Approximating the stability number and the chromatic number of a graph via semidefinite programming
Gvozdenovic, N.
Chapter 1

Introduction

In this thesis, we consider two well known graph parameters: the stability number and the chromatic number of a graph. We study semidefinite relaxations of integer and copositive programs defining these parameters. We extensively use techniques, known in the literature as block diagonalization and symmetry reduction, for reducing the sizes of matrices and the number of variables in semidefinite programs.

We give first a brief sketch of the results presented in this work, and go back to details in Section 1.3 after introducing background in Section 1.2.

1.1 Overview of results

We compare several hierarchies of semidefinite programming upper bounds for the stability number of a graph. The first order bounds in all these hierarchies coincide with the Lovász theta number. Combining the approaches of Lovász and Schrijver [65], and Lasserre [56, 57], we define a new hierarchy. As a relaxation of the hierarchy of Lasserre, it has an advantage that the semidefinite programs defining its bounds can be block diagonalized. Moreover, it is less costly and at least as strong as the hierarchy of Lovász and Schrijver. Besides, we introduce the hierarchy of de Klerk and Pasechik [50] and show that it is dominated by the hierarchy of Lasserre.

We next define and study the corresponding hierarchies of lower bounds for the (fractional) chromatic number. We introduce a special operator $\Psi$ which maps upper bounds for the stability number to lower bounds for the chromatic number. As an application, we prove that there is no polynomial time computable graph parameter nested between the fractional chromatic and the chromatic number of a graph, unless P=NP.

We compute bounds in the new block diagonal hierarchy for some interesting graph classes. In particular, we are able to compute the bounds, up to order three, for Paley graphs with at most 800 vertices, using the properties of their automorphism groups; and the bounds, of order one and two, for Hamming and Kneser graphs with up to $2^{20}$ vertices, using the explicit block diagonalization of the Terwilliger algebra of the Hamming scheme given by Schrijver in [85]. Finally, we introduce yet another variation of the second order bound in the
hierarchy of Lasserre via a semidefinite program which can be also block diagonalized, and we report computational results for some DIMACS benchmark instances.

1.2 Background and motivation

Graphs, stable sets and colourings

A graph consists of vertices and edges. An edge connects two vertices. A stable set is a set of vertices of a graph in which no two of them are connected with an edge. An assignment of colours to the vertices of a graph, such that no two connected vertices share same colour, is called a vertex colouring. Stable sets and vertex colourings are closely related. It is straightforward to see that vertex colouring of a graph is equivalent to partitioning of the set of vertices into stable sets.

Some problems of practical interest can be modelled as stable set or colouring problems, e.g. time tabling, scheduling, frequency assignment, register allocation, pattern matching or coding. In these applications, one is usually interested in finding a maximum-size stable set in a graph or a vertex colouring of a graph which uses the least possible number of colours.

For instance, one of the fundamental problems in coding theory is finding a code, i.e. a subset of possible words which differ from each other significantly, of a maximum size. Here words are all sequences of letters, from a given alphabet $\mathcal{A}$, of some predefined length $n$. Their difference is usually quantified as the number of places in which they differ, called the Hamming distance. Thus, given a positive integer $d$, two words differ significantly if their Hamming distance is at least $d$. Consider the graph whose vertices are the words, i.e. the elements of $\mathcal{A}^n$, two of them being connected with an edge if their Hamming distance is smaller than $d$. Finding an optimal code is now equivalent to finding a maximum size stable set in this graph.

The stability number $\alpha(G)$ of a graph $G$ is the cardinality of a maximum size stable set in the graph. The chromatic number $\chi(G)$ of $G$ is the minimum number of colours that have to be used in a vertex colouring of the graph. Determining $\alpha(G)$ and $\chi(G)$ are hard combinatorial optimization problems.

