least in our regime of interest, namely, $k \ge c_0 \log n$ for sufficiently large constant $c_0 > 0$. This intuition can be made precise as follows.

LEMMA 2.2. *Suppose there are a constant $c_1 > 0$ and a polynomial-time algorithm such that given an instance of the hidden clique problem with $k \ge c_1 \log n$, this algorithm finds a clique of size $100 \log n$ with probability at least $1/2$. Then there exists a constant $c_0 > 0$ and a randomized polynomial-time algorithm that solves the hidden clique problem for every $k \ge c_0 \log n$.*

This lemma is probably known, but since we could not provide a reference, we prove it in the appendix, essentially using ideas from [2] and [32]. Notice that one cannot employ simple techniques that are useful in worst-case instances, such as repeatedly finding and removing from the input graph a clique of size $100 \log n$ (using the assumed algorithm), because such operations affect the graph distribution (even one such iteration results in a graph not distributed like $G_{n,1/2,k}$) not to mention, of course, that the assumed algorithm succeeds only with probability $1/2$ (and repeating it need not amplify the success probability).

3. The reduction. We prove Theorem 1.1 by reducing the hidden clique problem to the Nash equilibrium problem. That is, given an input instance of the hidden clique problem we construct a two-player game such that with high probability (over the randomness in our construction and in the hidden clique instance), a high-value approximate equilibrium leads, in polynomial time, to a solution to the hidden clique instance.

Techniques. Our construction is motivated by an observation of Motzkin and Straus [34] (and independently by Halperin and Hazan [22]) that for every graph, the quadratic form corresponding to the graph's adjacency matrix, when considered over the unit simplex (i.e., all probability distributions over $[n]$), is maximized exactly at a (normalized) incidence vector of a maximum clique in the graph. We rely on this observation, as one portion of the game we construct is exactly the adjacency matrix of the hidden clique instance. However, this is not enough to obtain a suitable Nash equilibrium instance.

First, an equilibrium is a bilinear form rather than a quadratic form, so hence the results of [34, 22] are not directly applicable. We thus use (mainly in Lemma 5.2) the special properties of an approximate equilibrium to prove a relationship of similar flavor between bilinear forms on the adjacency matrix and large cliques (or actually dense subgraphs) in the graph.

Second, a simple use of the adjacency matrix yields a very small gap (between vectors corresponding to a clique and those that do not) that is by far insufficient to rule out a PTAS. To boost this gap we use an idea of Feder, Nazerzadeh, and Saberi [14] to eliminate from the game all equilibria of small support (cardinality at most $O(\log n)$).

The construction. Let $\hat{\varepsilon}, \hat{c}$ and M, c_1, c_2 be constants to be defined shortly. Given an instance $G \in G_{n,1/2,k}$ of the hidden clique problem, consider the two-player game defined by the following payoff matrices (for the row player and the column player, respectively):

$$R = \begin{pmatrix} A & -B^\top \\ B & \mathbf{0} \end{pmatrix}, \qquad C = \begin{pmatrix} A & B^\top \\ -B & \mathbf{0} \end{pmatrix}.$$

The matrices R, C are of size $N \times N$ for $N = n^{c_1}$. These matrices are constructed from the following blocks.

1. The matrix A, which appears in the upper left $n \times n$ block in both R and C, is the adjacency matrix of G with self-loops added.
2. The lower right block $\mathbf{0}$ in both R, C is the all-zeros matrix of size $(N-n) \times (N-n)$.
3. All other entries are set via an $(N-n) \times n$ matrix B whose entries are set independently at random to be

$$B_{i,j} = \begin{cases} M & \text{with probability } \frac{3}{4} \cdot \frac{1}{M}, \\ 0 & \text{otherwise.} \end{cases}$$

Notice that the game is symmetric, i.e., $C = R^\top$, and that outside the upper left block A, the game is zero-sum.

Choice of parameters. We set the parameters in our construction as follows. We start with two absolute constants:

- $M = 12$ and $c_2 = 2000$,

and use them to derive three other constants:

- $c_1 = 2 + c_2 \log(4M/3)$ (recall $N = n^{c_1}$),
- $\hat{c} = 32M^2(c_1 + 2)$, and
- $\hat{\varepsilon} = 1/50M$.

The values chosen here are just large enough for the proof to work easily; as one might expect, setting any of the first four constants to be larger, or the last constant $\hat{\varepsilon}$ to be smaller, would maintain the correctness.

