社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2006−AL−107 (11)
2006/7/3

Unique Gameの近似困難性について

高橋 敏明
東京大学大学院情報理工学系研究科 コンピュータ科学専攻 今井研究室

近年，Clique，MAX-3SAT，Set Coverといった問題の近似値比の限界が示されてきたが，一方，その他多くの問題の限界はまだ未解決である．

それらを解決する1つの統一的手法として，グラフの最適化問題であるUnique Gameが研究されている．Unique Gameの難しさに関しては推測がされており，この推測は"Unique Games Conjecture"と呼ばれる．もしこの推測が正しければ，Unique Gameからのリダクションの解析がグラフ問題の近似値比の限界を示す有用な手法になる．

本研究では，Unique Gameに対する近似アルゴリズムを実装し，問題を決定する様々な要因について制限した問題を使用することで，ある平行枝がUnique Gameを難しくすることを実験的に見出す．

On Hardness of Approximation for Unique Game

Toshiaki Takahashi
Imai Laboratory
Department of Computer Science
Graduate School of Information Science and Technology
The University of Tokyo

Recently, the boundaries of the approximation ratios of such problems as Clique, MAX-3SAT and Set Cover have been shown, whereas the boundaries of many other problems still remain open.

Unique Game, which is a graph optimization problem, has been studied as one unified approach to solve them. There is a conjecture as to the hardness of Unique Game, which is called "Unique Games Conjecture." If this conjecture is true, the analysis of the reductions from Unique Game is a useful method to show the boundaries of the approximation ratios of the graph problems.

In this study, we empirically found that existence of some parallel edge makes Unique Game hard by implementing the approximation algorithm for Unique Game and using instances restricted to various factors which determine the instances.

1 Introduction

In recent years, approximation algorithms for many NP-hard optimization problems have been studied and their approximation ratios have been improved. In particular, it is large contributions to develop the method which can make approximation ratios of various problems improved. LP-relaxation is one example.

In contrast, it has been difficult to show the hardness of approximation the problems, in other words, few methods to show the boundaries of the approximation ratios of the problems have been found. When we show that a yes-no problem is NP-hard, we construct a reduction to it from another NP-complete problem. We give consideration to only "yes" or "no". In the case of approximation, we have to give consideration to a value of objective function. Therefore, construction of reductions cannot easily show the hardness of approximation.

PCP theorem is one of methods to show hardness of approximation [2, 9]. It means that there exists a constant within which MAX-3SAT is NP-hard to approximate. The construction of reductions from MAX-3SAT leads many results about the hardness of NP-hard problems. As another method, 2-Prover 1-Round Game is proposed. It is simple to construct reductions from 2-Prover 1-Round Game to graph optimization problems, and the construction has the same capability as the construction from MAX-3SAT about showing hardness of NP-hard problems. As a result of them, the hardness of many problems have been shown. For example, CLIQUE is NP-hard to approximate within $n^{1-\varepsilon}$ for every ε [8], Set Cover is NP-hard to approximate within $(1-\varepsilon)\ln n$ for every ε [3]. These results are correspond with the results of achievement of approximation ratios. Therefore, it can be said that the studies of approximation for the problems have the large conclusion.

However, there are many open problems about hardness of NP-hard problems. Consequently, new problems have been studied. One of them is Unique Game. Unique Game is the special version of 2-Prover 1-Round Game. It is more simple to construct reductions from Unique game than from 2-Prover 1-Round Game, and it has been said that the construction might be more capability than the construction from MAX-3SAT. But it is still prediction, because little has been shown about the hardness of it.

Unique Games Conjecture, a conjecture about hardness of Unique Game, has been proposed by Khot [10]. By assuming it, many results have been shown. For example, Vertex Cover might be NP-hard to approximate within $2-\varepsilon$ for every ε [12], MAX-CUT might be NP-hard to approximate within $1/0.878...-\varepsilon$ for every ε [11], and Sparsest Cut is NP-hard to approximate within any constant factor [13]. Furthermore, these results correspond with the results of achievement of approximation ratios. If the conjecture is proved, many new large conclusions might be shown. Some results about it have been shown [13, 5], but none of them proves the conjecture. And good approximation algorithms for Unique Game also have been proposed [4, 10, 16], but none of them disproves the conjecture.