Complexity and combinatorial optimization

Optimization problems are problems in which one tries to find a best solution, satisfying certain properties, with respect to a given criterion. It is common to express the criterion of a problem as a function, called an objective function. The goal is then to find its optimal value (usually the maximum or the minimum) on a given domain, known as the feasibility domain or the set of feasible solutions. A combinatorial optimization problem is an optimization problem whose set of feasible solutions is finite. One of the most important issues when dealing with combinatorial optimization problems is their complexity.

A decision problem is a question whose answer depends on some input parameters and can be either ‘yes’ or ‘no’. When we fix input parameters we get an instance of a decision problem. An oracle for a decision problem is a machine
(black-box) which is able to solve it in a single operation. The complement of a
decision problem is the decision problem resulting from reversing the ‘yes’ and
‘no’ answers.

If a ‘yes’ answer to a decision problem is provided with a certificate, which
can be checked in polynomial time (in the size of the input), the problem is said
to belong to the class NP (‘Non-deterministic Polynomial time’). For example,
given a graph $G$ and a positive integer $k$, consider the following two problems:

(S) Does there exist a stable set in $G$ of size at least $k$?

(C) Does there exist a vertex colouring in $G$ which uses at most $k$ colours?

The question (S), known also as the ‘stable set problem’, is in NP. Namely,
a stable set of size at least $k$ is a certificate for a ‘yes’ answer since we can
quickly check that it contains at least $k$ vertices and that no two of them are
connected. Accordingly, a vertex colouring which uses at most $k$ colours is a
certificate for a ‘yes’ answer of the problem (C) since we can quickly check that
no two connected vertices received same colour. Hence the problem (C), known
as the ‘colouring problem’, is also in NP.

A decision problem is said to belong to the class co-NP if its complement is
in NP.

A problem is said to be in class P (‘Polynomial time’) if it can be solved,
i.e. the correct answer can be found, in polynomial time. It is also common
to say that such a problem is ‘easy’. There exist decision problems in NP for
which we still do not know if they are easy or not. In other words, we do not
know if $P=NP$. This is considered to be the most important open question in
complexity theory.

A decision problem $A$ can be reduced to a decision problem $B$ in polynomial
time if there exists a polynomial time algorithm $f$ which transforms instances
of $A$ into instances of $B$, such that for any instance $a$ of $A$ the answer to the
instance $f(a)$ of $B$ is ‘yes’ if and only if the answer to the instance $a$ is ‘yes’. As
an example we give a transformation from the colouring problem to the stable
set problem. Given a graph $G$ with $n$ vertices and a nonnegative integer $k$, make
$k$ copies of $G$, for every vertex of $G$ connect all pairs of its copies and call the
constructed graph $G_k$. Then, $G$ can be coloured with $k$ colours if and only if
there exists a stable set in $G_k$ of size (at least) $n$ (see Section 2.5 for details).

If a problem is in NP, and every other problem from NP can be reduced to
it in polynomial time, the problem is said to be NP-complete. A problem $A$ is
said to be NP-hard if and only if there is an NP-complete problem $B$ that can
be solved in polynomial time with an oracle for $A$.

The problem of determining if for a combinatorial optimization problem
there exists a feasible solution, with the objective value of a given quality (usu-
ally greater or smaller than some prescribed threshold value), is the decision
counterpart of the combinatorial optimization problem. Problem (S) is thus the
decision counterpart of the problem of finding $\alpha(G)$, and problem (C) is the
decision counterpart of the problem of finding $\chi(G)$. Combinatorial optimization
problems whose decision counterparts are NP-complete are NP-hard, i.e., they
are at least as hard as any problem in NP. The problems of determining the stability number and the chromatic number of a graph are both NP-hard (cf. [30]).

**Combinatorial optimization and semidefinite relaxations**

We cannot expect to find a polynomial time algorithm for an NP-hard combinatorial optimization problem. Still, we can try to solve it approximately by considering some relaxation of it, for which an efficient algorithm exists. In this thesis, we consider two approaches for modelling combinatorial optimization problems and related semidefinite relaxations. The main motivation for using semidefinite relaxations is the fact that semidefinite programs can be solved in polynomial time (up to a certain precision, see Subsection 2.3.2 for more details).

