Quasi-Random PCP and Hardness of 2-Catalog Segmentation

Rishi Saket[*]

Carnegie Mellon University
rsaket@cs.cmu.edu

Abstract

We study the problem of 2-Catalog Segmentation which is one of the several variants of segmentation problems, introduced by Kleinberg *et al.* [11], that naturally arise in data mining applications. Formally, given a bipartite graph $G = (U, V, E)$ and parameter r, the goal is to output two subsets $V_1, V_2 \subseteq V$, each of size r, to maximize,

$$\sum_{u \in U} \max\{|E(u, V_1)|, |E(u, V_2)|\},$$

where $E(u, V_i)$ is the set of edges between u and the vertices in V_i for $i = 1, 2$. There is a simple 2-approximation for this problem, and stronger approximation factors are known for the special case when $r = |V|/2$ [5, 16]. On the other hand, it is known to be NP-hard [11, 5, 12], and Feige [7] showed a constant factor hardness based on an assumption of average case hardness of random 3SAT.

In this paper we show that there is no PTAS for 2-Catalog Segmentation assuming that NP does not have subexponential time probabilistic algorithms, i.e. NP $\not\subseteq \cap_{\varepsilon>0}$ BPTIME$(2^{n^{\varepsilon}})$. In order to prove our result we strengthen the analysis of the Quasi-Random PCP of Khot [10], which we transform into an instance of 2-Catalog Segmentation. Our improved analysis of the Quasi-Random PCP proves stronger properties of the PCP which might be useful in other applications.

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2010.447

1 Introduction

In Computer Science many important problems are known to be NP-hard, i.e. a polynomial algorithm for any of these problems will imply P = NP. Many of these problems are in essence optimization questions, for example the MAX-3SAT problem of satisfying the maximum number of clauses of a 3SAT instance. It follows from the NP-hardness of SAT that it is NP-hard to compute the optimum of MAX-3SAT as well. This, however, motivates the study of efficient approximation algorithms for optimization problems. An algorithm (for a minimization problem) is said to have an approximation factor of $C > 1$ if it computes a solution which is at most factor C away from the optimum; and the definition is analogous for a maximization problem so that the approximation factor C is always greater than 1. An optimization problem is said to admit a Polynomial Time Approximation Scheme (PTAS) if it has a $1 + \varepsilon$ approximation algorithm for every constant $\varepsilon > 0$, which runs in time polynomial in the size of the problem. Several important problems have been found to admit a PTAS, such as the classic KNAPSACK [15] and EUCLIDEANTSP [2] problems.

For a long time it was an important open question as to whether MAX-3SAT has a PTAS, until the well known Probabilistically Checkable Proof (PCP) Theorem [4, 3] proved

[*] Supported in part by Venkatesan Guruswami's Packard Fellowship

IARCS Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010).
Editors: Kamal Lodaya, Meena Mahajan; pp. 447–458
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

a constant factor hardness for Max-3SAT . Equivalently, the PCP Theorem shows the existence of an efficient probabilistic verifier for NP : i.e. for any language L in NP, there is a verifier which can efficiently decide whether $x \in L$, given a proof whose size is polynomial in $|x|$. The verifier reads only a constant number of bits from the proof, uses only logarithmic (in $|x|$) randomness, and for every correct statement there is a proof such that the verifier always accepts it, while every proof for an incorrect statement is rejected with high probability. The PCP Theorem, along with tools such as Fourier Analysis and Parallel Repetition [13], has led to several important inapproximability results (many of them optimal) such as [9], [6], [8].

However, till some time ago, there remained some important problems such as Graph Min-Bisection, Dense k-Subgraph and Bipartite Clique for which no hardness of approximation was known. Feige [7] showed that these problems do not have a PTAS under the assumption that random 3SAT instances are hard on average. Subsequently, in a breakthrough work Khot [10] constructed the so called Quasi-Random PCP and used it to rule out PTAS for the above mentioned problems under the standard assumption that NP does not have subexponential time (randomized) algorithms, i.e. NP $\not\subseteq \cap_{\varepsilon>0}$ BPTIME($2^{n^{\varepsilon}}$). More recently, the Quasi-Random PCP was used by [1] to rule out PTAS for the well known Sparsest Cut problem.

In this paper we study the notorious question (as far as inapproximability results go) of 2-CatalogSegmentation. In this problem, one is given a set of items and a set of customers, where every customer is interested in a personal subset of items. Given a parameter r, the goal is to construct two *catalogs* of r items each, and send exactly one of the two catalogs to each customer. The payoff from any customer is the number of items on the catalog sent to him that he is interested in, and the goal is to maximize the total payoff for all the customers. 2-CatalogSegmentation is one of the several variants of segmentation problems first studied by Kleinberg, Papadimitriou and Raghavan [11]. Such problems arise naturally in data mining for devising marketing strategies or production plans. It has also been used for modeling certain coding theory problems [12].

The 2-CatalogSegmentation problem is known to be NP-hard [11], [5] and [12], while there is a simple 2-approximation for it. Dodis, Guruswami and Khanna [5] studied the special case when $r = n/2$, where n is the total number of items and gave a 1.76-approximation algorithm for this special case, which was subsequently improved to 1.56 in [16]. Feige [7] showed a constant factor hardness for 2-CatalogSegmentation under an assumption about average case hardness of random 3SAT. However, under standard complexity assumptions, no hardness of approximation was known till now for the 2-CatalogSegmentation problem.

In this paper we prove a hardness of approximation result for 2-CatalogSegmentation, which is stated informally below.

▶ **Theorem.** *There is no PTAS for the* 2-CatalogSegmentation *problem unless NP has subexponential time randomized algorithms, i.e. unless NP $\subseteq \cap_{\varepsilon>0}$ BPTIME($2^{n^{\varepsilon}}$).*

In order to prove our result we strengthen the analysis of the Quasi-Random PCP of Khot [10], which is then reduced to an instance of 2-CatalogSegmentation. Our improved analysis of the Quasi-Random PCP proves stronger properties of the PCP which might be useful in other applications.

