# Semidefinite programming relaxations for graph coloring and maximal clique problems [†]

Igor Djukanovic [‡], Franz Rendl [§]

February 23, 2005

**Abstract**

The semidefinite programming formulation of the Lovász theta number does not only give one of the best polynomial simultaneous bounds on the chromatic number $\chi(G)$ and the clique number $\omega(G)$ of a graph, but also leads to heuristics for graph coloring and extracting large cliques. This semidefinite programming formulation can be tightened toward either $\chi(G)$ either $\omega(G)$ by adding several types of cutting planes. We explore several such strengthenings, and show that some of them can be computed with the same effort as the theta number. We also investigate computational simplifications for graphs with rich automorphism groups.

**Keywords**: Lovász theta number, chromatic number, clique number, cutting planes.

## 1 Introduction

The Lovász theta number $\theta(G)$, introduced in [16], is sandwiched between the clique number and the chromatic number of a graph

$$\omega(G) \leq \theta(G) \leq \chi(G).$$

It can be computed to arbitrary fixed precision in polynomial time by the interior point methods as it can be formulated as a semidefinite program (SDP) [16]. The positive semidefinite matrix variable in this SDP can be used to either extract large cliques [9] or to approximately color the given graph [13]. There are graphs with arbitrarily large gaps $\theta(G) - \omega(G)$ and/or $\chi(G) - \theta(G)$. Well-known examples are triangle free ($\omega(G) = 2$) Mycielski graphs with arbitrarily large chromatic numbers.

There has been considerable scientific effort to strengthen the original bound of Lovász toward the clique number (or equivalently the stability number in the complement) [19, 27, 17]. To better approximate the chromatic number, Szegedy strengthened the Lovász theta number by adding nonnegativity constraints [28]. Meurdesoif further tightened this bound by the inclusion of triangle inequalities [21]. None of these tighter models can be solved routinely, see for instance [21]. We refer to [15] for a recent survey on relaxations of max-clique and coloring.

It is the purpose of the present paper to explore efficient algorithms to compute $\theta$ and its various strengthenings toward $\omega$ and $\chi$. The starting point are two independent models for $\theta$. The standard model (14) allows to compute $\theta$ very efficiently on dense graphs. The second model (15) is tailored for sparse graphs. It turns out that the strengthenings of $\theta$ toward $\chi$ can be computed with an effort comparable to compute $\theta$ for the dense model. A similar observation holds for the strengthenings of $\theta$ toward $\omega$. Finally, we will explore symmetries of a given graph to simplify the computations. The key

[†]Partial support by the EU project Algorithmic Discrete Optimization (ADONET), MRTN-CT-2003-504438, is gratefully acknowledged.

[‡]Univerza v Mariboru, Ekonomsko-poslovna fakulteta, Razlagova 14, 2000 Maribor, Slovenia, igor.mat@uni-mb.si

[§]Universität Klagenfurt, Institut für Mathematik, Universitätsstraße 65-67, 9020 Klagenfurt, Austria, franz.rendl@uni-klu.ac.at

observation is that the variables corresponding to vertices in the same orbit can be assumed to have equal value. The same is true for the orbits of edges and non-edges. Similar ideas have been investigated by Parrilo, [24]. We provide computational results on vertex transitive graphs, and show that $\theta$ and its variations can be computed efficiently for vertex transitive graphs with up to 4000 vertices and several millions of edges.

## 2 Notation

The vector of all ones is denoted by $e$. The matrix $J_n = ee^T$ (or simply $J$) is the $n \times n$ matrix of all ones. We denote by $e_i$ the $i$th column of the identity matrix $I$ of appropriate dimension, and we define $E_i = e_i e_i^T$ while for $i \neq j$, $E_{ij} = e_i e_j^T + e_j e_i^T$. The pointwise product of vectors $x$ and $y$ is the vector $x \circ y$. The trace inner product

$$\langle X, Y \rangle = \text{tr } XY$$

is a well-known inner product in the space of the real symmetric matrices $\mathcal{S}^n$. It is easy to check that the adjoint of operator diag , which takes a diagonal of a matrix, is operator Diag, which maps a vector into a diagonal matrix. $A \succeq B$ ($A \succ B$) denotes that $A - B$ is a positive semidefinite (respectively positive definite) matrix.
A graph with vertex set $V = \{1, ..., n\}$ and edge set $E$ will be denoted by $G = (V, E)$. We assume $G$ to be simple (loopless, without multiple edges) and undirected. The *complement* graph $\bar{G} = (V, \bar{E})$ is the graph on the same vertex set with edges $ij \in \bar{E}$ exactly if $ij = ji \notin E$, i.e. $\bar{E} = \{ij | ij \notin E\}$. We denote the number of edges $|E|$ by $m$ and the number of non-edges $|\bar{E}|$ by $\bar{m}$. *The chromatic number* $\chi(G)$ is the smallest number of colors needed to (properly) color a graph. *The clique number* $\omega(G)$ is the size of the largest *clique* (set of pairwise adjacent vertices) in the graph.

## 3 Graph Coloring

A $s$-coloring of a graph $G = (V, E)$ is traditionally defined as a mapping $c : V \to \{1, ..., s\}$ such that

$$ij \in E \Rightarrow c(i) \neq c(j).$$

This $s$-coloring $c$ defines a $s$-coloring relation $R$ by $iRj \iff c(i) = c(j)$. The coloring relation $R$ breaks the vertex set $V$ into the color classes $c^{-1}(1), ..., c^{-1}(s)$, so it is an equivalence relation.
In matrix terms, we can express a $s$-coloring of a graph by a matrix $M \in \{0, 1\}^{n \times n}$ such that $m_{ij} = 1 \iff c(i) = c(j)$. Clearly, for any such matrix $M$ there exists a permutation matrix $P$ such that

$$P^T M P = \begin{bmatrix} J_{n_1} & 0 & & \\ 0 & J_{n_2} & & \\ & & \ddots & \\ & & & J_{n_s} \end{bmatrix}. \tag{1}$$

In other words, $P^T M P$ is the direct sum of $s$ blocks of all ones, with block sizes $n_i \geq 1$ and $\sum_i n_i = n$. We call $M$ the matrix representation of the coloring $c$. Moreover we say that $M$ is a matrix representation of a **proper** coloring, if $m_{ij} = 0$ whenever $ij$ is an edge of the underlying graph.

**Remark 1** *Let $c$ be a coloring and $M$ be its matrix representation. The color classes $c^{-1}(i)$ are given by $C_i = \{v : c(v) = i\}$. Its characteristic vectors $\chi_{C_i} \in \{0, 1\}^n$, defined by*

$$(\chi_{C_i})_v = \begin{cases} 1 & v \in C_i \\ 0 & v \notin C_i \end{cases}$$

*can be used to express $M$ as*

$$M = \sum_{i=1}^{s} \chi_{C_i} \chi_{C_i}^T. \tag{2}$$

*Note in particular that the rank of $M$ is $s$.*

We first recall the following well known characterizations of matrices of the form given in (1).