**Semidefinite programs** are programs in which one aims to optimize a linear function over the intersection of an affine subspace and a cone of semidefinite matrices. It is common to write a semidefinite program as

$$\min \langle C, X \rangle \quad \text{subject to} \quad \langle A_j, X \rangle = b_j \quad (j = 1, \ldots, m),$$

and $X \in \mathbb{R}^{n \times n}$ is positive semidefinite, \hspace{1cm} (1.1)

where $\langle \cdot, \cdot \rangle$ denotes the standard inner product on $\mathbb{R}^{n \times n}$.

The set of positive semidefinite matrices in $\mathbb{R}^{n \times n}$, known as the semidefinite cone, is convex. Semidefinite programs thus belong to the class of convex optimization problems, and moreover, they generalize linear, quadratic and second order cone programming problems. If we restrict, for example, the matrix $X$ in (1.1) to be diagonal, we obtain a linear program.

**Integer programming approach.** A classical approach is to model a combinatorial optimization problem as an integer linear program

$$\max c^T x \quad \text{subject to} \quad Ax \leq b, \quad x \in \{0,1\}^n,$$  \hspace{1cm} (1.2)

where $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. The hardness of the problem is then hidden in the constraint $x \in \{0,1\}^n$. (In general, an integer linear program is NP-hard. See e.g. [30].)

The way to define a basic semidefinite relaxation for (1.2) is to introduce the matrix variable $X = \binom{1}{x} \binom{1}{x}^T$. We have that $x \in \{0,1\}^n$ if and only if the first row of $X$ equals its diagonal. The next step is to express the objective function $c^T x$ and the linear constraints $Ax \leq b$, respectively, as a linear function and linear constraints in terms of the entries of the matrix $X$. (It can be done in several ways, see e.g. [61].) In other words, we can rewrite (1.2) in terms of $X$, where instead of the condition $x \in \{0,1\}^n$ we require that $X$ is a symmetric, rank one matrix, whose first row equals its diagonal, and whose left upper corner equals one. In this way, the hardness of the problem is moved into the ‘rank one’ constraint. Finally, the basic semidefinite relaxation is then obtained by dropping the rank constraint, and by requiring positive semidefiniteness for $X$ instead. Note that, apart from the semidefinite constraint, all constraints are then linear.
The first such semidefinite relaxation was proposed by Lovász for the stable set problem in his seminal paper [64]. He introduced a parameter \( \vartheta(G) \) of a graph \( G \), nowadays called the Lovász theta number, nested between the stability number \( \alpha(G) \) and the chromatic number \( \chi(G) \) of the complement \( \overline{G} \) of \( G \). We will study this bound in detail in Chapter 3, and we will see in Chapters 4 and 5 how one can strengthen it to obtain stronger semidefinite bounds for \( \alpha(G) \) and \( \chi(G) \).

Another breakthrough result, obtained by using basic semidefinite relaxations for combinatorial optimization problems, is the paper [34] by Goemans and Williamson. They presented an approximation algorithm for the max cut problem\(^1\). Given a graph instance, the algorithm finds a solution whose (expected) objective value is not less than 0.878 times the optimum.

**Copositive programming approach.** A more recent approach is based on modelling combinatorial optimization problems as copositive programs. In a copositive program the goal is to minimize a linear function with respect to linear constraints. The variable is a square matrix, restricted to be copositive. In this approach, the hardness of a problem is put into the copositivity constraint. (Testing if a given matrix is not copositive is an NP-complete problem. Cf. [30].)

The way to relax this hard condition is to replace the copositive cone by some tractable subcone of it. For example, replacement by the semidefinite cone would give a semidefinite program. In [75], Parrilo defines a hierarchy of tractable subcones of the copositive cone, where the first subcone is the sum of the semidefinite cone and the cone of symmetric nonnegative matrices. We will see in Section 4.2 the application of Parrilo's idea to the stability number, which is due to de Klerk and Pasechnik [50], and in Subsection 5.2.4 the application to the chromatic number, due to Dukanovic and Rendl [24].

