Math. Program., Ser. B 97: 451–469 (2003)

Digital Object Identifier (DOI) 10.1007/s10107-003-0423-5

David Avis · Jun Umemoto

Stronger linear programming relaxations of max-cut

Received: April 30, 2002 / Accepted: November 26, 2002
Published online: May 7, 2003 –

Abstract. We consider linear programming relaxations for the max cut problem in graphs, based on k-gonal inequalities. We show that the integrality ratio for random dense graphs is asymptotically $1 + 1/k$ and for random sparse graphs is at least $1 + 3/k$. There are $O(n^k)$ k-gonal inequalities. These results generalize work by Poljak and Tuza, who gave similar results for $k = 3$.

1. Introduction

For a graph $G(V, E)$ with $|V| = n$ vertices, which we assume are labelled $1, 2, ..., n$, *max-cut* is the problem of partitioning V into two sets, such that the number of edges connecting the two sets is maximized. Although this problem is NP-hard, the well-known semidefinite programming (SDP) relaxation by Goemans and Williamson [10] gives a worst case bound on the integrality ratio of $sdp/opt \le 1.13823$, where sdp is the value of SDP and opt is the size of the maximum cut. In this paper, we consider linear programming (LP) relaxations of max-cut and discuss the integrality ratio lp/opt, where lp is the value of LP. A general reference for terminology not defined explicitly here is Deza and Laurent [8].

First we give a well known integer programming formulation of max-cut. Let $x = (x_{ij})_{1\le i<j\le n}$ and $c = (c_{ij})_{1\le i<j\le n}$ be vectors in $\mathbb{R}^{\binom{n}{2}}$. Given a graph G, for $1 \le i < j \le n$ we set $c_{ij} = 1$ if there is an edge joining vertices i and j, otherwise we set $c_{ij} = 0$. The max-cut problem can be formulated as the integer programming problem:

$$\begin{array}{lll} \max & c^T x & \\ \text{s.t.} & x_{ij} + x_{ik} + x_{jk} \le 2, & \\ & x_{ij} - x_{ik} - x_{jk} \le 0, & \text{for distinct } i, j, k \in \{1, \dots, n\} \\ & x_{ij} \in \{0, 1\} \text{ for } 1 \le i < j \le n. & \end{array} \tag{1}$$

It is not hard to show that the extreme points of the feasible region are the incidence vectors of cuts in the complete graph K_n. We can get a linear programming relaxation for max-cut by relaxing the constraint $x_{ij} \in \{0, 1\}$ to $x_{ij} \ge 0$. The integrality ratio for this

D. Avis: Computer Science, Mcgill University and GERAD, 3480 University, Montreal, Quebec, Canada H3A 2A7, e-mail: avis@cs.mcgill.ca

J. Umemoto: Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, Japan 606-8501, e-mail: umemoto@kuis.kyoto-u.ac.jp

relaxation has been well studied. Seymour [14] and Barahona and Mahjoub [3] show lp/opt=1 for graphs with no K_5 minor.

In this paper we will mainly be concerned with random graphs. In what follows, $G_{n,p}$ denotes a graph on n vertices, whose edges are chosen randomly and independently with probability p, $0 < p < 1$. We denote by $opt(G_{n,p})$ the cardinality of the max-cut for this graph. Poljak and Tuza [13] have obtained the following results for random graphs.

Theorem 1.1. *(Dense Graphs) Let $p = p(n)$ be a function such that $0 < p < 1$ and $p(n) = \Omega(\sqrt{\log n/n})$. Then the integrality ratio*

$$\frac{lp(G_{n,p})}{opt(G_{n,p})} \to \frac{4}{3}$$

as $n \to \infty$ with probability $1 - o(1)$. □

Theorem 1.2. *(Sparse Graphs) Let $p = p(n)$ be a function such that $0 < p < 1$, $p(n) \cdot n \to \infty$, and $p(n) \cdot n^{1-a} \to 0$ for every $a > 0$. Then the integrality ratio*

$$\frac{lp(G_{n,p})}{opt(G_{n,p})} \to 2$$

as $n \to \infty$, with probability $1 - o(1)$. □

This result shows that the linear programming relaxation behaves very badly for random sparse graphs: a simple greedy heuristic will also deliver a ratio bound of $greedy/opt \leq 2$. A much tighter relaxation is obtained by adding additional constraints known as k-gonal inequalities.

Definition 1.1. *Let $b_1, \ldots, b_n$ be a sequence of integers and let $k = \sum_{i=1}^{n} |b_i|$. The integers define the following k-gonal inequality:*

$$\sum_{1 \leq i < j \leq n} b_i b_j x_{ij} \leq \lfloor (\sum_{i=1}^{n} b_i)^2/4 \rfloor. \tag{2}$$

Definition 1.2. *The gap γ of an integer sequence $b_1, \ldots, b_n$ is defined by*

$$\gamma = \gamma(b) = min_{S \subseteq \{1,2,\ldots,n\}} |\sum_{i \in S} b_i - \sum_{i \notin S} b_i|. \tag{3}$$

A k-gonal inequality with gap $\gamma = 1$ is called hypermetric, and with gap $\gamma = 0$ is called negative type.

All k-gonal inequalities are valid for max-cut. Note in particular that the inequalities in (1) are 3-gonal with the b-vector a permutation of (1,1,1,0,...,0) and (1,1,-1,0,...,0) respectively. Since these b vectors have gap one, they are hypermetric. Laurent and Poljak ([11], see also [8] Section 28.4) have studied inequalities called *gap* inequalities, where the right hand side of (2) is strengthened to $\frac{1}{4}((\sum_{i=1}^{n} b_i)^2 - \gamma(b)^2)$, which are also valid for max-cut. These inequalities are stronger than the k-gonal inequalities if the gap is at least two. Since determining the gap is NP-hard, we do not consider these inequalities here.

Many results are known for hypermetric inequalities. In particular, for fixed n, Deza, Grishukhin and Laurent ([7], see also [8], Theorem 14.2.1) showed that there are only finitely many non-redundant hypermetric inequalities, so they form a polyhedron. This is not the case for negative type inequalities. It is unknown whether the k-gonal inequalities (2) form a polyhedron for fixed n. Deza ([6], see also [8], Prop. 6.1.3) has shown that for any integer $k \geq 1$, the $(2k+2)$-gonal negative type inequalities are implied by the $(2k+1)$-gonal hypermetric inequalities. In fact this is true without any assumptions on the gap, as we show in the next section. Hence in what follows we will assume that k is odd and so the right hand side of (2) can also be written $((\sum_{i=1}^{n} b_i)^2 - 1)/4$.