In this study, we implemented Khot's approximation algorithm for Unique Game [10], and experimented with the algorithm. We implemented the algorithm based on semidefinite programming (SDP) relaxation. SDP-relaxation is a useful method to approximate NP-hard programs. For example, the best approximation algorithm of MAX-CUT is based on it [7]. In the experiment, we used random instances restricted to some factors such as size of a set of vertices, size of a set of labels, degrees of a graph and the value of optimal solution. By changing the restriction, we empirically researched which factors make Unique Game difficult, and how difficult Unique Game is. It is shown that if there are parallel edges under some conditions, we can realize Unique Games Conjecture.

2 Preliminaries

2.1 2-Prover 1-Round Game

Definition 1 A 2-Prover 1-Round Game (2P1R Game) instance $U(G(V,E),[k],\{\pi_e\}_{e\in E})$ is defined as follows.

$[k]$ is a set $\{1,...,k\}$. $G=(V,E)$ is a graph with a set of vertices V and a set of edges E, with possibly parallel edges. An edge e whose endpoints are v and w is written as $e(v,w)$. For every $e(v,w)\in E$, there is a relation $\pi_e:[k]\times[k]$. π_e is a set of pairs of labels that takes a label of vertex v and a label of vertex w. $\pi_{e(w,v)}$ is trivially defined to be "inversion" of $\pi_{e(v,w)}$. We can describe the relation as $k\times 2$ vertices bipartite graph. The goal is to assign one label to every vertex of the graph from the set $[k]$. The labeling is supposed to satisfy the constraints given by sets π_e. A labeling $\lambda: V\rightarrow[k]$ satisfies an edge $e(v,w)$, if there is a pair of labels $(\lambda(v),\lambda(w))$ in π_e. Define the indicator function $I^{\lambda}(e)$, which is 1 if e is satisfied by λ and 0 otherwise. The optimum $\mathrm{OPT}(U)$ of the 2-Prover 1-Round Game instance is define to be $\max_{\lambda}\frac{1}{|E|}\sum_{e\in E}I^{\lambda}(e)$. In other words, the objective function is the fraction of satisfied edges in all edges.

Theorem 1 For every constant $\gamma>0$, there exists a sufficiently large constant $k:=k(\gamma)$, such that it is NP-hard to decide whether a 2-Prover 1-Round Game instance $U(G(V,E),[k],\{\pi_e\}_{e\in E})$, has $\mathrm{OPT}(U)\geq 1$, or $\mathrm{OPT}(U)\leq\gamma$.

2.2 Unique Game

Definition 2 A 2P1R Game is called Unique Game if every relation $\pi_{e(v,w)}$ can be denoted by a bijection, i.e., every π_e has exactly k pairs and there is a bijection $f:[k]\rightarrow[k]$ such that every pair in π_e is denoted by $(n,f(n))$. In other words, every bipartite graph describing the relation is perfect matching.

2.3 Unique Games Conjecture

Conjecture 1 For every pair of constants $\varepsilon,\gamma>0$, there exists a sufficiently large constant $k:=k(\varepsilon,\gamma)$, such that it is NP-hard to decide whether a Unique Game instance $U(G(V,E),[k],\{\pi_e\}_{e\in E})$, has $\mathrm{OPT}(U)\geq 1-\varepsilon$, or $\mathrm{OPT}(U)\leq\gamma$.

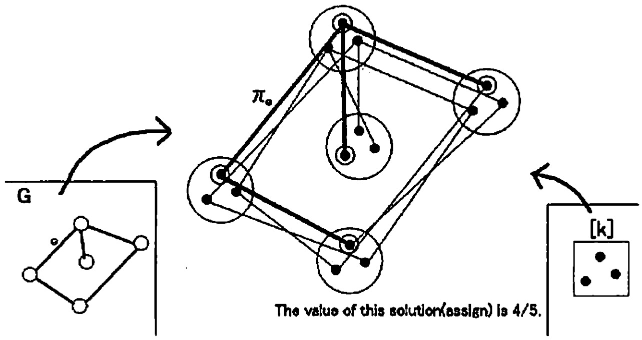

Figure 1: Image of Unique Game.

It is easy to decide whether a Unique Game instance has OPT(U) = 1 or not.