## 1.3 Outline of the thesis and contributions

### Chapter 2: Notation and preliminaries

In Chapter 2 we recall basic linear algebra results, the general framework of conic programming, important facts about semidefinite programming, necessary definitions from graph theory, and some useful polynomial optimization tools.

The key ideas of this thesis introduced in Chapter 2 can be listed as follows:

- In Section 2.4 we focus on block diagonalization and symmetry reduction techniques. We prove Lemma 2.4.5 which enables us to block diagonalize the new hierarchy defined in Subsection 4.1.4. We recall from [85] an explicit block diagonalization of the Terwilliger algebra of the Hamming scheme.

- In Section 2.5 we give a reduction from the colouring problem to the stable set problem. It is the key observation for building the hierarchies of lower bounds for the chromatic number of a graph in Chapter 5.

\(^1\)Given a graph and weights assigned to its edges, the max cut problem is the problem of splitting the vertex set of the graph into two sets such that the size of the cut, i.e. the sum of the weights of the edges connecting vertices from different sets, is maximized.
In Section 2.6 we explain the sum of squares approach to polynomial optimization problems, that is used in Section 4.2 to construct a series of subcones of a copositive cone. We also recall the dual approach based on moment matrices, whose applications are considered in Subsections 4.1.3, 5.2.1 and 5.2.2.

Chapter 3: The Lovász theta number

Throughout the whole thesis we work with hierarchies of semidefinite bounds. All these hierarchies have either the Lovász theta number \( \vartheta(G) \) of a graph \( G \), or some variation of it, as a starting point. Chapter 3 is devoted to this number.

Chapter 3 contains a tour through several semidefinite programming formulations for \( \vartheta(G) \), the proof of ‘the sandwich theorem’, and the definitions of some variations of \( \vartheta(G) \) obtained by adding nonnegativity and triangle constraints. We make a few small contributions that are not published, and which are new to the best of our knowledge:

- a simple proof, based on Lemma 3.1.1, of the equivalence between the standard definition of \( \vartheta(G) \) and the definition related to the theta body, which is stated in Proposition 4.1.1;

- an observation that nonnegativity constraints do not improve \( \vartheta(G) \) if \( G \) contains some edge symmetry, given in Proposition 3.3.2;

- and an explanation of the phenomenon, appearing in computational results reported by Dukanovic and Rendl in [25], that triangle constraints do not improve \( \vartheta(G) \) if \( G \) is a Hamming graph, given in Proposition 3.3.3.

Chapter 4: Semidefinite programming upper bounds for the stability number

This chapter deals with the hierarchies of semidefinite upper bounds for the stability number \( \alpha(G) \) of a graph \( G \).

We first recall in Section 4.1 the definitions of the stable set polytope \( \text{STAB}(G) \) and its well known relaxations. Among others, we describe the theta body \( \text{THETA}(G) \). We then introduce and compare three hierarchies of semidefinite relaxations of \( \text{STAB}(G) \), the matrix cut hierarchy of Lovász and Schrijver [65] (Subsection 4.1.2), the moment matrix hierarchy of Lasserre [57] (Subsection 4.1.3), and the new block diagonal hierarchy nested between the previous two (Subsection 4.1.4). They all start from \( \text{THETA}(G) \) and converge to \( \text{STAB}(G) \) in finitely many steps, for any fixed graph \( G \).

The main contributions of Section 4.1 are as follows:

- Theorem 4.1.4 about the convergence in \( \alpha(G) - 1 \) steps of the hierarchy of Lovász and Schrijver [65] applied to the clique-constrained polytope;

- Subsection 4.1.4, which explains in detail the application of the hierarchy, proposed in the paper [40] by Gvozdenović, Laurent and Vallentin, to the stable set problem.
1.3. OUTLINE OF THE THESIS AND CONTRIBUTIONS

In Section 4.2 we introduce the hierarchy of de Klerk and Pasechnik [50] based on the sum of squares approach to copositive programs. We discuss some convergence properties of this hierarchy and we compare it with the hierarchy of Lasserre.