In the next section we start with some preliminary definitions and statement of our results. In Section 3 we shall prove the inapproximability of 2-CatalogSegmentation based on the properties of the Quasi-Random PCP obtained by our strengthened analysis. The subsequent sections are devoted to proving the desired properties of the Quasi-Random PCP.

2 Preliminaries

We start with the formal definition of the 2-CatalogSegmentation problem.

▶ **Definition 1.** 2-CatalogSegmentation: Given a bipartite graph $G = (U, V, E)$, and a parameter r, the goal is to output two sets V_1 and V_2 such that $V_1, V_2 \subseteq V$ and $|V_1|, |V_2| = r$ to maximize the following quantity.

$$\sum_{u \in U} \max \{|E(u, V_1)|, |E(u, V_2)|\} .$$

where $E(u, V_i)$ is the set of edges incident on u with the other end in V_i, for $i = 0, 1$.

The vertices in U represent the customers and the ones in V represent the items, and an edge (u, v) signifies that the customer u is interested in item v. One is required to construct two catalogs V_1 and V_2 of r items each and send each customer one of the two catalogs. The objective is to maximize the sum, over all customers, of the number of items each customer receives in his catalog that he is interested in. Feige [7] proved a conditional inapproximability result for 2-CatalogSegmentation under a hypothesis about the average case hardness of 3SAT, both of which are stated below.

▶ **Hypothesis 2.** *(Random* 3SAT *Hypothesis) For every fixed $\varepsilon > 0$ and for Δ a sufficiently large constant independent of n, there is no polynomial time algorithm that on a random* 3CNF *formula with n variables and $m = \Delta n$ clauses, outputs YES if the formula is satisfiable, and NO at least half the time if the formula is unsatisfiable.*

▶ **Theorem 3.** *(Feige [7]) Assuming Hypothesis 2, there is no polynomial time algorithm to approximate* 2-CatalogSegmentation *within a factor of $1 + \varepsilon$ for some $\varepsilon > 0$.*

Feige [7] also proved inapproximability results for other problems such as Graph Min-Bisection, Dense k-subgraph and Bipartite Clique based on Hypothesis 2. As mentioned in the previous section, Khot [10] subsequently constructed the Quasi-Random PCP, which was used to rule out PTAS for Graph Min Bisection, Dense k-subgraph and Bipartite Clique, under the standard assumption that NP has no subexponential time algorithms. Before proceeding, we recall the formal statement of the Quasi-Random PCP.

▶ **Theorem 4.** *(Khot's Quasi-Random PCP [10]) For every $\varepsilon > 0$, there exists an integer $d = O(1/\varepsilon \log(1/\varepsilon))$ such that the following holds : there is a PCP verifier for a* SAT *instance of size n satisfying:*

1. *The proof for the verifier is of size $2^{O(n^{\varepsilon})}$.*
2. *The verifier reads $4d$ bits from the proof. Let Q be the $4d$ bits queried by the verifier in a random test.*
3. *Every query bit is uniformly distributed over the proof, though different query bits within Q are correlated.*
4. *(YES Case) Suppose that the* SAT *instance is satisfiable. Then there exists a correct proof Π^*, such that if Π^*_0 be the set of 0-bits in the proof Π^*, then,*

$$\Pr_Q [Q \subseteq \Pi^*_0] \geq D \frac{1}{2^{4d-1}},$$

where $D = \left(1 - O\left(\frac{1}{d^2}\right)\right)$ and the probability is taken over a random test of the verifier.

5. *(NO Case) Suppose that the* SAT *instance is unsatisfiable, and let* Π' *be any set of half the bits in the proof. Then,*

$$\left| \Pr_{Q}\left[Q \subseteq \Pi'\right] - \frac{1}{2^{4d}} \right| \leq \frac{1}{2^{40d}}.$$

As one can see, in the NO case the PCP exhibits a quasi-randomness property, in the sense that the probability is close to what is expected if each query bit were chosen uniformly at random in the proof. However, the above statement does not seem strong enough to prove an inapproximability for the 2-CatalogSegmentation problem. In our results we prove a strengthened statement for the Quasi-Random PCP and apply it to prove the desired result for 2-CatalogSegmentation. In the next few paragraphs we formally state the results of this paper.

2.1 Our Results

For the purpose of convenience, we let the query Q of the PCP verifier be a tuple of $4d$ bits in the proof, i.e. $Q = (q_1, q_2, \ldots, q_{4d})$. The verifier queries the bits q_i $(1 \leq i \leq 4d)$ as part of the query Q. For a given proof Π, let $val(\Pi, q)$ denote the 0 or 1 value of the proof Π at the bit q. We prove the following strengthened theorem regarding the Quasi-Random PCP.

▶ **Theorem 5.** *For every $\varepsilon > 0$, there exists an integer $d = O(1/\varepsilon \log(1/\varepsilon))$ such that the following holds : there is a PCP verifier for a* SAT *instance of size n satisfying the following properties.*

1. *The proof for the verifier is of size $2^{O(n^{\varepsilon})}$.*
2. *The verifier reads $4d$ bits from the proof. Let $Q = (q_1, \ldots q_{4d})$ be the tuple of $4d$ bits queried by the verifier.*
3. *Every query bit q_i is uniformly distributed over the proof, though different query bits within Q are correlated.*
4. *(YES Case) Suppose that the* SAT *instance is satisfiable. Then there exists a correct proof Π^*, which is 1 on exactly half the fraction of the bits and satisfies the following property. Fix any $4d$ boolean values $r_1, r_2, \ldots, r_{4d} \in \{0,1\}$, such that $\sum_{i=1}^{4d} r_i = 0 \pmod 2$. Then,*

$$\Pr_{Q}\left[\bigwedge_{i=1}^{4d} (val(\Pi^*, q_i) = r_i)\right] \geq D\frac{1}{2^{4d-1}},$$

   where $D = \left(1 - O\left(\frac{1}{d^2}\right)\right)$ is independent of $r_1, \ldots, r_{4d}$, and the probability is taken over a random test of the verifier.
5. *(NO Case) Suppose that the* SAT *instance is unsatisfiable, and let Π be any proof that is 1 on exactly half fraction of the bits. Fix any $4d$ boolean values $s_1, s_2, \ldots, s_{4d} \in \{0,1\}$. Then,*

$$\left| \Pr_{Q}\left[\bigwedge_{i=1}^{4d} (val(\Pi, q_i) = s_i)\right] - \frac{1}{2^{4d}} \right| \leq \frac{1}{2^{40d}}.$$

