Relating max-cut problems and binary linear feasibility problems

Florian Jarre, Preprint, Feb. 2009

Abstract

This paper explores generalizations of the Goemans-Williamson randomization technique. It establishes a simple equivalence of binary linear feasibility problems and max-cut problems and presents an analysis of the semidefinite max-cut relaxation for the case of a single linear equation. Numerical examples for feasible random binary problems indicate that the randomization technique is efficient when the number of linear equations is large.

Key words: Max-cut problem, semidefinite relaxation, randomized algorithm.

1. Introduction

By combining the global search feature of a randomized approach with a tight semidefinite approximation of the max-cut problem, the Goemans-Williamson approximation technique [3] provides a practical algorithm along with a good error bound. Moreover, and in contrast to many other semidefinite approximations, the semidefinite approximation of the max-cut problem has a special structure allowing its very efficient numerical solution. A local search starting from the Goemans-Williamson solution often generates nearly optimal solutions – measured in terms of the objective function value, see e.g. [8]. The error bound in [3] is established under certain conditions on the problem, but the algorithm with local refinement proves to be very effective also for instances where these assumptions are not satisfied. In fact, there exist sharper semidefinite approximations of the max-cut polytope than the one used by Goemans and Williamson, see e.g. [5], but these are also computationally more expensive.

Unfortunately, the max-cut problem only has few practical applications. While it is known that any NP-complete problem can be reduced to the max-cut problem (see [2]), the dimension of the problem size may increase (polynomially) during such a reduction. We show how a linear binary feasibility problem can be transformed to a max-cut problem of the same dimension, and conversely, how any max-cut problem can be reduced to a binary feasibility problem of the same dimension, thus establishing a certain equivalence of both problem classes. We also discuss practical differences of both problems and analyze the case of a single linear constraint in detail.

The goal of this paper is to test a generalization of the Goemans-Williamson approach to a larger class of problems. We briefly sketch an extension to mixed integer feasibility problems and discuss the practical limitations of this extension.

1.1 Notation

We use the standard notation, i.e. $\mathcal{S}^n$ is the set of symmetric $n \times n$-matrices, and the symbol $X \succeq 0$ is used to indicate that X is symmetric and positive semidefinite. By $e = (1, \ldots, 1)^T$ we denote the all-ones-vector of appropriate dimension, $E = ee^T$ is the all-ones-matrix, and e_i is the i-th unit vector. The usual inner product on the space of $n \times n$-matrices inducing the Frobenius norm is denoted by $A \bullet B = \text{trace}(A^T B)$. By $\text{diag}(X)$ we denote the vector formed by the diagonal entries of X, and conversely, for $x \in I\!R^n$ we denote the diagonal matrix with diagonal entries x_i by $\text{Diag}(x)$. The Hadamard product (componentwise product) of two matrices A, B of same dimension is denoted by $A \circ B$; likewise for vectors x, s of same dimension.

2. The max-cut problem and its SDP relaxation

Let $C \in \mathcal{S}^n$ be a given matrix. We consider the quadratic program

$$\text{minimize} \quad x^T C x \quad \text{s.t.} \quad x \in \{-1, 1\}^n. \tag{1}$$

The max-cut-problem can be written in the form (1), and conversely, when allowing for real edge weights any problem of the form (1) can be written as a weighted max-cut problem see, e.g. [6, 4]; so, we briefly refer to (1) as max-cut-problem.

Suppose that we are given a ± 1-vector $\bar{x}$ and like to bound its distance from optimality for problem (1). Let x be another ± 1-vector and y in $I\!R^n$. Then, $x^T \text{Diag}(y) x = e^T y = \bar{x}^T \text{Diag}(y) \bar{x}$. Thus, $\bar{x}$ is optimal for (1), if, and only if, it is also optimal for

$$\text{minimize} \quad x^T (C - \text{Diag}(y)) x \quad \text{s.t.} \quad x \in \{-1, 1\}^n \tag{2}$$

for any $y \in I\!R^n$.

