Extension Complexity via Lifting LP Hierarchy Lower Bounds

Xiangyu Guo

Nov 19, 2019

Abstract

Extension complexity measures the minimum size of all possible LP formulations that characterize a given polytope. But such lower bounds are generally not very easy to establish. On the other hand, LP hierarchies are *restricted* kinds of LP formulation, which is well-structured and thus relatively easier to analyze and prove size lower bound for them. In the past decade there's a line of research that *lift*s existing hierarchy lower bound to general LP size lower bound. This has resulted in the breakthrough result of [CLRS13], which showed that approximating maximum constraint satisfaction problems requires quasi-polynomial size LPs. Their result is later improved to sub-exponential by [KMR17]. Based on their results, people are able to prove extension complexity lower bounds for a much richer set of problems. In this note we'll expalain the result of [CLRS13] and some subsequent ones, in particular, an extension complexity lower bound for approximating VERTEX-COVER [BFPS15].

We'll also briefly discuss a very related line of research — SDP extension complexity, which studies the minimum size of SDP formulations. One of the most important paper in the recent years proves a quasi-polynomial SDP size lower bound for approximate MAX-3SAT and sub-exponential SDP size lower bound for the cut, TSP, and stable set polytopes[LRS15]. Remarkably, its overall strategy almost parallels [CLRS13], only lifting from SDP hierarchy lower bounds.

1 Introduction

Linear Programming (LP) is a prominent tool used in combinatorial optimization, especially for designing approximation algorithms. Due to the common belief of $\mathsf{P} \neq \mathsf{NP}$, for many NP-hard problems we don't expect there to be polynomial-size LP that approximates the optimal arbitrarily. So a natural question to ask is: how well can one approximate using small LP? Let's take MAX-CUT as an example. Given a graph $G = (V, E)$, we want to find the maximum cut in this graph. One classical LP relaxation for this problem is as follows:

$$\begin{aligned} \max_{y \in \mathbb{R}^{\binom{V}{2}}} \quad & \sum_{uv \in E} y_{uv} \qquad (1) \\ \text{s.t.} \quad & y_{uv} \le y_{uw} + y_{wv},\ \forall\, u, v, w \in V \\ & y_{uv} + y_{uw} + y_{vw} \le 2,\ \forall\, u, v, w \in V \\ & 0 \le y_{uv} \le 1,\ \forall\, u, v \in V \end{aligned}$$

Here each y_{uv} can be viewed as a indicator variable telling that whether edge (u, v) is in the cut. This LP is known to have integrality gap 2, which only matches the trivial algorithm that randomly partition the vertex set V into two parts. But (1) is just one possible LP formulation for MAX-CUT, and it's reasonable to ask: is there a clever LP formulation that beats the integrality gap 2, i.e., achieving $2 - \alpha$ for some absolute constant $\alpha > 0$? The *extension complexity* addresses such problems by giving negative answers like this: your LP cannot be too clever in the sense that its *size* must be really huge to beat some integrality gap. Here the size of a LP formulation is defined as the number of linear inequalities needed to specify this formulation. Intuitively, this is is the number of facets of the corresponding polytope.

Extension complexity are unconditional in the sense that it doesn't depend on any complexity assumptions like $\mathsf{P} \neq \mathsf{NP}$. Because such lower bound claims the limit for all possible LP formulations, it can often be hard to establish. But recent years' research showed that for the *maximum constraint satisfaction problems (MaxCSP)*, we only need to prove lower bound for formulations arising from the Sherali-Adams hierarchy[SA90]. This makes it possible to exploit many existing hierarchy lower bounds of MaxCSPs. The focus of this survey will be Chan *et al.*'s work[CLRS13], which first relates Sherali-Adams lower bound to general LP lower bound, showing that for MaxCSPs, the Sherali-Adams relaxation is "complete" among all polynomial-size LPs. Its idea has inspired several important follow-ups, including the *SDP extension complexity* lower bound for MaxCSP and some important families of polytopes[LRS15]. Chan *et al.*'s results also made it possible to prove extension complexity results by reducing from MaxCSPs. In particular, Bazzi *et al.*[BFPS15] proved a lower bound for approximating VERTEX-COVER within $2-\epsilon$ using reduction techniques developed in [BPZ15], which will be another focus of this survey.

Extension complexity. In its original form, extension complexity measures the LP size needed to capture some polytope *exactly*. Usually this polytope is the convex hull of all integral solutions for some discrete optimization problem: taking the MAX-CUT as example, its feasible solutions set is $S = \{y_{uv} \in \{0,1\} | \{y_{uv}\}_{uv \in E} \text{ satisfies (1)}\}$, and we'd like to know the smallest number of linear inequality needed to describe $\operatorname{conv}(S)$. If we can find a polynomial-size LP for $\operatorname{conv}(S)$, then we can solve MAX-CUT exactly in polynomial time, which is one popular way people sought to prove $\mathsf{P} = \mathsf{NP}$ in the early days. These efforts are heavily wrecked after Yannakakis[Yan91] proved that any *symmetric* LP for the TSP polytope has exponential size. The key idea in [Yan91] (which also lays the foundation for all subsequent works) is a *factorization theorem* that estabishes the equivalence between a polytope's extension complexity with the *non-negative rank* of its *slack matrix*. Here, the slack matrix M_P of a polytope P is defined as follows: each row corresponds to a supporting hyperplane (facet) F of P in the form of $F(x) = b - \langle a, x \rangle$, and each column corresponds to a vertex v of P, and $M_P(F, v) = F(v)$ measures the algebraic distance from v to F.

By analyzing a special family of slack matrices, Fiorini *et al.*[FMP$^+$12] was able to strengthen Yannakakis' result to include the *asymmetric* LPs, showing that the TSP polytope requires $2^{\Omega(n^{1/4})}$ LP to describe. Later Braun *et al.*[BFPS12] generalize the definition of extension complexity to approximation problems, and show that approximating MAX-CLIQUE within $O(n^{1/2-\epsilon})$ requires LPs of size $2^{\Omega(n^\epsilon)}$. Braverman and Moitra[BM13] also analyze this family of slack matrices, and building on the work of [BFPS12], they show that approximating MAX-CLIQUE within $O(n^{1-\epsilon})$ requires LP of size $2^{\Omega(n^\epsilon)}$.

Finally, we want to note the difference between the extension complexity and the computation complexity of a problem: a problem with exponential extension complexity can admit polynomial-time algorithms; and even an exponential-size LP may still be polynomial-time solvable, e.g., using the Ellipsoid Method. The most prominent example may be the perfect matching problem: a breakthrough results of [Rot14] proves that MATCHING polytope has exponential extension complexity, and later it's shown that even semidefinite programming (SDP) may not be powerful enough to catch this polytope [BBH$^+$17](but only for the symmetric case). However, the MATCHING problem is well-known to be in P.

LP hierarchies. We'll mostly use lower bound results of the Sherali-Adams hierarchy in this survey, but there also exists other important kinds of hierarchy. We refer the readers to Laurent's excellent survey [Lau03] for more discussions about different hierarchies and their relationship. The hierarchy technique has produced significant improvement over the approximation ratio for a few important problems, and we refer the readers to surveys like [Chl07, Rot11, BS14] for more information about these algorithmic results.

In this article we care more about the negative results, namely, the hierarchy lower bound. This line of research is initiated by Arora *et al.*[ABL02], who studied integrality gap results for LP relaxations arising from Lovász-Schrijver (LS) hierarchy[LS91]; specifically, they showed one needs $\Omega(\log n)$ round of LS lifting to approximate VERTEX-COVER within $2 - o(1)$. The hierarchy lower bound results of the MaxCSPs will be particularly useful for us, and we list a few of them here: Fernández de la Vega and Mathieu[dlVK07] showed that MAX-CUT has integrality gap $1/2 + \epsilon$ even after k-round Sherali-Adams lifting, for any fixed ϵ and k. This was later strengthened to n^δ-round lower bound by [CMM09].

Some other works proved lower bounds for the Sum-of-Square (SoS) SDP hierarchy, which is stronger than Sherali-Adams hierarchy, and therefore also imply Sherali-Adams lower bounds at least as same strong: Grigoriev[Gri01] showed MAX-3XOR requires $\Omega(n)$-round SoS (thus also for Sherali-Adams) to achieve $(1/2+\epsilon, 1-\epsilon)$-approximation. This was later rediscovered by Schoenebeck[Sch08], who also noticed that this implies a $\Omega(n)$-round SoS lower bound for $(7/8+\epsilon, 1-\epsilon)$-approximating MAX-3SAT.

Organization: The rest of this survey will be organized as follows: first we'll give some basic definitions about LP and MaxCSP, then introduce Yannakakis' Factorization Theorem, and in particular, its approximate form for MaxCSPs. In section 4 we'll explain the result of [CLRS13] in detail, giving the quasi-polynomial lower bound for some MaxCSPs. Then in section 5 we describe the lower bound of $(2-\epsilon)$-approximating VERTEX-COVER, based on the reduction technique of [BPZ15, BFPS15, BPR18]. Finally, we briefly survey some recent results on SDP extension complexity in section 6.

2 Preliminaries

Linear Programming (LP). Linear programming (LP) describes a broad class of optimization problems, whose objective and constraints are all linear functions. Formally, a LP formulation can be written as follows

$$\max_{x\in\mathbb{R}^n} c^\top x \tag{2}$$
$$\text{s.t. } Ax \le b, \tag{3}$$
$$x \ge 0, \tag{4}$$

where $c \in \mathbb{R}^n, A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$. We also use $\mathcal{LP}(\mathcal{P}, c)$ to denote the formulation 2, where $\mathcal{P} = \{x : Ax \ge b, x \ge 0\}$ is the set of feasible solutions. We note that there're many equivalent definitions of LP, but we will stick to 2 in this article. It's well-known that LP can be solved in time polynomial in m and n. In most problems we care about, there's $n = O(m)$, so we define the *size* of a LP formulation to be the number of constraints m.

Typically we're focusing on discrete optimization problems with integral solutions. A LP relaxation[1] for such a problem is a polytope that *contains* the convex hull of all its integral feasible solutions. The natural strategy for finding the LP relaxation is to first write down a *integer-linear program* (ILP) that characterize the integer solutions exactly, then relax its solutions to be real numbers. If the relaxed polytope is not too "loose" than the integral convex hull, one can expect to derive a good approximation via the relaxation. To measure the LP relaxation's ability of approximation, we define *integrality gap* to be the maximum ratio between the solution quality of the integer program and of its relaxation.

