Approximate the 2-Catalog Segmentation Problem Using Semidefinite Programming Relaxations

Dachuan Xu* Yinyu Ye† Jiawei Zhang ‡

May 6, 2002

Abstract

We consider the 2-Catalog Segmentation problem (2-CSP) introduced by Kleinberg, Papadimitriou and Raghavan [10]. In this problem, we are given a ground set I of n items, a family $\{S_1, S_2, \cdots, S_m\}$ of subsets of I and an integer $1 \leq k \leq n$. The objective is to find subsets $A_1, A_2 \subset I$ such that $|A_1| = |A_2| = k$ and $\sum_{i=1}^{m} \max\{|S_i \cap A_1|, |S_i \cap A_2|\}$ is maximized. It is known that a simple greedy algorithm has a performance guarantee $\frac{1}{2}$ [10]. Furthermore, using an semidefinite programming (SDP) relaxation Doids, Guruswami and Khanna [2] showed that 2-CSP can be approximated by a factor of 0.56 when $k = n/2$. On the negative side, Asodi and Safra [1] proved that a polynomial time $(\frac{1}{2} + \epsilon)$-approximation algorithm, for any constant $\epsilon > 0$, would imply $NP = P$. Motivated by the results of [2] and [1], we develop 'better-than-$\frac{1}{2}$' approximation algorithms for 2-CSP on a range of k in this paper. The performance guarantee of our algorithm is $\frac{1}{2}$ for general k, and is strictly greater than $1/2$ when $k \geq \frac{n}{3}$. In particular, we obtain a ratio of 0.67 for 2-CSP when $k = \frac{n}{2}$, which improves the previously known best result 0.56 [2]. Unlike the relaxation of [2], our new and direct SDP relaxation deals with general k, which enables us to obtain a better approximation for 2-CSP. Another contribution of this paper is a new variation of the random hyperplane rounding technique, which allows us to explore the structure of the problem under consideration. This rounding technique might be of independent interest. It can be also used to obtain improved approximations for several other graph partitioning problems considered by Feige & Langber [5], Ye & Zhang [12], and Halperin & Zwick [8].

*Institute of Computational Mathematics and Scientific/Engineering Computing, 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 Science and Engineering, Stanford University, Standford, CA, 94305-4026, USA. e-mail: yinyu-ye@stanford.edu. This work was partly supported by NSF grant DMI-9908077.

‡Department of Management Sciences, The University of Iowa, Iowa City, IA, 52242, USA. e-mail: jiazhang@stanford.edu. This work was partly supported by NSF grant DMI-9908077.

1 Introduction

In the 2-Catalog Segmentation problem (2-CSP), we are given a ground set I of n items, a family $\{S_1, S_2, \cdots, S_m\}$ of subsets of I and an integer $1 \leq k \leq n$. The objective is to find two subsets $A_1, A_2 \subset I$ such that $|A_1| = |A_2| = k$ and

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

is maximized. This problem, motivated by applications to data mining and common marketing questions, was recently introduced by Kleinberg, Papadimitriou and Raghavan [10].

An equivalent formulation of 2-CSP, also called its partition version, is the following: 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. Here, $w(X, Y)$ 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 the problem Disjoint 2-CSP.

1.1 Previous results

A greedy algorithm has been proposed in [10]. The algorithm simply selects $A_1 \in A$ to be the k nodes of largest degrees, $A_2 \in A$ be any k nodes in A, $B_1 = B$, $B_2 = \emptyset$. It is evident that the simple greedy algorithm has a performance guarantee of $\frac{1}{2}$. Although the algorithm is trivial, it turns out that no other polynomial time algorithms can do better (in the sense of worst case performance) if $P \neq NP$, because 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 $P = NP$.

With the strong hardness result by Asodi and Safra, we hope to find 'better-than-$\frac{1}{2}$' approximation algorithms for subcases of 2-CSP. Such subcases have been studied recently. In particular, Kleinberg, Papadimitriou and Paghavan [10] showed that a polynomial time approximation scheme exists for all dense instances of 2-CSP, in which each node of B has a degree of a constant fraction of n. Dodis, Guruswami and Khanna [2] developed a 0.56-approximation algorithm for the case $k = n/2$.

In [2], a semidefinite programming (SDP) relaxation is proposed for Disjoint 2-CSP with $k = n/2$. In this case, each node i of A is either in A_1 or A_2. Then we can assign a binary variable x_i for i such that $x_i = 1$ if and only if $i \in A_1$, and $x_i = -1$ otherwise. This leads to a binary quadratic formulation and then a straightforward SDP relaxation. Following the approach of Frieze and Jerrum [6] for Max-Bisection, Dodis, Guruswami and Khanna proved that this special case can be approximated by a factor of 0.65. Their SDP relaxation, however, cannot be extended to the case $A_1 \cap A_2 \neq \emptyset$ even for $k = n/2$. A factor of 0.56, for the case $k = n/2$ but $A_1 \cap A_2 \neq \emptyset$, was obtained by combining the SDP based approximation and a simple greedy algorithm. We remark here that combing the recent results for Max-Bisection of Ye [11], Halperin & Zwick [8] and Feige & Langberg [4] with that of [2], one can obtain a ratio of 0.58 for the case $k = n/2$.

1.2 Our results and techniques

In this paper, motivated by the results of [1] and [2], we attempt to answer the following question: in what range of k, can 2-CSP be approximated better than $\frac{1}{2}$ in polynomial time? We develop an approximation algorithm which has a performance guarantee of $\frac{1}{2}$ for general k, and the ratio is strictly greater than $\frac{1}{2}$ when $k \geq \frac{n}{3}$. In particular, we show that 2-CSP can be approximated by a factor of 0.67 when $k = n/2$, which improves the result 0.56 of [2].

