Coping with NP-hardness: Approximation algorithms based on semidefinite programming

Thesis for degree of Ph.D.

by

Michael Langberg

Under the supervision of
Prof. Uriel Feige
Department of Computer Science and Applied Mathematics
The Weizmann Institute of Science

Submitted to the Feinberg Graduate School
Weizmann Institute of Science
Rehovot 76100, Israel

April 23, 2003

Abstract

Many important and well known optimization problems are known to be NP-hard. This implies, under the assumption that P $\neq$ NP, that solving these problems exactly in polynomial time is impossible. In many applications, it is natural to sacrifice optimality in favor of computational efficiency. Such a tradeoff is the essence of approximation algorithms.

Much effort has been invested in the study of approximation algorithms for NP-hard problems. A special family of NP-hard problems, which has been extensively studied, are combinatorial optimization problems whose decision version is NP-complete. While these problems are known to be polynomially interreducible to each other it turns out that they behave in very diverse and intriguing ways when considering approximability. Namely, for some NP-hard optimization problems it is possible to efficiently compute solutions that are guaranteed to be arbitrarily close to optimal, whereas for others computing even a crude approximation to the optimum turns out to be NP-hard.

Over the past decade, our understanding of the approximability of many combinatorial optimization problems has been greatly improved. Efficient approximation algorithms have been suggested for various optimization problems. In the design of these algorithms many algorithmic techniques have been used (such as greedy algorithms, dynamic programming, and algorithms based on linear program relaxations).

In this work, we explore the use of semidefinite programming in the approximation of NP-hard optimization problems. Given an instance of a combinatorial optimization problem, the solution of a semidefinite program can be viewed as a geometrical embedding of this instance in Euclidean space. The aim of such an embedding is to force a geometrical structure on the given instance which may ease on the process of approximation. We study both the limitations and the contribution of such embeddings in the context of approximation algorithms.

In Part I of our work we study the maximum independent set and minimum coloring problems. These problems are not only NP-hard to solve exactly, but are also hard to approximate beyond a very crude factor of $n^{-1+\varepsilon}$. The approximation ratios for these problems significantly improve in graphs that have very large independent sets, or very small chromatic numbers. The algorithms achieving the best ratios in these cases are all based on semidefinite relaxations. In our work, we study the quality of some of these relaxations. We present a family of graphs which exhibit a large gap between the value of the optimal solution to the problem under discussion and the value of the semidefinite relaxation. These gaps are tight in the sense that they match known upper bounds.

In Part II of this thesis we study the Max-Cut problem and its variant the Max-Bisection problem. The approximation algorithms achieving the best ratios for these problems are based on semidefinite programming and follow a two phase scheme. In the first phase, a semidefinite relaxation to the problem at hand is solved, yielding as a solution a semidefinite matrix or equivalently a set of vectors in R^n (here n is the size of the given graph). One may view this vector solution as an embedding of

the given instance in Euclidean space. In the second phase, this embedding is used in order to find a valid solution for the problem at hand. In other words, the vector solution is *rounded* into a valid solution of the given instance. In our work, we consider applying this two phase algorithm based on semidefinite programming and analyze the use of two rounding techniques. The first is a new rounding scheme denoted as RPR^2 which combined the random hyperplane rounding introduced by Goemans and Williamson (JACM 1995) with the concept of randomized rounding used in the rounding of linear programs. The second is a rounding technique which first uses random hyperplane rounding and then alters the combinatorial solution obtained as to achieve local optimality. Using such rounding schemes, we obtain improved approximation ratios for the Max-Bisection problem and various restrictions of the Max-Cut problem. Furthermore, we study the ratio between the value of the maximum cut of a given graph and the value of its maximum bisection, and obtain improved approximation ratios for the Max-Bisection problem restricted to regular graphs.

This thesis is based on the following papers

1. U. Feige, M. Langberg, and G. Schechtman. Graphs with tiny vector chromatic numbers and huge chromatic numbers. *In proceedings of 43rd annual Symposium on Foundations of Computer Science*, pages 283–292, 2002.

2. U. Feige and M. Langberg. The RPR^2 rounding technique for semidefinite programs. *In proceedings of ICALP*, pages 213–224, 2001.

3. U. Feige, M. Karpinski, and M. Langberg. Improved approximation of Max-Cut on graphs of bounded degree. *Journal of Algorithms*, 43(2):201–219, 2002.

4. U. Feige, M. Karpinski, and M. Langberg. A note on approximating Max-Bisection on regular graphs. *Information Processing Letters*, 79:181–188, 2001.

Declaration The author, Michael Langberg, declares that this thesis summarizes his independent work under the supervision of Professor Uriel Feige, with the exception of Chapter 2, whose results were obtained jointly with Gideon Schechtman, and Chapters 4 and 5, whose results were obtained jointly with Marek Karpinski.

Acknowledgments

It is a great pleasure to acknowledge the many people who influenced me, collaborated with me and encouraged me throughout my studies and research. First and foremost I would like to express my deepest gratitude and thanks to my advisor Uri Feige. I am grateful for his guidance, patience, and unlimited devotion. In our many discussions I was exposed to a variety of fascinating and intriguing ideas, which will surely guide me in my future research.

I was very fortunate to spend my graduate studies at the Weizmann Institute of Science. The excellent faculty, staff and students make it a wonderful and warm place to study. I am grateful for the many courses and seminars I had the privilege to attend. I would like to express special thanks to Itai Benjamini, Uri Feige, Oded Goldreich, David Harel, Moni Naor, David Peleg, Amir Pnueli, Ran Raz, Omer Reingold, Muli Safra, Gideon Schechtman, Adi Shamir, and Oded Schramm; their teaching has influenced me very much.

I would like to thank my fellow students and friends at Weizmann for making the years I have spent at the institute very enjoyable. Thanks for the many academic discussions, informal conversations, riddles, jokes, and laughs. I thank Ilan Degani, Michael Elkin, Ronny Hadany, Danny Harnik, Tzvi Hartman, Yehuda Hassin, Moysh Kaminski, Amos Korman, Robi Krauthgamer, Hillel Kugler, Kobbi Nissim, Eran Ofek, Omer Reingold, Yoav Rodeh, Alon Rosen, Sitvanit Ruah, Elad Shahar and Udi Wieder.

I am grateful to colleges with whom I had the pleasure to collaborate and share ideas. I thank Cynthia Dwork, Marek Karpinski, Monique Laurent, Moni Naor, Kobbi Nissim, Amir Pnueli, Omer Reingold, Yoav Rodeh, Gideon Schechtman, Luca Trevisan and Uri Zwick.

During my PhD studies I had the privilege to spend some time at the University of Bonn. I would like to thank Marek Karpinski for his very warm hospitality and guidance.

Finally, I would like to thank my family. My parents Pnina and Naftali, for their unconditional love and support, for always encouraging me, and for the role model that they have been throughout my life. My brothers and sister, for their love and belief in me. My in-laws Klara and Igal, for their love, support, and for helping out in numerous ways since they entered my life a decade ago. Above all, I would like to thank my much loved wife and son - Maya and Nadav, for their endless love, support and patience, which made this work possible; this thesis is dedicated to them.

To Maya and Nadav

Contents

Chapter 1

Introduction

In this work, we explore the use of semidefinite programming in the approximation of NP-hard optimization problems. Given an instance of a combinatorial optimization problem, the solution of a semidefinite program can be viewed as a geometrical embedding of this instance in Euclidean space. The aim of such an embedding is to force a geometrical structure on the given instance which may ease on the process of approximation. In this research, we study both the limitations and the contribution of such embeddings in the context of approximation algorithms.

1.1 Approximation algorithms

Many important and well known optimization problems are known to be NP-hard. This implies, under the assumption that P $\neq$ NP, that solving these problems exactly in polynomial time is impossible. In many applications, it is natural to sacrifice optimality in favor of computational efficiency. Such a tradeoff is the essence of approximation algorithms.

1.1.1 Definitions

An optimization problem P is a problem in which for each instance I of P we define a set of feasible solutions $F(I)$ and a cost function $c : F(I) \to R^+$. When solving an instance I of an optimization problem P we aim to find the optimal solution s^* that satisfies

$$c(s^*) = \max_{s \in F(I)} c(s) \quad , \quad c(s^*) = \min_{s \in F(I)} c(s)$$

when P is a maximization or minimization problem respectively.

Roughly speaking, we say that an algorithm A for an optimization problem P is an r-approximation algorithm, or that A has an approximation ratio of r, if for **every** instance I of P, algorithm A produces a solution s whose cost is within a ratio of r from the cost of the optimal solution s^*.

More rigorously, let P be an optimization problem and A be an approximation algorithm for P. There are two ways to define the ratio r of A. One may define r as the ratio between $c(s)$ and $c(s^*)$ (which implies $r \leq 1$ if P is a maximization problem and $r \geq 1$ otherwise) or as the ratio between $c(s^*)$ and $c(s)$ (which implies $r \geq 1$ if P is a maximization problem or $r \leq 1$ otherwise). In this work we define the ratio r to always be less than or equal to 1 (that is we use the first definition when P is a maximization problem and the latter otherwise). Using these definitions, the quality of the approximation algorithm A increases as r gets larger (when $r = 1$ the algorithm A is an exact algorithm).

1.1.2 Overview

Over the past decade, much effort has been invested in the study of the approximability of NP-hard problems. A special family of NP-hard problems, which has been extensively studied, are combinatorial optimization problems whose decision version is NP-complete. While these problems are known to be polynomially interreducible to each other it turns out that they behave in very diverse and intriguing ways when considering approximability. Namely, for some NP-hard optimization problems it is possible to efficiently compute solutions that are guaranteed to be arbitrarily close to optimal, whereas for others computing even a crude approximation to the optimum turns out to be NP-hard.

This field of study is two sided. On the positive side, there is the study of approximation algorithms. The goal of this line of study is to construct (efficient) approximation algorithms which achieve an approximation ratio as close to 1 as possible. On the negative side, there is the study of the inapproximability of optimization problems. In this line of research, given an optimization problem P, ones goal is to prove the *impossibility* to design (polynomial time) approximation algorithms for P beyond a specified approximation ratio r. The negative inapproximability results are obtained via various reductions (usually based on the PCP theorem [FGL$^+$96, AS98, ALM$^+$98]) and are thus valid only under certain complexity assumptions.

In the following, we exhibit the diverse behavior of NP-hard optimization problems (w.r.t. approximation) by briefly presenting the known approximability vs. inapproximability results of a few problems. The reader may find details on many more NP-hard problems (including those whose decision version is NP-complete and appear in [GJ79]) in [CK95, Hoc97, Vaz01].

Almost polynomially solvable We start by considering NP-hard problems which are *almost* known to be solved exactly in polynomial time. Namely, it turns out that there are NP-hard optimization problems for which there exists a family of polynomial r-approximation algorithms for any constant r less than one. Such a family of algorithms are denoted as *polynomial time approximation schemes* (PTAS) or *fully polynomial time approximation schemes* (FPTAS), the latter applying to families which are polynomial not only in the size of the input but also in $1/(1-r)$. A few examples to this phenomenon are the Knapsack problem [IK75] which admits a FPTAS, the Bin-Packing problem [dlVL81] which admits a PTAS and the Euclidean-TSP problem which also admits a PTAS [Aro98].

Constant upper and lower bounds How about problems which under the assumption that P $\neq$ NP, cannot admit a PTAS but still can be approximated quit well in polynomial time? Many NP-hard optimization problems fall into this category. For example all Max-SNP complete [PY91] problems share on one hand the ability to be approximated within some constant approximation ratio r, and on the other cannot be approximated beyond another constant ratio r'. Clearly, under this setting the value of r is less than or equal to r'. For some problems these two ratios are equal (up to low order terms), implying a full understanding of the approximation ratio achievable on such problems. A couple of examples include the Max-3-SAT problem which is known to have an approximation ratio of 7/8 [KZ97] but under the assumption that P $\neq$ NP cannot be approximated beyond a ratio of $7/8 + \varepsilon$ for any constant $\varepsilon > 0$ [Hås01]; and the Max-Cut problem, which can be approximated within a ratio of $\simeq 0.87856$ [GW95] but cannot be approximated (again assuming P $\neq$ NP) beyond the ratio of $16/17 \simeq 0.9411$ [Hås01]. In Part II of this work we study the Max-Cut problem and its variant the Max-Bisection problem (which both fall into this category).

Nonconstant upper and lower bounds There are also problems which do not admit a constant approximation ratio. For some of these problems, the approximation ratio achievable (essentially) matches the inapproximability results, and for others there is a large gap between the two. For example the Set-Cover problem can be approximated within a ratio of $O(\log n)^{-1}$ [Joh74] while it has been shown in [LY94, Fei98] that this is actually the best possible (assuming NP $\not\subseteq$ DTIME$(n^{O(\log\log n)})$). Another example is the maximum independent set problem. In [Fei02] it is shown that the maximum independent set problem can be approximated within a ratio of $O(\frac{\log^3 n}{n(\log\log n)^2})$, whereas it is known that unless NP=ZPP, this problem cannot be approximated within a ratio of $n^{-1+\epsilon}$ for any constant $\epsilon > 0$ [Hås99] (nonconstant values of ε are discussed in [Kho01]). An example in which there is a large gap between the known approximation ratio and inapproximability results is the Closest Lattice Vector problem (CVP). In [Sch85] it is shown that the CVP problem can be approximated within an approximation ratio of $(1+\varepsilon)^{-n}$ for any constant $\varepsilon > 0$, this ratio is far from the currently best known inapproximability result of $n^{-c/\log\log n}$ for some constant $c > 0$ [DKS98, DKRS99] (under the assumption that P $\neq$ NP).

More examples Finally there are problems which are not known to fall into the categories mentioned above. For example there are problems which on one hand do not admit a PTAS, but on the other are not known to have a constant approximation ratio. Or problems that may or may not admit a PTAS, but are currently known to have only nonconstant approximation ratios. Two examples are the problem of coloring graphs that are known to be 3 colorable and the Max-k-Dense Subgraph problem. In [BK97] it is shown that 3-colorable graphs can be colored in polynomial time by roughly $n^{3/14}$ colors, surprisingly the strongest inapproximability result states that it is NP-hard to color such graphs with 4 colors or less. This problem and closely related ones are addressed in Part I of our work. Regarding the Max-k-Dense Subgraph problem, an approximation ratio of $\max(k/(2n), n^{-1/3+\varepsilon})$ is presented in [FKP01] (for all values of k and some constant $\varepsilon > 0$), but it is not known whether or not there exists some constant $r < 1$ such that it is NP-hard to approximate the Max-k-Dense Subgraph problem beyond the ratio r. In other words it may be the case that the Max-k-Dense Subgraph problem admits a PTAS.

1.2 Semidefinite programming

One may hope that despite the diverse behavior presented in Section 1.1, there exists some generic algorithmic technique which can be used when trying to approximate NP-hard problems. This is far from being true, as it is usually the case that approximation algorithms for NP-hard optimization problems are designed specially to *fit* a specific problem. Various techniques have been used in the construction of approximation algorithms (such as greedy algorithms, dynamic programming, and algorithms based on linear program relaxations). In this work we study algorithms based on semidefinite programming.

1.2.1 Definitions

Positive semidefinite matrices An $n \times n$ symmetric matrix $X = [x_{ij}]$ is said to be *positive semidefinite* iff $y^T X y \geq 0$ for every vector $y \in R^n$. The following statements are equivalent (see for example [GL89])

(a) X is positive semidefinite.

(b) All eigenvalues of X are nonnegative.

(c) There exists a matrix B such that $X = B^T B$. Equivalently there exist vectors $v_1, \ldots, v_n \in R^m$ (here $m \leq n$) such that $x_{ij} = \langle v_i, v_j \rangle$ (the inner product of v_i and v_j).

Semidefinite programming Semidefinite programming is the problem of maximizing (or minimizing) a linear objective function subject to a set of linear constraints (as in linear programming) with the additional constraint that the variables involved in such a program form the entries of a (symmetric) positive semidefinite matrix. There are many equivalent ways to formulate semidefinite programs. One such is as follows.

(SDP1)	Maximize	$C \cdot X$	
	subject to:	$A_i \cdot X \geq b_i$	for $1 \leq i \leq k$
		X is symmetric positive semidefinite.	

Where for two $n \times n$ matrices $X = [x_{ij}]$ and $Y = [y_{ij}]$ the Frobenius inner product $X \cdot Y$ is equal to $Tr(X^T Y) = \sum_{ij} x_{ij} y_{ij}$. Using statement (c) above, *(SDP1)* has the following geometrical interpretation:

(SDP2)	Maximize	$C \cdot X$	
	subject to:	$A_i \cdot X \geq b_i$	for $1 \leq i \leq k$
		$v_i \in R^n$	for $1 \leq i \leq n$
		$X = [x_{ij}] = \langle v_i, v_j \rangle$	for $1 \leq i, j \leq n$

The solution of *(SDP2)* is an optimal set of vectors $\{v_1^*, \ldots, v_n^*\}$ in R^n that maximize the given objective function. We use this formulation of semidefinite programming throughout our work.

1.2.2 Solving semidefinite programs

Roughly speaking, semidefinite programs such as *(SDP1)* can be solved in polynomial time using the ellipsoid method ([Kha79]) or interior point methods (*e.g.* [Ali95]). To be precise, these algorithms do not produce exact solutions. Rather they produce solutions of value within an arbitrarily small additive error of ε from the value of the optimal solution. Here $\varepsilon > 0$, and the running time of these algorithms will depend on ε. Moreover, these algorithms are polynomial only for certain "well behaved" instances (*e.g.* instances in which one can bound the size of primal and dual solutions by some polynomial in the size of the input, for details see [Ali95]).

In addition, given a positive semidefinite matrix X one can polynomially obtain (using the incomplete Cholesky decomposition [GL89]) a matrix B such that $X = B^T B$. Thus the geometric interpretation *(SDP2)* of *(SDP1)* can be solved in polynomial time (again up to an arbitrarily small additive error given our program is well behaved). We now describe how this fact can be used in approximation algorithms for NP-hard problems.

1.2.3 Semidefinite programming in approximation algorithms

Semidefinite programming in the context of combinatorial optimization was first used in the work of Grötschel, Lovász, and Schrijver [Lov79, GLS88] in the solution of the maximum independent set problem on perfect graphs. Later, Goemans and Williamson [GW95] initiated the use of semidefinite programming in the *approximation* of optimization problems in their work on the Max-Cut problem. Since, semidefinite programming has found many applications as a tool in the approximation of various NP-hard optimization problems. Just to mention a few: Max-Cut and its variants Max-k-Cut, Max-Bisection *e.g.* [GW95, FJ97, Zwi99, Ye01, HZ01, FL01a, FL01b, FKL02, GW01], Max-SAT and its variant Max-k-SAT *e.g.* [GW95, FG95, KZ97, HZ99, AW02], Min-Coloring, Max-Independent Set and Vertex Cover *e.g.* [KMS98, Sze94, BK97, Fei97, GK98, AK98, Hal00, HNZ01, Cha02, FLS02]. Scheduling [Sku01], and the Betweenness problem [CS98].

The general two phase scheme As in the standard use of linear programming, approximation algorithms based on semidefinite programming follow a general two phase scheme. Let P be an optimization problem and I be an instance of P. In the first step of this two phase scheme, a semidefinite relaxation to the problem P on I is formulated and solved, yielding as a solution a positive semidefinite matrix or equivalently a set of vectors. One may view this vector solution as an embedding of the instance I in Euclidean space. In the second step this embedding is used in order to find a valid solution for the problem at hand. In other words, the vector solution is *rounded* into a valid solution for the problem P on the given instance I. In what follows, we illustrate this scheme on the Max-Cut problem (as presented in [GW95]). Given a graph $G = (V, E)$ the Max-Cut problem is the problem of finding a partition (also referred to as a *cut*) of the vertex set V into two parts U and $V \setminus U$ which maximizes the number of edges with one end point in U and the other in $V \setminus U$.

The semidefinite relaxation In the first step of the two phase plan, a semidefinite relaxation for the problem at hand (in this case the Max-Cut problem) is designed and solved. There are various techniques to construct semidefinite relaxations, one common technique (which can be applied on many problems) is based on $\{-1, 1\}$ quadratic programs (*i.e.* the problem of maximizing a quadratic objective function subject to a set of quadratic constraints and the constraint that all variables receive the value -1 or 1).

Max-Cut can be formulated as a $\{-1, 1\}$ quadratic program. Let $G = (V, E)$ be a given graph with vertex set $V = \{1, \ldots, n\}$. The $\{-1, 1\}$ quadratic program corresponding to the Max-Cut problem on G is:

(QI-MC) Maximize $\sum_{e_{ij} \in E} \left(\frac{1 - x_i \cdot x_j}{2}\right)$

subject to:

$$x_i \in \{-1, 1\} \qquad \text{for } 1 \leq i \leq n$$

The above program can be understood as follows. With each vertex $i \in V$ we associate a variable x_i, and the value of x_i (which is either $+1$ or -1) indicates in which side of the cut the respective vertex is placed. For each edge $e_{ij} \in E$, if $x_i \neq x_j$ (corresponding to the case in which e_{ij} is cut) then the value of $\frac{1 - x_i \cdot x_j}{2}$ is 1, and if $x_i = x_j$ then this value is 0.

It is NP-hard to solve $\{-1, 1\}$ quadratic programs. However, *(QI-MC)* can be relaxed to a semidefinite program. The requirement $x_i \in \{-1, 1\}$ can be relaxed by representing each variable x_i by a unit n-dimensional vector $v_i \in S^{n-1}$ (here $S^{n-1} \subset R^n$ is the unit n-dimensional sphere) and the multiplication $x_i \cdot x_j$ by the inner product $\langle v_i, v_j \rangle$.

(SDP-MC) Maximize $\sum_{e_{ij} \in E} \frac{1-\langle v_i, v_j \rangle}{2}$
subject to:
$v_i \in S^{n-1}$ for $1 \le i \le n$

As every solution of *(QI-MC)* is also a solution of *(SDP-MC)*, the value of *(SDP-MC)* is at least as large as that of *(QI-MC)*, and hence can be viewed as a relaxation to the Max-Cut problem. *(SDP-MC)* can be solved (up to arbitrary precision) in polynomial time using semidefinite programming. A solution to *(SDP-MC)* is a set of unit vectors in R^n, rather than a cut of G. To obtain a cut $(U, V \setminus U)$ of G we *round* the set of optimal vectors $v_1, \ldots, v_n$ obtained by solving *(SDP-MC)*.

Rounding semidefinite programs We now describe the rounding technique presented in [GW95] which is usually referred to as the *random hyperplane* rounding technique. Let $r = r_1, \ldots, r_n$ be a random variable with an n dimensional standard normal distribution (*i.e.* each coordinate r_i is an independent random variable with standard normal distribution). It can be seen that r is spherically symmetric, namely the direction specified by the vector $r \in R^n$ is uniformly distributed (see for instance [Ren70]). In the random hyperplane rounding technique a random vector r of the above distribution is chosen [1] and the vectors $v_1, \ldots, v_n$ are partitioned into two sets according to the sign of the inner product $\langle v_i, r \rangle$. This partition implies a cut $(U, V \setminus U)$ where U is defined to be the set $\{i \mid \langle v_i, r \rangle > 0\}$ [2].

A number of additional rounding techniques (applied in the approximation of various problems, including Max-Cut) have been designed based on random hyperplane rounding. In some of these rounding schemes the vectors $v_1, \ldots, v_n$ are first rearranged and only then partitioned by random hyperplane rounding (*e.g.* [FG95, Nes98, Zwi99, Ye01, HYZ02]). In others, the vectors $v_1, \ldots, v_n$ are first rounded using the random hyperplane rounding technique and then the combinatorial solution obtained is changed in order to improve the value of the solution or to assure its feasibility (*e.g.* [FJ97, FKL02, FL01b]). Yet in others, the vectors $v_1, \ldots, v_n$ are partitioned into two parts by a random hyperplane with *skewed* distributed [MM01, LLZ02] (namely, the distribution is not spherically symmetric and is skewed in a *special* direction). Semidefinite programming has also been used in the design of approximation algorithms for combinatorial optimization problems in which a feasible solution consists of a partition of the variables into many parts (rather than a partition into two parts as presented above). Some examples include the Min-Coloring and Max-k-Cut problems [FJ97, KMS98, GW01]. The rounding techniques used in these cases consist of choosing many independent random vectors with n-dimensional standard normal distribution, and partitioning the vertex set according to these vectors.

Putting things together Using the semidefinite program *(SDP-MC)* and random hyperplane rounding, [GW95] obtain an $\alpha \simeq 0.87856$ approximation ratio for the Max-Cut problem. Namely, Goemans and Williamson prove that given a graph $G = (V, E)$, the algorithm which first solves *(SDP-MC)* on G and afterward rounds the optimal vector solution via random hyperplane rounding, obtains

[1] As the direction of r is uniformly distributed, choosing a random r can be viewed as choosing (the normal of) a random hyperplane through the origin.

[2] Equivalently, the vectors $v_1, \ldots, v_n$ are partitioned into two sets, those that lie *above* the random hyperplane and those that do not.

a cut $(U, V \setminus U)$ of G which cuts at least αZ edges, where Z is the value of the relaxation *(SDP-MC)*. As Z is greater than the value of the optimal cut in G, an approximation ratio of α is obtained.

1.3 Overview of our results

In this work, we study the use of semidefinite programming in the approximation of NP-hard optimization problems. Our research is two folded. In Part I of our work we study certain semidefinite relaxations for the maximum independent set problem and minimum coloring problem which have been extensively studied in the past (*e.g.* [Lov79, GLS88, KMS98, Sze94, Fei97, GK98, AK98, HNZ01, Cha02]). We concentrate on the quality of these relaxations, and present a family of graphs which exhibit a large gap between the value of the optimal solution to the problem under discussion and the value of the semidefinite relaxation. These gaps are tight in the sense that they match known upper bounds presented in [KMS98].

In Part II of this thesis we study the Max-Cut problem and its variant the Max-Bisection problem. We consider applying the standard two phase algorithm based on semidefinite programming to these problems and analyze the use of two rounding techniques. In Chapter 3 we present a new rounding scheme which we call RPR^2, which combines the random hyperplane rounding of [GW95] with the concept of randomized rounding [RT87] used in the rounding of linear programs. In Chapter 4 we consider a rounding technique which first uses random hyperplane rounding and then alters the combinatorial solution obtained as to achieve local optimality. Using such rounding schemes, we obtain improved approximation ratios for the Max-Bisection problem and various restrictions of the Max-Cut problem. In Chapter 5, we continue to study the ratio between the value of the maximum cut of a given graph and the value of its maximum bisection, and obtain improved approximation ratios for the Max-Bisection problem restricted to regular graphs. In the following we give a brief overview of our results.

1.3.1 Part I

An independent set in a graph G is a set of vertices that do not induce any edges. The size of the maximum independent set in G is denoted by $\alpha(G)$. For an integer K, a K coloring of G is a function $\sigma : V \to [1, \dots, K]$ which assigns colors to the vertices of G. A valid K coloring of G is a coloring in which each color class is an independent set. The chromatic number $\chi(G)$ of G is the smallest K for which there exists a valid K coloring of G.

Finding $\alpha(G)$ and $\chi(G)$ are fundamental NP-hard problems, closely related by the inequality $\alpha(G)\chi(G) \geq n$. Given G, the question of estimating the value of $\alpha(G)$ ($\chi(G)$) or finding large independent sets (small colorings) in G have been studied extensively. Let G be a graph of size n. Then both $\chi(G)$ and $\alpha(G)$ can be approximated within a ratio of $O(\frac{\log^3 n}{n(\log\log n)^2})$ [Hal93, Fei02]. It is known that unless NP=ZPP, neither $\alpha(G)$ nor $\chi(G)$ can be approximated within a ratio of $n^{-1+\epsilon}$ for any $\epsilon > 0$ [Hås99, FK98]. Under stronger complexity assumptions, there is some $0 < \delta < 1$ such that neither problem can be approximated within a ratio of $2^{\log^\delta n}/n$ [Kho01].

The approximation ratios for these problems significantly improve in graphs that have very large independent sets, or very small chromatic numbers. The algorithms achieving the best ratios in these cases [KMS98, AK98, BK97, HNZ01] are all based on the idea of vector coloring, introduced by Karger, Motwani and Sudan [KMS98]. A vector k-coloring of a graph is an assignment of unit vectors to its vertices, such that for every edge, the inner product of the vectors assigned to its endpoints is at most (in the sense that it can only be more negative) $-1/(k-1)$. Every k-colorable graph is also vector k-colorable (by identifying each color class with one vertex of a perfect $(k-1)$-dimensional simplex centered at the origin). Moreover, unlike the chromatic number, a vector k-coloring (when it exists) can be found in polynomial time using semidefinite programming (up to arbitrarily small error in the inner products). Given a vector k-coloring of a graph, Karger, Motwani and Sudan show how to color a graph with roughly $\Delta^{1-2/k}$ colors, where Δ is the maximum degree in the graph. (In comparison, the technique of inductive coloring might use $\Delta+1$ colors.) In fact, they show how to find an independent set of size roughly $n/\Delta^{1-2/k}$. Combined with other ideas, this leads to coloring algorithms and algorithms for finding independent sets with the best currently known performance guarantees. For example, there is a polynomial time algorithm that colors 3-colorable graphs with roughly $n^{3/14}$ colors [BK97]. For nonconstant values of k, it is known how to find an independent set of size $\Omega(\log n)$ in a vector $\log n$-colorable graph.