Constraint Satisfaction Problem (CSP). A constraint satisfaction problem (CSP) is denoted by a tuple $\Pi = (\Omega, \Phi)$, where Ω is the *domain* and Φ is a set of *predicates* $\{\psi : \Omega^r \mapsto \{0, 1\}\}$. Here r is called the *arity* of the CSP. Given CSP $\Pi = (\Omega, \Phi)$, an instance $\mathcal{I}$ of the maximum constraint satisfaction problem (MAX-CSP) MAX-Π is defined as follows: $\mathcal{I}$ contains a variable set $V = \{X_1, X_2, \ldots, X_n\}$ and a set of *constraints* $E = \{P_1, P_2, \ldots, P_m\}$, where each $P_i : \Omega^n \mapsto \{0, 1\}$ is associated with a ordered subset of coordinates $S_i \subset [n], |S_i| = k$ and some predicate $P \in \Phi$, such that for any $\mathbf{a} = (a_1, \ldots, a_n) \in \Omega^n$,

$$P_i(\mathbf{a}) = P(a_{i_1}, \ldots, a_{i_k}),\ S_i = (i_1, i_2, \ldots, i_k), \text{ where all } i_j\text{'s are distinct}$$

The objective of $\mathcal{I}$ is to find a value assignment $F : V \mapsto \Omega$ so as to optimizing the following problem:

$$\max_F \frac{1}{m}\sum_{i=1}^{m} P_i(F(X_1), F(X_2), \ldots, F(X_n)) \tag{5}$$

[1] Later on we'll use "LP formulation" and "LP relaxation" interchangeably, as the later just refers to a LP formulation associated with some discrete optimization problem.

Note the value assignment F is just some $x \in \Omega^n$, so we abuse notations and let $\mathcal{I}(x) = \frac{1}{m}\sum_{i=1}^m P_i(x)$. In a more general definition, each constraint can have a weight w_i and the objective (5) becomes a weighted sum.

Example 1. *Consider the boolean CSP with $\Omega = \{0, 1\}, \Phi = \{\neq\}$. Then the* MAX-CSP *instance $\mathcal{I} = (V, E)$ can be viewed as a graph with vertex set V, and each constraint $P_k \in E$ is an edge over some vertices i, j. The objective is to assign 0-1 labels to each vertex, so as to maximizing the number of edges with different ending labels. This is exactly the* MAX-CUT *problem.*

Example 2. *Now consider the CSP with $\Omega = [q], \Phi = \{\ell_i : [q] \times [q] \mapsto \{0, 1\}\}$, where each ℓ_i is defined via some permutation π_i on $[q]$, such that $\ell_i(u, v) = 1$ iff $\pi(u) = v$. Similar to the previous example, the* MAX-CSP *instance $\Pi = (V, E)$ can be viewed as a graph, and the objective is to assign labels in $[q]$ to each vertex, so as to maximizing the number of satifying edges. This is exactly the* UNIQUE-GAMES *problem.*

LP Relaxations for max-csp. To write a LP relaxation for MAX-CSP problems, we need to first linearize it. Here we follow the convention in [CLRS13]. For any CSP Π, let MAX-Π_n be the set of all MAX-Π instances over n variables. For now we will focus on boolean CSPs, so we can assume the variable set for MAX-Π_n is the n-dimension boolean cube $\{-1, +1\}^n$.

Linearization For each n we pick some fixed $D \in \mathbb{N}$, and associate every $\mathcal{I} \in$ MAX-Π_n with a vector $\tilde{\mathcal{I}} \in \mathbb{R}^D$, every $x \in \{-1, +1\}^n$ with a vector $\tilde{x} \in \mathbb{R}^D$. And the key requirement is that

$$\mathcal{I}(x) = \langle \tilde{\mathcal{I}}, \tilde{x} \rangle \text{ for all } \mathcal{I} \in \text{MAX-}\Pi_n, x \in \{-1, +1\}^n. \tag{6}$$

Feasible Region The feasible region is a closed convex polyhedron $\mathcal{P} \subset \mathbb{R}^D$ described by r linear inequality constraints, which contains all linearization of integral solutions, i.e.,

$$\forall\, x \in \{-1, +1\}^n, \text{ there's } \tilde{x} \in \mathcal{P}, \tag{7}$$

(c, s)-approximation For constant $1 \geq c > s \geq 0$, we say a LP relaxation $\mathcal{L}_n := \mathcal{LP}(\mathcal{P}, \tilde{\mathcal{I}})$ achieves (c, s)-approximation if for any $\mathcal{I} \in$ MAX-Π_n with $\mathsf{opt}(\mathcal{I}) \leq s$, the optimal solution of $\mathcal{L}_n$ has value no more than c. Formally, let $\mathcal{L}_n(\mathcal{I}) := \max_{y \in \mathcal{P}} \langle y, \tilde{\mathcal{I}} \rangle$ denote the optimal value of $\mathcal{L}_n$, we want

$$\mathsf{opt}(\mathcal{I}) \leq s \implies \mathcal{L}_n(\mathcal{I}) \leq c. \tag{8}$$

Fourier Analysis for Boolean Functions. We will mostly deal with functions defined on the boolean cube $\{-1, 1\}^n$. We will use some basic facts in Fourier analysis; for a more comprehensive treatment, we refer the readers to the textbook by O'Donnell[O'D14]. Every function $f : \{-1, 1\}^n \mapsto \mathbb{R}$ can be expressed as a multilinear polynomial as follows:

$$f(x) = \sum_{\alpha \subseteq [n]} \hat{f}(\alpha) \cdot \chi_\alpha(x)$$

where $\chi_\alpha(x) := \prod_{i \in \alpha} x_i$ is called *Fourier basis*, and

$$\hat{f}(\alpha) = \langle f, \chi_\alpha \rangle := \mathbb{E}\left[f \cdot \chi_\alpha\right] = \frac{1}{2^n} \sum_{x \in \{-1,1\}^n} f(x)\chi_\alpha(x).$$

The *degree* of f is defined to be the degree of its Fourier expansion, i.e., the maximum d such that there exists $|\alpha| = d$ and $\hat{f}(\alpha) \neq 0$.

In this article we're mostly interested in one family of boolean functions, *juntas*, which has the following simple structure: the output of a k-junta f depends only on at most k coordinates of the input, i.e., there exists some $S \subseteq [n], |S| \leq k$, s.t. $f = \sum_{\alpha \subseteq S} \hat{f}(\alpha) \cdot \chi_\alpha$. Apparently by definition, a

k-junta is at most degree-k, but a low-degree function may not necessary be a simple junta: consider the function $f(x) = \sum_{i=1}^{n} x_i$, which has degree 1 but is a n-junta.

Sometimes it's more convenient to define a problem over $\{0,1\}^n$ rather than $\{-1,+1\}^n$. But we can easily switch between these two cubes with the bijection $x \mapsto 1-2x$. Therefore for any $g : \{-1,1\}^n \mapsto \mathbb{R}$, we can identify it with a equivalent $f : \{0,1\}^n \mapsto \mathbb{R}$ defined as $f(x) = g(1-2x)$, and vice versa. This bijection also preserves the degree of variables, thus a degree-d function on one cube corresponds to a function on the other cube of the same degree. Therefore we won't differentiate between these two cubes in the rest of the article.

3 Factorization Theorems

The factorization theorem relates the *extension complexity* of a polytope to the *nonnegative rank* of a specific matrix. This converts the task of searching a set of infinitly many LP formulations into analyzing one single matrix, which is much more concrete to work with. In this section we'll formally define all the concepts mentioned in the previous sentence, and state the Factorization Theorem.

Extended Formulations. Given polytope P, we're interested in whether it can be obtained by projecting from some *simpler* polytope Q in higher dimensions. Q is called an *extended formulation* of P, which is formally defined as follows

Definition 1 (Extended Formulation)**.** *Given a polytope $P = \{x \in \mathbb{R}^n : Ax \leq b\} = \mathrm{conv}(V)$ with $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$, we say polytope $Q = \{(x,y) \in \mathbb{R}^{n+r} : Ex + Fy = g, y \in \mathbb{R}^r, y \geq 0\}$ is a extended formulation of P if $P = \pi_x(Q) := \{x \in \mathbb{R}^n : \exists y \in \mathbb{R}^r \text{ s.t. } (x,y) \in Q\}$. The* size *of Q is the number of its inequality constraints ($y \geq 0$), i.e. r.*

The extension complexity of P is the minimum size among all its possible extended formulations, denoted as $\mathsf{xc}(P)$.

Factorization Theorem. First we formally define the *slack matrix* of a given polytope, which (intuitively) characterize the distance of each vertex to each facet of the polytope.

Definition 2 (slack matrix)**.** *For a given polytope $P = \{x \in \mathbb{R}^n : Ax \leq b\} = \mathrm{conv}(V)$, suppose $A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m, V = \{v_1, \dots, v_k\}$. The slack matrix of P is defined as*

$$S_P(i,j) := b_i - \langle A_i, v_j \rangle \tag{9}$$

The Factorization Theorem establishes the equivalence between $\mathsf{xc}(P)$ and S_P's non-negative rank, which is defined as follows:

Definition 3 (Nonnegative rank)**.** *Given a nonnegative matrix $M \in \mathbb{R}^{m\times n}_{\geq 0}$, its nonnegative rank $\mathsf{rank}_+(M)$ is the minimum r such that there exists $U \in \mathbb{R}^{m\times r}_{\geq 0}, V \in \mathbb{R}^{r\times n}_{\geq 0}$ satisfying $M = UV$.*

Remark: By definition, the slack matrix S_P is of dimension $m \times k$, where m is the number of facets and k is the number of vertices. But for our purpose we can actually extend it to a infinite matrix: $\forall\, x \in P$, we can add a column in S_P for x; similarly for any valid inequality $b - \langle a, x \rangle \geq 0$ on P, we can add a row for it. This would *not* change the non-negative rank of S_P, since by Farkas' Lemma, all such rows (columns) can be obtained by a non-negative linear combination of facets (vertices). This property gives us much freedom when analyzing S_P, as in most situation S_P is not given explicitly and we don't know exactly what each facet looks like.

Now we formally state the Factorization Theorem:

Theorem 1 (Yannakakis' Factorization Theorem[Yan91])**.** *For a given polytope P, its slack matrix S_P has a rank-r non-negative factorization* if and only if *P has a size-r extended formulation.*

Remark: With the Factorization Theorem, to prove a lower bound for $\mathsf{xc}(P)$, we only need to prove a lower bound for $\mathsf{rank}_+(S_P)$. While in practice, we usually focus on proving a lower bound for some more structured sub-matrix M of S_P, because a lower bound for $\mathsf{rank}_+(M)$ also applies to $\mathsf{rank}_+(S_P)$.