Our algorithm is based on a 'direct' SDP relaxation of general 2-CSP, which is unknown before. For general 2-CSP, each node i of A has four possible choices: $A1 \setminus A2$, $A2 \setminus A1$, $A1 \cap A2$ and $A \setminus (A1 \cup A2)$. Therefore, one binary variable cannot characterize all choices of node i. For our purpose, we design a two-dimensional vector (x_i, y_i) for each node $i \in A$ such that $x_i = 1$ iff $i \in A_1$, otherwise $x_i = -1$; and $y_i = 1$ iff $i \in A_2$, otherwise $y_i = -1$. We assign variable z_j for each node $j \in B$ such that $z_j = 1$ iff $i \in B_1$, otherwise $z_j = -1$. A quadratic formulation can thus be obtained by using these vector variables together with an additional reference variable u_0, which makes the objective function and constraints homogeneous:

$$\begin{array}{rll} w_* := & \text{Maximize} & \frac{1}{4} \sum_{i \in A,\ j \in B} w_{ij}(2 + u_0 x_i + u_0 y_i + x_i z_j - y_i z_j) \\ & \text{subject to} & \sum_{i \in A} u_0 x_i = \sum_{i \in A} u_0 y_i = 2k - n, \\ & & u_0^2 = x_i^2 = y_i^2 = z_j^2 = 1 \quad i \in A, j \in B \end{array} \tag{1}$$

Let vector $u = (u_0;\ x \in R^n;\ y \in R^n;\ z \in R^m)$ and consider matrix $U = uu^T$. Then, it is straightforward to obtain an SDP relaxation of (1):

$$\begin{array}{rll} w_{SDP} := & \text{Maximize} & \frac{1}{4} \sum_{1 \leq i \leq n,\ 2n+1 \leq j \leq 2n+m} w_{ij}(2 + U_{0i} + U_{0(n+i)} + U_{ij} - U_{(n+i)j}) \\ & \text{subject to} & \sum_{i=1}^{n} U_{0i} = \sum_{i=n+1}^{2n} U_{0i} = 2k - n, \\ & & \sum_{1 \leq i,j \leq n} U_{ij} = \sum_{n+1 \leq i,j \leq 2n} U_{ij} = (2k-n)^2, \\ & & U_{ii} = 1, \quad i = 0, 1, ..., 2n + m \\ & & U \succeq 0. \end{array} \tag{2}$$

In order to obtain a $\{1, -1\}$ solution, we could apply the standard random hyperplane rounding technique developed by Goemans and Williamson [7]. Roughly speaking, once we get the optimal SDP solution, say U^*, we factorize it $U^* = V^T V$ where $V = (v_0, V_x, V_y, V_z)$ is a $(2n + m + 1) \times (2n + m + 1)$ matrix. Then randomly choose a unit vector u from S^{2n+m}. For each node i, if $u \cdot v_i$ has the same sign as $u \cdot v_0$, then it is 1; otherwise it is -1. This randomized rounding produces a solution which has an expected objective value close to the optimal one, and the expected number of nodes in A_1 or A_2 are close to k as desired. The problem is that the variances of $|A_1|$ or $|A_2|$ could be very large.

Outward rotations, proposed by Zwick [13] and Ye [11], is to rotate the vector v_i, for $i \in A, B, C$ prior to the hyperplane rounding. It can be used to reduce the variances of $|A_1|$ and $|A_2|$, with a small negative effect on the quality of the expected objective value. This technique has been generalized by Han, Ye & Zhang [9] for the purpose of designing better approximations for several graph partitioning problems.

These techniques could be directly applied to approximating 2-CSP. But we want more. Recall that $G = (A, B, E)$ is bipartite in 2-CSP, that is, the edges exist only between A and B, and the cardinality constraints are only on the nodes in A. Therefore, we can rotate the vectors non-uniformly According to which set the nodes belong to. Compared with *outward rotations*, the advantage of this rotation is that, in a randomized solution, the expected objective value and the variances of $|A_1|$ and $|A_2|$ can be balanced even better, which lead to improved overall approximation for 2-CSP.

We note that in [3], Feige and Goemans proposed another rotation technique. Unlike our rotation, their rotation has a positive effect on the expected objective value and a negative effect on the variances of $|A_1|$ and $|A_2|$. With this observation, we show that combining our rotation with Feige-Goemans's can further improve the overall approximation for 2-CSP.

1.3 Organization of the paper

In the next section, we present the details of our SDP relaxation and randomized rounding technique. The analysis and result of the algorithm will be given in Section 3. In Section 4, we combine the Feige-Goemans rotation with ours to generate even better results. Few concluding remarks are presented in Section 5.

2 Randomized algorithm

Before we present our randomized rounding, we introduce some matrices which will be used later. Assume that $\bar{U} \in R^{(2n+m+1)\times(2n+m+1)}$ is the optimal matrix solution of 2. For our purpose, $\bar{U} \succeq 0$ will be decomposed as

$$\bar{U} = \begin{pmatrix} \bar{U}_{00} & \bar{U}_{0x} & \bar{U}_{0y} & \bar{U}_{0z} \\ \bar{U}_{x0} & \bar{U}_{xx} & \bar{U}_{xy} & \bar{U}_{xz} \\ \bar{U}_{y0} & \bar{U}_{yx} & \bar{U}_{yy} & \bar{U}_{yz} \\ \bar{U}_{z0} & \bar{U}_{zx} & \bar{U}_{zy} & \bar{U}_{zz} \end{pmatrix}$$

according to sub-blocks corresponding to u_0, x, y, and z.