As is standard in computational complexity, we prove Theorem 1.1 by analyzing the completeness and soundness of the reduction.

Henceforth, when we say with high probability, it means over the choice of G from $G_{n,1/2,k}$, over the construction of the game (namely the randomness in B), and over the coin tosses of our algorithms. We note, however, that most of our algorithms are deterministic; the only exception is Lemma 2.2 (and of course the algorithms that invoke it).

4. Completeness.

LEMMA 4.1. *With high probability, the game above has an equilibrium of value* 1.

Proof. Consider the distributions (mixed strategies) $x = y$ which are a uniform distribution over the strategies corresponding to the planted k-clique, i.e., $x_i = \frac{1}{k}$ if i is in the planted clique, and $x_i = 0$ otherwise. The value of this strategy set is $\frac{1}{2}x^\top(R+C)y = 1$. We shall prove that with high probability this is an equilibrium.

Consider without loss of generality the row player, and observe that her payoff when she plays x is exactly 1. We need to show that she cannot obtain a larger payoff by playing instead any strategy $i \in [N]$. For $i \le n$, her new payoff is at most the largest entry in A, i.e., 1. For each $i > n$, her new payoff is the average of k distinct values in B, which is highly concentrated around its mean 3/4. Formally, we use the following Chernoff–Hoeffding bound.

THEOREM 4.2 (see [24]). *Let $X_1, \ldots, X_m$ be independent random variables bounded by $|X_j| \le C$, and let $\bar{X} = \frac{1}{m}\sum_j X_j$. Then for all $t > 0$,*

$$\Pr[\bar{X} - \mathbf{E}[\bar{X}] \ge t] \le \exp(-\tfrac{mt^2}{2C^2}).$$

In our case, the variables satisfy $|X_j| \le M$ and $\mathbf{E}[X_j] = \frac{3}{4}$, and $\bar{X}$ is the payoff of playing strategy $i > n$ (when the other player still plays $x = y$). We thus obtain

$$\Pr[\bar{X} \ge 1] \le \exp(-\tfrac{k}{32M^2}).$$

By a union bound over all strategies $i > n$ we have that the probability a strategy $i > n$ exists with payoff larger than 1 is at most $(N-n)\cdot e^{-\frac{k}{32M^2}} \le n^{c_1 - \frac{\hat{c}}{32M^2}} = 1/n^2$, where the last inequality follows by our choice of c_1 and $\hat{c}$. This completes the proof of Lemma 4.1. ∎

5. Soundness. To complete the proof of Theorem 1.1, we shall show that with high probability, every $\hat{\varepsilon}$-approximate equilibrium with value $\ge 1 - \hat{\varepsilon}$ in the game can be used to find the hidden clique efficiently. We do this in three steps, using the three lemmas below.

For our purpose, a bipartite subgraph is two equal-size subsets $V_1, V_2 \subseteq V(G)$ (not necessarily disjoint); its density is the probability that random $v_1 \in V_1$, $v_2 \in V_2$ are connected by an edge in the input graph with self-loops added, formally $\text{density}(V_1, V_2) = \mathbf{E}[A_{v_1,v_2}]$.

LEMMA 5.1. *With high probability, given an $\hat{\varepsilon}$-equilibrium in the game with value $\ge 1 - \hat{\varepsilon}$, we can efficiently compute a $(4M\hat{\varepsilon})$-equilibrium that is supported only on A and has value $\ge 1 - \hat{\varepsilon}$.*

LEMMA 5.2. *With high probability, given a $(4M\hat{\varepsilon})$-equilibrium supported only on the matrix A and with value $\ge 1 - \hat{\varepsilon}$, we can efficiently find a bipartite graph of size $c_2 \log n$ and density $\ge \frac{3}{5}$ in the input graph.*

LEMMA 5.3. *With high probability, given a bipartite subgraph of size $c_2 \log n$ and density $\ge \frac{3}{5}$ in the input graph, we can efficiently find the entire planted hidden k-clique.*

5.1. Proof of Lemma 5.1. The following two claims are stated with a general parameter $\delta > 0$, although we will later use them only with a specific value $\delta = \hat{\varepsilon}$.