To establish optimality of $\bar{x}$ for (2) we like to select y such that $\bar{x}$ is an eigenvector to the smallest eigenvalue $\lambda_{min}(C - \text{Diag}(y))$ of $C - \text{Diag}(y)$. Since all feasible x have the same Euclidean norm this is indeed a sufficient condition for optimality of $\bar{x}$.

Unfortunately, this sufficient condition cannot be satisfied in general, not even if $\bar{x}$ was an optimal solution. To obtain a bound on how close to optimality $\bar{x}$ might be, we therefore like to select y such that $\bar{x}$ is "close" to an eigenvector of $C - \text{Diag}(y)$ to $\lambda_{min}(C - \text{Diag}(y))$. More precisely, in view of $\|\bar{x}\|_2^2 = n$, the gap

$$\bar{x}^T (C - \text{Diag}(y)) \bar{x} - n \lambda_{min}(C - \text{Diag}(y)) \tag{3}$$

is an upper bound for the unknown distance $\bar{x}^T C\bar{x} - (x^{opt})^T C x^{opt}$. Adding a multiple of e to y does not change the eigenvectors, so that we may assume without loss of generality that $\lambda_{min}(C - \mathrm{Diag}(y)) = 0$. Minimizing the bound (3) reduces to the problem

$$\underset{y \in I\!R^n}{\text{minimize}} \quad \bar{x}^T (C - \mathrm{Diag}(y))\bar{x} \quad \text{s.t.} \quad C - \mathrm{Diag}(y) \succeq 0. \tag{4}$$

Note that the objective function of (4) is simply $const - e^T y$, independent of the solution $\bar{x}$. Hence, finding the solution y of (4) is the same as solving

$$\text{maximize} \quad e^T y \quad \text{s.t.} \quad C - \mathrm{Diag}(y) \succeq 0. \tag{5}$$

This is the dual of the standard semidefinite programming (SDP) relaxation of the max-cut problem,

$$\text{minimize} \quad C \bullet X \quad \text{s.t.} \quad \mathrm{diag}(X) = e, \quad X \succeq 0. \tag{6}$$

The optimal values of (5) and (6) coincide and will be denoted by $SDPval$. $SDPval$ forms a lower bound for the optimal value of (1), and it can be related to certain randomly generated feasible points for (1): For the case that all off-diagonal elements of C are nonnegative, Goemans and Williamson show in [3] how to generate random vectors $x \in \{-1, 1\}^n$ such that the improvement $x^T C x$ over $\Sigma^{GW} := e^T C e$ satisfies

$$x^T C x - \Sigma^{GW} \quad \leq \quad 0.878(SDPval - \Sigma^{GW}) \leq 0 \tag{7}$$

in the average. (The presentation in [3] is slightly simpler by modifying C such that $\Sigma^{GW} = 0$.)

The bound (7) has been generalized by Nesterov [7] to arbitrary symmetric matrices C. In this case, the normalization term Σ^{GW} in (7) is replaced with $\Sigma^N := \max\limits_{C + \mathrm{Diag}(y) \preceq 0} e^T y$. In this case, the same random vectors $x \in \{-1, 1\}^n$ as for (7) satisfy

$$x^T C x - \Sigma^N \quad \leq \quad \frac{2}{\pi}(SDPval - \Sigma^N) \leq 0 \tag{8}$$

in the average. Note that $\Sigma^N = \Sigma^{GW}$ when the off-diagonal elements of C are nonnegative; in this case, (8) has a somewhat weaker constant of $\frac{2}{\pi} \approx 0.636$ in place of 0.878. Just like for (7), also the result (8) in [7] is stated for the case that $\Sigma^N = 0$.