Assuming Unique Games Conjecture, Unique Game is hard to approximate within any constant factor. If there exists an approximation algorithm for Unique Game which approximate within a constant factor $c \geq 0$, it is possible to decide whether the instance has OPT(U) $\geq 1 - \frac{1}{2c}$, or OPT(U) $\leq \frac{1}{c} - \frac{1}{c^2}$ for every constant k. To the instance which has OPT(U) $\geq 1 - \frac{1}{2c}$, such an algorithm can find a solution whose value of the objective function is greater than or equal to $\frac{1}{c} - \frac{1}{2c^2} > \frac{1}{c} - \frac{1}{c^2}$.

By constructing a reduction from Unique Game, hardness of approximation for various problems can be shown similarly.

3 Methods

3.1 Used Algorithm

There are many algorithms based on SDP-relaxation [4, 10, 16]. We used Khot's one [10]. This is a natural algorithm and analogous to the best algorithm for MAX-CUT.

3.1.1 SDP-relaxation

We consider the following 0-1 integer programming formulation. For every vertex v of the instance, we have k boolean variables $v_1, ..., v_k$, with the intended meaning that if a label of v is i then $v_i = 1$ and $v_j = 0$ for $j \neq i$. $\pi^v_{e(v,w)}(i)$ means a label of w which satisfy e with a label i of v. Each constraint $\pi_{e(v,w)}$ contributes $\sum_{i\in[k]} w_{\pi^v_e(i)} v_i$ to the objective function. Therefore, the 0-1 integer programming is as follows.

$$\max \sum_{e(v,w)\in E} \sum_{i\in[k]} w_{\pi^v_e(i)} \cdot v_i$$

Subject to

$$v_i \cdot v_j = 0 \qquad (\forall v \in V, \forall i, j \in [k], i \neq j)$$

$$\sum_{i\in[k]} v_i = 1 \qquad (\forall v \in V)$$

$$v_i \in \{0, 1\} \qquad (\forall v \in V, \forall i \in [k])$$

In the vector program relaxation, each variable v_i is replaced by a vector $\boldsymbol{v_i}$ whose size is $k|V|$.

Next, we add some constraints to make the role of vector variables closer to 0-1 variables. In this part, the methods are various. But it is not clear whether they are equivalent or not.

$$
\begin{aligned}
&\max \sum_{e(v,w)\in E} \sum_{i\in[k]} w_{\pi_e^v(i)} \cdot v_i \\
&\text{Subject to} \\
&\quad v_i \cdot v_j = 0 \qquad (\forall v \in V, \forall i, j \in [k], i \neq j) \\
&\quad \sum_{i\in[k]} \|v_i\|^2 = 1 \qquad (\forall v \in V) \\
&\quad v_i \cdot w_j \geq 0 \qquad (\forall v, w \in V, \forall i, j \in [k]) \\
&\quad \sum_{i,j\in[k]} v_i \cdot w_j = 1 \qquad (\forall v, w \in V)
\end{aligned}
$$

Then, we translate the problem of the vector program into problem of the semidefinite programming.

3.1.2 Semidefinite Programming

The semidefinite programing was solved in polynomial time by the interior method [1, 14, 15]. We obtain the solution of the vector program from the solution of the semidefinite programming by Cholesky Decomposition.

3.1.3 Rounding Procedure

Finally, it is necessary to calculate the solution of Unique Game from the solution of the vector program. In this part, the methods are various, too. This part set the performance of the algorithm.

- Choose a vector r from the normal distribution, i.e. choose every coordinate of r from the distribution $N(0,1)$ independently.
- By replacing r by $-r$ if needed, assume that $r \cdot \sum_{i=1}^{k} v_i \geq 0$. For every $v \in V$, $\sum_{i=1}^{k} v_i$ is same by the constraints of the semidefinite programming.
- Construct the following assignment A : for every vertex v, let

$$A(v) = i_0 \qquad where \quad r \cdot v_{i_0} = \max_{i\in[k]} r \cdot v_i$$

3.2 Existence Analysis

Khot proved the following theorem [10].

