SCIENCE CHINA
Mathematics

Progress of Projects Supported by NSFC doi: 10.1007/s11425-014-4900-5

• **ARTICLES** •

An SDP randomized approximation algorithm for max hypergraph cut with limited unbalance

XU BaoGang[1], YU XingXing[2], ZHANG XiaoYan[1,3,*] & ZHANG Zan-Bo[4]

[1]*School of Mathematical Science & Institute of Mathematics, Nanjing Normal University, Nanjing* 210023*, China;*
[2]*School of Mathematics, Georgia Institute of Technology, Atlanta, GA* 30332*, USA;*
[3]*Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, Enschede* 7500*, The Netherlands;*
[4]*Department of Computer Engineering, Guangdong Industry Technical College, Guangzhou* 510300*, China*
Email: baogxu@njnu.edu.cn, yu@math.gatech.edu, royxyzhang@gmail.com, eltonzhang2001@gmail.com

Received February 19, 2014; accepted July 28, 2014; published online September 19, 2014

Abstract We consider the design of semidefinite programming (SDP) based approximation algorithm for the problem *Max Hypergraph Cut with Limited Unbalance* (MHC-LU): Find a partition of the vertices of a weighted hypergraph $H=(V,E)$ into two subsets V_1, V_2 with $||V_2|-|V_1||\leqslant u$ for some given u and maximizing the total weight of the edges meeting both V_1 and V_2. The problem MHC-LU generalizes several other combinatorial optimization problems including Max Cut, Max Cut with Limited Unbalance (MC-LU), Max Set Splitting, Max Ek-Set Splitting and Max Hypergraph Bisection. By generalizing several earlier ideas, we present an SDP randomized approximation algorithm for MHC-LU with guaranteed worst-case performance ratios for various unbalance parameters $\tau=u/|V|$. We also give the worst-case performance ratio of the SDP-algorithm for approximating MHC-LU regardless of the value of τ. Our strengthened SDP relaxation and rounding method improve a result of Ageev and Sviridenko (2000) on Max Hypergraph Bisection (MHC-LU with $u=0$), and results of Andersson and Engebretsen (1999), Gaur and Krishnamurti (2001) and Zhang et al. (2004) on Max Set Splitting (MHC-LU with $u=|V|$). Furthermore, our new formula for the performance ratio by a tighter analysis compared with that in Galbiati and Maffioli (2007) is responsible for the improvement of a result of Galbiati and Maffioli (2007) on MC-LU for some range of τ.

Keywords max hypergraph cut with limited unbalance, approximation algorithm, performance ratio, semidefinite programming relaxation

MSC(2010) 49M25, 90C22, 90C27

Citation: Xu B G, Yu X X, Zhang X Y, et al. An SDP randomized approximation algorithm for max hypergraph cut with limited unbalance. Sci China Math, 2014, 57, doi: 10.1007/s11425-014-4900-5

1 Introduction

A *hypergraph* is an ordered pair $H=(V,E)$ in which $V:=\{1,2,\ldots,m\}$ is a finite nonempty set and $E:=\{S_1,S_2,\ldots,S_n\}$ is a collection of distinct nonempty subsets of V. V and E are the sets of *vertices* and *edges* of H, respectively. A weighted hypergraph is a hypergraph together with a nonnegative real function $\omega: E\to\mathbb{R}^+$. For convenience, we write $\omega_j:=\omega(S_j)$. Given a partition $V=V_1\cup V_2$, the edge S_j is said to be a *cut edge* with respect to this partition if $S_j\cap V_i\neq\emptyset$ for $i=1,2$. The Max Hypergraph Cut with Limited Unbalance problem (MHC-LU) asks for a partition $V=V_1\cup V_2$ such that $||V_2|-|V_1||\leqslant u$ for some given $u\geqslant 0$ and the total weight of cut edges is maximized. Note that MHC-LU with $u=0$

*Corresponding author

is also called Max Hypergraph Bisection, and MHC-LU with $u = m$ is also called Max Set Splitting or Max Hypergraph Cut.

Hypergraphs partition arises naturally in important practical problems, including circuit design and network planning, etc., [25,29,36]. For most of them, the unbalanced constraints make sense. For example from the point of view of the circuit designer, the suitability of partition of circuit is not affected if one relaxes the bisection constraint to limited unbalance and get better results as far as approximating the optimum weight of the cut is concerned [17]. Since the partitioning of hypergraphs is critical in several practical applications, many heuristic algorithms were developed [10, 31]. In this paper, we present a polynomial time SDP randomized approximation algorithm for MHC-LU with guaranteed performance ratio.

When the hypergraph is 2-uniform (a standard graph), MHC-LU is known as the Maximum Cut with Limited Unbalance problem (MC-LU). Galbiati and Maffioli [17] first gave polynomial time randomized approximation algorithms with nontrivial performance guarantees for MC-LU.

The well-known *Max Cut* problem is simply MHC-LU with $u = m$ and $|S_j| = 2$ for all j. Goemans and Williamson [19], in a major breakthrough, used semidefinite programming relaxation and hyperplane rounding to obtain an approximation algorithm for the Max Cut problem with expected performance guarantee 0.87856. This well-known algorithmic paradigm, with more sophisticated techniques, has been applied to many previously studied problems [14–17, 21, 22, 41, 42].

When $u = 0$ and $|S_j| = 2$ for all j, MHC-LU is known as the *Max Bisection* problem. Frieze and Jerrum [16] and Andersson [4] presented a 0.65-approximation algorithm for the Max Bisection problem, and Ye [41] obtained a 0.699-approximation algorithm for the problem. Halperin and Zwick [21] improved the performance ratio for the Max Bisection problem to 0.701, which was further improved by Feige and Langberg [15] to 0.7028 using the RPR^2 rounding technique. For the case of regular graphs, Feige et al. improved the approximation ratio to 0.795 (0.834 for 3-regular graphs) [12, 13]. Recently, Raghavendra and Tan [35] significantly improved all the above results to 0.85 by using the Lasserre Hierarchy. This algorithm was further improved by Austrin et al. [7] to 0.8776 by also using a relaxation based on the Lasserre Hierarchy.

When $u = 1$, MHC-LU asks for *balanced* bipartitions, i.e., partitions $V = V_1 \cup V_2$ such that $||V_1| - |V_2|| \leqslant 1$. Bollobás and Scott [8] conjectured that if G is a graph with minimum degree at least 2, then $V(G)$ admits a balanced bipartition V_1, V_2 such that for each $i \in \{1, 2\}$ at most $|E(G)|/3$ edges have both ends in V_i. The minimum degree condition is necessary. Bollobás and Scott [9] established this conjecture for regular graphs. Xu et al. [38, 39] proved this conjecture for graphs G with $\Delta(G) \leqslant \frac{7}{5}\delta(G)$ or with $\delta(G) \geqslant 5$, where $\Delta(G)$ and $\delta(G)$ are maximum and minimum degrees of G, respectively. Lee et al. [30] proved a nice asymptotic result claiming that every graph with m edges and minimum degree $2k$ or $2k+1$ admits a balanced bipartition V_1, V_2 such that

$$\max\{e(V_1), e(V_2)\} \leqslant \left(\frac{k+1}{2(2k+1)} + o(1)\right)m$$

(when $k = 1$, its main item is $\frac{m}{3}$). The conjecture is confirmed recently by Xu and Yu [40]. They proved that every graph G with m edges and minimum degree at least 2 admits a balanced bipartition V_1, V_2 with $\max\{e(V_1), e(V_2)\} < m/3$ unless G is a triangle.

For $u = m$, MHC-LU becomes the so called Max Set Splitting problem. Andersson and Engebretsen [5] presented a 0.72-approximation algorithm for this problem, and the approximation ratio was improved to 0.7499 by Zhang et al. [42]. Gaur and Krishnamurti [18] gave a $k/(k+1)$-approximation algorithm for the problem, where $k \geqslant 3$ is the minimum number of vertices in a hyperedge. Arora et al. [6] designed a PTAS for dense instances of this problem. When restricted to k-uniform hypergraphs the Max Set Splitting problem is known as the *Max Ek-set splitting* problem. For any fixed $k \geqslant 2$, Lovász [32] and Petrank [34] have shown that Max Ek-set splitting problem is NP-hard and APX-complete, respectively. When $k = 2$, Max Ek-set splitting problem is exactly the Max Cut problem. When $k = 3$, the performance guarantee has been improved by Zwick [44] to 0.90871. (More precisely, Zwick [43, 44] obtained a 0.90871-approximation algorithm for MAX NAE-$\{3\}$-SAT, which is the restriction of MAX NAE SAT to instances in which all clauses are of size at most 3. MAX NAE SAT is a variant of the well known MAX SAT. The objective of MAX NAE SAT is to maximize the clauses which contain both true and false literals. Obviously, Max E3-set splitting is a special case of MAX NAE-{3}-SAT in which all literals appear unnegated and thus can also be approximated with the 0.90871 performance guarantee.) When $k \geqslant 4$, Alimonti [3] and Kann et al. [28] showed that Max Ek-set splitting problem can be approximated within $1 - 2^{1-k}$ which is best possible [20, 24].

Ageev and Sviridenko [1, 2] considered MHC-LU for $u = m - 2k$ with more strict condition that $||V_2| - |V_1|| = u$ (hence $|V_i| = k$ and $|V_{3-i}| = m - k$ for some $i \in \{1, 2\}$), and they gave a 0.5-approximation algorithm based on linear programming. For graphs, Hassin and Rubinstein [23] presented a different 0.5-approximation with a better running time. Feige and Langberg [14] combined the method in [1] with the semidefinite programming approach to design a $(0.5 + \epsilon)$-approximation algorithm, where ϵ is some unspecified small positive number. Han et al. [22] and Jäger and Srivastav [27] applied semidefinite programming to obtain better approximation factors than previously known.

Recently, some new results and techniques based on semidefinite programming relaxations are obtained, for example, see [7, 26, 35, 37]. Especially, the technique of the Lasserre Hierarchy can improve the approximation ratios of many graph partitioning problems significantly from the theoretical point of view. Although the new algorithms based on the Lasserre Hierarchy in [35] and [7] are very powerful, the running time of the algorithm is probably n^k in [35], where $10 < k < 100$, and is $O(n^{10^{100}})$ in [7] by loose estimation, respectively. Some classical methods like hyperplane rounding or outward rotations, still have their own merits of fast running and easy implemention, especially for the problems of hypergraph. In this paper, we apply semidefinite programming to the more general MHC-LU building on several classical ideas of Goemans and Williamson [19] and Halperin and Zwick [21] (for Max Cut), Galbiati and Maffioli [17] (for MC-LU), Ye [41] (for Max Bisection), Andersson and Engebretsen [5] and Zhang et al. [42] (for Max Set Splitting). By solving the semidefinite programming relaxation of MHC-LU, we obtain an (almost) optimal vector solution $(v_0^*, v_1^*, \ldots, v_m^*)$. In previous work, these vectors are usually rounded by applying an important technique named outward rotations [44] which combines the classical hyperplane rounding method [19] with independent random choice to partition the coordinates into two parts. Motivated by Halperin and Zwick [21] (for maximum graph bisection problems), we also apply the idea of outward rotations to a linear randomized rounding method [21] and obtain better performance ratios for MHC-LU when the minimum number of vertices in a hyperedge is 3 by combining the outward rotations of random hyperplane rounding procedure with that of linear randomized rounding procedure.

Moreover, we present a generalized formula for the performance ratio for MHC-LU using some additional parameters. However, it remains open to find a best set of the parameters which optimizes the generalized formula. In practice, the final ratios need to be obtained via a computer search over the parameter space. For fast computations and easy verifications of computational results, we may assign simple values to some of the parameters and simply perform 1-dimensional searches on the remaining parameters. This turns out to be sufficient to improve the following numerical results, where $\tau = u/m$ is known as the *unbalance parameters*:

- For $\tau = 0$, we obtain approximation ratio 0.6271, improving the 0.5-approximation ratio of Ageev and Sviridenko [2] for Max Hypergraph Bisection which was based on linear programming (see Table 1 for $\tau = 0$). The improvement is a consequence of our strengthened SDP relaxation based approximation algorithm.
- For $\tau = 1$, i.e., the version of Max Set Splitting problem, our strengthened SDP relaxation and a generalized formula of performance ratio (see Lemma 3.4) give approximation ratio 0.7524 which improves the ratio 0.7499 in [42] (see Table 1 for $\tau = 1$).

We report the lower bounds on the approximation ratios for MHC-LU, for a range of $0 < \tau < 1$, in Table 1. The corresponding approximation values obtained previously are shown in the last row of Table 1. Note that the "none", in the last row of Table 1, represents no known previous result (as far as we know). (All values reported in tables in this paper are truncated at the fourth decimal places.)

- We obtain approximation ratio 0.7741 for MHC-LU when the minimum number of vertices in a hyperedge is 3 which improves the result 0.75 in Gaur and Krishnamurti [18] (see Table 2 for $\tau = 1$).

Table 1 The new results of MHC-LU compared with the previous results for some τ

τ	0	0.2500	0.5000	0.7500	0.9000	0.9990	1.0000
the new results	0.6271	0.7105	0.7130	0.7194	0.7353	0.7522	0.7524
the previous results	0.5000	none	none	none	none	none	0.7499

Table 2 The new results of MHC-LU compared with the previous results for some τ when the minimum number of vertices in a hyperedge is 3

τ	0	0.2500	0.5000	0.7500	0.9000	0.9990	1.0000
the new results	0.7042	0.7459	0.7495	0.7564	0.7656	0.7740	0.7741
the previous results	none	none	none	none	none	none	0.7500

Table 3 The new results of MC-LU compared with the previous results in [17] for some $0.5 < \tau < 1$

τ	0.6000	0.7000	0.8000	0.8500	0.9000	0.9500	0.9999
the new results	0.7987	0.8052	0.8191	0.8291	0.8417	0.8584	0.8785
the previous results	0.7950	0.7930	0.7900	0.8126	0.8340	0.8560	0.8780

This improvement is also due to the strengthened SDP relaxation and an improved rounding method by combining the outward rotations of random hyperplane rounding procedure with that of linear randomized rounding procedure. The lower bounds on the approximation ratios for MHC-LU when the minimum number of vertices in a hyperedge is 3 are reported in Table 2, for a range of $0 \leqslant \tau \leqslant 1$.