Denote $\sigma = k/n$ and $\kappa = 2\sigma - 1$. Given $0 \le \theta < 1$, define

$$\bar{U}(\theta) = \begin{pmatrix} \bar{U}_{00} & \sqrt{\theta}\bar{U}_{0x} & \sqrt{\theta}\bar{U}_{0y} & \bar{U}_{0z} \\ \sqrt{\theta}\bar{U}_{x0} & \theta\bar{U}_{xx} & \theta\bar{U}_{xy} & \sqrt{\theta}\bar{U}_{xz} \\ \sqrt{\theta}\bar{U}_{y0} & \theta\bar{U}_{yx} & \theta\bar{U}_{yy} & \sqrt{\theta}\bar{U}_{yz} \\ \bar{U}_{z0} & \sqrt{\theta}\bar{U}_{zx} & \sqrt{\theta}\bar{U}_{zy} & \bar{U}_{zz} \end{pmatrix}$$

and

$$Q_{n\times n} = \begin{pmatrix} 1 & \kappa^2 & \cdots & \kappa^2 \\ \kappa^2 & 1 & \cdots & \kappa^2 \\ \cdots & \cdots & \cdots & \cdots \\ \kappa^2 & \kappa^2 & \cdots & 1 \end{pmatrix}, \quad P = \begin{pmatrix} 0 & 0_{1\times n} & 0_{1\times n} & 0_{1\times m} \\ 0_{n\times 1} & Q_{n\times n} & \kappa^2 E_{n\times n} & 0_{n\times m} \\ 0_{n\times 1} & \kappa^2 E_{n\times n} & Q_{n\times n} & 0_{n\times m} \\ 0_{m\times 1} & 0_{m\times n} & 0_{m\times n} & 0_{m\times m} \end{pmatrix}$$

where E is an all one matrix and 0 is an all zero matrix.

The following can be easily verified and is a key to our randomized rounding.

Lemma 1 *$\bar{U}(\theta)$ amd P are positive semidefinite matrix for any given $\theta \in [0, 1)$. Furthermore, all the diagonal entries of $\bar{U}(\theta) + (1-\theta)P$ are 1.*

Randomized Rounding: Generate vectors $(u_0;\ u_x;\ u_y;\ u_z) \in N(0, \bar{U}(\theta) + (1-\theta)P)$,, i.e., multi-normal distribution with mean 0 and covariance matrix $\bar{U}(\theta) + (1-\theta)P$. Then assign

$$\hat{u}_0 = \text{sign}(u_0), \quad \hat{x} = \text{sign}(u_x), \quad \hat{y} = \text{sign}(u_y) \quad \text{and} \quad \hat{z} = \text{sign}(u_z).$$

Let the subsets $\hat{A}_1 = \{i \in A : \hat{x}_i = \hat{u}_0\}$, $\hat{A}_2 = \{i \in A : \hat{y}_i = \hat{u}_0\}$, $B_1 = \{j \in B : \hat{z}_j = \hat{u}_0\}$ and $B_2 = B \setminus B_1$.

From Lemma 1, the above randomized rounding possesses (see [7] or [9]):

$$\begin{aligned}
E[\hat{x}_i\hat{u}_0] &= \frac{2}{\pi}\arcsin(\sqrt{\theta}\bar{U}_{0i}), && i \in A,\\
E[\hat{y}_i\hat{u}_0] &= \frac{2}{\pi}\arcsin(\sqrt{\theta}\bar{U}_{0(m+i)}), && i \in A,\\
E[\hat{z}_j\hat{u}_0] &= \frac{2}{\pi}\arcsin(\bar{U}_{0j}), && j \in B,\\
E[\hat{x}_i\hat{z}_j] &= \frac{2}{\pi}\arcsin(\sqrt{\theta}\bar{U}_{ij}), && i \in A,\ j \in B,\\
E[\hat{y}_i\hat{z}_j] &= \frac{2}{\pi}\arcsin(\sqrt{\theta}\bar{U}_{(m+i)j}), && i \in A,\ j \in B.
\end{aligned}$$

These equations will be used to prove the quality of subsets $\hat{A}_1$, $\hat{A}_2$, B_1 and B_2.

However, the above randomized rounding may not produce a feasible solution, i.e., neither $|\hat{A}_1|$ nor $|\hat{A}_2|$ equals k. Therefore, we need the following size-adjusting procedure to get a feasible solution.

Adjusting Procedure: Let A_1, A_2 be the subsets of A, each with cardinality k, such that $w(A_1, B_1)$ and $w(A_2, B_2)$ are maximized respectively.

Note that in the adjusting procedure, A_1 and A_2 has nothing to do with $\hat{A}_1$ and $\hat{A}_2$. They depend only on B_1 and B_2. However, $\hat{A}_1$ and $\hat{A}_2$ will be used in analyzing the quality of A_1 and A_2, as we'll see from the following lemma.

Lemma 2 *The subsets (A_1, A_2) and (B_1, B_2) satisfy, with probability 1,*

$$|A_i| = k, \quad i = 1, 2$$

$$w(A_1,B_1) + w(A_2,B_2) \geq \begin{cases} \frac{k}{|\hat{A}_1|}\cdot w(\hat{A}_1,B_1) + \frac{k}{|\hat{A}_2|}\cdot w(\hat{A}_2,B_2) & \text{if } |\hat{A}_i| \geq k,\ i=1,2;\\ w(\hat{A}_1,B_1) + \frac{k}{|\hat{A}_2|}\cdot w(\hat{A}_2,B_2) & \text{if } |\hat{A}_1| \leq k,\ |\hat{A}_2| \geq k;\\ \frac{k}{|\hat{A}_1|}\cdot w(\hat{A}_1,B_1) + w(\hat{A}_2,B_2) & \text{if } |\hat{A}_1| \geq k,\ |\hat{A}_2| \leq k;\\ w(\hat{A}_1,B_1) + w(\hat{A}_2,B_2) & \text{if } |\hat{A}_i| \leq k,\ i=1,2, \end{cases} \tag{3}$$

and

$$w(A_1,B_1) + w(A_2,B_2) \geq \max\{\frac{1}{2}, \frac{k}{n}\}w_*.$$

Proof : The first two claims are straightforward. The last inequality follows from the following observation. Consider subset $A' \in A$ such that $|A'| \leq k$ and $w(A', B)$ is maximized. It is easy to see that

$$w(A',B) \geq \frac{1}{2}w_* \quad \text{and} \quad w(A',B) \geq \frac{k}{n}w_*.$$

On the other hand, we have

$$w(A',B) = w(A',B_1) + w(A',B_2) \leq w(A_1,B_1) + w(A_2,B_2).$$

□

The lemma leads to

Corollary 3 *Our algorithm has a performance guarantee* $\max\{\frac{1}{2}, \frac{k}{n}\}$.