CLAIM 5.1. *In every pair of mixed strategies achieving value $\geq 1-\delta$, at most δ of the probability mass of each player resides on strategies not in $[n]$.*

Proof. The contribution to the value of the equilibrium from outside the upper left block is 0, because over there the game is zero-sum. Inside that block the two players receive identical payoffs, which are according to A and thus upper bounded by 1. Thus,

$$1-\delta \leq \frac{1}{2}x^\top(R+C)y = \sum_{i,j\in[n]} x_i y_j A_{ij} \leq \left(\sum_{i\in[n]} x_i\right)\left(\sum_{j\in[n]} y_j\right),$$

and it immediately follows that both $\sum_{i\in[n]} x_i$ and $\sum_{j\in[n]} y_j$ are at least $1-\delta$. ∎

CLAIM 5.2. *Given an $\hat{\varepsilon}$-equilibrium where at most δ of each player's probability mass resides on strategies not in $[n]$, we can find an $(\hat{\varepsilon}+3M\delta)$-equilibrium that is supported only on A and whose value is at least as large.*

Proof. Given an $\hat{\varepsilon}$-equilibrium (x,y), we obtain a new equilibrium $(\tilde{x},\tilde{y})$ by restricting each player's support to $[n]$, i.e., removing strategies not in $[n]$ and scaling to obtain a probability distribution. Since the game is zero-sum outside of A, removing strategies not in A does not change the value, and since the entries in A are nonnegative, the scaling operation can only increase the value, i.e., $\tilde{x}^\top(R+C)\tilde{y} \geq x^\top(R+C)y$.

To bound defections, consider without loss of generality the row player. First, her payoff when defecting to strategy $i \in [N]$ does not change much, i.e., $|e_i^\top R\tilde{y} - e_i^\top Ry| \leq M\delta$, because the total mass of y moved around is at most δ, and because entries in the same row (or same column) of R differ by at most M. Furthermore, her payoff in the new equilibrium does not change much, i.e., $|\tilde{x}^\top R\tilde{y} - x^\top Ry| \leq 2M\delta$, again because at most 2δ of the total probability mass of x and y was moved. ∎

Lemma 5.1 now follows immediately from the two claims above by setting $\delta = \hat{\varepsilon}$.

5.2. Proof of Lemma 5.2. To prove this lemma, we first need the following structural claim.

CLAIM 5.3. *With high probability, in every 1/2-equilibrium supported only on the matrix A, the total probability mass every player places on every set of $c_2 \log n$ pure strategies is $\leq 2/M$.*

Proof. For convenience, denote $d = c_2 \log n$. Suppose that one of the players, say the column player, has probability mass of more than $\frac{2}{M}$ on a given set of d strategies. Let us compute the probability that there exists a row in B in which the d corresponding entries all have a value of M. If this event happens, then we do not have an ε-equilibrium, since the row player can defect to this particular row, to obtain payoff $\geq \frac{2}{M}\cdot M = 2$, while her current payoff is ≤ 1. The probability this event happens for a single row is very small, namely p^d for $p = \frac{3}{4M}$. But we have $N-n$ rows, and they are independent. Thus, the probability that no row has a streak of M's in the particular d columns is at most

$$(1-p^d)^{N-n} \leq \exp(-p^d N/2) = \exp(-n^{c_1 - c_2 \log \frac{4M}{3}}/2) = \exp(-n^2/2),$$

where the last inequality is by our choice of c_1 and c_2. Hence with probability $\geq 1-e^{-n^2/2}$, there is such a row, and this cannot be an equilibrium.

We now need to rule out all possible subsets sets of size d. There are $\binom{n}{d} \leq n^d$ such subsets, and each one cannot be an equilibrium with probability $\geq 1-e^{-n^2/2}$. We can rule out all of them by a union bound, since $n^d \cdot e^{-n^2/2} \leq e^{-\Omega(n^2)}$. ∎