• We show that one can further improve the performance ratios as those in [17] for Max Cut with Limited Unbalance when $0.5 < \tau < 1$, using a new formula of performance ratio (see Lemma 6.1 with $|S_j| = 2$ for all j) by tighter analysis compared with that in [17]. In Table 3, we report the lower bounds on the approximation ratios for Max Cut with Limited Unbalance, for the same values of τ as in [17] when $0.5 < \tau < 1$. The corresponding approximation ratios in [17] are shown in the last row of Table 3. We also give a detailed explanation for the reason of improvements after Lemma 6.1 as the argument needs part of the proof of the Lemma in Section 6.

At the end of Section 6, we present the first worst-case performance ratio 0.6271 of the SDP-algorithm for approximating MHC-LU regardless of the value of τ.

This paper is organized as follows. In Section 2, we present a strengthened semidefinite programming based approximation algorithm for MHC-LU. In Sections 3 and 4, we obtain the bounds on the expected contribution of an edge before and after executing Step 5 which establish the performance guarantee of our SDP algorithm for MHC-LU when $\tau = 1$. In Section 5, we give the other expectation bound after executing Step 5 which is used to obtain the performance ratio of the algorithm for MHC-LU when $0 \leqslant \tau < 1$. In Section 6, we analyze the performance ratio of the SDP approximation algorithm and propose interesting questions for further research. In Appendix, we present the proof of a lemma.

2 An SDP relaxation of MHC-LU

We first formulate MHC-LU as an integer program. Let $H = (V, E)$ be a hypergraph with $V = \{1, \ldots, m\}$ and $E = \{S_1, \ldots, S_n\}$, and let ω be a nonnegative real function on E with $\omega_j := \omega(S_j)$.

Let $x_0 \in \{-1, 1\}$ be a reference variable. Let (V_1, V_2) be a cut of H such that $||V_1| - |V_2|| \leqslant u$. For each vertex i, let $x_i = x_0$ if $i \in V_1$ and otherwise let $x_i = -x_0$. Then

$$\left|\sum_{i=1}^{m} x_0 x_i\right| = \left|\sum_{i=1}^{m} x_i\right| = ||V_1| - |V_2|| \leqslant u,$$

so

$$\left(\sum_{i=1}^{m} x_i\right)^2 \leqslant u^2 \quad \text{and} \quad -u \leqslant \sum_{i=1}^{m} x_0 x_i \leqslant u.$$

Also note that

$$\left(\sum_{i\in S_j} x_i\right)^2 = |S_j| + 2\sum_{i<k\in S_j} x_i x_k,$$

which implies

$$\sum_{i<k\in S_j} x_i x_k \geqslant -\lfloor |S_j|/2 \rfloor,$$

since $|\sum_{i\in S_j} x_i| \geqslant 1$ when $|S_j|$ is odd.

Let $z_j = 1$ if S_j is in the cut (V_1, V_2), and let $z_j = 0$ otherwise. Then the total weight of the cut is

$$\omega(V_1, V_2) = \sum_{j=1}^{n} \omega_j z_j.$$

Note that if $x_i = x_k$ for all $i, k \in S_j$, i.e., $z_j = 0$, then

$$\sum_{i<k\in S_j} (1 - x_i x_k)/2 = 0,$$

and if $x_i \neq x_k$ for some $i, k \in S_j$, i.e., $z_j \neq 0$, then

$$\sum_{i<k\in S_j} (1 - x_i x_k)/2 \geqslant t(|S_j| - t),$$

for some $1 \leqslant t \leqslant |S_j| - 1$ (so $t(|S_j| - t) \geqslant |S_j| - 1$); hence

$$\frac{1}{|S_j| - 1} \sum_{i<k\in S_j} \frac{1 - x_i x_k}{2} \geqslant z_j.$$

Thus MHC-LU can be formulated as the following quadratic integer program.

$$\begin{aligned}
\omega^{\text{opt}} &= \max \sum_{j=1}^{n} \omega_j z_j, \\
\text{s.t.} \quad & \frac{1}{|S_j| - 1} \sum_{i<k\in S_j} \frac{1 - x_i x_k}{2} \geqslant z_j \quad \text{for all} \quad S_j \in E, && (2.1)\\
& \sum_{1\leqslant i,k\leqslant m} x_i x_k \leqslant u^2, && (2.2)\\
& -u \leqslant \sum_{1\leqslant i\leqslant m} x_0 x_i \leqslant u, && (2.3)\\
& \sum_{i<k\in S_j} x_i x_k \geqslant -\lfloor |S_j|/2 \rfloor \quad \text{for all} \quad S_j \in E, && (2.4)\\
& x_i \in \{-1, 1\}, \quad \text{for} \quad i = 0, 1, 2, \ldots, m, \\
& z_j \in \{0, 1\}, \quad \text{for} \quad j = 1, \ldots, n.
\end{aligned}$$

As mentioned in the Section 1, Max Set Splitting is the restricted versions of MHC-LU. We note that constraint (2.1) is used in Andersson and Engebretsen [5] for Max Set Splitting. Constraint (2.2) is the limited unbalance constraint used in Galbiati and Maffioli [17] for MC-LU. Constraint (2.4) is used in Zhang et al. [42] to strengthen the SDP relaxation in [5] for Max Set Splitting. Note that constraint (2.3) is not used in [17]. Although constraint (2.3) is implied by constraint (2.2), its corresponding constraint (2.8) in our SDP relaxation below, is not implied by constraint (2.7) (corresponding to constraint

(2.2) above); and we also add further triangle inequalities (2.10) and (2.11) which were first used by Feige and Goemans [11] to improve the performance ratio of the approximation algorithms for Max 2SAT and Max DICUT. The triangle constraints (2.10) and (2.11) does not appear in the relaxation presented in [5] and [42] for Max Set Splitting. So with constraints (2.8), (2.10) and (2.11) the following SDP relaxation strengthens the SDP relaxations in [17], [5] and [42] for MC-LU and Max Set Splitting, respectively.

The SDP relaxation of the above quadratic integer program becomes:

$$\omega^{\rm SDP} := \max \sum_{i=1}^{n} \omega_j z_j, \tag{2.5}$$

$$\text{s.t.} \quad \frac{1}{|S_j|-1} \sum_{i<k\in S_j} \frac{1-X_{ik}}{2} \geqslant z_j \quad \text{for all} \quad S_j \in E, \tag{2.6}$$

$$\sum_{1\leqslant i,k\leqslant m} X_{ik} \leqslant u^2, \tag{2.7}$$

$$-u \leqslant \sum_{1\leqslant i\leqslant m} X_{0i} \leqslant u, \tag{2.8}$$

$$\sum_{i<k\in S_j} X_{ik} \geqslant -\lfloor |S_j|/2 \rfloor \quad \text{for all} \quad S_j \in E, \tag{2.9}$$

$$-X_{0i} - X_{0k} + X_{ik} \geqslant -1 \quad \text{for} \quad 1\leqslant i\leqslant k\leqslant m, \tag{2.10}$$

$$X_{ik} + X_{il} + X_{kl} \geqslant -1 \quad \text{for} \quad 0\leqslant i\leqslant k\leqslant l\leqslant m, \tag{2.11}$$

$$X_{ii} = 1 \quad \text{for} \quad i = 0,1,\ldots,m,$$

$$X \succeq 0,$$

$$0 \leqslant z_j \leqslant 1 \quad \text{for} \quad j = 1,2,\ldots,n,$$

where $X \succeq 0$ means that the $(n+1)\times(n+1)$ symmetric matrix X is positive semidefinite.

Throughout the paper, $\omega(V_1, V\backslash V_1)$ denotes the weight of the cut $(V_1, V\backslash V_1)$, and $\tau = u/m$ denotes the unbalance parameter. We now present an SDP-based approximation algorithm for MHC-LU as follows.

SDP-algorithm for MHC-LU.

Step 1. Solve problem (2.5) to obtain an almost optimal positive semidefinite matrix X^* and a vector z^*. Apply a Cholesky decomposition to X^*, $X^* = v\cdot v^{\rm T}$, to obtain column vectors $(v_0^*, v_1^*, \ldots, v_m^*)$ of v. For a given $0<\epsilon<1$, repeat the following steps $K = O((1/\epsilon)\log(1/\epsilon))$ times, and output the best set $\overline{V}_1$.

Step 2. Choose two rotation coefficients $0<\theta,\theta'\leqslant 1$, and a probability $0\leqslant\nu\leqslant 1$. Construct two sets of unit vectors $(v_0, v_1, \ldots, v_m)$ and $(v_0', v_1', \ldots, v_m')$ such that $v_i\cdot v_j = \theta v_i^*\cdot v_j^*$, and $v_i'\cdot v_j' = \theta' v_i^*\cdot v_j^*$ for every $0\leqslant i<j<m$. Goto Step 3 with probability ν, and independently goto Step 4 with probability $1-\nu$.

Step 3. Choose a random vector r and for $0\leqslant i\leqslant m$ let $\widehat{x}_i = 1$ if $v_i\cdot r\geqslant 0$, and $\widehat{x}_i = -1$ otherwise. Set $\widehat{V}_1 = \{i : \widehat{x}_i = \widehat{x}_0\}$. Goto Step 5.

Step 4. For each $i\in V$, put i in $\widehat{V}_1$, with probability $(1+v_0'\cdot v_i')/2$, and independently in $V\setminus\widehat{V}_1$ with probability $(1-v_0'\cdot v_i')/2$.

Step 5. Given $0\leqslant p\leqslant 1/2$, each element in $\widehat{V}_1$ or in $V\setminus\widehat{V}_1$ has a probability of p of being assigned to the opposite subset, independently. This gives a new partition $(V_1, V\setminus V_1)$ of V. Let

$$\widetilde{V}_1 = \begin{cases} V_1, & \text{if} \ \ |V_1|\geqslant m/2, \\ V\setminus V_1, & \text{otherwise.} \end{cases}$$

If $|\widetilde{V}_1|\leqslant (m+u)/2$, let $\overline{V}_1 = \widetilde{V}_1$, else goto Step 6.

Step 6. Let $\widetilde{V}_1 = \{1, 2, \ldots, T\}$. For each $1 \leqslant t \leqslant T$, define

$$B(t) := \{S_j : t \in S_j \cap \widetilde{V}_1 \neq \emptyset \text{ and } S_j \cap (V \setminus \widetilde{V}_1) \neq \emptyset\}.$$

Set

$$c(t) := \sum_{S_j \in B(t)} \frac{w_j}{|S_j \cap \widetilde{V}_1|}$$

as the contribution of vertex t to the weight of the cut $(\widetilde{V}_1, V \setminus \widetilde{V}_1)$. Assume without loss of generality that $c(1) \leqslant \cdots \leqslant c(T)$. Let $\overline{V}_1 = \widetilde{V}_1 \setminus \{1, 2, \ldots, T - (m+u)/2\}$.

Note that if we set $\theta = 1$ in Step 2, the technique of randomized rounding in Step 3 is the hyperplane rounding due to Goemans and Williamson [19]. The technique of outward rotations used in Step 3 is a way of combining the classical hyperplane rounding method with independent random choice which was used in [14, 33, 41, 44].

If we set $\theta' = 1$ in Step 2, the technique of randomized rounding in Step 4 is the linear randomized rounding due to Halperin and Zwick [21]. In Step 4, we combine the linear randomized rounding method with independent random choice by using a new rotation parameter θ'. It turns out that we achieve better performance ratios by combining the outward rotations of the random hyperplane rounding procedure with that of the linear randomized rounding procedure for MHC-LU when the minimum number of vertices in a hyperedge is 3. We point out that $\widehat{V}_1$ in Steps 3 and 4 is only introduced for the rounding analysis in Section 3.

The first part of Step 5 is a probabilistic postprocessing step, which perturbs the initial partition $(\widehat{V}_1, V \setminus \widehat{V}_1)$ obtained in Step 3 or Step 4 to construct a new partition $(V_1, V \setminus V_1)$ correspondingly.

Step 6 adjusts the size of the sets in a partition to satisfy the limited unbalance constraint. We have the following lemma.