Definition 1.3. *We denote by LP_k the set of $x \in \mathbb{R}^{\binom{n}{2}}$ that satisfy all t-gonal inequalities, for $3 \leq t \leq k$. For $c \in \mathbb{Z}^{\binom{n}{2}}$, we denote by lp_k the value of max $c^T x, x \in LP_k$. For a graph G, $lp_k(G)$ denotes the value obtained when c is the 0-1 incidence vector of edges of G.*

The k-gonal inequalities have a close relation with SDP ([8], Section 28.4). Let X be the symmetric n by n matrix with zero diagonal and ijth off diagonal element equal to x_{ij}. Let J be the n by n matrix of all ones. Then $J - 2X$ is positive semidefinite if and only if x satisfies all the inequalities (2) when the right hand side is relaxed by dropping the floor function. (These are essentially the separation inequalities for SDP). The stronger $(2k+1)$-gonal inequalities are not valid for SDP and are natural candidates for an LP relaxation. This is especially true for the hypermetric inequalities, since many of them are facets of the cut polytope. Unfortunately separation for these inequalities seems hard in general. Avis and Grishukhin ([2], see also [8] Section 28.3) have shown that finding the smallest k such that x violates a k-gonal hypermetric inequality is NP-hard.

The purpose of this paper is generalize the results in Theorem 1.1 and 1.2 to LP relaxations using k-gonal inequalities:

Theorem 1.3. *(Dense Graphs) Let $p = p(n)$ be a function of n such that $0 < p < 1$ and $p(n)^{k-1} \cdot \frac{n}{\log n} \to \infty$*

$$\frac{lp_k(G_{n,p})}{opt(G_{n,p})} \to \frac{k+1}{k}$$

as $n \to \infty$ with probability $1 - o(1)$.

As will be clear from the proof, the result holds even when the linear programming relaxation is restricted to triangle inequalities and the k-gonal inequalities with b vector consisting only of zeroes and ones. We remark that Arora, Karger and Karpinski [1] have given a polynomial time approximation scheme for graphs with $\Omega(n^2)$ edges, using completely different methods.

Theorem 1.4. *(Sparse Graphs). Let $p = p(n)$ be a function such that $0 < p < 1$, $p(n) \cdot n \to \infty$, and $p(n) \cdot n^{1-a} \to 0$ for every $a > 0$. Then*

$$\frac{lp_k(G_{n,p})}{opt(G_{n,p})} \geq \frac{k+3}{k}$$

as $n \to \infty$, with probability $1 - o(1)$.

For the sparse graphs described in this theorem, we are unable to improve the trivial upper bound of two on the integrality ratio.

The final section of the paper gives some computational results. We wrote a program that computes LP relaxations of max-cut using k-gonal inequalities. Our program gives the maximum cut or an upper bound on its value by solving an LP relaxation. De Simone and Rinaldi [5] have also used these inequalities in developping an algorithm and a program to verify the optimality of a given cut. Their program checks the optimality of a cut generated by a heuristic, by solving a related LP problem using k-gonal inequalities.

2. Preliminary results

First we study the dual of an LP relaxation using k-gonal inequalities. Suppose that we use inequalities defined by vectors $b^{(1)}, \ldots, b^{(m)} \in \mathbb{Z}^n$. Let $d_l = \lfloor (\sum_{i=1}^n b_i^{(l)})^2/4 \rfloor$. Then, the primal problem is:

$$\begin{array}{lll} \max & \displaystyle\sum_{1 \le i < j \le n} c_{ij} x_{ij} & \\ \text{s.t.} & \displaystyle\sum_{1 \le i < j \le n} b_i^{(l)} b_j^{(l)} x_{ij} \le d_l, & 1 \le l \le m \\ & x_{ij} \ge 0 & 1 \le i < j \le n \end{array} \tag{4}$$

Let y_l be the dual variable corresponding to the inequality defined by $b^{(l)}$, the dual problem is:

$$\begin{array}{lll} \min & \displaystyle\sum_{l=1}^{m} d_l y_l & \\ \text{s.t.} & \displaystyle\sum_{l=1}^{m} b_i^{(l)} b_j^{(l)} y_l \ge c_{i,j} & 1 \le i < j \le n \\ & y_l \ge 0 & 1 \le l \le m \end{array} \tag{5}$$

We can give a graph theoretic interpretation of the dual. Suppose that each vector $b^{(l)}$ stands for a complete subgraph, or clique, where each edge ij has a weight $b_i^{(l)} b_j^{(l)}$. For example, in a five vertex graph, the vector $b = (1, 1, 1, 0, 0)$ stands for the triangle $(1, 2, 3)$ with every edge weight 1, and $b = (0, -1, 1, 1, 0)$ stands for the triangle $(2, 3, 4)$ with edges $(2, 3)$, $(2, 4)$ and $(3, 4)$ receiving weights -1, -1 and 1, respectively. Each inequality of the dual problem can be read as a constraint such that every edge weight c_{ij} must be covered by the sum of the weights $b_i^{(l)} b_j^{(l)}$ multiplied by y_l, where the sum is taken over all cliques. The right hand side value of a k-gonal inequality, d_l, can be read as the cost of the clique $b^{(l)}$. Hence, the dual problem is the problem of finding an edge covering of the given graph using weighted cliques $b^{(l)}$, while minimizing the sum of the cost of cliques used. We call this covering a *dual cover*. By weak duality, the cost of any dual cover is an upper bound on the cost of the primal problem.

For $k \geq 5$ the k-gonal inequalities are tighter than the triangle inequalities. The next two lemmas, which are interesting in their own right, quantify this. The first lemma shows how much a k-inequality needs to be weakened so that it is valid for all $x \in \mathrm{LP}_3$.

Lemma 2.1. *Let $b \in \mathbb{Z}^n$, $x \in \mathbb{R}^{\binom{n}{2}}$ and set $t = \sum_{i=1}^n b_i$, $k = \sum_{i=1}^n |b_i|$. Then the inequality*

$$\sum_{1 \leq i < j \leq n} b_i b_j x_{ij} \leq \frac{t^2 - 1}{4} + \frac{(k-1)(k-3)}{12} \tag{6}$$

is satisfied for all $x \in \mathrm{LP}_3$.

Proof. We consider 2 cases according to the b vector.

Case 1. Every element of b has a value 1 or -1. Hence $k = n$. Without losing generality, we may assume

$$b_1 = b_2 = \ldots = b_s = 1$$
$$b_{s+1} = b_{s+2} = \ldots = b_n = -1$$

where $t = s - (n - s) = 2s - n$. Since $x \in \mathrm{LP}_3$, for $1 \leq i < j \leq n$,

$$b_i b_j x_{ij} + b_i b_p x_{ip} + b_j b_p x_{jp} \leq \frac{(b_i + b_j + b_p)^2 - 1}{4}.$$

By summing this inequality for all triples i, j, p, noting that each pair i, j appears in $n - 2$ triples and the right hand side is 2 or 0, we have

$$(n-2) \sum_{1 \leq i < j \leq n} b_i b_j x_{ij} \leq 2 \left\{ \binom{s}{3} + \binom{n-s}{3} \right\}.$$

Expanding and dividing by $n - 2$ gives

$$\begin{aligned} \sum_{1 \leq i < j \leq n} b_i b_j x_{ij} &\leq \frac{2(n-2)(n^2 - n - 3ns + 3s^2)}{3 \cdot 2 \cdot (n-2)} \\ &= \frac{n^2 - n - 3ns + 3s^2}{3} \\ &= \frac{(2s-n)^2 - 1}{4} + \frac{n^2 - 4n + 3}{12} \end{aligned}$$

and (6) follows since $k = n$ and $t = 2s - n$.