Proof of Lemma 5.2. Let (x, y) be such a $(4M\hat{\varepsilon})$-equilibrium. Define $T = \{j \in \text{supp}(y) : x^\top A e_j \geq \frac{4}{5}\}$, where e_j is the jth standard basis vector. Observe that T is nonempty, since $x^\top A y \geq 1 - \hat{\varepsilon}$. Furthermore, its total probability mass must be $\sum_{j \in T} y_j > \frac{2}{M}$, as otherwise $x^\top A y \leq \frac{2}{M} \cdot 1 + (1 - \frac{2}{M}) \cdot \frac{4}{5} < 1 - \hat{\varepsilon}$ by our choice of $\hat{\varepsilon}$. Since $\sum_{j \in T} y_j > \frac{2}{M}$ and (x, y) is a $(4M\hat{\varepsilon})$-equilibrium, we have by Claim 5.3 that $|T| > c_2 \log n$. To get size exactly $c_2 \log n$, we can just take an arbitrary subset of T and rename it to be T. Thus we assume henceforth $|T| = c_2 \log n$. Denoting by u_T the uniform distribution on T, the pair (x, u_T) satisfies

$$x^\top A u_T \geq \tfrac{4}{5}.$$

Now define $S = \{i \in \text{supp}(x) : e_i^\top A u_T \geq \frac{3}{5}\}$. Its total probability mass must be $\sum_{i \in S} x_i > \frac{2}{M}$, as otherwise $x^\top A u_T \leq \frac{2}{M} \cdot 1 + (1 - \frac{2}{M}) \cdot \frac{3}{5} < \frac{4}{5}$, by our choice of $M > 4$. By again applying Claim 5.3 to the 1/2-equilibrium (x, y), we then have $|S| > c_2 \log n$. Again, to get size exactly $c_2 \log n$, we can just take an arbitrary subset of S and rename it to be the set S itself. Let u_S be the uniform distribution over the set S. Then $u_S^\top A u_T \geq \frac{3}{5}$, i.e., S, T define a bipartite subgraph of size $c_2 \log n$ and density $\geq \frac{3}{5}$. ∎

5.3. Proof of Lemma 5.3. The main step in proving Lemma 5.3 is the following claim, which shows that a dense bipartite subgraph can be used to recover a large clique.

Claim 5.4. *With high probability, given a bipartite subgraph of size $c_2 \log n$ and density $\geq \frac{3}{5}$ in the input graph, we can efficiently find a clique of size $100 \log n$.*

Proof. Let $S^*, T^* \subset V(G)$ be the two sets forming a bipartite subgraph with $\text{density}(S^*, T^*) \geq \frac{3}{5}$, and let $W \subset V(G)$ denote the vertices of the planted clique, i.e., $|W| = k$. In what follows, we use a union bound to show that with high probability at least $\frac{1}{20}$ of the vertices in S^* must lie in the planted clique W.

The number of possible sets $S, T \subset V(G)$ with $|S| = |T| = c_2 \log n$ and $|S \cap W| < \frac{1}{20}|S|$ is at most $\binom{n}{|S|}\binom{n}{|T|} \leq n^{2c_2 \log n}$. Now fix two such sets S, T. For the event $\text{density}(S, T) \geq 3/5$ to occur, it is necessary to have $\text{density}(S \setminus W, T) \geq 0.55$, because otherwise

$$\text{density}(S, T) = \tfrac{|S \cap W|}{|S|} \text{density}(S \cap W, T) + \tfrac{S \setminus W}{|S|} \text{density}(S \setminus W, T) < \tfrac{1}{20} \cdot 1 + \tfrac{19}{20} \cdot 0.55 < \tfrac{3}{5}.$$

We can bound the probability of this necessary event as follows. The density between $S \setminus W$ and T is essentially the average of $\Theta(c_2 \log n)^2$ Bernoulli random variables each with expectation 1/2 (strictly speaking, there are two exceptions: some Bernoulli variables are included twice, and at most $c_2 \log n$ terms correspond to self-loops). Thus, by the Chernoff–Hoeffding bound above, $\Pr[\text{density}(S \setminus W, T) \geq 0.55] \leq \exp(-\Omega(c_2 \log n)^2)$. Putting these facts together using a union bound,

$$\begin{aligned}
&\Pr\left[\exists S, T \text{ s.t. } |S| = |T| = c_2 \log n,\ \tfrac{|S \cap W|}{S} < \tfrac{1}{20},\ \text{density}(S \setminus W, T) \geq 0.55\right] \\
&\leq n^{2c_2 \log n} \cdot e^{-\Omega(c_2 \log n)^2} \\
&\leq 1/n^{\Omega(1)}.
\end{aligned}$$

We conclude that with high probability, the set S^* given to us must satisfy that at least $\frac{1}{20}$ of its vertices lie in the planted clique. We can try all subsets of S^* of size $\frac{1}{20} c_2 \log n$ by exhaustive search (the number of such sets is $n^{O(c_2)} = n^{O(1)}$), and find

the largest subset $S' \subset S^*$ that forms a clique in G. By the above analysis, with high probability $|S'| \geq \frac{1}{20} c_2 \log n = 100 \log n$. □

Lemma 5.3 now follows by combining the above claim with Lemma 2.2. This completes the proof of Theorem 1.1 also.