Lemma 2.1. $\omega(\overline{V}_1, V \setminus \overline{V}_1) \geqslant \frac{m+u}{2T}\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$ *if* $T > \frac{m+u}{2}$, *and* $\omega(\overline{V}_1, V \setminus \overline{V}_1) = \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$ *otherwise.*

Proof. Step 6 reduces the number of vertices in $\widetilde{V}_1$ to $(m+u)/2$ by moving from $\widetilde{V}_1$ to $V \setminus \widetilde{V}_1$ the $T - (m+u)/2$ vertices with the smallest contributions to the cut. (Note that the condition $||V_2| - |V_1|| \leqslant u$ may be written as $\frac{m-u}{2} \leqslant |V_i| \leqslant \frac{m+u}{2}$ for $i = 1, 2$.) Observe that

$$\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1) = \sum_{t=1}^{T} c(t).$$

Moreover, the construction of $\overline{V}_1$ guarantees that

$$\frac{\omega(\overline{V}_1, V \setminus \overline{V}_1)}{(m+u)/2} \geqslant \frac{\sum_{t=T-(m+u)/2+1}^{T} c(t)}{(m+u)/2} \geqslant \frac{\sum_{t=1}^{T} c(t)}{T} = \frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{T},$$

where the first inequality follows from definition of $\overline{V_1}$, and the second inequality holds because the average over the largest numbers of a sequence is at least the average over the entire sequence. Therefore,

$$\omega(\overline{V}_1, V \setminus \overline{V}_1) \geqslant \frac{m+u}{2T}\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$$

if $T > \frac{m+u}{2}$, and

$$\omega(\overline{V}_1, V \setminus \overline{V}_1) = \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$$

otherwise. □

To analyze the quality of the SDP approximation algorithm, we need to establish a lower bound on $E[\omega(\overline{V_1}, V \setminus \overline{V_1})]/\omega^{\text{opt}}$. By Lemma 2.1, we then wish to establish a lower bound on

$$E\left[\frac{m+u}{2T} \cdot \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)\right] / \omega^{\text{opt}},$$

which is also a lower bound on $E[\omega(\overline{V_1}, V \setminus \overline{V_1})]/\omega^{\text{opt}}$. However, it is not easy to calculate the expected value of $\frac{m+u}{2T} \cdot \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$, since both $\frac{m+u}{2T}$ and $\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$ are random variables and they are multiplied together where $T = |\widetilde{V}_1|$. Instead, we study a family of random variables $Z(\gamma)$ whose expected values can be easily estimated and bounded, an idea first used by Frieze and Jerrum [16] for Max Bisection and extended in [14, 21, 22, 41]. For fixed values $\tau = u/m \in [0, 1]$ and $\gamma > 0$,

$$Z(\gamma) = \frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{\omega^{\text{opt}}} + \gamma \frac{|\widetilde{V}_1|(m - |\widetilde{V}_1|)}{m^2}.$$

Whenever $Z(\gamma)$ fulfills its expectation, (i.e., $Z(\gamma) \geqslant \alpha^{(p)} + \gamma\beta - o(1)$ (see Section 6)), we have

$$\frac{m+u}{2T} \cdot \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1) \geqslant \overline{R}(\tau) \cdot \omega^{\text{opt}},$$

which establishes a performance ratio $\overline{R}(\tau)$ of the SDP approximation algorithm in Section 6. Therefore, the aim of the following sections (except the last one) is to give bounds on

$$\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)] \quad \text{and} \quad \mathbb{E}[|\widetilde{V}_1|(m - |\widetilde{V}_1|].$$

These bounds will be used in the last Section to give bounds on $\mathbb{E}[Z(\gamma)]$ and then to obtain the performance ratio $\overline{R}(\tau)$.

3 Bound on the expected contribution of an edge by Steps 1–4

To obtain the bound on $\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)]$ in terms of the maximum value ω^{opt}, we need to establish the bound on the expected contribution of an edge by executing Steps 1–4. Note that we have an initial cut $(\widehat{V}_1, V \setminus \widehat{V}_1)$ of H by executing Steps 1–4. An edge S_j is in this cut if there exists $i, k \in S_j$ such that $i \in \widehat{V}_1$ and $k \in \widehat{V}_2 := V \setminus \widehat{V}_1$. The indicator random variables $\widehat{X}_{ik}$ and $\widehat{U}(S_j)$ are defined as

$$\widehat{X}_{ik} = \begin{cases} -1, & \text{if } i \in \widehat{V}_i \text{ and } k \in V \setminus \widehat{V}_i, \quad \text{for } i = 1, 2, \\ 1, & \text{otherwise.} \end{cases}$$

$$\widehat{U}(S_j) = \begin{cases} 1, & \text{if } S_j \in (\widehat{V}_1, V \setminus \widehat{V}_1), \\ 0, & \text{otherwise.} \end{cases}$$

From [19, 44], we have the following equality if the algorithm executes Step 3:

$$E[\widehat{X}_{ik}] = (2/\pi) \arcsin(\theta X^*_{ik}),$$

where X^*, z^* is an optimal solution to (2.5).

If the algorithm executes Step 4, it is easily seen that

$$\begin{aligned} E[\widehat{X}_{ik}] &= (-1) \cdot \left(\frac{1+\theta' X^*_{0i}}{2} \cdot \frac{1-\theta' X^*_{0k}}{2} + \frac{1-\theta' X^*_{0i}}{2} \cdot \frac{1+\theta' X^*_{0k}}{2} \right) \\ &\quad + (+1) \left(\frac{1+\theta' X^*_{0i}}{2} \cdot \frac{1+\theta' X^*_{0k}}{2} + \frac{1-\theta' X^*_{0i}}{2} \cdot \frac{1-\theta' X^*_{0k}}{2} \right) \\ &= \theta'^2 X^*_{0i} X^*_{0k}. \end{aligned}$$

Hence, from Steps 1–4, we have

$$E[\widehat{X}_{ik}] = \nu(2/\pi) \arcsin(\theta X^*_{ik}) + (1-\nu)\theta'^2 X^*_{0i} X^*_{0k}.$$

The main objective here is to establish a lower bound on the expected contribution of an individual edge to the cut before executing Step 5. More precisely, we show that for any fixed $s \geqslant 2$ and $s \in \mathbb{N}$ there exists a constant $\alpha_s > 0$ such that for each $S_j \in E$ with $|S_j| = s$ we have $\mathbb{E}[\widehat{U}(S_j)] \geqslant \alpha_s z^*_j$.

For convenience let $S := S_j$ be an edge of H belonging to $(\widehat{V}_1, V \setminus \widehat{V}_1)$. For the same reason as for constraint (2.4),

$$\sum_{i<k\in S} \widehat{X}_{ik} = \sum_{i<k\in S} \widehat{x}_i\widehat{x}_k \geqslant -\lfloor |S|/2 \rfloor.$$

So

$$\sum_{i<k\in S} \frac{1-\widehat{X}_{ik}}{2} \leqslant (1/2)\left[\binom{|S|}{2} + \lfloor |S|/2 \rfloor\right] \leqslant L := \begin{cases} |S|^2/4, & \text{if } |S| \text{ is even,} \\ (|S|+1)(|S|-1)/4, & \text{otherwise.} \end{cases}$$

By the definition of $\widehat{U}(S)$, it follows that

$$\begin{aligned} \mathbb{E}[\widehat{U}(S)] &\geqslant \frac{1}{L}\mathbb{E}\bigg[\sum_{i<k\in S} \frac{1-\widehat{X}_{ik}}{2}\bigg] = \sum_{i<k\in S} \frac{1-\mathbb{E}[\widehat{X}_{ik}]}{2L} \\ &= \widehat{f}^* := \sum_{i<k\in S} \frac{1-(\nu(2/\pi)\arcsin(\theta X^*_{ik}) + (1-\nu)\theta'^2 X^*_{0i}X^*_{0k})}{2L}. \end{aligned} \tag{3.1}$$

It suffices to show that when $z^*_j > 0$, there exists a constant $\alpha_{|S|} > 0$ (dependent only on $|S|$) such that $\widehat{f}^*/z^*_j \geqslant \alpha_{|S|}$. If $|S| = 2$, then $L = 1$ and

$$\begin{aligned} \widehat{f}^*/z^*_j &= \frac{1-(\nu(2/\pi)\arcsin(\theta X^*_{ik}) + (1-\nu)\theta'^2 X^*_{0i}X^*_{0k})}{2z^*_j} \\ &\geqslant \frac{\nu(1-(2/\pi)\arcsin(\theta X^*_{ik})) + (1-\nu)(1-\theta'^2 X^*_{0i}X^*_{0k})}{2\frac{1-X^*_{ik}}{2}} \\ &\geqslant \min_{-1\leqslant X<1,-1\leqslant Y\leqslant 1} \frac{\nu(1-(2/\pi)\arcsin(\theta X)) + (1-\nu)(1-\theta'^2 Y)}{1-X}, \end{aligned}$$

where the first inequality holds in view of $z^*_j \leqslant \frac{1-X^*_{ik}}{2}$ by (2.6) and the last inequality holds because we assume $z^*_j > 0$ (so $-1 \leqslant X^*_{ik} < 1$) and $-1 \leqslant X^*_{0i}X^*_{0k} \leqslant 1$. We now define

$$\alpha_2 := \min_{-1\leqslant X<1,-1\leqslant Y\leqslant 1} \frac{\nu\,(1-(2/\pi)\arcsin(\theta X)) + (1-\nu)(1-\theta'^2 Y)}{1-X}. \tag{3.2}$$

When $\theta = 1$ and $\nu = 1$, α_2 was calculated by Goemans and Williamson [19] to obtain a 0.878 approximation result for Max Cut.

Next, we consider the case when $|S| \geqslant 3$. For our purpose, we fix parameters $\theta, \theta' \in (0,1]$ and $\nu \in [0,1]$. Let

$$N_{|S|} = |S|(|S|-1)/2, \quad I = [-\lfloor |S|/2 \rfloor, N_{|S|}) \quad \text{and} \quad I' = [-|S|/2, N_{|S|}].$$

For $-1 \leqslant X_{0i}, X_{0k}, X_{ik} \leqslant 1$ which satisfy the triangle inequalities (2.10) and (2.11), we put

$$\lambda = \sum_{i<k\in S} X_{ik} \quad \text{and} \quad \lambda' = \sum_{i<k\in S} X_{0i}X_{0k}.$$

Let

$$\begin{aligned} z'(\lambda) &= \min\left\{1, \frac{N_{|s|}-\lambda}{2(|S|-1)}\right\}, \\ h_1(\lambda) &= \sum_{i<k\in S} 1-(2/\pi)\arcsin(\theta X_{ik}), \quad \text{where } \lambda \in I, \\ h_2(\lambda') &= \sum_{i<k\in S} (1-\theta'^2 X_{0i}X_{0k}), \quad \text{where } \lambda' \in I'. \end{aligned}$$

Moreover, let $\alpha_{|S|} = \min_{\lambda\in I,\lambda'\in I'} \alpha_{|S|}(\lambda,\lambda')$, where

$$\alpha_{|S|}(\lambda,\lambda') = (\nu h_1(\lambda) + (1-\nu)h_2(\lambda'))/(2Lz'(\lambda)).$$

Then we have the following lemma.

Lemma 3.1. $\widehat{f}^*/z_j^* \geqslant \alpha_{|S|}$.

Proof. Let $\lambda^* = \sum_{i<k\in S} X_{ik}^*$. By the constraints (2.6), (2.9) and the assumption $z_j^* > 0$, we have

$$-\lfloor |S|/2 \rfloor \leqslant \lambda^* \leqslant N_{|S|} - 2z_j^*(|S|-1) < N_{|S|}$$

and

$$z_j^* \leqslant \min\left\{1, \frac{1}{|S|-1}\sum_{i<k\in S}\frac{1-X_{ik}^*}{2}\right\} = \min\left\{1, \frac{N_{|S|}-\lambda^*}{2(|S|-1)}\right\} = z'(\lambda^*). \tag{3.3}$$

Moreover, let $\lambda'^* = \sum_{i<k\in S} X_{0i}^* X_{0k}^*$. Then we have

$$-|S|/2 \leqslant \lambda'^* \leqslant N_{|S|},$$

since

$$\left(\sum_{i\in S} X_{0i}^*\right)^2 \leqslant |S| + 2\sum_{i<k\in S} X_{0i}^* X_{0k}^*,$$

and $-1 \leqslant X_{0i}^* \leqslant 1$ for $i \in S$.

From (3.1) and (3.3), we find

$$\begin{aligned}
\widehat{f}^*/z_j^* &\geqslant \sum_{i<k\in S}\frac{\nu\,(1-(2/\pi)\arcsin(\theta X_{ik}^*)) + (1-\nu)(1-\theta'^2 X_{0i}^* X_{0k}^*)}{2Lz'(\lambda^*)} \\
&\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik},\lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} \alpha_{|S|}(\lambda^*, \lambda'^*) \\
&\geqslant \min_{\lambda\in I,\lambda'\in I'}\ \min_{\lambda=\sum_{i<k\in S} X_{ik},\lambda'=\sum_{i<k\in S} X_{0i}X_{0k}} \alpha_{|S|}(\lambda, \lambda') \\
&= \alpha_{|S|},
\end{aligned}$$

where again $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities. □

Since $E[\widehat{U}(S)] \geqslant \widehat{f}^* \geqslant \alpha_{|S|} z_j^*$ if $z_j^* = 0$, we have by Lemma 3.1

Theorem 3.2. *The expected contribution of the edge S_j to the cut generated by executing Steps 1–4 of our algorithm satisfies*

$$E[\widehat{U}(S_j)] \geqslant \alpha_{|S_j|} z_j^*.$$

To establish the bound on the expected contribution of an edge after executing Step 5, we need to compute $\alpha_{|S|}$ efficiently. However, it is not easy to compute it directly. Therefore, we strive to provide a good lower bound for $\alpha_{|S|}$. The key to establish the lower bound on $\alpha_{|S|}$ is to obtain a lower bound on $h_1(\lambda)$ for $\lambda \in I$ first, since it is easy to calculate

$$h_2(\lambda') = \sum_{i<k\in S}(1-\theta'^2 X_{0i}X_{0k}) = N_{|S|} - \theta'^2\lambda', \quad \text{for } \lambda' = \sum_{i<k\in S} X_{0i}X_{0k} \in I'.$$

A lower bound on $h_1(\lambda)$ is obtained by applying the following lemma which has already been proven in [21] using the definitions of concavity and convexity.

Lemma 3.3. *Given $\theta \in (0,1]$, define*

$$d(x) = 1 - (2/\pi)\arcsin(\theta x).$$

Then $d(x)$ is convex in $[-1,0]$ and concave in $[0,1]$. Therefore, $d(x') + d(y') \leqslant d(x) + d(y)$ for $-1 \leqslant x \leqslant x' \leqslant y' \leqslant y \leqslant 0$ with $x + y = x' + y'$, and $d(x) + d(y) \leqslant d(x') + d(y')$ for $0 \leqslant x \leqslant x' \leqslant y' \leqslant y \leqslant 1$ with $x + y = x' + y'$.

With the definition of $d(x)$ in the above lemma, it is easily seen that

$$h_1(\lambda) \geqslant h_1'(\lambda) := \min \sum_{i<k\in S} d(X_{ik}), \tag{3.4}$$

where the minimum is taken over all X such that $\lambda = \sum_{i<k\in S} X_{ik}$, and $-1 \leqslant X_{ik} \leqslant 1$ for $1 \leqslant i < k \leqslant |S|$.