3. Binary feasibility problems

3.1 Preliminaries

For a vector $x \in I\!R^n$ with $x_1 \geq 0$ we denote by $\mathrm{Arw}(x)$ the symmetric $n \times n$-"arrow shaped matrix" which is all zero except from $\mathrm{diag}(\mathrm{Arw}(x)) = |x|$ and $\mathrm{Arw}(x)e_1 = x$ (and $e_1^T \mathrm{Arw}(x) = x^T$),

$$\mathrm{Arw}(x) = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_2 & |x_2| & & \\ \vdots & & \ddots & \\ x_n & & & |x_n| \end{pmatrix}.$$

This is *not* the standard arrow matrix as introduced in [1], but a matrix that comes up in our analysis.

Proposition 1 *The matrix Arw(x) is positive semidefinite if, and only if,* $x_1 \geq \sum_{j\geq 2} |x_j|$.

Proof. Straightforward. □

Note that due to the use of absolute values, the mapping $x \to \mathrm{Arw}(x)$ is not a linear mapping.

3.2 Binary feasibility and max-cut problems

Let $\hat{A} \in I\!R^{m\times(n-1)}$ and $\hat{b} \in I\!R^m$ be given. We consider the "binary feasibility problem"

$$\text{find} \quad \hat{z} \in \{0,1\}^{n-1} \quad \text{s.t.} \quad \hat{A}\hat{z} = \hat{b}. \tag{9}$$

Throughout this paper we will make the following assumption:
Assumption (A): We assume without loss of generality that $\hat{A}$ has linearly independent rows (else preprocess by Gaussian elimination) and that $m < n$ (to avoid trivial cases).

In the case that $m = 1$ this problem is the standard knapsack problem[1]. In slight generalization of (9) we also call the problem

$$\text{minimize} \quad \|\hat{A}\hat{z} - \hat{b}\|^2 \quad \text{s.t.} \quad \hat{z} \in \{0,1\}^{n-1}, \tag{10}$$

a "binary feasibility problem" and say that problems (9) or (10) are solvable if the optimal value of (10) is zero.

Setting $\hat{x} := 2\hat{z} - e$ and $\bar{b} := 2\hat{b} - \hat{A}e$ problem (10) is equivalent to

$$\text{minimize} \quad \|\hat{A}\hat{x} - \bar{b}\|^2 \quad \text{s.t.} \quad \hat{x} \in \{-1,1\}^{n-1},$$

or to

$$\text{minimize} \quad x^T A^T A x \quad \text{s.t.} \quad x = \begin{pmatrix} \hat{x} \\ x_n \end{pmatrix} \in \{-1,1\}^n, \tag{11}$$

where $A := [\hat{A}, -\bar{b}]$.

(When $x_n = 1$ the equivalence is evident, when $x_n = -1$, then $-x$ is also a feasible solution with the same objective value.)

Thus, the binary feasibility problem (10) can be reduced to a max-cut problem (1) of the same dimension[2] with $C = A^T A$. The converse direction, that any max-cut problem (1) can be transformed to a binary feasibility problem (10) satisfying Assumption **(A)** may be less well-known:

The solution of (5) yields a diagonal matrix $\mathrm{Diag}(y)$ such that $C - Diag(y) \succeq 0$. Moreover, $C - Diag(y)$ has a (typically multiple) eigenvalue zero. Diagonal matrices do not change the

[1] in the version of choosing (out of $n-1$ given items) a number of items such that the total weight of the selected items is equal to $\hat{b}$.

[2] Strictly speaking, the dimension increases by 1, but this is merely for the ease of notation; as indicated, one may fix $x_n = 1$ for example, and still have an equivalent problem.

optimal solution of the max-cut problem (1). Thus, given the optimal solution y^{opt} of the dual SDP relaxation (5) we can replace C with $C - \mathrm{Diag}(y^{opt}) \succeq 0$. Thus, $C \succeq 0$ and C has a (multiple) eigenvalue zero. Let $C = A^T A$ be a decomposition of C so that $A \in I\!R^{m \times n}$ with rank $m < n$. (Such a decomposition can be obtained form the eigenvalue decomposition of C or from a modified Cholesky decomposition.) With this definition of A, problem (1) is equivalent to (11) and hence also to (10). Summarizing, we obtain:

Lemma 1 *Given the solution of the SDP relaxation (5), problem (1) can be reduced to (10), and conversely, problem (10) can be reduced to (1) – both without increasing the dimension of the problem.*

As a consequence of the equivalence established in Lemma 1, any algorithm which "works well" for all problems in one class also works well for all problems in the other class. In particular, the overall success of the Goemans-Williamson approach is suggesting to apply this technique also to binary feasibility problems (10).