Case 2. b is any integer vector. Without losing generality, we set

$$b_1 \geq b_2 \geq \ldots \geq b_n.$$

Furthermore, we may assume that no integer b_i is zero, for in this case the inequality (6) reduces to an equivalent inequality with n reduced by one. We define $k_0 = 0$ and for $v = 1, \ldots, n$ set

$$k_v = \sum_{i=1}^{v} |b_i|$$

$$\overline{b}_p = \begin{cases} 1 & b_v > 0 \\ -1 & b_v < 0 \end{cases} \quad \text{for } p = k_{v-1}+1, k_{v-1}+2, \ldots, k_v.$$

Let $\overline{x} \in \mathbb{R}^{\binom{k}{2}}$, and define

$$\overline{x}_{pq} = \begin{cases} x_{ij} & k_{i-1} < p \le k_i,\ k_{j-1} < q \le k_j \\ 0 & \text{otherwise.} \end{cases} \quad \text{for } 1 \le i < j \le n$$

It is easy to check that $x \in \mathrm{LP}_3$ if and only if $\overline{x} \in \mathrm{LP}_3$. Now for $1 \le i < j \le n$,

$$b_i b_j x_{ij} = \sum_{k_{i-1} < p \le k_i} \sum_{k_{j-1} < q \le k_j} \overline{b}_p \overline{b}_q \overline{x}_{pq}.$$

Summing over all i, j, we have

$$\begin{aligned} \sum_{i \le i < j \le n} b_i b_j x_{ij} &= \sum_{1 \le p < q \le n} \overline{b}_p \overline{b}_q \overline{x}_{pq} - \sum_{v=1}^{n} \sum_{k_{v-1} < p < q \le k_v} \overline{b}_p \overline{b}_q \overline{x}_{pq} \\ &= \sum_{1 \le p < q \le n} \overline{b}_p \overline{b}_q \overline{x}_{pq} \\ &\le \frac{t^2 - 1}{4} + \frac{(k-1)(k-3)}{12}, \end{aligned}$$

where we note $\overline{x}_{pq}$ is zero in the second term of the right hand side of the first equation and use case 1 for the final inequality. □

We now show that if $x' \in \mathrm{LP}_3$ we can move it slightly towards the barycentrum $x_0 = (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2})$ of the cut polytope so that it is inside LP_k, for any fixed $k \ge 3$.

Lemma 2.2. *For any $k \ge 3$ and $x' \in \mathrm{LP}_3$,*

$$x = \frac{3}{k} x' + \frac{k-3}{k} x_0 \in \mathrm{LP}_k.$$

Proof. Let $b_1, b_2, \ldots, b_n$ be integers with $t = \sum_{i=1}^{n} b_i$, and $k = \sum_{i=1}^{n} |b_i|$. We first observe that

$$2 \sum_{1 \le i < j \le n} b_i b_j = \left(\sum_{i=1}^{n} b_i \right)^2 - \sum_{i=1}^{n} b_i^2 \le t^2 - \sum_{i=1}^{n} |b_i|^2 \le t^2 - \sum_{i=1}^{n} |b_i| = t^2 - k.$$

where the last inequality is due to the integrality of b_i. Therefore

$$\begin{aligned}\sum_{1\le i<j\le n} b_i b_j x_{ij} &= \frac{3}{k}\sum_{1\le i<j\le n} b_i b_j {x'}_{ij} + \frac{k-3}{k}\sum_{1\le i<j\le n}\frac{b_i b_j}{2}\\ &\le \frac{3}{k}\left\{\frac{t^2-1}{4}+\frac{(k-1)(k-3)}{12}\right\}+\frac{k-3}{k}\left(\frac{t^2-k}{4}\right)\\ &= \left(\frac{3}{k}+\frac{k-3}{k}\right)\frac{t^2-1}{4}+\frac{3}{k}\frac{(k-1)(k-3)}{12}+\frac{k-3}{k}\frac{1-k}{4}\\ &= \frac{t^2-1}{4}. \end{aligned} \tag{7}$$

□

We can show integrality ratios for some classes of graphs using a dual cover and the above lemmas. The following proposition considers the complete graph.

Proposition 2.1. *For the complete graph K_n and any odd integer $k \ge 3$, the integrality ratio $\frac{lp_k(K_n)}{opt(K_n)} \to \frac{k+1}{k}$ as $n \to \infty$.*

Proof. If we apply Lemma 2.2 with $x' = (\frac{2}{3}, \frac{2}{3}, \dots, \frac{2}{3})$ we see that $x = (\frac{k+1}{2k}, \frac{k+1}{2k}, \dots, \frac{k+1}{2k}) \in \mathrm{LP}_k$ and gives a primal objective value for $lp_k(K_n)$ of

$$\sum_{1\le i<j\le n} c_{ij}x_{ij} = \frac{n(n-1)(k+1)}{4k}. \tag{8}$$

This is an optimal solution because we can get a feasible dual cover whose cost is (8) as follows. Subgraphs K_k of K_n are defined by vectors b such that $b \in \{0,1\}^{\binom{n}{2}}$ and $\sum_{i=1}^n b_i = \sum_{i=1}^n |b_i| = k$. There are $\binom{n}{k}$ K_k and each K_k has a dual cost $\frac{k^2-1}{4}$. Noting that each edge is contained in $\binom{n-2}{k-2}$ K_k, K_n has a dual cover where every subgraph K_k is weighted by $1/\binom{n-2}{k-2}$. The cost of this dual cover is

$$\frac{k^2-1}{4}\binom{n}{k}\Big/\binom{n-2}{k-2} = \frac{(k+1)n(n-1)}{4k}$$

which is the same as (8). If n is even the max-cut has size $\frac{n^2}{4}$ and so

$$\frac{lp_k(K_n)}{opt(K_n)} = \frac{(n-1)(k+1)}{nk} \to \frac{k+1}{k}$$

as $n \to \infty$. If n is odd the max-cut has size $\frac{n^2-1}{4}$ and so

$$\frac{lp_k(K_n)}{opt(K_n)} = \frac{n(k+1)}{(n+1)k} \to \frac{k+1}{k}$$

as $n \to \infty$. □

Finally we show that Deza's observation [6] that the hypermetric inequalities imply the negative type inequalities, holds for k-gonal inequalities of arbitrary gap.