3 Analysis of the algorithm

We now analyze the (expected) quality of $w(A_1, B_1) + w(A_2, B_2)$, where inequality (3) is a key to prove a lower bound for our algorithm.

Since $|\hat{A}_1|$, $|\hat{A}_2|$, $w(\hat{A}_1, B_1)$ and $w(\hat{A}_2, B_2)$ are all random variables, it is hard to bound the expectation of the product of two random variables such as $k/|\hat{A}_1| \cdot w(\hat{A}_1, B_1)$. Therefore, we follow the approach of Frieze and Jerrum [6] to construct an artificial random variable, for given parameters $\gamma \geq 0$ and $\eta \in [-1, 1]$:

$$z(\eta, \gamma) := \frac{w}{w_{SDP}} + \gamma \frac{M_1 + M_2}{n^2} + \eta\gamma \frac{p_1 + p_2}{n}, \tag{4}$$

where $w := w(\hat{A}_1, B_1) + w(\hat{A}_2, B_2) = \frac{1}{4} \sum_{i \in A,\ j \in B} w_{ij}(2 + \hat{x}_i \hat{u}_0 + \hat{y}_i \hat{u}_0 + \hat{x}_i \hat{z}_j - \hat{y}_i \hat{z}_j)$,

$$p_1 := |\hat{A}_1| = \frac{1}{2} \sum_{i \in A} (1 + \hat{x}_i \hat{u}_0), \quad p_2 := |\hat{A}_2| = \frac{1}{2} \sum_{i \in A} (1 + \hat{y}_i \hat{u}_0),$$

and

$$M_1 := |\hat{A}_1|(n - |\hat{A}_1|) = \frac{1}{4} \sum_{i,j \in A} (1 - \hat{x}_i \hat{x}_j), \quad M_2 := |\hat{A}_2|(n - |\hat{A}_2|) = \frac{1}{4} \sum_{i,j \in A} (1 - \hat{y}_i \hat{y}_j).$$

It is easy to see that, if σ is a constant then $z(\eta, \gamma)$ is bounded above by a constant which depends on the given (constants) γ, η and σ. Now, if we can show that, for some $\mu_{\eta,\gamma} > 0$,

$$\mathrm{E}[z(\eta, \gamma)] \geq \mu_{\eta,\gamma}, \tag{5}$$

then, for any constant $\epsilon > 0$, we can generate $\hat{A}_1, \hat{A}_2$, B_1 and B_2, in polynomial time, such that

$$z(\eta, \gamma)] \geq \mu_{\eta,\gamma} - \epsilon.$$

For simplicity, we denote $\mu_{\eta,\gamma}$ as μ and assume that $z(\eta, \gamma) \geq \mu$ in the rest of our analyses.

We have the following theorem. (Since the proof of the theorem is technique and long, we leave it to the Appendices.)

Theorem 4 *If $z(\eta, \gamma)$ meets its expectation, i.e., $z(\eta, \gamma) \geq \mu$, then*

$$w(A_1, B_1) + w(A_2, B_2) \geq \min_{i=1,2} \{R_i(\eta, \gamma, \theta)\} \cdot w_*.$$

where

$$R_1(\eta, \gamma, \theta) = \sigma \left(2\sqrt{\gamma(\mu - (\frac{1+\eta}{2})^2 \gamma)} - \gamma - \eta\gamma \right),$$

and

$$R_2(\eta, \gamma, \theta) = \begin{cases} \mu - 2\gamma\sigma(1 - \sigma + \eta), & \text{if } \eta \geq 2\sigma - 1; \\ \mu - 2\gamma(\frac{1+\eta}{2})^2, & \text{otherwise.} \end{cases}$$

In the rest of this section, we are bounding w, p_1, p_2, M_1 and M_2, and then bounding μ of (5).

For a given $\theta \in [0,1]$, let

$$\alpha = \min_{(x,y,z)\in\mathcal{F}} \frac{1+\frac{2}{\pi}(\arcsin(\sqrt{\theta}x)+\arcsin(y)+\arcsin(\sqrt{\theta}z))}{1+x+y+z},$$

where

$$\mathcal{F} := \left\{ (x,y,z) \;\middle|\; \begin{array}{l} 1+x+y+z>0, \\ \begin{pmatrix} 1 & x & y \\ x & 1 & z \\ y & z & 1 \end{pmatrix} \succeq 0. \end{array} \right\};$$

$$\tau^+ = \tau^- = \min_{-1\le x\le 1} \frac{2}{\pi}\frac{\arcsin(\sqrt{\theta})-\arcsin(\sqrt{\theta}x)}{1-x};$$

$$\beta^+ = 1-\frac{2}{\pi}\arcsin(\sqrt{\theta})+2\sigma\tau^+;$$

$$\beta^- = 1-\frac{2}{\pi}\arcsin(\sqrt{\theta})+2(1-\sigma)\tau^-;$$

$$\tau = \min_{-1\le x\le 1} \frac{2}{\pi}\frac{\arcsin(\theta+(1-\theta)\kappa^2)-\arcsin(\theta x+(1-\theta)\kappa^2)}{1-x};$$

$$\beta = 1-\frac{2}{\pi}\arcsin(\theta+(1-\theta)\kappa^2)+4\sigma(1-\sigma)\tau.$$

We have the following inequalities regarding the expected values of w, q_1, q_2, M_1 and M_2.