Note that in the above statement, we prove a stronger property of the Quasi-Random PCP in the YES case, which is that the distribution of the $4d$ bit string read in the query Q is close to that of a uniform distribution over all $4d$ bit strings with an even number of 1s. This implies the property in the YES case proved in Theorem 4. In some sense we prove a partial quasi-randomness property even in the YES case, except that it is with respect to the uniform distribution over all $4d$ bit strings with even number of 1s. The property that we prove in the NO case is also a similar generalization of the corresponding property in

Theorem 4. Essentially, the distribution of the $4d$ bit string read by the query Q is close to what is obtained by picking $4d$ random bits from the proof given in the NO case.

Using the above strengthened statement of the Quasi-Random PCP we prove the following inapproximability of the 2-CatalogSegmentation problem.

▶ **Theorem 6.** *Let $\varepsilon > 0$ be an arbitrarily small constant. Assume that* SAT *has no algorithm in BPTIME(2^{n^ε}). Then there is no polynomial time algorithm for* 2-CatalogSegmentation *that achieves an approximation of $1 + \Omega(\frac{1}{d})$, where $d = O(1/\varepsilon \log(1/\varepsilon))$. In particular, the* 2-CatalogSegmentation *problem does not admit a PTAS unless NP $\subseteq \cap_{\varepsilon>0}$ BPTIME(2^{n^ε}).*

A sketch of the proof of Theorem 5 is given in Sections 4, 5 and 6. It requires describing, at least to some extent, the construction of the Quasi-Random PCP of [10]. We start with the description of HomAlgCSP problem in Section 4. This is the starting point for constructing an Outer Verifier in Section 5 and the final Inner Verifier in Section 5.2. The construction of these verifiers is same as in [10] except for some convenient notational changes and appropriate selection of parameters. Section 6 gives a brief sketch of analysis of the PCP, a key new ingredient in which is Lemma 11 that is used along with the techniques of [10] to prove the strengthened property in the YES case.

In the next section we prove Theorem 6 assuming Theorem 5. We reduce from the Quasi-Random PCP to an instance of 2-CatalogSegmentation. The reduction is similar to the one used to prove Theorem 3 in [7].

3 Reduction to 2-CatalogSegmentation and proof of Theorem 6

In this section we describe the reduction to 2-CatalogSegmentation from the Quasi-Random PCP given by Theorem 5. In the following construction, U and V will be the sets of customers and items respectively. There is an edge between a customer and an item if the customer is interested in that item. The reduction is as follows.

1. Let the set of customers U be the set of all the bits in the proof of the PCP.
2. For every tuple of $4d$ bits Q queried by the PCP verifier we have an item. We replicate every item (tuple of $4d$ bits) proportional to the probability it is queried by the verifier. Let V be the set of all items.
3. A bit in U is connected to all the tuples Q in V that contain it.
4. Set $r = D|V|\frac{1}{2^{4d-1}}$ to be the catalog size. Here $D = \left(1 - O\left(\frac{1}{d^2}\right)\right)$ as in the YES case of Theorem 5.

The analysis is as follows.

YES Case. Let Π^* be the correct proof to the PCP verifier given by Theorem 5. Construct two catalogs $V_1, V_2 \subseteq V$ where,

$$\begin{aligned} V_1 &\subseteq \{Q \mid \text{all bits of } Q \text{ are set to } 0 \text{ in } \Pi^*\}, & (1)\\ V_2 &\subseteq \{Q \mid \text{all bits of } Q \text{ are set to } 1 \text{ in } \Pi^*\} & (2)\end{aligned}$$

Setting $r_i = 0$ for $i = 1, \ldots, 4d$ in the property of the YES case in Theorem 5, we can ensure that,

$$|V_1| = D|V|\frac{1}{2^{4d-1}}.$$

Similarly, by setting $r_i = 1$ for $i = 1, \ldots, 4d$ we can ensure that,

$$|V_2| = D|V|\frac{1}{2^{4d-1}}.$$

We note that both V_1 and V_2 are disjoint subsets of V.

Now send catalog V_1 to the customers corresponding to the bits set to 0 in Π^* and V_2 to the complement, i.e. customers corresponding to the bits set to 1 in Π^*. Clearly, each item in V_1 and V_2 reaches $4d$ customers interested in it. Therefore the value of the solution is,

$$\begin{aligned} & 4d|V|2D\left(\frac{1}{2^{4d-1}}\right) \\ = \; & 8d|V|\left(1-O\left(\frac{1}{d^2}\right)\right)\left(\frac{1}{2^{4d-1}}\right) \end{aligned} \tag{3}$$