Factorization Theorem for Approximation. [BFPS12] first generalize definition of extended formulation to approximate problems by using nested polyhedron pairs, and define the extension complexity accordingly. For simplicity, we'll restrict the case to MAX-CSP problems. Suppose P is a polytope that achieves (c, s)-approximation for MAX-Π_n, as defined in (8). We define the slack matrix $M_{c,s}^{n,\Pi} \in \mathbb{R}^{\text{MAX-}\Pi_n \times \{0,1\}^n}$ as follows

$$M_{c,s}^{n,\Pi}(\mathcal{I}, x) = c - \mathcal{I}(x) \quad \forall\, x \in \{-1,1\}^n \text{ and } \mathcal{I} \in \text{MAX-}\Pi_n \text{ with } \mathsf{opt}(\mathcal{I}) \le s. \tag{10}$$

And the factorization theorem for MAX-Π_n is the following:

Theorem 2 (LP Factorization Theorem for Approximate MaxCSP[CLRS13])**.** *Given a* MAX-CSP *instance* MAX-Π_n *over n variables, there exists a LP relaxation of size R that (c, s)-approximates* MAX-Π_n *if and only if* $\mathsf{rank}_+\left(M_{c,s}^{n,\Pi}\right) \le R$

Remark: Here we are focusing on MAX-CSP problems, while it's also similar to define the approximate extended formulation for minimization problems. In section 5, we'll see a more general definition for approximate LP formulations subsuming that of [BFPS12], and use it to prove hardness result for VERTEX-COVER.

4 Lifting Lower Bounds from LP Hierarchy

The main theme of this section is to present the result of [CLRS13]: on (boolean) MaxCSPs, one can *lift* the hierarchy lower bound to general LP size lower bound[2], by showing that any LP formulation of similar size cannot perform *much better* than formulations produced by hierarchies. In the following we'll first introduce the Sherali-Adams hierarchy[SA90], then explain the overall strategy of [CLRS13], and discuss the implication and limits of their result.

4.1 Sherali-Adams Hierarchy

Sherali-Adams Lifting. One natural definition of Sherali-Adams hierarchy is to view it as a way to lift a base LP relaxation. The resulted LP formulation is called a *Sherali-Adams lifting.*

Definition 4 (Sherali-Adams lifting)**.** *Assume we're dealing with some $\{0,1\}$-optimization problem. Given a base LP relaxation $\mathcal{L} = \mathcal{LP}(c, \mathcal{P})$ with $\mathcal{P} = \{x \in \mathbb{R}^n : Ax \ge b, x \ge 0\}$, its d-round Sherali-Adams lifting $\mathsf{SA}_d(\mathcal{P})$ is defined as follows:*

1. *New constraints: for each constraint $a_i^T x \ge b_i$, and every $S, T \subseteq [n]$ such that $S \cap T = \emptyset, |S| + |T| \le d - 1$, introduce new constraint*
$$(a_i^\top x - b_i) \prod_{i \in S} x_i \prod_{j \in T} (1 - x_i) \ge 0$$
2. *Linearization: Expand the product in each new constraint, and for each monomial $\prod_{i\in\alpha} x_i$ introduce a new variable y_α to replace it. Then all constraints will become linear constraints for y. Finally, add the constraint $y_\emptyset = 1$*

And the final relaxation we get consists of the following constraints:

$$y_\emptyset = 1, \tag{11}$$

$$\sum_{T' \subseteq T} (-1)^{|T'|} y_{S \cup T'} \ge 0, \quad \forall\, S \cap T = \emptyset, |S| + |T| \le d \tag{12}$$

$$\sum_{T' \subseteq T} (-1)^{|T'|} \left(\left(\sum_{j \in [n]} a_{ij} y_{S \cup T' \cup \{j\}} \right) - b_i y_{S \cup T'} \right) \ge 0, \quad \forall\, i \in [m], S \cap T = \emptyset, |S| + |T| \le d - 1 \tag{13}$$

To get a solution of the original relaxation $\mathcal{L}$, just project y back as $x_i = y_{\{i\}}$.

[2] We'll use "extension complexity lower bound" and "LP size lower bound" interchangeably.

Since the variable $\{y_\alpha : \alpha \subseteq [n]\}$ reside in a higher-dimension space, this is also called *lift-and-project* method. It's easy to see that the new relaxation has $n^{O(d)}$ many constraints as well as variables, thus can be solved in $n^{O(d)}$ time. Intuitively, each variable y_α models the correlation of $\{x_i : i \in \alpha\}$, and can be thought as the moments up to degree-d of the variables under some distribution. Indeed, we can define a *pseudo-expectation* $\tilde{\mathbb{E}}$ based on any feasible SA_d solutions:

$$\tilde{\mathbb{E}}\left[\prod_{i\in\alpha} x_i\right] = y_\alpha$$

Then $\tilde{\mathbb{E}}$ is a linear functional over $L^2(\{0,1\}^n)$. One can show that actually $\tilde{\mathbb{E}}$ and y uniquely determines each other, and we can rewrite our relaxation as a equivalent program for finding pseudo-expecations. This viewpoint is more convenient to use when we define the Sherali-Adams relaxation for MaxCSPs.

Sherali-Adams Relaxation for MaxCSP. The MaxCSP problem is defined on $\{-1,1\}^n$ instead of $\{0,1\}^n$, but as said before, we can switch between the two cubes while preserving the degree of polynomials, so this doesn't affect the size of the relaxation. Another important distinction is that we will use one *canonical relaxation* for all MaxCSPs, while in Definition 4, we may get different relaxations when lifting from different base LP formulations. However, one can show that this will only result in quadratic difference in the number of rounds: given a d-round canonical SA relaxation (will be defined later), one can easily find a tighter $\Theta(d^2)$-round SA lifting from any naturaly base formulation, and vice versa. For more details, see Appendix A of [CLRS13].

We define the SA relaxation for MaxCSPs from the pseudo-expectation view. A feasible solution of the d-round Sherali-Adams relaxation for $\textsc{max}\text{-}\Pi_n$ is a linear functional $\tilde{\mathbb{E}}$ on $L^2(\{-1,1\}^n)$ which satisfies $\tilde{\mathbb{E}}\mathbf{1} = 1$ and $\tilde{\mathbb{E}}P \geq 0$ for any nonnegative d-junta P. By the self-duality of $L^2(\{-1,1\}^n)$, we can also view $\tilde{\mathbb{E}}$ as a *function* $\mu \in L^2(\{-1,1\}^n)$, such that $\tilde{\mathbb{E}}f = \langle \mu, f\rangle$, $\langle \mu, \mathbf{1}\rangle = 1$, and $\langle \mu, P\rangle \geq 0$ for all nonnegative d-junta P. Analogously we call μ as a *degree-d pseudo-distribution*, although it's essentially the same object as $\tilde{\mathbb{E}}$. Since μ is uniquely determined by its Fourier coefficients $\{\hat{\mu}(\alpha) = \langle \mu, \chi_\alpha\rangle : \alpha \subseteq [n]\}$, if we further require $\hat{\mu}(\alpha) = 0$ for all α with $|\alpha| > d$, we have

$$\mu = \sum_{\alpha:|\alpha|\leq d} \hat{\mu}(\alpha)\chi_\alpha = \sum_{\alpha:|\alpha|\leq d} \tilde{\mathbb{E}}\left[\chi_\alpha\right]\chi_\alpha$$

There's an alternative view of the degree-d pseudo-distribution μ: a *separating functional* between a given function $f : \{-1,1\}^n \mapsto \mathbb{R}$ and $\mathcal{J}_d$, the cone generated by all nonnegative d-juntas. If $f \in \mathcal{J}_d$ then there's $\langle \mu, f\rangle \geq 0$, otherwise $\langle \mu, f\rangle < 0$.

With the above definitions, we can easily derive the following properties of $\tilde{\mathbb{E}}$:

Claim 1. *Let $\tilde{\mathbb{E}}$ be a degree-d pseudo-expectation , then*

(i) $\tilde{\mathbb{E}}$ preserve the nonnegativity for low-degree juntas: For any non-negative d-junta $f : \{-1,1\}^n \mapsto \mathbb{R}_{\geq 0}$, $\tilde{\mathbb{E}}f \geq 0$.

(ii) $\|\tilde{\mathbb{E}}\|_\infty$ is bounded: For any $\alpha \subseteq [n]$, $|\tilde{\mathbb{E}}\chi_\alpha| \leq 1$, thus $\|\tilde{\mathbb{E}}\|_\infty \leq \sum_{i=0}^{d}\binom{n}{i}$.

Finally, the *d-round Sherali-Adams value* of a $\textsc{max}\text{-}\Pi_n$ instance $\mathcal{I}$ is defined as

$$\mathsf{sa}_d(\mathcal{I}) := \max_{\text{degree}-d \text{ pseudo expectation } \tilde{\mathbb{E}}} \tilde{\mathbb{E}}\left[\mathcal{I}\right]$$

4.2 Technique Overview

We give a overview of the strategy used in [CLRS13]. Recall the definition of slack matrix $M^{n,\Pi}_{c,s}$ for a MaxCSP problem $\textsc{max}\text{-}\Pi_n$:

$$M^{n,\Pi}_{c,s}(\mathcal{I}, x) = c - \mathcal{I}(x) \quad \forall\, x \in \{-1,1\}^n \text{ and } \mathcal{I} \in \textsc{max}\text{-}\Pi_n \text{ with } \mathsf{opt}(\mathcal{I}) \leq s.$$

By Theorem 2, to prove there's no small LP formulation that achieves (c,s)-approximation, it's equivalent to proving $M^{n,\Pi}_{c,s}$ has large nonnegative rank. The key idea of our proof can be summarized in one sentence: a low-nonnegative-rank factorization of $M_{c,s}^{n,\Pi}$ will imply a low-degree (approximate) Sherali-Adams solution. This would contradict with existing Sherali-Adams lower bounds for MaxCSP, thus proving $\mathsf{rank}_+(M_{c,s}^{n,\Pi})$ must be large. Specifically, the proof consist of two parts:

1. **Existence of simple certificates.** Suppose $\mathsf{rank}_+(M_{c,s}^{n,\Pi})$ has a rank-R nonnegative factorization, i.e., there is a set of R nonnegative functions $q_i : \{-1,1\}^n \mapsto \mathbb{R}_{\geq 0}$ such that for any $\mathcal{I} \in \text{MAX-}\Pi_n$ we have

$$c - \mathcal{I}(x) = \sum_{i=1}^{R} \lambda_i q_i(x) + \lambda_0 \tag{14}$$

   where all $\lambda_i \geq 0$ only depend on $\mathcal{I}$. Here we can assume for example $R \leq n^{d/2}$, because we know SA_d has size $n^{\Theta(d)}$ and we want to prove any LP $\mathcal{L}$'s size cannot be too smaller than this. One can show that there's a way to choose a subset of q_i's such that every chosen q_i is approximately a K-junta with $K = o(n)$, and furthermore, those not chosen has very small "magnitude" that can be ignored.