**Lemma 2** *Let $M$ be a symmetric 0-1 matrix. Then the following statements are equivalent.*

*a) There exists a permutation matrix $P$ such that $P^T M P$ is the direct sum of $s$ blocks of all ones.*

*b) $diag\,(M) = e,\ rank(M) = s,\ M \succeq 0$.*

*c) $diag\,(M) = e,\ rank(M) = s$ and $M$ satisfies the triangle inequalities:*

$$m_{ij} + m_{jk} - m_{ik} \leq 1 \quad \forall (i,j,k). \tag{3}$$

Proof. Since $J \succeq 0$, we obviously have $a) \Rightarrow b)$. To see that $b) \Rightarrow c)$, we note that a violated triangle inequality would have $m_{ij} = m_{jk} = 1,\ m_{ik} = 0$. The corresponding submatrix

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

is indefinite, contradicting $M \succeq 0$. Finally we note that the triangle inequalities imply that $M$ represents a transitive relation, so the rows and columns of $M$ can be permuted to be a direct sum of all ones matrices. Since the rank of $M$ is $s$, there must be $s$ such blocks. □

The previous characterizations of matrix representations of colorings involve a rank constraint on $M$. The following result shows that we can trade the rank constraint for a semidefiniteness constraint.

**Theorem 3** *Let $M$ be a symmetric 0-1 matrix. Then $M$ is of the form (1) if and only if*

$$d) \quad diag\,(M) = e \quad and\ (tM - J \succeq 0 \iff t \geq s).$$

Proof: Suppose $M$ is of the form (1). Then any two rows in $tM - J$ corresponding to two vertices of the same color are equal. Therefore the rows of any nonzero minor of $tM - J$ correspond to (at most) $s$ vertices of different colors, and thus the corresponding submatrix is $tI_s - J_s$, or submatrix of it. But $tI_s - J_s \succeq 0 \iff t \geq s$. So a) above implies d). We observe in particular that $sM - J \succeq 0$ in this case, and that $tM - J \not\succeq 0$ for $t < s$.
To see the other direction we observe that diagonal elements in $sM - J \succeq 0$ are non-negative. So $s \geq 1$, and $M \succeq \frac{1}{s} J \succeq 0$. Let $rankM = r$. Then by lemma 2, $M$ is a direct sum of $r$ blocks of all ones, and the previous observation shows that $r = s$. □
As a consequence we can write the coloring number $\chi(G)$ of a graph $G$ as the optimal value of the following SDP in bivalent variables.

**Corollary 4** *[21, 13]*

$$\chi(G) = \min\{t :\ Y - J \succeq 0,\ diag Y = te,\ Y = Y^T,\ y_{ij} = 0\ \forall ij \in E,\ y_{ij} \in \{0,t\}\} \tag{4}$$

Proof. From the previous theorem we have that

$$\chi(G) = \min\ t \text{ such that } tM - J \succeq 0,\ \text{diag}\,(M) = e,\ m_{ij} = 0\ \forall ij \in E,\ m_{ij} \in \{0,1\}.$$

Substituting $Y = tM$ gives the bivalent linear SDP (4). □

## 4 A hierarchy of relaxations toward $\chi(G)$

Since solving (4) is well known to be NP-hard, several relaxations were introduced. The initial relaxation of Lovasz [16] simply drops the condition that the elements of $Y$ are 0 or $t$.

$$\theta(G) = \min\{t : Y - J \succeq 0,\ \operatorname{diag} Y = te,\ Y = Y^T,\ y_{ij} = 0\ \ \forall ij \in E\} \tag{5}$$

Szegedy [28] maintains $Y \geq 0$ and gets the stronger relaxation $\theta^+(G)$.

$$\theta^+(G) = \min\{t : Y - J \succeq 0,\ \operatorname{diag} Y = te,\ Y = Y^T,\ y_{ij} = 0\ \ \forall ij \in E,\ y_{ij} \geq 0\ \ \forall ij \notin E\} \tag{6}$$

Finally Meurdesoif [21] includes the triangle inequalities (3) as well. We denote the resulting bound by $\theta^{+\triangle}$.

$$\begin{aligned}\theta^{+\triangle}(G) = \min\{t :\ & Y - J \succeq 0,\ \operatorname{diag} Y = te,\ Y = Y^T,\ y_{ij} = 0\ \ \forall ij \in E,\\ & y_{ij} \geq 0\ \ \forall ij \in \bar{E},\ y_{ij} + y_{jk} - y_{ik} \leq t\ \ \forall ij, jk \in \bar{E}\}\end{aligned} \tag{7}$$

We clearly have a chain

$$\theta(G) \leq \theta^+(G) \leq \theta^{+\triangle}(G) \leq \chi(G) \tag{8}$$

**Remark 5** *It is known that the gaps $\chi(G) - \theta(G)$ [13] and $\chi(G) - \theta^+(G)$ [6] can be arbitrarily large. We are not aware of any theoretical investigations of the gap $\chi(G) - \theta^{+\triangle}(G)$.*
*The representation of $M$, given in (2) shows that the relaxations above could be further tightened by constraining $Y$ to be in the cone $\mathcal{C}$ of completely positive matrices, with $\mathcal{C} = \{\sum y_i y_i^T : y_i \geq 0\}$. However just checking whether a matrix is completely positive is co-NP-complete [22]. Polynomial relaxations based on copositive programming are investigated in [23, 4], but practical implementations still remain a challenge.*
*All relaxations of $\chi(G)$, given in (8), can be computed in polynomial time to some prescribed precision. Several methods have been proposed in the literature to compute $\theta(G)$, ranging from interior-point methods, to methods based on eigenvalue optimization. We refer to the website of Hans Mittelmann[1] for a summary of state-of-the-art software to compute $\theta(G)$. Computing the other relaxations $\theta^+$ and $\theta^{+\triangle}$ is significantly more involved. We are not aware of any computational study based on these relaxations, aside from the preliminary work reported in [21]. We will show in the following that these bounds can be computed quite efficiently with an effort comparable to compute $\theta$ for reasonably dense graphs.*