The contributions related to the hierarchy of de Klerk and Pasechnik are derived from the papers [38, 39] by Gvozdenović and Laurent (the paper [39] is the journal version of the paper [38]). The most important contributions are as follows:

- Theorem 4.2.13 that partially solves a conjecture of de Klerk and Pasechnik about the convergence of their hierarchy.
- Theorem 4.2.17 and Proposition 4.2.23 in which this hierarchy is compared with the hierarchy of Lasserre and the new block diagonal hierarchy.

Chapter 5: Semidefinite programming lower bounds for the chromatic number

Although a vast literature exists about hierarchies of relaxations for the stability number \( \alpha(G) \) of a graph \( G \), to the best of our knowledge no such hierarchy for the chromatic number \( \chi(G) \) had been studied before we started our research.

In Chapter 5 we essentially follow the work of Gvozdenović and Laurent [37]. We start with the Lovász theta number \( \vartheta(G) \) of the complement of a graph \( G \), and try to strengthen it towards the fractional chromatic number \( \chi^*(G) \) and the chromatic number \( \chi(G) \) of \( G \).

In Section 5.1 we apply the reduction from the colouring problem to the stable set problem, given in Section 2.5. We first introduce an operator \( \Psi \). It is monotone nonincreasing and maps any graph parameter nested between \( \alpha(\cdot) \) and \( \chi(\cdot) \) to a parameter lying between the clique number \( \omega(\cdot) \) and \( \chi(\cdot) \). Moreover, if a graph parameter is polynomial time computable, the same holds for its image under \( \Psi \). As a direct consequence of the properties of \( \Psi \), there is no polynomial time computable graph parameter nested between \( \chi^*(\cdot) \) and \( \chi(\cdot) \) unless \( P=NP \). We conclude the section with quadratic and copositive programming formulations for \( \chi(G) \).

In Section 5.2, we define and study hierarchies of lower bounds for \( \chi^*(G) \) and \( \chi(G) \), which are closely connected to the hierarchies presented in Chapter 4. In particular,

- we present the hierarchies based on the moment matrix approach of Lasserre [56, 57];

- we define, using the same framework, new hierarchies corresponding to the new block diagonal hierarchy.

Finally, we recall the hierarchy of Dukanovic and Rendl [24] for \( \chi^*(G) \). Their hierarchy corresponds to the hierarchy of de Klerk and Pasechnik [50]. We observe that, for vertex transitive graphs, it is dominated by the moment matrix based hierarchy.

It should be mentioned that none of the hierarchies for \( \chi^*(G) \) and \( \chi(G) \) was known before we started to work on this topic.
CHAPTER 1. INTRODUCTION

Chapter 6: Computational results

We show in Chapter 6 how to compute the semidefinite bounds studied in this thesis.

Section 6.1 contains results for Paley graphs from Gvozdenović, Laurent and Vallentin [40]. We compute the bounds on the stability number from the new block diagonal hierarchy, up to order three, for Paley graphs with at most 800 vertices. The properties of the automorphism groups of these graphs allow us to significantly reduce the number of variables and the number of blocks in the semidefinite programs that define these bounds.

In the remaining sections of Chapter 6 we follow the paper [36] by Gvozdenović and Laurent.

We consider lower bounds for the (fractional) chromatic numbers of Hamming and Kneser graphs in Sections 6.2 and 6.3, respectively. The bounds, of order one and two, from the new block diagonal hierarchies are computed for graphs with up to $2^{20}$ vertices. As the key ingredient, we use the explicit block diagonalization of the Terwilliger algebra of the Hamming scheme given by Schrijver in [85].

In Section 6.4, we introduce a new lower bound for the chromatic number of a graph. It is a variation of the second order bound in the Lasserre type hierarchy, suitable for nonsymmetric graphs. We report experimental results on some DIMACS benchmark instances. For several instances, our bounds improve the best known lower bounds.