2. **Random restriction**: To use SA_d lower bound, we need further simplify all q_i's to d-juntas. We use the idea of *random restriction*[3]: For each q_i, if we randomly pick a small subset of coordinates $S \subset [n], |S| = m$ and restrict q_i to it (i.e., fix the input value in all other coordinates), then with hight probability q_i will become d-juntas. The "restriction" is achieved by randomly embedding a $\text{MAX-}\Pi_m$ instance into n variables to get a $\text{MAX-}\Pi_n$, s.t. any constraint is imposed only on (a subset of) S. With the assumption that R is not too large, we can take union bound for all $q_i, i \in [R]$, and deduce that there's a choice of S that makes all q_i's become d-junta simultaneously.

Now as all q_i's are approximately nonnegative d-juntas[4], the desired contradiction can easily be seen: since by assumption d-round SA lift cannot achieve (c, s)-approximation, the optimal value of $\mathsf{sa}_d(\mathcal{I}) > c$, i.e, applying the corresponding degree-d pseudo-expectation $\tilde{\mathbb{E}}$ on the LHS of (14) will get some constant $\beta < 0$; On the other side, if we apply $\tilde{\mathbb{E}}$ on the RHS of (14), since all q_i's are close to nonnegative d-junta, we'll get a value $-\epsilon_n$ with some diminishing ϵ_n. Choosing n large enough leads to the contradiction.

4.3 A Quasi-polynomial Extension Complexity Lower Bound for MaxCSP

In this section we sketch the proof of the main result in [CLRS13], which can be summarized in the following theorem

Theorem 3 ([CLRS13])**.** *For k-ary MaxCSP, any general LP relaxation satisfying requirements* (6) *and* (7) *cannot be much stronger than Sherali-Adams relaxation in the sense that*

(i) In the polynomial regime, Sherali-Adams relaxation is the most powerful: for fixed $d \geq k, d \in \mathbb{N}$, if SA_d cannot achieve (c, s)-approximation for $\text{MAX-}\Pi$*, then no sequences of LP of size at most $n^{d/2}$ can achieve (c, s)-approximation for* $\text{MAX-}\Pi$.

(ii) In the super-polynomial regime, general LP cannot be exponentially stronger than Sherali-Adams relaxation: let $d : \mathbb{N} \mapsto \mathbb{N}$ be a monotone increasing function with $d(n) \leq n$. If $d(n)$-round Sherali-Adams relaxation cannot achieve (c, s)-approximation for $\text{MAX-}\Pi_n$*, then no LP relaxation of size $n^{d(n)^2}$ can achieve (c, s)-approximation for* $\text{MAX-}\Pi_N$*, where $N \leq n^{10d(n)}$.*

A straighforward calculation shows that Theorem 3(ii) implies quasi-polynomial extension complexity lower bound if the MaxCSP problem is hard for $n^{\Omega(1)}$-round Sherali-Adams relaxation, which

[3]The random restriction idea is heavily used in the analysis of boolean functions and circuit complexity, e.g. the famous Switching Lemma by Håstad.

[4]We'll see later that q_i actually only needs to be a *conical* d-junta, i.e., nonnegative linear combination of nonnegative d-juntas. This is easier to approximate and lead to better lower bound.

is the case for e.g., $(1-\epsilon, 1/2+\epsilon)$-approximating MAX-CUT, $(1-\epsilon, 7/8+\epsilon)$-approximating MAX-3SAT. In the rest of this section, we will focus on proving Theorem 3, and we'll organize the proof in the framework as described in section 4.2.

Existence of simple certificate. Consider the factorization (14): since all λ_i and q_i are nonegative, by scaling we can further assume that every q_i is a *density*, i.e.

$$\mathbb{E}\left[q_i\right] := \mathbb{E}_{x\in\{-1,1\}^n}[q_i(x)] = 1$$

We further define the *entropy* of a density as the entropy of its corresponding probability measure

$$H(q) := -\sum_{x\in\{-1,1\}^n} (2^{-n}q(x))\log_2(2^{-n}q(x))$$

Then we can apply the following Chang's Lemma[Cha02, IMR14] to each high-entropy q_i:

Lemma 1. *Let $h : \{1,1\}^n \mapsto \mathbb{R}_{\geq 0}$ be a density function and suppose it has entropy at least $n-t$ for some $t < n$, then for every $1 \leq d \leq n$ and $\gamma > 0$, there exists a set $J \subseteq [n]$ with*

$$|J| \leq \frac{2td}{\gamma^2} \tag{15}$$

such that for all subsets $\alpha \nsubseteq J$ with $|\alpha| \leq d$, we have $|\hat{h}(\alpha)| \leq \gamma$.

Intuitively, the lemma states that if a density function has high entropy, then its *low-degree* part can be approximated by a $|J|$-junta with error γ. There're several parameters need to be specified in Lemma 1, but at least we want a $|J| = o(n)$ and $\gamma = o(1)$, otherwise the result doesn't make much sense. For example, one can choose $\gamma \sim \left(\frac{td}{\sqrt{n}}\right)^{1/2}$, then $|J| \lesssim \sqrt{n}$. And we claim that this is all we need, because we can safely ignore the high-degre parts or those q_i with low-entropy:

1. Recall the property of degree-d pseudo-expectation $\tilde{\mathbb{E}}$: it evaluates to 0 for any monomial with degree higher than d. Thus when we apply $\tilde{\mathbb{E}}$ over a q_i, only its low-degree part — $\sum_{|\alpha|\leq d} \hat{h}_i(\alpha)\chi_\alpha$ — matters.
2. Suppose some density q_i has low-entropy $H(q_i) < n-t$, then there's some x' that makes $q_i(x') > 2^t$ by the definition of entropy. Now look at (14) with $x = x'$:

   $$c - \mathcal{I}(x') = \sum_{i=1}^{R} \lambda_i q_i(x') + \lambda_0$$

   Since the LHS is always less than or equal to 1 as $c \leq 1$, we conclude that $\lambda_i < 2^{-t}$. Thus by choosing a suitable t, we can easily control the error due to $\tilde{\mathbb{E}}q_i$.

Random Restriction. The idea is as follows: with Lemma 1 we can say that every high-entropy q_i has a $|J_i|$-junta approximation for its low-degree part, where $J_i \subset [n], |J_i| = o(n)$. Now if we randomly pick a subset S with $|S| = m = o(n)$, and restrict q_i to coordinates in S, then with large probability $|S \cap J_i| \leq d$. Based on *the fact that there're at most $R = n^{d/2}$-many q_i's*, we can apply union bound to show that there exists a set S s.t. $|S \cap J_i| \leq d$ for every $i \in [R]$.

Lemma 2. *For any $d \in \mathbb{N}$, let Q be a collection of densities $h : \{-1,1\}^n \mapsto \mathbb{R}_{\geq 0}$ such that each of them has entropy at least nt. If $|Q| \leq n^{d/2}$, then for all integers m with $3 \leq m \leq n/4$, there exists a set $S \subseteq [n]$ such that*

- $|S| = m$

- *For every $q \in Q$, there exists a set of at most d coordinates $J(q) \subset S$ such that all degre-d terms of q not contained in $J(q)$ are small. Quantitatively, we have*

$$|\hat{q}(\alpha)| \leq \left(\frac{16mtd}{\sqrt{n}}\right)^{1/2}, \forall \alpha \subset S, \alpha \nsubseteq J(q), |\alpha| \leq d$$

Bounding the approximation error. Now we put everything together: Let $m \leq n$ be parameters that will be chosen later. With the existence of the size-m set S as claimed in Lemma 2, our final MAX-Π_n instance is build as follows: take a (c,s)-hard instance $\mathcal{I}_0 \in$ MAX-Π_m for SA_d, i.e., $\mathsf{opt}(\mathcal{I}_0) \leq s$ but $\mathsf{SA}_d(\mathcal{I}_0) > c$, then "plant" $\mathcal{I}_0$ on the variable in S to get a MAX-Π_n instance $\mathcal{I}$; All variables outside S are dummy variables and are not imposed with any constraints. This way it's easy to see that $\mathsf{opt}(\mathcal{I}) = \mathsf{opt}(\mathcal{I}_0) \leq s$.

Let $\tilde{\mathbb{E}}_0$ be the SA_d solution for $\mathcal{I}_0$. We extend $\tilde{\mathbb{E}}_0$ to a solution $\tilde{\mathbb{E}}_S$ for $\mathcal{I}$:

$$\tilde{\mathbb{E}}_S[\chi_\alpha] = \begin{cases} \tilde{\mathbb{E}}_0[\chi_\alpha], & \text{if } \alpha \subseteq S \\ 0, & \text{otherwise.} \end{cases}$$

By definition $\tilde{\mathbb{E}}_S[\mathcal{I}] = \mathsf{sa}_d(\mathcal{I}_0)$, and it's easy to verify that $\tilde{\mathbb{E}}_S$ is a valid degree-d pseudo-expectation for functions over $\{-1,1\}^n$. Now we apply $\tilde{\mathbb{E}}_S$ to both sides of (14) to get.

$$c - \mathsf{sa}_d(\mathcal{I}_0) = \sum_{i=1}^{R} \lambda_i \tilde{\mathbb{E}}_S[q_i] + \lambda_0 \tag{16}$$

By assumption, the LHS of (16) is bounded away from 0, i.e., less than some constant $\beta < 0$. We now show the RHS$\geq -\epsilon_n$ for some diminishing error ϵ_n. First let $q_i^S := \sum_{\alpha \subseteq S} \hat{q}_i(\alpha)\chi_\alpha$ be the restriction of q_i, then $\tilde{\mathbb{E}}_S[q_i] = \tilde{\mathbb{E}}_S[q_i^S] = \tilde{\mathbb{E}}_0[q_i^S]$. The error comes from two parts:

1. If q_i is a low-entropy density with $H(q_i) < n-t$, as discussed previously, $\lambda_i < 2^{-t}$. By Claim 1(ii) and $R \leq n^{d/2}$, we have

$$\sum_{i:H(q_i)<n-t} \lambda_i \tilde{\mathbb{E}}_S[q_i] = \sum_{i:H(q_i)<n-t} \lambda_i \tilde{\mathbb{E}}_0[q_i^S] \geq -2^{-t} n^{d/2} \sum_{i=1}^{d} \binom{m}{i} \tag{17}$$

2. If $H(q_i) \geq n-t$, by Lemma 2 we know the degree-d part of q_i^S can be decomposed as a nonnegative d-junta plus some error e_i, s.t. $|\hat{e}_i(\alpha)| \leq \left(\frac{16mtd}{\sqrt{n}}\right)^{1/2}, \forall\, \alpha \subseteq S, |\alpha| \leq d$. So we have

$$\tilde{\mathbb{E}}_0[q_i^S] \geq -|\tilde{\mathbb{E}}_0[e_i]| \geq -\sum_{\alpha:|\alpha|\leq d} |\hat{e}_i(\alpha)| \cdot |\tilde{\mathbb{E}}_0[\chi_\alpha]| \geq -\left(\frac{16mtd}{\sqrt{n}}\right)^{1/2} \sum_{i=0}^{d} \binom{m}{i}.$$

   Therefore

$$\begin{aligned} \sum_{i:H(q_i)\geq n-t} \lambda_i \tilde{\mathbb{E}}_S[q_i] &= \sum_{i:H(q_i)\geq n-t} \lambda_i \tilde{\mathbb{E}}_0[q_i^S] \\ &\geq -\left(\frac{16mtd}{\sqrt{n}}\right)^{1/2} \sum_{i=0}^{d} \binom{m}{i} \sum_{i:H(q_i)\geq n-t} \lambda_i \\ &\geq -\left(\frac{16mtd}{\sqrt{n}}\right)^{1/2} \sum_{i=0}^{d} \binom{m}{i} \end{aligned} \tag{18}$$

   where the last inequality is by $\sum_{i=1}^{R} \lambda_i \leq 1$ (this can be seen by taking expectation on both sides of (14)).