Proposition 2.2. *Let $k \geq 1$ be an integer. The $(2k+2)$-gonal inequalities are implied by the $(2k+1)$-gonal inequalities.*

Proof. Let $b \in \mathbb{Z}^n$ with $\sum_{i=1}^n b_i = 2t$ and $\sum_{i=1}^n |b_i| = 2k+2$. Then, the $(2k+2)$-gonal inequality defined by b is

$$\sum_{1 \leq i < j \leq n} b_i b_j x_{ij} \leq t^2. \tag{9}$$

Without loss of generality we may assume that b_i is non-zero, $i = 1, 2, ..., n$. We show that the $(2k+2)$-gonal inequality (9) can be expressed as a nonnegative linear combination of $(2k+1)$-gonal inequalities. For each $p = 1, \ldots, n$, we define a vector $b^{(p)} \in \mathbb{Z}^n$:

$$b_i^{(p)} = \begin{cases} b_i & i \neq p \\ b_i - 1 & i = p, b_p > 0 \\ b_i + 1 & i = p, b_p \leq 0 \end{cases}.$$

Hence,

$$\sum_{i=1}^n |b_i^{(p)}| = 2k+1$$
$$\sum_{i=1}^n b_i^{(p)} = \begin{cases} 2t-1 & b_p > 0 \\ 2t+1 & b_p \leq 0 \end{cases}$$

and $b^{(p)}$ defines a $(2k+1)$-gonal inequality. The left hand side of this inequality is

$$\sum_{1 \leq i < j \leq n} b_i^{(p)} b_j^{(p)} x_{ij} = \sum_{1 \leq i < j \leq n} b_i b_j x_{ij} \mp \left(\sum_{i=1}^{p-1} b_i x_{ip} + \sum_{i=p+1}^{n} b_i x_{pi} \right),$$

where $\mp$ is interpreted as a minus sign if $b_p > 0$, and a plus sign otherwise. Multiplying by $|b_p|$, the $(2k+1)$-gonal inequalities become

$$|b_p| \sum_{1 \leq i < j \leq n} b_i b_j x_{ij} - \left(\sum_{i=1}^{p-1} b_i b_p x_{ip} + \sum_{i=p+1}^{n} b_p b_i x_{pi} \right) \leq |b_p| t(t \mp 1).$$

Summing up these inequalities, the we obtain an inequality with

$$
\begin{aligned}
\text{(LHS)} &= \sum_{p=1}^{n} |b_p| \sum_{1\le i<j\le n} b_i b_j x_{ij} - \sum_{p=1}^{n} \left(\sum_{i=1}^{p-1} b_i b_p x_{ip} + \sum_{i=p+1}^{n} b_p b_i x_{pi} \right) \\
&= (2k+2) \sum_{1\le i<j\le n} b_i b_j x_{ij} - 2 \sum_{1\le i<j\le n} b_i b_j x_{ij} \\
&= 2k \sum_{1\le i<j\le n} b_i b_j x_{ij} \qquad (10) \\
\text{(RHS)} &= t(t-1) \sum_{b_p>0} |b_p| + t(t+1) \sum_{b_p\le 0} |b_p|.
\end{aligned}
$$

For the right hand side,

$$
\sum_{b_p\le 0} |b_p| = \frac{\sum_{i=1}^{n} |b_i| - \sum_{i=1}^{n} b_i}{2} = \frac{2k+2-2t}{2} = k+1-t
$$

$$
\sum_{b_p>0} |b_p| = \sum_{i=1}^{n} |b_i| - \sum_{b_p\le 0} |b_p| = k+1+t.
$$

Therefore,

$$
\begin{aligned}
\text{(RHS)} &= t(t-1)(k+1+t) + t(t+1)(k+1-t) \\
&= t(tk+t^2+t-k-t-1+tk-t^2+t+k-t+1) \\
&= 2kt^2. \qquad (11)
\end{aligned}
$$

Dividing (10) and (11) by $2k$, we have (9). □

3. Sparse graphs

In this section we prove Theorem 1.4 giving a lower bound on the integrality ration for sparse graphs. We require the following strong bound on the value of the max-cut in a random graph.

Lemma 3.1 (Nguyen Van Ngoc and Zs. Tuza, [15]). *Let $p(n) \cdot n \to \infty$. Then*

$$
\frac{1}{2}|E| \le opt(G_{n,p}) \le |E| \left(\frac{1}{2} + o(1) \right)
$$

with probability $1 - o(1)$ *as* $n \to \infty$. □

Proof of Theorem 1.4. Let $c \in \{0,1\}^{\binom{n}{2}}$ be the incidence vector of the edges of the randomly chosen graph $G_{n,p}$. By Theorem 1.2 and Lemma 3.1,

$$
lp_3(G_{n,p}) = (1 - o(1))|E|
$$

with probability $1 - o(1)$ as $n \to \infty$. For any fixed $\epsilon > 0$, if n is large enough, with probability $1 - o(1)$ there is a primal solution x' for which the primal objective function is at least $(1-\epsilon)|E|$. We construct $x \in \mathrm{LP}_k$ using Lemma 2.2. Then

$$\begin{aligned} lp_k(G_{n,p}) \geq c^T x &= \frac{3}{k} c^T x' + \frac{k-3}{k} c^T x_0 \\ &\geq \frac{3}{k}|E|(1-\epsilon) + \frac{k-3}{k}\frac{|E|}{2} \geq \frac{k+3}{2k}|E|(1-\epsilon). \end{aligned}$$

Since ϵ is arbitrary, we can combine this with Lemma 3.1 to obtain

$$\frac{lp_k(G_{n,p})}{opt(G_{n,p})} \geq \frac{k+3}{k}(1 - o(1))$$

as $n \to \infty$, with probability $1 - o(1)$. □

4. Dense graphs

In this section we prove that random dense graphs have asymptotically the same integrality ratio as complete graphs. Our proof is modelled along the lines of that of Poljak and Tuza [13] for Theorem 1.1.

Lemma 4.1 (Chernoff Inequality [4]). *Let $X_1, X_2, \ldots, X_n$ be independent Bernoulli trials with* $\Pr[X_i = 1] = p$, $0 < p < 1$. *Then if X is the sum of the X_i and if μ is $E[X]$, for any $\epsilon > 0$,*

$$\Pr[|\mu - X| > n\epsilon] < e^{-\frac{n\epsilon^2}{2p(1-p)}}. \qquad \square$$

Definition 4.1 (Uniform Cover). *For any odd integer $k \geq 3$, let $\mathcal{K}_k = \mathcal{K}_k(G)$ denote the set of all k-cliques, K_k, in a graph G. A function $r : \mathcal{K}_k \cup E \to \mathbb{R}_+$ is called a uniform cover of G by K_k and edges if*

$$r(e) + \sum_{K_k \ni e} r(K_k) = 1$$

for every edge $e \in E$, where the sum is taken over all K_k containing the edge e. The cost of a uniform cover r is defined by

$$cost(r) := \sum_{e \in E} r(e) + \frac{k^2-1}{4} \sum_{K_k \in \mathcal{K}_k} r(K_k).$$

Lemma 4.2. *For any odd integer $k \geq 3$ and graph G, a uniform cover r yields the upper bound $lp_k(G) \leq cost(r)$.*

Proof. Consider the primal and dual problems given by (4) and (5), where the primal constraints are all t-gonal inequalities, $1 \leq t \leq k$. Given a uniform cover r, we show how to construct a dual feasible solution y. Initially all components of y are set to zero.