Lemma 5 *Let*

$$\beta_1 = \begin{cases} \frac{\beta^+}{2} & \text{if } \eta \ge 0, \\ \frac{2-\beta^-}{2} & \text{otherwise;} \end{cases}$$

and

$$\beta_2 = \frac{\beta-(\beta-4\sigma(1-\sigma)\tau)/n}{4}.$$

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

$$E[w] \ge \alpha\cdot w_{SDP}, \tag{6}$$

$$E[\frac{p_i}{n}] \ge \beta_1, \quad i=1,2, \tag{7}$$

$$E[\frac{M_i}{n^2}] \ge \beta_2 \quad i=1,2. \tag{8}$$

Proof : Define

$$\alpha' = \min_{(x,y,z)\in\mathcal{F}'} \frac{1+\frac{2}{\pi}(\arcsin(\sqrt{\theta}x)-\arcsin(y)-\arcsin(\sqrt{\theta}z))}{1+x-y-z},$$

where

$$\mathcal{F}' := \left\{ \begin{array}{c|c} (x,y,z) & \begin{array}{l} 1+x-y-z>0, \\ \begin{pmatrix} 1 & x & y \\ x & 1 & z \\ y & z & 1 \end{pmatrix} \succeq 0. \end{array} \end{array} \right\}.$$

If we set $x = x'$, $y = -y'$ and $z = -z'$, it is easy to verify that $\alpha = \alpha'$ according to the following fact

$$\begin{pmatrix} 1 & x & y \\ x & 1 & z \\ y & z & 1 \end{pmatrix} \succeq 0 \iff \begin{pmatrix} 1 & x & -y \\ x & 1 & -z \\ -y & -z & 1 \end{pmatrix} \succeq 0.$$

Then, we have

$$\begin{aligned} E[w] &= \frac{1}{4}\sum_{i\in A,\ j\in B} w_{ij}(2+E[\hat{x}_i\hat{u}_0]+E[\hat{y}_i\hat{u}_0]+E[\hat{x}_i\hat{z}_j]-E[\hat{y}_i\hat{z}_j]) \\ &= \frac{1}{4}\sum_{i\in A,\ j\in B} w_{ij}\{(1+E[\hat{x}_i\hat{u}_0]+E[\hat{z}_j\hat{u}_0]+E[\hat{x}_i\hat{z}_j]) \\ &\qquad\qquad +(1+E[\hat{y}_i\hat{u}_0]-E[\hat{z}_j\hat{u}_0]-E[\hat{y}_i\hat{z}_j])\} \\ &= \frac{1}{4}\sum_{i\in A,\ j\in B} w_{ij}\{[1+\frac{2}{\pi}(\arcsin(\sqrt{\theta}\bar{U}_{0i})+\arcsin(\bar{U}_{0j})+\arcsin(\sqrt{\theta}\bar{U}_{ij}))] \\ &\qquad +[1+\tfrac{2}{\pi}(\arcsin(\sqrt{\theta}\bar{U}_{0(m+i)})-\arcsin(\bar{U}_{0j})-\arcsin(\sqrt{\theta}\bar{U}_{(m+i)j}))]\} \\ &\geq \frac{1}{4}\sum_{i\in A,\ j\in B} w_{ij}[\alpha(1+\bar{U}_{0i}+\bar{U}_{0j}+\bar{U}_{ij})+\alpha'(1+\bar{U}_{0(m+i)}-\bar{U}_{0j}-\bar{U}_{(m+i)j})] \\ &= \frac{\alpha}{4}\sum_{i\in A,\ j\in B} w_{ij}(2+\bar{U}_{0i}+\bar{U}_{0(m+i)}+\bar{U}_{ij}-\bar{U}_{(m+i)j}) \\ &= \alpha\cdot w_{SDP}. \end{aligned}$$

The proof of (7) and (8) is similar to Lemma 1 of [9]. □

We have numerically computed Table 1 for a list of σ (or k) and corresponding choices of θ, η and γ, where the last column R is $\min_{i=1,2}\{R_i(\eta,\gamma,\theta)\}$. We see that the approximation ratio $R > \max\{1/2, \sigma\}$ for $\sigma \in [0.37, 0.67]$ and $R > 0.63$ for $\sigma \geq 0.50$. (Recall that $\sigma = k/n$.)

4 Using the Feige-Goemans rotation

As we mentioned in Introduction, our rotation has a negative effect on the expected objective value but a positive effect on reducing the variance of $\hat{A}_1$ and $\hat{A}_2$. In contrast to our rotation, the mapping technique of Feig and Goemans [3] has a positive effect on the expected objective value but a negative effect on the variance of $\hat{A}_1$ and $\hat{A}_2$. By combining these two rotations, we can further improve the approximation ratios for 2-CSP. The details about the combination can be found in Appendix 2.

We have also numerically calculated Table 2 for a list of σ (or k) and corresponding choices of ρ_1, ρ_2, θ, η and γ, where the last column R is the approximation ratio. We see improvements:

$R(\eta, \gamma, \rho_1, \rho_2, \theta) > \max\{1/2, \sigma\}$ for $\sigma \in [0.33, 0.71]$ and $R(\eta, \gamma, \rho_1, \rho_2, \theta) > 0.63$ for $\sigma \geq 0.46$.