3.3 The knapsack problem

The knapsack problem (case $m = 1$) is interesting since one may assume without loss of generality that $A \geq 0$, so that the bound (7) is always applicable. Indeed, any component $A_i < 0$ can be replaced with $-A_i$ when flipping the sign of the associated variable x_i.

Unfortunately, when $m = 1$, there are simple greedy type algorithms that yield better bounds than (7). Moreover, as shown in Theorem 1, when $m = 1$, either the SDP relaxation (5) has the trivial optimal solution $y^{opt} = 0$ or the knapsack problem (11) itself has a trivial solution. As illustrated in Section 4., even when $y^{opt} = 0$ is the optimal solution of the dual problem (5), the Goemans Williamson approach based on the primal optimal solution still has a slightly better performance than plain random sampling.

For the sake of generality we do not make use of the assumption $A \geq 0$ in Theorem 1.

Theorem 1 *For $m = 1$ and $C = A^T A$, the vector $y = 0$ is not optimal for (5), if, and only if, there is a component $\bar{i}$ such that $|A_{\bar{i}}| > \sum_{j \neq \bar{i}} |A_j|$.*

Proof. We assume without loss of generality, that $A_1 \geq |A_j| \geq |A_{j+1}|$ for $2 \leq j < n$. (If $A_1 < 0$ replace A with $-A$, and if $|A_j| < |A_{j+1}|$, then permute A and x.) We may also assume that $A_i \neq 0$ for all i (if not the dimension of the problem can be reduced).

We first consider the case where $\bar{i} = 1$ satisfies $\rho := A_1 - \sum_{j \geq 2} |A_j| > 0$. Then, it is easy to see that x^{opt} with $x_1^{opt} := 1$ and $x_j^{opt} := -\mathrm{sign}(A_j)$ for $j \geq 2$ is an optimal solution of (1) and (11). Let $\bar{X} := x^{opt}(x^{opt})^T$ and $\bar{y} := \rho A^T \circ x^{opt}$. By construction it follows that $e^T \bar{y} = \rho^2 > 0$.

Let $\bar{S} := C - \mathrm{diag}(\bar{y})$, then,

$$\bar{S} \bullet \bar{X} = (x^{opt})^T \bar{S} x^{opt} = (A^T x^{opt})^2 - e^T \bar{y} = \rho^2 - \rho^2 = 0.$$

Setting $a := (A_1 - \rho, A_2, \ldots, A_n)^T$, the matrix $\bar{S}$ can also be written as

$$\begin{aligned}
\bar{S} &= (a + \rho e_1)(a + \rho e_1)^T - \mathrm{Diag}(\bar{y}) \\
&= aa^T + \rho(ae_1^T + e_1 a^T + \rho e_1 e_1^T - \mathrm{Diag}((a + \rho e_1) \circ x^{opt}) \\
&= aa^T + \rho \mathrm{Arw}(a) \\
&\succeq 0.
\end{aligned}$$

The last line follows from Proposition 1 and shows that $\bar{X}$, $\bar{y}$, and $\bar{S}$ are optimal for (6) and (5). In particular, $y = 0$ is not an optimal solution for (5). This proves the first part of the lemma.