NO Case. For convenience we allow the two catalogs to be of size $|V|\frac{1}{2^{4d-1}}$, as this can only increase the payoff. In the NO Case, one of the catalogs, call it V' reaches at most half of the customers. Clearly, the payoff obtained by this catalog only increases if we enlarge the set of customers to which V' is sent, without changing the other catalog and the set of customers to which it is sent. Therefore, we may assume that V' is sent to exactly half of the customers. Let the proof Π be constructed by setting the bits corresponding to these customers to be 1 and the rest to 0. From the NO case of Theorem 5, by setting $s_i = 1$ for $i = 1, \dots, 4d$, we obtain that at most $\frac{1}{2^{4d}} + \frac{1}{2^{40d}}$ fraction of all the tuples Q queried have all the $4d$ bits set to 1. Therefore, there are at most $\left(\frac{1}{2^{4d}} + \frac{1}{2^{40d}}\right)|V|$ items in V' that reach $4d$ customers interested in them. Therefore, at least $\left(\frac{1}{2^{4d}} - \frac{1}{2^{40d}}\right)|V|$ items in V' reach at most $4d-1$ customers interested in them. The other catalog has a payoff of at most $4d|V|\frac{1}{2^{4d-1}}$. Hence, the value of any solution in the NO case is at most,

$$\begin{aligned} & |V|\left(4d\left(\frac{1}{2^{4d-1}}\right) + 4d\left(\frac{1}{2^{4d}} + \frac{1}{2^{40d}}\right) + (4d-1)\left(\frac{1}{2^{4d}} - \frac{1}{2^{40d}}\right)\right) \\ \leq \; & |V|\left(4d\left(\frac{1}{2^{4d-1}}\right) + 4d\left(\frac{3}{2}\cdot\frac{1}{2^{4d}}\right) + (4d-1)\left(\frac{1}{2}\cdot\frac{1}{2^{4d}}\right)\right) \\ = \; & 8d|V|\left(1-\frac{1}{32d}\right)\left(\frac{1}{2^{4d-1}}\right). \end{aligned} \tag{4}$$

Hardness Factor. From Equations (3) and (4) we obtain that the ratio of the value of the solution in the YES case to the best solution in the NO case is at least,

$$\frac{\left(1-O\left(\frac{1}{d^2}\right)\right)}{\left(1-\frac{1}{32d}\right)} = 1+\Omega\left(\frac{1}{d}\right) \tag{5}$$

Therefore, if 2-CatalogSegmentation is approximable within factor $1+\Omega\left(\frac{1}{d}\right)$, then NP $\subseteq$ BPTIME$(2^{n^{\varepsilon}})$, where $d = O(1/\varepsilon \log 1/\varepsilon)$. This rules out PTAS for 2-CatalogSegmentation unless NP $\subseteq \cap_{\varepsilon>0}$BPTIME$(2^{n^{\varepsilon}})$. This proves Theorem 6.

4 Homogeneous Algebraic CSP

We define the HomAlgCSP problem which is the starting point of the reduction in this paper. This definition is a (slightly modified) restatement of Definition 3.1 of [10].

▶ **Definition 7.** Given parameters k, d, m and a field $\mathbb{F}$, let HomAlgCSP instance $\mathcal{A}(k, d, m, \mathbb{F}, \mathcal{C})$ be the following problem (think of k as a fixed integer like 21, and d as a large constant integer) :

1. $\mathcal{C}$ is a system of constraints on functions $f : \mathbb{F}^m \mapsto \mathbb{F}$ where every constraint is on values of f on k different points and is given by a conjunction of homogeneous linear constraints on those k values. A typical constraint $C \in \mathcal{C}$ looks like

$$\sum_{i=1}^{k} \gamma_{ij} f(\overline{p}_i) = 0 \quad \text{for } j = 1, 2, \ldots \qquad \text{where } \overline{p}_i \in \mathbb{F}^m \text{ and } \gamma_{ij} \in \mathbb{F}.$$

We denote a constraint C by the set of points $\{\overline{p}_i\}_{i=1}^{k}$, while the γ_{ij}'s will be implicit.

2. $\mathcal{C}$ has $|\mathbb{F}|^{O(m)}$ constraints.

The goal is to find a m-variate polynomial f, not identically zero, so as to maximize the fraction of constraints satisfied. Let $OPT(\mathcal{A})$ denote the maximum fraction of constraints satisfied by any such polynomial of degree at most d.

5 Construction of the PCP

This section describes construction given in [10] of a PCP Outer Verifier and Inner Verifiers for a HomAlgCSP instance $\mathcal{A}(k = 21, d^*, m, \mathbb{F}, \mathcal{C})$ based on a variant of the Low-Degree test of Rubinfeld and Sudan [14].

The Outer Verifier is given the polynomial f as a table of values at each point in $\mathbb{F}^m$. It picks a constraint in $\mathcal{C}$ uniformly at random and checks whether it is satisfied. Before the description we need the definition of a curve.

▶ **Definition 8.** A *curve* L in $\mathbb{F}^m$ is a function $L : \mathbb{F} \mapsto \mathbb{F}^m$, where $L(t) = (a_1(t), \ldots, a_m(t))$. It is a degree d curve if each of the coordinate functions a_i $(1 \leq i \leq m)$ is degree d (univariate) polynomial.

A *line* is a curve of degree 1.

Let $C(\{\overline{p}_i\}_{i=1}^{k}) \in \mathcal{C}$, denote the constraint that the Verifier chooses at random to check. Let $t_1, t_2, \ldots, t_{k+3}$ be distinct field elements in $\mathbb{F}$ which we fix for the rest of the paper. For $\overline{a}, \overline{b}, \overline{c} \in \mathbb{F}^m$, let $L = L_{\overline{a},\overline{b},\overline{c}}$ be the unique degree $k+2$ curve that passes through the points $\{\overline{p}_i\}_{i=1}^{k}, \overline{a}, \overline{b}, \overline{c}$. More precisely,

$$L(t_i) = \overline{p}_i, \quad 1 \leq i \leq k, \quad L(t_{k+1}) = \overline{a}, \quad L(t_{k+2}) = \overline{b}, \quad L(t_{k+3}) = \overline{c}.$$

In brief the strategy of the Outer Verifier is as follows. Suppose f is a degree d^* multivariate polynomial over the vector space $\mathbb{F}^m$. Clearly, its restriction to the curve $L(t) = L_{\overline{a},\overline{b},\overline{c}}(t)$ is a degree $d - 1 := (k+2)d^*$ univariate polynomial in t. This polynomial, denoted by $f|_L$, can be interpolated from any d values of f on the curve. This is precisely what the verifier does: it picks $d+1$ points on the curve L, interpolates $f|_L$ from the first d points and verifies that the value of f and $f|_L$ is the same on the last point. In addition, it checks that the values of $f|_L$ at the points $\{\overline{p}_i\}_{i=1}^{k}$ satisfies the constraint C. Note that the values $t_1, \ldots, t_{k+3}$ on which L depends, are fixed. This is combined with the line-point Low Degree Test. Given a line ℓ, the restriction of f, denoted by $f|_\ell$ is a degree d^* univariate polynomial, but we allow it degree up to $d - 2$, and interpolate it using the values of f at $d - 1$ random points on ℓ.

