A Note on Approximating the 2-Catalog Segmentation Problem

Dachuan Xu* Yinyu Ye and Jiawei Zhang †

March 2002

Abstract

We present a .73-approximation algorithm for a disjoint 2-Catalog Segmentation and .63-approximation algorithm for the joint version of the problem. Previously best known results are .65 and .56, respectively. The results are based on semidefinite programming and a subtle rounding method.

Keywords: 2-Catalog Segmentation, Semidefinite Programming.

1 Introduction

In the 2-Catalog Segmentation problem (2-CSP), we are given a ground set I of n items and a family $\{S_1, S_2, \cdots, S_m\}$ of subsets of I, and it is desired to find subsets $A_1, A_2 \subset I$ such that $|A_1|, |A_2| \leq k$ and

$$\sum_{i=1}^{m} \max\{|S_i \cap A_1|, |S_i \cap A_2|\}$$

*Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100080, P.R. China. e-mail: xudc@lsec.cc.ac.cn. This work was partly supported by Chinese NSF grant 19731001.

†Department of Management Sciences, Henry, B. Tippie College of Business, The University of Iowa, Iowa City, IA, 52242, USA. e-mail: {yinyu-ye, jiawei-zhang}@uiowa.edu. This work was partly supported by NSF grant DMI-9908077.

is maximized. (2-CSP) was recently introduced by Kleinberg, Papadimitriou and Paghavan [7].

As observed by Dodis, Guruswami and Khanna [2], (2-CSP) is equivalent to the following graph partitioning problem: given a bipartite graph $G = (A, B, E)$ with $|A| = n$ and $|B| = m$, find a partition of $B = B_1 \cup B_2$, and two subsets A_1, A_2 of A, such that $|A_1| = |A_2| = k$ and $w(A_1, B_1) + w(A_2, B_2)$ is maximized, where $w(X, Y)$ denotes the number of edges with one end in X and the other in Y. Note that it is possible that $A_1 \cap A_2 \neq \emptyset$. If it is required that $A_1 \cap A_2 = \emptyset$, we call it Disjoint (2-CSP).

A greedy algorithm has been proposed in [7]. The algorithm simply selects A_1 to be the k nodes with most degrees and A_2 be any k nodes in A, and $B_1 = B$ and $B_2 = \emptyset$. It is easy to see that the simple greedy algorithm has a performance guarantee of $\frac{1}{2}$. Although the algorithm is trivial, it turns out to be the best possible approximation algorithm for general (2-CSP), since it has been shown recently by Asodi and Safra [1] that for any constant $\epsilon > 0$, the existence of a $(\frac{1}{2} + \epsilon)$-approximation algorithm would imply $P = NP$.

However, approximation algorithms with better performance guarantees have been studied for special cases of (2-CSP). In particular, a polynomial time approximation scheme exists for all dense instances of (2-CSP) in which Degree$(i) = \Omega(m)$ for $i \in B$, see [7]. Dodis, Guruswami and Khanna [2] developed a 0.56-approximation algorithm for the case where $k = n/2$. (Note that the case $k \geq n/2$ can be handled by adding $2k - n$ dummy nodes to A, reducing the problem to a $k = n/2$ case.) In addition, Disjoint (2-CSP) can be approximated by a factor of 0.65 when $k = n/2$, also see [2].

In this short note, we study the (2-CSP) under the assumption of $k = n/2$ as in [2]. We denote this case by (2-CSP_{2k}). The Disjoint (2-CSP_{2k}) problem is similarly defined. Our main results are a 0.73-approximation algorithm for Disjoint (2-CSP_{2k}), and a 0.63-approximation algorithm for (2-CSP_{2k}).

A closely related problem of Disjoint (2-CSP_{2k}) is the Max Bisection problem. As showed by Dodis, Guruswami and Khanna [2], using certain semidefinite programming (SDP) relaxation similar to the one for Max Bisection, one can obtain the same performance guarantee for Disjoint (2-CSP_{2k}) as that for Max Bisection. Thus, by the recent approximation results for Max Bisection [9, 6, 4], Disjoint (2-CSP_{2k}) can be approximated by a factor of 0.70.