There have also been negative results regarding vector k-colorable graphs. Examples appearing in [KMS98] (and improved by Noga Alon and Mario Szegedy) show vector 3-colorable graphs that do not have independent sets larger than roughly $n^{0.95}$. The case of nonconstant k is addressed in [Fei97] (technically, the results there deal with the Lovász theta function [Lov79], which is even a stronger notion than vector coloring), where graphs that are vector $2^{O(\sqrt{\log n})}$-colorable are shown not to have independent sets larger than $2^{O(\sqrt{\log n})}$. In all these negative examples, the vertex sets of the graphs involved can be viewed as a subset of $\{0,1\}^n$, with two vertices connected by an edge if their Hamming distance is larger than some prespecified value.

In our work we present a different family of graphs with stronger negative properties. For every constant k and every $\epsilon > 0$, we show graphs that are vector k-colorable, with $\alpha(G) \leq n/\Delta^{1-\frac{2}{k}-\epsilon}$. This essentially matches the positive results of [KMS98]. As a function of n rather than Δ, we show vector 3-colorable graphs with $\alpha(G) < n^{0.843}$. Moreover, for $k = O(\log n/\log\log n)$, we show vector k-colorable graphs with $\alpha(G) \leq (\log n)^c$, for some universal c. This shows that the vector coloring number by itself does not approximate the chromatic number with a ratio better than polylogn/n. Another consequence of this (when combined with the results of [Sze94]) is that certain semidefinite programs do not approximate the size of the maximum independent set within a ratio better than polylogn/n.

As part of our proof we analyze "property testing" [GGR98] algorithms that distinguish between graphs that have an independent set of size n/k, and graphs that are "far" from having such an independent set. Extending ideas appearing in [GGR98, AK02], we obtain bounds on the sample size which improve previous bounds of [GGR98] for this problem. These results are of interest in the context of property testing regardless of there applications to the vector coloring issue.

The results presented above are described in Chapter 2 of this thesis. A preliminary version of these results appear in [FLS02].

1.3.2 Part II

Given a graph $G = (V, E)$ the Max-Cut problem is the problem of finding a partition of V into two parts which maximizes the number of edges cut by the partition (an edge is said to be cut if its endpoints appear in different parts of the partition). In a seminal work, Goemans and Williamson [GW95] successfully use semidefinite programming in the design of an approximation algorithm for Max-Cut. Their algorithm achieves the best approximation ratio currently known for the Max-Cut problem. In Part II of this thesis we study the Max-Cut problem and its variant the Max-Bisection problem (in which the objective is that of Max-Cut under the restriction that the vertex set V is to be partitioned into two equally sized parts).

The RPR^2 rounding technique Let G be a given graph. Recall the two phase algorithm of [GW95] for Max-Cut presented in Section 1.2.3, which since has been used successfully on other combinatorial optimization problems other than Max-Cut (*e.g.* [FG95, KMS98, FJ97, KZ97, CS98, Zwi99, Sku01]). First the graph G is embedded in the n dimensional unit sphere S^{n-1} by solving a semidefinite relaxation. Afterwards, this embedding is rounded using the random hyperplane rounding technique, to obtain a solution to the problem at hand.

In our work we address the second step of the this two phase algorithm, and suggest a new rounding procedure called RPR^2 (Random Projection followed by Randomized Rounding) for rounding the solution of semidefinite programs. Our new procedure combines the original rounding procedure of [GW95] with the "randomized rounding" procedure [RT87] used in the rounding of linear programs.

Let $v_1, \ldots, v_n$ be a vector solution of a semidefinite program (*i.e.* the embedding of G in the unit sphere). In the random hyperplane rounding technique presented by [GW95], these vectors are rounded into a 0/1 solution in two steps. In the first step, the vectors $v_1, \ldots, v_n$ are projected onto a random line through the origin to obtain the vectors $\hat{v}_1, \ldots, \hat{v}_n$. For each i let x_i be the directed distance of the projection $\hat{v}_i$ from the origin. In the second step, the fractional values $x_1, \ldots, x_n$ are rounded using *threshold* rounding (that is a threshold is set, each value x_i is rounded to 1 if x_i lies above the threshold and to 0 otherwise). The threshold chosen by Goemans and Williamson is 0, rounding vectors with positive projection to 1 and negative projection to 0 [3].

In the RPR^2 procedure we introduce, the values $x_1, \ldots, x_n$ are rounded using *randomized* rounding [RT87] instead of threshold rounding. Namely, a *rounding function* $f : R \to [0, 1]$ is chosen. Each value x_i is rounded independently to 1 with probability $f(x_i)$, and to 0 otherwise.

The random hyperplane rounding technique is a special case of RPR^2. The rounding function f corresponding to random hyperplane rounding is the function which is 1 for all $x > 0$, and zero otherwise. We show that a variant of the random hyperplane rounding technique that involves *outward rotation* [Nes98, Zwi99, Ye01] is also a special case of RPR^2. We illustrate the use of RPR^2 by presenting two applications one on the Max-Bisection problem, and the other on certain restrictions of the Max-Cut problem. For the Max-Bisection problem we improve the previously best known approximation ratio of [HZ01] from 0.7016 to 0.7028. For the Max-Cut problem, we improve the tradeoff curve (presented by Zwick [Zwi99]) that relates the approximation ratio to the size of the maximum cut in a graph.

The RPR^2 rounding technique, its superiority over outward rotation, and two applications (on Max-Bisection and Max-Cut) are presented in Chapter 3. A preliminary version of these results appear in [FL01b].

Max-Cut on graphs with bounded degree Let $G = (V, E)$ be a given graph. Consider again the algorithm of Goemans and Williamson (GW) for the Max-Cut problem [GW95]. Recall that the GW algorithm is shown to have an expected approximation ratio of at least $\alpha \simeq 0.87856$. Let $(U, V \setminus U)$ be the partition of the vertex set V obtained by the GW algorithm. It is evident from the analysis in

[3] The presentation above of the random hyperplane rounding technique differs from that originally presented in [GW95], nevertheless it is not hard to verify that both presentations are equivalent.

[GW95] that it might be the case that the partition $(U, V \setminus U)$ is not *locally optimal*. Namely, it might be the case that a vertex of G and more than half its neighbors end up on the same side of the partition $(U, V \setminus U)$. We denote such vertices as *misplaced* vertices. Clearly in such a case the partition $(U, V \setminus U)$ can be improved by moving the misplaced vertex from one side of the partition to the other, thus yielding a new partition cutting more edges.

In our work, we consider enhancing the GW algorithm with an additional local step that moves misplaced vertices from one side of the partition to the other, until no such vertices are left. We prove that the enhanced algorithm yields an approximation ratio which improves upon the original approximation ratio of α by a constant fraction when the maximum degree Δ of the given graph is constant. Namely we prove an approximation ratio of 0.921 on graphs of maximum degree three, of 0.924 on regular graphs of degree three, and of $\alpha + \Omega(1/\Delta^4)$ on graphs of maximum degree Δ. To the best of our knowledge, the best known approximation ratio for the Max-Cut problem on bounded degree graphs prior our work is that of the general problem which is $\alpha \simeq 0.87856$. (In a recent paper Halperin, Livnat, and Zwick [HLZ02] improve over the approximation ratio of 0.921 that we present for graphs of degree at most three, and obtain a ratio of 0.9326. Their algorithm follows the general framework we present - the addition of a local improvement step to the GW algorithm.)

Our results are achieved using an algorithm that differs from the one presented in [GW95] not only in the additional local improvement step, but also in the fact that an *improved* semidefinite relaxation of the Max-Cut problem is used as a base of our algorithm. Specifically, we add *triangle* constraints (mentioned in [FG95]) to our relaxation. These additional constraints have been studied in the past in the context of several problems including the Max-Cut problem. As the original semidefinite relaxation used in the GW algorithm has an integrality gap of at least 1/0.885 on certain two regular graphs [GW95], we conclude that the addition of such triangle constraints to our semidefinite relaxation is crucial to the success of our algorithm.

A preliminary version of the results presented above appear in [FKL02]. These results are presented in Chapter 4 of this thesis.

Max-Bisection on regular graphs Let $G = (V, E)$ be an undirected graph. A graph G is said to be *regular* if all its vertices have equal degree. In this portion of our work, we analyze the ratio between the maximum bisection of any given regular graph G, and its maximum cut. For general graphs it is not hard to see that this ratio can be arbitrarily close to 1/2. For regular graphs we show that this ratio is of value at least $\theta \simeq 0.9027$, and that there are infinitely many regular graphs which obtain a ratio arbitrarily close to θ. We then use this property to present a 0.795 approximation algorithm for the Max-Bisection problem restricted to regular graphs. In the case of three regular graphs our results imply an approximation ratio of 0.834.

To the best of our knowledge, the best known approximation ratio, prior our work, for Max-Bisection on regular graphs is that obtained on general graphs (this ratio is 0.7028; as presented in Chapter 3). The above results are presented in Chapter 5. A preliminary version of these results appear in [FKL01].

Part I

Graph coloring

Chapter 2

The chromatic number of vector k-colorable graphs

2.1 Introduction

An independent set in a graph G is a set of vertices that induce a subgraph which does not contain any edges. The size of the maximum independent set in G is denoted by $\alpha(G)$. For an integer k, a k coloring of G is a function $\sigma : V \to [1 \dots k]$ which assigns colors to the vertices of G. A valid k coloring of G is a coloring in which each color class is an independent set. The chromatic number $\chi(G)$ of G is the smallest k for which there exists a valid k coloring of G.

Finding $\alpha(G)$ and $\chi(G)$ are fundamental NP-hard problems, closely related by the inequality $\alpha(G)\chi(G) \geq n$. Given G, the question of estimating the value of $\alpha(G)$ ($\chi(G)$) or finding large independent sets (small colorings) in G have been studied extensively. Let G be a graph of size n. Both $\chi(G)$ and $\alpha(G)$ can be approximated within a ratio of $O(\frac{n(\log\log n)^2}{\log^3 n})$ [Hal93, Fei02]. It is known that unless NP=ZPP, neither $\alpha(G)$ nor $\chi(G)$ can be approximated within a ratio of $n^{1-\epsilon}$ for any $\epsilon > 0$ [Hås99, FK98]. Under stronger complexity assumptions, there is some $0 < \delta < 1$ such that neither problem can be approximated within a ratio of $n/2^{\log^\delta n}$ [Kho01].

The approximation ratios for these problems significantly improve in graphs that have very large independent sets, or very small chromatic numbers. The algorithms achieving the best ratios in these cases [KMS98, AK98, BK97, HNZ01] are all based on the idea of vector coloring, introduced by Karger, Motwani and Sudan [KMS98]. A vector k-coloring of a graph is an assignment of unit vectors to its vertices, such that for every edge, the inner product of the vectors assigned to its endpoints is at most (in the sense that it can only be more negative) $-1/(k-1)$. Every k-colorable graph is also vector k-colorable (by identifying each color class with one vertex of a perfect $(k-1)$-dimensional simplex centered at the origin). Moreover, unlike the chromatic number, a vector k-coloring (when it exists) can be found in polynomial time using semidefinite programming (up to arbitrarily small error in the inner products). Given a vector k-coloring of a graph, Karger, Motwani and Sudan show how to color a graph with roughly $\Delta^{1-2/k}$ colors, where Δ is the maximum degree in the graph. (In comparison, the technique of inductive coloring might use $\Delta + 1$ colors.) In fact, they show how to find an independent set of size roughly $n/\Delta^{1-2/k}$. Combined with other ideas, this leads to coloring algorithms and algorithms for finding independent sets with the best currently known performance guarantees. For example, there is a polynomial time algorithm that colors 3-colorable graphs with roughly $n^{3/14}$ colors [BK97]. For nonconstant values of k, it is known how to find an independent set of size $\Omega(\log n)$ in a vector $\log n$-colorable graph.

There have also been negative results regarding vector k-colorable graphs. Examples appearing in [KMS98] (and improved by Noga Alon and Mario Szegedy) show vector 3-colorable graphs that do not have independent sets larger than roughly $n^{0.95}$. The case of nonconstant k is addressed in [Fei97] (technically, the results there deal with the Lovász theta function, which is even a stronger notion than vector coloring), where graphs that are vector $2^{O(\sqrt{\log n})}$-colorable are shown not to have independent sets larger than $2^{O(\sqrt{\log n})}$. In all these negative examples, the vertex sets of the graphs involved can be viewed as a subset of $\{0,1\}^n$, with two vertices connected by an edge if their Hamming distance is larger than some prespecified value.

Our results In this work we present a different family of graphs with stronger negative properties. For every constant k and every $\epsilon > 0$, we show graphs that are vector k-colorable, with $\alpha(G) \leq n/\Delta^{1-\frac{2}{k}-\epsilon}$. This essentially matches the positive results of [KMS98]. As a function of n rather than Δ, we show vector 3-colorable graphs with $\alpha(G) < n^{0.843}$. Moreover, for $k = O(\log n/\log\log n)$, we show vector k-colorable graphs with $\alpha(G) \leq (\log n)^c$, for some universal c. This shows that the vector coloring number by itself does not approximate the chromatic number within a ratio better than $n/\text{polylog}n$. Another consequence of this (that is touched upon in Remark 2.5.1 in Section 2.5) is that certain semidefinite programs do not approximate the size of the maximum independent set with a ratio better than $n/\text{polylog}n$.

Theorem 2.1.1

1. *For every constant $\varepsilon > 0$ and constant $k > 2$, there are infinitely many graphs G that are vector k-colorable and satisfy $\alpha(G) \leq \frac{n}{\Delta^{1-\frac{2}{k}-\varepsilon}}$, where n is the number of vertices in G, Δ is the maximum degree in G, and $\Delta > n^{\delta}$ for some constant $\delta > 0$.*

2. *For some constant c, there are infinitely many graphs G that are vector $O(\frac{\log n}{\log\log n})$-colorable and satisfy $\alpha(G) \leq (\log n)^c$.*

3. *There are infinitely many graphs G which are vector 3-colorable and satisfy $\alpha(G) \leq n^{0.843}$.*

Proof techniques The graphs that we use are essentially the same graphs that were used in [FS02] to show integrality gaps for semidefinite programs for Max-Cut. Namely, they are obtained by placing n points at random on a d-dimensional unit sphere, and connecting two points by an edge if the inner product of their respective vectors is below $-1/(k-1)$. Such graphs are necessarily vector k-colorable, as the embedding on the sphere is a vector k-coloring. So the bulk of the work is in proving that they have no large independent set. For this we use a two phase plan. First we consider a continuous (infinite) graph, where every point on the sphere is a vertex, and two vertices are connected by an edge if the inner product of their respective vectors is below $-1/(k-1)$. On this continuous graph we use certain symmetrization techniques in order to analyze its properties. Specifically, we prove certain inequalities regarding its expansion. In the second phase, we consider a (finite) random vertex induced subgraph of the continuous graph. Based on the expansion properties of the continuous graph, we show that the random sample has no large independent set.

Two remarks are in order here. One is that it is very important for our bounds that the final random graph does not contain too many vertices compared to the dimension d. A small number of vertices implies low degree, and allows for a more favorable relation between the maximum degree and the chromatic number. For this reason we cannot use the continuous graph (or a finite discrete approximation of the continuous graph) as is. Its degree is very large compared to its chromatic number. We thus consider the graph obtained by randomly sampling the vertices of the continuous graph. In Section 2.6, we follow a suggestion of Luca Trevisan (private communication) and present an alternative proof of Theorem 2.1.1 (1) by considering the graph obtained by randomly sampling the *edges* of the continuous graph. It appears that parts (2) and (3) of Theorem 2.1.1 can not be proven using edge sampling.

The other remark is that we do not get an explicit graph as our example, but rather a random graph (or a distribution on graphs). This is to some extent unavoidable, given that there are no known efficient deterministic constructions of Ramsey graphs (graphs in which the size of the maximum independent set and maximum clique are both bounded by some polylog in n). The graphs we construct (when $k = \log n / \log \log n$) are Ramsey graphs, because it can be shown that the maximum clique size is never larger than the vector coloring number.

Property testing The following problem in property testing is addressed by Goldreich, Goldwasser and Ron [GGR98]. For some value of $\rho < 1$, consider a graph with the following "promise": either it has an independent set of size ρn, or it is far from any such graph, in the sense that any vertex induced subgraph of ρn vertices induces at least ϵn^2 edges. One wants an algorithm that samples as few vertices as possible, looks at the subgraph induced on them, and based on the size of the maximum independent set in it decides correctly (with high probability) which of the two cases above hold. In [GGR98] it is shown that a sample of size proportional to ϵ^{-4} suffices. We are in a somewhat similar situation when we move from the continuous graph to our random sample. The continuous graph is far from having an independent set of *measure* ρ [1]. We want to take a small as possible sample (its size will be denoted by s) such that the induced subgraph does not contain an independent set of size ρs. Luckily in our case, we can use a stronger guarantee on the continuous graph. We know that even every set of measure $\rho/2$ induces at least the measure of ε edges [2]. In this case we show that a sample size s proportional to $1/\epsilon$ suffices. This dramatic improvement over the [GGR98] bound is crucial to the success of our analysis. We note that this improvement is not only based on our stronger guarantee on the continuous graph G. Even in exactly the same setting of [GGR98], we show that a sample size s proportional to ϵ^{-3} suffices. These results are of interest in the context of property testing regardless of their applications to the vector coloring issue.

Strict vector coloring One may strengthen the notion of vector k-coloring by requiring that for every edge, the inner product of the unit vectors corresponding to its endpoints be exactly $-1/(k-1)$, rather than at most $-1/(k-1)$. This is called a strict vector coloring. This notion is known to be equivalent to the theta function of Lovász [Lov79, KMS98]. Every k-colorable graph is also strictly vector k-colorable. As strict vector coloring is a stronger requirement than vector coloring, then potentially, strict vector k-colorable graphs have smaller chromatic numbers than vector k-colorable

[1] As the continuous graph in infinite, terms such as the *number* of vertices is replaced by the continuous analog measure. The measure we use on the unit sphere is the natural measure which associates with each subset its relative area on the sphere.

[2] The measure we use on the edge set is the product measure induced by the standard uniform measure on S^{d-1}.

graphs. So far, there has not been any algorithmic technique that could use this observation to further improve the approximation ratios for chromatic number or for independent set. We remark however that the negative results in the current chapter apply only to vector coloring and not to strict vector coloring. It is an open question whether similar negative results are true for strict vector coloring, or equivalently, whether the same gaps (such as $n/\text{polylog} n$) can be shown between the value of the theta function and the size of the maximum independent set. Note that the weaker negative results of [Fei97] and some of the negative results in [KMS98] do apply also to strict vector coloring.

The remainder of the chapter is organized as follows. In Section 2.2 (and Section 2.7) we present our results regarding property testing. In Section 2.3 we present the continuous graph and analyze its properties. In Section 2.4 we prove Theorem 2.1.1. The semidefinite programs that compute the vector chromatic number and its variants are briefly reviewed in Section 2.5. In Section 2.6 we present an alternative proof of Theorem 2.1.1 (1). In Section 2.8 we discuss some problems that remain open. Finally, in Section 2.9 we present the proof of several technical Lemmas that appear throughout this chapter.

2.2 Random sampling

Let G be a graph of size n which does not have an independent set of size ρn (*i.e.* $\alpha(G) < \rho n$). Let H be a random subgraph of G of size s (*i.e.* H is the subgraph induced by a random subset of vertices in G of size s). In this section we study the minimal value of s for which $\alpha(H) \leq \rho s$ with high probability.

In general, if our only assumption on G is that $\alpha(G) < \rho n$, we cannot hope to set s to be smaller than n. Hence, we strengthen our assumption on G, to graphs G which not only satisfy $\alpha(G) \leq \rho n$ but are also *far* from having an independent set of size ρn (we defer defining the exact notion of "far" until later in this discussion). That is, given a graph G which is *far* from having an independent set of size ρn, we ask for the minimal value of s for which (with high probability) a random subgraph of size s does not have an independent set of size ρs. This question (and many other closely related ones) have been studied in [GGR98] under the title of *property testing*.

In [GGR98], a graph G of size n is said to be ε-far from having an independent set of size ρn if any set of size ρn in G has at least εn^2 induced edges. It was shown in [GGR98] that if G is ε-far from having an independent set of size ρn then with high probability a random subgraph of size $s = \frac{c \log(1/\varepsilon)\rho}{\varepsilon^4}$, for a sufficiently large constant c, does not have an independent set of size ρs.

The results of [GGR98] do not suffice for the proof (as we present it) of Theorem 2.1.1. We thus turn to strengthening their results. To do so, we introduce a stronger notion of being "ε-far". Roughly speaking, we prove that under our new notion of distance, choosing s to be of size $\frac{\rho}{\varepsilon}$ suffices. Furthermore, by applying our proof techniques on the original notion of distance presented in [GGR98], we improve the result of [GGR98] stated above and obtain a sample size proportional to $1/\varepsilon^3$. In Section 2.7 we continue to study the original notion of ε-far of [GGR98] and present a lower bound on the sample size which is proportional to $1/\varepsilon^2$. The proof techniques used in this section are based on the techniques appearing in [GGR98] and [AK02] (in the latter, property testing of the chromatic number is considered). We start with the following definitions

Definition 2.2.1 *Let A and B be (not necessarily disjoint) subsets of G. For each vertex $v \in A$ let $d_v(B)$ be the number of neighbors v has in B. Let $E(A, B) = \sum_{v \in A} d_v(B)$.*

Definition 2.2.2 *Let $\rho < 1$. A graph $G = (V, E)$ is said to be $\langle \rho, \varepsilon \rangle$-connected iff every subset A of V of size ρn satisfies $E(A, A) \geq \varepsilon n^2$ (*i.e.* the number of edges in the subgraph induced by A is greater than $\frac{\varepsilon}{2} n^2$).*

Notice that $\varepsilon \leq \rho^2$. Furthermore notice, that a graph G is $\langle \rho, \varepsilon \rangle$-connected iff G is $\varepsilon/2$-far (by the definitions presented in [GGR98]) from having an independent set of size ρn.

Definition 2.2.3 *Let $\rho < 1$. A graph $G = (V, E)$ is said to be* pairwise *$\langle \rho, \varepsilon \rangle$-connected iff every two subsets A and B of V of size ρn satisfy $E(A, B) \geq \varepsilon n^2$.*

As mentioned above for $\langle \rho, \varepsilon \rangle$-connected and pairwise $\langle \rho, \varepsilon \rangle$-connected graphs G, we study the minimal value of s for which $\alpha(H) \leq \rho s$ with high probability. We do so by analyzing the probability that a random subset H of G satisfies $\alpha(H) \leq \rho s$ (as a function of ρ, ε, and the sample size s). The main idea behind our proof is as follows. Given a sample size s, we start by bounding the probability that a random subset R of G of size $k > \rho s$ is an independent set. Then, using the standard union bound on all subsets R of H of size greater than ρs, we bound the probability that $\alpha(H) > \rho s$. In Section 2.2.1 we analyze this strategy and show that it suffices to bound a condition slightly weaker than the condition $\alpha(H) > \rho s$. Namely, using this scheme, we are able to bound the probability for which $\alpha(H) > \delta \rho s$ for sufficiently large constants δ. This result is later used in Section 2.4 to prove Theorem 2.1.1. In Section 2.2.2 we refine our scheme and obtain the main result of this section.

Theorem 2.2.4 *Let G be a $\langle \rho, \varepsilon \rangle$-connected graph. Let H be a random sample of G of size s. For any constant $c_1 > 0$ there exists a constant $c_2 > 0$ (depending on c_1 alone) s.t.*

1. *If $s \geq c_2 \frac{\rho^4}{\varepsilon^3} \log\left(\frac{\rho}{\varepsilon}\right)$ then the probability that H has an independent set of size $> \rho s$ is at most $e^{-c_1 \frac{\rho}{\varepsilon}}$.*
2. *If G is pairwise $\langle \rho, \varepsilon \rangle$-connected, and $s \geq c_2 \frac{\rho^5}{\varepsilon^3} \log\left(\frac{\rho}{\varepsilon}\right) \log(\frac{1}{\rho})$ then the probability that H has an independent set of size $> \rho s$ is at most $e^{-c_1 \frac{\rho^2 \log(1/\rho)}{\varepsilon}}$.*

Throughout this section we analyze the properties of random subsets H which are assumed to be *small*. Namely, we assume that the value of s (as stated in the Theorem 2.2.4) and the parameters ρ, ε and n satisfy (a) $s < c\sqrt{n}$ and (b) $s < c\rho n$ for a sufficiently small constant c. In our applications (and also in standard ones) these assumptions hold.

2.2.1 The Naive scheme

Let $G = (V, E)$ be a $\langle \rho, \varepsilon \rangle$-connected graph (pairwise or not). In this section we study the probability that a random subset R of V of size k is an independent set. We then use this result to bound the probability that a random subset H of G of size s has a large independent set.

We would like to bound (from above) the probability that R induces an independent set. Let $\{r_1, \ldots, r_k\}$ be the vertices of R. Consider choosing the vertices of R one by one, such that at each step the random subset chosen so far is $R_i = \{r_1, \ldots, r_i\}$. Assume that at some stage R_i is an independent set. We would like to show (with high probability) that after adding the remaining vertices of $\{r_{i+1}, \ldots, r_k\}$ to R_i, the final set R will not be an independent set.

Let $I(R_i)$ (for *independent*) be the set of vertices in V which are not adjacent to any vertices in R_i, and let $N(R_i)$ be the set of vertices that are adjacent to a vertex in R_i. Consider the next random vertex $r_{i+1} \in R$. If r_{i+1} is chosen from $N(R_i)$ then R_{i+1} is no longer an independent set (implying that R isn't either), and we view this round as a success. Otherwise, r_{i+1} happens to be in $I(R_i)$ and R_{i+1} is still an independent set. But if r_{i+1} also happens to have many neighbors in $I(R_i)$ then adding it to R_i will substantially reduce the size of $I(R_{i+1})$ which works in our favor. This later case is also viewed as a successful round regarding R_i.

Motivated by the discussion above, we continue with the following definitions. As before, let $G = (V, E)$ be a $\langle \rho, \varepsilon \rangle$-connected graph (pairwise or not), let $R = \{r_1, \ldots, r_k\}$ be a set of vertices in V, and let $R_i = \{r_1, \ldots, r_i\}$. Each subset R_i of V defines the following partition (LI_i, HI_i, N_i) of V:

- Let I_i be the vertices that are not adjacent to any vertex in R_i. I_i is now partitioned into two parts: vertices in I_i which have *low* degree, denoted as the set LI_i, and vertices of *high* degree, denoted as HI_i. Namely LI_i is defined to be the ρn vertices of I_i with minimal degree (in the subgraph induced by I_i), and HI_i is defined to be the remaining vertices of I_i. Ties are broken arbitrarily or in *favor* of vertices in R_i (namely, vertices in R_i are placed in I_i before other vertices of identical degree). If it is the case that $|I_i| \leq \rho n$ then LI_i is defined to be I_i, and HI_i is defined to be empty.

- N_i is defined to be the remaining vertices of V (namely, the vertices that share an edge with some vertex in R_i).

We define the partition corresponding to $R_0 = \phi$ as (LI_0, HI_0, NI_0), where LI_0 are the ρn vertices of G of minimal degree, HI_0 are the remaining vertices of G, and $N_0 = \phi$.

Notice, using this notation, that the subset R_i is an independent set iff $R_i \cap N_i = \phi$, or equivalently $R_i \subseteq I_i$. Moreover, in this case $R_i \subseteq LI_i$ (all vertices of R_i have degree 0 in the subgraph induced by I_i). Furthermore, each vertex r_i in an independent set $R = R_k = \{r_1, \ldots, r_k\}$ satisfies $r_i \in I_{i-1}$.

We are now ready to bound the probability that a random subset $R = \{r_1, \ldots, r_k\}$ of G is independent. Let $R_i = \{r_1, \ldots, r_i\}$, and let (LI_i, HI_i, N_i) be the corresponding partitions of V defined by R_i. Consider the case in which R is an independent set. As mentioned above, this happens iff for every i the vertex r_i is chosen to be independent from the subset R_{i-1}, or in other words $r_i \in I_{i-1} = LI_{i-1} \cup HI_{i-1}$. We would like to show that this happens with small probability (if k is large enough).

Initially, the subset I_0 is large (the entire vertex set V) and it gets smaller and smaller as we proceed in the choice of vertices in R. Each vertex in $r_i \in HI_{i-1}$ reduces the size of I_{i-1} substantially, while each vertex in LI_{i-1} may only slightly change the size of I_{i-1}. In the following, we show that there cannot be many vertices $r_i \in R$ that happen to fall into HI_{i-1} (as each such vertex reduces the size of I_{i-1} substantially). We thus turn to consider vertices r_i that fall in LI_{i-1} (there are almost k such vertices). The size of LI_i is bounded by ρn. Hence, the probability that $r_i \in LI_i$ is bounded by ρ (by our definitions $R_{i-1} \subseteq LI_{i-1}$ and the vertex r_i is random in $V \setminus R_{i-1}$). This implies that the probability that R is an independent set is roughly bounded by ρ^k. Details follow.