We next give the detailed description of the *Modified Outer Verifier*, which for technical reasons, reads more values from the proof and makes additional tests, while building upon the Outer Verifier. Also, it abstracts out the tasks of interpolation into multiplication by an invertible matrix, and checking the homogeneous constraints of the Outer Verifier into checking orthogonality with a certain subspace.

5.1 Modified Outer Verifier

We first observe that $\mathbb{F}$ is an extension of $\mathbb{F}[2]$. Therefore, we can represent the elements of $\mathbb{F}$ as bit strings of a length $l = \log|\mathbb{F}|$. Moreover, the representation can be chosen such that addition over $\mathbb{F}$ and multiplication by a constant in $\mathbb{F}$ are homogeneous linear operations on these bit strings. The Modified Outer Verifier is given a table of values $f(\overline{v})$ (in the form of l bit strings) for every point $\overline{v} \in \mathbb{F}^m$ and it executes the following steps:

Steps of the Modified Outer Verifier

1. Pick a constraint $C = \{\overline{p}_i\}_{i=1}^k \in \mathcal{C}$ at random.
2. Pick a random line ℓ (in $\mathbb{F}^m$) and pick random points $\overline{v}_1, \ldots, \overline{v}_{d-1}, \overline{v}_d$ on the line.
3. Pick $t \in \mathbb{F} \setminus \{t_1, \ldots, t_{k+3}\}$ at random, points $\overline{a}, \overline{b}$ at random from $\mathbb{F}^m$ and let L be the unique degree $k+2$ curve $L = L_{\overline{a},\overline{b},\overline{c}}$ such that, $L(t_i) = \overline{p}_i$, $1 \le i \le k$, $L(t_{k+1}) = \overline{a}$, $L(t_{k+2}) = \overline{b}$, and $L(t) = \overline{v}_d$ so that $\overline{c}$ is automatically defined to $L(t_{k+3})$.
4. Pick random points $\overline{v}_{d+1}, \ldots, \overline{v}_{2d}$ on the curve L.
5. Pick additional random points $\overline{u}_1 \ldots \overline{u}_d$ on the line ℓ and $\overline{u}_{d+1}, \ldots, \overline{u}_{2d}$ from the curve L. (We assume that all the points chosen on the line ℓ and curve L are distinct, which happens w.h.p)
6. Let $T_{2ld \times 2ld}$ be an appropriate invertible matrix over $\mathbb{F}[2]$ and H be an appropriate subspace of $\mathbb{F}[2]^{2ld}$. Both depend only on the choice of the points $\{\overline{v}_i\}_{i=1}^{2d}$ and $\{\overline{u}_j\}_{j=1}^{2d}$. Remark 1 explains how T and H are chosen.
7. Read the values of the function f from the table at the points $\overline{v}_1, \ldots, \overline{v}_{2d}$ and $\overline{u}_1, \ldots, \overline{u}_{2d}$. Since all the elements of the field are represented by bit strings, let
$$x = f(\overline{v}_1) \circ f(\overline{v}_2) \circ \cdots \circ f(\overline{v}_{2d}) \tag{6}$$
$$y = f(\overline{u}_1) \circ f(\overline{u}_2) \circ \cdots \circ f(\overline{u}_{2d}) \tag{7}$$
where $\circ$ represents concatenation of strings.
8. Accept iff,
$$x \neq 0,\ x = Ty \qquad \text{and} \qquad h \cdot x = 0 \quad \forall\, h \in H \quad \text{(i.e. } x \perp H\text{)}. \tag{8}$$

▶ **Remark 1.** *The subspace H is chosen such that the constraint $h \cdot x = 0\ \forall\ h \in H$ ensures that the values at the field elements $\{t_i\}_{i=1}^k$ of the degree $d-1$ univariate polynomial interpolated from $f(\overline{v}_{d+1}) \ldots f(\overline{v}_{2d})$, (which are supposed to be the values of f at $\{\overline{p}_i\}_{i=1}^k$) satisfy the homogeneous linear constraints of C. In addition, H is chosen such that the polynomial interpolated from the values $f(\overline{v}_1) \ldots f(\overline{v}_{d-1})$ agrees with the degree $d-1$ polynomial interpolated from $f(\overline{v}_{d+1}) \ldots f(\overline{v}_{2d})$ at the point $\overline{v}_d$, where both evaluate to $f(\overline{v}_d)$.*

The invertible matrix T is chosen such that the constraint $x = Ty$ ensures the following conditions are satisfied:

1. *The degree $d-1$ polynomial interpolated from the values $f(\overline{v}_1) \ldots f(\overline{v}_d)$ is the same as the polynomial interpolated from the values $f(\overline{u}_1) \ldots f(\overline{u}_d)$. (This polynomial will actually be of degree $d-2$ due to the constraint enforced by the subspace H).*
2. *The degree $d-1$ polynomial interpolated from $f(\overline{v}_{d+1}) \ldots f(\overline{v}_{2d})$ is the same as the polynomial interpolated from the values $f(\overline{u}_{d+1}) \ldots f(\overline{u}_{2d})$.*

The condition $x \neq 0$ essentially ensures that f is not a zero polynomial.