Theorem 2 If there exists an assignment that satisfies constraints with total weight $1-\varepsilon$, then the above algorithm produces an assignment that satisfies constraints with expected weight $1 - O(k^2\varepsilon^{1/5}\sqrt{\log\frac{1}{\varepsilon}})$

This means that the algorithm can decide whether a Unique Game instance U has $\mathrm{OPT}(U) > 1-\varepsilon$ or $\mathrm{OPT}(U) \leq 1 - O(k^2\varepsilon^{1/5}\sqrt{\log\frac{1}{\varepsilon}})$. However, it does not disprove Unique Games Conjecture, because for every ε, γ, there exists sufficiently large k such that $1 - O(k^2\varepsilon^{1/5}\sqrt{\log\frac{1}{\varepsilon}}) < \gamma$. Even if we can decide whether U has $\mathrm{OPT}(U) > 1-\varepsilon$ or $\mathrm{OPT}(U) \leq 1 - O(k^2\varepsilon^{1/5}\sqrt{\log\frac{1}{\varepsilon}})$, we cannot decide whether U has $\mathrm{OPT}(U) > 1-\varepsilon$ or $\mathrm{OPT}(U) \leq \gamma > 1 - O(k^2\varepsilon^{1/5}\sqrt{\log\frac{1}{\varepsilon}})$,

3.3 Implementation

We implement the algorithm by C++. Semidefinite programming problems are solved by SDPA [6]. This can solve semidefinite programming problems whose all constraints are equality constraints. Thus, we have to translate the problem by SDP-relaxation into the problem whose all constraints are equality constraints. As a result, the size of the matrix variable is $(k|V| + O((k|V|)^2)) \times (k|V| + O((k|V|)^2)$. Since the rounding procedure is randomized algorithm, we calculate more than once, in our implementation 10000 times, and calculate the average to obtain expectation.

3.4 Used Data

We generated random instances restricted to some factors. We can restrict to following.

- **"Size of a vertices set"**
- **"Size of a labels set"**
- **"The number of parallel edges"** We can specify the number of parallel edges between every pair of two vertices which have edge.
- **"Density of edges"** In this regard, parallel edges between same two vertices are counted as 1 edge.
- **"Assured value"** When we generate random instances, we can give the specified value to one assignment. If the specified value is sufficiently large, we can regard the value as the optimal value.
- **"Others"** We can restrict to other factors according to need.

4 Results

4.1 Approximation Ratio and Size of a Labels Set

We changed the size of a labels set and measured approximation ratio. Every graph G of instances is a complete graph whose size of a vertices set is 4. We experiment 100 times without changing restriction. We consider the sum of values of implemented program divided by the sum of optimal values as "the average of approximation ratio."

Figure 2 shows that the size of a labels set do not make instances hard to approximate. It seems that the size of a labels set is not only factors of hardness of Unique Game.

4.2 Factors with Little Influence

We found that some factors have little influence on hardness of Unique Game by experiments. In the experiments, the way to calculate the average of approximation ratios is same as 4.1.

- **"Parallel edges and density of edges"** Table 1 shows that increase of edges does not make instance hard to approximate.
- **"Assured large value"** Table 2 shows that an instance whose optimal solution have a large value is easy to approximate.

  An instance whose optimal solution have a small value could be hard. However, the way to make it is not clear.

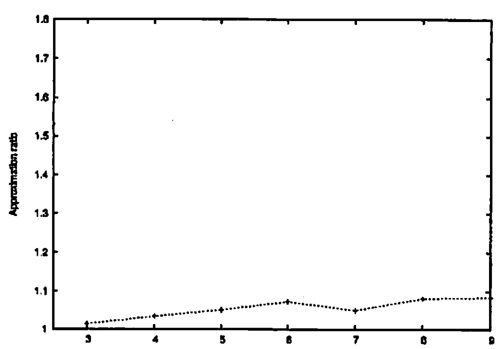

Figure 2: Graph with approximation ratio on the y-axis and size of a labels set on x.

Table 1: Optimal value and approximation ratio for instances $(|V|, k) = (5, 5)$ with various graphs.

Parallel edges	1	2	4	1	2	4
Density of edges	1	1	1	1/2	1/2	1/2
Approximation ratio	1.043	1.037	1.028	1.046	1.048	1.020

Table 2: Optimal value and approximation ratio for instances $(|V|, k) = (5, 5)$ with various optimal values.