## 5 The clique number and semidefinite relaxations

In this section we briefly recall how (5) can alternatively be obtained as a semidefinite relaxation of the Max-Clique problem [16]. Suppose $S$ is a clique in $G$. Let $\chi_S$ be its characteristic vector. Then clearly $(\chi_S)_i(\chi_S)_j = 0\, \forall ij \in \bar{E}$, because the vertices $i$ and $j$ are not in the same clique, if $ij \in \bar{E}$. Using $\chi_S$ we form the following matrix

$$X = \frac{\chi_S \chi_S^T}{\chi_S^T \chi_S}. \tag{9}$$

We get $|S| = \langle X, J\rangle$. Moreover, $X$ satisfies the following conditions: $X \succeq 0,\ \text{tr}\ (X) = 1,\ x_{ij} = 0\ \forall ij \in \bar{E},\ rank(X) = 1$. We leave it to the reader to verify that in fact

$$\omega(G) = \max\{\langle X, J\rangle :\ X \succeq 0,\ \text{tr}\ (X) = 1,\ x_{ij} = 0\ \forall ij \in \bar{E},\ rank(X) = 1\}.$$

Dropping the rank constraint on $X$, we obtain the problem (5) in dual form:

$$\theta(G) = \{\max\langle X, J\rangle :\ \text{tr}\ X = 1,\ x_{ij} = 0 \forall ij \in \bar{E},\ X \succeq 0.\} \tag{10}$$

[1] http://plato.la.asu.edu

Therefore $\omega(G) \leq \theta(G)$. By adding additional nonnegativity for $X$, (10) can be strengthened to the Schrijver's number, see [19, 27].

$$\theta^-(G) = \max\{\langle X, J\rangle : \text{tr } X = 1,\ x_{ij} = 0 \forall ij \in \bar{E}, x_{ij} \geq 0 \forall ij \in E,\ X \succeq 0\}. \tag{11}$$

Further strengthenings, see [17] lead to

$$\theta^{-\triangle}(G) = \max\langle X, J\rangle \text{ such that tr } X = 1,\ X \succeq 0, \quad \text{and}$$

$$x_{ij} = 0\ \forall ij \in \bar{E},\ x_{ij} \geq 0\ \forall ij \in E,\ x_{ij} \leq x_{ii},\ x_{ik} + x_{jk} \leq x_{ij} + x_{kk}\ \forall i, j, k \in V. \tag{12}$$

Now notice that $\omega(G) \leq \theta^{-\triangle}(G)$ since matrices of the form (9) are also feasible for (12). Since the feasible sets of (12), (11) and (10) form a chain, we get $\theta^{-\triangle}(G) \leq \theta^-(G) \leq \theta(G)$.
Thus we finally have the following inequalities

$$\omega(G) \leq \theta^{-\triangle}(G) \leq \theta^-(G) \leq \theta(G) \leq \theta^+(G) \leq \theta^{+\triangle}(G) \leq \chi(G).$$

**Remark 6** *It should be observed that model (12) is slightly weaker than $N^+(FRAC(\bar{G}))$ introduced in [17]. Burer and Vandenbussche [3] have recently proposed a computational scheme for (approximately) optimizing over the later. They provide computational results on graphs with up to 200 nodes.*

**Remark 7** *The notation for the Lovász numbers $\theta(G)$ comes in various flavours. In the context of the stability number, (5) is sometimes denoted by $\theta(\bar{G})$ or $\bar{\theta}(G)$. The Schrijver's number is sometimes denoted by $\theta'(G)$ or $\theta_2(G)$, while $\theta^+$ was originally denoted as $\theta_{1/2}$, see [28].*

## 6 Sparse and dense graphs

In this section we concentrate on computationally efficient formulations of our SDP relaxations, by exploiting the density (or sparsity) of the given graph $G$. We say that $G$ is dense, if $m > \frac{1}{2}\binom{n}{2}$ and sparse otherwise. Note that this definition differs from the usual notion, where a graph is dense if $m = \Omega(n^2)$ and sparse if $m = o(n^2)$.
The following linear operator $A_G : \mathcal{S}^n \to R^E$ will be useful.

$$A_G(X)_{ij} = \langle X, E_{ij}\rangle = x_{ij} + x_{ji} = 2x_{ij} \qquad \forall ij \in E. \tag{13}$$

Its adjoint operator $A_G^T$ is given by ${A_G}^T(y) = \sum_{ij \in E(G)} y_{ij} E_{ij}$. Using this map we can write any symmetric matrix $Z$ as

$$Z = \sum_i y_i E_{ii} + A_G^T(u) + A_{\bar{G}}^T(v),\ \text{i.e. } z_{ij} = \begin{cases} y_i & i = j \\ u_{ij} & ij \in E \\ v_{ij} & ij \notin E \end{cases}$$

Note that feasible matrices $Y$ for (5) are of the form $Y = tI + A_{\bar{G}}^T(y)$, where $y \in R^{\bar{E}}$. Thus we get

$$\begin{array}{rl} \theta(G) & = \min\ t \text{ such that } tI + A_{\bar{G}}^T(y) - J \succeq 0 \\ & = \ \max\langle J, X\rangle \text{ such that tr } (X) = 1,\ A_{\bar{G}}(X) = 0,\ X \succeq 0. \end{array} \tag{14}$$

This SDP has $\bar{m}+1$ equations (in the dual). We call it the *dense model* for $\theta(G)$. If we write $Z = Y - J$, then we get the following formulation.

$$\begin{array}{rl} \theta(G) & = \min\ t \text{ such that } te - \text{diag}\,(Z) = e,\ -A_G(Z) = 2e_m,\ Z \succeq 0 \\ & = \ \max e^T x + 2e_m^T \xi \text{ such that } e^T x = 1,\ \text{Diag}\ (x) + A_G^T(\xi) \succeq 0. \end{array} \tag{15}$$

This model has $n + m$ equations (in the primal), and we call it the *sparse model*. Clearly the sparse model is computationally more efficient for sparse graphs ($m$ small), while the dense model is better for dense graphs ($\bar{m}$ small).

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | n | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
| .1 | $\|E\|$ | 487 | 1137 | 2047 | 3149 | 4531 | 6098 | 7949 |
| | time | 2 | 13 | 52 | 164 | 470 | 997 | 2240 |
| .25 | $\|E\|$ | 1240 | 2802 | 5099 | 7926 | | | |
| | time | 11 | 109 | 560 | 2113 | | | |
| .5 | $\|E\|$ | 2531 | 5665 | 10026 | | | | |
| | time | 59 | 638 | 3982 | | | | |
| .75 | $\|E\|$ | 3734 | 8451 | 15049 | 23541 | | | |
| | time | 10 | 91 | 448 | 1678 | | | |
| .9 | $\|E\|$ | 4460 | 10079 | 17942 | 28127 | 40436 | 55099 | 71863 |
| | time | 1 | 9 | 39 | 124 | 354 | 833 | 1984 |

Table 1: Times in seconds for computing $\theta(G)$ on random graphs. Graphs with density <0.5 are computed using (15), the others with (14).