σ	θ	α	β	η	γ	R
0.37	0.83	0.7631	0.9228	-0.3778	2.8360	0.5097
0.38	0.82	0.7609	0.9320	-0.3527	2.9609	0.5211
0.40	0.80	0.7563	0.9481	-0.3015	3.2769	0.5436
0.42	0.77	0.7492	0.9624	-0.2497	3.7116	0.5655
0.44	0.76	0.7468	0.9717	-0.2057	4.0976	0.5869
0.46	0.74	0.7418	0.9794	-0.1611	4.6344	0.6073
0.48	0.72	0.7366	0.9845	-0.1196	5.2417	0.6262
0.50	0.72	0.7366	0.9856	-0.0867	5.5362	0.6430
0.52	0.72	0.7366	0.9845	-0.0580	5.6337	0.6569
0.54	0.73	0.7392	0.9802	-0.0350	5.3572	0.6677
0.56	0.75	0.7443	0.9727	-0.0023	4.9318	0.6756
0.58	0.79	0.7540	0.9601	-0.0000	3.6678	0.6808
0.60	0.82	0.7609	0.9452	0.0000	2.8456	0.6829
0.62	0.85	0.7674	0.9267	0.0000	2.2961	0.6833
0.64	0.86	0.7695	0.9088	0.0000	1.9047	0.6827
0.66	0.87	0.7716	0.8880	0.0039	1.6326	0.6813
0.67	0.88	0.7737	0.8751	0.0095	1.5368	0.6805

Table 1: Performance guarantees of the randomized algorithm for $0.37n \leq k \leq 0.67n$.

σ	ρ_1	ρ_2	θ	α	β	η	γ	R
0.33	0.22	0.51	0.83	0.8108	0.8777	-0.4919	2.8329	0.5086
0.34	0.21	0.54	0.82	0.8088	0.8898	-0.4611	2.9648	0.5194
0.36	0.18	0.54	0.82	0.8074	0.9093	-0.4257	2.9544	0.5403
0.38	0.18	0.55	0.78	0.7991	0.9299	-0.3656	3.3080	0.5617
0.40	0.17	0.55	0.77	0.7965	0.9442	-0.3208	3.5227	0.5830
0.42	0.15	0.57	0.75	0.7917	0.9578	-0.2721	3.8477	0.6036
0.44	0.15	0.57	0.74	0.7895	0.9665	-0.2326	4.0956	0.6239
0.46	0.15	0.57	0.74	0.7895	0.9720	-0.1977	4.2650	0.6428
0.48	0.15	0.58	0.74	0.7896	0.9749	-0.1664	4.3397	0.6592
0.50	0.15	0.58	0.74	0.7896	0.9757	-0.1374	4.3492	0.6736
0.52	0.15	0.58	0.74	0.7896	0.9749	-0.1095	4.3315	0.6869
0.54	0.15	0.58	0.74	0.7896	0.9720	-0.0689	4.3392	0.6980
0.56	0.15	0.58	0.74	0.7896	0.9665	-0.0392	3.9585	0.7056
0.58	0.15	0.58	0.74	0.7896	0.9587	-0.0086	3.6338	0.7107
0.60	0.15	0.58	0.74	0.7896	0.9487	-0.0000	2.9720	0.7133
0.62	0.16	0.59	0.76	0.7942	0.9336	0.0008	2.4089	0.7145
0.64	0.17	0.60	0.78	0.7985	0.9157	0.0180	2.1316	0.7147
0.66	0.18	0.61	0.82	0.8061	0.8916	0.0331	1.8885	0.7152
0.68	0.18	0.61	0.86	0.8135	0.8632	0.0449	1.6682	0.7150
0.70	0.18	0.61	0.86	0.8135	0.8400	0.0667	1.5307	0.7142
0.71	0.18	0.61	0.88	0.8170	0.8219	0.0728	1.4476	0.7136

Table 2: Performance guarantees of the randomized algorithm combined with Feige-Goemans rotation for $0.33n \leq k \leq 0.71n$.

5 Final remarks

During the writing of this paper, the authors have observed application of the enhanced randomized rounding technique to several other graph partitioning problems considered in Feige & Langber [5], Ye & Zhang [12], and Halperin & Zwick [8]. Preliminary results show that our technique is helpful in improving the approximation ratios of these problems as well.

To get the approximation ratios for 2-CSP, we need to solve several "global" minimization problems which are either 1-dimensional or 3-dimensional in a bounded set. For solving a 1-dimensional minimization problem, a lower bound of the minimum can be found by compare the endpoints of the bounded interval and those (at most two) stationary points. For solving a 3-dimensional problem in Section 4, we can reduce them to few 1-dimensional problems, as Goemans and Williamson did in [7]; see also Han, Ye & Zhang [9]. However, for the minimization problems analyzed in Section 5, we do not have a more rigorous proof. The lower bounds of the minimum are numerically computed by the aid of Matlab, as it has been done by Feige and Goemans [3].

References

[1] V. Asodi and S. Safra. On the complexity of the catalog-segmentation problem. unpublished manuscript, 2001.

[2] Y. Doids, V. Guruswami, and S. Khanna. The 2-catalog segmentation problem. In *Proceedings of SODA*, pages 378–380, 1999.

[3] U. Feige and M. X. Goemans. Approximating the value of two prover proof systems, with applications to max 2sat and max dicut. In *Proceedings of the 3nd Israel Symposium on Theory and Computing Systems*, pages 182–189, 1995.

[4] U. Feige and M. Langberg. The rpr^2 rounding technique for semidefinte programs. In *ICALP*, Lecture Notes in Computer Science. Springer, Berlin, 2001.

[5] U. Feige and M. Langberg. Approximation algorithms for maximization problems arising in graph partitioning. *Journal of Algorithms*, 2002.

[6] A. Frieze and M. Jerrum. Improved approximation algorithms for max k-cut and max bisection. *Algorithmica*, 18:67–81, 1997.

[7] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.

[8] E. Halperin and U. Zwick. A unified framework for obtaining improved approximation algorithms for maximum graph bisection problems. In *IPCO*, Lecture Notes in Computer Science. Springer, Berlin, 2001.