Lemma 2.2.5 *Let G be a $\langle \rho, \varepsilon \rangle$-connected graph. Let $R = \{r_1, \ldots, r_k\}$ be a set in G. The number of vertices r_i which satisfy $r_i \in HI_{i-1}$ is bounded by $t = \frac{\rho}{\varepsilon}$. If G is pairwise $\langle \rho, \varepsilon \rangle$-connected, then the number of vertices r_i which satisfy $r_i \in HI_{i-1}$ is bounded by $t = \frac{2\rho^2 \lceil \log(1/\rho) \rceil}{\varepsilon}$.*

Proof: We start with the following claim

Claim 2.2.6 *Let R_i be as defined above, and let (LI_i, HI_i, N_i) be its corresponding partition. Let $I_i = LI_i \cup HI_i$. If G is $\langle\rho, \varepsilon\rangle$-connected then, every vertex in HI_i has degree at least $\frac{\varepsilon}{\rho}n$ (in the subgraph induced by I_i). If G is pairwise $\langle\rho, \varepsilon\rangle$-connected then, every vertex in HI_i has degree at least $\frac{\varepsilon}{2\rho^2}|I_i|$ (in the subgraph induced by I_i).*

Proof: Assume that $|I_i| = \alpha\rho n$ for some $\alpha \geq 1$ (otherwise the set HI_i is empty, and the claim holds). Notice that this implies $|LI_i| = \rho n$. For the first part of our claim, recall by the definition of $\langle\rho, \varepsilon\rangle$-connected graphs that $E(LI_i, LI_i) \geq \varepsilon n^2$. Hence, we conclude that there exists a vertex in LI_i of degree at least $\frac{\varepsilon}{\rho}n$ (in the subgraph induced by LI_i). The set $LI_i \subseteq I_i$, so also in the subgraph induced by I_i, there exists a vertex in LI_i of degree at least $\frac{\varepsilon}{\rho}n$. As LI_i are the vertices of minimal degree in I_i, we conclude the first part of our assertion.

For the second part, let $I_i = X_1 \cup X_2 \cup, \ldots, \cup X_\ell$, where $\{X_1, \ldots, X_\ell\}$ is a partition of I_i into ℓ sets in which the size of X_j for all $j \neq \ell$ is ρn. Notice that $\ell = \lfloor\alpha\rfloor + 1$. For each $v \in I_i$ let $d_v(I_i)$ be the degree of v in the subgraph induced by I_i.

In this case our graph G is pairwise $\langle\rho, \varepsilon\rangle$-connected. This implies that the value of $E(LI_i, X_j)$ for each j (except $j = \ell$) is at least εn^2. Hence, $\sum_{v \in LI_i} d_v(I_i)$ is at least $\lfloor\alpha\rfloor\varepsilon n^2 \geq \frac{\alpha}{2}\varepsilon n^2$. This implies that LI_i must include a vertex v with degree $d_v(I_i) \geq \alpha\frac{\varepsilon}{2\rho}n$. As LI_i are the vertices of minimal degree in I_i, we conclude our assertion. $\square$

Now to prove our lemma, consider the subsets $R_i = \{r_1, \ldots, r_i\}$ and their corresponding partitions (LI_i, HI_i, N_i). Let $I_i = LI_i \cup HI_i$. Let $N(r_i)$ be the vertices adjacent to r_i in I_{i-1}. We would like to bound the number of vertices r_i that are in HI_{i-1}. We start with the case in which G is $\langle\rho, \varepsilon\rangle$-connected. Consider a vertex r_i in HI_i. By Claim 2.2.6, its degree in I_{i-1} is $|N(r_i)| \geq \frac{\varepsilon}{\rho}n$. Each vertex $r_i \in HI_{i-1}$ increases the size of N_{i-1} by at least $|N(r_i)|$. Initially, N_0 is empty, and after r_k is chosen $|N_k| \leq n$. We conclude that there are at most $\frac{\rho}{\varepsilon}$ vertices r_i in R which are in HI_{i-1}. This bound can be further improved be a factor of approximately ρ to $\frac{2\rho^2\lceil\log(1/\rho)\rceil}{\varepsilon}$ using tighter analysis when G is also pairwise $\langle\rho, \varepsilon\rangle$-connected, details follow.

Let $x \geq 0$ be an integer, and let $S_x = \{i \mid r_i \in HI_{i-1} \text{ and } |I_{i-1}| \in [\frac{n}{2^{x+1}}, \frac{n}{2^x})\}$. We would like to bound the size of S_x for all possible values of x. We start by considering values of x between 0 and $\lceil\log(1/\rho)\rceil - 1$. Consider a vertex r_i in which $i \in S_x$. That is, $r_i \in HI_{i-1}$ and $\frac{n}{2^x} \geq |I_{i-1}| > \frac{n}{2^{x+1}}$. By Claim 2.2.6, the degree of r_i in I_{i-1} is $|N(r_i)| \geq \frac{\varepsilon}{2\rho^2}|I_{i-1}| \geq \frac{\varepsilon}{2\rho^2}\frac{n}{2^{x+1}}$. Each vertex r_i in which $i \in S_x$ increases the size of N_{i-1} by at least $|N(r_i)|$. For such vertices, N_{i-1} is of size at least $n - \frac{n}{2^x}$ and at most of size $n - \frac{n}{2^{x+1}}$. We conclude that $|S_x|$ is of size at most $\frac{2\rho^2}{\varepsilon}$.

For $x \geq \lceil\log(1/\rho)\rceil$, the set S_x is a subset of $\{i \mid r_i \in HI_{i-1} \text{ and } |I_{i-1}| \leq \rho n\}$. Recall that $HI_{i-1} = \phi$ whenever $|I_{i-1}| \leq \rho n$. This implies that $S_x = \phi$ in these cases. All in all, we conclude that $\sum_x |S_x| \leq \frac{2\rho^2\lceil\log(1/\rho)\rceil}{\varepsilon}$, which concludes our proof. $\square$

Theorem 2.2.7 *Let G be a $\langle\rho, \varepsilon\rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.5. Let $k \geq 2t$. The probability that k random vertices of G induce an independent set is at most*

$$\rho^k \left(\frac{ek}{t\rho}\right)^t$$

Proof: Let $R = \{r_1, \ldots, r_k\}$ be a set of k random vertices. As mentioned previously, the probability that $r_i \in LI_{i-1}$ is at most ρ. This follows from the fact that (1) The size of LI_{i-1} is at most ρn, (2) $R_{i-1} \subseteq LI_{i-1}$ (by our definitions), and (3) The vertex r_i is random in $V \setminus R_{i-1}$.

Now in order for R to be an independent set, every vertex r_i of R must be in the set I_{i-1}. Furthermore, by Lemma 2.2.5 all but t vertices r_i of R must satisfy $r_i \in LI_{i-1}$. Hence, the probability that R is an independent set is at most

$$\binom{k}{t}\rho^{k-t} \leq \left(\frac{ke}{t}\right)^t \rho^{k-t} = \rho^k \left(\frac{ek}{t\rho}\right)^t.$$

$\square$

Let δ be a large constant. We now use Theorem 2.2.7 to bound the probability that a random subset H of G of size s has an independent set of size $> \delta\rho s$. The result is the following Corollary 2.2.8, which will be used in Section 2.4 to prove Theorem 2.1.1 (the main theorem of this work). In Section 2.2.2 we refine our proof techniques and *get rid* of the parameter δ. That is, we bound the probability that a random subset H of G of size s has an independent set of size $> \rho s$.

Corollary 2.2.8 *Let G be a $\langle\rho, \varepsilon\rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.5. Let H be a random sample of G of size s. Let $\delta \geq e^2$, and let c be a sufficiently large constant. If $s \geq ct\frac{\log(1/\rho)}{\rho}$ then the probability that $\alpha(H) > \delta\rho s$ is at most $\left(\frac{e}{\delta}\right)^{\Omega(\delta\rho s)}$.*

Proof: Let $k = \delta\rho s$. Using Theorem 2.2.7 and the fact that a subset R of H is random in G, the probability that there is an independent set R in H of size k is at most

$$\binom{s}{k}\rho^k \left(\frac{ek}{t\rho}\right)^t \leq \left[\left(\frac{e}{\delta}\right)^{\frac{k}{t}} \frac{ek}{t\rho}\right]^t \leq \left(\frac{e}{\delta}\right)^{\Omega(k)}.$$

In the last inequality we use the fact that $\frac{k}{t}$ is greater than $c\log(1/\rho)$, and $\frac{e}{\delta}$ is smaller than $1/e$. $\square$

2.2.2 An enhanced analysis

Let $G = (V, E)$ be a $\langle\rho, \varepsilon\rangle$-connected graph (pairwise or not), and let $H = \{h_1, \ldots, h_s\}$ be a set of random vertices of size s in V. In the previous section we presented a bound on the probability that $\alpha(H) > \delta\rho s$ for large constant values of δ. In this section we enhance our analysis, and bound the probability that $\alpha(H) > \rho s$ (namely, we get rid of the additional parameter δ).

Recall our proof technique from Section 2.2.1. We started by analyzing the probability that a subset R of H of size k is an independent set. Afterwards we bounded the probability that $\alpha(H) > \delta\rho n$ by using the standard union bound on all subsets R of H of size greater than $k = \delta\rho n$. In this section we enhance the first part of this scheme by analyzing the probability that a subset R of H of size k is a *maximum* independent set in H (rather than just an independent set of H). Then, as before, using the standard union bound on all large subsets R of H, we bound the probability that $\alpha(H) > \rho s$. We show that taking the maximality property of R into account will suffice to prove Theorem 2.2.4. Details follow.

Let $H = \{h_1, \ldots, h_s\}$ be s random vertices in G. We would like to analyze the probability that a given subset R of H of size k is a *maximum* independent set. Recall (Section 2.2.1), that the probability that R is an independent set is bounded by approximately ρ^k. An independent set R is a maximum independent set in H only if adding any other vertex in H to R will yield a set which is no longer independent. Let $R = R_k$ be an independent set, and let (LI_k, HI_k, N_k) be the partition (as defined in Section 2.2.1) corresponding to R. Consider an additional random vertex h from H. The probability that $R \cup h$ is no longer an independent set is approximately $|N_k|/n$ (here we assume that $|R|$ is small compared to n). The probability that for every $h \in H \setminus R$ the subset $R \cup \{h\}$ is no longer independent is thus $\simeq (|N_k|/n)^{s-k}$. Hence, the probability that a given subset R of H of size k is a *maximum* independent set is bounded by approximately $\rho^k\,(|N_k|/n)^{s-k}$. This value is substantially smaller than ρ^k iff $|N_k|$ is substantially smaller than n. We conclude that it is in our favor to somehow ensure that $|N_k|$ is not too large. We do this in an artificial manner.

Let $R = \{r_1, \ldots, r_k\}$ be an independent set, let $R_i = \{r_1, \ldots, r_i\}$, and let (LI_i, HI_i, N_i) be the partition (as defined in Section 2.2.1) corresponding to R_i. Roughly speaking, in Section 2.2.1, every time a vertex r_i was chosen, the subset N_i was updated. If r_i was chosen in HI_{i-1}, then N_{i-1} grew substantially, and if r_i was chosen in LI_{i-1}, the subset N_{i-1} was only slightly changed. We would like to change the definition of the partition (LI_i, HI_i, N_i) corresponding to R_i as to ensure that N_i is always substantially smaller than n. This cannot be done unless we relax the definition of N_i. In our new definition N_i will no longer represent the entire set of vertices adjacent to R_i, rather N_i will only include a subset of vertices adjacent to R_i (a subset which is substantially smaller than n). Namely, in our new definition of the partition (LI_i, HI_i, N_i) the set N_{i-1} is changed only if r_i was chosen in HI_{i-1}. In the case in which $r_i \in LI_{i-1} \cup N_{i-1}$, we do not change N_{i-1} at all. As we will see, such a definition will imply that $|N_i| \leq (1-\rho)s$ (the missing details will be given shortly) which will now suffice for our proof.

A new partition Let $H = \{h_1, \ldots, h_s\}$ be a subset of V. Let $R_i = \{r_1, \ldots, r_i\}$ be a subset of H of size i. Each such subset R_i defines a partition (LI_i, HI_i, N_i) of V. As before, let $I_i = LI_i \cup HI_i$.

1. Initially $R_0 = \phi$, LI_0 is the ρn vertices in V of minimal degree (in V), $HI_0 = V \setminus LI_0$, and $N_0 = \phi$. In the above, ties are broken by an assumed ordering on the vertices in V (in this case we do not *favor* vertices in R_i).

2. Let (LI_i, HI_i, N_i) be the partition corresponding to R_i, let r_{i+1} be a new random vertex. Let $R_{i+1} = R_i \cup \{r_{i+1}\}$, we define the partition $(LI_{i+1}, HI_{i+1}, N_{i+1})$. Let $N(r_{i+1})$ be the neighbors of r_{i+1} in I_i. We consider the following cases:

   - If $r_{i+1} \in LI_i$ then the partition corresponding to R_{i+1} will be exactly the partition corresponding to R_i namely $LI_{i+1} = LI_i$, $HI_{i+1} = HI_i$, and $N_{i+1} = N_i$. Notice that this implies that N_{i+1} no longer represents all neighbors of R_{i+1}. There may be vertices adjacent to R_{i+1} which are in I_{i+1}.
   - If $r_{i+1} \in HI_i$ then we consider two sub-cases:
     - If $|N_i \cup N(r_{i+1})| \leq (1-\rho)n$, then LI_{i+1}, HI_{i+1}, and N_{i+1} are defined as in Section2.2.1. Namely, $N_{i+1} = N_i \cup N(r_{i+1})$. I_{i+1} is defined to be $V \setminus N_{i+1}$. LI_{i+1} is defined to be the ρn vertices of I_{i+1} with minimal degree (in the subgraph induced by I_{i+1}) and HI_{i+1} is defined to be the remaining vertices of I_{i+1}. Ties are broken by the assumed ordering on V.

– If $|N_i \cup N(r_{i+1})| > (1-\rho)n$, then let $\hat{N}(r_{i+1})$ be the first (according to the assumed ordering on V) $(1-\rho)n - |N_i|$ vertices in $N(r_{i+1})$ and set $N_{i+1} = N_i \cup \hat{N}(r_{i+1})$. Furthermore, set LI_{i+1} to be the remaining ρn vertices of G, and HI_i to be empty. Notice that in this case $|N_{i+1}|$ is of size exactly $(1-\rho)n$.

- If $r_{i+1} \in N_i$ then, once again, the partition corresponding to R_{i+1} will be exactly the partition corresponding to R_i.

A few remarks are in place. First of all it is not hard to verify that the definition above implies

Claim 2.2.9 *Let $i \in \{1, \ldots, k\}$. The partitions (LI_i, HI_i, N_i) corresponding to R_i as defined above satisfy (a) $I_i \subseteq I_{i-1}$. (b) $N_{i-1} \subseteq N_i$. (c) $|N_i| \leq (1-\rho)n$. (d) $|LI_i| = \rho n$. (e) The set LI_i is the ρn vertices of minimal degree in I_i.*

Secondly, due to the iterative definition of our new partition, the partitions (LI_i, HI_i, N_i) corresponding to the subsets R_i depend strongly on the specific ordering of the vertices in R_i. Namely, in contrast to the partitions used in Section 2.2.1, a single subset R with two different orderings may yield two different partitions. For this reason, in the remainder of this section, we will assume that the vertices of H are chosen one by one. This will imply an ordering on H and on any subset R of H. The partitions we will study will correspond to these orderings only.

Finally, in Section 2.2.1, an (ordered) subset $R = \{r_1, \ldots, r_k\}$ was independent iff $\forall i \;\; r_i \in I_{i-1}$ (according to the definition of I_{i-1} appearing in Section 2.2.1). In this section, if R is independent then it still holds that $\forall i \;\; r_i \in I_{i-1}$. However, it may be the case that $\forall i \;\; r_i \in I_{i-1}$ but R is not an independent set. In the remainder of this section, we call ordered subsets R for which $\forall i \;\; r_i \in I_{i-1}$ *free* sets. We analyze the probability that a random ordered subset H of V of size s does not have any free sets of size larger then ρs. This implies, that H does not include any independent sets of size ρs. Details follow.

Definition 2.2.10 *An ordered subset $R_i = \{r_1, \ldots, r_i\}$ is said to be $free$ if it is the case that $r_j \in I_{j-1}$ for all $j \leq i$.*

Claim 2.2.11 *Let $H = \{h_1, \ldots, h_s\}$ be an ordered set of vertices in a (pairwise) $\langle \rho, \varepsilon \rangle$-connected graph G. If $\alpha(H) > \rho s$ then the maximum free set in H (w.r.t. the ordering implied by H) is of size $> \rho s$.*

Proof: Let I be an independent set of size $> \rho s$ in H. It is not hard to verify that I (under the ordering implied by H) is a free set. Let I' be a maximal free set including I, where I' is ordered by the ordering implied by H. I' is of size $> \rho s$. We conclude that the maximum free set R in H (ordered by the ordering implied by H) is of size $\geq |I'| > \rho s$. □

Claim 2.2.11 implies that to prove Theorem 2.2.4 it suffices to analyze the maximum free set $R \subseteq H$. Moreover, the only ordered subsets R that we need to consider are those ordered by the ordering implied by H. We now turn to prove Theorem 2.2.4. Roughly speaking, we start by analyzing the probability that a random subset R is a free set. We then analyze the probability that a given subset R in H is a maximum free set. Finally, we use the union bound on all subsets R of H of size $> \rho s$ to obtain our results.

In the remainder of this section, we will assume that the subset H is chosen from G randomly *with repetitions*. That is H is a random multi-set of size s. Our results (with minor modifications) apply also to the case in which H is a random subset of G (and not a multi-set) if the size of H is not very large (here we assume that $|H| << \sqrt{n}$). As in such cases, a set H of size s which is randomly chosen from V with repetitions will not include the same vertex twice (with high probability).

We start by stating the following lemmas which are analogous to Lemma 2.2.5 and Theorem 2.2.7 from Section 2.2.1. The main difference between the lemmas below (and their proof), and those of the previous section is in the definition of the partition (LI_i, HI_i, N_i) and in the fact that they address free sets instead of independent sets. Proof of the lemmas is omitted.

Lemma 2.2.12 *Let G be a $\langle\rho,\varepsilon\rangle$-connected graph. Let $R = \{r_1, \dots, r_k\}$ be an ordered set in G of size k. The number of vertices r_i which satisfy $r_i \in HI_{i-1}$ is bounded by $t = \frac{\rho}{\varepsilon}$. If G is pairwise $\langle\rho,\varepsilon\rangle$-connected then the number of vertices r_i which satisfy $r_i \in HI_{i-1}$ is bounded by $t = \frac{2\rho^2\lceil\log(1/\rho)\rceil}{\varepsilon}$.*

Lemma 2.2.13 *Let G be a $\langle\rho,\varepsilon\rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.12. Let $k \geq 2t$. Let $R = \{r_1, \dots, r_k\}$ be k random vertices of G. The probability that R_k induces a free set is at most*

$$\rho^k \left(\frac{ek}{t\rho}\right)^t$$

We now address the probability that a random subset R of H is a maximum free set. We will then use the union bound on all subsets R of H of size $> \rho s$ to obtain our results.

Lemma 2.2.14 *Let G be a $\langle\rho,\varepsilon\rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.12. Let $k \geq 2t$. Let H be an ordered random sample of G of size $s \geq k$. The probability that a given subset R of H is a maximum free set is at most*

$$\rho^k \left(\frac{ek}{t\rho}\right)^t (1-\rho)^{s-k}$$

Proof: Let $R = \{r_1, \dots, r_k\}$ (ordered by the ordering induced by H). The set R is a maximum free set in H only if (a) R is free, and (b) For each vertex $h \in H$ which is not in R, the ordered set $R^+ = \{r_1, \dots, r_j, h, r_{j+1}, \dots, r_k\}$ is not free. Here the index j is such that r_j appears before h in the ordering of H, and r_{j+1} appears after h (*i.e.* R^+ is ordered according to the ordering of H).

The probability that R is free has been analyzed in Lemma 2.2.13. It is left to analyze the probability that R^+ is not free for every vertex $h \notin R$, given that R is free. Consider a vertex $h \in H$ which is not in R, and let $R^+ = \{r_1, \dots, r_j, h, r_{j+1}, \dots, r_k\}$.

Claim 2.2.15 *Let $R = \{r_1, \dots, r_k\}$ be a free set. Let $R^+ = \{r_1, \dots, r_j, h, r_{j+1}, \dots, r_k\}$. Let the partition corresponding to $R_j = \{r_1, \dots, r_j\}$ be (LI_j, HI_j, N_j). If $h \in LI_j$ then R^+ is also a free set.*

Proof: We will use the following notation. Let $R_i = \{r_1, \dots, r_i\}$ denote the first i vertices of R, and let (LI_i, HI_i, N_i) be its corresponding partition. For $i > j$, let $R_i^+ = \{r_1, \dots, r_j, h, r_{j+1}, \dots, r_i\}$ denote the first $i+1$ vertices of R^+, and let (LI_i^+, HI_i^+, N_i^+) be its corresponding partition. Finally let R_h^+ denote the subset $\{r_1, \dots, r_j, h\}$ and (LI_h^+, HI_h^+, N_h^+) be its corresponding partition.

We would like to prove that R^+ is free. That is, we would like to show (a) that $r_i \in I_{i-1}$ for each $i \leq j$, (b) that $h \in I_j$, (c) that $r_{j+1} \in I_h^+$, and (d) that $r_i \in I_{i-1}^+$ for $i \geq j+2$. Recall that R is free and thus $r_i \in I_{i-1}$ for all $i \in \{1, \dots, k\}$.

The first assertion follows from the fact that the first j vertices of R and R^+ are identical. The second follows from the assumption that $h \in LI_j$. For the third assumption, notice (as $h \in LI_j$) that the partition corresponding to $R^+_h = \{r_1, \ldots, r_j, h\}$ is equal to the partition corresponding to $R_j = \{r_1, \ldots, r_j\}$. This follows from our definition of the partition (LI^+_h, HI^+_h, N^+_h). As $r_{j+1} \in I_j$ we conclude that $r_{j+1} \in I^+_h$.

For the final assertion, observe that for any $i \geq j+1$, the partition corresponding to R^+_i is equal to the partition corresponding to R_i. This can be seen by induction (on i). We start with the partitions corresponding to R_{j+1} and R^+_{j+1}. The partition $(LI_{j+1}, HI_{j+1}, N_{j+1})$ is defined uniquely by the partition corresponding to R_j and the vertex r_{j+1}. Similarly, the partition $(LI^+_{j+1}, HI^+_{j+1}, N^+_{j+1})$ is defined uniquely by the partition corresponding to R^+_h and the vertex r_{j+1}. As the partition corresponding to R^+_h is equal to the partition corresponding to R_j, we conclude that the same hold for the partitions corresponding to R_{j+1} and R^+_{j+1}. The inductive step is done similarly. The partition corresponding to R_i (R^+_i) is defined uniquely by the partition corresponding to R_{i-1} (R^+_{i-1}) and the vertex r_i. As the partition corresponding to R_{i-1} equals that corresponding to R^+_{i-1} we conclude our claim. As R is free, $r_i \in I_{i-1}$ for every $i \geq j+2$. This implies, that $r_i \in I^+_{i-1}$ also, which proves the final assertion. □

Claim 2.2.15 implies that the probability that $R^+ = \{r_1, \ldots, r_j, h, r_{j+1}, \ldots, r_k\}$ is not free given that R is free is at most $(1-\rho)$ (recall that the set LI_j is of size exactly ρn). This holds independently for every vertex h in $H \setminus R$. We conclude that the probability that R is a maximum free subset of H is at most the probability that R is free times $(1-\rho)^{s-k}$. □

We now turn to analyze the probability that a random ordered subset H of G of size s has a free set of size larger than ρs. We follow the line of analysis given in Section 2.2.1 and analyze the probability that H has a free set of size larger than $\delta\rho s$ for any $\delta > 1$. We then get rid of the factor δ to obtain our main theorem of this section.

Corollary 2.2.16 *Let G be a $\langle\rho, \varepsilon\rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.12. Let H be a random sample of G of size s. Let $\delta > 1$, and let c be a sufficiently large constant. Let $\Gamma = \ln\delta - \frac{\delta-1}{\delta}$. If $s \geq \frac{ct}{\rho\Gamma}(\log(1/\rho) + \log(1+1/\Gamma))$ then the probability that H has a free set of size $> \delta\rho s$ is at most*

$$s\left(\frac{1}{e^{\Gamma}}\right)^{\Omega(\delta\rho s)}$$

Proof: Let $k = \delta\rho s$, let $\delta' > \delta$ and let $k' = \delta'\rho s > k$. Using Lemma 2.2.14, the probability that there is a maximum free set R in H of size k' is at most

$$\begin{aligned}
\sum_{k'>k}\binom{s}{k'}\rho^{k'}\left(\frac{ek'}{t\rho}\right)^t(1-\rho)^{s-k'} &\leq \sum_{k'>k}\left(\frac{ek'}{t\rho}\right)^t\frac{s^s(1-\rho)^{s-k'}}{k'^{k'}(s-k')^{s-k'}}\rho^{k'} \\
&\leq \sum_{k'>k}\left(\frac{ek'}{t\rho}\right)^t\frac{e^{k'\frac{\delta'-1}{\delta'}}}{\delta'^{k'}} \\
&\leq \sum_{k'>k}\left[\frac{ek'}{t\rho}\left(\frac{1}{e^{\ln\delta' - \frac{\delta'-1}{\delta'}}}\right)^{\frac{k'}{t}}\right]^t
\end{aligned}$$

$$\leq \sum_{k'>k} \left[\frac{ek'}{t\rho} \left(\frac{1}{e^{\Gamma}}\right)^{\frac{k'}{t}} \right]^t$$

$$\leq \sum_{k'>k} \left(\frac{1}{e^{\Gamma}}\right)^{\Omega(k')} \leq s \left(\frac{1}{e^{\Gamma}}\right)^{\Omega(k)}$$

We use the facts that $\frac{k'}{t}$ is greater than both $c\frac{1}{\Gamma}\log(1+1/\Gamma)$ and $c\frac{1}{\Gamma}\log(1/\rho)$ for a sufficiently large constant c. $\square$

It is left to get rid of the additional parameter δ of Corollary 2.2.16 (namely to analyze the probability that $\alpha(H) > \rho s$).

Lemma 2.2.17 *If G is (pairwise) $\langle \rho, \varepsilon \rangle$-connected then G is also (pairwise) $\left\langle \rho(1-\frac{\varepsilon}{4\rho^2}), \frac{\varepsilon}{2} \right\rangle$-connected.*

Proof: We present proof for the case in which G is $\langle \rho, \varepsilon \rangle$-connected, a similar proof holds for the case in which G is pairwise $\langle \rho, \varepsilon \rangle$-connected. Let A be some subset of G of size $\rho(1-\frac{\varepsilon}{4\rho^2})n$. Let A^c be any set in $V \setminus A$ of size $\frac{\varepsilon}{4\rho}n$. It is known that the number of edges induced by the set $A \cup A^c$ is at least $\frac{\varepsilon}{2}n^2$ (notice that $|A \cup A^c| = \rho n$ and $E(A \cup A^c, A \cup A^c) \geq \varepsilon n^2$). The number of edges (in $A \cup A^c$) adjacent to vertices in A^c is bounded by $\frac{\varepsilon}{4\rho}\rho n^2 = \frac{\varepsilon}{4}n^2$. Hence, the number of edges induced by vertices in A is at least $\frac{\varepsilon n^2}{4}$ implying that $E(A, A) \geq \frac{\varepsilon}{2}n^2$. $\square$

Theorem 2.2.4 *Let G be a $\langle \rho, \varepsilon \rangle$-connected graph (pairwise or not). Let t be as in Lemma 2.2.12. Let H be a random sample of G of size s. Let c be a sufficiently large constant. If $s \geq ct\frac{\rho^3}{\varepsilon^2}\log\left(\frac{\rho}{\varepsilon}\right)$ then the probability that H has an independent set of size $> \rho s$ is at most $e^{-\Omega(t)}$.*

Proof: By Lemma 2.2.17, G is also $\left\langle \rho(1-\frac{\varepsilon}{4\rho^2}), \frac{\varepsilon}{2} \right\rangle$-connected (pairwise or not). Let $\rho' = \rho(1-\frac{\varepsilon}{4\rho^2})$ and $\varepsilon' = \varepsilon/2$. We would like to bound the probability that H does not have any independent sets of size greater than ρs. Let $\delta = 1 + \frac{\varepsilon}{4\rho^2}$. Notice that $\delta\rho' \leq \rho$. Hence, it suffices to bound the probability that $\alpha(H) > \delta\rho' s$. This probability, in turn, is at most the probability that H has a maximum free set of size greater than $k = \delta\rho' s$ (Claim 2.2.11).