For (2-CSP_{2k}), a "better of two" type algorithm has been used in [2] to obtain a good approximation performance guarantee. In particular, two algorithms are considered: one is the simple greedy algorithm which selects A_1 be the set of k largest-degree nodes in A and A_2 be any k nodes in A; and $B_1 = B$ and $B_2 = \emptyset$. The second is the SDP ρ-approximation algorithm for (2-CSP_{2k}) requiring $A_1 \cap A_2 = \emptyset$, that is, treating it as Disjoint (2-CSP_{2k}). It has been shown in [2] that the solution, which is the better of the two produced by the two algorithms, has the performance guarantee

$$R \geq \min_{0 \leq t \leq 1} \max\{\frac{1}{2-t}, (1 - \frac{t}{2-t})\rho\}. \tag{1}$$

In [2], $\rho = 0.651$ is used to get $R \geq 0.56$ by setting $\frac{1}{2-t} = (1 - \frac{t}{2-t})\rho$. This ratio can be improved to 0.58 if we use the results from [9, 6, 4] that $\rho = 0.70$.

By taking advantage of the fact that G is a bipartite graph, we are able to improve the rounding method of [9, 10] for SDP relaxation and obtain a 0.73-approximation for Disjoint (2-CSP_{2k}), that is, making $\rho = 0.73$.

Furthermore, we can tighten the inequality (1) to

$$R \geq \min_{0 \leq t \leq 1} \max\{\frac{1}{2-t}, (1 - \frac{t}{4-2t})\rho\}, \tag{2}$$

which enables us to prove a 0.63 performance guarantee for (2-CSP_{2k})

2 SDP relaxation for Disjoint (2-CSP_{2k})

Let $w_{ij} = 1$ if $i \in A, j \in B$ and $(i,j) \in E$; otherwise $w_{ij} = 0$. Then, Disjoint (2-CSP_{2k}) can be formulated as follows:

$$\begin{aligned} w^* := \quad & \text{Maximize} \quad \frac{1}{2} \sum_{i \in A, \quad j \in B} w_{ij}(1 + x_i x_j) \\ & \text{subject to} \quad \left(\sum_{i \in A} x_i \right)^2 = 0 \\ & \qquad\qquad\quad x_i^2 = 1, \quad i \in A \cup B. \end{aligned}$$

It is easy to see that the following is an SDP relaxation of Disjoint (2-CSP_{2k})

$$
\begin{array}{rll}
w^{SDP} := & \text{Maximize} & \frac{1}{2} \sum_{i \in A, \quad j \in B} w_{ij}(1 + X_{ij}) \\
& & \\
& \text{subject to} & \sum_{i,j \in A} X_{ij} = 0 \\
& & X_{ii} = 1, \quad i \in A \cup B \\
& & X \succeq 0.
\end{array}
\tag{3}
$$

Thus, we have $w^{SDP} \geq w^*$.

Let the nodes in A be indexed by $1, ..., |A|$ and the nodes in B indexed by $|A|+1, ..., |A|+|B|$, and let

$$
X^* = \begin{pmatrix} X^*_{AA} & X^*_{AB} \\ X^*_{BA} & X^*_{BB} \end{pmatrix}
$$

be an optimal SDP solution of (3), where the blocks of the matrix are partitioned according to the indices in A and B.

We first present an approximation algorithm for Disjoint (2-CSP_{2k}). The algorithm is similar to the one of [2] except Step 2: Randomized Rounding. Our new rounding method is an improved version of [9] and [10], and it can be derandomized by the technique of Mahajan and Ramesh [8].