Optimal value	0.55	0.70	0.92
Approximation ratio	1.037	1.002	1.000

4.3 Regular 2-Prover 1-Round Game

4.3.1 Definition

Definition 3 Let "r-regular 2-Prover 1-Round Game" denote a 2-Prover 1-Round Game (Definition 1) such that every relation $\pi_{e(v,w)}$ have rk pairs and there are exactly r labels of v which satisfy e for every label of w. Naturally, there are exactly r labels of w which satisfy e for every label of v. In other words, every bipartite graph describing the relation is r-regular bipartite graph.

Unique Game is also "1-regular 2-Prover 1-Round Game." In addition, every r-regular 2-Prover 1-Round Game can be translated into Unique Game by decomposing every edge to r parallel edges. In this regard, the value of every solution of original regular 2-Prover 1-Round Game is r times as much as such Unique Game's.

4.3.2 Result

We generated instances of Unique Game from r-regular 2P1R Game and measured approximation ratios. The number of parallel edges between every two vertices is r. The way to calculate the average of approximation ratios is same as 4.1.

Figure 3 shows that in this case, the size of a labels set may make instances hard to approximate endlessly. Furthermore, it also shows that the number of edges decrease more, the instance become harder.

Figure 4 shows that there may be a "borderline". If r is greater than or equals to 3, the size of a labels set may make instances hard to approximate endlessly. Otherwise it may not do. We can find such borderlines in Figure 5 and Figure 6. In fact, these borderlines were determined by optimal values of original regular 2P1R Game. In the case of Figure 4, if r was greater than or equals to 3, the optimal value of original regular 2P1R Game was 1 and otherwise less than 1.

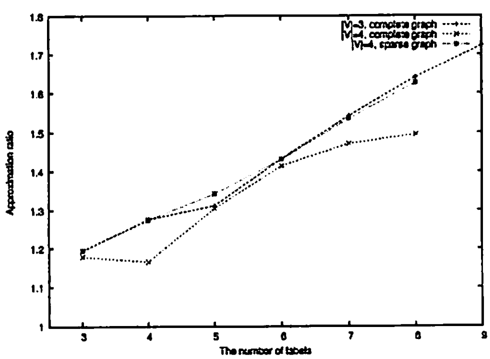

Figure 3: Graph with maximal approximation ratio on the y-axis and size of a labels set on x (instances from various regular 2P1R Game).

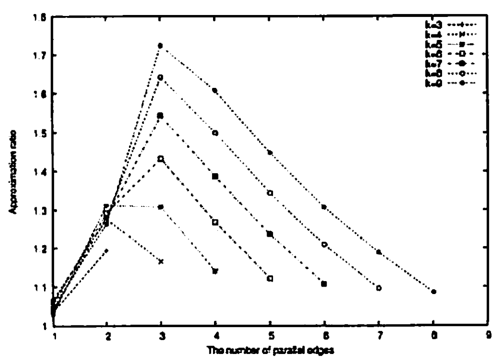

Figure 4: Graph with approximation ratio on the y-axis and r on x. (instances from r-regular 2P1R Game, $|V| = 3$, complete graph)

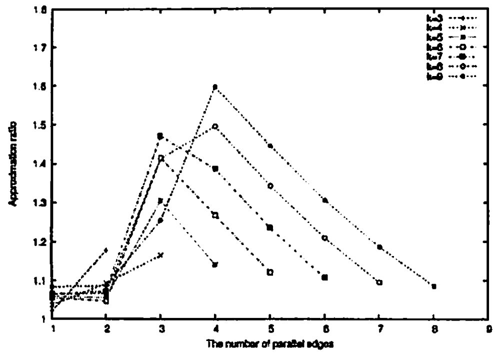

Figure 5: Graph with approximation ratio on the y-axis and r on x. (instances from r-regular 2P1R Game, $|V| = 4$, complete graph)

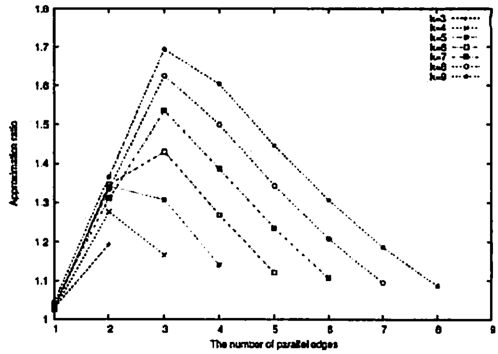

Figure 6: Graph with approximation ratio on the y-axis and r on x. (instances from r-regular 2P1R Game, $|V| = 4$, sparse graph)

The results show that the large optimal value of original regular 2P1R Game make instances of Unique Game hard to approximate. We could make such instances hard endlessly by increasing the number of labels. It means that we can realize Unique Games Conjecture.