5.2 Inner Verifier

We now construct the Inner Verifier which is essentially identical to the one constructed in [10], except for some notational complications that we need to introduce. It expects, for every point $\overline{v} \in \mathbb{F}^m$, the Hadamard Code of the string $f(\overline{v}) \in \{0,1\}^l$ (refer to Appendix A.3 of [10] for an overview). The following are the steps executed by the verifier.

Steps of the Inner Verfier

1. Pick a constraint $C \in \mathcal{C}$ and the points $\overline{v}_1, \ldots, \overline{v}_{2d}$ and $\overline{u}_1, \ldots, \overline{u}_{2d}$ as in steps $1-5$ of the Modified Outer Verifier.
2. Let $T_{2ld \times 2ld}$ and H be the matrix and subspace respectively chosen as in step 7 of the Modified Outer Verifier.
3. Pick a random string $z \in \mathbb{F}^{2ld}$ and a random $h \in H$. Write, $z = z_1 \circ z_2 \circ \cdots \circ z_{2d}$, $h = h_1 \circ h_2 \circ \cdots \circ h_{2d}$, and $zT = w_1 \circ w_2 \circ \cdots \circ w_{2d}$.
4. Let $A_1, \ldots, A_{2d}$ and $B_1, \ldots, B_{2d}$ be the (supposed) Hadamard Codes of $f(\overline{v}_1), \ldots, f(\overline{v}_{2d})$ and
$f(\overline{u}_1), \ldots, f(\overline{u}_{2d})$ respectively, given by the proof Π.
5. Let Q be defined as the tuple of $4d$ 'positions' queried by the Inner Verifier. It is formally set as: $Q = (q_1, \ldots, q_{2d}, q_{2d+1} \ldots, q_{4d})$, where q_i is the bit read at the position $z_i \oplus h_i$ of the Hadamard Code A_i for $1 \leq i \leq 2d$. Similarly, q_{j+2d} is the bit read at position w_j of the Hadamard Code B_j for $1 \leq j \leq 2d$.
6. Let $val(q_i, \Pi) \in \{0,1\}^{4d}$ be the value of the i^{th} bit in the tuple Q given by the proof Π. From the construction of Π and Q we have: $val(q_i, \Pi) = A_i(z_i \oplus h_i), \quad 1 \leq i \leq 2d$, and $val(q_{j+2d}, \Pi) = B_j(w_j), \quad 1 \leq j \leq 2d$.
6. Accept iff $\oplus_{i=1}^{4d} val(q_i, \Pi) = 0$.

For our eventual application, we are in fact not interested in the acceptance probabilities of the Inner Verifier in the YES and NO cases. Instead, we wish to study the distribution of the number of 1s and 0s in the tuple of $4d$ bits Q.

6 Sketch of Analysis

We begin this sketch by first stating the two key lemmas regarding the behaviour of the Inner Verifier depending on the instance $\mathcal{A}$ of HomAlgCSP.

The first lemma states that if the instance $\mathcal{A}$ of HomAlgCSP has a very good optimum, then there is a proof to the Inner Verifier such that the distribution of the $4d$ bits of Q, read by the verifier from the proof, is close to the uniform distribution over $4d$ bit strings with even number of 1s.

▶ **Lemma 9.** *Let $r_1, \ldots, r_{4d} \in \{0,1\}$ be any fixed boolean values such that $\sum_{i=1}^{4d} r_i = 0 \pmod 2$. Suppose the $\mathcal{A}$ is an instance of* HomAlgCSP *with optimum $OPT(\mathcal{A})$, given by the polynomial f. Let Π^* be the proof (for the Inner Verifier) constructed taking the Hadamard Code for every value of f. Let $Q = (q_1, \ldots, q_{4d})$ be the (random) tuple of length $4d$ bits queried, as described in the steps of the Inner Verifier. Similarly, let $val(q_i, \Pi^*)$ be the value in the proof Π^* at the i^{th} bit in Q. Then,*

$$\Pr_Q\left[\bigwedge_{i=1}^{4d} (val(q_i, \Pi^*) = r_i)\right] \geq OPT(\mathcal{A}) \cdot \frac{1}{2^{4d-1}}, \tag{9}$$

where the probability is taken over the random test of the Inner Verifier.

Note that in the above lemma, the proof Π^* is balanced i.e. it is 1 on half fraction of the bits. This is because Hadamard Codes of non-zero values are balanced and since f is not identically zero and has degree at most d^*, it is non-zero on all except a negligibly small fraction of points in $\mathbb{F}^m$.

The next lemma states that if there is a proof to the Inner Verifier such that if the distribution of the $4d$ bits of Q (read by the verifier from the proof) deviates significantly from the uniform distribution over $4d$ bit strings from the proof, there is a constant degree polynomial that satisfies a significant fraction of constraints of $\mathcal{A}$.

▶ **Lemma 10.** *Let $\mathcal{A}$ be an instance of* HomAlgCSP *and suppose Π is a proof to the Inner Verifier for $\mathcal{A}$, with the property that Π is 1 on exactly half fraction of the total bits. As before, let $Q = (q_1, \ldots, q_{4d})$ be the tuple of $4d$ bits queried by the Inner Verifier, and $val(q_i, \Pi)$ be the value in the proof Π at the i^{th} bit of the tuple Q. Let $s_1, \ldots, s_{4d} \in \{0,1\}$ be any $4d$ boolean values. If,*

$$\left| \Pr_{Q}\left[\bigwedge_{i=1}^{4d} (val(q_i, \Pi) = r_i) \right] - \frac{1}{2^{4d}} \right| \geq \delta \geq 0 \tag{10}$$

then there is a polynomial of degree at most $50d^*$, not identically zero, which satisfies $\delta^{C'}$ fraction of the constraints of $\mathcal{A}$, where C' is an absolute constant.