Combining (17) and (18), we conclude that the RHS of (16) has

$$\sum_{i=1}^{R} \lambda_i \tilde{\mathbb{E}}_S[q_i] + \lambda_0 \geq \sum_{i=1}^{R} \lambda_i \tilde{\mathbb{E}}_S[q_i] \geq -O\left(\left(\left(\frac{16mtd}{\sqrt{n}}\right)^{1/2} + n^{d/2}2^{-t}\right) \cdot m^d\right)$$

Let $-\epsilon_n$ denote the RHS of the above inequality. If we take $t = d \log n$, then

$$\epsilon_n = O\left(\frac{m^d \sqrt{md \log n}}{n^{1/4}}\right) \tag{19}$$

For fixed m, d, apparently $\epsilon_n \to 0$ when $n \to \infty$, thus proving Theorem 3(i). While for superconstant $d = d(m)$, taking $n = m^{10d}$ still make $\epsilon_n = o(1)$, and the size lower bound is $n^{d/2} \geq m^{5d(m)^2}$.

4.4 Discussion and Improvement

For MaxCSP with exponentially large Sherali-Adams lower bound, Theorem 3 only gives quasi-polynomial extension complexity lower bound. To see why Theorem 3 fails to give a stronger lower bound, first recall the final error bound (19) obtained in the proof:

$$\epsilon_n = O\left(\frac{m^d \sqrt{md \log n}}{n^{1/4}}\right).$$

To make $\epsilon_n \to 0$, $n^{1/4}$ has to be at least larger than m^d, and this is why we only get a quasi-polynomial lower bound. If we can improve the denominator to $n^{\omega(1)}$ or $n^{\Omega(d)}$, then we can significanly improve the lower bound to e.g., sub-exponential in the later case. The $n^{-1/4}$ factor comes from the approximation error given by Chang's Lemma (Lemma 1). Unfortunately, this is essentially tight and one can find counter examples showing that it's impossible to get $n^{-\omega(1)}$ or $n^{-\Omega(d)}$ error rate. For more details regarding this bottleneck, we refer the readers to section 3.1 of [CLRS13].

However, not all hope is lost: observe that in Chang's Lemma we're trying to approximate a high-entropy density by *a single junta*, but for our final purpose, it suffices to have an approximation in the form of *non-negative linear combination of non-negative juntas*. This is how [KMR17] finally improved the lower bound to sub-exponential.

5 Lower Bounds via Reductions

In the area of computational complexity, the most widely-used approach to proving hardness is by reductions: starting from some known hard problem A, use the instance of problem B to build gadgets that can efficiently represent any A-instance. Then this implies if one can solve B efficiently, so can one solve A. So a natural question is: can we find a general reduction strategy for proving extension complexity lower bound? Braun *et al.*[BPZ15, BPR18] proposed an abstract framework to address this problem. We will briefly introduce their framework, and present the result of [BFPS15] that proves a $(2-\epsilon)$-LP hardness[5] for approximating VERTEX-COVER.

5.1 The framework

We first give a abstract definition that captures the type of optimization problems we're dealing with:

Definition 5 (Optimization problem)**.** *An optimization problem $\mathcal{P}$ is defined as a tuple $\mathcal{P} = (\mathfrak{S}, \mathfrak{I})$, where $\mathfrak{S}$ is the set of* feasible solutions*, and $\mathfrak{I}$ is the set of* instances*. Every $\mathcal{I} \in \mathfrak{I}$ can be viewed as an objective function $\mathcal{I} : \mathfrak{S} \mapsto \mathbb{R}$.*

To study the extension complexity of an optimization problem, we need first define an appropriate linearization of it. The linearization is very similar to how we define it for MaxCSP in Section 2, except that the definition now covers a broader range of problems:

[5] We say a problem is "LP-hard to c-approximate" if there's no *polynomial-size* LP formulation that c-approximates the problem.

Definition 6 (LP formulation for optimization problem)**.** *Given any optimization problem* $\mathcal{P} = (\mathfrak{S}, \mathfrak{I})$ *and functions* $C, S : \mathfrak{I} \mapsto \mathbb{R}$*, a* (C, S)*-approximate LP formulation of* $\mathcal{P}$ *consists of a linear program* $Ax \leq b$ *with* $x \in \mathbb{R}^D$ *for some* $D > 0$ *such that:*

Feasible solutions *For every* $s \in \mathfrak{S}$*, there's a* $x^s \in \mathbb{R}^D$ *satisfying* $Ax^s \leq b$*, i.e.,* $\operatorname{conv}(\{x^s : s \in \mathfrak{S}\})$ *is contained in the feasible region.*

Objective functions *For every* $\mathcal{I} \in \mathfrak{I}$*, there's an* affine *function* $w_{\mathcal{I}} : \mathbb{R}^D \mapsto \mathbb{R}$ *satisfying* $w_{\mathcal{I}}(x^s) = \mathcal{I}(s), \forall\, s \in \mathfrak{S}$.

(C, S)**-approximation guarantee** *Let* $\mathfrak{I}^S := \{\mathcal{I} \in \mathfrak{I} : \max_{s \in \mathfrak{S}} \mathcal{I}(s) \leq S(\mathcal{I})\}$*, we require that*

$$\max\{w_{\mathcal{I}}(x) | Ax \leq b\} \leq C(\mathcal{I}) \quad \textit{for all } \mathcal{I} \in \mathfrak{I}^S$$

The above definition is for when $\mathcal{P}$ *is a* maximization *problem. For* minimization *problems we can analogously define* $\mathfrak{I}^S = \{\mathcal{I} \in \mathfrak{I} : \max_{s \in \mathfrak{S}} \mathcal{I}(s) \geq S(\mathcal{I})\}$ *and require*

$$\min\{w_{\mathcal{I}}(x) | Ax \leq b\} \geq C(\mathcal{I}) \quad \textit{for all } \mathcal{I} \in \mathfrak{I}^S.$$

The *size* of the LP is the number of inequalities in $Ax \leq b$. And we let $\mathsf{fc}_{\mathsf{LP}}(\mathcal{P}, C, S)$ denote the minimal size of all the (C, S)-approximate LP formulation for $\mathcal{P}$; we also use $\mathsf{fc}_{\mathsf{LP}}(\mathcal{P}, \alpha)$ to denote case when $C/S = \alpha$ (or $S/C = \alpha$, for minimization problems).

We can also give an analogy of Yannakakis' Factorization Theorem in our setting. First we define the slack matrix of a (C, S)-approximate optimization problem

Definition 7 (Slack matrix for approximate optimization problem)**.** *Given optimization problem* $\mathcal{P} = (\mathfrak{S}, \mathfrak{I})$ *with* (C, S)*-approximation guarantee, we define the* (C, S)-approximation slack matrix *of* $\mathcal{P}$ *as the nonnegative* $\mathfrak{I}^S \times \mathfrak{S}$ *matrix* $M_{\mathcal{P},C,S}$*:*

$$M_{\mathcal{P},C,S}(\mathcal{I}, s) := C(\mathcal{I}) - \mathcal{I}(s)$$

Then we define a type of matrix factorization that's almost identical as the nonnegative factorization, only that they may differ by one affine shift:

Definition 8 (LP factorization)**.** *For any* $M \in \mathbb{R}_{\geq 0}^{m \times n}$*, it has a size-*$r$ LP factorization *if* $M = TU + \mu \mathbf{1}^\top$ *with* $T \in \mathbb{R}_{\geq 0}^{m \times r}, U \in \mathbb{R}_{\geq 0}^{r \times n}, \mu \in \mathbb{R}_{\geq 0}^{m}$*, and* $\mathbf{1} \in \mathbb{R}^n$ *is the all-*1 *vector. The* LP rank *of* M *is defined as the minimum* r *s.t. there exists a size-*r *LP factorization of* M*, denoted as* $\mathsf{rank}_{\mathsf{LP}}(M)$*.*

And the factorization theorem establishes the equivalence between LP rank and LP formula size:

Theorem 4 (Factorization Theorem for LP formula size)**.** *For any approximation problem* $\mathcal{P} = (\mathcal{S}, \mathcal{F}, \mathcal{F}^*)$ *with* (C, S)*-approximation guarantee, we have*

$$\mathsf{fc}_{\mathsf{LP}}(\mathcal{P}, C, S) = \mathsf{rank}_{\mathsf{LP}}(M_{\mathcal{P},C,S})$$

Remark: One may wonder why we not just use the extension complexity xc and nonnegative rank rank_+ as previously. The extension complexity *for approximation problems*, as defined in [BFPS12], may not equal the slack matrix's nonnegative rank *exactly*: they may differ by 1 (see Theorem 1 and 2 in [BFPS12]). Furthermore, they're defined *with respect to specific linear encoding of the problem.* On the other hand, the $\mathsf{fc}_{\mathsf{LP}}$ and $\mathsf{rank}_{\mathsf{LP}}$ not only have a slightly cleaner factorization theorem, they also subsume the framework of [BFPS12] in the sense that: $\mathsf{fc}_{\mathsf{LP}}$ can be viewed as the minimum extension complexity with respect to all possible linear encodings. Therefore, $\mathsf{fc}_{\mathsf{LP}}$ is a stronger notion than xc for approximation problems, and a lower bound for $\mathsf{fc}_{\mathsf{LP}}$ is a lower bound for xc. For more detailed discussion about the relation to approximate extended formulations, we refer the readers to Appendix B of [BPZ15].