Let $X = (X_{ij})$ be a minimum solution to (3.4). Since $d(x)$ is convex in $[-1, 0]$ and concave in $[0, 1]$ by Lemma 3.3, we may choose a minimum solution X to (3.4) so that at most one entry of X belongs to $(0, 1)$ and all entries of X in $[-1, 0]$ are equal. This nice structural property of the minimum solution helps us calculate $h_1'(\lambda)$ easily.

To derive a lower bound on $\alpha_{|S|}$, we need the following lemma. Let $\overline{N} = (|S| - 4)(|S| - 1)/2$, N be a nonnegative integer and

$$\begin{aligned}
f_1(x, N) &= N_{|S|} + N(2/\pi)\arcsin(\theta) - (N_{|S|} - N - 1)(2/\pi)\arcsin(\theta) \\
&\qquad - (2/\pi)\arcsin(\theta(x - N_{|S|} + 2N + 1)), \\
f_2(x, N) &= N_{|S|} - (N_{|S|} - N)(2/\pi)\arcsin(\theta) - N(2/\pi)\arcsin(\theta x), \\
l_1 &= \min_{N\in[0,N_{|S|}-1]} \left\{ \min_{\lambda\in I(N),\lambda\in I'} [\nu f_1(\lambda, N) + (1-\nu)h_2(\lambda')] \right\}, \\
&\qquad \text{where } I(N) = [-\lfloor |S|/2 \rfloor, \overline{N}] \cap (N_{|S|} - 2N - 1, N_{|S|} - 2N), \\
l_2 &= \min_{N\in[1,N_{|S|}]} \left\{ \min_{\lambda\in J(N),\lambda'\in I'} \left[\nu f_2\left(\frac{\lambda - (N_{|S|} - N)}{N}, N \right) + (1-\nu)h_2(\lambda') \right] \right\}, \\
&\qquad \text{where } J(N) = [-\lfloor |S|/2 \rfloor, \overline{N}] \cap [N_{|S|} - 2N, N_{|S|} - N], \\
l_3 &= \min_{N\in[0,N_{|S|}-1]} \left\{ \min_{\lambda\in K(N),\lambda'\in I'} \frac{\nu f_1(\lambda, N) + (1-\nu)h_2(\lambda')}{N_{|S|} - \lambda} \right\}, \\
&\qquad \text{where } K(N) = [\overline{N}, N_{|S|}) \cap (N_{|S|} - 2N - 1, N_{|S|} - 2N), \\
l_4 &= \min_{N\in[1,N_{|S|}]} \left\{ \min_{\lambda\in M(N),\lambda'\in I'} \frac{\nu f_2(\frac{\lambda - (N_{|S|} - N)}{N}, N) + (1-\nu)h_2(\lambda')}{N_{|S|} - \lambda} \right\}, \\
&\qquad \text{where } M(N) = [\overline{N}, N_{|S|}) \cap [N_{|S|} - 2N, N_{|S|} - N].
\end{aligned}$$

Lemma 3.4. *Given fixed parameters $\theta, \theta' \in (0, 1]$ and $\nu \in [0, 1]$, we have*

$$\alpha_{|S|} = \min_{\lambda\in I,\lambda'\in I'} \alpha(\lambda, \lambda') \geqslant \min\left\{ \frac{1}{2L} l_1, \frac{1}{2L} l_2, \frac{|S|-1}{L} l_3, \frac{|S|-1}{L} l_4 \right\},$$

where again $\lambda = \sum_{i<k\in S} X_{ik}$, $\lambda' = \sum_{i<k\in S} X_{0i}X_{0k}$, and $-1 \leqslant X_{0i}, X_{0k}, X_{ik} \leqslant 1$ satisfy the triangle inequalities.

As the proof of this lemma is full of technical details, we leave it in the appendix. Note that for any fixed $|S|$ and nonnegative integer N, l_1, l_2, l_3 and l_4 can be evaluated numerically. (We used the optimization toolbox of Matlab.) It is easily seen that our scheme enumerates at most $O(|S|^2)$ nonnegative integers N with $N \leqslant N_{|S|}$ for calculating l_1, l_2, l_3 or l_4. We note that in the case when $|S| = 3$, it is not hard to verify that $\alpha_3 = 0.90871$ by setting $\theta = 0.9789$ and $\nu = 1$ which achieves the same bound as in Zwick [44] for Max E3-Set Splitting as mentioned in Section 1.

4 Bounding $\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)]$ after Step 5

The aim of this section is to derive a bound on $\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)]$ in terms of ω^{opt} using the results in the previous section. After executing Step 5 in our SDP-algorithm for MHC-LU, we have the second partition $V = \widetilde{V}_1 \cup (V \setminus \widetilde{V}_1)$.

Let $\widetilde{U}(S_j)$ be an indicator random variable, defined as follows:

$$\widetilde{U}(S_j) = \begin{cases} 1, & \text{if } S_j \in (\widetilde{V}_1, V \setminus \widetilde{V}_1), \\ 0, & \text{otherwise.} \end{cases}$$

Recall the definition of α_2 (see (3.2)) and from Lemma 3.1 the definition of $\alpha_s := \alpha_{|S|}$ (with $|S| = s$) in the previous section. For a suitable nonnegative integer $M \geqslant 4$ and for any integer s with $2 \leqslant s < M$, let

$$r_s(p) = \alpha_s(1 - p(1-p)^{s-1} - p^{s-1}(1-p)) + (1-\alpha_s)(1 - p^s - (1-p)^s),$$
$$r_M(p) = 1 - p^M - (1-p)^M.$$

Note that p is the probability used in Step 5. The following lemma is due to Andersson and Engebretsen [5], who also used the first part of Step 5 as a probabilistic postprocessing step to perturb an initial partition to a new one in their algorithm for Max Set Splitting.

Lemma 4.1. *For any integer $M \geqslant 4$ and any $|S_j| \geqslant 2$,*

$$\mathbb{E}[\widetilde{U}(S_j)] \geqslant \begin{cases} r_{|S_j|}(p) z_j^*, & \text{if } |S_j| < M, \\ r_M(p), & \text{otherwise.} \end{cases}$$

As an immediate consequence, we have the following corollary.

Corollary 4.2. *For any integer $M \geqslant 4$ and any real number $0 \leqslant p \leqslant 1/2$,*

$$\sum_{\{j:|S_j|\geqslant 2\}} \omega_j E[\widetilde{U}(S_j)] \geqslant \sum_{\{j:2\leqslant|S_j|<M\}} \omega_j r_{|S_j|}(p) z_j^* + \sum_{\{j:|S_j|\geqslant M\}} \omega_j r_M(p).$$

Let $\alpha^{(p)} = \min_{2\leqslant s\leqslant M} r_s(p)$. Recall that

$$\omega^{\mathrm{SDP}} = \sum_j \omega_j z_j^* \geqslant \max \sum_j \omega_j z_j = \omega^{\mathrm{opt}}.$$

So we have the following lower bound on $\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)]$.

Corollary 4.3. *For any integer $M \geqslant 4$ and any real number $0 \leqslant p \leqslant 1/2$,*

$$\begin{aligned}
\mathbb{E}[\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)] &= \sum_{\{j:|S_j|\geqslant 2\}} \omega_j \mathbb{E}[\widetilde{U}(S_j)] \\
&\geqslant \sum_{\{j:2\leqslant|S_j|<M\}} \omega_j r_{|S_j|}(p) z_j^* + \sum_{\{j:|S_j|\geqslant M\}} \omega_j r_M(p) z_j^* \\
&\geqslant \alpha^{(p)} \omega^{\mathrm{SDP}} \geqslant \alpha^{(p)} \omega^{\mathrm{opt}}.
\end{aligned}$$

For the hyperedge S_j with size $|S_j| = 2$, the ratio $r_2(p)$ can be improved due to the following lemma given by Zhang et. al [42]. Let $\varphi_\theta(t) = (1/\pi)\arccos(\theta(1-2t))$ and let γ_θ be the minimizer of $\varphi_\theta(t)/t$ in the interval (0, 1]. By setting $x = 1 - 2t$ one can show (see [19]) that

$$\frac{\varphi_\theta(\gamma_\theta)}{\gamma_\theta} = \min_{-1\leqslant x<1} \frac{1 - (2/\pi)\arcsin(\theta x)}{1 - x}.$$

If the algorithm executes Steps 3 and 5 which are also used in [42] (for Max Set Splitting), Zhang et al. [42] give an improved bound on the contribution to $\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)$ of the size 2 edges (in terms of their contribution to the objective function) as follows.

Lemma 4.4. *For $|S_j| = 2$,*

$$\omega_j \mathbb{E}[\widetilde{U}(S_j)] \geqslant \min_{\gamma_\theta\leqslant x\leqslant 1} \frac{\varphi_\theta(x)(1-2p)^2 + (2p - 2p^2)}{x} \omega_j z_j^*.$$

It is easily seen that

$$r_2(p) = \frac{1 - \theta'^2}{2}(1-2p)^2 + (2p - 2p^2)$$

is the ratio for the hyperedge with size $S_j = 2$ if the algorithm executes Steps 4 and 5. This follows by the definition of $r_2(p)$, since

$$\min_{-1\leqslant X<1, -1\leqslant Y\leqslant 1} \frac{1 - \theta'^2 Y}{1 - X} = \frac{1 - \theta'^2}{2}$$

is the bound on the expected contribution of an edge with size $S_j = 2$ by executing Steps 1, 2 and 4 (see Section 3). Since our algorithm executes independently Step 3 with probability ν, and Step 4 with probability $1-\nu$, we have the following corollary.

Corollary 4.5. *For* $|S_j| = 2$,

$$\omega_j \mathbb{E}[\widetilde{U}(S_j)] \geqslant \nu \min_{\gamma_\theta \leqslant x \leqslant 1} \frac{\varphi_\theta(x)(1-2p)^2 + (2p-2p^2)}{x}\omega_j z_j^* + (1-\nu)\left(\frac{1-\theta'^2}{2}(1-2p)^2 + (2p-2p^2)\right)\omega_j z_j^*.$$

We then let

$$r_2(p) = \nu \min_{\gamma_\theta \leqslant x \leqslant 1} \frac{\varphi_\theta(x)(1-2p)^2 + (2p-2p^2)}{x} + (1-\nu)\left(\frac{1-\theta'^2}{2}(1-2p)^2 + (2p-2p^2)\right),$$

in the computation of $\alpha^{(p)}$. Given θ, θ', ν and M, for each $0 \leqslant p \leqslant 1/2$, we can compute (using the results in the previous section) the lower bounds on $\alpha^{(p)}$ which is the performance guarantee of the algorithm for MHC-LU when $\tau = 1$, i.e., Max Set Splitting, by Corollary 4.3. For fast computations and easy verifications of computational results, we only use the triangle inequalities for computing α_3 and α_4 in the previous section and it is also sufficient for us to fix $M = 15$ for obtaining the desired performance guarantee of our algorithm from numerical results. We then obtain approximation ratio 0.7524 for Max Set Splitting by setting the other parameters $\theta = 0.967$, $p = 0.1$ and $\nu = 1$ which improves the result 0.7499 in Zhang et al. [42]. The improvement is a consequence of our strengthened SDP relaxation and the generalized formula for the performance ratio (see Lemma 3.4). If we instead use $\alpha^{(p)} = \min_{3\leqslant s\leqslant M} r_s(p)$ by setting the parameters $\theta = 0.961$, $\theta' = 0.1$, $p = 0.1$ and $\nu = 0.242$, we then obtain an approximation ratio 0.7741 for Max Set Splitting when the minimum number of vertices in a hyperedge is 3. Our algorithm of combining the outward rotations of the random hyperplane rounding procedure with that of a linear randomized rounding procedure improves the result 0.75 in Gaur and Krishnamurti [18].

5 Bounding $\mathbb{E}[|\widetilde{V}_1|(m-|\widetilde{V}_1|)]$

We now proceed to establish a bound on $\mathbb{E}[|\widetilde{V}_1|(m-|\widetilde{V}_1|)]$, which will be used in the next section to obtain the performance guarantee of our SDP algorithm for MHC-LU when $0 \leqslant \tau < 1$.

First, let $\widetilde{X}_{ij}$ be an indicator random variable after the algorithm executes Step 5, defined as ($\widetilde{V}_2 := V \setminus \widetilde{V}_1$)

$$\widetilde{X}_{ij} = \begin{cases} -1, & \text{if } i \in \widetilde{V}_i \text{ and } j \in V \setminus \widetilde{V}_i, \quad \text{for } i = 1, 2, \\ 1, & \text{otherwise.} \end{cases}$$

We then have the following lemma if the algorithm executes Steps 3 and 5.

Lemma 5.1.

$$E[\widetilde{X}_{ij}] = \begin{cases} 1, & \textit{if } i = j, \\ (2/\pi)(1-2p)^2 \arcsin(\theta X_{ij}^*), & \textit{otherwise.} \end{cases}$$

Proof. It is straightforward to see that if $i = j$, then $E[\widetilde{X}_{ij}] = 1$. Now assume $i \neq j$. By Step 3 of our SDP algorithm in Section 2 and the definition of $\widehat{X}_{ij}$ in Section 3, we have

$$\Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij} = -1] = p^2 + (1-p)^2 = 1 - 2p(1-p),$$
$$\Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij} = 1] = 2p(1-p) = 2p - 2p^2,$$

and

$$E[\widetilde{X}_{ij} \mid \widehat{X}_{ij}] = \Pr[\widetilde{X}_{ij} = 1 \mid \widehat{X}_{ij}] - \Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij}]$$
$$= 1 - \Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij}] - \Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij}]$$

$$= 1 - 2\Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij}].$$

From [19, 44], we have

$$\Pr(\widehat{X}_{ij} = -1) = \arccos(\theta X^*_{ij})/\pi.$$

Thus,

$$\begin{aligned}
E[\widetilde{X}_{ij}] &= \Pr(\widehat{X}_{ij} = -1)E[\widetilde{X}_{ij} \mid \widehat{X}_{ij} = -1] + \Pr(\widehat{X}_{ij} = 1)E[\widetilde{X}_{ij} \mid \widehat{X}_{ij} = 1] \\
&= \Pr(\widehat{X}_{ij} = -1)(1 - 2\Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij} = -1]) + \Pr(\widehat{X}_{ij} = 1)(1 - 2\Pr[\widetilde{X}_{ij} = -1 \mid \widehat{X}_{ij} = 1]) \\
&= \Pr(\widehat{X}_{ij} = -1)(4p - 4p^2 - 1) + (1 - \Pr(\widehat{X}_{ij} = -1))(1 - 4p + 4p^2) \\
&= (1 - 2p)^2(1 - 2\Pr(\widehat{X}_{ij} = -1)) \\
&= (2/\pi)(1 - 2p)^2(\pi/2 - \arccos(\theta X^*_{ij})) \\
&= (2/\pi)(1 - 2p)^2 \arcsin(\theta X^*_{ij}).
\end{aligned}$$

□

Next, for $1 \leqslant i \leqslant m$ let $p(i)$ be the probability of putting i into V_1 if the algorithm executes Steps 4 and 5. We have the following lemma.