Now suppose that $y = 0$ is not an optimal solution for (5), i.e. there is a feasible $\bar{y}$ with $e^T\bar{y} > 0$. Then, $\bar{S} := C - \mathrm{diag}(\bar{y}) \succeq 0$. Without loss of generality, $\bar{S} \succ 0$ (else replace $\bar{y}$ with $\bar{y} - (e^T\bar{y}/(2n))e$). To facilitate the proof we observe that we may assume, without loss of generality that $\bar{y}_i \neq 0$ and reorder the indices of A (and $\bar{y}$) such that $\bar{y}_i \geq \bar{y}_{i+1}$ for $1 \leq i < n$. In particular, since $e^T\bar{y} > 0$ it follows $\bar{y}_1 > 0$. By considering the leading 2×2 principal submatrix of $\bar{S}$ we may conclude that $\bar{y}_2 < 0$ (else its determinant is negative), and thus, $\bar{y}_i < 0$ for all $i \geq 2$.

Let us consider $\det(\bar{S})$. By subtracting multiples of the first row from rows $2 \ldots n$, we obtain

$$\begin{aligned}
\det(\bar{S}) &= \det(C - \mathrm{Diag}(\bar{y})) \\
&= \det(\begin{pmatrix} A_1 A \\ A_2 A \\ \vdots \\ A_n A \end{pmatrix} - \mathrm{Diag}(\bar{y})) \\
&= \det(\begin{pmatrix} A_1 A \\ 0 \\ \vdots \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \bar{y}_1 A_2/A_1 \\ \vdots \\ \bar{y}_1 A_n/A_1 \end{pmatrix} e_1^T - \mathrm{Diag}(\bar{y})).
\end{aligned}$$

This is the determinant of another "arrow shaped" matrix. Its value is given by

$$\begin{aligned}
\det(\bar{S}) &= (A_1^2 - \bar{y}_1) \prod_{i \geq 2}(-\bar{y}_i) - \sum_{i \geq 2}((\bar{y}_1 A_i/A_1)(A_1 A_i) \prod_{j \neq 1,\ j \neq i}(-\bar{y}_j) \\
&= (A_1^2 - \bar{y}_1) \prod_{i \geq 2}(-\bar{y}_i) + \sum_{i \geq 2} A_i^2 \prod_{j \neq i}(-\bar{y}_j) \\
&= \sum_{i \geq 1} A_i^2 \prod_{j \neq i}(-\bar{y}_j) + \prod_{j \geq 1}(-\bar{y}_j).
\end{aligned}$$

Since also $\bar{S}(\alpha) := C - \alpha\mathrm{Diag}(\bar{y}) \succ 0$ for $\alpha \in (0, 1)$, it follows that $\det(\bar{S}(\alpha)) > 0$. The term $\prod_{j \geq 1}(-\alpha\bar{y}_j)$ associated with α^n is dominated for small $\alpha > 0$ so that

$$\sum_{i=1}^{n} A_i^2 \prod_{j \neq i}(-\bar{y}_j) > 0.$$

We recall the assumption $\bar{y}_1 > 0 > \bar{y}_2 \geq \ldots \geq \bar{y}_n$ and divide by $\prod_{j=1}^{n}(-y_j) < 0$ yielding

$$\sum_{i=1}^{n} \frac{A_i^2}{-\bar{y}_j} < 0.$$

We consider the problem

$$\max\ e^T y \ \mid\ \sum_{i=2}^{n} \frac{A_i^2}{-y_i} \leq \frac{A_1^2}{\bar{y}_1},\quad y_1 = \bar{y}_1,\quad y_2, \ldots, y_n < 0.$$

By assumption, the vector $\bar{y}$ is feasible for this problem, and $e^T \bar{y} > 0$.