We will now define the reduction between optimization problems that also translate the approximation guarantee.

Definition 9 (Reduction)**.** *Let $\mathcal{P}_1 = (\mathfrak{S}_1, \mathfrak{J}_1), \mathcal{P}_2 = (\mathfrak{S}_2, \mathfrak{J}_2)$ be optimization problems with approximation guarantees $(C_1, S_1), (C_2, S_2)$, respectively. Furthermore, let $\tau_1 = +1$ if $\mathcal{P}_1$ is a maximization problem and $\tau_1 = -1$ if it's a minimization problem, and define τ_2 for $\mathcal{P}_2$ similarly. Then a* reduction *from $\mathcal{P}_1$ to $\mathcal{P}_2$ respecting approxation guarantee consists of*

1. *two mappings: $\alpha : \mathfrak{J}_1 \mapsto \mathfrak{J}_2$ and $\beta : \mathfrak{S}_1 \mapsto \mathfrak{S}_2$ translating instances and feasible solutions, respectively;*
2. *two nonnegative $\mathfrak{J}_1 \times \mathfrak{S}_1$ matrices M_1, M_2*

such that for any $\mathcal{I}_1 \in \mathfrak{J}_1, s_1 \in \mathfrak{S}_1$ and $\mathcal{I}_2 = \alpha(\mathcal{I}_1), s_2 = \beta(s_1)$, there's

$$\text{Completeness:} \quad \tau_1[C_1(\mathcal{I}_1) - \mathcal{I}_1(s_1)] = \tau_2[C_2(\mathcal{I}_2) - \mathcal{I}_2(s_2)] \cdot M_1(\mathcal{I}_1, s_1) + M_2(\mathcal{I}_1, s_1) \tag{20}$$
$$\text{Soundness:} \quad \tau_2 \cdot \mathsf{opt}(\mathcal{I}_2) \le S_2(\mathcal{I}_2) \text{ if } \tau_1 \cdot \mathsf{opt}(\mathcal{I}_1) \le S_1(\mathcal{I}_1) \tag{21}$$

Since by definition $M_{\mathcal{P}_1,C_1,S_1}(\mathcal{I}_1, s_1) = \tau_1[C_1(\mathcal{I}_1) - \mathcal{I}_1(s_1)]$ and $M_{\mathcal{P}_2,C_2,S_2}(\mathcal{I}_2, s_2) = \tau_2[C_2(\mathcal{I}_2) - \mathcal{I}_2(s_2)]$, the completeness requirement (20) is essentially describing a mapping from $M_{\mathcal{P}_1,C_1,S_1}$ to $M_{\mathcal{P}_2,C_2,S_2}$. Intuitively, if M_1, M_2 is relatively simple, then $M_{\mathcal{P}_1,C_1,S_1}$ and $M_{\mathcal{P}_2,C_2,S_2}$ should have similar $\mathsf{rank}_{\mathsf{LP}}$. This intuition is formalized in the following theorem:

Theorem 5 ([BPR18])**.** *Given optimization problems $\mathcal{P}_1, \mathcal{P}_2$ with corresponding completeness and soundness guarantees $(C_1, S_1), (C_2, S_2)$, if we have a reduction from $\mathcal{P}_1$ to $\mathcal{P}_2$ that respects their approximation guarantees as in Definition 9, then*

$$\mathsf{fc}_{\mathsf{LP}}(\mathcal{P}_1, C_1, S_1) \le \mathsf{rank}_{\mathsf{LP}}(M_2) + \mathsf{rank}_{\mathsf{LP}}(M_1) + \mathsf{rank}_{+}(M_1) \cdot \mathsf{fc}_{\mathsf{LP}}(\mathcal{P}_2, C_2, S_2) \tag{22}$$

5.2 Extension complexity for approximating vertex-cover

5.2.1 LP-hardness for $(\frac{3}{2} - \epsilon)$-approximating vertex-cover

It's known that the basic relaxation for VERTEX-COVER has a integrality gap $2 - \epsilon$ even after $\Omega(n^{\gamma})$-rounds of Sherali-Adams lift[CMM09]. But we can't use Theorem 3 here because it only applies to MaxCSP problems. We will prove the LP-hardness result using the reduction theorem 5 stated in the last section. First, as a direct consequence of Theorem 3, we have the following corollary about MAX-CUT:

Corollary 1. *For every $\epsilon > 0$ and for infinitely many $n \in \mathbb{N}$, the* MAX-CUT *problem on n-vertex graph has $\mathsf{fc}_{\mathsf{LP}}(\text{MAX-CUT}, 1 - \epsilon, 1/2 + \epsilon) \ge n^{\Omega(\log n / \log\log n)}$.*

By reducing from MAX-CUT, we'll show the following LP-hardness result for VERTEX-COVER:

•TODO: Define the VC problem, emphasize the "non-uniformity" of this problem. The LP for MaxCut is a "uniform" model in the sense that the LP only depends on input size n, while the LP for VC can (and should) depend on the actual input.

Theorem 6. *For every $\epsilon > 0$ and for infinitely many $n \in \mathbb{N}$, there exists a n-vertex graph $G = (V, E)$ such that $\mathsf{fc}_{\mathsf{LP}}(\text{VERTEX-COVER}(G), 1.5 - \epsilon) \ge n^{\Omega(\log n / \log\log n)}$.*

Proof. The reduction consists of two mappings: $\mathcal{H}$, between the problem instances, and $\mathcal{U}$, between feasible solutions. We first define the mapping $\mathcal{H}$ from MAX-CUT instance to VERTEX-COVER instance. Let MAX-CUT$_n$ denote the set of all MAX-CUT instances on n-vertex graph, then each $G \in$ MAX-CUT$_n$ can be viewed as a weight-assignment on all edges of K_n, the n-vertex complete graph. In particular, we can focus on $0-1$ edge weights, and G is just a subset of edges of K_n. To construct the mapping from MAX-CUT$_n$ to VERTEX-COVER, we first build a *conflict graph* K^* from K_n, then our reduction will map each $G \in$ MAX-CUT$_n$ to a VERTEX-COVER instance on some subgraph $G^* \subseteq K^*$. The conflict graph K^* is defined as follows (see Figure 1 for an illustration):

vertices: Use $[n]$ to index the vertices of K_n. For each $(i, j) \in E(K_n)$, it corresponds to a XOR constraint P defined on variable X_i, X_j, and we create two vertices to represent the two satisfying partial assignment $\sigma : (X_i, X_j) \mapsto \{0, 1\}^2$ for this constraint: specifically, we use $v_{P,(0,1)}, v_{P,(1,0)}$ to denote the two vertices.

edges: We connect two vertices v_{P_1,σ_1} and v_{P_2,σ_2} if the partial assignments σ_1, σ_2 are incompatible: i.e., there exists some X_i that's in both P_1 and P_2's variable set, and $\sigma_1(i) \neq \sigma_2(i)$.

K^* is known as a *universal FGLSS graph*[FGL+91] which encodes all possible choices of predicates simultaneously. Note $|V(K^*)| = 2\binom{n}{2}$, so the mapping is between MAX-CUT$_n$ and VERTEX-COVER$_{n(n-1)}$. With K^*, the mapping between problem instances is naturally defined: given any $G \in$ MAX-CUT$_n$, the VERTEX-COVER instance $\mathcal{H}(G)$ is the subgraph $G^* \subseteq K^*$ induced by the vertex set mapped from $E(G)$.

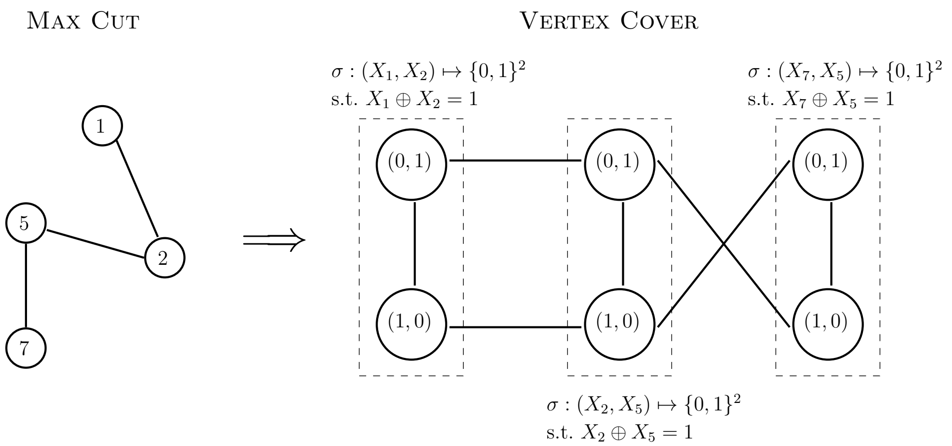

Figure 1: Conflict graph of the 2-XOR clause, used for reduction from MAX-CUT to VERTEX-COVER.

To finish the definition of the reduction, we now define the mapping $\mathcal{U}$ between feasible solutions. Given any MAX-CUT$_n$ solution $x \in \{0,1\}^n$, define vertex set $\mathcal{U}(x) := \{v_{P,\sigma} \in V(G^*) : \sigma \nsubseteq x\}$ as the set of all vertices corresponding to partial assignments *incompatible* with x. We claim this is a valid vertex cover: for any edge $(v_{P_1,\sigma_1}, v_{P_2,\sigma_2}) \in E(G^*)$, since by construction σ_1 and σ_2 are incompatible, at least one of them will be incompatible with x, thus being included in $\mathcal{U}(x)$.

Now we use Theorem 5 to show the above reduction gives a $(1.5-\epsilon)$-LP hardness for VERTEX-COVER. Let $\mathcal{P}_1 = (\mathfrak{S}_1, \mathfrak{J}_1)$ be the MAX-CUT problem, and $\mathcal{P}_2 = (\mathfrak{S}_2, \mathfrak{J}_2)$ be the VERTEX-COVER problem; also, let $G = \mathcal{I}_1 \in \mathfrak{J}_1$ be the MAX-CUT instance we reduce from, and $G^* = \mathcal{I}_2 = \mathcal{H}(\mathcal{I}_1)$ be the VERTEX-COVER instance reduced to.

- *Soundness:* First we prove the soundness requirement (21). Observe that if $\mathcal{I}_1$ has m constraints, i.e., $|E(G)| = m$, then $|V(G^*)| = 2m$. Suppose $\mathsf{opt}(\mathcal{I}_1) \leq 1/2 + \epsilon$, we show that there must be $\mathsf{opt}(\mathcal{I}_2) \geq (1.5-\epsilon)m$: for any valid vertex cover $U \subset V(G^*)$, its complement $\bar{U}$ is an independent set of size $|V(G^*)| - |U| = 2m - |U|$. Furthermore for any two $v_{P_1,\sigma_1}, v_{P_2,\sigma_2} \in \bar{U}$, the partial assignment σ_1 and σ_2 are compatible with each other. Take the union of all such partial assignment defined by $\bar{U}$, and extend it arbitrarily to other variables in $[n]$ to get an full assignment $x \in 0, 1^n$, then x satisfies at least $|\bar{U}|$ many constraints in G. Therefore we have $2m - |U| \leq (1/2+\epsilon)m$, i.e., $|U| \geq (1.5-\epsilon)m$.