Let $\Gamma = \ln(\delta) - \frac{\delta-1}{\delta}$. It is not hard to verify that $\Gamma = \theta((\delta-1)^2) = \theta\left(\frac{\varepsilon^2}{\rho^4}\right)$ for our value of δ. By our assumption

$$s \geq ct\frac{\rho^3}{\varepsilon^2}\log\left(\frac{\rho}{\varepsilon}\right) \geq \frac{ct}{\rho(\delta-1)^2}\left(\log(1/\rho') + \log(1+1/(\delta-1)^2)\right) \geq \frac{ct}{\rho'\Gamma}\left(\log(1/\rho') + \log(1+1/\Gamma)\right)$$

where in the above the constant c may change values from line to line. Now, by Corollary 2.2.16, for our choice of s, the probability that H has a maximum free set of size greater than $k = \delta\rho' s$ is at most $s\left(\frac{1}{e^{\Gamma}}\right)^{\Omega(\delta\rho s)} \leq e^{-\Omega(t)}$. $\square$

Roughly speaking, Theorem 2.2.4 states that given a $\langle \rho, \varepsilon \rangle$-connected graph G, a random sample H of G of size s proportional to $\frac{\rho^4}{\varepsilon^3}$ (or larger) will not have an independent set of size ρs (with high probability). This improves upon the bound of $s \simeq \frac{\rho}{\varepsilon^4}$ presented in [GGR98] both in the dependence on ρ (as $\rho < 1$) and in the dependence on ε. Moreover, we present a further improvement to $s \simeq \frac{\rho^5}{\varepsilon^3}$ if our graphs are considered to be pairwise $\langle \rho, \varepsilon \rangle$-connected. In Section 2.7 we continue to study the minimal value of s for which $\alpha(H) < \rho s$ with high probability, and present a lower bound on the size of s which is proportional to $\frac{\rho^3}{\varepsilon^2}$.

2.3 The continuous graph

Let d be a large constant, and let $S^{d-1} = \{v \in R^d \mid \|v\| = 1\}$ be the d dimensional unit sphere. Let $G_k = (V, E)$ be the continuous graph in which: *(a)* The vertex set V consists of all the points on the unit sphere S^{d-1}. *(b)* The edge set E of G consists of all pairs of vertices whose respective vectors (from the origin) form an angle of at least $\arccos(-1/(k-1))$. As the size of V and E is infinite (and uncountable), terms such as the *number* of vertices in V will be replaced by the continuous analogue *measure*.

In this section we prove that G_k is pairwise $\langle \rho, \varepsilon \rangle$-connected (for certain values of ρ and ε). We then use this fact in Section 2.4 to prove the main theorem of our work. In our analysis, we will assume that the dimension d is (at least) a very large constant (our proofs rely on such d). Additional constants that will be presented in the remainder of this section are to be viewed as independent of d.

Definition 2.3.1 (Sphere measure) *Let μ be the normalized $(d-1)$ dimensional natural measure on S^{d-1}, and let μ^2 be the induced measure on $S^{d-1} \times S^{d-1}$. For any two (not necessarily disjoint) subsets A and B of V, we define the measure of edges from A to B as*

$$E(A, B) = \mu^2 \left(\{(x, y) \mid x \in A,\ y \in B,\ (x, y) \in E\} \right).$$

Definition 2.3.2 (Sphere caps) *Let $a \in [0, 1]$, and $x \in S^{d-1}$. An a-cap centered at x is defined to be the set $\mathcal{C}_a = \{u \in S^{d-1} \mid \langle u, x \rangle \geq a\}$. Denote the measure of an a-cap by $\rho(a)$.*

A few remarks are in place. Notice that for every x, x', an a-cap centered at x has the same measure as an a-cap centered at x' (sphere symmetry). Furthermore, notice that *large* caps have *small* corresponding values of a and vice versa. The value of $\rho(a) = \mu(\mathcal{C}_a)$ is approximately given by the following lemma proven for example in [FS02].

Lemma 2.3.3 *Let $\mathcal{C}_a$ be an a-cap centered at some $x \in S^{d-1}$. There exists a constant $c > 0$ (independent of a and d) s.t.*

$$\frac{c}{\sqrt{d}} \left(1 - a^2\right)^{\frac{d-1}{2}} \leq \mu(\mathcal{C}_a) = \rho(a) \leq \frac{1}{2} \left(1 - a^2\right)^{\frac{d-1}{2}}.$$

Lemma 2.3.4 *Let $\rho(\frac{1}{k-1})$ be the measure of a $\frac{1}{(k-1)}$-cap. Every vertex v in the graph G_k has degree $\rho(\frac{1}{k-1})$.*

Proof: Each vertex v in G_k is adjacent to every vertex in the $\frac{1}{(k-1)}$-cap centered at $-v$ (here $-v$ is the vertex antipodal to v). Hence, the measure of vertices adjacent to a given vertex v in G_k is that of a $\frac{1}{(k-1)}$-cap. $\square$

The main property of G_k of our interest is the measure of edges between any two given subsets of V of a specified size. We first prove that the sets in G_k which share the least amount of edges are caps with the same center.

Theorem 2.3.5 *Let $1 \geq a > 0$ and let A and B be two (not necessarily disjoint) measurable sets in V of measure $\rho(a)$. Let x be an arbitrary vertex of S^{d-1}. The minimum of $E(A, B)$ is obtained when $A = B = \mathcal{C}_a$ where $\mathcal{C}_a$ is an a-cap of measure $\rho(a)$ centered at x.*

The proof of Theorem 2.3.5 is based on symmetrization techniques similar to those presented in [BT76, FS02]. We prove Theorem 2.3.5 in Section 2.9. We now turn to analyze the measure of edges between caps of measure $\rho(a)$. Namely we study the value of $E(\mathcal{C}_a, \mathcal{C}_a)$. For *large* values of a, it is not hard to see that $\mathcal{C}_a$ will not induce any edges. This follows from the fact that any two vertices u and v in $\mathcal{C}_a$ satisfy $\langle u, v\rangle \geq 2a^2 - 1$. Hence, in the following we consider values of a which satisfy $2a^2 - 1 < -\frac{1}{k-1}$. For such values of a we show that $E(\mathcal{C}_a, \mathcal{C}_a)$ is proportional to $\mu^2(E)\rho(a)^{2+\frac{2}{k-2}}$ (which we denote by $\lambda(a)$). Notice that two random subsets A and B of G_k of measure $\rho(a)$ are expected to satisfy $E(A, B) = \mu^2(E)\rho(a)^2$ which is fairly close to $\lambda(a)$. Actually we show that $E(\mathcal{C}_a, \mathcal{C}_a)$ is in the range $\left[\left(1 - c\sqrt{\frac{\log(d)}{d}}\right)^{\frac{d-1}{2}} \lambda(a), \lambda(a)\right]$ for some large constant c. It is not hard to verify that the term $\left(1 - c\sqrt{\frac{\log(d)}{d}}\right)^{\frac{d-1}{2}}$ is negligible compared to $\lambda(a)$ once d is taken to be large enough.

Theorem 2.3.6 *Let $\sqrt{\frac{k-2}{2(k-1)}} > a > 0$ and $k > 2$ be constant. Let $x \in S^{d-1}$, let $\mathcal{C}_a$ be an a-cap centered at x. Let $\varepsilon(a)$ be the value of $E(\mathcal{C}_a, \mathcal{C}_a)$. Finally let*

$$\lambda(a) = \left(\left(1 - \frac{1}{(k-1)^2}\right)\left(1 - \frac{2(k-1)}{k-2}a^2\right)\right)^{\frac{d-1}{2}}.$$

Then $\varepsilon(a) \in \left[\left(1 - c\sqrt{\frac{\log(d)}{d}}\right)^{\frac{d-1}{2}} \lambda(a), \lambda(a)\right]$ for some constant $c > 0$.

Proof: Let $x \in S^{d-1}$. Let $\mathcal{C}_a$ be an a-cap centered at x. W.l.o.g. we will assume that $x = (1, 0, \ldots, 0)$. Consider a vertex $v \in \mathcal{C}_a$ on the boundary of $\mathcal{C}_a$. Let $N(v)$ be the set of vertices adjacent to v. We start by computing the measure of vertices that are neighbors of v and are in the cap $\mathcal{C}_a$, *i.e.* the measure of $N(v) \cap \mathcal{C}_a = \{u = (u_1, \ldots, u_d) \in S^{d-1} \mid u_1 \geq a \text{ and } \langle v, u\rangle \leq -1/(k-1)\}$.

Claim 2.3.7 *Let a, k be as in Theorem 2.3.6. Let $v = (a, \sqrt{1-a^2}, 0, \ldots, 0)$ be a vertex on the boundary of $\mathcal{C}_a$. Let $N(v)$ be the set of neighbors of v. Let $z = \sqrt{a^2 + \frac{(1/(k-1)+a^2)^2}{1-a^2}}$. Finally let $\delta = c\sqrt{\frac{\log(d)}{d}}$ for a sufficiently large constant c. The measure of vertices in $N(v) \cap \mathcal{C}_a$ satisfies*

$$(1-\delta)^{\frac{d-1}{2}}\left(1 - z^2\right)^{\frac{d-1}{2}} \leq \mu(N(v) \cap \mathcal{C}_a) \leq \left(1 - z^2\right)^{\frac{d-1}{2}}$$

Claim 2.3.7 addresses the measure of $\mathcal{C}_a \cap N(v)$, and states that it is essentially the measure of a z-cap (where $z = \sqrt{a^2 + \frac{(1/(k-1)+a^2)^2}{1-a^2}}$). To prove Claim 2.3.7 we study the measure of certain restricted sets in S^{d-1}. These sets are studied in Claim 2.3.8. Claims 2.3.7 and 2.3.8 are depicted in Figure 2.1 and proven in Section 2.9.

Claim 2.3.8 *Let a, z, δ be as in Claim 2.3.7. Let $b^2 = z^2 - a^2$. Let $N = \{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 = a,\ u_2 = b\}$. Let N_δ be a δ neighborhood of N (*i.e.* all points which are of distance less that δ from the set N). The measure of the set N_δ is at least $\left(1 - z^2\right)^{\frac{d-1}{2}}$.*

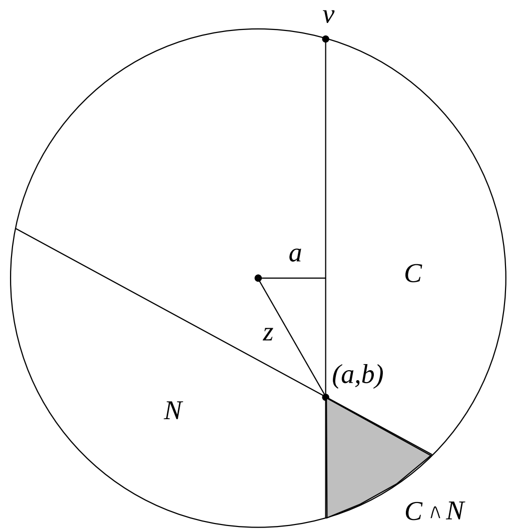

Figure 2.1: Projecting S^{d-1} onto the two dimensional subspace $\{(r_1, r_2, 0, \ldots, 0) \mid r_1, r_2 \in R\}$ of R^d we obtain the circle above. C is the projection of an a-cap $\mathcal{C}_a$ centered at $(1, 0, \ldots, 0)$ (where a is the distance of the cap from the origin). The vertex $v = (a, \sqrt{1-a^2}, 0, \ldots, 0) \in \mathcal{C}_a$ is on the boundary of $\mathcal{C}_a$. N is the projection of the set of points $N(v)$ on the sphere that are adjacent to v (*i.e.* form an angle of at least $\arccos(-1/(k-1))$ with v). The shaded section is $C \cap N$ (the projection of $\mathcal{C}_a \cap N(v)$). The point (a, b) is the closest point of the projection of $\mathcal{C}_a \cap N(v)$ to the origin. It is not hard to verify that $b^2 = \frac{(1/(k-1)+a^2)^2}{1-a^2}$. Finally, we denote the value of $\sqrt{a^2+b^2}$ by z. Claim 2.3.7 addresses the measure of $\mathcal{C}_a \cap N(v)$, and states that it is essentially the measure of a z-cap. This is done by studying the points in S^{d-1} who's projection falls *close* to (a, b). Roughly speaking, we first show that such points are in $\mathcal{C}_a \cap N(v)$, then using Claim 2.3.8 we show that the measure of these points is essentially the measure of a z-cap.

To complete the proof of Theorem 2.3.6, let $a,\ z,\ \delta$ be as in Claim 2.3.7. For a vertex $v \in \mathcal{C}_a$ let $N(v) \cap \mathcal{C}_a$ be the set of vertices adjacent to v in $\mathcal{C}_a$. For the upper bound, notice that of all vertices in $\mathcal{C}_a$, the vertices v in which $\mu(N(v) \cap \mathcal{C}_a)$ is largest are the vertices on the boundary of $\mathcal{C}_a$. By Claim 2.3.7, for these vertices $\mu(N(v) \cap \mathcal{C}_a)$ is bounded by $(1-z^2)^{\frac{d-1}{2}}$. We thus conclude, that $\varepsilon(a)$ is bounded by the measure of vertices in $\mathcal{C}_a$ times $(1-z^2)^{\frac{d-1}{2}}$. That is,

$$\varepsilon(a) \leq \rho(a)(1-z^2)^{\frac{d-1}{2}} \leq (1-a^2)\left(1 - \left(a^2 + \frac{(1/(k-1)+a^2)^2}{1-a^2}\right)\right)^{\frac{d-1}{2}} = \lambda(a).$$

As for the lower bound, let $w = (w_1, w_2, \ldots, w_d)$ be a vertex in $\mathcal{C}_a$ with first coordinate w_1 of value $a+\delta$. Consider any vertex $v = (v_1, v_2, \ldots, v_d) \in \mathcal{C}_a$ with first coordinate v_1 of value less than $a+\delta$. It is not hard to verify that the measure of $N(v) \cap \mathcal{C}_a$ is greater than the measure of $N(w) \cap \mathcal{C}_a$. Using an analysis similar to that of Claim 2.3.7 we have that

$$\mu(N(w) \cap \mathcal{C}_a) \geq (1-c\delta)^{\frac{d-1}{2}} \left(1 - \left(a^2 + \frac{(1/(k-1)+a(a+\delta))^2}{1-(a+\delta)^2}\right)\right)^{\frac{d-1}{2}} \geq (1-c\delta)^{\frac{d-1}{2}} \left(1-z^2\right)^{\frac{d-1}{2}}$$

for a sufficiently large constant c (which changes values between both sides of the second inequality). Furthermore, for our choice of δ, the measure of vertices $v = (v_1, v_2, \ldots, v_d)$ in $\mathcal{C}_a$ with $v_1 \le a + \delta$ is at least $\rho(a)/2$ (Claim 2.9.3). Hence, we conclude that $E(\mathcal{C}_a, \mathcal{C}_a)$ is at least $\frac{\rho(a)}{2}(1-c\delta)^{\frac{d-1}{2}}\left(1-z^2\right)^{\frac{d-1}{2}}$. Simplifying the above expression, we conclude our assertion. □

Theorem 2.3.6 addresses the case in which k is constant and the caps considered are both of measure $\rho(a)$ for a constant value of a. For the proof of Theorem 2.1.1 (2) we also need to address nonconstant values of a and k which depend on d.

Theorem 2.3.9 *Let $a = \left(\frac{\log(d)}{d}\right)^{\frac{1}{4}}$. Let k satisfy $\frac{1}{k-1} = a^2$. Let $x \in S^{d-1}$, let $\mathcal{C}_a$ be an a-cap centered at x. Let $\varepsilon(a)$ be the value of $E(\mathcal{C}_a, \mathcal{C}_a)$. The value of $\varepsilon(a)$ is in the range*

$$\left[\frac{1}{poly(d)}\left(1-\frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}},\ \left(1-\frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}}\right].$$

The outline of the proof of Theorem 2.3.9 is similar to that of Theorem 2.3.6. Full proof appears in Section 2.9. Combining Theorems 2.3.5, 2.3.6, 2.3.9 we obtain:

Corollary 2.3.10 *Let a, k, $\varepsilon(a)$ be defined as in Theorem 2.3.6 or 2.3.9. The value of $E(A, B)$ for any two subsets A and B both of measure $\rho(a)$ is at least $\varepsilon(a)$.*

Roughly speaking, Corollary 2.3.10 implies (for certain values of a) that the graph G_k is pairwise $\langle\rho(a), \varepsilon(a)\rangle$-connected. In Section 2.2.1 the notion of $\langle\rho, \varepsilon\rangle$-connected graphs was defined for finite graphs only. In what follows we define the infinite analog of $\langle\rho, \varepsilon\rangle$-connected graphs, and show that the Theorems proven in Section 2.2 apply for such graphs also.

Definition 2.3.11 *Let $\rho < 1$. A graph $G = (V, E)$ is said to be $\langle\rho, \varepsilon\rangle$-connected iff every subset A of V of measure ρ satisfies $E(A, A) \ge \varepsilon$ (*i.e. *the measure of edges in the subgraph induced by A is greater than $\frac{\varepsilon}{2}$).*

Definition 2.3.12 *Let $\rho < 1$. A graph $G = (V, E)$ is said to be pairwise $\langle\rho, \varepsilon\rangle$-connected iff every two subsets A and B of V of measure ρ satisfy $E(A, B) \ge \varepsilon$.*

Corollary 2.3.13 *Let a, k, $\varepsilon(a)$ be defined as in Theorem 2.3.6 or 2.3.9. The graph G_k is pairwise $\langle\rho(a), \varepsilon(a)\rangle$-connected.*

We would now like to show that the analysis of Section 2.2 also holds for our continuous graph G_k. Namely we would like to prove the following analog of Corollary 2.2.8.

Theorem 2.3.14 *Let a, k, $\varepsilon(a)$ be defined as in Theorem 2.3.6 or 2.3.9. Let H be a random sample of s vertices of G_k (according to the uniform distribution on S^{d-1}). Let c be a sufficiently large constant. If $s \ge c\frac{\rho(a)}{\varepsilon(a)}\log^2(1/\rho(a))$ then the probability that $\alpha(H) > e^2\rho(a)s$ is at most $1/4$.*

Proof: Let $H = \{h_1, \ldots, h_s\}$ be s random points of the unit sphere (that is H is a random subset of G_k of size s). Let $R_k = \{r_k, \ldots, r_k\}$ be a subset of H of size k. Finally, for $i \in \{1, \ldots, k\}$ let $R_i = \{r_1, \ldots, r_i\}$ and (LI_i, HI_i, N_i) be its corresponding partition (as defined in Section 2.2.1). For each i the subsets LI_i, HI_i, and N_i are measurable (this follows from the fact that the neighborhood of each vertex is a cap of S^{d-1}). Hence, the proof of Lemma 2.2.5 and Theorem 2.2.7 hold under Definitions 2.3.11 and 2.3.12 presented above. This suffices to conclude our assertion. □

2.4 Proof of Theorem 2.1.1

In this section we prove Theorem 2.1.1. Recall that we are looking for a graph H for which both the vector chromatic number and the size of the maximum independent set are small. The graphs H we present are random subgraphs of the graphs G_k defined in Section 2.3. The three assertions of Theorem 2.1.1 are all proven similarly, the main difference between their proof is the choice of parameters used. To avoid confusion, we restate Theorem 2.1.1 using a slightly different notation then that appearing in the original presentation.

Theorem 2.1.1 (restated)

1. *For every constant $\gamma > 0$ and constant $k > 2$, there are infinitely many graphs H that are vector k-colorable and satisfy $\alpha(H) \leq \frac{s}{\Delta_H^{1-\frac{2}{k}-\gamma}}$, where s is in the number of vertices in H, Δ_H is the maximum degree in H, and $\Delta_H > s^{\delta}$ for some constant $\delta > 0$.*
2. *For some constant c, there are infinitely many graphs H of size s that are $O(\frac{\log s}{\log\log s})$ vector colorable and satisfy $\alpha(H) \leq (\log s)^c$.*
3. *There are infinitely many graphs H of size s which are vector 3-colorable and satisfy $\alpha(H) \leq s^{0.843}$.*

Proof (Theorem 2.1.1 (1)): Let $k > 2$ be constant. Let $\gamma > 0$ be an arbitrarily small constant. Let c be a sufficiently large constant and let $a = \gamma/c$. Let $G = G_k = (V, E)$ be the continuous graph from Section 2.3 (here and in the remainder of the proof, we assume the dimension d of the graph G_k is taken to be significantly larger than $1/\gamma$). Finally, let $\Delta = \rho(\frac{1}{k-1})$ be the measure of vertices adjacent to any given vertex of G.

Recall (Corollary 2.3.13) that G is pairwise $\langle \rho(a), \varepsilon(a)\rangle$-connected. Let $\rho = \rho(a)$ and $\varepsilon = \varepsilon(a)$. This implies (Theorem 2.3.14) that with probability $\geq 3/4$ a random subset H of G of size $s \geq c\frac{\rho}{\varepsilon}\log^2(1/\rho)$ satisfies $\alpha(H) \leq e^2\rho s$ (recall that c is a sufficiently large constant). We start by simplifying the expression bounding s.

Claim 2.4.1 *A random subset H of G of size $s = \frac{1}{\Delta\rho^{\frac{k}{k-2}+\gamma}}$ satisfies $\alpha(H) \leq e^2\rho s = \frac{e^2}{\Delta\rho^{\frac{2}{k-2}+\gamma}}$ with probability $\geq 3/4$.*

Proof: It suffices to prove that $s = \frac{1}{\Delta\rho^{\frac{k}{k-2}+\gamma}} \geq c\frac{\rho}{\varepsilon}\log^2(1/\rho)$. The claim follows (by basic calculations) from the fact that ε can be bounded by $\Delta\rho^{\frac{2(k-1)}{k-2}+\gamma}$. By Theorem 2.3.6, we have that

$$\varepsilon(a) = \left[\left(1 - c\left(\sqrt{\frac{\log(d)}{d}}\right)\right)\left(1 - \frac{1}{(k-1)^2}\right)\left(1 - \frac{2(k-1)}{k-2}a^2\right)\right]^{\frac{d-1}{2}}$$

It is not hard to verify that $\left(1 - c\left(\sqrt{\frac{\log(d)}{d}}\right)\right)^{\frac{d-1}{2}} > \rho^{\gamma/2}$ Furthermore, by Lemma 2.3.3 we have that $\left(1 - \frac{1}{(k-1)^2}\right)^{\frac{d-1}{2}} > \Delta$. Hence, $\varepsilon(a) \geq \rho^{\gamma/2}\left(1 - \frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}}\Delta$. Recall that a was defined as γ/c for a sufficiently large constant c. This implies that $\gamma \geq 2\frac{a^2\left(\frac{2(k-1)}{k-2}\right)^2}{1-a^2\frac{2(k-1)}{k-2}}$ ($k > 2$ is constant). This expression is designed to fit the requirement appearing in Claim 2.9.1 (of Section 2.9). Now by Claim 2.9.1, it holds that $\left(1 - \frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}} \geq \rho^{\frac{2(k-1)}{k-2}+\gamma/2}$. We conclude that $\varepsilon(a) \geq \Delta\rho^{\frac{2(k-1)}{k-2}+\gamma}$. □

Let H be a random subgraph of G of size $s = \frac{1}{\Delta\rho^{\frac{k}{k-2}+\gamma}}$. We will show that H satisfies the asserted conditions with probability greater than 1/2. First notice that any subgraph of G is vector k-colorable, including the subgraph H. Secondly, by Claim 2.4.1, $\alpha(H) \leq \frac{e^2}{\Delta\rho^{\frac{2}{k-2}+\gamma}}$ with probability $\geq 3/4$. It is left to analyze the maximum degree of H.

Claim 2.4.2 *With probability greater than* $3/4$*, the maximum degree* Δ_H *of the subgraph* H *is in the range* $[\frac{1}{2}\Delta s, 2\Delta s]$.

Proof: Consider a vertex $h \in H$. Let d_h be the degree of h. As every vertex in G is of degree Δ, the expected value of d_h is $\Delta(s-1)$. Thus (using standard bounds) the probability that d_h deviates from its expectation by more than a constant fraction of its expectation is at most $2^{-\Omega(\Delta s)}$. The probability that some vertex in H has degree $\notin [\frac{1}{2}\Delta s, 2\Delta s]$ is thus at most $2^{\log(s)-\Omega(\Delta s)} \leq 3/4$ for our choice of s. □

The first assertion of Theorem 2.1.1 now follows using basic calculations. □

Proof (Theorem 2.1.1 (2)) Let d be a large constant. Let $k-1 = \sqrt{\frac{d}{\log(d)}}$. Let $a^2 = \frac{1}{k-1}$. Let $G = G_k = (V, E)$ be the continious graph from Section 2.3.

Recall (Corollary 2.3.13) that G is pairwise $\langle\rho(a), \varepsilon(a)\rangle$-connected. Let $\rho = \rho(a)$ and $\varepsilon = \varepsilon(a)$. This implies (Theorem 2.3.14) that with probability $\geq 3/4$ a random subset H of size $s \geq c\frac{\rho}{\varepsilon}\log^2(1/\rho)$ satisfies $\alpha(H) \leq e^2\rho s$ (here c is a sufficiently large constant). As before we start by simplifying the expression bounding s.

Claim 2.4.3 *There exists a constant* γ *s.t with probability at least* $3/4$ *a random set* H *of* G *of size* $s = \frac{d^\gamma}{\rho}\log^2(1/\rho)$ *satisfies* $\alpha(H) \leq e^2\rho s$ *with probability* $\geq 3/4$.

Proof: Recall that $k-1 = \sqrt{\frac{d}{\log(d)}}$, $a^2 = \frac{1}{k-1}$. As before it suffices to bound $\varepsilon = \varepsilon(a)$. By Theorem 2.3.9, we have

$$\varepsilon \geq \frac{1}{poly(d)}\left(1 - a^2\frac{2(k-1)}{k-2}\right)^{\frac{d-1}{2}}.$$

Furthermore using Claims 2.9.1 and 2.9.2 (of Section 2.9) we obtain,

$$\left(1 - a^2\frac{2(k-1)}{k-2}\right)^{\frac{d-1}{2}} = \left(1 - a^2\left(2 + \frac{2}{k-2}\right)\right)^{\frac{d-1}{2}} \geq \frac{1}{poly(d)}\left(1-a^2\right)^{d-1} \geq \frac{\rho^2}{poly(d)}.$$

We conclude that there exists a constant γ such that $\varepsilon \geq \frac{\rho^2}{d^\gamma}$. □

Let H be a random subset of vertices of G of size $s = \frac{d^\gamma}{\rho}\log^2(1/\rho)$. Notice that $\log(s) = \theta(\sqrt{d \log d})$ and $s \leq \theta(\frac{d^{\gamma+2}}{\rho})$. By definition, G is k vector colorable. This implies that any subgraph of G (including that induced by H) is $k = O\left(\frac{\log(s)}{\log(\log(s))}\right)$ vector colorable, which completes the proof of the first part of our assertion. For the second part of our assertion, by Claim 2.4.3 the subset H does not have an independent set of size $e^2\rho s \leq \log^\gamma(s)$ with probability at least 3/4 (for some different constant γ). $\square$

Proof (Theorem 2.1.1 (3)) The proof follows the line of proof appearing above. In general, we use the graph $G = G_3$, but this time the value of a is set to be $a = 0.36$. Again, G is pairwise $\langle \rho(a), \varepsilon(a)\rangle$-connected where $\varepsilon(a)$ can be bounded by approximately $\left(\frac{3}{4}(1-4a^2)\right)^{\frac{d-1}{2}}$. Let $\rho = \rho(a)$ and $\varepsilon = \varepsilon(a)$. By Corollary 2.3.14, a random subset H of size $s \geq c\frac{\rho}{\varepsilon}\log^2(1/\rho)$ does not have an independent set of size $e^2\rho s$ (with probability $\geq 3/4$). Computing the value of $\log_s \rho s$ we obtain our assertion. We would like to note, that results of a similar nature can be obtained using the above techniques for any value of k. $\square$

2.5 The Lovász θ function and its variants

There are many equivalent ways to define the Lovász θ function and its variants. We will follow the definitions suggested in [KMS98, Cha02]. Let $G = (V, E)$ be a graph of size n. For convenience we will assume that $V = [1, \ldots, n]$. The semidefinite relaxations below assign unit vectors to every vertex $i \in V$. These unit vectors are to satisfy certain constraints which will in turn determine the value of the relaxations.

$COL_1(G)$ Minimize k subject to:

$$\begin{array}{ll} \langle v_i, v_j\rangle \leq -\frac{1}{k-1} & \forall (i,j) \in E \\ \langle v_i, v_i\rangle = 1 & \forall i \in V \end{array}$$

$COL_2(G)$ Minimize k subject to:

$$\begin{array}{ll} \langle v_i, v_j\rangle = -\frac{1}{k-1} & \forall (i,j) \in E \\ \langle v_i, v_i\rangle = 1 & \forall i \in V \end{array}$$

$COL_3(G)$ Minimize k subject to:

$$\begin{array}{ll} \langle v_i, v_j\rangle = -\frac{1}{k-1} & \forall (i,j) \in E \\ \langle v_i, v_j\rangle \geq -\frac{1}{k-1} & \forall i,j \in V \\ \langle v_i, v_i\rangle = 1 & \forall i \in V \end{array}$$

The function $COL_1(G)$ is the vector chromatic number of G as defined in [KMS98]. The function $COL_2(G)$ is the strict vector chromatic number of G and is equal to the Lovász θ function on $\bar{G}$, where $\bar{G}$ is the *complement* graph of G. Finally, the function $COL_3(G)$ will be referred to as the *strong* vector chromatic number as defined in [Sze94, Cha02]. Let $\omega(G)$ denote the size of the maximum clique in G, in the following we show that

$$\omega(G) \leq COL_1(G) \leq COL_2(G) \leq COL_3(G) \leq \chi(G)$$

It is not hard to verify that $COL_1(G) \leq COL_2(G) \leq COL_3(G)$. To show the other inequalities we need the following fact. For every integer k, the k unit vectors $\{v_1, \ldots, v_k\}$ that minimize the value of $\max_{i \neq j \in [1,\ldots,k]} \langle v_i, v_j \rangle$ are the vertices of a simplex in R^{k-1} centered at the origin. For each $i, j \in [1, \ldots, k]$, these vectors satisfy $\langle v_i, v_j \rangle = -\frac{1}{k-1}$.