1. SDP Solving: Solve (3) to obtain an optimal semidefinite symmetric matrix X^*.

2. Randomized Rounding: Given $0 \leq \theta < 1$, define two positive semidefinite matrices

$$
I_A(\theta) = \begin{pmatrix} (1-\theta)I & 0 \\ 0 & 0 \end{pmatrix}
$$

and

$$
X^*(\theta) = \begin{pmatrix} \theta X^*_{AA} & \sqrt{\theta} X^*_{AB} \\ \sqrt{\theta} X^*_{BA} & X^*_{BB} \end{pmatrix}.
$$

Then, generate a vector u from a multivariate normal distribution with 0 mean and the covariance matrix $X^*(\theta) + I_A(\theta)$, i.e.,

$$
u \in N(0, X^*(\theta) + I_A(\theta)),
$$

then assign

$$
\hat{x} = \text{sign}(u),
$$

i.e.,

$$\hat{x}_i = \begin{cases} 1 & \text{if } u_i \geq 0 \\ -1 & \text{if } u_i < 0. \end{cases}$$

Select blocks $\hat{A}_1 = \{i \in A : \hat{x}_i = 1\}$ and $\hat{A}_2 = A \setminus \hat{A}_1 = \{i \in A : \hat{x}_i = -1\}$, $B_1 = \{j \in B : \hat{x}_j = 1\}$ and $B_2 = B \setminus B_1 = \{j \in B : \hat{x}_j = -1\}$. Without losing of generality, we assume $|\hat{A}_1| \geq k = |A|/2$.

3. Node Swapping: For each $i \in \hat{A}_1$, let $\zeta(i) = \sum_{j \in B_1} w_{ij}$, and reorder the nodes in $\hat{A}_1$ such that $\zeta(i_1) \geq \zeta(i_2) \geq ... \geq \zeta(i_{|\hat{A}_1|})$. Let $A_1 = \{i_1, i_2, ..., i_k\}$ and $A_2 = A \setminus A_1$. ■

The following inequality is straightforward:

$$w(A_1, B_1) + w(A_2, B_2) \geq \frac{k}{|\hat{A}_1|} \cdot \left(w(\hat{A}_1, B_1) + w(\hat{A}_2, B_2) \right). \tag{4}$$

In order to analyze the quality of the partitions (A_1, A_2) and (B_1, B_2), we define two random variables similar to that in [3]:

$$w := w(\hat{A}_1, B_1) + w(\hat{A}_2, B_2) = \frac{1}{2} \sum_{i \in A,\ j \in B} w_{ij}(1 + \hat{x}_i \hat{x}_j)$$

and

$$M := |\hat{A}_1|(2k - |\hat{A}_1|) = \frac{1}{4} \sum_{i,j \in A} (1 - \hat{x}_i \hat{x}_j).$$

Lemma 1 *Our approximation method yields the partitions* $(\hat{A}_1, \hat{A}_2)$ *and* (B_1, B_2)*, satisfying the following two inequalities:*

$$E[w] \geq \alpha \cdot w^{SDP},$$

$$E[M] \geq \beta \cdot k^2.$$

where $\alpha := \alpha(\theta)$, $\beta := \beta(\theta)$, *and*

$$\alpha(\theta) = \min_{-1 < y \leq 1} \frac{1 + \frac{2}{\pi} \arcsin(\sqrt{\theta} y)}{1 + y}, \tag{5}$$

$$\beta(\theta) = 1 - \frac{2}{\pi} \arcsin(\theta) - \frac{1 - \frac{2}{\pi} \arcsin(\theta)}{n} + \min_{-1 \leq y < 1} \frac{2}{\pi} \frac{\arcsin(\theta) - \arcsin(\theta y)}{1 - y}, \tag{6}$$

Proof. From Lemma 2.2 of [5], the randomized rounding in our algorithm leads to the following:

$$\mathrm{E}[\hat{x}_i \hat{x}_j] = \frac{2}{\pi} \arcsin(\sqrt{\theta} X^*_{ij}), \quad i \in A, \quad j \in B \tag{7}$$

and

$$\mathrm{E}[\hat{x}_i\hat{x}_j] = \frac{2}{\pi}\arcsin(\theta X^*_{ij}), \quad i,j \in A, \quad i \neq j. \tag{8}$$