Fix a complete subgraph K_k of G. It corresponds to a k-gonal inequality defined by a vector $b^{(K_k)}$ such that

$$b_i^{(K_k)} = \begin{cases} 1 & i \in K_k \\ 0 & \text{otherwise.} \end{cases}$$

with right hand side $\frac{k^2-1}{4}$. We continue the construction of our dual solution by setting $y_{K_k} = r(K_k)$, contributing a cost of $r(K_k)\frac{k^2-1}{4}$ to the dual objective function.

Now fix any edge e of G, and suppose it joins vertices u and v. We consider two 3-gonal inequalities $b^{(s)}$ and $b^{(t)}$ for u, v, and any $w \in V \backslash u, v$:

$$b_i^{(s)} = \begin{cases} 1 & i = u, v, \text{ or } w \\ 0 & \text{otherwise} \end{cases} \quad \text{and } b_i^{(t)} = \begin{cases} 1 & i = u \text{ or } v \\ -1 & i = w \\ 0 & \text{otherwise.} \end{cases}$$

We observe that by dividing each inequality by two and adding them together, we have the inequality

$$x_{uv} = \frac{1}{2} \sum_{1 \le i < j \le n} (b_i^{(s)} b_j^{(s)} + b_i^{(t)} b_j^{(t)}) x_{ij} \le \frac{1}{2}(2 + 0) = 1.$$

We continue the construction of our dual solution by setting $y_s = y_t = r(e)/2$. This contributes a cost of $2y_s + 0y_t = r(e)$ to the dual objective value. The first equation in Definition 4.1 guarantees that the dual variables are feasible for the dual, and the second equation guarantees that it has dual objective value $cost(r)$. The result follows from the weak duality theorem of linear programming. □

Lemma 4.3. *Let X_k denote the vertices $\{x_1, \dots, x_k\}$ of a k-clique in $G_{n,p}$, and $g(X_k)$ denote the number of $k+1$-cliques which contain X_k. Then, the next inequality holds with probability $1 - o(1)$ as $n \to \infty$:*

$$p^k(n-k) - C\sqrt{p^k(n-k)\log n} < g(X_k) < p^k(n-k) + C\sqrt{p^k(n-k)\log n}$$

with a constant $C > 2$.

Proof. For every $x \in V \backslash X_k$, let E_x denote the event

$$E_x := \{\text{the vertices } \{x, x_1, \cdots, x_k\} \text{ form a } k+1-clique\}.$$

Then $Pr[E_x] = p^k$. Moreover, for fixed X_k, the events E_x are independent. Therefore, we can apply the Chernoff inequality. The expected number $\mu_{E_x} = E[g(X_k)]$ is equal to $p^k(n-k)$. Using the Chernoff inequality,

$$\Pr\left[|\mu_{E_x} - g(X_k)| > (n-k)\epsilon\right] < e^{-\frac{(n-k)\epsilon^2}{2p^k(1-p^k)}} \tag{12}$$

for any $\epsilon > 0$. Now let $\epsilon = \alpha\sqrt{\frac{p^k(1-p^k)}{n-k}}$ for any $\alpha > 0$, the inequality (12) becomes

$$\Pr\left[|\mu_{E_x} - g(X_k)| > \alpha\sqrt{\mu_{E_x}(1-p^k)}\right] < e^{-\frac{\alpha^2}{2}}.$$

For $\alpha = C\sqrt{\log n}$ with a constant $C > 2$, the right hand side is

$$e^{-\frac{\alpha^2}{2}} = n^{-\frac{C^2}{2}} = o(n^{-2}).$$

Therefore, $|\mu_{E_x} - g(X_k)| \le C\sqrt{\mu_{E_x}(1-p^k)\log n}$ holds with probability $1 - o(1)$. Since $\sqrt{\mu_{E_x}(1-p^k)\log n} < \sqrt{\mu_{E_x}\log n}$,

$$p^k(n-k) - C\sqrt{p^k(n-k)\log n} < g(X_k) < p^k(n-k) + C\sqrt{p^k(n-k)\log n}$$

holds for all sets X_k with probability $1 - o(1)$. □

In the next lemma we make use of the following function.

Definition 4.2. *Let* $n > t \ge 3$ *be integers, and let* $0 < p < 1$. *We define* $g(n, p, t)$ *by:*

$$g(n, p, t) = p^{\frac{(t-1)^2}{2}-1}\binom{n-2}{t-3}\sqrt{n\log n} \tag{13}$$

By straightforward calculation we have

$$g(n, p, t+1) = p^{t-\frac{1}{2}}\frac{n-t+1}{t-2}g(n, p, t) \ge p^{t-\frac{1}{2}}\frac{n-t}{t-1}g(n, p, t) \ge p^{t}\frac{n-t}{t-1}g(n, p, t) \tag{14}$$

Lemma 4.4. *Let* $k \ge 3$ *be an integer and* $p = p(n)$ *be such that* $0 < p < 1$ *and* $p(n)^{k-1} \cdot \frac{n}{\log n} \to \infty$. *For each edge* uv *of* $G_{n,p}$, *we let* $f_k(uv)$ *denote the number of* k*-cliques which contain* u *and* v. *Then with probability* $1 - o(1)$ *as* $n \to \infty$ *we have for all edges* uv

$$p^{\binom{k}{2}-1}\binom{n-2}{k-2} - C_k g(n, p, k) < f_k(uv) < p^{\binom{k}{2}-1}\binom{n-2}{k-2} + C_k g(n, p, k) \tag{15}$$

for some constant $C_k > 0$.

Proof. For $k = 3$, we get (15) by setting $k = 2$ in Lemma 4.3. We proceed by induction on k.

Suppose that (15) holds when $k = t \ge 3$. Fix and edge uv of $G_{n,p}$. The number of K_{t+1} which contain a given K_t is bounded by Lemma 4.3. Each K_{t+1} containing uv contains $(t-1)$ K_t that contain uv. Therefore, the number of K_{t+1} containing uv can be bounded as follows with probability $1 - o(1)$.

$$
\begin{aligned}
f_{t+1}(uv) &< \frac{1}{t-1}\cdot\left(p^t(n-t)+C\sqrt{p^t(n-t)\log n}\right)\\
&\quad\times\left(p^{\binom{t}{2}-1}\binom{n-2}{t-2}+C_t g(n,p,t)\right)\\
&= p^{\binom{t+1}{2}-1}\binom{n-2}{t-1}+C_t\frac{n-t}{t-1}p^t g(n,p,t)\\
&\quad+\frac{C}{t-1}p^{\frac{t^2}{2}-1}\binom{n-2}{t-2}\sqrt{(n-t)\log n}\\
&\quad+\frac{CC_t}{t-1}\sqrt{p^t(n-t)\log n}\,g(n,p,t)\\
&\le p^{\binom{t+1}{2}-1}\binom{n-2}{t-1}+g(n,p,t+1)\\
&\quad\times\left(C_t+\frac{C}{t-1}+\frac{CC_t}{t-1}\frac{t-1}{n-t}\sqrt{\frac{(n-t)\log n}{p^{t-1}}}\right)
\end{aligned}
$$

The three coefficients of $g(n, p, t+1)$ in the last line are obtained as follows: the first from the last inequality in (14), the second from the definition of $g(n, p, t+1)$, and the third from first inequality in (14). To complete the proof of the upper bound on $f_{t+1}(uv)$, we will set $C_{t+1} = C_t + \frac{C}{t-1}$ and show that the third coefficient vanishes. Indeed, simplifying we have

$$
CC_t\sqrt{\frac{\log n}{(n-t)p^{t-1}}}\to 0
$$

as $n \to \infty$ since t, C, C_t are constants, and $p(n)^{t-1}\cdot\frac{n}{\log n}\to\infty$. Thus the upper bound in (15) holds for $k = t+1$, and the proof is complete.