- *Completeness:* For completeness (20), observe that (1) if a constraint $P(X_i, X_j) = \mathbf{1}[X_i \oplus X_j = 1]$ is satisfied by x, then exactly one of $v_{P,(1,0)}, v_{P,(0,1)}$ is included in $\mathcal{U}(x)$; (2) if $P(X_i, X_j)$ is not satisfied by x, then both $v_{P,(1,0)}, v_{P,(0,1)}$ are included in $\mathcal{U}(x)$. Therefore we have the following equation:

$$\mathcal{I}_1(x) = \frac{1}{m}(2m - \mathcal{I}_2(\mathcal{U}(x)))$$

  And the completeness requirement (20) is satisfied as follows:

$$1 - \epsilon - \mathcal{I}_1(x) = -\frac{1}{m}((1+\epsilon)m - \mathcal{I}_2(\mathcal{U}(x)))$$

Thus by Theorem 5, we have

$$\mathsf{fc}_{\mathsf{LP}}(\mathcal{P}_1, 1-\epsilon, 1/2+\epsilon) \le \mathsf{fc}_{\mathsf{LP}}(\mathcal{P}_2, 1.5-\epsilon) + 1$$

Apply Corollary 1 we get the desired lower bound □

5.2.2 Improvement: LP hardness of $(2-\epsilon)$-approximating vertex-cover

From the proof of Theorem 6, we can see that if the MAX-CUT problem provides a larger gap, say $(1-\epsilon, \epsilon)$-LP hardness, then the same reduction can prove a $(2-\epsilon)$-LP hardness result for VERTEX-COVER. Unfortunately, the $(1-\epsilon, 1/2+\epsilon)$-LP hardness of MAX-CUT is known to be tight: the natural LP has integrality gap 2. A natural thought is to seek harder MaxCSP problems that can utilize the conflict graph for reduction. Indeed, the key property of the CSP we need is the *one-free bit* property defined as follows:

Definition 10 (One-free bit CSP(1F-CSP)[BK09])**.** *A* 1F-CSP *instance of arity k is a special binary* MAX-CSP *instance of the same arity, such that each constraint has exactly two satisfying assignments out of the 2^k possible ones.*

Apparently MAX-CUT is just a special kind of 1F-CSP. If we reduce from 1F-CSP to VERTEX-COVER using the conflict graph, then an *almost unsatisfiable* instance will translate to a VERTEX-COVER instance that has $\mathsf{opt} \sim 2m$, where m is the number of constraints in the 1F-CSP instance. This gives the desired $2-\epsilon$ soundness parameter for VERTEX-COVER.

A natural candidate of hard MaxCSPs is the UNIQUE-GAMES problem, which is known to be $(1-\epsilon, \epsilon)$-LP hard for any $\epsilon > 0$. Unfortunately, UNIQUE-GAMES is not a 1F-CSP, so we cannot use it as base problems directly. But we can actually reduce UNIQUE-GAMES to 1F-CSP under the framework of Theorem 5 while preserving the completeness-soundness gap. Then following a similar approach as Theorem 6, we can get the $(2-\epsilon)$-LP hardness of VERTEX-COVER.

For completeness, we restate the definition of UNIQUE-GAMES (see Example 2) below, but in the form of a graph coloring problem:

Definition 11 (UNIQUE-GAMES)**.** *An* UNIQUE-GAMES(n, q) *instance consists of a graph $G = (V, E)$, a set of labels $\Omega = [q]$, and a set of permutations $\{\pi_{uv} : [q] \mapsto [q] \,|\, (u, v) \in E\}$ associated with each edge in E. The goal is to find a labeling $\ell : V \mapsto [q]$ to maximize the number of satisfied edges, where an edge (u, v) is satisfied iff $\pi(\ell(u)) = \ell(v)$.*

And we have the follwoing LP-hardness result for UNIQUE-GAMES:

Theorem 7 ([LRS15],Corollary 7.7)**.** *For every $q \ge 2, \delta > 0$, and $d \ge 1$, there exists constant $c > 0$, such that for all $n \ge 1$,*

$$\mathsf{fc}_{\mathsf{LP}}\left(\text{UNIQUE-GAMES}(n, q), 1-\delta, \frac{1}{q}+\delta\right) \ge cn^d$$

In other words, no LP family of polynomial size can $(1-\delta, 1/q+\delta)$-approximate UNIQUE-GAMES(n, q)*.*

In particular, we can restric the UNIQUE-GAMES instance to regular graphs, and the above theorem still hold. Let UNIQUE-GAMES$_\Delta(n, q)$ denote the instance on Δ-regular graphs, and we reduce UNIQUE-GAMES to 1F-CSP as follows:

Lemma 3 (UNIQUE-GAMES $\Longrightarrow$ 1F-CSP)**.** *For any $\eta, \epsilon, \delta, \zeta > 0$, and positive integers t, q, Δ that only depend on η, ϵ, δ, we have*

$$\mathsf{fc}_{\mathsf{LP}}(\text{UNIQUE-GAMES}_\Delta(n, q), 1-\zeta, \delta) - n\Delta^t q^{t+1} \le \mathsf{fc}_{\mathsf{LP}}(\text{1F-CSP}, (1-\epsilon)(1-\zeta t), \eta) \tag{23}$$

The reduction from UNIQUE-GAMES to 1F-CSP is fairly standard: using 1F-CSP constraint to build a *long code tester* (a.k.a. *dictatorship tester*) testing the validity of the given labeling for the UNIQUE-GAMES instance. Since a tester is equivalent to a CSP instance, we get the 1F-CSP instance. For detailed proof, we refer the readers to Theorem 9.2 in [BPR18]. Combined with Theorem 7, this implies $(1-\epsilon,\epsilon)$-approximating 1F-CSP requires $n^{\omega(1)}$-size LP formulations (since the $n\Delta^{t+1}q^{t+1} = \mathsf{poly}(n)$ as Δ, q, t are independent with n). The last step is reducing from 1F-CSP to VERTEX-COVER, which is just the same as in the proof of Theorem 6.

Theorem 8 (1F-CSP $\Longrightarrow$ VERTEX-COVER)**.** *For every $\epsilon > 0$ and for infinitely many $n \in \mathbb{N}$, there exists a n-vertex graph $G = (V, E)$ such that $\mathsf{fc}_{\mathsf{LP}}(\text{VERTEX-COVER}(G), 2-\epsilon) \geq n^{\omega(1)}$.*

Proof. The proof is almost the same as that of Theorem 6: In the conflict graph, for every constraint P from the 1F-CSP we create two vertices $v_{P,\sigma_1}, v_{P,\sigma_2}$ corresponding to the two satisfying partial assignment of P; Two vertices are connected if the two partial assignment conflict. The rest of the proof follows as in the proof of Theorem 6, only that the 1F-CSP is $(1-\epsilon,\epsilon)$-LP hard. □

In Theorem 8 we only get a $n^{\omega(1)}$ lower bound for $(2-\epsilon)$-approximate VERTEX-COVER, which is weaker than the $n^{\Omega(\log n/\log\log n)}$ bound for $(1.5+\epsilon)$-approximate VERTEX-COVER. This is mainly because the base problem (UNIQUE-GAMES) only has a $n^{\omega(1)}$ extension complexity lower bound. But we can actually circumvent this barrier: using a reduction almost identical in Lemma 3, one can construct a 1F-CSP instance that requires the same *Sherali-Adams lower bound* as the UNIQUE-GAMES, while still preserving the large completeness-soundness gap $(1-\epsilon,\epsilon)$. As it's already known UNIQUE-GAMES survives $n^{\Theta(1)}$-round of Sherali-Adams lifting[CMM09], this gives a same Sherali-Adams lower bound for 1F-CSP, which in turn implies[6] a $n^{\Omega(\log n/\log\log n)}$ extension complexity lower bound due to Theorem 3, and thus a $n^{\Omega(\log n/\log\log n)}$ lower bound for $(2-\epsilon)$-approximate VERTEX-COVER.

5.3 Discussion

The nature of reduction means that any improvement on the $\mathsf{fc}_{\mathsf{LP}}$ of the base problem will imply the same improvement on $\mathsf{fc}_{\mathsf{LP}}(\text{VERTEX-COVER})$. Indeed, this already happens for $(1.5+\epsilon)$-approximate VERTEX-COVER: as mentioned in section 4.4, the lower bound for $(1-\epsilon, 1/2+\epsilon)$-approximate MAX-CUT can be improved to $2^{n^{c(\epsilon)}}$ where $c(\epsilon)$ is some constant only depending on ϵ; This automatically implies subexponential extension complexity lower bound for $(1.5+\epsilon)$-approximate VERTEX-COVER.

Apart from VERTEX-COVER, one can deduce extension complexity lower bound for many other problems by reduction. A most direct one is INDEPENDENT-SET as it's the complement of VERTEX-COVER: we can get a LP hardness result for INDEPENDENT-SET by the same reduction as in Theorem 6 and 8. Indeed, one can show that $\mathsf{fc}_{\mathsf{LP}}(\text{INDEPENDENT-SET}, 1/\epsilon) \geq n^{\Omega(\log n/\log\log n)}$ using the same reduction gadgets. For more results we refer the readers to [BPR18].

6 Related results for SDP extension complexity

Semidefinite programming (SDP) can be thought as a generalization of LP, where the variables are required to be in a positive semidefinite (PSD) cone. There's a counterpart of the nonnegative rank called *PSD rank*, and a factorization theorem that relates the PSD rank of slack matrices with SDP formulation size[GPT13, FMP+12]. The SDP extension complexity results surveyed in this section, although utilize very different techniques, are similar to the LP results from a highlevel viewpoint. The SDP extension complexity lower bound is also proved by lifting from hierarchy lower bound: the hierarchy used is the Sum-of-Square (SoS) hierarchy (a.k.a. Lasserre hierarchy) that can be thought as an SDP counterpart of Sherali-Adams hierarchy.