The maximum of this problem is attained. (Indeed, the first constraint implies that all y_i are bounded away from zero, and the objective implies that none of the y_i tends to $-\infty$.) Moreover, the first constraint is active at the optimal point, and the KKT conditions imply that there exists a Lagrange parameter $\mu \geq 0$ such that

$$\frac{A_i^2}{y_i^2} \equiv \mu \ \text{ or } \ y_i = -|A_i|/\sqrt{\mu} \ \text{ for all } \ i \geq 2.$$

From $e^T y > 0$ follows $\bar{y}_1 > -\sum_{i\geq 2} y_i$. Inserting this into the first constraint we obtain

$$\sqrt{\mu}\sum_{i=2}^{n} |A_i| = \frac{A_1^2}{\bar{y}_1} < \frac{A_1^2}{-\sum_{i\geq 2} y_i} = \sqrt{\mu}\frac{A_1^2}{\sum_{i\geq 2} |A_i|}.$$

Hence, $\sum_{i\geq 2} |A_i| < A_1$. This completes the proof. □

When $y_i = 0$, then the constraints $X_{i,i} = 1$ of the primal problem (6) are redundant in the sense that their omission does not change the optimal value of (6). For random A_i, problem (9) typically is not solvable, but the SDP-relaxation (5) will provide a certificate for the nonexistence of a solution of (9) only in the trivial case where one of the $|A_i|$ is larger than the sum of all other $|A_j|$.

4. Random examples

As established in Section 3.2, max-cut problems and binary feasibility problems are – in principle – equivalent. There are two practical differences, however:

- For binary feasibility problems one is often interested in instances, where the binary quadratic program (10) is solvable. In this case the optimal values of (1) and (6) coincide, i.e. the error bound derived from (6) is sharp. This situation is not typical for max-cut problems in general. As we will see below, for small values of m solvable instances of (10) are nevertheless rather difficult to solve.

- For typical instances of a max-cut problem the optimal solution $C - \mathrm{Diag}(y^{opt})$ of (5) may have a rank close to n. (When C is generated at random – iid normally distributed entries – then the rank of $C - \mathrm{Diag}(y^{opt})$ is near 94.4 in the average for $n = 100$, and it is near 985.5 in the average for $n = 1000$). In the equivalent binary feasibility problem (11) the matrix A thus has dimension $m \times n$ with m very close to n. For typical binary feasibility problems, however, the rank m may be much smaller.

The examples below show that solvable problems of the form (10) or (11) tend to be easy when m is moderately large. Non solvable instances of (11) correspond to general max-cut problems for which the Goemans-Williamson approach has been tested extensively; we therefore concentrate on solvable problems.

In Table (12) we list the results of 1000 random solvable test problems of dimension 100. The entries of $\hat{A}$ are iid standard normally distributed. For each $\hat{A}$ a vector $\hat{x}$ is chosen uniformly in $\{-1, 1\}$ and $\bar{b}$ is set directly $\bar{b} := \hat{A}\hat{x}$ (without forming $\hat{b}$ from Section 3.2). For each random example, 100 random vectors are generated according to

1. Goemans-Williamson (column (GW)),
2. randomly in $A^{\perp}$ (column $(A^{\perp})$), and
3. randomly in $\mathbb{R}^n$ (column $(\mathbb{R}^n)$).

For each problem and each of the 3 approaches the best residual $\|\hat{A}x - \bar{b}\|_2$ is recorded. Table (12) lists the average of these best values.

m	(GW)	$(A^{\perp})$	$(\mathbb{R}^n)$
1	0.102	0.182	0.302
30	21.7	37.8	53.8
50	9.75	52.3	74.8
70	0	60.7	91.4

(12)