Similarly, we can show

$$
f_k(uv) > p^{\binom{k}{2}-1}\binom{n-2}{k-2} - C_k g(n,p,k) \qquad \square
$$

Lemma 4.5. *Let $p = p(n)$ be such that $0 < p < 1$ and $p(n)^{k-1}\cdot\frac{n}{\log n}\to\infty$ hold, where k is odd and $k \ge 3$. Then, with probability $1 - o(1)$ as $n \to \infty$, there exists a uniform cover r of $G_{n,p}$ such that*

$$
cost(r) \le \left(\frac{k+1}{2k}+o(1)\right)|E|. \tag{16}
$$

Proof. By Lemma 4.4, with probability $1 - o(1)$ as $n \to \infty$, $f_k(uv)$ satisfies (15) for each edge uv of $G_{n,p}$. Let P denote $p^{\binom{k}{2}-1}\binom{n-2}{k-2}$, let Q denote $p^{\frac{(k-1)^2}{2}-1}\binom{n-2}{k-3}\sqrt{n\log n}$, and let C_k be the constant from Lemma 4.4. We define a uniform cover r by

$$
r(K_k) = \frac{1}{P + C_k Q} \qquad \text{for every } K_k \in \mathcal{K}_k
$$

and

$$r(e) = 1 - \sum_{K_k \ni e} r(K_k) \quad \text{for every } e \in E.$$

Clearly, $r(e) \geq 0$ by (15), and hence r is a uniform cover. We estimate the cost of r. Since every edge belongs to at most $P + C_k Q$ k-cliques by (15), the total number of K_k is at most $\frac{1}{\binom{k}{2}}|E|(P + C_k Q)$. Hence,

$$\frac{k^2-1}{4} \sum_{K_k \in \mathcal{K}_k} r(K_k) \leq \frac{k^2-1}{4} \frac{1}{\binom{k}{2}} |E|(P + C_k Q) \frac{1}{P + C_k Q} = \frac{k+1}{2k}|E|$$

On the other hand, since every edge belongs to at least $P - C_k Q$ of K_k again by (15), we have

$$r(e) \leq 1 - \frac{P - C_k Q}{P + C_k Q} = \frac{2C_k Q}{P + C_k Q} = \frac{2C_k p^{\frac{(k-1)^2}{2}-1}\binom{n-2}{k-3}\sqrt{n \log n}}{p^{\binom{k}{2}-1}\binom{n-2}{k-2} + C_k p^{\frac{(k-1)^2}{2}-1}\binom{n-2}{k-3}\sqrt{n \log n}}$$
$$= \frac{2C_k \sqrt{n \log n}}{p^{\frac{k-1}{2}} \frac{n-k+1}{k-2} + C_k \sqrt{n \log n}} = \frac{2C_k}{\frac{n-k+1}{k-2} \frac{1}{n} p^{\frac{k-1}{2}} \left(\frac{n}{\log n}\right)^{\frac{1}{2}} + C_k} = o(1),$$

since $p^{\frac{k-1}{2}} \left(\frac{n}{\log n}\right)^{\frac{1}{2}} \to \infty$. Hence

$$\text{cost}(r) = \sum_{e \in E} r(e) + \frac{k^2-1}{4} \sum_{K_k \in \mathcal{K}_k} r(K_k) = o(1)|E| + \frac{k+1}{2k}|E| = \left(\frac{k+1}{2k} + o(1)\right)|E|$$

□

Proof of Theorem 1.3. Let r be the uniform cover of $G_{n,p}$ constructed in Lemma 4.5. By Lemma 4.2, r can be used to define a dual feasible solution to the linear programming relaxation. By the weak duality theorem, $lp_k(G_{n,p}) \leq \text{cost}(r)$. In the proof of Proposition 2.1 we saw that $x = (\frac{k+1}{2k}, \frac{k+1}{2k}, \ldots, \frac{k+1}{2k}) \in \text{LP}_k$. Therefore with probability $1 - o(1)$ as $n \to \infty$,

$$\frac{k+1}{2k}|E| \leq lp_k(G_{n,p}) \leq cost(r) = \left(\frac{k+1}{2k} + o(1)\right)|E|.$$

On the other hand by Lemma 3.1, we have under the same conditions,

$$\frac{1}{2}|E| \leq opt(G_{n,p}) \leq \left(\frac{1}{2} + o(1)\right)|E|.$$

Hence with probability $1 - o(1)$

$$\frac{lp_k(G_{n,p})}{opt(G_{n,p})} \to \frac{k+1}{k} \text{ as } n \to \infty.$$

□

5. Experimental results

In this section we give some experimental results. A complete description of the program is contained in [16]. We implemented a program in C that solves the dual problem by the column generation technique. The columns are generated by all permutations of the vectors $b^{(1)} = (1, 1, \pm 1)$, $b^{(2)} = (1, 1, 1, \pm 1, \pm 1)$, $b^{(3)} = (2, 1, \pm 1, \pm 1, \pm 1, \pm 1)$, and $b^{(4)} = (1, 1, 1, 1, \pm 1, \pm 1, \pm 1)$. These vectors correspond to all 3, 5 and 7-gonal facets of the cut polytope, see [8]. We know of no method to find a violated k-gonal inequality in less than $O(n^k)$ time in the worst case. Heuristic ordering of the inequalities can speed this up in practice, and we used a greedy heuristic for this.

Let $b^{(k)}$ be one of the input vectors corresponding to a k-gonal inequality, and y be the current dual solution. A column $b^{(k.l)}$ which is lth permutation of $b^{(k)}$ can be the entering column, if

$$\sum_{1\le i<j\le n} b_i^{(k.l)} b_j^{(k.l)} y_{ij} > \lfloor \frac{(\sum_{i=1}^{n} b_i^{(j)})^2}{4} \rfloor. \tag{17}$$

Assume that we know a permutation of $b^{(k)}$ which maximizes the left hand side of (17). Then, if (17) is satisfied, this column can be the entering column, otherwise, no column corresponding to $b^{(k)}$ can be the entering column. The idea is find a cut in the graph with maximum number of edges between vertices assigned opposite signs by the vector $b^{(k)}$. Since this is a hard problem we use a greedy algorithm for max cut which gives an approximate maximum for the left hand side of (17).