Lee *et al.*[LRS15] proved a lifting theorem similar to Theorem 3, claiming that if degree-$d(n)$ SoS relaxation cannot achieve $(c+\epsilon, s)$-approximation for MAX-Π_n, then no SDP relaxation of size at most $O(n^{d(n)^2/8})$ can achieve a (c,s)-apprxoimation for MAX-Π_N for $N > n^{4d(n)}$. This combined with known $\Omega_\epsilon(n)$-SoS lower bound of MAX-3SAT [Gri01, Sch08], gives a quasi-polynomial SDP size-lower bound for $(7/8+\epsilon)$-approximate MAX-3SAT. The proof strategy of [LRS15] can also be divided into three steps as in [CLRS13]: First, a low psd rank factorization gives a "high-entropy" certificate Q (like the set of q_i's in (14)); Then one can show this certificate Q is well-approximated by a relatively

[6]Note we cannot apply Theorem 3 directly to UNIQUE-GAMES, since it's not a binary CSP.

simple function R: in the SDP setting, R is a not-very-high degree sum-of-squares; Finally one uses the idea of random restriction to reduce the degree of R to some small number d. Then R would serve as a degree-d SoS solution for the MaxCSP, which would violates known SoS lower bound. The main source of complicatedness here is Q being a matrix-valued function, and [LRS15] uses a "quantum-learning" argument to approximate Q, which is basically a sub-gradient descent algorithm. The same approach is also used for obtaining sub-exponential SDP size lower bound for polytopes like $\textsc{cut}_n, \textsc{tsp}_n, \textsc{stab}_n$. Their techniques can also be adapted to derive LP size lower bound, and in particular, they give a $n^{\omega(1)}$-LP size lower bound for $(1-\delta, 1/q+\delta)$-approximate $\textsc{unique-games}(n,q)$. Another interesting result obtained via lifting SoS lower bound is an SDP size lower bound for the MATCHING polytope: Braun *et al.*[BBH+17] proved that approximating $\textsc{matching}_n$ within $1-\epsilon/(n-1)$ will require $2^{\Omega(n)}$-sized *symmetric* SDP.

The reduction technique presented in section 5 also naturally extends to SDPs. Using the quasi-polynomial lower bound of MAX-3SAT mentioned in previous paragraphs, Braun *et al.*[BPR18] is able to show a $n^{\Omega(\log n/\log\log n)}$ SDP size lower bound for $(4/5-\epsilon, 3/4+\epsilon)$-approximate MAX-CUT. For more results obtained via reduction, we refer the readers to [BPZ15, BPR18].

7 Open problems

TODO: This part needs revision.

At the end of this survey we'd like to highlight some major open problems remained in this area. All the currently known results are for relatively simple problems, like the binary MaxCSP, or very structured polytopes like CUT, STAB, CORR. While for many other important combinatorial optimization problems like Set-Cover, Scheduling, few is known. The situation is true even for hierarchy lower bounds. There're a few problems that admit efficient lift-and-project algorithms, like Knapsack[KMN11], Directed Steiner Tree[Rot11], and some variants of Scheduling[LR16, GKL19]; But many seems to resist hierarchy liftings, although few lower bounds are known either.

The lack of stronger LP size lower bound for non-binary CSPs, especially the UNIQUE-GAMES problem, is also unsatisfying. Right now we only have a $n^{\omega(1)}$-LP size lower bound for UNIQUE-GAMES from [LRS15][7]. Given the central importance of UNIQUE-GAMES in hardness of approximation, any progress would probably imply some extension complexity results for lots of problems that are "UG-hard".

Another natural question is whether one can prove stronger SDP size lower bound. For MaxCSPs, the best known lower bound is still quasi-polynomial[LRS15]. Apart from the NP-hard problems, the MATCHING polytope is also very intriguing: although its LP extension complexity is settled by Rothvoß[Rot14], much less is known about the SDP case. For now we know any *symmetric* SDP extended formulation of MATCHING has exponential size[BBH+17], but nothing is known for asymmetric SDPs.

References

[ABL02] Sanjeev Arora, Béla Bollobás, and László Lovász. Proving integrality gaps without knowing the linear program. In *43rd Symposium on Foundations of Computer Science (FOCS 2002), 16-19 November 2002, Vancouver, BC, Canada, Proceedings*, pages 313–322, 2002.

[BBH+17] Gábor Braun, Jonah Brown-Cohen, Arefin Huq, Sebastian Pokutta, Prasad Raghavendra, Aurko Roy, Benjamin Weitz, and Daniel Zink. The matching problem has no small symmetric SDP. *Math. Program.*, 165(2):643–662, 2017.

[BCK15] Boaz Barak, Siu On Chan, and Pravesh K. Kothari. Sum of squares lower bounds from pairwise independence. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 97–106, 2015.

[7] The result of [LRS15] seems to directly imply a 2^{n^ϵ} LP-size lowerbound for UNIQUE-GAMES, but this is never stated explicitly (as far as I know) in any literatures.

[BFPS12] Gábor Braun, Samuel Fiorini, Sebastian Pokutta, and David Steurer. Approximation limits of linear programs (beyond hierarchies). In *53rd Annual IEEE Symposium on Foundations of Computer Science, FOCS 2012, New Brunswick, NJ, USA, October 20-23, 2012*, pages 480–489, 2012.

[BFPS15] Abbas Bazzi, Samuel Fiorini, Sebastian Pokutta, and Ola Svensson. No small linear program approximates vertex cover within a factor 2 - ϵ. In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1123–1142, 2015.

[BHK$^+$16] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 428–437, 2016.

[BK09] Nikhil Bansal and Subhash Khot. Optimal long code test with one free bit. In *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25-27, 2009, Atlanta, Georgia, USA*, pages 453–462, 2009.

[BM13] Mark Braverman and Ankur Moitra. An information complexity approach to extended formulations. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 161–170, 2013.

[BPR18] Gábor Braun, Sebastian Pokutta, and Aurko Roy. Strong reductions for extended formulations. *Math. Program.*, 172(1-2):591–620, 2018.

[BPZ15] Gábor Braun, Sebastian Pokutta, and Daniel Zink. Inapproximability of combinatorial problems via small LPs and SDPs. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 107–116, 2015.

[BS14] Boaz Barak and David Steurer. Sum-of-squares proofs and the quest toward optimal algorithms. *CoRR*, abs/1404.5236, 2014.

[Cha02] Mei-Chu Chang. A polynomial bound in Freiman's theorem. *Duke Mathematical Journal*, 113(3):399–419, 2002.

[Chl07] Eden Chlamtac. Approximation algorithms using hierarchies of semidefinite programming relaxations. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2007), October 20-23, 2007, Providence, RI, USA, Proceedings*, pages 691–701, 2007.

[CLRS13] Siu On Chan, James R. Lee, Prasad Raghavendra, and David Steurer. Approximate constraint satisfaction requires large LP relaxations. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 350–359, 2013.

[CMM09] Moses Charikar, Konstantin Makarychev, and Yury Makarychev. Integrality gaps for Sherali-Adams relaxations. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 283–292, 2009.

[dlVK07] Wenceslas Fernandez de la Vega and Claire Kenyon-Mathieu. Linear programming relaxations of MaxCut. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, pages 53–61, 2007.

[FGL$^+$91] Uriel Feige, Shafi Goldwasser, László Lovász, Shmuel Safra, and Mario Szegedy. Approximating clique is almost np-complete (preliminary version). In *32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991*, pages 2–12, 1991.

[FMP+12] Samuel Fiorini, Serge Massar, Sebastian Pokutta, Hans Raj Tiwary, and Ronald de Wolf. Linear vs. semidefinite extended formulations: exponential separation and strong lower bounds. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 95–106, 2012.

[GKL19] Shashwat Garg, Janardhan Kulkarni, and Shi Li. Lift and project algorithms for precedence constrained scheduling to minimize completion time. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, pages 1570–1584, 2019.

[GPT13] João Gouveia, Pablo A. Parrilo, and Rekha R. Thomas. Lifts of convex sets and cone factorizations. *Math. Oper. Res.*, 38(2):248–264, 2013.

[Gri01] Dima Grigoriev. Linear lower bound on degrees of Positivstellensatz calculus proofs for the parity. *Theor. Comput. Sci.*, 259(1-2):613–622, 2001.

[GT17] Mrinalkanti Ghosh and Madhur Tulsiani. From weak to strong LP gaps for all csps. In *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, pages 11:1–11:27, 2017.

[IMR14] Russell Impagliazzo, Cristopher Moore, and Alexander Russell. An entropic proof of Chang's inequality. *SIAM J. Discrete Math.*, 28(1):173–176, 2014.

[KMN11] Anna R. Karlin, Claire Mathieu, and C. Thach Nguyen. Integrality gaps of linear and semi-definite programming relaxations for knapsack. In *Integer Programming and Combinatoral Optimization - 15th International Conference, IPCO 2011, New York, NY, USA, June 15-17, 2011. Proceedings*, pages 301–314, 2011.

[KMR17] Pravesh K. Kothari, Raghu Meka, and Prasad Raghavendra. Approximating rectangles by juntas and weakly-exponential lower bounds for LP relaxations of csps. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 590–603, 2017.

[KTW14] Subhash Khot, Madhur Tulsiani, and Pratik Worah. A characterization of strong approximation resistance. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 634–643, 2014.

[Lau03] Monique Laurent. A comparison of the Sherali-Adams, Lovász-Schrijver, and Lasserre relaxations for 0-1 programming. *Math. Oper. Res.*, 28(3):470–496, 2003.

[LR16] Elaine Levey and Thomas Rothvoss. A (1+epsilon)-approximation for makespan scheduling with precedence constraints using LP hierarchies. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 168–177, 2016.

[LRS15] James R. Lee, Prasad Raghavendra, and David Steurer. Lower bounds on the size of semidefinite programming relaxations. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 567–576, 2015.

[LRST14] James R. Lee, Prasad Raghavendra, David Steurer, and Ning Tan. On the power of symmetric LP and SDP relaxations. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 13–21, 2014.

[LS91] László Lovász and Alexander Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, 1(2):166–190, 1991.

[O'D14] Ryan O'Donnell. *Analysis of boolean functions.* Cambridge University Press, 2014.

[Rot11] Thomas Rothvoß. Directed steiner tree and the Lasserre hierarchy. *CoRR*, abs/1111.5473, 2011.

[Rot14] Thomas Rothvoß. The matching polytope has exponential extension complexity. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 263–272, 2014.

[SA90] Hanif D. Sherali and Warren P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. Discrete Math.*, 3(3):411–430, 1990.

[Sch08] Grant Schoenebeck. Linear level Lasserre lower bounds for certain k-CSPs. In *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*, pages 593–602, 2008.

[Tul09] Madhur Tulsiani. CSP gaps and reductions in the lasserre hierarchy. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 303–312, 2009.

[Yan91] Mihalis Yannakakis. Expressing combinatorial optimization problems by linear programs. *J. Comput. Syst. Sci.*, 43(3):441–466, 1991.