Using the same argument in [9] and by the definition of α from (5), and β from (6), we completes the proof of the lemma. ■

Consider a new random variable

$$z(\gamma) := \frac{w}{w^{SDP}} + \gamma\frac{M}{k^2}, \tag{9}$$

where

$$\gamma = \frac{\alpha}{2\beta}(\frac{1}{\sqrt{1-\beta}} - 1). \tag{10}$$

By Lemma 2, we have

$$\mathrm{E}[z(\gamma)] \geq \alpha + \gamma\beta \quad \text{and} \quad z(\gamma) \leq 1 + \gamma.$$

Now we can state the following Lemma, whose proof was established in Lemma 1 of [9]

Lemma 2 *If the random variable $z(\gamma)$ meets its expectation, i.e., $z(\gamma) \geq \alpha + \gamma\beta$, then*

$$w(A_1, B_1) + w(A_2, B_2) \geq \frac{\alpha}{1+\sqrt{1-\beta}} w^{SDP} \geq \frac{\alpha}{1+\sqrt{1-\beta}} w^*.$$

Now, if we choose $\theta = 0.75$, then we have $\alpha(\theta) = 0.8252$ and $\beta(\theta) = 0.9831$. By Lemma 2, we derive a 0.73-approximation algorithm for Disjoint $(2 - CSP_{2k})$.

3 Approximation for (2-CSP_{2k})

We consider the following two algorithms as in [2] for (2-CSP_{2k}).

Algorithm 1. Let $A_1^{(1)}$ be the subset of the k largest degree nodes in A and $A_2^{(1)} = A \setminus A_1^{(1)}$, and $B_1^{(1)} = B$ and $B_2^{(1)} = \emptyset$.

Algorithm 2. Let $(A_1^{(2)}, A_2^{(2)})$ and $(B_1^{(2)}, B_2^{(2)})$ be the subsets produced by the SDP based algorithm in Section 2 for the disjoint case.

Our main result here is

Lemma 3 *The algorithm which outputs the better of the two solutions of Algorithm 1 and Algorithm 2 has the performance guarantee*

$$R \geq \min_{0\leq t\leq 1} \max\{\frac{1}{2-t}, (1-\frac{t}{4-2t})\rho\},$$

where ρ is the performance guarantee of Algorithm 2.

Proof. Suppose that the w^{OPT} and w^* are the optimal values for general (2-CSP_{2k}) and Disjoint (2-CSP_{2k}), respectively. Also, let (A_1^*, A_2^*) and $B = (B_1^*, B_2^*)$ be an optimal solution of general (2-CSP_{2k}).

We observe that

$$w(A_1^*, B_1^*) + w(A_2^*, B_2^*) \leq w(A_1^*, B_1^*) + w(A \setminus A_1^*, B_2^*) + w(A_1^* \cap A_2^*, B_2^*),$$

and

$$w(A_1^*, B_1^*) + w(A_2^*, B_2^*) \leq w(A_2^*, B_2^*) + w(A \setminus A_2^*, B_1^*) + w(A_1^* \cap A_2^*, B_1^*).$$

Furthermore,

$$w(A_1^* \cap A_2^*, B_2^*) + w(A_1^* \cap A_2^*, B_1^*) = w(A_1^* \cap A_2^*, B).$$

Therefore, we have

$$\begin{aligned} & \max\{w(A_1^*, B_1^*) + w(A \setminus A_1^*, B_2^*), w(A_2^*, B_2^*) + w(A \setminus A_2^*, B_1^*)\} \\ \geq\ & w(A_1^*, B_1^*) + w(A_2^*, B_2^*) - \frac{1}{2}w(A_1^* \cap A_2^*, B) \\ =\ & w^{OPT} - \frac{1}{2}w(A_1^* \cap A_2^*, B) \end{aligned}$$