Now to prove the inequality $COL_3(G) \leq \chi(G)$, consider a k coloring σ of G. The coloring σ partitions the vertex set V into k color classes $\{V_1, \ldots, V_k\}$. Assigning each color class V_i with the corresponding vector v_i above, we obtain a valid assignment for $COL_3(G)$. To show that $\omega(G) \leq COL_1(G)$, consider a graph G with maximum clique size $\omega(G)$. To obtain a valid assignment of vectors to $COL_1(G)$ of value k we require that all pairs of vectors corresponding to the vertices of the maximum clique will have inner product of value at most $-\frac{1}{k-1}$. As mentioned above, this can happen only if $\omega(G) \leq k$.

Remark 2.5.1 *The results of this chapter show a large gap between $COL_1(G)$ and $\chi(G)$ (Theorem 2.1.1). Combining these result with certain proof techniques appearing in [Sze94], a similar gap between $\omega(G)$ and $COL_3(G)$ can also be obtained. Details omitted.*

2.6 An alternative proof of Theorem 2.1.1 (1)

In Theorem 2.1.1 (1), we are interested in presenting a vector k-colorable graph H in which a special relationship is satisfied between its maximum degree and its maximum independent set. It is not hard to verify that the graphs G_k presented in Section 2.3 are far from satisfying this relationship, as the maximum degree Δ of these graphs is too large. We overcome this problem in Section 2.2 and Section 2.4 by considering a random (vertex induced) subgraph of G_k. We have shown that such a subgraph will suffice for proving the three parts of Theorem 2.1.1.

Another method for coping with the large maximum degree Δ of G_k was suggested by Luca Trevisan (private communication). Instead of sampling vertices from G_k at random in order to obtain a sparse graph, consider sampling **edges** at random. In the following section we combine the idea of edge sampling with our results from Section 2.3 and prove the first part of Theorem 2.1.1. It appears that the remainder of Theorem 2.1.1 can not be proven using edge sampling (as we are interested in graphs in which the maximum independent set is small with respect to the number of vertices in the graph).

To construct the graph H we will follow a three phase plan. Our starting point will be the continuous graph G_k (of Section 2.3) which is vector k-colorable and proven to be pairwise $\langle \rho, \varepsilon \rangle$-connected. We then define and analyze a (finite) discrete version G_k^d of G_k which will be shown to inherit many of the properties of G_k. Namely this discrete graph will be *almost* vector k-colorable, and will be pairwise $\langle \rho, \varepsilon \rangle$-connected. Finally, we will define H to be the graph obtained by randomly removing edges from the discrete graph G_k^d.

The discrete graph G_k^d We now define a discrete analog G_k^d of the continuous graph G_k from Section 2.3. Recall that the vertex set of G_k is the d dimensional unit sphere S^{d-1}. It is shown in [FS02] that S^{d-1} can be partitioned into $n = 2^{\theta(d^2)}$ cells each of equal size and of diameter at most 2^{-d}. Let $\mathcal{P} = \{C_1, \ldots, C_n\}$ denote the cells obtained in the above partition. The graph G_k^d will be of size n, in which each vertex $v_i \in V$ corresponds to a cell $C_i \in \mathcal{P}$. The edge set of G_k^d consists of the edges (u, v) iff there is a positive measure of edges in G_k^c between their corresponding cells C_u, C_v.

Lemma 2.6.1 *Let A and B be subsets of G_k^d, and let A_c and B_c be the corresponding subsets of G_k. (a) The size of A (B) is ρn iff A_c (B_c) has measure ρ. (b) $E(A,B) \geq E(A_c, B_c)n^2$. The definition of $E(A,B)$ is given in Definition 2.2.1 of Section 2.2.*

Proof: For the first part of the lemma, assume that the set A has ρn vertices, thus the corresponding subset A_c consists of ρn cells each of measure n^{-1}. We conclude that the subset A_c has measure exactly ρ. On the other hand, if A_c has measure ρ and consists of the union of k cells of measure n^{-1}, then k must be ρn. For the second part, assume that $E(A,B) = \varepsilon n^2$, then by the fact that each edge between A and B corresponds to at most the measure of n^{-2} edges between A_c and B_c we conclude that $E(A_c, B_c)$ is at most ε. $\square$

Theorem 2.6.2 *Let a, k, $\varepsilon(a)$ be as defined in Theorem 2.3.6 or 2.3.9. The graph G_k^d is pairwise $\langle \rho(a), \varepsilon(a) \rangle$-connected.*

Proof: Let A and B be subsets (in G_k^d) of size $\rho(a)n$. The corresponding subsets A_c and B_c of G_k^c are also of measure $\rho(a)$ (Lemma 2.6.1). By Corollary 2.3.13, $E(A_c, B_c) \geq \varepsilon(a)$. We conclude that $E(A,B) \geq E(A_c, B_c)n^2 \geq \varepsilon(a)n^2$. $\square$

Lemma 2.6.3 *The graph G_k^d is vector $k\left(1 + \frac{ck^2}{2^d}\right)$-colorable for some constant $c > 0$.*

Proof: Recall that each cell in G_k^d has diameter at most 2^{-d}. Hence, two vertices in G_k^d are connected only if their inner product is less than $-1/(k-1) + \theta(1)2^{-d} \leq -\frac{1}{k\left(1+\frac{\theta(k)}{2^d}\right)-1}$. $\square$

By definition, the continuous graph G_k is vector k-colorable. In Lemma 2.6.3 we show that the finite approximation G_k^d to G_k is *almost* vector k-colorable. In general, this does will suffice for the proof of Theorem 2.1.1 as we are interested in graphs which are vector k-colorable (rather than "almost vector k-colorable"). This can be fixed by starting with a continuous graph with vector coloring number slightly less than k (e.g. $k/\left(1 + \frac{ck^2}{2^d}\right)$). In order to simplify our presentation, we neglect this point and consider the graph G_k^d to be exactly vector k-colorable. This is possible due to the fact that the properties of G_k are *continuous* in k. Namely, choosing d large enough, it can be seen that the multiplicative error of $(1 + \frac{ck^2}{2^d})$ in the value of k does not effect the analysis appearing throughout this section.

Lemma 2.6.4 *Let $\rho(\frac{1}{k-1})$ be the measure of a $\frac{1}{(k-1)}$-cap. Every vertex v in the graph G_k^d has degree $d_v \in [\frac{1}{poly(d)}\rho(\frac{1}{k-1})n, poly(d)\rho(\frac{1}{k-1})n]$.*

Proof: Consider a vertex v in G_k^d and its corresponding cell C_v. The degree of v is the number of cells in G_k which share a positive measure of edges with the cell C_v. The total measure of these cells is at least the measure of a $\left(\frac{1}{k-1} + \theta(2^{-d})\right)$-cap and at most the measure of a $\left(\frac{1}{k-1} - \theta(2^{-d})\right)$-cap. Hence, by Lemma 2.6.1 we conclude our theorem. $\square$

We now prove the first part of Theorem 2.1.1 by considering the graph H obtained by randomly sampling the edges of G_k^d.

Theorem 2.6.5 *For every constant $\gamma > 0$ and constant $k > 2$, there are infinitely many graphs H that are vector k-colorable and satisfy $\alpha(H) \leq \frac{n}{\Delta_H^{1-\frac{2}{k}-\gamma}}$, where n is the number of vertices in H and Δ_H is the maximum degree in H.*

Proof: Let $k > 2$ be constant. Let $\gamma > 0$ be an arbitrarily small constant. Let $a = \gamma/c$ for a sufficiently large constant that will be defined later in the proof. Let $G = G_k^d = (V, E)$ be the dense graph defined above. Let n be the size of the vertex set V of G, and let Δ be the maximum degree of G. Recall that $n = 2^{\theta(d^2)}$ where d is the dimension in which the corresponding graph G_k was defined. We will assume the dimension d is a very large constant determined after fixing a. Finally, let $\rho = \rho(a)$.

Recall (Lemma 2.6.4 and Claim 2.4.1) that all vertices in G have degree which is in the range $[\frac{1}{poly(d)}\Delta, \Delta]$ where $\Delta \sim \rho(\frac{1}{k-1})n$, and that every subset of vertices U in G of size ρn has has at least $\Delta\rho^{\frac{2(k-1)}{k-2}+\gamma}n$ edges. Let $p = \frac{1}{\Delta\rho^{\frac{k}{k-2}+2\gamma}}$. Let H be the subgraph of G obtained by deleting each edge of G independently with probability $(1-p)$.

Lemma 2.6.6 *With probability $\geq 3/4$ all vertices v of H will have degree $d_v(H)$ in the range*

$$[\frac{1}{poly(d)}\rho^{-\frac{k}{k-2}-2\gamma}, 2\rho^{-\frac{k}{k-2}-2\gamma}].$$

Proof: The expected degree $d_v(H)$ of each vertex v in H satisfies

$$d_v(H) \in \left[\frac{1}{poly(d)}\rho^{-\frac{k}{k-2}-2\gamma}, \rho^{-\frac{k}{k-2}-2\gamma}\right].$$

It is not hard to verify (using standard bounds) that with probability $\geq 3/4$ it is the case that all vertices v have degree $d_v(H)$ which does not deviate from their expectation by more than a constant fraction of their expectation. $\square$

Lemma 2.6.7 *With probability $\geq 3/4$ the size of the maximum independent set in H ($\alpha(H)$) is at most ρn.*

Proof: It suffices to show that every subset U of H of size ρn has at least a single edge. Using Claim 2.9.2, it follows that for each subset U of H, the probability that all its edges were removed is at most

$$(1-p)^{\Delta\rho^{\frac{2(k-1)}{k-2}+\gamma}n} \leq e^{-\rho^{1-\gamma}n}.$$

The number of subsets U of size ρn is

$$\binom{n}{\rho n} \leq e^{\rho n \ln n} \leq \frac{1}{4}e^{\rho^{1-\gamma}n}$$

Applying the union bound on all subsets U of H of size ρn we conclude our assertion $\square$

Now with probability at least $1/2$ both Lemma 2.6.6 and Lemma 2.6.7 hold implying our theorem. $\square$

2.7 Lower bounds for the testing of $\alpha(G)$

In this section we present graphs G which are $\langle\rho,\varepsilon\rangle$-connected, but with some constant probability, a random sample R of G of size $s \sim \rho^3/\varepsilon^2$ is likely to have an independent set of size greater than ρs.

Lemma 2.7.1 *Let ρ be a small constant and $\varepsilon < \rho^2$ s.t. $\rho^3/\varepsilon^2 << n$. For n large enough, there exists a graph G on n vertices for which (a) G is $\langle\rho,\varepsilon\rangle$connected, (b) With constant probability (independent of ρ and ε) a random set R of size $s = \frac{\rho^3}{\varepsilon^2}$ will have an independent set of size ρs.*

Proof: Consider the graph $G = (V, E)$ in which $|V| = n$, V consists of two disjoint sets A and $V \setminus A$, where A is an independent set of size $(1 - \frac{\varepsilon}{\rho^2})\rho n$, $V \setminus A$ induces a clique, and every vertex in A is adjacent to every vertex in $V \setminus A$. On one hand, every subset of size ρn in G induces a subgraph with at least $\varepsilon n^2/2$ edges (implying that G is $\langle\rho,\varepsilon\rangle$connected). On the other, let R be a random subset of V obtained by picking each vertex independently with probability $\frac{\rho^3}{\varepsilon^2 n}$. The expected size of R is $s = \frac{\rho^3}{\varepsilon^2}$. In the following we assume R is exactly of size s, minor modifications in the proof are needed if this assumption is not made. The set $R \cap A$ is an independent set in the subgraph induced by R. The expected size of $R \cap A$ is $(1 - \frac{\varepsilon}{\rho^2})\rho s$. Let $N(0, 1)$ denote a standard normal variable. It can be seen using the central limit theorem (for example [Fel66]) that for our choice of parameters, the probability that $|R \cap A|$ deviates from its expectation by more than a square root of its expectation is at least

$$\Pr\left[|R \cap A| > \left(1 - \frac{\varepsilon}{\rho^2}\right)\rho s + \sqrt{\rho s}\right] > \Pr\left[N(0, 1) > 1\right]$$

which is some constant probability independent of ε and ρ. In such a case the size of $R \cap A$ will be greater than $(1 - \frac{\varepsilon}{\rho^2})\rho s + \sqrt{\rho s} = \rho s$ for our value of s. Hence implying assertion (b) of the lemma. $\square$

2.8 Discussion

In this chapter we have presented *tight* bounds on the chromatic number of vector k-colorable graphs. Tight in the sense that they match the upper bounds presented in [KMS98]. Many questions still remain open.

Stronger coloring relaxations As mentioned in the Introduction and Section 2.5, there are stronger relaxations to the minimum coloring problem that also have a geometrical interpretation. For example, one such relaxation is the well known (and extensively studied) Lovász theta function [Lov79]. It is not hard to verify that these relaxations can be used *as is* in the coloring algorithm presented in [KMS98]. Hence, it is only natural to expect that using such stronger relaxations will yield improved coloring results. At the moment this is not known to be true.

One may consider proving that even the use of such stronger relaxations in the [KMS98] algorithm cannot yield stronger coloring results. Or in other words, one may try to extend our negative results to stronger relaxations as well. A few remarks are in place. It seems as if the techniques we use in this chapter do not extend to the stronger coloring relaxations presented in Section 2.5. Our graphs (continuous and random), or to be precise their embeddings are not valid with respect to these stronger coloring relaxations. Therefore, in order to extend our negative results, one must change these embeddings appropriately without increasing their vector chromatic number. It seems as if changing the embedding of our graphs as to satisfy these stronger coloring relaxations has a large effect on their vector chromatic number. This implies that such an approach will yield weaker negative results.

Secondly, one may consider using our proof techniques on graphs other than the ones presented in this chapter. One natural candidate is the graph $G = (V, E)$ in which the vertex set V consists of the set $\{0, 1\}^n$ and two vertices are connected by an edge if their Hamming distance is equal to some prespecified value. The graph G and certain subgraphs of G have been used in the past in the context under discussion (*e.g* [KMS98, Fei97, GK98, Cha02]). Using our proof technique on such graphs involves the analysis of certain edge isoperimetric inequalities (analogous to those presented in Theorem 2.3.5). Unfortunately, little is known regarding the edge isoperimetric inequalities of the above graphs G. Such inequalities have been studies in the past ([Bez02, KKL88]) yielding partial results. However, these results do not suffice to extend our proof techniques. A better understanding of edge isoperimetric inequalities of these graphs is of great interest regardless their application to the vector coloring issue.

Property testing In Section 2.2 and Section 2.7 we study the property testing paradigm with respect to the independent set problem. We present improved results on the sample size needed when testing graphs which are *far* from having large independent sets. Namely, in Theorem 2.2.4 we prove that if a graph G of size n is $\langle\rho, \varepsilon\rangle$-connected then w.h.p. a random induced subgraph of G of size $s \sim 1/\varepsilon^3$ will not have an independent set of size ρs (this improves upon the sample size of $s \sim 1/\varepsilon^4$ presented in [GGR98]). Moreover, in Lemma 2.7.1 we show that the sample size s must be of size at least $\sim 1/\varepsilon^2$ if we wish the probability of success to be nonconstant. It would be interesting if the factor ε gap between the upper and lower bounds presented above could be settled.

2.9 Proof of claims

Claim 2.9.1 *For $1 \geq p \geq 0$ and $y \geq 1$, we have that $(1-p)^y \geq 1 - py \geq (1-p)^{y + \frac{py^2}{1-py}}$.*

Proof: It is well known that $e^y \geq 1 + y$ for any real y. Hence,

$$1 - py = \frac{1}{1 + py + \frac{p^2y^2}{1-py}} \geq e^{-p(y + \frac{py^2}{1-py})} \geq (1-p)^{y + \frac{py^2}{1-py}}.$$

As for the other direction, for every $y \geq 1$

$$\frac{d}{dp}((1-p)^y - 1 + py) = y(1 - (1-p)^{y-1}) \geq 0.$$

□

Claim 2.9.2 *For all $x > 1$*

$$\left(1 - \frac{1}{x}\right)\frac{1}{e} \leq \left(1 - \frac{1}{x}\right)^x \leq \frac{1}{e}$$

Proof: It is known that for all $x > 1$

$$\left(1 - \frac{1}{x}\right)^x \leq \frac{1}{e} \leq \left(1 - \frac{1}{x}\right)^{x-1}$$

□

Claim 2.9.3 *Let $a > \delta > 0$ such that $a\delta \geq 2\log(d)/d$ then*

$$\rho(a) - \rho(a+\delta) \geq (1 - 1/d)\rho(a).$$

Proof: Recall that $\frac{c}{\sqrt{d}}\left(1-a^2\right)^{\frac{d-1}{2}} \leq \rho(a) \leq \left(1-a^2\right)^{\frac{d-1}{2}}$, for some constant $c > 0$. Thus,

$$\rho(a+\delta) \leq \Big(1 - a^2 - \delta^2 - 2a\delta\Big)^{\frac{d-1}{2}} \leq \Big((1-a^2)(1-2a\delta)\Big)^{\frac{d-1}{2}} \leq \Big(1-a^2\Big)^{\frac{d-1}{2}} \frac{2}{d^2} \leq \rho(a)/d$$

□

Theorem 2.3.5 *Let $1 \geq a > 0$ and let A and B be two (not necessarily disjoint) measurable sets in V of measure $\rho(a)$. Let r be an arbitrary vertex of S^{d-1}. The minimum of $E(A, B)$ is obtained when $A = B = \mathcal{C}_a$ where $\mathcal{C}_a$ is an a-cap of measure $\rho(a)$ centered at r.*

Proof: Our proof is similar to that presented in [FS02]. We start by reviewing the *two point symmetrization* procedure introduced in [BT76]. Let $r \in S^{d-1}$ and H be the hyperplane passing through the origin with normal r. The hyperplane H partitions S^{d-1} into three sets: points *above*, *below*, and *on* the hyperplane. Define S_0 to be the set $\{u \in S^{d-1} \mid \langle u, r\rangle = 0\}$, S^+ to be $\{u \in S^{d-1} \mid \langle u, r\rangle > 0\}$, and S^- to be $\{u \in S^{d-1} \mid \langle u, r\rangle < 0\}$. For any $x \in S^{d-1}$ denote its reflection with respect to H as $\sigma(x)$. Finally, given any (measurable) subset X of S^{d-1}, we define the *symmetrization* X^* of X w.r.t. H as follows. If x is a point in $X \cap S^-$ such that $\sigma(x) \notin X$ replace x with $\sigma(x)$. All other points $x \in X$ remain in X^*, and no other new points are added to X^*. Formally

$$X^* = X \setminus \{x \mid x \in X \cap S^-,\ \sigma(x) \notin X\} \cup \{\sigma(x) \mid x \in X \cap S^-,\ \sigma(x) \notin X\}$$

Notice that the measure of X and X^* are identical.

Now, let Λ be the set consisting of all pairs of closed subsets in S^{d-1}. Given two closed subsets A and B of measure $\rho(a)$, let $\lambda(A, B) = \lambda \subset \Lambda$ be the set of pairs $(\alpha, \beta) \in \Lambda$ that satisfy:

1. $\mu(A) = \mu(\alpha)$, $\mu(B) = \mu(\beta)$.
2. $\forall \varepsilon > 0 \quad \mu(A_\varepsilon) \geq \mu(\alpha_\varepsilon)$, $\mu(B_\varepsilon) \geq \mu(\beta_\varepsilon)$.
3. $E(A, B) \geq E(\alpha, \beta)$.

where for any set A, the set A_ε is defined as $\{x \in S^{d-1} \mid \exists y \in A \ \ s.t. \ \ \|x - y\| \leq \varepsilon\}$.

We start by showing that λ is closed under the two point symmetrization technique. Next we prove that λ is a closed subset of Λ (under the Hausdorff topology). Afterwards, we show that there exists a cap C of measure $\rho(a)$ such that $(C, C) \in \lambda = \lambda(A, B)$, implying that $E(A, B) \geq E(C, C)$. This concludes our proof for subsets A and B which are closed. Finally, we turn to the general case.

Lemma 2.9.4 *The set λ is closed under the two point symmetrization procedure (w.r.t. any $r \in S^{d-1}$ and the hyperplane H passing through the origin defined by r).*

Proof: Let H be some hyperplane passing through the origin. Let (α, β) be a pair in λ. Let α^* and β^* be the sets obtained after the symmetrization procedure w.r.t. H applied to the sets α and β. It suffices to show that $(\alpha^*, \beta^*) \in \lambda$. In [Ben84] it is shown that both α^* and β^* preserve the first two properties of λ. Hence, we turn to the third property. We need to show that the subsets α^* and β^* obtained after the symmetrization procedure share fewer edges than the original sets α and β (which in turn share fewer edges than A and B). Let x^+ and y^+ be two points in S^+, and let x^- and y^- be their corresponding mirror images. Let α' be the restriction of α to these four points (that is $\alpha' = \alpha \cap \{x^+, x^-, y^+, y^-\}$). Define β', $\alpha^{*'}$ and $\beta^{*'}$ similarly. We will show that $E(\alpha', \beta') \geq E(\alpha^{*'}, \beta^{*'})$. It is not hard to verify that this implies our assertion.

Let E' be the edge set induced by the vertices $\{x^+, x^-, y^+, y^-\}$. Recall by the definition of our continuous graph G^c, the edge set E consists of pairs of points x and y in S^{d-1} which satisfy $\langle x, y\rangle \leq -\frac{1}{k-1}$. This implies that E' is symmetric and has the following properties:

1. $(x^+, y^+) \in E' \leftrightarrow (x^-, y^-) \in E'$.
2. $(x^+, y^-) \in E' \leftrightarrow (x^-, y^+) \in E'$.
3. $(x^+, y^+) \in E' \rightarrow (x^+, y^-) \in E'$.

Consider the case in which E' consists of the edges (x^+, y^-) and (x^-, y^+) only, the subset α' is equal to $\{x^+, x^-\}$ and the subset β' is equal to $\{y^-\}$. This case in depicted in Figure 2.2 (a). In this specific case we have that $\alpha^{*'} = \alpha'$, and $\beta^{*'} = \{y^+\}$, implying that $E(\alpha', \beta') = E(\alpha^{*'}, \beta^{*'}) = 1$. Notice that as α' and β' are finite sets, we measure the amount of edges between α' and β' using the discrete analog of $E(\alpha', \beta')$ defined in Definition 2.2.1 of Section 2.3 (the same goes for $\alpha^{*'}$ and $\beta^{*'}$).

There are of course several other cases to consider (12 edge configurations and 256 cases of different subsets α' and β'). Some of these cases are depicted in Figure 2.2. This large case analysis may be significantly reduced using various observations (involving the equivalence of many different cases). We have checked our assertion on the full case analysis (using a computer program). A similar proof holds when x^+ or y^+ are in S_0. Details omitted. □

Lemma 2.9.5 *λ is closed in Λ.*

Proof: Let (α_n, β_n) be a sequence in λ tending to (α, β) (in the Hausdorff topology). We will show that $(\alpha, \beta) \in \lambda$. For every $\varepsilon, \delta > 0$ we have for large enough values of n that

$$\alpha_\varepsilon \subseteq (\alpha_n)_{\varepsilon+\delta}, \quad \beta_\varepsilon \subseteq (\beta_n)_{\varepsilon+\delta}; \quad \mu((\alpha_n)_{\varepsilon+\delta}) \leq \mu(A_{\varepsilon+\delta}), \quad \mu((\beta_n)_{\varepsilon+\delta}) \leq \mu(B_{\varepsilon+\delta}).$$

Sending δ to zero and using the fact that for all closed sets α, $\mu(\alpha_\varepsilon) \rightarrow \mu(\alpha)$ we conclude the second property of λ. Namely that $\mu(\alpha_\varepsilon) \leq \mu(A_\varepsilon)$ and $\mu(\beta_\varepsilon) \leq \mu(B_\varepsilon)$.

For the first property, sending ε to zero we have on one hand that $\mu(\alpha) \leq \mu(A)$ and $\mu(\beta) \leq \mu(B)$. For the other direction observe that $\mu(\alpha_\varepsilon) \geq \mu(\alpha_n) = \mu(A)$ and $\mu(\beta_\varepsilon) \geq \mu(\beta_n) = \mu(B)$ for any $\varepsilon > 0$ provided n is large enough.

For the final property of λ, let $\theta(\varepsilon) = \mu^2(\{(x, y) \mid x \in \alpha_\varepsilon,\ y \in \beta_\varepsilon,\ (x, y) \in E\})$. Notice that $E(\alpha, \beta) \leq \theta(\varepsilon)$. Let n be large enough to ensure that $\alpha_\varepsilon \subseteq (\alpha_n)_{2\varepsilon}$, $\beta_\varepsilon \subseteq (\beta_n)_{2\varepsilon}$. For such n it is the case that $\theta(\varepsilon) \leq E((\alpha_n)_{2\varepsilon}, (\beta_n)_{2\varepsilon})$. Furthermore, as $(\alpha_n, \beta_n) \in \lambda$ for all values of n we have that $\mu((\alpha_n)_{2\varepsilon} \setminus \alpha_n) \leq \mu(A_{2\varepsilon} \setminus A)$ and $\mu((\beta_n)_{2\varepsilon} \setminus \beta_n) \leq \mu(B_{2\varepsilon} \setminus B)$. Which both tend to zero as ε tends to zero. We conclude that $E((\alpha_n)_{2\varepsilon}, (\beta_n)_{2\varepsilon})$ tends to $E(\alpha_n, \beta_n)$ as ε tends to zero *uniformly* in n.

Hence, given $\delta > 0$ one can find $\varepsilon > 0$ such that $E((\alpha_n)_{2\varepsilon}, (\beta_n)_{2\varepsilon}) \leq E(\alpha_n, \beta_n) + \delta$ for all values of n. We conclude, that for n large enough

$$E(\alpha, \beta) \leq \theta(\varepsilon) \leq E((\alpha_n)_{2\varepsilon}, (\beta_n)_{2\varepsilon}) \leq E(\alpha_n, \beta_n) + \delta \leq E(A, B) + \delta.$$

Since this holds for all δ we conclude our assertion. □

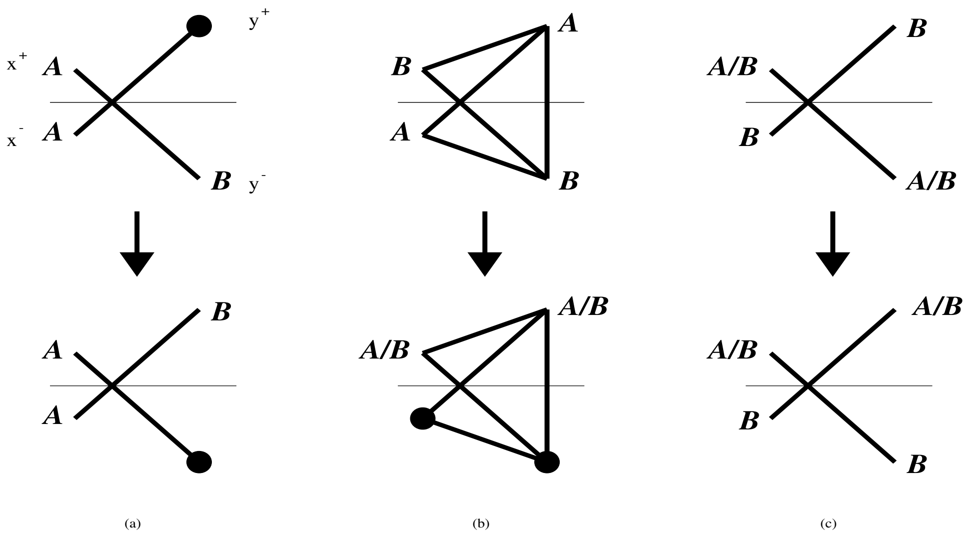

Figure 2.2: Three cases of E', α' and β' are presented. The set $\alpha' = \alpha \cap \{x^+, x^-, y^+, y^-\}$ and $\alpha^{*'}$ are presented by the letter A. The set $\beta' = \beta \cap \{x^+, x^-, y^+, y^-\}$ and $\beta^{*'}$ are presented by the letter B. A point in $\alpha' \cap \beta'$ ($\alpha^{*'} \cap \beta^{*'}$) is presented by A/B. A point neither in α or β is represented by a solid dot. The edge set E' is depicted by solid lines. Finally, the hyperplane H is represented as a horizontal line. Each configuration is presented before (above) and after (below) the symmetrization procedure. In case (a) the set α' is closed under the symmetrization procedure w.r.t H and $-H$. In such cases it holds that $E(\alpha', \beta') = E(\alpha^{*'}, \beta^{*'})$. In cases (b) and (c) notice that there is a difference between the value of $E(\alpha^{*'}, \beta^{*'})$ and the number of undirected edges between the sets $\alpha^{*'}$ and $\beta^{*'}$ (for example in case (b) $E(\alpha^{*'}, \beta^{*'}) = \{(x^+, y^+), (y^+, x^+)\}$ is of size 2).