To give some impression on the practical impact, we provide in Table 1 computation times to compute $\theta(G)$ with the two models. The codes have been run on a 2200 MHz laptop with 1G RAM under Linux. The primal-dual predictor-corrector interior-point method was used. The algorithm was stopped once the duality gap was less than $10^{-4}$. The same machine setting was used in all the computational results reported in this paper. We consider random graphs of densities $p \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$. The dense model is used except for $p < 0.5$. The computationally most demanding instances are those with edge density $\approx 0.5$, i.e. with roughly an equal number of edges and non-edges.

**Remark 8** *Computational results for $\theta(G)$ are usually obtained using the dense model (14), see [1, 10, 21]. We are not aware of any computational study using the sparse model.*

### 6.1 $\theta^{+\triangle}$ in the dense model

Even though the tighter relaxations (6) and (7) for $\chi(G)$ have substantially more constraints than (14), we will see next that on dense graphs these relaxations can be solved with about the same effort as the basic model (14).
We first observe that any feasible matrix for (6) implies $y \geq 0$ in (14). This does not increase the number of (dual) equations in (14), but only changes some of the equations to inequalities.
The triangle inequalities in (7) for $ij, jk \notin E$ are either of the form

$$y_{ij} + y_{jk} \leq t \text{ in case } ik \in E \text{ or}$$

$$y_{ij} + y_{jk} - y_{ik} \leq t \text{ in case } ik \notin E.$$

We collect all these inequalities in $By \leq te$. Again, these constraints do not increase the number of equations in the dual of (14). More specifically, we get the following equivalent form of (7):

$$\begin{array}{lll} \theta^{+\triangle}(G) = & \min t & = \quad \max\langle J, X\rangle \\ & tI + A_{\bar{G}}^T(y) - J \succeq 0 & \quad \text{tr } X + e^T v = 1 \\ & y \geq 0 & \quad A_{\bar{G}}(X) + u - B^T v = 0 \\ & te - By \geq 0 & \quad X \succeq 0,\ u, v \geq 0 \end{array} \tag{16}$$

This formulation clearly shows that solving (16) should not take much longer than solving (14), because the number of dual equations is the same in both models.
In Table 2 we report computation times for $\theta^{+\triangle}$ using the model (16). Even though the primal SDP has additional constraints in the order of up to $\mathcal{O}(n^3)$, the number of dual equations is unchanged. The

| | n | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
|---|---|---|---|---|---|---|---|---|
| .25 | $\|E\|$ | 1240 | 2802 | | | | | |
| | time | 448 | 5208 | | | | | |
| .5 | $\|E\|$ | 2531 | 5665 | 10026 | | | | |
| | time | 89 | 1355 | 8989 | | | | |
| .75 | $\|E\|$ | 3734 | 8451 | 15049 | 23541 | | | |
| | time | 15 | 105 | 812 | 3932 | | | |
| .9 | $\|E\|$ | 4460 | 10079 | 17942 | 28127 | 40436 | 55099 | 71863 |
| | time | 2 | 11 | 41 | 184 | 652 | 1679 | 4029 |

Table 2: Times in seconds for computing $\theta^{+\triangle}(G)$ on random graphs.

| | n | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
|---|---|---|---|---|---|---|---|---|
| .1 | $\|E\|$ | 487 | 1137 | 2047 | 3149 | 4531 | 6098 | 7949 |
| | time | 4 | 27 | 138 | 469 | 1362 | 2966 | 7616 |
| .25 | $\|E\|$ | 1240 | 2802 | 5099 | | | | |
| | time | 33 | 247 | 1299 | | | | |

Table 3: Times in seconds for computing $\theta^{+\triangle}(G, N)$.

slightly higher computation times of Table 2 in comparison to Table 1 (dense model) are mostly due to the fact that (14) needs roughly 12 interior point iterations, while (16) needs about twice as many iterations. A more carefully selected starting point might reduce this number of iterations.

### 6.2 $\theta^{+\triangle}$ in the sparse model

Turning to the sparse model (15), the introduction of the sign constraints leads to additional inequalities

$$z_{ij} \geq -1 \; \forall ij \notin E,$$

and hence increases the number of primal constraints. For sparse graphs, the inclusion of all sign constraints may be too large for computational purposes.
We therefore follow a more pragmatic approach and consider adding the most violated sign constraints only. We first compute $\theta(G)$ and the optimal $Z$ using the sparse model, which can be done very fast. A good initial guess on violated sign constraints is now given by the most negative entries of $Z$. So (like in [21]) we define

$$\begin{aligned} \theta^{+\triangle}(G, N) = \min\{t : \quad & Z \succeq 0, \; te - \operatorname{diag} Z = e, \; z_{ij} = -1 \, \forall ij \in E, \\ & z_{ij} \geq -1 \, \forall ij \in N, z_{ij} + z_{jk} - z_{ik} \leq t - 1 \, \forall ij, jk \in N, ik \in E \cup N\} \end{aligned} \tag{17}$$

where $N \subset \bar{E}$. The computation times for this approach where $|N| = |V|$ are given in the Table 3. We note that the computation times are not much higher than the times for the sparse model from Table 1. It is also instructive to compare the computation times in Tables 2 and 3 for graphs with density 0.25.

**Remark 9** *A similar analysis holds for the relaxations of $\theta$ toward the clique number. In this case, the inclusion of the sign constraints leading to $\theta^-(G)$ does not increase the number of equations in the sparse model. Therefore the same approach can be used to make computations of $\theta^{-\triangle}(G)$ and $\theta^{-\triangle}(G, N)$ efficient. Further details are contained in the forthcoming dissertation [7].*

# 7 Exploiting symmetry

Exploiting symmetry often leads to significant simplifications. Parrilo and Sturmfels [25] exploit symmetry to reduce the number of variables in minimizing polynomials. Margot [18] uses isomorphisms for pruning in branch-and-cut algorithms. Recently, Schrijver [26] uses the Terwilliger algebra to simplify some bound calculations in coding theory.
The permutation $\alpha : V \to V$ is an **automorphism** of the graph $G = (V, E)$, if

$$ij \in E \iff (\alpha(i)\alpha(j)) \in E$$

Aut $G = \{\alpha : \alpha$ automorphism of $G\}$ denotes the set of all automorphisms of $G$. It is well known that Aut $G$ is a group with respect to composition.
Notice that Aut $G$ = Aut $\bar{G}$. Though the problem of finding all automorphisms of a given graph is graph isomorphism complete, a very efficient heuristic named NAUTY can find large subgroups of Aut $G$ [20]. Also many important families of graphs have rich while easy-to-compute groups of automorphisms (Hamming, Johnson, Kneser, circular graphs).
Let $A \leq$ Aut $G$ be a subgroup of automorphisms of the graph $G$. We denote by

$$[i] := \{\alpha(i) : \alpha \in A\}$$

the orbit of a vertex $i$, and by

$$[ij] := \{(\alpha(i)\alpha(j) : \alpha \in A\}$$

the orbit of edge (or non-edge) $ij$.