[9] Q. Han, Y. Ye, and J. Zhang. An improved rounding method and semidefinite programming relaxation for graph partition. *Math. Programming.*

[10] J. Kleinberg, C. Papadimitriou, and P. Raghavan. Segmentation problems. In *Proceedings of the 30th Symposium on Theory of Computation*, pages 473–482, 1998.

[11] Y. Ye. A .699-approximation algorithm for max-bisection. *Mathematical Programming*, 90:101–111, 2001.

[12] Y. Ye and J. Zhang. Approximation for dense-n/2-subgraph and the complement of min-bisection. *Journal of Global Optimization.*

[13] U. Zwick. Outward rotationvs: a tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems. In *Proceedings of the 31th Symposium on Theory of Computation*, pages 679–687, 1999.

6 Appendix 1. The proof of Theorem 1

Suppose that $w(\hat{A}_1, B_1) = \lambda_1 w_{SDP}$ and $w(\hat{A}_2, B_2) = \lambda_2 w_{SDP}$. Furthermore, suppose that $|\hat{A}_1| = \delta_1 n$ and $|\hat{A}_2| = \delta_2 n$. By the assumption, we have that

$$\lambda_1 + \lambda_2 + \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) + \eta\gamma(\delta_1 + \delta_2) \geq \mu.$$

Now we prove the lemma by considering four cases.

Case 1. $\delta_1 \geq \sigma$ and $\delta_2 \geq \sigma$. Without losing of generality, we assume that $\delta_1 \geq \delta_2$. It is easy to see that

$$\delta_2(1-\delta_2) + \eta\delta_2 \leq h_1(\eta),$$

where

$$h_1(\eta) = \begin{cases} \sigma(1-\sigma+\eta), & \text{if } \eta \leq 2\sigma - 1; \\ (\frac{1+\eta}{2})^2, & \text{otherwise.} \end{cases}$$

Then

$$\begin{aligned}
&\frac{w(A_1,B_1) + w(A_2,B_2)}{w_{SDP}} \\
\geq &\frac{\frac{k}{|\hat{A}_1|}\cdot w(\hat{A}_1,B_1) + \frac{k}{|\hat{A}_2|}\cdot w(\hat{A}_2,B_2)}{w_{SDP}} \\
= &\sigma(\frac{\lambda_1}{\delta_1} + \frac{\lambda_2}{\delta_2}) \\
\geq &\sigma[\frac{\mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2) - \lambda_2}{\delta_1} + \frac{\lambda_2}{\delta_2}] \\
\geq &\sigma[\frac{\mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2)}{\delta_1}] \\
\geq &\sigma[\frac{\mu - \gamma\delta_1(1-\delta_1) - \eta\gamma\delta_1 - h_1(\eta)\gamma}{\delta_1}] \\
\geq &\sigma[\frac{\mu - h_1(\eta)\gamma}{\delta_1} + \gamma\delta_1 - \gamma - \eta\gamma] \\
\geq &\sigma[2\sqrt{\gamma(\mu - h_1(\eta)\gamma)} - \gamma - \eta\gamma] \\
\geq &R_1(\eta,\gamma,\theta).
\end{aligned}$$

Case 2. $\delta_1 \leq \sigma$ and $\delta_2 \geq \sigma$. It is easy to see that

$$\delta_1(1-\delta_1) + \eta\delta_1 \leq h_2(\eta),$$

where

$$h_2(\eta) = \begin{cases} \sigma(1-\sigma+\eta), & \text{if } \eta \geq 2\sigma - 1; \\ (\frac{1+\eta}{2})^2, & \text{otherwise.} \end{cases}$$

Then

$$\frac{w(A_1,B_1) + w(A_2,B_2)}{w_{SDP}}$$

$$
\begin{aligned}
&\geq \frac{w(\hat{A}_1, B_1) + \frac{k}{|A_2|} \cdot w(\hat{A}_2, B_2)}{w_{SDP}} \\
&= \lambda_1 + \sigma \frac{\lambda_2}{\delta_2} \\
&\geq \lambda_1 + \sigma \frac{\mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2) - \lambda_1}{\delta_2} \\
&\geq (1 - \frac{\sigma}{\delta_2})\lambda_1 + \sigma \frac{\mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2)}{\delta_2} \\
&\geq \sigma \frac{\mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2)}{\delta_2} \\
&\geq \sigma \frac{\mu - \gamma\delta_2(1-\delta_2) - \eta\gamma\delta_2 - h_2(\eta)\gamma}{\delta_2} \\
&\geq \sigma[\frac{\mu - h_2(\eta)\gamma}{\delta_2} + \gamma\delta_2 - \gamma - \eta\gamma] \\
&\geq \sigma[2\sqrt{\gamma(\mu - h_2(\eta)\gamma)} - \gamma - \eta\gamma] \\
&\geq R_1(\eta, \gamma, \theta).
\end{aligned}
$$

Case 3. $\delta_1 \geq \sigma$ and $\delta_2 \leq \sigma$. This case can be easily reduced to Case 2.
Case 4. $\delta_1 \leq \sigma$ and $\delta_2 \leq \sigma$. Again, we have

$$\delta_1(1-\delta_1) + \eta\delta_1 \leq h_2(\eta)$$

and

$$\delta_2(1-\delta_2) + \eta\delta_2 \leq h_2(\eta).$$

Then

$$
\begin{aligned}
&\frac{w(A_1, B_1) + w(A_2, B_2)}{w_{SDP}} \\
&\geq \frac{w(\hat{A}_1, B_1) + w(\hat{A}_2, B_2)}{w_{SDP}} \\
&= \lambda_1 + \lambda_2 \\
&\geq \mu - \gamma(\delta_1(1-\delta_1) + \delta_2(1-\delta_2)) - \eta\gamma(\delta_1+\delta_2) \\
&\geq \mu - 2\gamma h_2(\eta) \\
&= R_2(\eta, \gamma, \theta).
\end{aligned}
$$

This completes the proof.