Lemma 2.9.6 *There exists a cap C of measure $\rho(a)$ such that $(C, C) \in \lambda$.*

Proof: Fix a point $x_0 \in S^{d-1}$ and let C be the cap centered at x_0 of measure $\rho(a)$. The function $f_1(\alpha, \beta) = \mu(C \cap \alpha)$ is upper semi continuous on Λ [3]. Λ with the Hausdorff topology is a compact set, since λ is closed (Lemma 2.9.5) the function f_1 achieves its maximum over λ on some pair $(\alpha_{max}, \beta_{max}) \in \lambda$. In [FS02] it is shown (using Lemma 2.9.4) that $\alpha_{max} = C$. This is done by showing that $f_1(\alpha^*_{max}, \beta) \geq f_1(\alpha_{max}, \beta)$ with equality iff $\alpha_{max} = C$ where α^*_{max} is the subset obtained by the symmetrization procedure with respect to a hyperplane H s.t. $x_0 \notin H$; and β is arbitrary.

Now consider the function $f_2(C, \beta) = \mu(C \cap \beta)$ defined on $\Lambda' = \{(C, \beta) \mid (C, \beta) \in \Lambda\}$. The function f_2 is also upper semi continuous (this time on Λ'). Furthermore, it is shown in [FS02] that Λ' with the Hausdorff topology is a compact set and that $\lambda' = \{(C, \beta) \mid (C, \beta) \in \lambda\}$ is closed in Λ'. Therefore, the function f_2 achieves its maximum over λ' on some pair (C, β_{max}). As before it can be seen that $\beta_{max} = C$. We conclude that $(C, C) \in \lambda$. □

As mentioned previously, Lemma 2.9.6 implies our theorem for subsets A and B which are closed. The proof for general subsets A and B follows by considering any series $\{A_i\}$ and $\{B_i\}$ of closed sets that satisfy: (a) $\forall n \;\; A_n \subseteq A$ and $B_n \subseteq B$, (b) $\mu(A_n) \to \mu(A)$ and $\mu(B_n) \to \mu(B)$, and (c) $\forall n \;\; \mu(A_n) = \mu(B_n)$. Now let C, $\{C_i\}$ be caps, all centered at the same point $x \in S^{d-1}$ such that

[3] $f(\alpha)$ is upper semi continuous on Λ if for every sequence $\alpha_n \to \alpha$ in Λ it is the case that $f(\alpha)$ is at least $\limsup f(\alpha_n)$.

$\mu(C) = \mu(A) = \mu(B)$ and $\forall n \;\; \mu(C_n) = \mu(A_n) = \mu(B_n)$. It now holds that

$$E(A, B) \geq E(A_n, B_n) \geq E(C_n, C_n) \to E(C, C)$$

$\square$

Claim 2.3.8 *Let a, b be constant, let $z^2 = a^2 + b^2$, and let δ be as in Claim 2.3.7. Let $N = \{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 = a, \; u_2 = b\}$. Let N_δ be a δ neighborhood of N (*i.e.* all points which are of distance less that δ from the set N). The measure of the set N_δ is at least $(1 - z^2)^{\frac{d-1}{2}}$.*

Proof: Recall that $z = \sqrt{a^2 + b^2}$. Let $\hat{N} = \{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 = z, \; u_2 = 0\}$ be the set obtained from N by rotating the unit sphere by an angle of $\arccos \frac{a}{z}$. Using spherical symmetry, we have that the measure of the set N_δ is equal to the measure of the δ neighborhood of the set $\hat{N}$ (which will be denoted as $\hat{N}_\delta$). Hence, in the following we will present a lower bound on the measure of $\hat{N}_\delta$. We start by computing the measure of the strip

$$S = \{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 \in [z - \delta^2/1000, z + \delta^2/1000]\}.$$

Using Claims 2.9.2 and 2.9.3 we conclude (for sufficiently large d)

$$\mu(S) = \rho(z - \delta^2/1000) - \rho(z + \delta^2/1000) \geq \frac{1}{2}\rho(z - \delta^2/1000) \geq \frac{1}{\delta^3}\left(1 - z^2\right)^{\frac{d-1}{2}}.$$

Let γ be a constant such that $\sqrt{1 - z^2} \geq |\gamma| > \delta/10$. Consider the set $R = \{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 = z, \; u_2 = \gamma\}$. Denote the $\delta^2/200$ neighborhood of R as R_γ. Using spherical symmetry, one can upper bound the measure of the set R_γ by the measure of the cap $\{(u_1, \ldots, u_d) \in S^{d-1} \mid u_1 > \sqrt{z^2 + \gamma^2} - \delta^2/200\}$ which is at most

$$\left(1 - \left(\sqrt{z^2 + \gamma^2} - \delta^2/200\right)^2\right)^{\frac{d-1}{2}} \leq \left(1 - z^2\right)^{\frac{d-1}{2}} \leq \delta^3 \mu(s).$$

Let $\mathcal{R}$ be the union of the sets R_γ for $|\gamma| = \delta/10, \delta/10 + \delta^2/1000, \delta/2 + 2\delta^2/1000, \ldots, \sqrt{1 - z^2}$. We conclude that the measure of $\mathcal{R}$ is at most $\frac{1}{2}\mu(S)$.

Finally, each $v = (v_1, \ldots, v_d) \in S \setminus \mathcal{R}$ satisfies $v_1 \in [z - \delta^2/1000, z + \delta^2/1000]$ and $v_2 \in [-\delta/10, \delta/10]$ and thus is in $\hat{N}_\delta$. Hence $S \setminus \mathcal{R} \subseteq \hat{N}_\delta$, implying that the measure of $\hat{N}_\delta$ is at least $\frac{1}{2}\mu(S) \geq (1 - z^2)^{\frac{d-1}{2}}$. $\square$

Claim 2.3.7 *Let a, k be as in Theorem 2.3.6. Let $v = (a, \sqrt{1 - a^2}, 0, \ldots, 0)$ be a vertex on the boundary of $\mathcal{C}_a$. Let $N(v)$ be the set of neighbors of v. Let $z = \sqrt{a^2 + \frac{(1/(k-1) + a^2)^2}{1 - a^2}}$. Finally let $\delta = c\sqrt{\frac{\log(d)}{d}}$ for a sufficiently large constant c. The measure of vertices in $N(v) \cap \mathcal{C}_a$ satisfies*

$$(1 - \delta)^{\frac{d-1}{2}} \left(1 - z^2\right)^{\frac{d-1}{2}} \leq \mu(N(v) \cap \mathcal{C}_a) \leq \left(1 - z^2\right)^{\frac{d-1}{2}}$$

Proof: The set $N(v)\cap\mathcal{C}_a$ is equal to $\left\{(u_1,\ldots,u_d)\in S^{d-1}\mid u_1\geq a \text{ and } u_2\leq -\frac{1/(k-1)+au_1}{\sqrt{1-a^2}}\right\}$. Denote $\frac{1/(k-1)+a^2}{\sqrt{1-a^2}}$ by $f(a)$. Let $z=\sqrt{a^2+f(a)^2}$. By spherical symmetry, it can be seen that the above set is of measure at most $\left(1-z^2\right)^{\frac{d-1}{2}}$, and of measure greater or equal to the measure of $\Gamma=\{(u_1,\ldots,u_d)\in S^{d-1}\mid u_1\geq z \text{ and } 0\leq u_2\leq c(u_1-z)\}$, where $1\geq c\geq \tan\left(\frac{\arccos(a)}{2}\right)$ (for any $a\in[0,1/2]$, c is in the range $(1/2,1)$). Hence, it suffices to bound the measure of Γ by below.

Let $\delta=c\sqrt{\frac{\log(d)}{d}}$ for a sufficiently large constant c. The set Γ above is of measure larger than the measure of the δ neighborhood of

$$N=\{(u_1,\ldots,u_d)\in S^{d-1}\mid u_1=z+\delta\left(1+\frac{2}{c}\right) \text{ and } u_2=\delta\}$$

for c as above. We denote this set as N_δ. By Claim 2.3.8, the measure of N_δ is at least of value $\left(1-z^2-\theta(\delta)\right)^{\frac{d-1}{2}}$ (notice that $z>1/k$ which in turn is independent of d, thus Claim 2.3.8 can be applied in our case). Hence, the measure of $N(v)\cap\mathcal{C}_a$ is at least $\left(1-\theta(\delta)\right)^{\frac{d-1}{2}}\left(1-z^2\right)^{\frac{d-1}{2}}$ which concludes our proof. $\square$

Theorem 2.3.9 *Let $a=\left(\frac{\log(d)}{d}\right)^{\frac{1}{4}}$. Let k satisfy $\frac{1}{k-1}=a^2$. Let $r\in S^{d-1}$, let $\mathcal{C}_a$ be an a-cap centered at r. Let $\varepsilon(a)$ be the value of $E(\mathcal{C}_a,\mathcal{C}_a)$. The value of $\varepsilon(a)$ is in the range*

$$\left[\frac{1}{poly(d)}\left(1-\frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}},\ \left(1-\frac{2(k-1)}{k-2}a^2\right)^{\frac{d-1}{2}}\right].$$

Proof: We start by stating the following claim.

Claim 2.9.7 *Let $\mathcal{C}_a$ be an a-cap centered at $r=(1,0,\ldots,0)$. Let $v=(a,\sqrt{1-a^2},0,\ldots,0)$ be a vertex on the boundary of $\mathcal{C}_a$. Let $N(v)$ be the set of neighbors of v. Let $z=\sqrt{a^2+\frac{(1/(k-1)+a^2)^2}{1-a^2}}$. If $z^4=O(\log(d)/d)$ then the measure of vertices in $N(v)$ which are in the cap $\mathcal{C}_a$ satisfies*

$$\frac{1}{poly(d)}\left(1-z^2\right)^{\frac{d-1}{2}}\leq\mu(N(v)\cap\mathcal{C}_a)\leq\left(1-z^2\right)^{\frac{d-1}{2}}$$

Proof: The set $N(v)\cap\mathcal{C}_a$ is equal to $\left\{(u_1,\ldots,u_d)\in S^{d-1}\mid u_1\geq a \text{ and } u_2\leq -\frac{1/(k-1)+au_1}{\sqrt{1-a^2}}\right\}$. Denote $\frac{1/(k-1)+a^2}{\sqrt{1-a^2}}$ by $f(a)$. Let $z=\sqrt{a^2+f(a)^2}$. By spherical symmetry, it can be seen that the above set is of measure at most $\left(1-z^2\right)^{\frac{d-1}{2}}$, and of measure greater or equal to the measure of $\Gamma=\{(u_1,\ldots,u_d)\in S^{d-1}\mid u_1\geq z \text{ and } 0\leq u_2\leq c(u_1-z)\}$, where $1\geq c\geq\tan\left(\frac{\arccos(a)}{2}\right)$. Hence, it suffices to bound the measure of Γ by below.

Let $r_1,\ldots,r_d$ be independent standard normal variables, let $s=\sqrt{\sum_{i=1}^d r_i^2}$, and let $s_3=\sqrt{\sum_{i=3}^d r_i^2}$. Let $\phi(x)$ be the standard normal density function. The measure of Γ is given by the probability

$$\mu(\Gamma)\quad=\quad\Pr\left[\frac{r_1}{s}\geq z \text{ and } 0\leq\frac{r_2}{s}\leq c(\frac{r_1}{s}-z)\right]$$

$$
\begin{aligned}
&\geq \Pr\left[r_1 \geq \sqrt{\frac{z^2(r_2^2+s_3^2)}{1-z^2}} \text{ and } 0 \leq r_2 \leq A\right] \\
&\geq \Pr[s_3^2 \leq d]\Pr\left[r_1 \geq \sqrt{\frac{z^2(r_2^2+d)}{1-z^2}} \text{ and } 0 \leq r_2 \leq A_d\right] \\
&\geq \frac{1}{4}\Pr\left[r_1 \geq \sqrt{\frac{z^2 d}{1-z^2}} \text{ and } 0 \leq r_2 \leq \min(A_d, B)\right] \\
&\geq \frac{1}{4}\Pr\left[\sqrt{\frac{z^2 d}{1-z^2}} \leq r_1 \leq \sqrt{\frac{z^2(d+1)}{1-z^2}} \text{ and } 0 \leq r_2 \leq \min(A_d, B)\right]
\end{aligned}
$$

Where $A = c\frac{r_1 - z\sqrt{r_1^2(1+c^2-c^2z^2)+s_3^2(1-c^2z^2)}}{1-c^2z^2}$, $A_d = c\frac{r_1 - z\sqrt{r_1^2(1+c^2-c^2z^2)+d(1-c^2z^2)}}{1-c^2z^2}$, and B is of value $\sqrt{\frac{r_1^2(1-z^2)}{z^2} - d}$. Notice that we have used the fact that $\Pr[s_3^2 \leq d] \geq 1/4$.

We proceed by evaluating $\min(A_d, B)$. Both A_d and B are functions of r_1. Furthermore r_1^2 is in the range $\left[\frac{z^2 d}{1-z^2}, \frac{z^2(d+1)}{1-z^2}\right]$. Let $x \in [0,1]$, we can represent A_d and B as functions of x by setting r_1 to be $\sqrt{\frac{z^2(d+x)}{1-z^2}}$. That is $A_d = z\frac{\sqrt{d+x}-\sqrt{d+xz^2(1+c^2-c^2z^2)}}{\sqrt{1-z^2}(1-c^2z^2)}$ and $B = \sqrt{x}$. It is not hard to verify that for our value of z it is the case that $\min(A_d, B) = A_d$.

$$
\begin{aligned}
\mu(\Gamma) &\geq \frac{1}{4}\Pr\left[\sqrt{\frac{z^2 d}{1-z^2}} \leq r_1 \leq \sqrt{\frac{z^2(d+1)}{1-z^2}} \text{ and } 0 \leq r_2 \leq A_d\right] \\
&= w\int_{r_1 \in R_1}\int_{r_2=0}^{A_d} e^{-\frac{r_1^2+r_2^2}{2}} \geq w\int_{r_1 \in R_1}\frac{e^{-\frac{r_1^2}{2}}}{A_d}\int_{r_2=0}^{A_d} r_2 e^{-\frac{r_2^2}{2}} \\
&= w\int_{r_1 \in R_1}\frac{e^{-\frac{r_1^2}{2}}}{A_d}(1-e^{-\frac{A_d^2}{2}}) \geq w\int_{r_1 \in R_1} A_d e^{-\frac{r_1^2}{2}} \\
&\geq w\int_{x=0}^{1} A_d e^{-\frac{z^2(d+x)}{2(1-z^2)}}\frac{z}{2\sqrt{1-z^2}\sqrt{d+x}}dx \\
&\geq w\frac{z^2}{d(1-z^2)(1-c^2z^2)}e^{-\frac{dz^2}{2(1-z^2)}}\int_{x=0}^{1} xe^{-\frac{z^2x}{2(1-z^2)}} \geq w\frac{z^2}{d(1-z^2)(1-c^2z^2)}e^{-\frac{dz^2}{2(1-z^2)}} \\
&\geq w\frac{z^2}{d(1-z^2)(1-c^2z^2)}\left(1-\frac{z^2}{1-z^2}\right)^{\frac{d-1}{2}} \\
&\geq w\frac{z^2}{d(1-z^2)(1-c^2z^2)}\left(1-z^2\right)^{\frac{d-1}{2}}\left(1-2z^4\right)^{\frac{d-1}{2}}
\end{aligned}
$$

Where R_1 is the range $\left[\sqrt{\frac{z^2 d}{1-z^2}}, \sqrt{\frac{z^2(d+1)}{1-z^2}}\right]$, and w is some constant independent of d (the value of w may change from line to line). Above we use the fact that $e^x > 1 + x$, and the fact that A_d is bounded by above by a constant independent of d. In the final inequalities we use a change of variables $r_1 = \sqrt{\frac{z^2(d+x)}{1-z^2}}$, and use the fact that $1+\alpha/3 \leq \sqrt{1+\alpha} \leq 1+\alpha/2$ for $\alpha < 3$. As $\mu(\Gamma) \geq \frac{1}{poly(d)}\left(1-z^2\right)^{\frac{d-1}{2}}$ for $z^4 = O(\log(d)/d)$, we conclude our assertion. □

For the proof of Theorem 2.3.9, let $z = \sqrt{a^2 + \frac{(1/(k-1)+a^2)^2}{1-a^2}}$ (as in Claim 2.9.7), thus $z^4 = O(\log(d)/d)$. For the upper bound, use a bound on the measure of $\mathcal{C}_a$, and the upper bound in Claim 2.9.7. As for the lower bound, let $\delta = 2\left(\frac{\log(d)}{d}\right)^{\frac{3}{4}}$. Such a choice of δ will satisfy $a\delta \geq 2\frac{\log(d)}{d}$, and $z\delta = O(\log(d)/d)$.

Let $w = (w_1, w_2, \ldots, w_d) \in \mathcal{C}_a$ with first coordinate w_1 of value $a+\delta$. Consider a vertex $v = (v_1, v_2, \ldots, v_d) \in \mathcal{C}_a$ with first coordinate v_1 of value less than $a+\delta$. By Claim 2.9.7, it is not hard to verify that the measure of $N(v) \cap \mathcal{C}_a$ is greater than the measure of $N(w) \cap \mathcal{C}_a$ which is greater than $\frac{1}{poly(d)}\left(1-cz\delta)\right)^{\frac{d-1}{2}}\left(1-z^2\right)^{\frac{d-1}{2}} \geq \frac{1}{poly(d)}\left(1-z^2\right)^{\frac{d-1}{2}}$ for some constant c (we use Claim 2.9.2 and the fact that $z\delta = O(\log(d)/d)$). As $a\delta \geq 2\log(d)/d$, we conclude using Corollary 2.9.3, that the measure of edges in $E(\mathcal{C}_a, \mathcal{C}_a)$ is at least

$$(\rho(a) - \rho(a+\delta))\frac{1}{poly(d)}\left(1-z^2\right)^{\frac{d-1}{2}} \geq \frac{\rho(a)}{poly(d)}\left(1-z^2\right)^{\frac{d-1}{2}} \geq \frac{1}{poly(d)}\left(1-a^2\right)^{\frac{d-1}{2}}\left(1-z^2\right)^{\frac{d-1}{2}}.$$

Simplifying the above expression with the help of Claim 2.9.2, we conclude our assertion. □

Part II

Graph partitioning

Chapter 3

The RPR^2 rounding technique

3.1 Introduction

For NP-hard maximization problems, we are interested in polynomial time approximation algorithms that for every instance produce solutions whose value is guaranteed to be within a ratio of at least α from the value of the optimal solution. The parameter $0 \leq \alpha \leq 1$ is known as the *approximation ratio* of the algorithm, and the larger α is, the better.

A common method for obtaining an approximation algorithm for a combinatorial optimization problem is based on linear programming:

1. Formulate the problem as an integer linear program.

2. Relax the problem to a linear program.

3. Solve the relaxation in polynomial time, obtaining a fractional solution $x_1, \ldots, x_n$.

4. Round the fractional solution to a 0/1 solution.

There are several approaches of how to round a fractional solution $x_1, \ldots, x_n$, and the approach to choose depends on the problem. Two common approaches are: *(a)* **Threshold rounding** in which a threshold t is set and each variable x_i is rounded to 1 if $x_i \geq t$, and to 0 otherwise (this approach is used for example on the Vertex Cover problem in [Hoc82]). *(b)* **Randomized rounding** in which a (monotone) *rounding function* $f : R \rightarrow [0, 1]$ is chosen. Each variable x_i is rounded independently to 1 with probability $f(x_i)$ and to 0 otherwise (randomized rounding was introduced in [RT87], and used for example in the approximation of the Max-SAT problem [GW94]). When the rounding function f is a threshold function (0 below the threshold and 1 above it), we get threshold rounding as a special case of randomized rounding.

Goemans and Williamson [GW95] successfully extended this approach to semidefinite programming. They use a *random hyperplane* rounding technique. In our presentation below we break this rounding technique into two steps (steps 4 and 5).

1. Formulate the problem as an integer quadratic program.

2. Relax the problem to a semidefinite program.

3. Solve the relaxation in polynomial time, obtaining a vector solution $v_1, \ldots, v_n$.

4. Project the vector solution on a random line through the origin, obtaining a fractional solution $x_1, \ldots, x_n$. The value x_i is defined to be the (directed) distance of the projection of the vector v_i form the origin.

5. Round the fractional solution $x_1, \ldots, x_n$ to a 0/1 solution using threshold rounding. (The threshold chosen by Goemans and Williamson is 0, rounding vectors with positive projection to 1 and negative projection to 0.)

Hence both the linear programming approach and the semidefinite programming approach eventually round a fractional solution to a 0/1 solution. The main difference in this respect is that for semidefinite programming threshold rounding has always been used at this stage, whereas for linear programming it has often been the case that randomized rounding is preferred.

In this chapter we study the use of a randomized rounding procedure instead of threshold rounding for the last step of the semidefinite programming approach. We call this rounding technique RPR^2 (*random projection, randomized rounding*) for short. The main contribution of this work is in adding RPR^2 to the "tool kit" of rounding techniques for semidefinite programs. To achieve this, we do several things:

1. We show how the tool can be used. This includes methods for choosing the rounding function f, and methods for analyzing (or lower bounding) the resulting approximation ratio.

2. We identify classes of optimization problems for which RPR^2 has potential of improving the known approximation ratios (or possibly even achieving approximation ratios that match the integrality gap of the semidefinite program).

3. We illustrate the usefulness of RPR^2 by improving the known approximation ratios for some of these problems.

We now go on to discuss the types of problems for which RPR^2 may be useful. For simplicity and concreteness, we shall concentrate on variants of the Max-Cut problem. Given a graph $G = (V, E)$ the Max-Cut problem is the problem of finding a maximum cut of G (*i.e.* a partition of the vertex set V into two sets $(U, V \setminus U)$ that maximizes the number of edges with one end point in U and the other in $V \setminus U$). Goemans and Williamson [GW95] use the method described above (with semidefinite programming and threshold rounding) to obtain a partition $(U, V \setminus U)$ of value at least $\alpha \simeq 0.87856$ times the value of the maximum cut in G. Recently in [FS02] it was shown that this approximation ratio matches the integrality ratio (for the particular semidefinite program used by Goemans and Williamson), and hence we shall not try to improve it (at least not here). Instead we shall consider special cases of Max-Cut for which threshold rounding (that is, the random hyperplane) produces a solution that is clearly not optimal. A simple sufficient condition for a solution not to be optimal is that of local optimality. Call a vertex *misplaced* if most of its neighbors lie in the same side of the cut. A solution is *locally optimal* if it does not have misplaced vertices. Clearly, a solution that is not locally optimal can be improved by having misplaced vertices change sides. For some instances of Max-Cut, it can be shown that if the approximation ratio of the Goemans and Williamson algorithm is indeed as bad as $\alpha \simeq 0.87856$, then necessarily the solution produced has (a substantial number of) misplaced vertices. Hence the approximation ratio can be improved by adding a local correction step to the algorithm that moves vertices from side to side until the solution becomes locally optimal. The questions that remain are how to best guide the local correction step in its choices, and how to analyze the effect of the local correction step. In some cases, RPR^2 answers both questions simultaneously.

Consider *light* Max-Cut, the problem of Max-Cut on instances where the maximum cut in a graph is not very large (below a 0.844 fraction of the edges in the graph). For this case Zwick [Zwi99] showed how to obtain approximation ratios above $\alpha \simeq 0.87856$, using the tool of *outward rotation*. We propose to use RPR^2 instead. We observe that for instances of light Max-Cut for which threshold rounding is at its worse, there are misplaced vertices. (This may not be obvious to the reader at this point, but is true nevertheless.) Hence necessarily the approximation ratio can be improved, and the only question is by how much. By a suitable choice of a rounding function f, we can use the RPR^2 technique to give solutions that are locally optimal, and moreover, we can lower bound the approximation ratio that is obtained. As we shall see, this approximation ratio is better than the one obtained by Zwick using outward rotations. This is no coincidence, because as we shall show, outward rotation can be viewed as a special case of RPR^2, but with a choice of rounding function f that produces a solution that is not locally optimal.

The use of RPR^2 above can be viewed as using a threshold scheme (random hyperplane) followed by a randomized local correction step (in the randomized rounding phase vertices change sides with probabilities related to their distance from the hyperplane). Hence the choice of rounding function f guides the local correction step, and the RPR^2 methodology gives us a way of quantifying the effect of using a local correction step. Moreover, it is straightforward to derandomize the local correction step (using the method of conditional probabilities) giving a local correction step that is deterministic.

It is fair to remark that not in all cases it is advantageous to use the RPR^2 approach in order to guide and analyze local corrections. For example, local corrections are used successfully in Chapter 4 to improve the approximation ratio for Max-Cut for graphs of bounded degree. We do not think that the analysis appearing in Chapter 4 can be cast in the terminology of RPR^2.

Another class of problems for which RPR^2 may be useful is cases where threshold rounding produces infeasible solutions. Given a graph $G = (V, E)$ the Max-Bisection problem is the problem of finding a partition $(U, V \setminus U)$ of V into two equally sized sets (*i.e.* a *bisection* of V) that maximizes the number of edges cut by the partition. The algorithm of [GW95] (described above) for the Max-Cut problem on G will probably yield a partition $(U, V \setminus U)$ which is not a bisection (*i.e.* $|U| \neq |V \setminus U|$). Hence, in order to obtain a feasible partition of V (*i.e.* a bisection), the partition $(U, V \setminus U)$ must be modified (e.g., by moving vertices from the large side of the partition to the smaller one until both sides are equal). It is very difficult to analyze the effect of this additional step. There has been a sequence of papers [FJ97, Ye01, HZ01], each improving the bounds of the previous papers. We observe that RPR^2 is a natural rounding technique to use in this context, because by an appropriate choice of the rounding function f (possibly, based on the outcome of the random projection), we can guarantee that the two sides are of (nearly) the same size. We show a particular choice of rounding function f that modestly improves the known approximation ratio for Max-Bisection. We suspect that there are choices of f that give more dramatic improvements, though we are still struggling with their analysis.

A function f is called *s-linear* if for some $s \geq 0$ it is of the form $f(x) = 0$ for $x \leq -s$, $f(x) = 1$ for $x \geq s$, and $f(x) = \frac{1}{2} + \frac{x}{2s}$ for $-s \leq x \leq s$. As concrete examples of our results, we have the following theorem:

Theorem 3.1.1 *Using RPR^2 with an s-linear rounding function f, one obtains the following approximation ratios.*

- *For light Max-Cut for instances in which the optimal cut contains at most* 0.6 *of the edges, the ratio is at least 0.9128. (Previous best bound was below 0.9119.)*

- *For Max-Bisection the ratio is at least 0.7028. (Previous best bound was below 0.7016.)*

This chapter is organized as follows. In Section 3.2 we review the random hyperplane and outward rotation rounding techniques. In Section 3.3 we define RPR^2. In Section 3.4 we show that outward rotation is a special case of RPR^2. In Section 3.5 we analyze the use of RPR^2 on the Max-Cut and Max-Bisection problems. In Section 3.6 we briefly discuss some research directions which include the use of RPR^2 in the design of approximation algorithms for problems other than Max-Cut and Max-Bisection. Finally, in Section 3.7 we present the proof of several technical lemmas that appear throughout this chapter.

3.2 SDP relaxation of Max-Cut and various roundings

Consider the Max-Cut problem on a graph $G = (V, E)$ with $|V| = \{1, \dots, n\}$. Recall its representation as a quadratic integer program presented in Section 1.2.

(QI-MC)	Maximize	$\sum_{e_{ij} \in E} \frac{1 - x_i \cdot x_j}{2}$	
	subject to:		
		$x_i \in \{-1, 1\}$	for $1 \leq i \leq n$

The requirement $x_i \in \{-1, 1\}$ can be relaxed by representing each variable x_i by a unit n-dimensional vector $v_i \in S^{n-1}$ (here S^{n-1} is the unit sphere) and the multiplication $x_i \cdot x_j$ by the inner product $\langle v_i, v_j \rangle$.

(SDP-MC)	Maximize	$\sum_{e_{ij} \in E} \frac{1 - \langle v_i, v_j \rangle}{2}$	
	subject to:		
		$v_i \in S^{n-1}$	for $1 \leq i \leq n$

A solution to *(SDP-MC)* is a set of unit vectors in R^n, rather than a cut of G. To obtain a cut $(U, V \setminus U)$ of G we *round* the set of optimal vectors $v_1, \dots, v_n$ obtained by solving *(SDP-MC)*. One such rounding technique, presented by Goemans and Williamson [GW95] is the *random hyperplane* rounding technique.