We carried out three types of computational experiments: for dense random graphs, for sparse random graphs, and for all graphs for $n = 5, \dots, 10$. We also tested Fujisawa's SDP program *SDPA (SemiDefinite Programming Algorithm)* [9] for comparison. The following is the meaning of each column of the tables:

G The type of input graph and method of solution. The description "$graph$-(k|sdp)" in column G means that the row is the result of solving $graph$ by LP relaxation using 3, ... , k-gonal inequalities or SDPA.

time The average running time (in seconds).

a.ratio The average integrality ratio.

w.ratio The worst integrality ratio.

p The fraction of problems where the maximum cut was obtained.

iter The average number of iterations of the simplex method.

(3|5|7)**gon** The number of (3|5|7)-gonal inequalities actually used in the simplex method respectively.

greedy The success ratio of the greedy heuristic We let the greedy heuristic run up to 10 times for each inequality type, and it is counted as a failure when the greedy heuristic has failed for all inequality types.

Table 1 gives results for dense random graphs. The description "n-p-method" in column G means that the instances are 2-connected graphs of n vertices and edge probability p. The number of the instances is 100 for each graph type except "30-0.9-7". Only 10 instances have been tested for "30-0.9-7" because the running time was quite long. Table 2 gives results for random sparse graphs. The description "n-m-method" in

Table 1. Dense random graphs

G	time	a.ratio	w.ratio	p	iter	3gon	5gon	7gon	greedy
20-0.5-3	0.31	1.00694	1.05556	0.57	1124.3	301.9	–	–	0.91
-5	0.83	1.00001	1.00051	0.99	2091.7	363.7	34.5	–	0.99
-7	0.78	1	1	1	2096.0	363.6	34.1	1.8	0.99
-sdp	0.03	1.01656	1.27491	–	–	–	–	–	–
-0.7-3	0.57	1.07036	1.10833	0.01	1661.9	369.3	–	–	0.74
-5	6.91	1.00069	1.0122	0.85	11117.0	726.0	189.0	–	0.98
-7	21.79	1	1.00023	0.99	11674.6	725.0	193.7	19.0	0.99
-sdp	0.03	1.02029	1.0366	–	–	–	–	–	–
-0.9-3	0.62	1.16359	1.21333	0.00	2094.6	313.5	–	–	0.89
-5	20.80	1.04732	1.092	0.00	22670.2	975.1	225.6	–	0.99
-7	1720.12	1.00858	1.04	0.13	102681.0	1209.1	2871.6	1297.2	0.96
-sdp	0.03	1.00666	1.01834	–	–	–	–	–	–
25-0.5-3	0.77	1.00331	1.03271	0.69	2032.0	464.7	–	–	0.93
-5	3.05	1	1	1	3803.5	540.0	60.1	–	0.99
-7	2.67	1	1	1	3855.0	537.0	56.1	3.2	1
-sdp	0.05	1.00932	1.2872	–	–	–	–	–	–
-0.7-3	2.25	1.0926	1.125	0.00	3889.4	662.4	–	–	0.67
-5	123.48	1.00238	1.01938	0.57	46059.1	1515.4	606.7	–	0.95
-7	367.19	1.00003	1.00307	0.99	51666.3	1521.1	665.1	103.1	0.99
-sdp	0.06	1.02087	1.08009	–	–	–	–	–	–
-0.9-3	2.34	1.18181	1.25	0.00	4916.7	516.8	–	–	0.84
-5	191.85	1.06363	1.125	0.00	65018.2	1884.4	352.0	–	0.99
-7	20499.7	1.01627	1.07143	0.03	392378.0	2350.9	9127.3	3319.2	0.97
-sdp	0.06	1.00817	1.03876	–	–	–	–	–	–
30-0.5-3	6.41	1.03532	1.07383	0.01	6518.6	1257.9	–	–	0.73
-5	112.73	1.00001	1.0014	0.99	41699.0	2312.9	343.7	–	0.99
-7	112.14	1	1	1	41226.4	2284.4	330.8	10.0	0.99
-sdp	0.08	1.04107	1.1732	–	–	–	–	–	–
-0.7-3	7.28	1.1097	1.13726	0.00	7598.4	1037.6	–	–	0.63
-5	870	1.00548	1.02384	0.20	135733.0	2746.0	1409.6	–	0.92
-7	9002.9	1.00007	1.00241	0.94	168668.0	2780.3	1800.3	428.2	0.98
-sdp	0.09	1.0035	1.04996	–	–	–	–	–	–
-0.9-3	8.05	1.19588	1.21922	0.00	9729.8	769.0	–	–	0.81
-5	1219.44	1.07629	1.0973	0.00	153730.0	3343.0	519.5	–	0.99
-7	102828	1.02311	1.04505	0.00	1017640.0	4075.6	21035	5925.8	0.98
-sdp	0.09	1.00816	1.01362	–	–	–	–	–	–

Table 2. Sparse random graphs

G	time	a.ratio	w.ratio	p	iter	3gon	5gon	7gon	greedy
20-30-3	0.08	1	1	1	378.2	206.7	–	–	0.98
-5	0.10	1	1	1	443.1	218.2	3.6	–	0.99
-7	0.10	1	1	1	449.3	219.5	3.5	0.2	1
-sdp	0.03	1.0392	1.07966	–	–	–	–	–	–
-40-3	0.11	1.0001	1.01042	0.99	480.0	227.6	–	–	0.98
-5	0.12	1	1	1	535.1	231.8	4.2	–	0.99
-7	0.12	1	1	1	531.5	231.8	4.2	0.2	1
-sdp	0.03	1.04086	1.07922	–	–	–	–	–	–
30-45-3	0.70	1	1	1	1371.7	565.6	–	–	0.98
-5	0.79	1	1	1	1625.6	579.2	10.1	–	0.99
-7	0.76	1	1	1	1580.7	575.6	9.8	0.9	1
-sdp	0.09	1.04094	1.07668	–	–	–	–	–	–
-60-3	0.94	1.00021	1.01418	0.98	1850.2	614.5	–	–	0.97
-5	1.35	1	1	1	2394.9	658.0	16.8	–	0.99
-7	1.31	1	1	1	2394.1	651.4	16.0	1.6	1
-sdp	0.09	1.04289	1.06869	–	–	–	–	–	–
40-60-3	4.41	1	1	1	3627.7	1118.1	–	–	0.98
-5	5.69	1	1	1	4418.4	1164.9	18.2	–	0.99
-7	6.32	1	1	1	4717.7	1175.8	18.5	3.4	0.99
-sdp	0.19	1.04107	1.0673	–	–	–	–	–	–
-80-3	6.31	1.00005	1.00513	0.99	5132.8	1216.0	–	–	0.97
-5	8.38	1	1	1	6556.2	1295.1	24.1	–	0.99
-7	8.75	1	1	1	7005.2	1295.0	24.5	5.1	0.99
-sdp	0.18	1.04357	1.06838	–	–	–	–	–	–
50-75-3	21.92	1.0001	1.01026	0.99	8167.2	1923.5	–	–	0.98
-5	36.95	1	1	1	11162.5	2079.3	33.1	–	0.99
-7	39.01	1	1	1	12169.0	2058.1	32.8	9.4	0.99
-sdp	0.37	1.04394	1.067	–	–	–	–	–	–
-100-3	42.98	1.00011	1.00406	0.97	13822.6	2197.7	–	–	0.98
-5	101.11	1	1	1	19615.7	2327.6	78.6	–	0.99
-7	125.96	1	1	1	22406.2	2370.4	112.3	11.1	0.99
-sdp	0.37	1.04671	1.07381	–	–	–	–	–	–