On the other hand, the partitions $(A_i^*, A \setminus A_i^*)$ and (B_1^*, B_2^*), $i = 1, 2$, are feasible solutions for Disjoint 2-CSP_{2k}. Thus, we must have

$$w^* \geq w^{OPT} - \frac{1}{2}w(A_1^* \cap A_2^*, B). \tag{11}$$

Now, for any given $t \in [0, 1]$, if $w(A_1^{(1)}, B) \geq \frac{1}{2-t}w^{OPT}$ (note it is always true that $w(A_1^{(1)}, B) \geq \frac{1}{2}w^{OPT}$), then

$$w(A_1^{(1)}, B_1^{(1)}) + w(A_2^{(1)}, B_2^{(1)}) = w(A_1^{(1)}, B) \geq \frac{1}{2-t}w^{OPT}.$$

Otherwise, we have

$$w(A_1^* \cap A_2^*, B) \leq \frac{t}{2-t}w^{OPT}$$

which follows from

$$\begin{aligned} w^{OPT} &\leq w(A_1^* \cup A_2^*, B) \\ &= w(A_1^*, B) + w(A_2^*, B) - w(A_1^* \cap A_2^*, B) \\ &\leq 2w(A_1^{(1)}, B) - w(A_1^* \cap A_2^*, B) \\ &\leq \frac{2}{2-t} w^{OPT} - w(A_1^* \cap A_2^*, B). \end{aligned}$$

Then, from (11) we have

$$w^* \geq w^{OPT} - \frac{t}{4-2t} w^{OPT}.$$

Therefore, using the ρ-approximation algorithm for Disjoint (2-CSP_{2k}) we have partitions $A = (A_1^{(2)}, A_2^{(2)})$ and $B = (B_1^{(2)}, B_2^{(2)})$ such that

$$w(A_1^{(2)}, B_1^{(2)}) + w(A_2^{(2)}, B_2^{(2)}) \geq \rho \cdot w^* \geq (1 - \frac{t}{4-2t})\rho \cdot w^{OPT}.$$

The desired results thus follow. ■

We have proved that $\rho \geq 0.73$. Then, setting $\frac{1}{2-t} = (1 - \frac{t}{4-2t})\rho$ (that is, $t = .42$), we develop the main result of this section:

Theorem 1 *(2-CSP_{2k}) can be approximated with a factor of at least* 0.63*.*

References

[1] V. Asodi and S. Safra, "On the Complexity of the Catalog-Segmentation Problem," *Manuscript*, 2001.

[2] Y. Doids, V. Guruswami, and S. Khanna, "The 2-catalog segmentation problem," *SODA*, 1998.

[3] A. Frieze and M. Jerrum, "Improved approximation algorithms for max k-cut and max bisection," *Algorithmica* 18(1997) 67-81.

[4] U. Feige and M. Langberg, "The RPR^2 rounding technique for semidefinite programs," ICALP, 2001.

[5] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for Maximum Cut and Satisfiability problems using semidefinite programming," Journal of the ACM, 42 (1995), pp. 1115-1145.

[6] E. Halperin and U. Zwick, *A unified framework for obtaining improved approximation algorithms for maximum graph bisection problems*, Manuscript, 2001.

[7] J. Kleinberg, C. Papadimitriou and P. Raghavan, "Segmentation Problems," *STOC 98*, pp. 473-482.

[8] S. Mahajan and H. Ramesh, "Derandomizing semidefinite programming based approximation algorithms," *SIAM J. of Computing*, 28, pp. 1641–1663, 1999.

[9] Y. Ye, "A .699-approximation algorithm for Max-Bisection," *Mathematical Programming*, 90 (2001), pp. 101-111.

[10] U. Zwick, *Outward rotations: a tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems*, in Proceedings of the 30th Symposium on Theory of Computation (STOC), 1999, pp. 551-560.