Random hyperplane rounding: Let $r = r_1, \dots, r_n$ be a random variable with an n dimensional standard normal distribution (*i.e.* each coordinate r_i is an independent random variable with standard normal distribution). It can be seen that r is spherically symmetric, namely the direction specified by the vector $r \in R^n$ is uniformly distributed (see for instance [Ren70]). In the random hyperplane rounding technique a random vector r of the above distribution is chosen [1] and the vectors $v_1, \dots, v_n$ are partitioned into two sets according to the sign of the inner product $\langle v_i, r \rangle$. That is, a cut $(U, V \setminus U)$ is defined by the set $U = \{i \mid \langle v_i, r \rangle > 0\}$. [2]

[1] As the direction of r is uniformly distributed, choosing a random r can be viewed as choosing (the normal of) a random hyperplane.

[2] Equivalently, the vectors $v_1, \dots, v_n$ are partitioned into two sets, those that lie *above* the random hyperplane and those that do not.

Using the semidefinite program *(SDP-MC)* and the random hyperplane rounding, [GW95] obtain a $\simeq 0.87856$ approximation ratio for the Max-Cut problem. A number of other approximation algorithms for various problems, including Max-Cut, have been designed using semidefinite programming and variations of the random hyperplane rounding technique (for example [FG95, KMS98, FJ97, KZ97, CS98, Zwi99, FKL02, Sku01]). In some of these algorithms, the vectors $v_1, \ldots, v_n$ are rearranged and only then rounded using a random hyperplane. In others, the vectors $v_1, \ldots, v_n$ are rounded using the random hyperplane technique and then the combinatorial solution obtained is changed in order to improve the value of the solution or to assure its feasibility.

Outward rotations: One method used to rearrange the vectors obtained by the solution of a semidefinite relaxation (prior to the use of random hyperplane rounding) is *outward rotations* [Nes98, Ye01, Zwi99]. Let $\gamma \in [0,1]$. Given a set of vectors $v_1, \ldots, v_n$ in R^n, obtained by the solution of a semidefinite program, the γ-outward rotation of $v_1, \ldots, v_n$ are a set of new vectors $\hat{v}_1, \ldots, \hat{v}_n$ in R^{2n}. The vector $\hat{v}_i$ is defined to be $\sqrt{1-\gamma}v_i + \sqrt{\gamma}e_i \in R^{2n}$ where the original vectors v_i are viewed as vectors in R^{2n} (R^n being a subspace of R^{2n}) and $e_1, \ldots, e_n$ are a set of orthonormal vectors in R^{2n} that are also orthogonal to the vectors $v_1, \ldots, v_n$. In general, when $\gamma = 1$ the γ-outward rotation of the vectors $v_1, \ldots, v_n$ is a new vector configuration in which all vectors are orthogonal, and when $\gamma = 0$ the γ-outward rotation does not change the vectors $v_1, \ldots, v_n$. For intermediate γ, the γ-outward rotation is somewhere in between. Outward rotation has been used in the design of approximation algorithms for special instances of the Max-Cut problem, and other problems closely related to Max-Cut (for example [Zwi99, Ye01, HZ01]).

3.3 The RPR^2 family of rounding procedures

Let $v_1, \ldots, v_n$ be a set of vectors obtained by the solution of a semidefinite relaxation. We define a family of rounding procedures parameterized by a function $f : R \to [0,1]$. We denote this family of rounding procedures as the *random projection, randomized rounding* (RPR^2) family. An RPR^2 procedure using f has two steps and is defined as follows:

- **Step 1 (Projection):** Project the vectors $v_1, \ldots, v_n$ onto a random one dimensional subspace (*i.e.* a line). This is done by choosing a random variable r with n dimensional standard normal distribution, and projecting each v_i onto the one dimensional subspace containing r. For each i, let x_i be the directed distance (times $\|r\|$) of the projected vector v_i from the origin (*i.e.* $x_i = \langle v_i, r \rangle$).
- **Step 2 (Randomized rounding):** Define the $\{0,1\}$ solution $a_1, \ldots, a_n$: for each i set a_i to be 1 independently with probability $f(x_i)$.

The standard random hyperplane rounding technique presented in [GW95] in the approximation of the Max-Cut problem is a member of the RPR^2 family. The function f corresponding to random hyperplane rounding is the function which is 1 for all $x > 0$, and zero otherwise. In Section 3.4 we show that the outward rotation rounding technique is also a special case of RPR^2. In Section 3.5 we study the use of RPR^2 on the "light" Max-Cut problem and on the Max-Bisection problem. For both these problems outward rotation was used in order to obtain the previously best approximation ratios. We show functions f which when used in RPR^2 give better approximation ratios.

Analyzing RPR^2: Let $v_1, \ldots, v_n$ be the solution of a semidefinite relaxation on a given graph $G = (V, E)$, and let $a_1, \ldots, a_n$ be the $\{0, 1\}$ solution obtained by using RPR^2 with some function f. An edge e_{ij} is cut if $a_i \neq a_j$. As RPR^2 is a randomized procedure, the number of edges cut is a random variable. We wish to compute its expectation. For this, we analyze the probability of the event "$a_i \neq a_j$".

Let $r = r_1, \ldots, r_n$ be an n dimensional standard normal vector. In general, given r, the probability that "$a_i \neq a_j$" depends on the vectors v_i, v_j, and the function f. Hence, integrating over all possible r one can compute the probability of this event. However, as r is spherically symmetric this probability can be computed using two independent standard normal random variables r_1, r_2, and the angle between v_i and v_j alone. Given two vectors v_i and v_j that form an angle of θ_{ij}, let $P_f(\theta_{ij})$ denote the probability that the corresponding values a_i and a_j differ. Let $\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ be the density function of a standard normal random variable.

Lemma 3.3.1 *Let $\theta \in [0, \pi]$, $z(r_1, r_2) = \cos(\theta) r_1 + \sin(\theta) r_2$. Given two vectors v_i and v_j that form an angle of θ the probability that the corresponding values a_i and a_j differ after using RPR^2 with a function f is:*

$$P_f(\theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} [f(r_1)(1 - f(z(r_1, r_2))) + f(z(r_1, r_2))(1 - f(r_1))]\phi(r_1)\phi(r_2) dr_1 dr_2$$

Proof: Let $r = r_1, \ldots, r_n$ be an n dimensional standard normal random vector. It is not hard to verify that the probability that "$a_i \neq a_j$" is exactly

$$\int_r [f(\langle v_i, r\rangle)(1 - f(\langle v_j, r\rangle)) + f(\langle v_j, r\rangle)(1 - f(\langle v_i, r\rangle))]\phi(r) dr$$

where $\phi(r)$ is the density function of r. As r is spherically symmetric, we may choose r_1 to be in the direction of v_i and r_2 to be orthogonal to r_1 in the plane formed by v_i and v_j. In such a case, the inner product $\langle v_i, r\rangle$ equals r_1 and the inner product $\langle v_j, r\rangle$ equals $r_1 \cos(\theta) + r_2 \sin(\theta) = z(r_1, r_2)$. The lemma follows by integrating over such r. $\square$

By linearity of expectation, the expected number of edges cut is $\sum_{e_{ij} \in E} P_f(\theta_{ij})$, where θ_{ij} is the angle formed by the vectors v_i and v_j corresponding to e_{ij}. Dividing this value by $|E|$ we get the expected *fraction* of edges cut, which we denote by $E[Cut_f]$.

For an edge that makes an angle of θ, let $SDP(\theta) = (1 - \cos\theta)/2$ be its contribution to the semidefinite program. Then a convenient lower bound on the approximation ratio achieved for Max-Cut by RPR^2 with a rounding function f is $\min_{\theta > 0} P_f(\theta)/SDP(\theta)$. For light Max-Cut, this lower bound is too pessimistic. The angle θ minimizing the above expression cannot hold simultaneously for all edges, because then the graph would contain a cut that is too large. A stronger lower bound on the approximation ratio can be derived by more detailed analysis, following principles outlined in [Zwi99]. For Max-Bisection, there are more complications, because the cut obtained by the rounding technique is not necessarily a bisection. An additional step of moving vertices from the larger side to the smaller one is used, and analyzing its effect (or at least, providing lower bounds), can be done using the techniques outlined in [FJ97, Ye01, HZ01]). In Section 3.5 we analyze the use of RPR^2 on both the light Max-Cut and Max-Bisection problems. The numerical bounds for the approximation ratios that are presented in this chapter were derived using analysis that follows the principles developed in [FJ97, Zwi99, Ye01, HZ01]).

3.4 Outward rotation is a special case of RPR^2

Let $v_1, \ldots, v_n$ be the optimal set of unit vectors corresponding to the vertices of a given graph $G = (V, E)$ obtained by solving a semidefinite program. Let γ be some value in $[0, 1]$. Recall that the γ-outward rotation $\hat{v}_1, \ldots, \hat{v}_n$ of the vectors $v_1, \ldots, v_n$ is defined by $\hat{v}_i = \sqrt{1-\gamma}v_i + \sqrt{\gamma}e_i \in R^{2n}$.

The standard use of outward rotation: In the standard use of outward rotation, a $\{0, 1\}$ solution $a_1, \ldots, a_n$ is obtained by rounding the vectors $\hat{v}_1, \ldots, \hat{v}_n$ by a random hyperplane. Specifically let $r = r_1, \ldots, r_{2n}$ be a random vector with a $2n$-dimensional standard normal distribution. Define the solution $a_1, \ldots, a_n$ by setting a_i to be one iff the inner product $\langle \hat{v}_i, r\rangle$ is positive. It is convenient to describe the solution $a_1, \ldots, a_n$ as the subset $U = \{i \in V \mid a_i = 1\}$ of V, *i.e.* $U = \{i \in V \mid \langle \hat{v}_i, r\rangle > 0\}$. Using the definition of $\hat{v}_i$ and the spherical symmetry of r, we have that the set U obtained is equal to $\{i \in V \mid \sqrt{1-\gamma}\langle v, r_1, \ldots, r_n\rangle + \sqrt{\gamma}r_{n+i} > 0\}$.

We would like to obtain the exact set U without the use of outward rotations. Instead we would like to use RPR^2.

Theorem 3.4.1 *For any $\gamma \in [0, 1]$, there is a function f_γ s.t. using RPR^2 with f_γ is equivalent to γ-outward rotation followed by random hyperplane rounding.*

Proof: Recall that the set obtained using γ-outward rotation is

$$U = \{i \in V \mid \sqrt{1-\gamma}\langle v_i, r_1, \ldots, r_n\rangle + \sqrt{\gamma}r_{n+i} > 0\},$$

where r_i are independent standard normal random variables. Denoting the inner product $\langle v_i, r_1, \ldots, r_n\rangle$ as x_i we have that $U = \left\{i \in V \mid r_{n+i} > -x_i\sqrt{\frac{1-\gamma}{\gamma}}\right\}$. This implies that once the values x_i are determined (by the random hyperplane rounding), the choice whether any vertex i is in U or not depends only on x_i, γ and r_{n+i}; and independent of the choice made regarding all other vertices. In other words, for each vertex $i \in V$, the probability that i is in U, given the value of x_i, is exactly the probability that a standard normal random variable is greater than $-x_i\sqrt{\frac{1-\gamma}{\gamma}}$. Let $\phi(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$ and $\Phi(x) = \int_{-\infty}^{x} \phi(x)dx$ be the density function and distribution function of a standard normal random variable. Setting $f_\gamma(x)$ to be $\Phi\left(x\sqrt{\frac{1-\gamma}{\gamma}}\right)$ we may define the set U as the outcome of RPR^2. In the first step the values x_i are determined, and in the second step each vertex i is determined to be in U independently with probability $f_\gamma(x_i)$. A plot of the function $f_\gamma(x)$ for $\gamma = 1/2$ is given in Figure 3.1. □

In cases were outward rotation is used, it is natural to ask whether RPR^2 in combination with a different rounding function f can give better approximation ratios. It is our belief that the answer to this question is in general positive. That is, whenever outward rotation gives better approximation ratios than the random hyperplane rounding technique, one should expect RPR^2 to offer further improvements.

Let us note that for RPR^2, if we restrict ourselves to *nice* behaving rounding functions, finding the optimal rounding function f is not really a problem. For any fixed $\epsilon > 0$, there is a constant number of functions f (where this constant depends exponentially on $1/\epsilon$) such that at least one of them has expected approximation ratio within an additive error of at most ϵ from the optimal f. (This can be shown using concepts such as ϵ-nets.) An RPR^2 algorithm can even try out all these functions at run time and take the best result obtained. Hence, we may always assume that RPR^2 is performed

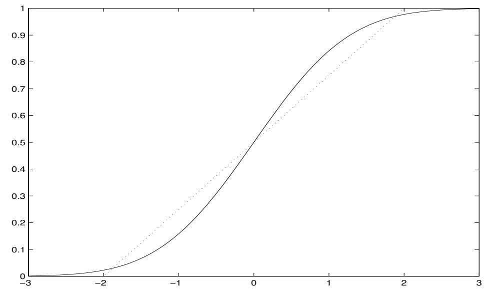

Figure 3.1: The function $f(x)$ representing γ-outward rotations for $\gamma = 0.5$ in the range $[-3, 3]$. (solid line). The s-linear function $f_s^*(x)$, for $s = 2$ (dotted line).

with the best possible rounding function. The problem is in analyzing the approximation ratio that one obtains. Here it is useful to select one particular easy to analyze rounding function f, to compute the expected approximation ratio that this f gives, and use it as a lower bound on the approximation ratio of the RPR^2 scheme. In this respect, outward rotations are helpful, as they can be analyzed not only using the integrals of Lemma 3.3.1, but also via other techniques (as in [Ye01, Zwi99]), and these other techniques are often simpler to use. It would be fair to say that previous work on outward rotation served as inspiration to much of the work reported in this chapter.

3.5 Applications of RPR^2

3.5.1 Light Max-Cut

Let $G = (V, E)$ be a given graph, let $v_1, \dots, v_n$ be the optimal vector configuration obtained by solving the semidefinite relaxation *(SDP-MC)* presented in [GW95] (and in Section 3.2) of G, and let Z be the value of this relaxation. Can RPR^2 (with some specific f) be used on the Max-Cut problem in order to improve the approximation ratio of $\alpha \simeq 0.87856$ proved in [GW95]? If we define α to be the ratio between the expected value of the cut obtained using RPR^2, and the value of the semidefinite relaxation Z, the answer is negative. This is due to a recent work of Feige and Schechtman [FS02] that shows that the integrality gap of this relaxation is arbitrarily close to $1/\alpha$. Therefore, we will not try to improve the approximation ratio α on general instances G. Instead we shall consider special cases of Max-Cut.

Consider parameterizing the instances of Max-Cut according to the ratio between the value of the semidefinite relaxation Z and the total number of edges W. Goemans and Williamson [GW95] study Max-Cut restricted to instances G for which this ratio is greater that 0.844. For each value $t \in (0.844, 1]$, they show that using standard random hyperplane rounding on instances G for which $Z = tW$, will yield a cut of value at least $\alpha_t Z$ where $\alpha_t > \alpha$ (this implies an approximation ratio of α_t). In [FS02] it is shown that the integrality gap of *(SDP-MC)* on these restricted instances is arbitrarily close to $1/\alpha_t$. Therefore, we will not try to improve the algorithm of [GW95] restricted on such instances either.

Zwick [Zwi99] studies Max-Cut restricted to instances G for which $Z = tW$ and $t < 0.844$. We call this "light" Max-Cut. For these instances Zwick shows how to obtain approximation ratios $\alpha_t > \alpha$, using outward rotation followed by random hyperplane rounding. The value of the integrality gap of

(SDP-MC) on these restricted instances is not clear. We analyze the use of RPR^2 on light Max-Cut. Roughly speaking, we show

(a) Necessary conditions for a rounding function f to be the one that maximizes the expected value of the cut obtained by RPR^2.

(b) Outward rotation (a special case of RPR^2) is not the best rounding function for RPR^2, as it does not satisfy these necessary conditions.

(c) We present an s-linear rounding function that gives an approximation ratio strictly above the ratio of α_t presented in [Zwi99].

We suspect that RPR^2 (with the optimal choice of f) achieves an approximation ratio that matches the inverse of the integrality gap of *(SDP-MC)* (as a function of t). We are trying to extend the techniques of [FS02] in order to prove this.

Our analysis involves the numerical evaluation of double integrals (presented in Lemma 3.3.1). These evaluations have been performed using MATLAB functions within precision of 10^{-6}. As such computations are time consuming, item *(b)* and *(c)* above are shown for a few values of t (namely $t = 0.55, 0.6, 0.7$).

Properties of the best function f for RPR^2: Given a graph $G = (V, E)$ and a set of vectors $v_1, \ldots, v_n$ obtained by solving *(SDP-MC)* on G, let $E[Cut_f]$ be the expected fraction of edges cut by using RPR^2 with a function f. Call a function *well behaved* if it is piecewise differentiable and the (first) derivatives in each piece are bounded by some global (negative) constant. We are interested in finding a well behaved function f^* that maximizes $E[Cut_f]$. Given any constant $\varepsilon > 0$, an approximation (related to ε) to the *best* such function f^* can be found by exhaustively trying out all possible f within a family whose size is exponential in $poly(\varepsilon^{-1})$ (as mentioned in Section 3.4). This can be done by approximating well behaved functions f by piecewise linear functions [3]. The value $E[Cut_f]$ can then be computed either numerically (as suggested in Lemma 3.3.1) or by actually rounding the vectors $v_1, \ldots, v_n$ using RPR^2 with the function f.

For light Max-Cut we identify a necessary condition for any well behaved function f that maximizes $E[Cut_f]$. We use this necessary condition to prove that the functions f corresponding to outward rotation are not optimal. Moreover, this necessary condition helps to guide us in finding a better rounding function (in our case, an s-linear function), without resorting to a tedious exhaustive search type approach for such a function.

A natural property that one could expect from an optimal function f^* is that rounding the vectors $v_1, \ldots, v_n$ using RPR^2 with f^* yields a cut $(U, V \setminus U)$ which is expected to be locally optimal (*i.e.* there are no vertices with an expected majority of neighbors on their side of the cut). For instance for the function $f(x) \equiv 1/2$ this property holds. The necessary condition we suggest is closely related to this "local optimality" property.

Let f be some RPR^2 function. Recall that $P_f(\theta)$ (defined in Lemma 3.3.1) measures the probability that using RPR^2 with the function f, two vectors v_i and v_j that form an angle of θ have corresponding values a_i and a_j that differ. One can examine this probability conditioned on the event that the inner product between v_i and the random vector r used in RPR^2 is fixed to be a specific value r_1. We

[3] A closer look at $E[Cut_f]$ assures us that we may assume some large constant M (depending on ε) for which $f(x) = 1$ if $x > M$ and $f(x) = 0$ if $x < -M$, without losing much.

denote this probability as $P_f(\theta)|r_1$. Let $E[Cut_f]$ be the expected fraction of edges cut by using RPR^2 with f. Recall that $E[Cut_f]$ is a normalized sum of $P_f(\theta_{ij})$, where θ_{ij} is the angle between the vectors corresponding to edges e_{ij} in E. Define $E[Cut_f \mid r_1]$ as the corresponding normalized sum of $P_f(\theta)|r_1$. In Section 3.7 we prove

Theorem 3.5.1 *If f^* is an optimal (well behaved) RPR^2 function then for all r_1 we have that $E[Cut_{f^*} \mid r_1] \geq 1/2$ with equality if $0 < f(r_1) < 1$ (*i.e. *$f(r_1) \notin \{0,1\}$).*

Theorem 3.5.1 is proven in a constructive manner. That is, if f is some RPR^2 function that does not satisfy the above conditions in some interval Δ, we show how to modify f in Δ to obtain a new function f^* such that $E[Cut_{f^*}] > E[Cut_f]$, thus implying that f is not optimal.

Outward rotation is not the best function for RPR^2: For an instance $G = (V, E)$ of Max-Cut, let $W = |E|$, let $v_1, \ldots, v_n$ be the vector configuration obtained by solving relaxation *(SDP-MC)* on G and let Z be the value of the relaxation. Let $t \in [0.5, 0.844)$. Assume a graph $G = (V, E)$, with a corresponding vector configuration $v_1, \ldots, v_n$ of value $Z = tW$. In [Zwi99] it is shown that rounding $v_1, \ldots, v_n$ using γ-outward rotation, an expected approximation ratio strictly above α_t will be obtained unless for each edge e_{ij} in E, the corresponding vectors v_i and v_j form an angle of either zero or θ_t (where θ_t is some specific angle greater than $\pi/2$ that depends on t).

In other words, only on graphs $G = (V, E)$ with a corresponding vector configuration in which a δ fraction of edges e_{ij} in E have corresponding vectors v_i and v_j that form an angle of zero, and a $1 - \delta$ fraction of edges have corresponding vectors that form an angle of θ_t, does the algorithm of [Zwi99] obtain an approximation ratio of no better than α_t. On all other graphs the algorithm of [Zwi99] has an approximation ratio strictly greater than α_t. In [ASZ02] it is shown that these worst case instances actually exists (*i.e.* there are graphs G with $Z = tW$ such that the algorithm presented in [Zwi99] yields a cut of expected value no better than $Z\alpha_t$).

Let f_γ be the RPR^2 function corresponding to γ-outward rotation. In Section 3.7 we show for such worst case graphs the existence of a non negligible interval $\Delta \subseteq R$, and a constant $\varepsilon > 0$ such that $E[Cut_{f_\gamma} \mid r_1] < 1/2 - \varepsilon$ for all $r_1 \in \Delta$ (we depict our results in Figure 3.2 of this section). We then state a quantitative version of Theorem 3.5.1, which explicitly shows how to construct a new function f^* by modifying f_γ in the interval Δ such that $E[Cut_{f^*}] > E[Cut_{f_\gamma}] + poly(\varepsilon)$.

Theorem 3.5.2 *There exists a constant $\varepsilon > 0$, such that using RPR^2 on the worst case graphs of [Zwi99] an approximation ratio of $\alpha_t + \varepsilon$ can be obtained.*

This implies an improved approximation algorithm for Max-Cut on general instances G with $Z = tW$. If the given graph has a vector configuration *close* to the worst case configuration use the best function for RPR^2, otherwise use the original algorithm of [Zwi99] (we rely on the fact that RPR^2 rounding is continuous with respect to the vector configuration $v_1, \ldots, v_n$).

As noted previously, our analysis involves the numerical evaluation of integrals, thus the above theorem has been proven for $t = 0.55, 0.6, 0.7$. We have no reason to believe that our results depend on these particular values of t.

An example for superior RPR^2 functions: We have shown that an approximation ratio greater than α_t can be obtained on graphs G with $Z = tW$ by improving the approximation ratio obtained on worst case graphs of [Zwi99]. In the following, we show that such an improvement can be proven

t	Ratio α_t [Zwi99]	RPR^2 ratio	Additive improvement	s
0.55	0.941282	0.942562	0.0012	0.96
0.6	0.911890	0.912809	0.0009	0.635
0.7	0.886251	0.886453	0.0002	0.263

Table 3.1: The approximation ratio obtained in [Zwi99] for various values of t, along side the improved approximation ratio obtained by RPR^2 using an s-linear function f_s^*. For each value t we state the ratio α_t of [Zwi99], the improved ratio obtained by RPR^2, the difference between these ratios, and the value s used in f_s^*.

directly. That is, given a value t and a graph G with $Z = tW$, we are interested in proving a lower bound on the expected value of the cut obtained by RPR^2. This can be done by choosing some function f, and analyzing the value of $P_f(\theta)$ for every $\theta \in [0, \pi]$.

Lemma 3.5.3 *Let $Z = tW$ for $t \in [1/2, 1]$, and f be some RPR^2 function. If for every $\theta \in [0, \pi]$ the value $P_f(\theta)$ is greater than*

$$A\left(\frac{1-\cos(\theta)}{2}\right) + B$$

*then using RPR^2 with the function f will yield a cut of expected value at least $Z(A + B/t)$ (*i.e.* using RPR^2 with the function f an expected approximation ratio of $A + B/t$ is achieved).*

Proof: Let $v_1, \ldots, v_n$ be the optimal vectors obtained by solving the semidefinite relaxation *(SDP-MC)* on a given graph $G = (V, E)$, and let Z be its optimal value. Recall that $Z = \sum_{e_{ij}} \frac{1-\cos(\theta_{ij})}{2}$, where for each edge e_{ij} the angle between its corresponding vectors v_i and v_j is θ_{ij}.

Let W be the total weight of edges in G, recall that $Z = tW$. The expected value of the cut obtained by RPR^2 rounding $v_1, \ldots, v_n$ is

$$\sum_{e_{ij}} P_f(\theta_{ij}) \geq \sum_{e_{ij}} A\left(\frac{1-\cos(\theta_{ij})}{2}\right) + B = AZ + BW = Z(A + B/t)$$

□

Consider replacing the function f_{γ_t} that corresponds to outward rotation, by the s-linear function f_s^* depicted by a dotted line in Figure 3.1 (in Section 3.4). Let s be some threshold, we define f_s^* to be the continuous function that is zero for all $x \leq -s$, one for all $x \geq s$, and linear for all $x \in (-s, s)$ (specifically for $x \in (-s, s)$ the value of $f_s^*(x)$ is $\frac{1}{2} + \frac{x}{2s}$).

Let $t = 0.55, 0.6, 0.7$. Using the above lemma, we have analyzed the expected approximation ratio achieved by using RPR^2 (with the functions f_s^*) on graphs G for which $Z = tW$. Our results are summarized in Table 3.1.

In Figure 3.2 we show that the value of $E[Cut_{f_s^*} \mid r_1]$ (for $t = 0.6$) comes very close to satisfying the necessary conditions of Theorem 3.5.1. For comparison we present the value of $E[Cut_f \mid r_1]$ for the threshold function of [GW95] and the function f_{γ_t} of [Zwi99].

The functions f_s^* are not claimed to be optimal, but as we have checked many different functions, we believe that they are close to being so. This implies that the original functions corresponding to outward rotation seem to be very close to being optimal.

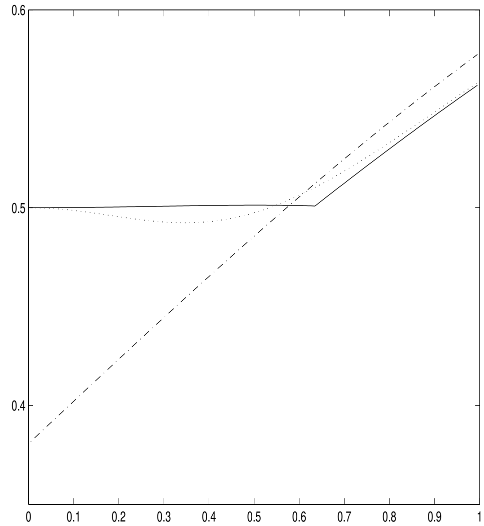

Figure 3.2: The value of $E[Cut_f \mid r_1]$ as a function of r_1 in the interval $[0,1]$. $E[Cut_f \mid r_1]$ is evaluated on the worst case graph of [Zwi99] (for $t = 0.6$) for various RPR^2 functions. *(a)* The threshold function of [GW95] (dash-dot), *(b)* the function f_{γ_t} of [Zwi99] (dotted), and *(c)* the s-linear function $f^*_{0.635}$ (solid). The threshold function of [GW95] violates the first optimality condition of Theorem 3.5.1, namely that $E[Cut_f \mid r_1] \geq 1/2$ for all r_1. The function f_γ of [Zwi99] improves on the threshold function but still violates both the first and second condition of Theorem 3.5.1. The s-linear function used in our analysis is very close to obeying both optimality conditions. Note that the *knee* of $f^*_{0.635}$ is exactly in the right place (*i.e.* $r_1 = 0.635$).

3.5.2 Max-Bisection

Given a graph $G = (V, E)$ the Max-Bisection problem is the problem of finding a partition $(U, V \setminus U)$ of V into two equally sized sets (*i.e.* a *bisection* of V) that maximizes the number of edges cut by the partition. As in the case of Max-Cut, the Max-Bisection problem can be phrased as a quadratic integer program, and relaxed as a semidefinite program. Given a graph $G = (V, E)$ a semidefinite relaxation of the Max-Bisection problem on G is

(SDP-MB)	Maximize	$\sum_{e_{ij} \in E} \frac{1 - \langle v_i, v_j \rangle}{2}$	
	subject to:		
	(1)	$\sum_{i,j} \langle v_i, v_j \rangle = 0$	
	(2)	$v_i \in S^{n-1}$	for $1 \leq i \leq n$

Let $U \subseteq V$ of size $n/2$. Set v_i to be the vector $(1, 0, \ldots, 0)$ for every $i \in U$ and $(-1, 0, \ldots, 0)$ for $i \in V \setminus U$. It is not hard to verify that $v_1, \ldots, v_n$ is a valid solution to *(SDP-MB)*. Under this assignment, the value of the objective function of *(SDP-MB)* is exactly the number of edges cut in the partition $(U, V \setminus U)$. Hence the value of *(SDP-MB)* is at least as high as the optimal value of the corresponding Max-Bisection problem.

A number of approximation algorithms for Max-Bisection based on semidefinite programming have been suggested ([FJ97, Ye01, HZ01]). In these algorithms, given $G = (V, E)$ with $|V| = n$, the semidefinite relaxation *(SDP-MB)* (or similar programs) are solved yielding a set of vectors $v_1, \dots, v_n$. These vectors are then rounded (using the random hyperplane or outward rotation technique) in order to obtain a cut $(U, V \setminus U)$ of G. As this cut is not necessarily a bisection (*i.e.* U is not necessarily of size $n/2$) the cut $(U, V \setminus U)$ is modified by moving vertices from the large side of the cut to the smaller side until both sides are equal.