**Theorem 10** *Let $[E] = \{[ij] : ij \in E\}$, $[\bar{E}] = \{[ij] : ij \in \bar{E}\}$ and $[V] = \{[i] : i \in V\}$ be the sets of all orbits of edges, non-edges and vertices for a given group $A \leq Aut\ (G)$ of automorphisms. Then*

$$\begin{array}{rl} \theta(G) = & \min t \\ & tI + \sum_{[ab]\in[\bar{E}]} \bar{y}_{[ab]}(\sum_{ij\in[ab]} E_{ij}) \succeq J \end{array} \tag{18}$$

*and also*

$$\begin{array}{rl} \theta(G) = & \max\langle \bar{X}, J\rangle \\ & tr\ \bar{X} = 1 \\ & \bar{X} = \sum_{[a]\in[V]} \bar{x}_{[a]}(\sum_{i\in[a]} E_{ii}) + \sum_{[ab]\in[E]} \bar{x}_{[ab]}(\sum_{ij\in[ab]} E_{ij}) \succeq 0 \end{array} \tag{19}$$

**Remark 11** *Models (18) and (19) have only one variable per orbit. So in the case of a rich automorphism group they need considerably less variables than (14) and (15). Note in particular that $\bar{x}_{ij} = \bar{x}_{\beta(i)\beta(j)}$ implies $\bar{X} = P_\beta \bar{X} P_\beta^T$ for each $\beta \in A$.*

Proof. We show that for every feasible solution of (14) there is a feasible solution (20) of (18) with the same value of the cost function. Since any feasible solution of (18) is already feasible for (14) this establishes equality.
Let $Y = tI + A_{\bar{G}}^T(y) \succeq J$ be a feasible solution of (14). Denote by $P_\alpha$ the permutation of $V$ such that $P_\alpha e_i = e_{\alpha(i)} \forall i \in V$. Then $P_\alpha E_{ij} P_\alpha^T = E_{\alpha(i)\alpha(j)}$.
Let us shorthand the average

$$0 \preceq \frac{1}{|A|}\sum_{\alpha\in A} P_\alpha Y P_\alpha^T = tI + \frac{1}{|A|}\sum_{\alpha\in A}\sum_{ij\in\bar{E}} y_{ij} P_\alpha E_{ij} P_\alpha^T = tI + \frac{1}{|A|}\sum_{\alpha\in A}\sum_{ij\in\bar{E}} y_{ij} E_{\alpha(i)\alpha(j)} \tag{20}$$

by $tI + \sum_{ij\in\bar{E}} \bar{y}_{ij} E_{ij}$. Since $A$ is a group, $A = \{\alpha \circ \beta^{-1} : \alpha \in A\}$ proving $\bar{y}_{ij} = \frac{1}{|A|}\sum_{\alpha\in A} y_{\alpha^{-1}(i)\alpha^{-1}(j)}$ $= \frac{1}{|A|}\sum_{\alpha\in A} y_{\alpha^{-1}\circ\beta(i)\alpha^{-1}\circ\beta(j)} = \bar{y}_{\beta(i)\beta(j)}$ for any $\beta \in A$.
Forming analogous averages in (15) establishes (19). □

### 7.1 Strengthenings of Schrijver and Szegedy

Now notice that if in (18) $y \geq 0$, the final average $\bar{y} \geq 0$, too (while the cost function equals $t$). To see that the Szegedy's number equals

$$\begin{array}{rl} \theta^+(G) = & \min t \\ & \bar{Z} = tI + \sum_{[ab]\in[\bar{E}]} \bar{y}_{[ab]} (\sum_{ij\in[ab]} E_{ij}) \succeq J \\ & y_{[ab]} \geq 0 \quad \forall [ab] \in [\bar{E}] \end{array} \tag{21}$$

one only needs to notice that all feasible solutions of (21) are also feasible for (6).
The same argument shows that the Schrijver's number equals

$$\begin{array}{rl} \theta^-(G) = & \max \langle \bar{X}, J \rangle \\ & \text{tr } \bar{X} = 1 \\ & \bar{X} = \sum_{[a]\in[V]} \bar{x}_{[a]} (\sum_{i\in[a]} E_i) + \sum_{[ab]\in[E]} \bar{x}_{[ab]} (\sum_{ij\in[ab]} E_{ij}) \succeq 0 \\ & \bar{x}_{[AB]} \geq 0 \quad \forall [ab] \in [E] \end{array} \tag{22}$$

Since the number of the orbits is typically much smaller then $n^2$, this bound can be computed very efficiently.

### 7.2 Strengthening by the triangle inequalities

Strengthening (7) involves a large number of up to $\mathcal{O}(n^3)$ of additional triangle inequalities. After "averaging" $Z$ (like in (20)) they are transformed into

$$\bar{y}_{[ij]} + \bar{y}_{[jk]} \leq t \text{ in case } [ik] \in [E] \text{ or}$$

$$\bar{y}_{[ij]} + \bar{y}_{[jk]} - \bar{y}_{[ik]} \leq t \text{ in case } [ik] \notin [E].$$

Also, if $[ik] \in \{[ij], [jk]\}$ the constraint becomes redundant. Similar ideas apply to the model (12).

### 7.3 The bounds for vertex-transitive graphs

The graph $G = (V, E)$ is **vertex transitive**, if $\forall i, j \in V \; \exists \alpha \in A$ such that $\alpha(i) = j$.
If $G$ is vertex transitive, by taking $A := \text{Aut } G$ (19) simplifies to

$$\begin{array}{rl} \theta(G) = & \max \langle \bar{X}, J \rangle \\ & \bar{X} = \frac{1}{n} I + \sum_{[ab]\in[E]} \bar{x}_{[ab]} \sum_{ij\in[ab]} E_{ij} \succeq 0 \end{array} \tag{23}$$