Evidently, when solving $Ax = 0$ for some $\{\pm 1\}$-vector x it is more efficient to round a random vector from the null space of A to $\{\pm 1\}$ than a random vector from $\mathbb{R}^n$ (columns $(A^{\perp})$ and $(\mathbb{R}^n)$ in Table (12).) And evidently, the Goemans-Williamson approach is a further improvement over $(A^{\perp})$. For $m = 1$, the optimal solution y^{opt} of (5) is zero for all of the above examples. Thus, the columns of the Cholesky factor R^T of $X^{opt} = R^T R$ span the null space of A. A somewhat surprising fact is that nevertheless, the improvement of Goemans-Williamson (GW) over $(A^{\perp})$ is significant. (We repeated the experiment several times and obtained very similar average values.) Both, $(A^{\perp})$ and (GW) generate random vectors in $A^{\perp}$, the only difference is that the random vectors are generated with a (slightly) different distribution in case of (GW). For $m = 30$ there were 56 cases with $\|y\|_1$ larger than 1 (indicating a nontrivial solution of (5)) with a tendency that larger values of $\|y\|_1$ also result in a larger gap between (GW) and $(A^{\perp})$. For $m = 50$, $\|y\|_1$ was always larger than 1 and there were 561 cases, where (GW) found the exact optimal solution; the average of the residuals over the remaining 431 cases was 22.2. Finally, for $m = 70$ (GW) found the exact optimal solution for all 1000 examples.

Table (12) illustrates the behavior of (GW) or $(A^{\perp})$ compared to " blind random sampling" in column $(I\!R^n)$. This certainly does not prove effectiveness of (GW). In fact, random sampling is typically used with a local refinement strategy. We implemented a very simple such strategy, testing at each step whether a flip of sign of one or two indices will improve the objective value. One refinement step thus tests about $n^2/2$ pairs of indices.

We ran another 100 random test problems of dimension 100 where a local refinement (limited to 100 refinement steps each) was applied for each of 100 random starting points. Table (12) lists the average of the best values obtained after such a local search.

m	(GW)	$(A^{\perp})$	$(I\!R^n)$
1	$8.34e-6$	$9.78e-6$	$9.46e-6$
30	9.38	9.66	9.53
50	3.58	16.5	15.3
70	0	0.700	0.252

(13)

For small values of m the advantage of (GW) over $(I\!R^n)$ is greatly reduced by the local search. (The amount of work used in the local search was comparable for all three approaches; an average of about 4 refinement steps for each random starting point.) For $m = 70$ there were only 3 instances for $(A^{\perp})$ where the optimal solution was not found and only one instance for $(I\!R^n)$ where the optimal solution was not found.

For $m = 1$ the Goemans-Williamson rounding technique appears to be least efficient – even though the assumptions (non negative edge weights) are satisfied. To test this conjecture also with other software, we generated a solvable random knapsack problem with $n = 49$ and weights "$1 + 0.1 * rand(n, 1)$" resulting in an input graph with 50 nodes and 1225 edges with edge weights in $[1,\ 1.21]$. This problem was submitted to BiqMac [9], a program that generally solves max-cut problems of dimension ≤ 100 to optimality. The result of BiqMac confirms the conjecture that this type of problem is not well suitable for semidefinite relaxations. After a total number of 944741 branch-and-bound nodes and exceeding the time limit of three hours the algorithm did find an approximate optimal solution within its default accuracy (even though the error bound from (5) was sharp).

5. Outlook

While it is known that there exist stronger bounds than the one used by Goemans and Williamson (see [5]), the smart combination with a randomization technique proved to be very efficient for practical problems. This suggests to extend this efficient technique not only to binary feasibility problems but also to the very important class of mixed integer linear feasibility problems,

$$\text{find} \quad x \in I\!R^{n_1} \times I\!N_+^{n_2} \quad | \quad \hat{A}x = b. \tag{14}$$

By introducing slack variables, inequality constrained problems can be reduced to the form (14). After solving the LP-relaxation, rounding, and projecting, a correction step is sought in which the change of an integer variable x_i $(i > n_1)$ is limited to small increments of the

form $-2^k+x_{i,1}+2x_{i,3}+\ldots+2^k x_{i,k}$ for some small fixed value of k. Here, $x_{i,j}$ are $0,1$-variables. One obtains a problem of the form

$$\text{minimize} \quad x^T A^T A x \quad \text{s.t.} \quad x \in \{1\} \times I\!R_+^{n_1} \times \{-1,1\}^{kn_2},$$

a generalization of (11) to mixed binary and nonnegative variables. In the primal semidefinite relaxation, the leading $(1+n_1)\times(1+n_1)$-block is nonnegative, and the diagonal elements $1,\ n_1+2,\ldots,n_1+n_2+1$ are set to 1. Based on the solution of the semidefinite relaxation, the variables x_i for $i \in \{1, n_1+2,\ldots,n_1+n_2+1\}$ can be determined by the usual Goemans-Williamson rounding technique. Without loss of generality the result satisfies $x_1 = 1$. Thereafter, the nonnegative variables can be adjusted by solving a linear program.