Lemma 5.2.

$$p(i) = \frac{1 + (1 - 2p)\theta' X^*_{0i}}{2}.$$

Proof. Recall that $\widehat{V}_1$ is formed in Step 4 by adding, independently, each $1 \leqslant i \leqslant m$ with probability $\frac{1+\theta' X^*_{0i}}{2}$, and V_1 is constructed in Step 5 by independently perturbing each element of $\widehat{V}_1$ with probability p to the opposite subset and each element of the opposite subset to $\widehat{V}_1$. It is easily seen that

$$p(i) = \frac{1 + \theta' X^*_{0i}}{2}(1 - p) + \frac{1 - \theta' X^*_{0i}}{2}p = \frac{1 + (1 - 2p)\theta' X^*_{0i}}{2}.$$

□

Recall that m is the number of vertices and $\tau = u/m$ is the unbalance parameter. Let

$$\begin{aligned}
\chi &= \min_{-1 \leqslant x < 1} (2/\pi)(1 - 2p)^2 \frac{\arcsin(\theta) - \arcsin(\theta x)}{1 - x}, \\
\varphi_\chi &= 1 - (2/\pi)(1 - 2p)^2 \arcsin(\theta).
\end{aligned}$$

We can now present the bound on $\mathbb{E}(|\widetilde{V}_1|(m - |\widetilde{V}_1|))$ as follows.

Lemma 5.3. *For sufficiently large m,*

$$\mathbb{E}[|\widetilde{V}_1|(m - |\widetilde{V}_1|)] \geqslant \frac{m^2}{4}(\nu(\varphi_\chi + (1 - \tau^2)\chi) + (1 - \nu)(1 - \theta'(1 - 2p)\tau)^2 + o(1)).$$

Proof. Note that the algorithm executes independently Step 3 with probability ν, and Step 4 with probability $1 - \nu$. In Step 5, we have $\widetilde{V}_1 = V_1$ if $|V_1| \geqslant m/2$, and $\widetilde{V}_1 = V \setminus V_1$ otherwise. So $|\widetilde{V}_1|(m - |\widetilde{V}_1|)$ $= |V_1|(m - |V \setminus V_1|)$ and

$$\begin{aligned}
\mathbb{E}[|\widetilde{V}_1|(m - |\widetilde{V}_1|)] &= \nu\mathbb{E}[(1/4)\sum_{i,j}(1 - \widetilde{X}_{ij})] + (1 - \nu)\sum_{1 \leqslant i \neq j \leqslant m} p(i)(1 - p(j)) \\
&= \nu(1/4)\sum_{i \neq j}(1 - (2/\pi)(1 - 2p)^2 \arcsin(\theta X^*_{ij})) \\
&\quad + (1 - \nu)\sum_{1 \leqslant i \neq j \leqslant m}\left(\frac{1 + (1 - 2p)\theta' X^*_{0i}}{2} \cdot \frac{1 - (1 - 2p)\theta' X^*_{0j}}{2}\right) \\
&= \nu(1/4)\sum_{i \neq j}(\varphi_\chi + (2/\pi)(1 - 2p)^2(\arcsin(\theta) - \arcsin(\theta X^*_{ij}))) \\
&\quad + (1 - \nu)\sum_{i=1}^{m}\left(\frac{1 + (1 - 2p)\theta' X^*_{0i}}{2} \cdot \sum_{j \neq i}\frac{1 - (1 - 2p)\theta' X^*_{0j}}{2}\right)
\end{aligned}$$

$$
\begin{aligned}
&\geqslant \nu(1/4)\sum_{i\neq j}(\varphi_\chi + (1-X^*_{ij})\chi)\\
&\quad + (1-\nu)\sum_{i=1}^{m}\bigg(\frac{1+(1-2p)\theta' X^*_{0i}}{2}\cdot\sum_{j=1}^{m}\bigg(\frac{1-(1-2p)\theta' X^*_{0j}}{2}-\frac{1-(1-2p)\theta' X^*_{0i}}{2}\bigg)\bigg)\\
&\geqslant \nu(1/4)\bigg(\sum_{i\neq j}\varphi_\chi + (m^2-u^2)\chi\bigg)\\
&\quad + (1-\nu)\bigg(\frac{m}{2}-\frac{(1-2p)\theta' u}{2}\bigg)\bigg(\sum_{j=1}^{m}\frac{1-(1-2p)\theta' X^*_{0j}}{2}-1\bigg)\\
&\geqslant \nu(1/4)(m(m-1)\varphi_\chi + m^2(1-\tau^2\chi))\\
&\quad + (1-\nu)m\bigg(\frac{1}{2}-\frac{\theta'(1-2p)\tau}{2}\bigg)\bigg(\frac{m}{2}-\frac{\theta'(1-2p)u}{2}-1\bigg)\\
&= \nu(1/4)m^2((1-1/m)\varphi_\chi + (1-\tau^2)\chi)\\
&\quad + (1-\nu)m^2\bigg(\frac{1}{2}-\frac{\theta'(1-2p)\tau}{2}\bigg)\bigg(\frac{1}{2}-\frac{\theta'(1-2p)\tau}{2}-\frac{1}{m}\bigg)\\
&= (1/4)m^2(\nu(\varphi_\chi + (1-\tau^2)\chi) + (1-\nu)(1-\theta'(1-2p)\tau)^2 + o(1)) \quad \text{(since } m \text{ is large)},
\end{aligned}
$$

where the second equality follows from Lemmas 5.1 and 5.2, the first inequality follows from the definition of χ, and the second inequality follows from (2.7) (note that when $i=j$, $X^*_{ii}=1$, so $\sum_{i\neq j}(1-X^*_{ij}) \geqslant m^2-u^2$), and (2.8) ($\sum_{i=1}^m X^*_{0i}\leqslant u$). □

6 The quality of the SDP approximation algorithm.

Recall that $\tau = u/m \in [0,1]$ is the unbalance parameter. For $\gamma > 0$, let

$$Z(\gamma) = \frac{\omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)}{\omega^{opt}} + \gamma\frac{|\widetilde{V}_1|(m-|\widetilde{V}_1|)}{m^2}.$$

Also recall that $\overline{V}_1$ is given in Step 6 of our SDP algorithm. We now give a lower bound on $\omega(\overline{V}_1, V\setminus\overline{V}_1)$ in terms of $\omega^{\rm opt}$, whenever $Z(\gamma)$ fulfills its expectation. This will be used to establish the bound on the quality of the SDP approximation algorithm for MHC-LU when $0\leqslant\tau<1$.

First, we show that

$$\omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega^{\rm opt} \leqslant 2/(1+\tau).$$

If $|\widetilde{V}_1|\leqslant (m+u)/2$ then $\overline{V}_1 = \widetilde{V}_1$ (see Step 6 of our algorithm in Section 2), and hence

$$\omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega^{\rm opt} \leqslant \omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega(\overline{V}_1, V\setminus\overline{V}_1) = 1 \leqslant 2/(1+\tau).$$

So we may assume $|\widetilde{V}_1| > (m+u)/2$. Then by Lemma 2.1,

$$\omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega(\overline{V}_1, V\setminus\overline{V}_1) \leqslant 2|\widetilde{V}_1|/(m+u),$$

and hence

$$\omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega^{\rm opt} \leqslant \omega(\widetilde{V}_1, V\setminus\widetilde{V}_1)/\omega(\overline{V}_1, V\setminus\overline{V}_1) \leqslant 2|\widetilde{V}_1|/(m+u) \leqslant 2m/(m+u) = 2/(1+\tau).$$

Since

$$\gamma\frac{|\widetilde{V}_1|(m-|\widetilde{V}_1|)}{m^2} = \bigg(-\frac{|\widetilde{V}_1|^2}{m^2}+\frac{|\widetilde{V}_1|}{m}\bigg)\gamma \leqslant \frac{\gamma}{4},$$

we have

$$Z(\gamma) \leqslant A := \frac{2}{1+\tau}+\frac{\gamma}{4}. \tag{6.1}$$

Let

$$\beta = (\nu(\varphi_\chi + (1-\tau^2)\chi) + (1-\nu)(1-\theta'(1-2p)\tau)^2 + o(1))/4.$$

Then by Corollary 4.3 and Lemma 5.3, we have

$$\mathbb{E}(Z(\gamma)) = \mathbb{E}(\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1))/\omega^{\text{opt}} + \gamma \mathbb{E}(|\widetilde{V}_1|(m - |\widetilde{V}_1|))/m^2 \geqslant B := \alpha^{(p)} + \gamma\beta. \tag{6.2}$$

For any given $0 < \epsilon < 1$, let

$$r = \Pr\{Z(\gamma) < B - \epsilon(A - B)\}.$$

Then

$$\mathbb{E}[Z(\gamma)] < r(B - \epsilon(A-B)) + (1-r)\max(Z(\gamma)). \tag{6.3}$$

From (6.1)–(6.3), we have

$$r < \frac{A - B}{A - B + \epsilon(A - B)} = \frac{1}{1+\epsilon}.$$

We run the proposed algorithm $K = O((1/\epsilon)\log(1/\epsilon))$ times, and denote the maximum value of $Z(\gamma)$ in these K runs by Z_K. Then

$$\Pr(Z_K < B - \epsilon(A-B)) \leqslant r^K < \left(\frac{1}{1+\epsilon}\right)^K \leqslant \epsilon.$$

This implies that

$$Z_K \geqslant B - O(\epsilon) = \alpha^{(p)} + \gamma\beta - O(\epsilon)$$

with probability at least $1-\epsilon$. Since ϵ can be made arbitrarily small, almost surely

$$\frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{\omega^{\text{opt}}} + \gamma\frac{|\widetilde{V}_1|(m - |\widetilde{V}_1|)}{m^2} = Z(\gamma) \geqslant \alpha^{(p)} + \gamma\beta - o(1). \tag{6.4}$$

Let $|\widetilde{V}_1| = qm$, where $1/2 \leqslant q \leqslant 1$ (see Step 5 in Section 2). Then from (6.4), almost surely,

$$\frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{\omega^{\text{opt}}} \geqslant \alpha^{(p)} + \gamma\beta - \gamma q(1-q) - o(1).$$

By Lemma 2.1, if $|\widetilde{V}_1| \leqslant \frac{m+u}{2}$, we obtain

$$\omega(\overline{V}_1, V \setminus \overline{V}_1) = \omega(\widetilde{V}_1, V \setminus \widetilde{V}_1),$$

otherwise, we have (almost surely)

$$\begin{aligned}\frac{\omega(\overline{V}_1, V \setminus \overline{V}_1)}{\omega^{\text{opt}}} &\geqslant \frac{m+u}{2|\widetilde{V}_1|} \cdot \frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{\omega^{\text{opt}}} = \frac{1+\tau}{2q} \cdot \frac{\omega(\widetilde{V}_1, V \setminus \widetilde{V}_1)}{\omega^{\text{opt}}} \\ &\geqslant \frac{1+\tau}{2q}(\alpha^{(p)} + \gamma\beta - \gamma q(1-q) - o(1)).\end{aligned}$$

Write

$$\mu(q) = \alpha^{(p)} + \gamma\beta - \gamma q(1-q).$$

Then almost surely,

$$\frac{\omega(\overline{V}_1, V \setminus \overline{V}_1)}{\omega^{\text{opt}}} \geqslant \min\left\{\mu(q) - o(1), \frac{1+\tau}{2q}\mu(q) - o(1)\right\}.$$

We now compute the lower bound on $\omega(\overline{V}_1, V \setminus \overline{V}_1)$ in terms of ω^{opt}, whenever $Z(\gamma)$ fulfills its expectation. Let

$$\gamma_0 = \frac{\alpha^{(p)}}{2\beta}\left(\frac{1}{\sqrt{1-4\beta}} - 1\right),$$

$$\gamma_1 = \frac{4\alpha^{(p)}}{(1+2\tau)^2 - 4\beta},$$
$$\gamma_2 = \frac{\alpha^{(p)}(2-2\tau)}{1-2(1-\tau)\beta}.$$

Note that γ_0, γ_1 and γ_2 are chosen to optimize certain functions. Now we have the following lemma.