Any feasible solution $\bar{X}$ of (23) with the value of the cost function $\langle \bar{X}, J \rangle = e^T \bar{X} e =: k$ can be transformed into a feasible solution of (18)

$$\bar{Y} := \frac{n^2}{k} \bar{X} \tag{24}$$

for the complementary graph $\bar{G}$ with the cost $t = \frac{n^2}{k} \frac{1}{n} = \frac{n}{k}$. Remark 11 now implies that $e$ is an eigenvector of any feasible solution $\bar{X}$ of (23). Since $\bar{X} e = \frac{k}{n} e$, $\bar{Y} \succeq J$. The inverse transformation

$$\bar{X} := \frac{1}{tn} \bar{Y} \tag{25}$$

transforms feasible primal solution of (18) for the graph $\bar{G}$ into a feasible dual solution of (19) for graph $G$ regardless if the graph is vertex transitive or not.
Transformations (24) and (25) carry sign constraints of (6) into the sign constraints of (11), and vice versa. Moreover the triangle inequalities of $\theta^{+\triangle}(G)$ get transformed into the triangle inequalities of $\theta^{-\triangle}(\bar{G})$, and on vertex transitive graphs also vice versa, since the inequalities $\bar{x}_{[ij]} \leq \bar{x}_{[ii]} = \frac{1}{n}$ are in the positive semidefinite matrix $\bar{X}$ redundant.
This proves

| density \ | n | 100 | 150 | 200 | 250 | 300 | 350 | 400 |
|---|---|---|---|---|---|---|---|---|
| .1 | $\theta^{-\triangle}(G)$ | 4.0000 | 5.0000 | 4.4428 | 4.8014 | 5.1217 | 5.3839 | 5.7118 |
| | $\theta(G)$ | 4.0000 | 5.0000 | 4.4471 | 4.8047 | 5.1275 | 5.3893 | 5.7166 |
| | $\theta^{+\triangle}(G,N)$ | 4.0000 | 5.0000 | 4.4681 | 4.8256 | 5.1505 | 5.4137 | 5.7403 |
| | $\theta^{+\triangle}$ | 4.0000 | | | | | | |
| .25 | $\theta^{-\triangle}(G)$ | 5.8056 | 6.8409 | 7.7852 | 8.5052 | | | |
| | $\theta(G)$ | 5.8228 | 6.8636 | 7.8104 | 8.5328 | | | |
| | $\theta^{+\triangle}(G,N)$ | 5.8571 | 6.9055 | 7.8568 | | | | |
| | $\theta^{+\triangle}(G)$ | 5.8685 | 6.9194 | | | | | |
| .5 | $\theta^{-\triangle}(G)$ | 10.7476 | 12.8249 | | | | | |
| | $\theta(G)$ | 10.8246 | 12.9034 | 14.6807 | | | | |
| | $\theta^{+\triangle}(G)$ | 10.8976 | 12.9998 | 14.7830 | | | | |
| .75 | $\theta^{-\triangle}(G)$ | 19.2699 | | | | | | |
| | $\theta(G)$ | 19.4945 | 24.3363 | 28.6875 | 31.8306 | | | |
| | $\theta^{+\triangle}(G)$ | 19.5539 | 24.4414 | 28.7988 | 31.9376 | | | |
| .9 | $\theta^{-\triangle}(G)$ | 32.3485 | | | | | | |
| | $\theta(G)$ | 33.1647 | 41.9934 | 50.1470 | 58.0653 | 63.0662 | 69.4530 | 73.9517 |
| | $\theta^{+\triangle}(G)$ | 33.2152 | 42.0462 | 50.2200 | 58.1541 | 63.1388 | 69.5523 | 74.0238 |

Table 4: The bounds on $\omega(G)$ and $\chi(G)$ on random graphs

**Theorem 12** *Let $G$ be a simple graph on $n$ vertices. Then*

$$\theta(G)\theta(\bar{G}) \geq n \qquad \theta^{+}(G)\theta^{-}(\bar{G}) \geq n \qquad \theta^{+\triangle}(G)\theta^{-\triangle}(\bar{G}) \geq n$$

*If $G$ is vertex transitive, then all these inequalities are tight.* □

See [14] for an alternative proof of $\theta(G)\theta(\bar{G}) \geq n$ for general graphs and $\theta(G)\theta(\bar{G}) = n$ for the vertex transitive graphs, while Szegedy proved $\theta^{+}(G)\theta^{-}(\bar{G}) \geq n$ in [28]. The proof of the theorem implies that solving the SDP relaxation on a vertex transitive graph $G$ produces the bound on $\bar{G}$ as well as a corresponding matrix variable, i.e. these two problems are equivalent from the standpoint of the applications, see section 8.4.

# 8 Computational results

We have discussed the sparse (15) and the dense (14) model to compute $\vartheta(G)$. In Table 1 we focused on the computational efficiency of these methods. Now we take a closer look at the quality of the various relaxations to approximate both $\chi(G)$ and $\omega(G)$.

## 8.1 Comparing $\vartheta(G)$ and $\vartheta(G)^{+\triangle}$ on random and DIMACS graphs

We have seen in Tables 1 and 2 that the effort to compute $\vartheta(G)$ and $\vartheta(G)^{+\triangle}$ is comparable for dense graphs ($m > \frac{n^2}{4}$). In Table 4 we compare these bounds on randomly generated graphs of given densities. It turns out that the inclusion of the additional sign constraints (6) and the triangle inequalities (7) does not change the value of the relaxation in a significant way. In fact, in all these instances we have

$$\frac{\vartheta(G^{+})}{\vartheta(G)} < 1.01.$$

This is consistent with the very preliminary experience reported by Meurdesoif [21].

The situation is not much different if we look at some of the dense instances from the DIMACS collection, see Tables 5 and 6. To get some impression also for sparse graphs, we consider also including a limited number of the additional constraints in the columns labeled $\theta^{+\triangle}(G,N)$, as described in section 6.2. Even though the difference between the relaxations is a little bit bigger than for completely random graphs, notably in case of the Mycielski graphs, the increment of the lower bound toward $\chi(G)$ is still only marginal.

| name | n | m | $\lvert\bar{E}\rvert$ | $\theta(G)$ | $\theta^{+}(G)$ | $\theta^{+\triangle}(G)$ | $\theta^{+\triangle}(G,N)$ |
|---|---|---|---|---|---|---|---|
| myciel5 | 47 | 236 | 845 | 2.6387 | 2.6387 | 3.0933 | 2.7583 |
| myciel6 | 95 | 755 | 3710 | 2.7342 | 2.7342 | 3.2538 | 2.8340 |
| 1-insertions_4 | 67 | 232 | 1979 | 2.2333 | 2.2333 | 2.5230 | 2.2634 |
| 4-insertions_3 | 79 | 156 | 2925 | 2.0480 | 2.0480 | 2.1818 | 2.0701 |
| 1-FullIns_4 | 93 | 593 | 3685 | 3.1244 | 3.1244 | 3.4869 | 3.2657 |
| 2-FullIns_3 | 52 | 201 | 1125 | 4.0282 | 4.0282 | 4.2408 | 4.2408 |
| 3-FullIns_3 | 80 | 346 | 2814 | 5.0158 | 5.0158 | 5.1935 | 5.1198 |
| dsjc125.5 | 125 | 3891 | 3859 | 11.7844 | 11.8674 | 11.8674 | 11.8555 |
| dsjc125.9 | 125 | 6961 | 789 | 37.7678 | 37.8028 | 37.8031 | |
| dsjc250.9 | 250 | 27897 | 3228 | 55.1527 | 55.2155 | 55.2156 | |

Table 5: Computational results for some small DIMACS Graph Coloring Problems.