The proof of Lemma 10 follows from Theorem 7.6 of [10] which bounds the acceptance probability of the Modified Outer Verifier. The proof of Theorem 5 follows from the above two lemmas combined with Theorem 3.4 of [10] which proves the inapproximability of HomAlgCSP. The various parameters need to be chosen appropriately and the analysis follows the same scheme as given in Section 10 of [10].

Our main contribution is to strengthen the analysis in the YES case of Theorem 5, which is done by proving a more general Lemma 9 as compared to Lemma 10.2 of [10]. The main ingredient is the following lemma which we state and prove below. Before we do so, let us recall some notation.

We shall consider a test of the Modified Outer Verifier. Let T be the $2ld \times 2ld$ invertible matrix over $\mathbb{F}[2]$ and H be the appropriate subspace of $\mathbb{F}[2]^{2ld}$ constructed by the Modified Outer Verifier depending on the (randomized) choice of the constraint C, line ℓ, curve L and the points $\overline{v}_1, \ldots, \overline{v}_{2d}, \overline{u}_1, \ldots, \overline{u}_{2d}$, as explained in Remark 1. Note that the values queried by the Modified Outer Verifier are represented by l-bit strings over $\mathbb{F}[2]$.

▶ **Lemma 11.** *Let C be a constraint in $\mathcal{C}$ that is satisfied by the polynomial f (of degree at most d^*) given by Lemma 9. Let $\alpha, \beta \in \mathbb{F}[2]^{2ld}$ such that $\alpha := \alpha_1 \circ \cdots \circ \alpha_{2d}$ and $\beta := \beta_1 \circ \cdots \circ \beta_{2d}$ where $\alpha_i, \beta_j \in \mathbb{F}[2]^l$ for $1 \leq i, j \leq 2d$. Also, let property* `P1` *for α and β be defined as follows.*

`P1`*: Each α_i is either 0 or $f(\overline{v}_i)$ for $1 \leq i \leq 2d$, and each β_j is either 0 or $f(\overline{u}_j)$ for $1 \leq j \leq 2d$.*

Then with probability $1 - O(d^2/|\mathbb{F}|)$ over the choice of the line ℓ, curve L, and the points $\{\overline{v}_i\}_{i=1}^{2d}$ and $\{\overline{u}_j\}_{j=1}^{2d}$, the following holds:
The only two solutions to $\alpha \perp H$, $\beta = T^{-1}\alpha$ satisfying property `P1` *are,*

$$\alpha_i = \beta_j = 0 \quad \forall 1 \leq i, j \leq 2d, \tag{11}$$

and,

$$\alpha_i = f(\overline{v}_i), \quad \beta_j = f(\overline{u}_j) \ \ \forall 1 \leq i, j \leq 2d. \tag{12}$$

Proof. Firstly, we have that with probability at least $1 - O(d^2/|\mathbb{F}|)$, none of the values $\{f(\overline{v}_i)\}_{i=1}^{2d}$ and $\{f(\overline{u}_j)\}_{j=1}^{2d}$ are 0. This is because each of the $4d$ points are uniformly distributed over $\mathbb{F}^m$ and since f is not identically zero and of degree at most $d^* \leq d$, by

Schwartz-Zippel Lemma the probability that any of the $4d$ points is a root of f is at most $4d \cdot O(d/|\mathbb{F}|) = O(d^2/|\mathbb{F}|)$. Since this probability is negligible, we can assume for the rest of the argument that none of the values $\{f(\overline{v}_i)\}_{i=1}^{2d}$ and $\{f(\overline{u}_j)\}_{j=1}^{2d}$ are 0.

Next, from the construction of the matrix T (refer to Remark 1) we have that $\beta = T^{-1}\alpha$ implies the following two properties.

P2: The polynomial interpolated by the values α_i at point $\overline{v}_i$ for $1 \leq i \leq d$ is identical to the one interpolated by the values β_j at point $\overline{u}_j$ for $1 \leq j \leq d$.

P3: The polynomial interpolated by the values α_i at point $\overline{v}_i$ for $d+1 \leq i \leq 2d$ is identical to the one interpolated by the values β_j at point $\overline{u}_j$ for $d+1 \leq j \leq 2d$.

Also, from the construction of the subspace H (refer to Remark 1) we have that $\alpha \perp H$ implies the following property.

P4: The polynomial interpolated from the values α_i at points $\overline{v}_i$ $(1 \leq i \leq d-1)$ agrees with the polynomial interpolated from the values α_j at points $\overline{v}_j$ $(d+1 \leq j \leq 2d)$ at the point $\overline{v}_d$ where both evaluate to α_d.

Clearly the solution $\alpha_i = \beta_j = 0$ for all $1 \leq i, j \leq 2d$ is a valid solution to $\alpha \perp H, \alpha = T^{-1}\beta$ and satisfying property P1. Also, since f is a degree d^* polynomial that satisfies the constraint C, the solution $\alpha_i = f(\overline{v}_i)$ and $\beta_j = f(\overline{u}_j)$ for all $1 \leq i, j \leq 2d$ is a valid solution as well.

Suppose that there is another solution α, β, different from the above two, satisfying the properties P1, P2, P3 and P4. Then, at least one of the following eight cases must happen, all of which we show have a low probability of occurring.

Case 1. $\alpha_i = 0$ for $1 \leq i \leq d$ and $\alpha_j = f(\overline{v}_j)$ for $d+1 \leq j \leq 2d$. This along with property P4 implies that the univariate f' polynomial interpolated from the values $\alpha_j = f(\overline{v}_j)$ at points $\overline{v}_j$ for $d+1 \leq j \leq 2d$ evaluates to $\alpha_d = 0$ at the point $\overline{v}_d$. Clearly, since f' is unique, it has to evaluate to $f|_L(\overline{v}_d) = f(\overline{v}_d)$ at point $\overline{v}_d$. This implies that $f(\overline{v}_d) = 0$ which contradicts our earlier assumption that all the values $f(\overline{v}_i)$ and $f(\overline{u}_j)$ $(1 \leq i, j \leq 2d)$ are non zero.