6. Concluding remarks. We have shown that a PTAS for the "best" Nash equilibrium implies an efficient algorithm to a seemingly difficult combinatorial optimization problem. The measure of quality for equilibria we have used is the total payoff to the two players. It seems plausible that other quality measures, such as those considered by Gilboa and Zemel [21], can be reduced to the hidden clique problem in much the same way.

The problem from which we carry out the reduction, namely the hidden clique problem, is nonstandard in the sense that it is an assumption about average-case hardness (in addition to being easily solvable in quasi-polynomial time). It is plausible that it suffices to assume worst-case hardness, and show, e.g., a reduction from the problem of finding in an n-vertex graph a clique of size $O(\log n)$. In light of the known quasi-polynomial time algorithms, an alternative complexity assumption that could potentially be used is that (say) maximum clique cannot be solved in time $2^{O(\sqrt{n} \log n)}$; see, e.g., [17]. Yet another direction is to relate the hardness of computing the best Nash equilibria to the complexity class logarithmically-restricted NP (LOGNP) of [37], because it naturally captures the known (quasi-polynomial time) algorithms for approximating Nash equilibria.

Finally, an intriguing question is whether such techniques can possibly be applied to the problem of finding a PTAS for any Nash equilibrium, i.e., to the regime of the PPAD complexity class.

Appendix. Deferred proof from section 2.

Proof of Lemma 2.2. Suppose there exists a polynomial-time algorithm $\mathcal{A}^*$ that, given the hidden clique problem with $k \geq c_1 \log n$, finds a clique of size $100 \log n$. We prove that there exists a (randomized) polynomial-time algorithm that solves the hidden clique problem exactly for every $k \geq c_0 \log n$, where $c_0 = 2tc_1$ for a sufficiently large t to be determined later. The algorithm is composed of two stages.

Stage 1. Randomly partition the graph vertices into t parts. In each part, the expected number of vertices from the planted clique vertices is $k/t \geq \frac{c_0 \log n}{t} = 2c_1 \log n$. Furthermore, using Chernoff bounds it can be shown that with probability $> 7/8$, every part contains at least $c_1 \log n$ vertices from the hidden clique. In our analysis we shall assume henceforth that this is the case.

In each part separately, first complete it into an instance of hidden clique of size exactly n by adding $n - n/t$ vertices and connecting all new potential edges with probability $\frac{1}{2}$. Then apply the polynomial-time algorithm $\mathcal{A}^*$. Observe that each part is distributed exactly as a hidden clique instance, and by our assumption its hidden clique size is large enough that algorithm $\mathcal{A}^*$ succeeds, with probability $\geq 1/2$, in finding a clique of size at least $100 \log n$. Since the different parts are independent, the probability that $\mathcal{A}^*$ succeeds in one or less parts is $\leq (t+1)2^{-t} < 1/8$ for, say, $t = 6$.

Even if this does not occur, report fail. Henceforth assume that $\mathcal{A}^*$ succeeds in at least two parts.

In each part where $\mathcal{A}^*$ succeeds, we may assume that the clique size is exactly $100 \log n$ by removing arbitrary vertices from it. It can be shown that in the random portion of the graph (i.e., not using the planted clique), the maximum clique size is with very high probability at most $3 \log n$. In our analysis we shall assume henceforth

that this is the case in all parts of the partition, and hence at least $97 \log n$ among the $100 \log n$ vertices of the clique found belong to the planted clique.

Stage 2. In each part i apply the following. Identify *another* part $j \neq i$, where $\mathcal{A}^*$ succeeded in finding a clique of size $100 \log n$. Select the vertices in the part i whose degree towards the clique found in part j is at least $97 \log n$. Call these vertices Q_i and report all selected vertices, i.e., $Q = \cup_i Q_i$.