| name | n | m | $\theta(G)$ | $\theta^{\triangle}(G,N)$ | $\chi(G)$ |
|---|---|---|---|---|---|
| myciel7 | 191 | 2360 | 2.8146 | 2.9206 | 8 |
| 1-insertions_5 | 202 | 1227 | 2.2765 | 2.3084 | 6 |
| 2-insertions_4 | 149 | 541 | 2.1334 | 2.1802 | 5 |
| 3-insertions_4 | 281 | 1046 | 2.0868 | 2.1229 | 5 |
| 4-insertions_4 | 475 | 1795 | 2.0612 | 2.0893 | 5 |
| 1-FullIns_5 | 282 | 3247 | 3.1811 | 3.2935 | 6 |
| 2-FullIns_4 | 212 | 1621 | 4.0559 | 4.3021 | 6 |
| 4-FullIns_3 | 114 | 541 | 6.0100 | 6.1612 | 7 |
| 5-FullIns_3 | 154 | 792 | 7.0068 | 7.0811 | 8 |
| dsjc125.1 | 125 | 736 | 4.1061 | 4.2084 | 5 |
| dsjc250.1 | 250 | 3218 | 4.9063 | 4.9317 | $\leq 9$ |

Table 6: Computational results for some medium-sized DIMACS Graph Coloring Problems.

### 8.2 Comparing $\theta$ and $\theta^{-\triangle}$

A similar situation occurs, if one looks at the improvements of $\vartheta(G)$ toward the clique number. We collect some representative results in Table 7. The improvement of the random graphs labeled dsjc is small while the situation changes significantly, once we look at highly structured generalized Hamming graphs labeled $H$. The vertex set of a Hamming graph $H(a,b,c)$ consists of numbers $0,...,b^a-1$ written in base $b$. The number of different digits is the Hamming distance between two numbers. Two vertices $u,v$ are connected, if the Hamming distance $d(u,v)=c$, respectively $d(u,v)\leq c$ in $H^{-}(a,b,c)$, and $d(u,v)\geq c$ in $H^{+}(a,b,c)$. These graphs are important in coding theory.

| name | n | m | $\leq \omega(G)$ | $\theta^{-\triangle}(G)$ | $\theta(G)$ |
|---|---|---|---|---|---|
| $H^-(5,2,2)$ | 32 | 240 | 4 | 4.0000 | 5.3333 |
| $H^-(6,2,2)$ | 64 | 1312 | 4 | 4.0000 | 5.3333 |
| $H^-(7,2,2)$ | 128 | 1856 | 2 | 3.0000 | 3.5556 |
| $H^-(8,2,2)$ | 256 | 4736 | 2 | 3.0000 | 3.5556 |
| $\overline{\text{dsjc125.5}}$ | 125 | 3859 | 10 | 11.4019 | 11.4730 |
| dsjc125.5 | 125 | 3891 | 10 | 11.7105 | 11.7844 |
| dsjc125.1 | 125 | 736 | 4 | 4.0671 | 4.1062 |

Table 7: Computational results for the Clique Number on some DIMACS Stable Set Problems

| name | n | m | or. | cliq. | $\theta^{-\triangle}(G)$ | $\theta(G)$ | $\theta^{+}(G)$ | $\theta^{+\triangle}(G)$ | col. | time |
|---|---|---|---|---|---|---|---|---|---|---|
| H(6,2,2) | 64 | 480 | 1 | 4 | 4.0000 | 4.0000 | 5.3333 | 5.3333 | 8 | 0 |
| H(6,2,4) | 64 | 480 | 1 | 6 | 6.0000 | 6.0000 | 8.0000 | 8.0000 | 8 | 0 |
| H(9,2,3) | 512 | 21504 | 8 | 4 | 4.0000 | 4.0000 | 6.4000 | 6.4000 | 11 | 40 |
| H(10,2,2) | 1024 | 23040 | 9 | 2 | 2.6667 | 2.6667 | 3.2000 | 3.2000 | 7 | 390 |
| H(11,2,3) | 2048 | 168960 | 10 | 2 | 3.2000 | 3.2000 | 4.9382 | 4.9382 | 12 | 3036 |
| $H^+(9,2,3)$ | 512 | 119040 | 3 | 130 | 160.0000 | 160.0000 | 192.0000 | 192.0000 | 256 | 23 |
| $H^+(9,2,4)$ | 512 | 97536 | 4 | 74 | 80.0000 | 80.0000 | 128.0000 | 128.0000 | 128 | 29 |
| $H^+(11,2,5)$ | 2048 | 1520640 | 5 | 232 | 265.8461 | 265,8461 | 512.0000 | 512.0000 | 512 | 1888 |
| $H^+(12,2,5)$ | 4096 | 6760448 | 5 | 462 | 531.6923 | 531.6923 | 1024.0000 | 1024.0000 |  | 25950 |
| H(6,3,3) | 729 | 58320 | 5 | 9 | 9.0000 | 9.0000 | 11.5714 | 11.5714 | 22 | 105 |
| H(7,3,4) | 2187 | 306180 | 6 | 9 | 9.0000 | 9.0000 | 11.5714 | 11.5714 | 32 | 2980 |
| J(10,5,2) | 252 | 12600 | 4 | 6 | 6.0000 | 6.0000 | 8.2500 | 8.2500 | 12 | 4 |
| J(12,5,3) | 792 | 83160 | 4 | 7 | 15.0000 | 15.0000 | 22.0000 | 22.0000 | 39 | 119 |
| J(12,7,3) | 792 | 69300 | 4 | 3 | 3.6924 | 3.6923 | 6.6000 | 6.6000 | 12 | 114 |
| J(14,7,3) | 3432 | 2102100 | 6 | 8 | 8.0000 | 8.0000 | 11.8182 | 11.8182 |  | 16568 |
| $C_{97}$ | 97 | 97 | 47 | 2 | 2.0001 | 2.0005 | 2.0005 | 2.0208 | 3 | 2 |
| $\overline{(C_5)^4}$ | 625 | 170000 | 4 | 21 | 25.0000 | 25.0000 | 25.0000 | 27.9508 | 62 | 44 |
| $\overline{(C_7)^4}$ | 2401 | 2785160 | 4 | 88 | 121.1521 | 121.1521 | 121.1521 | 127.9508 | 233 | 2364 |
| $\overline{(C_9)^3}$ | 729 | 255879 | 3 | 67 | 82.8870 | 82.8870 | 82.8870 | 85.5466 | 119 | 51 |

Table 8: Computational results on vertex transitive graphs.