7 Appendix 2. Using the mapping technique

Here, we explain an improve approximation for 2-CSP using a tightened relaxation and mapping technique of [6].

First, we add the following so-called triangle inequalities to (2) (cf. [7] or [8]),

$$\begin{array}{lcll} U_{i_1i_2}+U_{i_1i_3}+U_{i_2i_3} & \geq & -1, & 0\leq i_1,i_2,i_3\leq 2n+m, \\ -U_{i_1i_2}-U_{i_1i_3}+U_{i_2i_3} & \geq & -1, & 0\leq i_1,i_2,i_3\leq 2n+m, \\ -U_{i_1i_2}+U_{i_1i_3}-U_{i_2i_3} & \geq & -1, & 0\leq i_1,i_2,i_3\leq 2n+m, \\ U_{i_1i_2}-U_{i_1i_3}-U_{i_2i_3} & \geq & -1, & 0\leq i_1,i_2,i_3\leq 2n+m. \end{array} \tag{9}$$

Secondly, define the mapping function $f(x;\rho)$ (see [6]):

$$f(x;\rho) := (1-\rho)x+\rho[\frac{\pi}{2}(1-\cos x)],$$

where $\rho\in[0,1]$ is a parameter.

Then we solve (2), with constraints (9)), to obtain an optimal semidefinite matrix $\bar{U}$. Next, we run the following mapping procedure.

Mapping: Given $0\leq\rho_1\leq\rho_2\leq 1$, we transform the semidefinite matrix $\bar{U}$ to a new semidefinite matrix $\bar{U}'$ of the following form:

$$\bar{U}'_{ij} = \begin{cases} \bar{U}_{00} & \text{if } i=j=0, \\ \cos f(\arccos\bar{U}_{0i};\rho_1) & \text{if } i=1,\cdots,2n, \\ \cos f(\arccos\bar{U}_{0i};\rho_2) & \text{if } i=2n+1,\cdots,2n+m, \\ g_1(\arccos\bar{U}_{0i},\arccos\bar{U}_{0j},\arccos\bar{U}_{ij}) & \text{if } i,j=1,\cdots,2n, \\ g_2(\arccos\bar{U}_{0i},\arccos\bar{U}_{0j},\arccos\bar{U}_{ij}) & \text{if } i=1,\cdots,2n, j=2n+1,\cdots,2n+m, \end{cases}$$

where

$$g_1(x,y,z)=\cos f(x;\rho_1)\cos f(y;\rho_1)+\frac{\cos z-\cos x\cos y}{\sin x\sin y}\sin f(x;\rho_1)\sin f(y;\rho_1)$$

and

$$g_2(x,y,z)=\cos f(x;\rho_1)\cos f(y;\rho_2)+\frac{\cos z-\cos x\cos y}{\sin x\sin y}\sin f(x;\rho_1)\sin f(y;\rho_2).$$

Using $\bar{U}'$ in replacing $\bar{U}$, we continue the rest procedures of Section 4.

The mapping function f has the property $f(\pi-x)=\pi-f(x)$ which guarantees that the nodes of $A_i(i=1,2)$ (or B_1) are treated symmetrically as the nodes of $A\setminus A_i(i=1,2)$ (or $B\setminus B_1$). The original rounding scheme of [7] corresponds to case $f(x;0)=x$. The motivation of using the mapping function is to get a better quality of α while making a small sacrifice on the quality of β.

The remaining analyses will be the same as those in Section 4 with modified definitions of α, τs and βs. For given $\rho_1,\rho_2,\theta\in[0,1]$, let

$$\alpha=\min_{(x,y,z)\in\mathcal{F}}\frac{\tilde{f}(x,y,z)}{1+\cos x+\cos y+\cos z},$$

where

$$\begin{aligned}\tilde{f}(x,y,z) &= 1+\frac{2}{\pi}[\arcsin(\sqrt{\theta}\cos f(x;\rho_1)) \\ &\quad +\arcsin(\cos f(y;\rho_2))+\arcsin(\sqrt{\theta}g_2(x,y,z))]\end{aligned}$$

and

$$\mathcal{F} := \left\{ (x,y,z) \;\middle|\; \begin{array}{l} \begin{pmatrix} 1 & \cos x & \cos y \\ \cos x & 1 & \cos z \\ \cos y & \cos z & 1 \end{pmatrix} \succeq 0, \\ \cos x + \cos y + \cos z \geq -1, \\ -\cos x - \cos y + \cos z \geq -1, \\ -\cos x + \cos y - \cos z \geq -1, \\ \cos x - \cos y - \cos z \geq -1, \\ 0 \leq x, y, z \leq \pi. \end{array} \right\};$$

$$\tau^{+} = \tau^{-} = \min_{0 \leq x \leq \pi} \frac{2}{\pi} \frac{\arcsin(\sqrt{\theta}) - \arcsin(\sqrt{\theta} \cos f(x; \rho_1))}{1 - \cos x};$$

$$\beta^{+} = 1 - \frac{2}{\pi} \arcsin(\sqrt{\theta}) + 2\sigma\tau^{+};$$

$$\beta^{-} = 1 - \frac{2}{\pi} \arcsin(\sqrt{\theta}) + 2(1-\sigma)\tau^{-};$$

$$\tau = \min_{(x,y,z) \in \mathcal{F}} \frac{2}{\pi} \frac{\arcsin(\theta + (1-\theta)\kappa^2) - \arcsin(\theta g_1(x,y,z) + (1-\theta)\kappa^2)}{1 - \cos z};$$

$$\beta = 1 - \frac{2}{\pi} \arcsin(\theta + (1-\theta)\kappa^2) + 4\sigma(1-\sigma)\tau.$$

Note that Lemmas 5 and 4 still hold for applying the mapping technique.