To analyze this stage, observe that a vertex v from part i that belongs to the planted clique must have degree at least $97 \log n$ towards the clique found in another part j, and thus belongs to Q_i. On the other hand, for a vertex v in part i that does not belong to the planted clique, the expected degree towards any fixed subset of $100 \log n$ vertices in part j is $50 \log n$. Notice that the choice of the part j and the clique found in it are determined only by the edges internal to the different parts (possibly in a complicated way, e.g., if $\mathcal{A}^*$ succeeds in more than two parts, then j will depend on the tie breaking method), and are thus completely independent of the edges connecting different parts. Thus, the probability that v has degree at least $97 \log n$ towards the corresponding clique in part j is, using the Chernoff bound, $\ll 1/n^2$. By a union bound over all vertices we get that with very high probability all such vertices do not belong to Q_i. Combining this with the events mentioned earlier, we get by a union bound that $Q = \cup_i Q_i$ contains exactly all the hidden clique vertices with probability at least $2/3$. □

Acknowledgments. We thank Uri Feige, Nimrod Megiddo, and Aranyak Mehta for useful conversations regarding different problems and concepts studied in this paper.

REFERENCES

[1] N. Alon, A. Andoni, T. Kaufman, K. Matulef, R. Rubinfeld, and N. Xie, *Testing k-wise and almost k-wise independence*, in Proceedings of the 39th Annual ACM Symposium on Theory of Computing, ACM, New York, 2007, pp. 496–505.

[2] N. Alon, M. Krivelevich, and B. Sudakov, *Finding a large hidden clique in a random graph*, Random Structures Algorithms, 13 (1998), pp. 457–466.

[3] H. Bosse, J. Byrka, and E. Markakis, *New algorithms for approximate Nash equilibria in bimatrix games*, in WINE 2007, Lecture Notes in Comput. Sci. 4858, Springer-Verlag, Berlin, 2007, pp. 17–29.

[4] S. C. Brubaker and S. S. Vempala, *Random tensors and planted cliques*, in Proceedings of the 13th International Workshop on Randomization and Computation (RANDOM), Lecture Notes in Comput. Sci. 5687, Springer-Verlag, Berlin, 2009, pp. 406–419.

[5] X. Chen, X. Deng, and S.-H. Teng, *Settling the complexity of computing two-player Nash equilibria*, J. ACM, 56 (2009), article 14.

[6] V. Conitzer and T. Sandholm, *Complexity results about Nash equilibria*, in Proceedings of the 18th International Joint Conference on Artificial Intelligence, Morgan Kaufmann Publishers Inc., San Francisco, 2003, pp. 765–771.

[7] C. Daskalakis, *private communication*, 2008.

[8] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou, *The complexity of computing a Nash equilibrium*, SIAM J. Comput., 39 (2009), pp. 195–259.

[9] C. Daskalakis, P. W. Goldberg, and C. H. Papadimitriou, *The complexity of computing a Nash equilibrium*, Commun. ACM, 52 (2009), pp. 89–97.

[10] C. Daskalakis, A. Mehta, and C. Papadimitriou, *Progress in approximate Nash equilibria*, in Proceedings of the 8th ACM Conference on Electronic Commerce, ACM, New York, 2007, pp. 355–358.

[11] C. Daskalakis, A. Mehta, and C. H. Papadimitriou, *A note on approximate Nash equilibria*, in WINE 2006, Lecture Notes in Comput. Sci. 4286, Springer-Verlag, Berlin, 2006, pp. 297–306.

[12] C. Daskalakis and C. H. Papadimitriou, *Discretized multinomial distributions and Nash equilibria in anonymous games*, in Proceedings of the 49th Annual IEEE Symposium on

Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2008, pp. 25–34.

[13] E. Elkind, L. A. Golberg, and P. W. Goldberg, *Computing good Nash equilibria in graphical games*, in Proceedings of the 8th ACM Conference on Electronic Commerce, ACM, New York, 2007, pp. 162–171.

[14] T. Feder, H. Nazerzadeh, and A. Saberi, *Approximating Nash equilibria using small-support strategies*, in Proceedings of the 8th ACM Conference on Electronic Commerce, ACM, New York, 2007, pp. 352–354.

[15] U. Feige, *Relations between average case complexity and approximation complexity*, in Proceedings of the 34th Annual ACM Symposium on Theory of Computing, ACM, New York, 2002, pp. 534–543.

[16] U. Feige, *private communication*, 2008.

[17] U. Feige and J. Kilian, *On limited versus polynomial nondeterminism*, Chicago J. Theoret. Comput. Sci., 1997, article 1.