Unfortunately, the semidefinite programs obtained this way are computationally more expensive. We compare the timings obtained from SeDuMi [10, 11] for the case of $n/2$ inequality constraints (SeDuMi1) and for the case of binary constraints only (SeDuMi2) with the timings of a self made max-cut sdp-code (MC-SDP) that does not require sparse matrix techniques.

n	(SeDuMi1)	(SeDuMi2)	(MC-SDP)
50	2.67	0.430	0.056
100	69.4	1.31	0.159

(15)

All timings refer to the same computer. The timings for problems with inequality constraints are about 50 times higher than for plain max-cut sdp-relaxations. Given the at best "moderately convincing" numerical results for the binary feasibility problem, the above timings do not indicate that this approach will be practical.

6. Conclusion

By exploiting a simple equivalence of binary linear feasibility problems and max-cut problems this paper aims at testing possible practical generalizations of the Goemans-Williamson technique to a wider class of problems. While the technique may be useful for solvable instances with moderately large values of m, the results for smaller values of m are less encouraging. For the extreme case of $m = 1$ linear equation (and a nontrivial binary feasibility problem) the dual solution of the semidefinite max-cut relaxation is always zero.

Acknowledgment

The author likes to thank Paul Tseng for some helpful comments on the paper.

References

[1] I. Adler and F. Alizadeh, "Primal-Dual Interior Point Algorithms for Convex Quadratically Constrained and Semidefinite Optimization Problems", RUTCOR Research Reports 46-95, http://rutcor.rutgers.edu/ rrr/1995.html, 1995.

[2] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman, 1979.

[3] M.X. Goemans and D.P. Williamson, "Improved Approximation Algorithms for maximum cut and satisfiability problems using semidefinite programming", *Journal of ACM*, 42:1115-1145, 1995.

[4] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz, "An Interior-Point Method for Semidefinite Programming", *SIAM J. Optim.*, Vol. 6, No. 2, pp. 342–361, 1996.

[5] M. Laurent, "Tighter Linear and Semidefinite Relaxations for Max-Cut Based on the Lovasz-Schrijver Lift-and-Project Procedure", *SIAM J. Optim.*, Vol. 12, 345–375, 2002.

[6] B. Mohar and S. Poljak, "Eigenvalues in Combinatorial Optimization", in *Combinatorial and graph-theoretic problems in linear algebra*, R. Brualdi, S. Friedland, V. Klee, Eds., *IMA Volumes in Mathematics and its Applications*, Vol. 50 (1993) Springer Verlag.

[7] Y. Nesterov, "Semidefinite Relaxation and Nonconvex Quadratic Optimization", *Optimization Methods and Software* 9, 141–160, 1998.

[8] F. Rendl, G. Rinaldi, and A. Wiegele, "Solving Max-Cut to Optimality by Intersecting Semidefinite and Polyhedral Relaxations", Technical Report, Alpen-Adria-Universität Klagenfurt, http://biqmac.uni-klu.ac.at/rrw_plain.pdf, 2008.

[9] F. Rendl, G. Rinaldi, and A. Wiegele, "Biq Mac Solver - Binary quadratic and Max cut Solver" http://biqmac.uni-klu.ac.at/

[10] J.F. Sturm, "Using SeDuMi 1.02, a MATLAB Toolbox for Optimization Over Symmetric Cones", *Optimization Methods and Software* 11-12 (1999) 625–653.

[11] SeDuMi download page, http://sedumi.ie.lehigh.edu/