Case 2. $\alpha_i = f(\overline{v}_i)$ for $1 \leq i \leq d$ and $\alpha_j = 0$ for $d+1 \leq j \leq 2d$. Again, property P4 implies that zero polynomial interpolated from the values $\alpha_j = 0$ at points $\overline{v}_j$ for $d+1 \leq j \leq 2d$, evaluates to $\alpha_d = f(\overline{v}_d)$ at point $\overline{v}_d$. This means that $f(\overline{v}_d) = 0$, which is a contradiction to our assumption.

Case 3. $\beta_i = 0$ for $1 \leq i \leq d$ and $\beta_j = f(\overline{u}_j)$ for $d+1 \leq j \leq 2d$. From properties P2 and P3, this implies that $\alpha_i = 0$ for $1 \leq i \leq d$ and $\alpha_j = f(\overline{v}_j)$ for $d+1 \leq j \leq 2d$. Thus this reduces to Case 1.

Case 4. $\beta_i = f(\overline{u}_i)$ for $1 \leq i \leq d$ and $\beta_j = 0$ for $d+1 \leq j \leq 2d$. Again, properties P2 and P3 imply that $\alpha_i = f(\overline{v}_i)$ for $1 \leq i \leq d$ and $\alpha_j = 0$ for $d+1 \leq j \leq 2d$. Thus this reduces to Case 2.

Case 5. There exist $1 \leq i', j' \leq d$ such that $\alpha_{i'} = f(\overline{v}_{i'})$ and $\beta_{j'} = 0$. Let f_1 be the univariate polynomial interpolated by the values α_i at point $\overline{v}_i$ for $1 \leq i \leq d$. This polynomial is not identically zero since $\alpha_{i'} = f(\overline{v}_{i'}) \neq 0$ (by our initial assumption). Also the degree of the polynomial is at most d. Property P2 implies that f_1 takes the value $\beta_{j'} = 0$ at the point $\overline{u}_{j'}$. However, since the points $\{\overline{u}_j\}_{j=1}^{d}$ are chosen uniformly at random on the line ℓ (refer to Section 5.1), the probability that one of them is a root of f_1 is at most $O(d^2/|\mathbb{F}|)$. Therefore, this case occurs with probability at most $O(d^2/|\mathbb{F}|)$.

Case 6. There exist $d+1 \leq i', j' \leq 2d$ such that $\alpha_{i'} = f(\overline{v}_{i'})$ and $\beta_{j'} = 0$. Let f_2 be the univariate polynomial interpolated by the values α_i at point $\overline{v}_i$ for $d+1 \leq i \leq 2d$. This polynomial is not identically zero since $\alpha_{i'} = f(\overline{v}_{i'}) \neq 0$. Also the degree of the polynomial is at most d. Property P3 implies that f_2 takes the value $\beta_{j'} = 0$

at the point $\overline{u}_{j'}$. However, since the points $\{\overline{u}_j\}_{j=d+1}^{2d}$ are chosen uniformly at random on the curve L, the probability that one of them is a root of f_2 is at most $O(d^2/|\mathbb{F}|)$. Therefore, this case also occurs with probability at most $O(d^2/|\mathbb{F}|)$.

Case 7. There exist $1 \leq i', j' \leq d$ such that $\alpha_{i'} = 0$ and $\beta_{j'} = f(\overline{u}_{j'})$. This is analogous to Case 5 and omitting the analysis we conclude that it occurs with probability at most $O(d^2/|\mathbb{F}|)$.

Case 8. There exist $d+1 \leq i', j' \leq 2d$ such that $\alpha_{i'} = 0$ and $\beta_{j'} = f(\overline{u}_{j'})$. This is analogous to Case 6 and we omit the analysis to conclude that this case occurs with probability at most $O(d^2/|\mathbb{F}|)$ as well.

Combining the above completes the proof of the lemma. ◀

References

1 C. Ambühl, M. Mastrolilli, and O. Svensson. Inapproximability results for sparsest cut, optimal linear arrangement, and precedence constrained scheduling. In *Proc. 48th IEEE FOCS*, pages 329–337, 2007.

2 S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *J. ACM*, 45(5):753–782, 1998.

3 S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998.

4 S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *J. ACM*, 45(1):70–122, 1998.

5 Y. Dodis, V. Guruswami, and S. Khanna. The 2-catalog segmentation problem. In *SODA*, pages 897–898, 1999.

6 U. Feige. A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634–652, 1998.

7 U. Feige. Relations between average case complexity and approximation complexity. In *Proc. 34th ACM STOC*, pages 534–543, 2002.

8 J. Håstad. Clique is hard to approximate within n$^{\text{1-epsilon}}$. In *Proc. 37th IEEE FOCS*, pages 627–636, 1996.

9 J. Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001.

10 S. Khot. Ruling out PTAS for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM J. Comput.*, 36(4):1025–1071, 2006.

11 J. Kleinberg, C. Papadimitriou, and P. Raghavan. A microeconomic view of data mining. *Data Min. Knowl. Discov.*, 2(4):311–324, 1998.

12 M. Mitzenmacher. On the hardness of finding optimal multiple preset dictionaries. *IEEE Transactions on Information Theory*, 50(7):1536–1539, 2004.

13 R. Raz. A parallel repetition theorem. *SIAM J. Comput.*, 27(3):763–803, 1998.

14 R. Rubinfeld and M. Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM J. Comput.*, 25(2):252–271, 1996.

15 S. Sahni. Approximate algorithms for the 0/1 knapsack problem. *J. ACM*, 22(1):115–124, 1975.

16 Y. Yubo and X. Chengxian. Improved randomized algorithm for the equivalent 2-catalog segmentation problem. *Numerical Mathematics*, 14(2):128–135, 2005.