However, r must be sufficiently large to make the optimal value large and large r makes the value of Unique Game small. We can not conclude that the results are evidence for Unique Games Conjecture. In addition, large k makes the value of Unique Game small and approximation ratio may be very small (Figure 5, $r = 3$). If it arises at every r and $|V|$, it does not suggest that Unique Games Conjecture is true.

There are too few results to understand the factors which make Unique Game hard. Nevertheless, there may be some difference between small r and large r.

5 Conclusions and Future Works

We have described the hardness of general instances of Unique Game. It shows that the size of the labels set is causing the hardness when there are parallel edges which come as decomposition of edges of regular 2P1R Game. Large size of a labels set and existence of such parallel edges might make instances hard to approximate endlessly, whereas in the case of instances which have large value such that parallel edges can not exist, we do not find the way to make it hard.

However, the results of this study cannot be taken as neither evidence nor contrary evidence for Unique Games Conjecture, because our sample was very small. The run-time of the program is too long to experiment with the large instances. The size of a instance's input is caused by the size of not only the vertices set but also the labels set. Additionally, the semidefinite programming is difficult to solve practically in large scale. Even if it can be solved in polynomial time, the order and the coefficient are too large. Therefore, the available graph of Unique Game is smaller than other graph optimization problems.

Our next step will be to experiment with improved program to solve semidefinite programming problems. Almost all well-known algorithms for Unique Game are based on SDP-relaxation. Therefore, improvement of the algorithm for semidefinite programming is important. By them, we can obtained results in large scale and analyze the hardness of Unique Game.

Another future research will be on proving Unique Games Conjecture by theoretical analysis or invention of new approximation method which disprove Unique Games Conjecture. It might be applied to the research that existence of parallel edges make Unique Game hard to approximate.

6 Acknowledgments

We thank professor Imai for helpful suggestion. And we are grateful to the member of Imai laboratory for useful discussions.

References

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, Vol. 5, No. 1, pp. 13–51, 1995.

[2] S. Arora and S. Safra. Probabilistic checking of proofs; a new characterization of np. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science (FOCS'92)*, pp. 2–13, 1992.

[3] U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, pp. 634–652, 1998.

[4] U. Feige and L. Lovász. Two-prover one-round proof systems: their power and their problems. In *Proceedings of the 24th Annual ACM Symposium on Theory of Computing (STOC'92)*, pp. 733–744, 1992.

[5] U. Feige and D. Reichman. On systems of linear equations with two variablesper equation. In *Proc. of APPROX-RANDOM'04*, pp. 117–127, 2004.

[6] Katsuki Fujisaway, Masakazu Kojima, Kazuhide Nakata, and Makoto Yamashitaz. SDPA (semidefinite programming algorithm) user's manual — version 6.2.0, 2004.

[7] M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, pp. 1115–1145, 1995.

[8] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science (FOCS'96)*, pp. 627–636, 1996.

[9] J. Håstad. Some optimal inapproximability results. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC'97)*, pp. 1–10, 1997.

[10] S. Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC'02)*, pp. 767–775, 2002.

[11] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for max-cut and other 2-variable csps? In *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)*, pp. 146–154, 2004.

[12] S. Khot and O. Regev. Vertex cover might be hard to approximate to within $2-\varepsilon$. In *Proceedings of the 18th Annual IEEE Conference on Computational Complexity (CCC'03)*, pp. 379–386, 2003.

[13] S. Khot and N. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into l_1. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pp. 53–62, 2005.

[14] Y. Nesterov and A. Nemirovskii. *Interior Point Polynomial Algorithms in Convex Programming.* SIAM, Philadelphia, PA, 1994.

[15] M.L. Overton. On minimizing the maximum eigenvalue of a symmetric matrix. *SIAM J. Matrix Anal. Appl*, Vol. 9, No. 2, pp. 256–268, 1995.

[16] L. Trevisan. Approximation algorithms for unique games. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pp. 197–205, 2005.