Lemma 6.1. *If the random variable fulfills its expectation, i.e., $Z(\gamma) \geqslant \alpha^{(p)} + \gamma\beta - o(1)$, then almost surely,*

$$\omega(\overline{V}_1, V \setminus \overline{V}_1)/\omega^{\rm opt} \geqslant \overline{R}(\tau) - o(1),$$

where

$$\overline{R}(\tau) = \begin{cases} \overline{R}_0, & \text{if } \tau = 0,\ \beta < 1/4 \text{ and } \frac{\alpha^{(p)}}{1-\beta} \leqslant \gamma = \gamma_0 \leqslant \frac{4\alpha^{(p)}}{1-4\beta}, \\ \overline{R}_1, & \text{if } 0 < \tau \leqslant 1/2, \text{ and } \gamma = \gamma_1, \\ \overline{R}_2, & \text{if } 1/2 < \tau < 1, \text{ and } \gamma = \gamma_2, \end{cases}$$

and

$$\overline{R}_0 = \frac{\alpha^{(p)}}{1+\sqrt{1-4\beta}},$$
$$\overline{R}_1 = \frac{\alpha^{(p)}(1+\tau)\tau}{(1+2\tau)^2/4 - \beta},$$
$$\overline{R}_2 = \frac{\alpha^{(p)}(1+\tau)}{2-4(1-\tau)\beta}.$$

Proof. When $\tau = 0$, we have

$$\begin{aligned}\omega(\overline{V}_1, V \setminus \overline{V}_1)/\omega^{\rm opt} &\geqslant \min\left\{\mu(q) - o(1), \frac{1+\tau}{2q}\mu(q) - o(1)\right\} \\ &= \min\left\{\mu(q) - o(1), \frac{\mu(q)}{2q} - o(1)\right\} \\ &\geqslant \frac{\mu(q)}{2q} - o(1) \\ &= (\alpha^{(p)} + \gamma\beta - \gamma q(1-q))/(2q) - o(1) \\ &\geqslant \sqrt{\gamma(\alpha^{(p)} + \gamma\beta)} - \gamma/2 - o(1) \\ &\geqslant \alpha^{(p)}/(1+\sqrt{1-4\beta}) - o(1) \\ &= \overline{R}_0 - o(1).\end{aligned}$$

The second inequality follows from $q \geqslant 1/2$. The third inequality follows from the simple calculus that $q = q^* = \sqrt{\alpha^{(p)}/\gamma + \beta}$ yields the minimal value for $\frac{\mu(q)}{2q}$ when $q \in [1/2, 1]$. It is easy to verify that $q^* \in [1/2, 1]$ if $4\beta < 1$ and

$$\alpha^{(p)}/(1-\beta) \leqslant \gamma \leqslant 4\alpha^{(p)}/(1-4\beta).$$

Substituting $\gamma = \gamma_0$ into the third inequality yields the maximal value for $\sqrt{\alpha^{(p)} + \gamma\beta} - \gamma/2$ and gives the last inequality. So we obtain the first result in the lemma.

Next, when $\tau > 0$, we select $\gamma = \gamma_1$ and $\gamma = \gamma_2$ when $0 < \tau \leqslant 1/2$ and $1/2 \leqslant \tau \leqslant 1$, respectively, to make the minimal values of $\omega(\overline{V}_1, V \setminus \overline{V}_1)/\omega^{\rm opt}$ equal in both of the cases of

$$\frac{1+\tau}{2q} \leqslant 1 \quad \text{and} \quad \frac{1+\tau}{2q} \geqslant 1.$$

Case 1. Suppose $\frac{1+\tau}{2q} \leqslant 1$. Then $q \in [\frac{(1+\tau)}{2}, 1]$, and

$$\omega(\overline{V}_1, V \setminus \overline{V}_1)/\omega^{\rm opt} \geqslant \frac{1+\tau}{2q}\mu(q) - o(1).$$

Simple calculations show that the function $\frac{1+\tau}{2q}\mu(q)$ has $q^*=\sqrt{\alpha^{(p)}/\gamma+\beta}$ as its stationary point and is decreasing in $(0,q^*)$.

If $0<\tau\leqslant 1/2$ and $\gamma=\gamma_1$, then $q^*=(1+2\tau)/2$; so $\frac{(1+\tau)}{2}<q^*\leqslant 1$, and the function $\frac{1+\tau}{2q}\mu(q)$ achieves its minimal value at $q=q^*\in(\frac{(1+\tau)}{2},1]$. Thus,

$$\begin{aligned}\frac{\omega(\overline{V}_1,V\setminus\overline{V}_1)}{\omega^{opt}}&\geqslant\frac{1+\tau}{2q^*}\mu(q^*)-o(1)\\&=(2\sqrt{(\alpha^{(p)}+\gamma\beta)\gamma}-\gamma)\frac{1+\tau}{2}-o(1)\\&=\frac{\alpha^{(p)}(1+\tau)\tau}{(1+2\tau)^2/4-\beta}-o(1)\\&=\overline{R}_1-o(1).\end{aligned}$$

If $1/2<\tau<1$ and $\gamma=\gamma_2$, then

$$q^*=\sqrt{\frac{1}{2-2\tau}}.$$

Since $\tau>1/2$, $q^*>1$, it follows that the minimal value of $\frac{1+\tau}{2q}\mu(q)$ is achieved at 1 by the fact that $\frac{1+\tau}{2q}\mu(q)$ is decreasing in $(0,q^*)$ and $q\in[\frac{(1+\tau)}{2},1]$. Then

$$\begin{aligned}\frac{\omega(\overline{V}_1,V\setminus\overline{V}_1)}{\omega^{\rm opt}}&\geqslant\frac{1+\tau}{2q}\mu(q)-o(1)\geqslant\frac{1+\tau}{2}\mu(1)-o(1)\\&=\frac{\alpha^{(p)}(1+\tau)}{2-4(1-\tau)\beta}-o(1)=\overline{R}_2-o(1).\end{aligned}$$

Case 2. Now assume $\frac{1+\tau}{2q}\geqslant 1$. Then $1/2\leqslant q\leqslant\frac{1+\tau}{2}$, and

$$\omega(\overline{V}_1,V\setminus\overline{V}_1)/\omega^{\rm opt}\geqslant\mu(q)-o(1).$$

Simple calculations show that $\mu(q)$ has a minimum value at $q^*=\frac{1}{2}$.

If $0<\tau\leqslant 1/2$ and $\gamma=\gamma_1$, we have

$$\begin{aligned}\omega(\overline{V}_1,V\setminus\overline{V}_1)/\omega^{\rm opt}&\geqslant\mu(q^*)-o(1)\\&=\alpha^{(p)}+\gamma\beta-\gamma/4-o(1)\\&=\frac{\alpha^{(p)}(1+\tau)\tau}{(1+2\tau)^2/4-\beta}-o(1)\\&=\overline{R}_1-o(1).\end{aligned}$$

If $1/2<\tau<1$ and $\gamma=\gamma_2$, we have

$$\omega(\overline{V}_1,V\setminus\overline{V}_1)/\omega^{\rm opt}\geqslant\mu(q^*)-o(1)=\frac{\alpha^{(p)}(1+\tau)}{2-4(1-\tau)\beta}-o(1)=\overline{R}_2-o(1).$$

This completes the proof. □

For given fixed value τ, we select θ, θ', ν, p as solution to

$$\begin{aligned}&\max\overline{R}(\tau)\\ s.t.\quad&0<\theta\leqslant 1,\\&0<\theta'\leqslant 1,\\&0\leqslant\nu\leqslant 1,\\&0\leqslant p\leqslant 0.5.\end{aligned}\tag{6.5}$$

Then, we have the following conclusion:

Theorem 6.2. *For given fixed value τ, the worst-case performance ratio of the SDP-algorithm for approximating MHC-LU is at least $\overline{R}(\tau) - o(1)$.*

The quality of the approximation depends on the optimization of $\overline{R}(\tau)$ over all parameters θ, θ', ν and p. However, it is not easy to compute $\max \overline{R}(\tau)$ from (6.5). It would be interesting to have an efficient scheme for finding a best set of these parameters.

In practice, instead of solving (6.5) one could perform 1-dimensional searches on the parameters (with τ fixed) and obtain a lower bound $\overline{R}^*(\tau)$ on the performance guarantee of the algorithm. The use of numerical methods to evaluate the performance guarantee is rather common for SDP-based approximation algorithms, see [11,14–17,19,21,22,41,42]. For fast computations and easy verifications of computational results, we may assign simple values to some of the parameters and simply perform 1-dimensional searches on the remaining parameters. This turns out to be sufficient to improve some previous known results mentioned in the introduction. Table 4 shows the values of parameters used in such computations of $\overline{R}^*(\tau)$ for MHC-LU for the same range of $0 \leqslant \tau < 1$ as that in Table 1. It seems that $\nu = 1$ is the optimal choices for the problem from numerical results. Table 5 shows the values of parameters used in such computations of $\overline{R}^*(\tau)$ for MHC-LU for the same range of $0 \leqslant \tau < 1$ as that in Table 2 when the minimum number of vertices in a hyperedge is 3. In such cases, our algorithm of combining the outward rotations of a random hyperplane rounding procedure with that of a linear randomized rounding procedure can always obtain better performance ratios from numerical results.

Note that $\alpha^{(p)} = r_2(p)$ holds when the hypergraph is 2-uniform (a standard graph). If we replace $\alpha^{(p)}$ by $r_2(p)$ in Lemma 6.1 and Theorem 6.2, we obtain the corresponding results for Max Cut with Limited Unbalance. It seems that $\nu = 1$ and $p = 0$ are the optimal choices for the problem from numerical results. Table 6 shows the values of parameters used in such computations of $\overline{R}^*(\tau)$ for MC-LU for the same range of τ as that in Table 3.

We next give a detailed explanation for the reason of numerical improvements shown in Table 3. When $0 \leqslant \tau \leqslant 0.5$, we have the same performance ratios as those in [17], as we use the same idea for estimating the values of performance ratios in these cases. The main idea is to choose a suitable formula of γ such that the minimum value of $\frac{(1+\tau)\mu(q)}{2q}$ is equal to that of $\mu(q)$ which occurs if $q = q^*$ (see the proof of Lemma 6.1). However, when $0.5 < \tau < \sqrt{2/3}$, we obtain improved results compared with that in [17], since we have shown in the proof of Lemma 6.1 that the minimum value of $\frac{(1+\tau)\mu(q)}{2q}$ is $\frac{(1+\tau)\mu(1)}{2}$. This value is greater than or equal to the value $\frac{(1+\tau)\mu(q^*)}{2q^*}$ used in [17] (due to $q^* > 1$ and the fact that $\frac{1+\tau}{2q}\mu(q)$ is decreasing in $(0, q^*)$).

At last, we explain the reason that our performance ratios are still better than those in [17] despite a new formula ($= 0.87856\frac{1+\tau}{2}$) for the performance ratios for $\tau \geqslant \sqrt{2/3}$ is used in [17]. In these cases, our formula for the performance ratio is

$$\overline{R}_2 = \frac{\alpha^{(p)}(1+\tau)}{2 - 4(1-\tau)\beta} = \frac{r_2(p)}{1 - 2(1-\tau)\beta} \cdot \frac{1+\tau}{2},$$

where $\nu = 1$ and $p = 0$. Then by choosing the other parameter values shown as in Table 6 for $\tau \geqslant \sqrt{2/3}$ where $1 - 2(1-\tau)\beta \leqslant 1$, it is easy to verify that the better numerical results ($\geqslant 0.87856\frac{1+\tau}{2}$) can be obtained (see Table 3).

Table 4 Values of parameters used in the computation of $\overline{R}^*(\tau)$ for MHC-LU for some τ

τ	0	0.25000	0.50000	0.75000	0.80000	0.90000	0.99900
θ	0.92300	0.96700	0.96700	0.96700	0.96700	0.96700	0.96700
p	0.10000	0.10000	0.10000	0.10000	0.10000	0.10000	0.10000
γ	7.21650	2.27350	0.95060	0.41060	0.32160	0.15480	0.00150
β	0.24240	0.23150	0.20580	0.16920	0.16040	0.13950	0.11164
$\alpha^{(p)}$	0.73600	0.75240	0.75240	0.75240	0.75240	0.75240	0.75240
$\overline{R}^*(\tau)$	0.62710	0.71050	0.71300	0.71940	0.72450	0.73530	0.75220

Table 5 Values of parameters used in the computation of $\overline{R}^*(\tau)$ for MHC-LU for some τ when the minimum number of vertices in a hyperedge is 3

τ	0	0.2500	0.5000	0.7500	0.9000	0.9990
θ	0.9610	0.9610	0.9610	0.9610	0.9610	0.9610
θ'	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
p	0.1000	0.1000	0.1000	0.1000	0.1000	0.1000
ν	0.2420	0.2420	0.2420	0.2420	0.2420	0.2420
γ	14.1884	2.3870	0.9993	0.4322	0.1612	0.0015
β	0.2475	0.2382	0.2253	0.2089	0.1974	0.1891
$\alpha^{(p)}$	0.7741	0.7741	0.7741	0.7741	0.7741	0.7741
$\overline{R}^*(\tau)$	0.7042	0.7459	0.7495	0.7564	0.7656	0.7740

Table 6 Values of parameters used in the computation of $\overline{R}^*(\tau)$ for MC-LU for some τ

τ	0.6000	0.7000	0.8000	0.8500	0.9000	0.9500	0.9999
θ	0.9800	0.9900	0.9900	0.9900	1.0000	1.0000	1.0000
γ	0.7987	0.5684	0.1272	0.2689	0.1772	0.0880	0.0002
β	0.1591	0.1272	0.0964	0.0795	0.0417	0.0214	0.0000
α^p	0.8713	0.8750	0.8750	0.8750	0.8786	0.8786	0.8786
$\overline{R}^*(\tau)$	0.7987	0.8052	0.8191	0.8291	0.8417	0.8584	0.8785

Due to the use of Lemma 5.3, the values reported in the above tables are correct only for sufficiently large m.

Finally, we give the worst-case performance ratio of the SDP-algorithm for approximating MHC-LU regardless of the value of τ.

Theorem 6.3. *The worst-case performance ratio of the SDP-algorithm for approximating MHC-LU is at least* $\overline{R}_0 - o(1)$.