### 8.3 Numerical results on vertex transitive graphs

We have just seen that on unstructured random graphs, the variations of $\theta(G)$ do not lead to significantly improved bounds. The situation is quite different, once we consider structured graphs. In Table 8, we give results on some vertex transitive graphs. We consider Hamming graphs, Johnson graphs, and some complements of powers of cycles. In particular, we provide information about the graph ($n$, $m$, number of orbits). The clique size in column labeled cliq. is obtained by an iterated greedy heuristic while the number of colors in column labeled col. is obtained by heuristic TABU [12]. This heuristic cannot handle the two largest graphs (with more than 3000 vertices), so we have no good estimate of $\chi$ for these two graphs. While the number of edges can be very large, only the number of orbits in columns labeled or. effects the computation times. It is remarkable that the inclusion of the sign constraints nearly doubles the bound on some of the $H^+$ graphs. The results in this table also suggest that the additional inclusion of the triangle inequalities does not improve $\theta^+$, except for the last instances.

### 8.4 Extracting colorings and cliques

Besides computing a bound on $\chi(G)$ or $\omega(G)$ the SDP approach also produces matrix variables ($Z$ and $X$) which can be utilized for finding suboptimal coloring [13] or extracting large clique, see [2]. The former idea can be generalized to approximately solving max-$k$-cut problem, see [8]. The search for stronger bounds is mainly motivated by the hope that this matrix variables will better approximate optimal coloring, clique or max-$k$-cut, respectively. In fact, we have found 8-colorings of H(6,2,2) and H(6,2,4) by utilizing the strengthened bound $\theta^{+\triangle}$ in a variant of Karger-Motwani-Sudan graph coloring heuristic while applying Lovász theta number instead produced a 10-coloring both times. Also $\omega(J(12,7,3)) = 7$ was computed by an exact branch and bound algorithm applying $\omega^{-\triangle}$ on the non-isomorphic subgraphs.

## 9 Conclusions

Considering all these computational results, we draw the following conclusions.

- The Lovász number can be computed efficiently on very sparse and very dense graphs.
- On random graphs the strengthened bounds do not improve the original Lovász number considerably.
- Exploiting symmetry can simplify computations significantly.
- On some vertex transitive graphs the non-negativity constraints yield a substantial improvement over the original relaxation.
- The additional inclusion of the triangle constraints often does not improve upon the simpler relaxation $\theta^{+}$.

Our numerical results have been obtained by computer programs downloadable from www.math.uni-klu.ac.at/or. Further research is necessary to investigate the computation of these bounds by faster first order methods like [5, 11, 29].

## References


[1] S. BENSON, Y. YE, and X. ZHANG. Solving large-scale sparse semidefinite programs for combinatorial optimization. *SIAM Journal on Optimization*, 10:443–461, 2000.

[2] S. BENSON and Y. YE. Approximating maximum stable set and minimum graph coloring problems with the positive semidefinite relaxation, working paper, University of Iowa. 1999.

[3] S. BURER, D. VANDENBUSSCHE. Solving lift-and-project relaxations of binary integer programs, working paper, Optimization Online, 2004.

[4] I. M. BOMZE and E. de KLERK. Solving standard quadratic optimization problems via linear, semidefinite and copositive programming. pages 1–19, 2001.

[5] S. BURER and R. D. C. MONTEIRO. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95:329–357, 2003.

[6] M. CHARIKAR. On semidefinite programming relaxations for graph coloring and vertex cover. In *Proceedings of the 41th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 616–620, 2002.

[7] I. DUKANOVIC. *Semidefinite Programming applied to Graph Coloring Problem.* PhD thesis, University of Klagenfurt, Austria, 2005 forthcoming.

[8] E. de KLERK, D. V. PASECHNIK, and J. P. WARNERS. On approximate graph colouring and max-k-cut algorithms based on the $\theta$-function. *Manuscript*, pages 1–19, 2000.

[9] M. GRÖTSCHEL, L. LOVÁSZ, and A. SCHRIJVER. *Geometric Algorithms and Combinatorial Optimization.* Springer New York, 1988.

[10] G. GRUBER and F. RENDL. Computational experience with stable set relaxations. *SIAM J. Optimization*, 13:1014–1028, 2003.

[11] C. HELMBERG and F. RENDL. A spectral bundle method for semidefinite programming. *SIAM Journal on Optimization*, 10:673–696, 2000.

[12] A. HERTZ and DE WERRA D. Using tabu search for graph coloring. *Computing*, 39:345–351, 1987.

[13] D. KARGER, R. MOTWANI, and M. SUDAN. Approximate graph coloring by semidefinite programming. *Journal of the Association for Computing Machinery*, 45:246–265, 1998.

[14] D. E. KNUTH. The sandwich theorem. *Electron. J. Combin.*, 1:1–48, 1994.

[15] M. LAURENT and F. RENDL. Semidefinite programming and integer programming. In K. Aardal et al., editor, *Handbook in OR & MS*, pages 393–514. Elsevier B. V., forthcoming.

[16] L. LOVÁSZ. On the Shannon capacity of a graph. *IEEE Trans. Inform. Theory*, 25:1–7, 1979.

[17] L. LOVÁSZ and A. SCHRIJVER. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, 1:166–190, 1991.

[18] F. MARGOT. Pruning by isomorphism in branch-and-cut. *Lecture Notes in Computer Science*, 2081:304–317, 2001.

[19] Jr. R.J. McELIECE, E.R. RODEMICH, and H.C. RUMSEY. The Lovász bound and some generalizations. *Journal of Combinatorics and System Sciences*, 3:134–152, 1978.

[20] B. D. McKAY. Practical graph isomorphism. *Congressus Numerantium*, 30:45–87, 1981.

[21] P. MEURDESOIF. Strengthening the Lovász $\theta(\bar{G})$ bound for graph coloring. *Mathematical Programming (forthcoming).*

[22] K. G. MURTY. Some NP-complete problems in quadratic and nonlinear programming. *Mathematical Programming*, 39:117–129, 1987.

[23] P. A. PARRILO. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization.* PhD thesis, California Institute of Technology, 2000.

[24] P. A. PARRILO. Symmetries in semidefinite programming, and how to exploit them. *Program of the First MPS Conference on Continuous Optimization ICCOPT I, Troy*, page 51, 2004.

[25] P. A. PARRILO and B. STURMFELS. Minimizing polynomial functions. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 1–16, 2001.

[26] A. SCHRIJVER. New code upper bounds from the Terwilliger algebra. Technical report, CWI, 2004.

[27] A. SCHRIJVER. A comparison of the Delsarte and Lovász bounds. *IEEE Transactions on Information Theory*, IT-25:425–429, 1979.

[28] M. SZEGEDY. A note on the theta number of Lovász and the generalized Delsarte bound. *35th Annual Symposium on Foundations of Computer Science*, pages 36–39, 1994.

[29] K.C. TOH and M. KOJIMA. Solving some large scale semidefinite programs via the conjugate residual method. *SIAM Journal on Optimization*, 12:669–691, 2002.