[18] U. Feige and R. Krauthgamer, *Finding and certifying a large hidden clique in a semirandom graph*, Random Structures Algorithms, 16 (2000), pp. 195–208.

[19] U. Feige and R. Krauthgamer, *The probable value of the Lovász–Schrijver relaxations for maximum independent set*, SIAM J. Comput., 32 (2003), pp. 345–370.

[20] A. M. Frieze and R. Kannan, *A new approach to the planted clique problem*, in Foundations of Software Technology and Theoretical Computer Science (FSTTCS), Dagstuhl Seminar Proceedings 08004, Schloss Dagstuhl, Germany, 2008 pp. 187–198.

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

[22] E. Halperin and E. Hazan, *HAPLOFREQ—estimating haplotype frequencies efficiently*, in Research in Computational Molecular Biology, Lecture Notes in Comput. Sci. 3500, Springer, Berlin, 2005, pp. 553–568.

[23] E. Hazan and R. Krauthgamer, *How hard is it to approximate the best Nash equilibrium?*, in Proceedings of the 20th Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadelphia, 2009, pp. 720–727.

[24] W. Hoeffding, *Probability inequalities for sums of bounded random variables*, J. Amer. Statist. Assoc., 58 (1963), pp. 13–30.

[25] M. Jerrum, *Large cliques elude the Metropolis process*, Random Structures Algorithms, 3 (1992), pp. 347–359.

[26] A. Juels and M. Peinado, *Hiding cliques for cryptographic security*, Des. Codes Cryptogr., 20 (2000), pp. 269–280.

[27] M. J. Kearns, M. L. Littman, and S. P. Singh, *Graphical models for game theory*, in Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers, San Francisco, 2001, pp. 253–260.

[28] S. C. Kontogiannis, P. N. Panagopoulou, and P. G. Spirakis, *Polynomial algorithms for approximating Nash equilibria of bimatrix games*, in WINE 2006, Lecture Notes in Comput. Sci. 4286, Springer-Verlag, Berlin, 2006, pp. 286–296.

[29] M. Krivelevich and V. H. Vu, *Approximating the independence number and the chromatic number in expected polynomial time*, J. Comb. Optim., 6 (2002), pp. 143–155.

[30] L. Kučera, *Expected complexity of graph partitioning problems*, Discrete Appl. Math., 57 (1995), pp. 193–212.

[31] R. J. Lipton, E. Markakis, and A. Mehta, *Playing large games using simple strategies*, in Proceedings of the 4th ACM Conference on Electronic Commerce, ACM, New York, 2003, pp. 36–41.

[32] F. McSherry, *Spectral partitioning of random graphs*, in Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science, IEEE Computer Society, Los Alamitos, CA, 2001, pp. 529–537.

[33] L. Minder and D. Vilenchik, *Small clique detection and approximate Nash equilibria*, in APPROX '09 / RANDOM '09: Proceedings of the 12th International Workshop and 13th International Workshop on Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 673–685.

[34] T. S. Motzkin and E. G. Straus, *Maxima for graphs and a new proof of a theorem of Turán*, Canad. J. Math., 17 (1965), pp. 533–540.

[35] C. H. Papadimitriou, *The complexity of finding Nash equilibria*, in Algorithmic Game Theory, N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, eds., Cambridge University Press, Cambridge, UK, 2007, pp. 29–51.

[36] C. H. Papadimitriou, *The search for equilibrium concepts*, in SAGT 2008, Lecture Notes in Comput. Sci. 4997, Springer-Verlag, Berlin, 2008, pp. 1–3.

[37] C. H. Papadimitriou and M. Yannakakis, *On limited nondeterminism and the complexity of the V-C dimension*, J. Comput. System Sci., 53 (1996), pp. 161–170.
[38] T. Roughgarden, *Algorithmic game theory: Some greatest hits and future directions*, in IFIP TCS, IFIP 273, G. Ausiello, J. Karhumäki, G. Mauri, and C.-H. L. Ong, eds., Springer, Boston, 2008, pp. 21–42.
[39] P. G. Spirakis, *Approximate equilibria for strategic two person games*, in SAGT 2008, Lecture Notes in Comput. Sci. 4997, Springer, Berlin, 2008, pp. 5–21.
[40] H. Tsaknakis and P. G. Spirakis, *An optimization approach for approximate Nash equilibria*, in WINE 2007, Lecture Notes in Comput. Sci. 4858, Springer, Berlin, 2007, pp. 42–56.