Proof. Since $\tau \geqslant 0$, we have, with the same arguments as that in the proof of Lemma 6.1,

$$\begin{aligned}\omega(\overline{V}_1, V\setminus\overline{V}_1)/\omega^{\text{opt}} &\geqslant \min\left\{\mu(q)-o(1), \frac{(1+\tau)\mu(q)}{2q}-o(1)\right\}\\ &\geqslant \min\left\{\mu(q)-o(1), \frac{\mu(q)}{2q}-o(1)\right\}\\ &\geqslant \overline{R}_0 - o(1).\end{aligned}$$

Thus, the worst-case performance ratio of our SDP-algorithm for approximating MHC-LU is at least $\overline{R}_0 - o(1)$, regardless of the value of τ. □

It is easy to see that the numerical worst-case performance ratio of the SDP-algorithm for approximating MHC-LU, regardless of the value of τ, is 0.6271 which is the same as that for MHC-LU when $\tau = 0$.

The technique of the Lasserre Hierarchy is to use a family of stronger SDP relaxations derived by the so-called Lasserre lift-and-project system whose vector solutions enjoy nice structural properties. It is very interesting to consider whether an application of Lasserre Hierarchy can lead to a significantly improved approximation ratio for MHC-LU.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 11171160, 11331003 and 11471003), the Priority Academic Program Development of Jiangsu Higher Education Institutions, the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant No. 13KJB1100188), Natural Science Foundation of Guangdong Province (Grant No. S2012040007521) and Science and Technology Planning Project in Guangzhou (Grant No. 2013J4100077). The authors are indebted to the helpful comments and suggestions of the referees and they are also indebted to Professor Yinyu Ye and Professor Qiaoming Han for their helpful discussion on Max Bisection, and Professor Deren Han and Professor Georg Still for their helpful suggestions on MHC-LU.

References

1 Ageev A A, Sviridenko M I. Approximation algorithms for maximum coverage and max cut with given sizes of parts. In: Proceedings of Integer Programming and Combinatorial Optimization. Berlin: Springer, 1999, 17–30

2 Ageev A A, Sviridenko M I. An approximation algorithm for Hypergraph Max k-Cut with given sizes of parts. In: Paterson M, ed. Proceedings of the 8th Annual European Symposium on Algorithms. Lecture Notes in Comput Sci. New York: Springer-Verlag, 2000, 32–41

3 Alimonti P. Non-oblivious local search for graph and hypergraph coloring problems. In: Proceedings of 21st International Workshop on Graph-Theoretic Concepts in Computer Science. Lecture Notes in Computer Sciences, vol. 1017. Berlin: Springer, 1995, 167–180

4 Andersson G. An approximation algorithm for max p-section. In: Proceedings of 16th International Symposium on Theoretical Aspects of Computer Science. Lecture Notes in Computer Science, vol. 1563. Berlin: Springer-Verlag, 1999, 237–247

5 Andersson G, Engebretsen L. Better approximation algorithms for Set splitting and Not-All-Equal SAT. Inform Process Lett, 1998, 65: 305–311

6 Arora S, Karger D, Karpinshi M. Polynomial Time Approximation Schemes for Dense Instances of NP-hard problems. J Comput Syst Sci, 1999, 58: 193–210

7 Austrin P, Benabbas S, Georgiou K. Better Balance by Being Biased: A 0.8776-Approximation for Max Bisection. ArXiv:1205.0458

8 Bollobás B, Scott A D. Problems and results on judicious partitions. Random Struct Alg, 2002, 21: 414–430

9 Bollobás B, Scott A D. Judicious partitions of bounded-degree graphs. J Graph Theory, 2004, 46: 131–143

10 Cong J, Li H P, Lim S K, et al. Large scale circuit partitioning with loose stable net removal and signal flow based clustering. ICCAD, 1997, 441–446

11 Feige U, Goemans M X. Approximating the value of two prover proof systems, with applications to MAX 2SAT and MAX DICUT. In: Proceedings of 3rd Israel Symposium on the Theory of Computing and Systems. Tel Aviv: IEEE Computer Society, 1995, 182–189

12 Feige U, Karpinski M. Langberg M. A note on approximating Max-Bisection on regu-lar graphs. Inform Process Lett, 2001, 79: 181–188

13 Feige U, Karpinski M, Langberg M. Improved approximation of max-cut on graphs of bounded degree. J Algorithms, 2002, 43: 201–219

14 Feige U, Langberg M. Approximation algorithms for maximization problems arising in graph partitioning. J Algorithms, 2001, 41: 174–211

15 Feige U, Langberg M. The RPR^2 rounding technique for semidefinite programs. J Algorithms, 2006, 60: 1–23

16 Frieze V, Jerrum M. Improved approximation algorithms for max k-cut and max bisection. Algorithmica, 1997, 18: 67–81

17 Galbiati G, Maffioli F. Approximation algorithms for maximum cut with limited unbalance. Theor Comput Sci, 2007, 385: 78–87

18 Gaur D R, Krishnamurti R. Simple approximation algorihtms for MAXNAEESP and Hypergraph 2-colorability. J Combin Optim, 2001, 5: 167–173

19 Gomans M X, Williamson D P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. J ACM, 1995, 42: 1115–1145

20 Guruswami V. Inapproximability results for set splitting and satisfiability problems with no mixed clauses. Algorithmica, 2003, 38: 451–469

21 Halperin E, Zwick U. A unified framework for obtaining improved approximation algorithms for maximum graph bisection problems. Random Struct Alg, 2002, 20: 382–402

22 Han Q, Ye Y, Zhang J. An improved rounding method and semidefinite programming relaxation for graph partition. Math Program, 2002, 92: 509–535

23 Hassin R, Rubinstein S. Approximation algorithms for maximum linear arrangement. In: Halldórsson M M, ed. Proceedings of the 7th Scandinavian Workshop on Algorithm Theory. Lecture Notes In Comput Sci, vol. 1851. New York: Springer-Verlag, 2000, 231–236

24 Håstad J. Some optimal inapproximability results. J ACM, 2001, 48: 798–859

25 Hauck S, Borriello G. An Evaluation of Bipartitioning Techniques. IEEE Trans CAD, 1997, 16: 849–866

26 Huang Y, Zhang S. Approximation algorithms for indefinite complex quadratic maximization problems. Sci China

Math, 2010, 53: 2697–2708
27 Jäger G, Srivastav A. Improved approximation algorithms for maximum graph partitioning problems. J Combin Optim, 2005, 10: 133–167
28 Kann V, Lagergren J, Panconesi A. Approximability of maximum splitting of k-sets and some other APX-complete problems. Inform Process Lett, 1996, 58: 105–110
29 Karypis G, Aggarwal R, Kumar V, et al. Multilevel hypergraph partitioning: Applications in VLSI domain. IEEE Trans VLSI Syst, 1999, 7: 69–79
30 Lee C, Loh P S, Sudakov B. Bisections of graphs. ArXiv:1109.3180, 2011
31 Liu L T, Kuo M T, Huang S C, et al. A Gradient Method on the Initial Partition of Fiduccia-Mattheyses Algorithm. In: Proceedings of the 1995 IEEE/ACM international conference on Computer-aided design. Washington, DC: IEEE Computer Society, 1995, 229–234
32 Lovász L. Coverings and colorings of hypergraphs. In: Proceedings of the 4th Southeastern Conference on Combinatorics, Graph Theory, and Computing. Winnipeg: Utilitas Mathematica Publishing, 1973, 3–12
33 Nesterov Y E. Semidefinite relaxation and nonconvex quadratic optimization. Optimi Methods Softw, 1998, 9: 141–160
34 Petrank E. The hardness of approximation: gap location. Comput Complex, 1994, 4: 133–157
35 Raghavendra P, Tan N. Approximating CSPs with global cardinality constraints using SDP hierarchies. ArXiv:1110.1064, 2012
36 Sanchis L. Multiple-way Network Partitioning with Different Cost Functions. IEEE Trans Comp, 1993, 42: 1500–1504
37 Xia Y. New semidefinite programming relaxations for box constrained quadratic program. Sci China Math, 2013, 56: 877–886
38 Xu B, Yan J, Yu X. Balanced judicious partitions of graphs. J Graph Theory, 2010, 63: 210–225
39 Xu B, Yan J, Yu X. A note on balanced bipartition. Discrete Math, 2010, 310: 2613–2617
40 Xu B, Yu X. On judicious bisections of graphs. J Combin Theory Ser B, 2014, 106: 30–69
41 Ye Y. A 699-approximation algorithm for max-bisection. Math Program Ser A, 2001, 90: 101–111
42 Zhang J, Ye Y, Han Q. Improved Approximations for Max Set Splitting and Max NAE SAT. Disc Appl Math 2004, 142: 133–149
43 Zwick U. Approximation algorithms for constraint satisfaction problems involving at most three variables per constraint. In: Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms. Philadelphia: Society for Industrial and Applied Mathematics, 1998, 201–210
44 Zwick U. Outward rotations: A tool for rounding solutions of semidefinite programming relaxations, with applications to MAX CUT and other problems. In: Proceedings of the 31st Annual ACM Symposium on Theory of Computing. New York: ACM, 1999, 679–687

A Appendix: Proof of Lemma 3.4

We give the proof of Lemma 3.4 by using Lemma 3.3.

Proof of Lemma 3.4. First, we show that one can choose a minimum solution X to (3.4) (in Section 3) so that at most one entry of X belongs to $(0,1)$ and all entries of X in $[-1,0]$ are equal. Let $X=(X_{ij})$ be a minimum solution to (3.4). By Lemma 3.3, $d(x)$ is convex in $[-1,0]$ and concave in $[0,1]$.

Then we may choose a minimum solution X to (3.4) so that at most one entry of X belongs to $(0,1)$. Suppose $X_{ik}<X_{st}$ and $X_{ik},X_{st}\in(0,1)$. Let $\epsilon=\min\{1-X_{st},X_{ik}\}$. By Lemma 3.3, replacing X_{ik},X_{st} with $X_{ik}-\epsilon,X_{st}+\epsilon$, respectively, does not increase $\sum_{i<k\in S}d(X_{ik})$ (while maintaining the equation constraint). So we obtain a new minimum solution to (3.4) with fewer entries in $(0,1)$. Continuing this process, we arrive at the desired solution X.

We may further choose X so that all entries of X in $[-1,0]$ are equal. Let $X_{ik},X_{st}\in[-1,0]$ with $X_{ik}<X_{st}$. Now replacing X_{ik} and X_{st} with $(X_{ik}+X_{st})/2$, we obtain a minimum solution to (3.4) with smaller number of pairs of non-equal entries of X in $[-1,0]$. Continuing this process, we obtain the desired X.

If there is some entry $X_{pq}\in(0,1)$, we claim that all other entries of X must be in $\{-1,1\}$. Otherwise, let $X_{pq}\in(0,1)$ and assume that there are some entries of X in $(-1,0]$ (which must be equal by the choice of X above). Since X is a minimum solution to (3.4), every $X_{ik}\in(-1,1)$ must satisfy the Karush-Kuhn-Tucker condition: There exists a Lagrange multiplier t for the equality constraint $\lambda=\sum_{i<k\in S}$ in (3.4) such that every $X_{ik}\in(-1,1)$ satisfies

$$-2\theta/(\pi\sqrt{1-(\theta X_{ik}})=t.$$

So

$$X_{ik} = (\pm\sqrt{1-(2\theta/\pi t)^2})/\theta. \tag{A.1}$$

Now suppose $X_{st} \in (-1,0]$. Then $X_{st} < X_{pq}$ and both X_{st} and X_{pq} belong to $(-1,1)$. Therefore, we have $X_{st} = (-\sqrt{1-(2\theta/\pi t)^2})/\theta$, $X_{pq} = (\sqrt{1-(2\theta/\pi t)^2})/\theta$, and

$$1-(2\theta/\pi t)^2 > 0. \tag{A.2}$$

Then

$$\begin{aligned} d(X_{st})+d(X_{pq}) &= 1-(2/\pi)\arcsin(\theta X_{st})+1-(2/\pi)\arcsin(\theta X_{pq}) \\ &= 1-(2/\pi)\arcsin(-\theta\sqrt{1-(2\theta/\pi t)^2})+1-(2/\pi)\arcsin(\theta\sqrt{1-(2\theta/\pi t)^2}) \\ &= 2-2\cdot(2/\pi)\arcsin(0) \\ &= 2\cdot d(0). \end{aligned}$$

By replacing both X_{st} and X_{pq} with 0 and keeping all other entries of X unchanged, we obtain a new minimal solution with some entries in $(-1,1)$ not satisfying (A.1) and (A.2), a contradiction. Thus, if there is some entry $X_{pq} \in (0,1)$, then all other entries of X must be in $\{-1,1\}$.

The above properties of the minimum solution will help us calculate $h_1'(\lambda)$ (in (3.4)) and then obtain a lower bound on $\alpha_{|S|}$. Recall that

$$z'(\lambda^*) = \min\left\{1, \frac{N_{|S|}-\lambda^*}{2(|S|-1)}\right\}$$

from (3.3) in the proof of Lemma 3.1. So $z'(\lambda^*) = 1$ when $\lambda^* \in I_1 := [-\lfloor |S|/2\rfloor, \overline{N}]$, where $\overline{N} = (|S|-4)(|S|-1)/2$; and $z'(\lambda^*) = \frac{N_{|s|}-\lambda^*}{2(|S|-1)}$ when $\lambda^* \in I_2 := [\overline{N}, N_{|S|})$, where $N_{|S|} = |S|(|S|-1)/2$. To obtain a lower bound on $\alpha_{|S|}$, we next distinguish two cases.

Subcase 2.1. $\lambda^* \in I_1$. In this case we have $z'(\lambda^*) = 1$.

From the proof of Lemma 3.1, it follows

$$\begin{aligned} \widehat{f}^*/z_j^* &\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik},\lambda'^*=\sum_{i<k\in S}(X_{0i}X_{0k})} \alpha_{|S|}(\lambda^*,\lambda'^*) \\ &= \min_{\lambda^*=\sum_{i<k\in S} X_{ik},\lambda'^*=\sum_{i<k\in S}(X_{0i}X_{0k})} (\nu h_1(\lambda^*)+(1-\nu)h_2(\lambda'^*))/(2Lz'(\lambda^*)) \\ &= \min_{\lambda^*=\sum_{i<k\in S} X_{ik},\lambda'^*=\sum_{i<k\in S}(X_{0i}X_{0k})} (\nu h_1'(\lambda^*)+(1-\nu)h_2(\lambda'^*))(2L), \end{aligned}$$

where $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities.