Table 3. All graphs for $n = 5, \ldots, 10$

G	time	a.ratio	w.ratio	p	iter	3gon	5gon	7gon	greedy
all5-3	0.0082	1.0101	1.11111	0.91	11.8	8.4	–	–	0.99
-5	0.0073	1	1	1	11.9	8.4	0.1	–	1
-sdp	0.0043	1.04583	1.13064	–	–	–	–	–	–
all6-3	0.0077	1.00243	1.11111	0.97	20.3	13.8	–	–	0.99
-5	0.0069	1	1	1	20.6	13.9	0.1	–	1
-sdp	0.0026	1.02866	1.125	–	–	–	–	–	–
all7-3	0.0087	1.00192	1.16667	0.96	30.7	20.2	–	–	0.99
-5	0.0097	1.00026	1.05	0.99	31.0	20.2	0.1	–	0.99
-7	0.0092	1.00007	1.03333	0.99	31.1	20.2	0.1	0.0	0.99
-sdp	0.0038	1.02962	1.12876	–	–	–	–	–	–
all8-3	0.0107	1.00102	1.16667	0.97	45.5	28.6	–	–	0.99
-5	0.0103	1.00003	1.05	0.99	45.8	28.6	0.1	–	0.99
-7	0.0104	1.00000	1.02778	0.99	46.0	28.8	0.1	0.0	0.99
-sdp	0.0056	1.02769	1.13064	–	–	–	–	–	–
all9-3	0.0117	1.00084	1.2	0.97	64.5	38.4	–	–	0.99
-5	0.0117	1.00002	1.08	0.99	65.1	38.5	0.2	–	0.99
-7	0.0121	1.00001	1.03571	0.99	65.1	38.5	0.2	0.0	0.99
-sdp	0.0097	1.02791	1.13064	–	–	–	–	–	–
all10-3	0.0113	1.00076	1.2	0.96	87.9	49.7	–	–	0.99
-5	0.0114	1.00001	1.08	0.99	89.4	50.0	0.3	–	0.99
-7	0.0125	1.00000	1.03571	0.99	89.4	50.0	0.3	0.0	0.99
-sdp	0.0131	1.02787	1.13064	–	–	–	–	–	–

Table 4. The number of all 2-connected graphs

n	5	6	7	8	9	10
$\#G$	11	61	507	7442	197772	9808209

column G means that the instances are 2-connected graphs with n vertices and exactly m edges. The number of the instances is 100 for each graph type. Table 3 gives results for all 2-connected graphs for $n = 5, \ldots, 10$. The number of the instances is shown in Table 4. The instances are generated by a graph generator *geng* included in *gtools* by McKay [12].

If we compare the integrality ratio for graphs of different density we can see the difference between the LP relaxation and the SDP relaxation. The integrality ratio of the LP relaxation gets bad on average as the density of the graph grows, and it is opposite in case of the SDP relaxation. As we showed in this paper, the integrality ratio of the LP relaxations can be bad for sparse graphs. However we got good average integrality ratios for sparse graphs, since the LP relaxation finds the maximum cut for sparse graphs with high probablity for the values of n considered here. On the other hand, the integrality ratio of the SDP relaxation got better both on average and in the worst case as the density grows.

As we can see from the running time in the tables, our program is not competitive with SDPA. Especially, in case of solving $G_{30,0.9}$ with 3,5,7-gonal inequalities (cf. Table 1) our program took longer than enumerating all 2^{30} cuts. There are many ways the program could be improved. We did not handle sparse vectors in any special way. The program could be made to terminate when the duality gap is smaller than a given value. On the other hand for sparse graphs (cf. Table 2), our program often obtained a maximum cut in reasonable time.

Acknowledgements. The authors greatfully acknowledge discussions with Luc Devroye and Bruce Reed. They are also indebted to an anonymous referee for numerous suggestions for improving the original draft.

References

[1] Arora, S., Karger, D., Karpinski, M.: Polynomial time approximation schemes for dense instances of NP-Hard problems. Proc. 27th ACM STOC 1995, 284–293
[2] Avis, D., Grishukhin, V.P.: A bound on the k-gonality of facets of the hypermetric cone and related complexity problems. Computational Geometry: Theor. Appl. **2**, 241–254 (1993)
[3] Barahona, F., Mahjoub, A.: On the cut polytope. Math. Prog. **36**, 157–173 (1986)
[4] Chernoff, H.: A measure of the assymptotic efficiency for tests of a hypothesis based on the sum of observations. Annal. Math. Statistics **23**, 493–509 (1952)
[5] De Simone, C., Rinaldi, G.: A cutting plane algorithm for the Max-Cut problem. Optim. Method & Softw. **3**, 195–214 (1994)
[6] Deza, M.(M.E. Tylkin): Realizablility of distance matrices in unit cubes (in Russian). Problemy Kybernetiki **7**, 31–42 (1962)
[7] Deza, M., Grishukhin, V.P., Laurent, M.: The hypermetric cone is polyhedral. Combinatorica **13**, 397–411 (1993)
[8] Deza, M., Laurent, M.: Geometry of Cuts and Metrics Springer, 1997
[9] Fujisawa, K., Kojima, M., Nakata, K.: SDPA(SemiDefinite Programming Algorithm) User's Manual – Version 5.00. Department of Mathematical and Computer Science, Tokyo Institute of Technology, Research Reports, 1999
[10] Goemans, M., Williamson, D.: 0.878-Approximation Algorithms for MAX CUT and MAX 2SAT. Proc 26th STOC, 1994, pp. 422–431
[11] Laurent, M., Poljak, S.: Gap inequalities for the gap polytope. Europ. J. Combinatorics **17**, 233–254 (1996)
[12] McKay, B.: Practical graph isomorphism. Congr. Numer. **30**, 45–87 (1981)
[13] Poljak, S., Tuza, Zs.: The expected relative error of the polyhedral approximation of the Max-Cut problem. Oper. Res. Lett. **16**, 191–198 (1994)
[14] Seymour, P.: Matroids and multicommodity flows. European J. Combinatorics **2**, 257–290 (1981)
[15] Van Ngoc, N., Tuza, Zs.: Linear-time Approximations of the Max-Cut Problem. In: G. Halász, D. Miklós and T. Szönyi (eds) Graphs and Numbers, Colloq. Math. Soc. János Bolyai 60, North-Holland, Amsterdam 1992, pp. 569–581
[16] Umemoto, J.: Linear Programming Relaxations of Max-Cut. Master's Thesis, Graduate School of Informatics, Kyoto University, 2002