From (3.4), we have

$$h_1'(\lambda^*) = \min \sum_{i<k\in S} d(X_{ik}), \tag{A.3}$$

where the minimum is taken over all X such that $\lambda^* = \sum_{i<k\in S} X_{ik}$, and $-1 \leqslant X_{ik} \leqslant 1$ for $1 \leqslant i < k \leqslant |S|$.

As above, we may choose a minimum solution X to (A.3) so that at most one entry of X belongs to $(0,1)$ and all entries of X in $[-1,0]$ are equal, and if some entry $X_{pq} \in (0,1)$ then all other entries of X are in $\{-1,1\}$. Thus, there exists a nonnegative integer $N_1 \in [0, N_{|S|}-1]$ such that

$$\lambda^* = N_1(-1) + X_{pq} + (N_{|S|}-N_1-1).$$

Note that $N_1 \neq N_{|S|}$ (because of X_{pq}). Since $X_{pq} \in (0,1)$,

$$\begin{aligned} &\lambda \in I(N_1) := I_1 \cap (N_1(-1)+0+(N_{|S|}-N_1-1), \\ &N_1(-1)+1+(N_{|S|}-N_1-1)) = I_1 \cap (N_{|S|}-2N_1-1, N_{|S|}-2N_1). \end{aligned}$$

By definitions of $h_1'(\lambda^*)$ and $d(x)$, we have

$$h_1'(\lambda^*) = N_{|S|} - N_1(2/\pi)\arcsin(\theta(-1)) - (N_{|S|}-N_1-1)(2/\pi)\arcsin(\theta)$$

$$
\begin{aligned}
&- (2/\pi)\arcsin(\theta X_{pq})\\
&= N_{|S|} + N_1 2/\pi \arcsin(\theta) - (N_{|S|} - N_1 - 1)2/\pi \arcsin(\theta)\\
&\quad - 2/\pi \arcsin(\theta(\lambda^* + N_1 - (N_{|S|} - N_1 - 1)))\\
&= f_1(\lambda^*, N_1).
\end{aligned}
$$

It is easily seen that

$$
\begin{aligned}
\widehat{f}^*/z_j^* &\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu h_1'(\lambda^*) + (1-\nu)h_2(\lambda'^*))/(2L)\\
&= \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu f_1(\lambda^*, N_1) + (1-\nu)h_2(\lambda'^*))/(2L)\\
&\geqslant \min_{\lambda\in I(N_1), \lambda'\in I'} \min_{\lambda=\sum_{i<k\in S} X_{ik}, \lambda'=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu f_1(\lambda, N_1) + (1-\nu)h_2(\lambda'))/(2L)\\
&\geqslant \frac{1}{2L} \min_{N\in[0, N_{|S|}-1]} \left\{ \min_{\lambda\in I(N), \lambda'\in I'} [\nu f_1(\lambda, N) + (1-\nu)h_2(\lambda')] \right\}\\
&= \frac{1}{2L} l_1,
\end{aligned}
$$

where

$$
\lambda = \sum_{i<k\in S} X_{ik}, \quad \lambda' = \sum_{i<k\in S} X_{0i}X_{0k},
$$

and $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities.

Now assume that there is no entry $X_{pq} \in (0,1)$. Then by the choice of X as above, except the entries which are equal to 1, all other entries of X are in $[-1,0]$ and equal. Let $X_{ik} \in [-1,0]$ be an arbitrary entry of X. Then there exists a nonnegative integer $N_1' \in [1, N_{|S|}]$ such that

$$
\lambda^* = N_1' X_{ik} + (N_{|S|} - N_1').
$$

Note that $N_1' \neq 0$, since $\lambda^* < N_{|S|}$. Since $X_{ik} \in [-1,0]$, we have

$$
\lambda^* \in J(N_1') := I_1 \cap [N_1'(-1) + (N_{|S|} - N_1'), N_{|S|} - N_1'] = I_1 \cap [N_{|S|} - 2N_1', N_{|S|} - N_1'].
$$

By definitions of $h_1'(\lambda^*)$ and $d(x)$, we find

$$
\begin{aligned}
h_1'(\lambda^*) &= N_{|S|} - N_1'(2/\pi)\arcsin(\theta X_{ik}) - (N_{|S|} - N_1')(2/\pi)\arcsin(\theta)\\
&= N_{|S|} - N_1'(2/\pi)\arcsin\left(\theta \frac{(\lambda^* - (N_{|S|} - N_3'))}{N_1'}\right)\\
&\quad - (N_{|S|} - N_1')(2/\pi)\arcsin(\theta)\\
&= f_2\left(\frac{\lambda^* - (N_{|S|} - N_1')}{N_1'}, N_1'\right).
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\widehat{f}^*/z_j^*\\
&\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu h_1'(\lambda^*) + (1-\nu)h_2(\lambda'^*))/(2L)\\
&= \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} \left(\nu f_2\left(\frac{\lambda^* - (N_{|S|} - N_1')}{N_1'}, N_1'\right) + (1-\nu)h_2(\lambda'^*)\right)/(2L)\\
&\geqslant \min_{\lambda\in J(N_1'), \lambda'\in I'} \min_{\lambda=\sum_{i<k\in S} X_{ik}, \lambda'=\sum_{i<k\in S} X_{0i}X_{0k}} \left(\nu f_2\left(\frac{\lambda - (N_{|S|} - N_1')}{N_1'}, N_1'\right) + (1-\nu)h_2(\lambda')\right)/(2L)\\
&\geqslant \min_{N\in[1, N_{|S|}]} \left\{ \min_{\lambda\in J(N), \lambda'\in I'} \left(\nu f_2\left(\frac{\lambda - (N_{|S|} - N)}{N}, N\right) + (1-\nu)h_2(\lambda')\right) \right\}/(2L)\\
&= \frac{1}{2L} l_2,
\end{aligned}
$$

where $\lambda = \sum_{i<k\in S} X_{ik}, \lambda' = \sum_{i<k\in S} X_{0i}X_{0k}$, and $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy triangle inequalities.

Subcase 2.2. $\lambda^* \in I_2$. In this case, we have

$$z'(\lambda^*) = \frac{N_{|s|} - \lambda^*}{2(|S|-1)}.$$

From the proof of Lemma 3.1, we have

$$\begin{aligned}
\widehat{f}^*/z_j^* &\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} \alpha_{|S|}(\lambda^*, \lambda'^*) \\
&= \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu h_1(\lambda^*) + (1-\nu)h_2(\lambda'^*))/(2Lz'(\lambda^*)) \\
&\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu h_1'(\lambda^*) + (1-\nu)h_2(\lambda'^*)) \frac{|S|-1}{L(N_{|S|}-\lambda^*)},
\end{aligned}$$

where $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities.

Similarly, we may choose a minimum solution X to (A.3) so that at most one entry of X belongs to $(0,1)$ and all entries of X in $[-1,0]$ are equal, and if some entry $X_{pq} \in (0,1)$ then all other entries of X are in $\{-1,1\}$. Thus in this case there exists a nonnegative integer $N_2 \in [0, N_{|S|}-1]$ such that

$$\lambda^* = N_2(-1) + X_{pq} + (N_{|S|} - N_2 - 1).$$

Note that $N_2 \neq N_{|S|}$ (because of X_{pq}). Since $X_{pq} \in (0,1)$, it follows $\lambda^* \in K(N_2) := I_2 \cap (N_2(-1) + N_{|S|} - N_2 - 1, N_2(-1) + 1 + N_{|S|} - N_2 - 1) = I_2 \cap (N_{|S|} - 2N_2 - 1, N_{|S|} - 2N_2)$. By definitions of $h_1'(\lambda^*)$ and $d(x)$, we have

$$\begin{aligned}
h_1'(\lambda^*) &= N_{|S|} - N_2(2/\pi)\arcsin(\theta(-1)) - (N_{|S|} - N_2 - 1)(2/\pi)\arcsin(\theta) \\
&\quad - (2/\pi)\arcsin(\theta X_{pq}) \\
&= N_{|S|} + N_2(2/\pi)\arcsin(\theta) - (N_{|S|} - N_2 - 1)(2/\pi)\arcsin(\theta) \\
&\quad - (2/\pi)\arcsin(\theta(\lambda^* + N_2 - (N_{|S|} - N_2 - 1))) \\
&= f_1(\lambda^*, N_2).
\end{aligned}$$

It is easily seen that

$$\begin{aligned}
\widehat{f}^*/z_j^* &\geqslant \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu h_1'(\lambda^*) + (1-\nu)h_2(\lambda'^*)) \frac{|S|-1}{L(N_{|S|}-\lambda^*)} \\
&= \min_{\lambda^*=\sum_{i<k\in S} X_{ik}, \lambda'^*=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu f_1(\lambda^*, N_2) + (1-\nu)h_2(\lambda'^*)) \frac{|S|-1}{L(N_{|S|}-\lambda^*)} \\
&\geqslant \min_{\lambda\in K(N_2), \lambda'\in I'} \min_{\lambda=\sum_{i<k\in S} X_{ik}, \lambda'=\sum_{i<k\in S} X_{0i}X_{0k}} (\nu f_1(\lambda, N_2) + (1-\nu)h_2(\lambda')) \frac{|S|-1}{L(N_{|S|}-\lambda)} \\
&\geqslant \frac{|S|-1}{L} \min_{N\in[0,N_{|S|}-1]} \left\{ \min_{\lambda\in K(N), \lambda'\in I'} \frac{\nu f_1(\lambda, N) + (1-\nu)h_2(\lambda')}{N_{|S|}-\lambda} \right\} \\
&= \frac{|S|-1}{L} l_3,
\end{aligned}$$

where $\lambda = \sum_{i<k\in S} X_{ik}, \lambda' = \sum_{i<k\in S} X_{0i}X_{0k}$, and $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities.

Now assume that there is no entry $X_{pq} \in (0,1)$. Then by the choice of X, except the entries which are equal to 1, all other entries of X are in $[-1,0]$ and equal. Let $X_{ik} \in [-1,0]$ be an arbitrary entry of X. Hence there exists a nonnegative integer $N_2' \in [1, N_{|S|}]$ such that

$$\lambda^* = N_2' X_{ik} + (N_{|S|} - N_2').$$

Note that $N_2' \neq 0$ because $\lambda^* < N_{|S|}$. Since $X_{ik} \in [-1,0]$, it follows $\lambda^* \in M(N_2') := I_2 \cap [N_2'(-1) + (N_{|S|} - N_2'), N_{|S|} - N_2'] = I_2 \cap [N_{|S|} - 2N_2', N_{|S|} - N_2']$. By definitions of $h_1'(\lambda^*)$ and $d(x)$, we have

$$h_1'(\lambda^*) = N_{|S|} - (N_{|S|} - N_2')(2/\pi)\arcsin(\theta) - N_2'(2/\pi)\arcsin(\theta X_{ik})$$

$$= N_{|S|} - (N_{|S|} - N_2')(2/\pi)\arcsin(\theta) - N_2'(2/\pi)\arcsin\left(\frac{\theta(\lambda^* - (N_{|S|} - N_2'))}{N_2'}\right)$$
$$= f_2\left(\frac{\lambda^* - (N_{|S|} - N_2')}{N_2'}, N_2'\right).$$

Hence,

$$\begin{aligned}
&\widehat{f}^*/z_j^*\\
&\geqslant \min_{\lambda^*=\sum_{i<k\in S}X_{ik},\lambda'^*=\sum_{i<k\in S}X_{0i}X_{0k}} (\nu h_1'(\lambda^*) + (1-\nu)h_2(\lambda'^*))\frac{|S|-1}{L(N_{|S|}-\lambda^*)}\\
&= \min_{\lambda^*=\sum_{i<k\in S}X_{ik},\lambda'^*=\sum_{i<k\in S}X_{0i}X_{0k}} \left(\nu f_2\left(\frac{\lambda^* - (N_{|S|} - N_2')}{N_2'}, N_2'\right) + (1-\nu)h_2(\lambda'^*)\right)\frac{|S|-1}{L(N_{|S|}-\lambda^*)}\\
&\geqslant \min_{\lambda\in M(N_2'),\lambda'\in I'}\ \min_{\lambda=\sum_{i<k\in S}X_{ik},\lambda'=\sum_{i<k\in S}X_{0i}X_{0k}} \left\{\nu f_2\left(\frac{\lambda - (N_{|S|} - N_2')}{N_2'}, N_2'\right)\right.\\
&\quad \left. + (1-\nu)h_2(\lambda')\right\}\frac{|S|-1}{L(N_{|S|}-\lambda)}\\
&\geqslant \frac{|S|-1}{L}\min_{N\in[1,N_{|S|}]}\left\{\min_{\lambda\in M(N),\lambda'\in I'} \frac{\nu f_2(\frac{\lambda-(N_{|S|}-N)}{N}\cdot N) + (1-\nu)h_2(\lambda')}{N_{|S|}-\lambda}\right\}\\
&= \frac{|S|-1}{L}l_4,
\end{aligned}$$

where $\lambda = \sum_{i<k\in S}X_{ik}, \lambda' = \sum_{i<k\in S}X_{0i}X_{0k}$, and $-1 \leqslant X_{ik}, X_{0i}, X_{0k} \leqslant 1$ satisfy the triangle inequalities.

Summing up, by the definition of $\alpha_{|S|}$, we have

$$\alpha_{|S|} \geqslant \min\left\{\frac{1}{2L}l_1, \frac{1}{2L}l_2, \frac{|S|-1}{L}l_3, \frac{|S|-1}{L}l_4\right\}. \qquad \square$$

Note that for any fixed $|S|$ and nonnegative integer N, the quantities l_1, l_2, l_3 and l_4 can be evaluated numerically. (For example, we used the optimization toolbox of Matlab.) It is easily seen that our scheme enumerates at most $O(|S|^2)$ nonnegative integers N with $N \leqslant N_{|S|}$.