In the algorithm of [FJ97], the relaxation *(SDP-MB)* and random hyperplane rounding is used to obtain an approximation ratio of 0.6514. In [Ye01] the relaxation *(SDP-MB)* and outward rotation is used to obtain an approximation ratio of 0.699. In [HZ01] the relaxation *(SDP-MB)* with additional *triangle* constraints [4], and outward rotation is used to obtain an approximation ratio of 0.7016.

As in the case of Max-Cut we study the use of RPR^2 in the algorithms above. It is natural to assume the existence of a function f other than those corresponding to outward rotation that will yield an improved approximation ratio for the Max-Bisection problem. Using the analysis of these algorithms, and *plugging* in an s-linear function f_s^* (as defined in Section 3.5.1) for $s = 0.605$ we obtain an approximation ratio of 0.7017 using the relaxation *(SDP-MB)* (as opposed to 0.699 of [Ye01]), and an approximation ratio of 0.7028 using the relaxation *(SDP-MB)* with additional *triangle* constraint (as opposed to 0.7016 of [HZ01]). As before our analysis involves the numerical evaluation of integrals. For completeness, in Section 3.7 we sketch the proof of the second part of Theorem 3.1.1 (the 0.7028 approximation ratio for Max-Bisection). Our proof follows the line of proof appearing in [HZ01].

3.6 Discussion

The RPR^2 method of rounding the solution of semidefinite programs was presented. We showed that the optimal rounding function f can be chosen efficiently on a per-instance basis, and discussed methods of lower bounding the resulting approximation ratio. We also discussed the question of which problems lend themselves most naturally to this method. We presented improved approximation ratios for some problems related to Max Cut. The value of these approximation ratios was computed by numerically evaluating certain integrals. Though potentially there may be errors in numerical evaluation of integrals (hardware/software/methodological errors), the main contribution of our work does not depend on the exact outcomes of these integrals. Even without computing exact numbers, it is rigorously proven that RPR^2 is at least as good as the random hyperplane rounding technique and the outward rotation technique, and it would be a remarkable coincidence if for any of the problems discussed in this chapter RPR^2 does not actually produce approximation ratios strictly better than those previously proven. (Not to mention the fact that the previously published bounds are also results of numerical analysis.)

Many questions remain open, but seem within reach. For "light" Max Cut, we suspect that RPR^2 (with the optimal choice of f) achieves an approximation ratio that matches the integrality ratio (as a function of the relative size of the maximum cut). We are trying to extend the techniques of [FS02] in order to prove this. For Max Bisection, we suspect that more substantial improvements of the approximation ratio can be proven for other choices of rounding function f. For some other problems, especially those currently analyzed using outward rotation (such as Not-All-Equal-3SAT [Zwi99]), it is natural to assume that the approximation ratio can be improved using RPR^2, but this needs to be seen.

[4] We denote the constraints $\langle v_i, v_j\rangle + \langle v_i, v_k\rangle + \langle v_j, v_k\rangle \geq -1$; $-\langle v_i, v_j\rangle - \langle v_i, v_k\rangle + \langle v_j, v_k\rangle \geq -1$ for all $1 \leq i, j, k \leq n$ as triangle constraints,

3.7 Proof of claims

3.7.1 Necessary optimality condition

Let $v_1, \ldots, v_n$ be an optimal vector configuration obtained by solving the semidefinite relaxation of the Max-Cut problem of $G = (V, E)$. For each edge e_{ij} let θ_{ij} be the angle formed by the vectors v_i and v_j. Let δ_0 be the fraction of edges e_{ij} with corresponding vectors v_i and v_j that form an angle of 0. In the following discussion we assume that $\delta_0 > 0$. Similar analysis can be made for $\delta_0 = 0$.

Let $\theta_0 = 0$ and $\Theta = \{\theta_0, \theta_1, \ldots, \theta_m\} = \cup_{e_{ij} \in E}\{\theta_{ij}\}$ (that is, $\theta \in \Theta$ iff $\exists e_{ij} \in E$ s.t. $\theta = \theta_{ij}$). Let $\delta'_1, \ldots, \delta'_m$ be the fraction of edges e_{ij} with corresponding vectors v_i and v_j that form an angle of θ_i. Set δ_i to be $\delta'_i/(1-\delta_0)$ for $i = 1 \ldots m$. Note that $\sum_{i=1}^m \delta_i = 1$. Given a function f, the expected fraction of edges cut using RPR^2 on $v_1, \ldots, v_n$ is

$$E[Cut_f] = \delta_0 P_f(0) + (1-\delta_0)\sum_{i=1}^{m} \delta_i P_f(\theta_i).$$

By setting $z_i = z_i(a,b) = a\cos(\theta_i) + b\sin(\theta_i)$, and $F_i(a,b) = f(a)(1-f(z_i)) + f(z_i)(1-f(a))$ we have (Lemma 3.3.1) that $E[Cut]$ is equal to:

$$2\delta_0 \int_{r_1} f(r_1)(1-f(r_1))\phi(r_1)dr_1 + (1-\delta_0)\sum_{i=1}^{m}\int_{r_1}\int_{r_2} \delta_i F_i(r_1,r_2)\phi(r_1)\phi(r_2)dr_1 dr_2.$$

Define the value of $E[Cut_f]$ conditioned on r_1 as

$$E[Cut_f \mid r_1] = 2\delta_0 f(r_1)(1-f(r_1)) + (1-\delta_0)\sum_{i=1}^{m}\int_{r_2} \delta_i F_i(r_1,r_2)\phi(r_2)dr_2.$$

Given Θ and the fractions δ_i we are interested in finding a well behaved function f which maximizes $E[Cut_f]$.

Theorem 3.5.1 *If f^* is an optimal (well behaved) RPR^2 function then for all r_1 we have that $E[Cut_{f^*} \mid r_1] \geq 1/2$. Furthermore, for every r_1 s.t. $f^*(r_1) \in (0,1)$ it is the case that $E[Cut_{f^*} \mid r_1] = 1/2$.*

Proof: We start by proving the following technical lemma.

Lemma 3.7.1 *Let $\delta \in (0,1]$, $g(x,y) = \delta x(1-x) + (1-\delta)(x(1-y) + y(1-x))$, and $h(x,y) = 2\delta x(1-x) + (1-\delta)(x(1-y)+y(1-x))$. If $x, y \in [0,1]$ such that $h(x,y) < 1/2$, then there exists $x' \in [0,1]$ such that $g(x',y) > g(x,y)$. Furthermore, if $x \in (0,1)$ and $h(x,y) \neq 1/2$ then there exists $x' \in [0,1]$ such that $g(x',y) > g(x,y)$.*

Proof (sketch): We start with the first assertion. Let $x, y \in [0,1]$ and $h(x,y) < 1/2$. Fix y and view $g_y(x) = g(x,y)$ as a function of x. $g_y(x)$ is a polynomial of degree 2 which obtains its maximum at $x' = \frac{1-2y(1-\delta)}{2\delta}$. Assume $x' \in (0,1)$. It is not hard to verify that $h(x',y) = 1/2$, implying that $x' \neq x$, which in turn implies that $g(x,y) < g(x',y)$. Otherwise $g_y(x)$ obtains its maximum in the range $[0,1]$ on $x' \in \{0,1\}$. Namely, $h(x',y) = g_y(x') > g_y(1-x') = h(1-x',y)$. Using the fact that $\frac{1-2y(1-\delta)}{2\delta} \notin (0,1)$, it can now be seen that $h(x',y)$ must be greater or equal to $1/2$, which again implies that $x' \neq x$ and $g(x,y) < g(x',y)$. For the second assertion we need only consider the first case discussed above. $\square$

We turn to the proof of our theorem. Let $f : R \to [0,1]$ be a continuous well behaved function (minor changes in the proof are needed for piecewise continuous f). Assume the existence of $r_1 \in R$ such that $E[Cut_f \mid r_1] < 1/2$. Let Δ be a small neighborhood of r_1. We present a well behaved function f^* which differs from f on the set Δ s.t. $E[Cut_{f^*}] > E[Cut_f]$, implying that f cannot be optimal.

Let $f^*(r_1)$ be the function that is equal to f on all $r_1 \notin \Delta$, in the following we define f^* on $r_1 \in \Delta$. For any $r_1 \in \Delta$, let $x = f(r_1)$, $y = \sum_{i=1}^m \int_{r_2} \delta_i f(z_i(r_1, r_2))\phi(r_2)dr_2$. Note that under these definitions

$$g(x,y) = \delta_0 f(r_1)(1 - f(r_1)) + (1-\delta_0)\sum_{i=1}^m \int_{r_2} \delta_i F_i(r_1, r_2)\phi(r_2)dr_2$$

and $h(x,y) = E[Cut_f \mid r_1]$. By our assumption we have that $h(x,y) < 1/2$. By Lemma 3.7.1, there exists x' s.t $g(x', y) > g(x,y)$. Set the value of $f^*(r_1)$ to be x'. It can be seen that f^* is continuous in Δ. Let

$$y' = \sum_{i=1}^m \int_{r_2;\ z_i \notin \Delta} \delta_i f(z_i(r_1, r_2))\phi(r_2)dr_2.$$

By the continuity of f and g, we may set Δ to be small enough such that $g(x', y') > g(x, y)$.

Let Λ_{f^*} be the part of $E[Cut_{f^*}]$ that depends on the value of f^* in Δ.

$$\begin{aligned}
\Lambda_{f^*} &= 2\delta_0 \int_{r_1 \in \Delta} f^*(r_1)(1 - f^*(r_1))\phi(r_1)dr_1 + \\
&\quad (1-\delta_0)\sum_{i=1}^m \int_{r_1 \in \Delta} \int_{r_2;\ z_i \notin \Delta} \delta_i F_i(r_1, r_2)\phi(r_1)\phi(r_2)dr_1 dr_2 + \\
&\quad (1-\delta_0)\sum_{i=1}^m \int_{r_1 \notin \Delta} \int_{r_2;\ z_i \in \Delta} \delta_i F_i(r_1, r_2)\phi(r_1)\phi(r_2)dr_1 dr_2 + \\
&\quad (1-\delta_0)\sum_{i=1}^m \int_{r_1 \in \Delta} \int_{r_2;\ z_i \in \Delta} \delta_i F_i(r_1, r_2)\phi(r_1)\phi(r_2)dr_1 dr_2
\end{aligned}$$

Define Λ_f analogously. To prove that $E[Cut_{f^*}]$ is strictly larger than $E[Cut_f]$ it suffices to analyze the difference between Λ_{f^*} and Λ_f.

$$\begin{aligned}
\Lambda_{f^*} &= 2\delta_0 \int_{r_1 \in \Delta} f^*(r_1)(1 - f^*(r_1))\phi(r_1)dr_1 + \\
&\quad 2(1-\delta_0)\sum_{i=1}^m \int_{r_1 \in \Delta} \int_{r_2;\ z_i \notin \Delta} \delta_i F_i(r_1, r_2)\phi(r_1)\phi(r_2)dr_1 dr_2 + \\
&\quad (1-\delta_0)\sum_{i=1}^m \int_{r_1 \in \Delta} \int_{r_2;\ z_i \in \Delta} \delta_i F_i(r_1, r_2)\phi(r_1)\phi(r_2)dr_1 dr_2 \\
&\geq 2\int_{r_1 \in \Delta} g\left(f^*(r_1), \sum_{i=1}^m \int_{r_2;\ z_i \notin \Delta} \delta_i f^*(z_i)\phi(r_2)dr_2\right)\phi(r_1)dr_1 \\
&= 2\int_{r_1 \in \Delta} g\left(x', y'\right)\phi(r_1)dr_1 \\
&> 2\int_{r_1 \in \Delta} g(x,y)\phi(r_1)dr_1 = 2\int_{r_1 \in \Delta} g\left(f(r_1), \sum_{i=1}^m \int_{r_2} f(z_i)\phi(r_2)dr_2\right)\phi(r_1)dr_1 \\
&\geq \Lambda_f.
\end{aligned}$$

t	θ_t	γ_t	ε_t	Δ_t
0.55	2.142538	0.267078	0.004	$[0.4, 0.7]$
0.6	2.184268	0.116382	0.007	$[0.3, 0.4]$
0.7	2.253619	0.021379	0.003	$[0.1, 0.2]$

Table 3.2: γ_t outward rotation is not the optimal function for RPR^2. For each value t we state the worst case angle θ_t from the algorithm of [Zwi99], the value γ_t used in [Zwi99], and the pair $\Delta_t,\ \varepsilon_t$ for which γ_t-outward rotation is suboptimal by Theorem 3.7.3

The second part of our theorem is proven analogously. □

3.7.2 Outward rotation is not the optimal function for RPR^2

The following lemma is a quantitative version of Lemma 3.7.1 (proof follows from basic computations).

Lemma 3.7.2 *Let $\varepsilon > 0$, $\delta \in (0, 1]$, $g(x, y) = \delta x(1 - x) + (1 - \delta)(x(1 - y) + y(1 - x))$, and $h(x, y) = 2\delta x(1 - x) + (1 - \delta)(x(1 - y) + y(1 - x))$. If $x, y \in [0, 1]$ such that $h(x, y) < 1/2 - \varepsilon$, then there exists $x' \in [0, 1]$ such that for all $y' \in [y - \frac{\delta\varepsilon^2}{100}, y + \frac{\delta\varepsilon^2}{100}]$ we have $g(x', y') > g(x, y) + \frac{\delta\varepsilon^2}{100}$.*

Let $t \in [0.5, 0.844)$. Consider a worst case graph $G = (V, E)$, and its corresponding vector configuration $v_1, \dots, v_n$ as described in Section 3.5.1. Recall that a δ fraction of edges e_{ij} in E have corresponding vectors v_i and v_j that form an angle of zero, and a $1 - \delta$ fraction of edges have corresponding vectors that form an angle of θ_t.

Let f be an RPR^2 function. Let $\delta_0 > 0$ and $\varepsilon > 0$ be constant. Let $\Delta \subseteq [0, 1]$ be an interval in R. Assume that $E[Cut_f \mid r_1]$ is less than $1/2 - \varepsilon$ for all $r_1 \in \Delta$, and that the probability of $r \in \Delta$ for a standard normal random variable r is $\theta\left(\delta_0\varepsilon^2\right)$ (for larger Δ, we take a subset of Δ). The following theorem is a quantitative version of Theorem 3.5.1.

Theorem 3.7.3 *Under these conditions, the function f can be modified to a function f^* (that differs from f in the interval Δ) such that $E[Cut_{f^*}] > E[Cut_f] + \theta\left(\delta_0^2\varepsilon^4\right)$.*

For a few values of t, we have analyzed the continuous functions f_{γ_t} corresponding to γ_t-outward rotation (used in the algorithm of [Zwi99] on graphs G for which $Z = tW$). Using the above theorem, we have shown that the functions f_{γ_t} are not optimal. Specifically, for each function we present an interval Δ_t and a value ε_t such that $E[Cut_{f_{\gamma_t}} \mid r_1] < 1/2 - \varepsilon_t$ for all $r_1 \in \Delta_t$. The value $E[Cut_{f_{\gamma_t}} \mid r_1]$ is analyzed numerically. Our results are summarized in Table 3.2.

In Figure 3.2 (of Section 3.5) we plot the value of $E[Cut_f \mid r_1]$ as a function of r_1 for $t = 0.6$ and three different functions *(a)* the threshold function of [GW95], *(b)* the function f_{γ_t} of [Zwi99] and *(c)* the s-linear function $f^*_{0.635}$ presented in Section 3.5.1. Note the improvement obtained by each function regarding the optimality condition of Theorem 3.5.1.

3.7.3 Proof of Theorem 3.1.1 (2)

In the following section we sketch the proof of Theorem 3.1.1 (2) (the 0.7028 approximation ratio for Max-Bisection). The proof follows the line of proof appearing in [HZ01].

Let $G = (V, E)$ be a given graph. Let *(SDP)* be the semidefinite relaxation for Max-Bisection presented in Section 3.5.2 enhanced with additional triangle constraints. Consider the algorithm that solves *(SDP)* and then rounds the optimal vector solution $\{v_1, \ldots, v_n\}$ obtained using the following rounding procedure:

1. Round the vectors using RPR^2 rounding with an 0.605-linear function to obtain a partition $(U, V \setminus U)$ of V. Assume w.l.o.g. that the size of U is greater or equal than the size of $V \setminus U$.

2. Move vertices from U to $V \setminus U$ in a greedy manner until both sides of the partition are of size $n/2$. Namely, at each step move the vertex from U that reduces the value of the cut by the least.

Let $w(U)$ be the value of the partition $(U, V \setminus U)$ obtained after the first part of the rounding procedure, and let $|U|$ be the size of U. Let Z be the value of the relaxation *(SDP)*, and let $\gamma > 0$ be a constant that will be defined later in the proof. Consider the random variable

$$X = \frac{w(U)}{Z} + 4\gamma \frac{|U|(n - |U|)}{n^2}$$

In the following we will analyze the expectation of X. We will then compute the approximation ratio obtained by the above algorithm given that X is arbitrarily close to its expectation. If X happens to be far from its expectation, we can perform the first step of the above rounding procedure again to obtain a new partition (and a new value for X). It is not hard to verify that a polynomial number of repetitions will suffice to obtain a partition with a corresponding variable X which is of value at least $E[X](1 - o(1))$.

Computing $E[X]$ Given two vectors v_i and v_j that from an angle of θ_{ij}, let $P(\theta_{ij})$ be the probability that their corresponding vertices will be on different sides of the partition $(U, V \setminus U)$ after the first step of the above rounding procedure. For an edge e_{ij} with corresponding vectors v_i and v_j that from an angle of θ_{ij} let $SDP(\theta_{ij}) = \frac{1-\cos(\theta_{ij})}{2}$ be its contribution to the objective function of *(SDP)*.

It is not hard to verify that $E\left[\frac{w(U)}{Z}\right] \geq \min_\theta \frac{P(\theta)}{SDP(\theta)}$. In Figure 3.3, $P(\theta)$ is plotted as a function of $SDP(\theta)$ (namely the function $p(x) = P(\arccos(1 - 2x))$ is plotted). Basic calculation show that the minimum of $P(\theta)/SDP(\theta)$ is obtained at $\theta \simeq 0.81627\pi$ and is at least $\alpha \simeq 0.83386$.

We now turn to compute $E[|U|(n - |U|)]$. Notice that $E[|U|(n - |U|)] = \sum_{i<j} P(\theta_{ij})$, where θ_{ij} is the angle between the vectors v_i and v_j. Moreover, notice that $\sum_{i<j} SDP(\theta_{ij}) = \frac{n^2}{4}$ (this is due to constraint (1) of *(SDP-MB)*) from Section 3.5.2. Finally, it is not hard to verify that $P(\theta)$ is bounded by below by the function $\frac{A(1-\cos(\theta))}{2} + B$ for $A = 0.65428$ and $B = 0.15514$ (or in other words the function $p(x)$ defined above is greater than $Ax+B$). The function $Ax+B$ is also depicted in Figure 3.3. Hence, by Lemma 3.5.3 we conclude that $E[|U|(n - |U|)] \geq \frac{n^2}{4}(A + 2B)(1 - o(1)) \geq 0.96456\frac{n^2}{4}$. This bound will not suffice for our proof as we present it (although it does suffice for the proof of a 0.7017 approximation ratio for Max-Bisection). To obtain the ratio of 0.7028 we use the following observation (used in [HZ01]) based on Turán's theorem. Turán's theorem states that any undirected graph on n vertices with more than $n^2/4$ edges contains a triangle ([Tur41]). This observation. together with the triangle constraints in *(SDP)* imply that the set $\{(i, j) \mid \langle v_i, v_j \rangle < -1/3\}$ is of size at most $n^2/4$. We use this fact in the following Lemma.

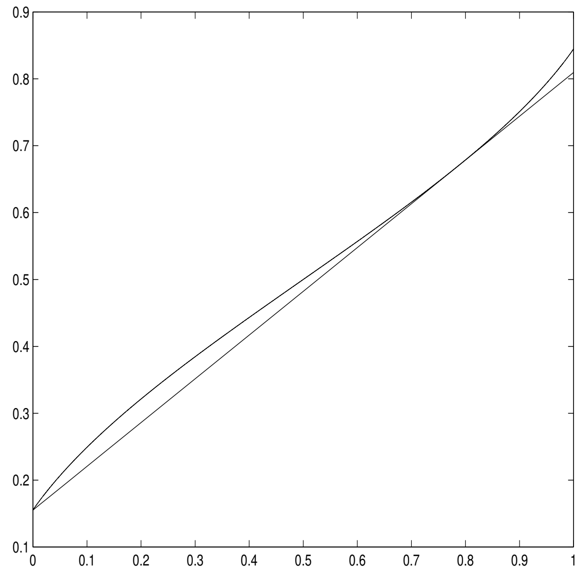

Figure 3.3: The functions $p(x) = P(\arccos(1 - 2x))$ and $0.65428x + 0.15514$.

Lemma 3.7.4 *Let $p(x) = P(\arccos(1 - 2x))$. Let y be a value between $2/3$ and 1. Let A_1, A_2 and B satisfy (a) $p(x) \geq A_1 x + B$ for x in the range $[0, 2/3] \cup [y, 1]$, (b) $p(x) \geq A_2 x + B$ for x in the range $(2/3, y)$, and (c) $A_1 \geq A_2$. Then $E[|U|(n - |U|)] \geq \frac{n^2}{4}(2B + A_2 + (A_1 - A_2)(1 - y))(1 - o(1))$.*

Proof: For $i < j$ and an angle θ_{ij} let $x_{ij} = \frac{1-\cos(\theta_{ij})}{2}$.

$$
\begin{aligned}
E[|U|(n - |U|)] &= \sum_{i<j} P(\theta_{ij}) = \sum_{i<j} p(x_{ij}) = \sum_{x_{ij} \in (2/3, y)} p(x_{ij}) + \sum_{x_{ij} \notin (2/3, y)} p(x_{ij}) \\
&\geq \sum_{x_{ij} \in (2/3, y)} (A_2 x_{ij} + B) + \sum_{x_{ij} \notin (2/3, y)} (A_1 x_{ij} + B) \\
&= \sum_{i<j} B + \sum_{i<j} A_2 x_{ij} + \sum_{x_{ij} \notin (2/3, y)} (A_1 - A_2) x_{ij} \\
&\geq (1 - o(1)) \frac{n^2}{4} (2B + A_2 + (A_1 - A_2)(1 - y))
\end{aligned}
$$

where the last inequality is due to the fact that (a) $\sum_{x_{ij} \in (2/3, y)} x_{ij} \leq \frac{n^2}{4} y$ (this follows from $\{(i, j) \mid x_{ij} \in (2/3, y)\} \subseteq \{(i, j) \mid \langle v_i, v_j \rangle < -1/3\}$ and Turán's Theorem) and (b) $\sum_{i<j} x_{ij} = \sum_{i<j} SDP(\theta_{ij}) = \frac{n^2}{4}$. $\square$

It can be seen that for $y = 0.8912$; the values $A_1 = 0.660668$, $A_2 = 0.654284$, and $B = 0.15514$ satisfy Lemma 3.7.4. Thus we conclude that $E[|U|(n - |U|) \geq \frac{n^2}{4}(2B + A_2 + (A_1 - A_2)(1 - y))(1 - o(1)) \geq 0.965258\frac{n^2}{4}$. Let $\beta = 0.965258$, we conclude that $E[X] \geq \alpha + \gamma\beta$.

Computing the approximation ratio Consider the partition $(U, V \setminus U)$ obtained after the first step of our rounding technique. Let X be the corresponding random variable and assume that $X \geq$

$E[X]$. Let $\lambda > 0$ and $\mu \in [1/2, 1]$ be such that $w(U) = \lambda Z$, and $|U| = \mu n$. It is not hard to verify that the final solution obtained by our rounding technique (after both step 1 and 2) is of value $\frac{\lambda}{2\mu} Z$. As $X = \lambda + 4\gamma\mu(1-\mu) \geq \alpha + \gamma\beta$ we conclude that the final solution is of value at least $\frac{\alpha+\gamma\beta-4\gamma\mu(1-\mu)}{2\mu} Z$. Setting γ to be equal to $\frac{\alpha}{2\beta}\left(\frac{1}{\sqrt{1-\beta}} - 1\right)$, and minimizing over μ it can be seen that the above expression is always greater than $0.7028Z$ (details can be found in [Ye01]), which concludes our proof.

Chapter 4

Max-Cut on bounded degree graphs

4.1 Introduction

Given a graph $G = (V, E)$, the Max-Cut problem on G is the problem of finding a partition $(U, V \setminus U)$ of the vertex set V which maximizes the number of edges with one endpoint in U and another in $V \setminus U$. An r approximation algorithm for the Max-Cut problem is an algorithm that when evoked on a graph G yields a partition $(U, V \setminus U)$ of value at least r times the value of the optimal partition (such an algorithm is also said to have an approximation ratio of r).

The currently best known approximation algorithm for the Max-Cut problem, is the algorithm of Goemans and Williamson [GW95]. Throughout this chapter we denote the algorithm of [GW95] as the GW algorithm. Given a graph G with vertex size n, the GW algorithm has two steps. In the first step a semidefinite relaxation of the Max-Cut problem on G is solved yielding a set of unit vectors $\{v_1, \ldots, v_n\}$ in R^n. In the second step this set of vectors is randomly rounded into a partition $(U, V \setminus U)$. The GW algorithm is shown to have an expected approximation ratio of at least $\alpha \simeq 0.87856$.

Given G, let $(U, V \setminus U)$ be the partition obtained by the GW algorithm. It is evident from [GW95] that it might be the case that a vertex of G and more than half its neighbors end up on the same side of the partition $(U, V \setminus U)$. We denote such vertices as *misplaced* vertices. Clearly in such a case the partition $(U, V \setminus U)$ can be improved by moving the misplaced vertex from one side of the partition to the other, thus yielding a new partition cutting more edges.

In the following chapter, we consider enhancing the GW algorithm with an additional local step that moves misplaced vertices from one side of the partition to the other, until no such vertices are left. We prove that the enhanced algorithm improves upon the original approximation ratio of α by a constant fraction when the maximum degree Δ of the given graph is constant. Namely we prove an approximation ratio of 0.921 on graphs of maximum degree three, of 0.924 on regular graphs of degree three, and of $\alpha + \Omega(1/\Delta^4)$ on graphs of maximum degree Δ. Our results apply to graphs G for which we assume that there are no weights associated with the edge set E. Our analysis can be extended to graphs with edge weights as well, although the results we obtain are weeker (we briefly discuss this in Section 4.4).

Previous work The Max-Cut problem on bounded degree graphs is mentioned in [BK98] where it is shown that it is *NP-hard* to approximate the Max-Cut problem on regular graphs of degree 3 beyond the ratio of 0.997. To the best of our knowledge, the best known approximation ratio for the

Max-Cut problem on bounded degree graphs prior our work was that of the general problem which is $\alpha \simeq 0.87856$. The Max-Cut problem restricted to 3 regular graphs has been previously studied in [CFMP99] (where a parallel approximation algorithm of ratio 3/4 is presented), and in [Pol95] (where it is shown that a local maximum can be found in polynomial time). In a recent paper Halperin, Livnat, and Zwick [HLZ02] improve over the approximation ratio of 0.921 that we present for graphs of degree at most three, and obtain a ratio of 0.9326. Their algorithm follows the general framework we present (the addition of a local improvement step to the GW algorithm) and differs from ours in the greedy procedure analyzed.

Triangle constraints Our results are achieved using an algorithm that differs from the one presented in [GW95] not only in the additional improvement step, but also in the fact that an *improved* semidefinite relaxation of the Max-Cut problem is used as a base of our algorithm. Specifically, we add *triangle* constraints (mentioned in [FG95]) to our relaxation. These additional constraints have been studied in the past in the context of several problems including the Max-Cut problem. The results of [BM86a] imply that the value of the Max-Cut semidefinite relaxation with triangle constraints is **equal** to the value of the optimal cut on planar graphs. Without these constraints it is shown in [FS02] that the original semidefinite relaxation has an integrality gap arbitrarily close to $1/\alpha$. In [GW95] it is shown that the integrality gap of the original relaxation has an integrality gap of least 1/0.885 even on two regular graphs (the 5 cycle). In [Zwi99] these constraints are used in order to achieve an improved approximation ratio on the Max-Cut problem on graphs which have a fractional triangle covering. The addition of such triangle constraints to the semidefinite relaxation of the Max-Cut problem on general graphs is not known to improve the *worst case* integrality gap of such a relaxation nor is it known to yield an improved approximation algorithm. In [Kar96] it is shown that the addition of such constraints will not improve the approximation ratio of the algorithm presented in [GW95] as is. In our work, the addition of such triangle constraints to the semidefinite relaxation of the Max-Cut problem on graphs of maximum degree three yields an improved approximation algorithm with a ratio of 0.921. As the original semidefinite relaxation of [GW95] has an integrality gap of at least 1/0.885 on a two regular graph, we conclude that the addition of such triangle constraints to our semidefinite relaxation is crucial to the success of our algorithm.

Local analysis As in the algorithm of [GW95], many other algorithms based on semidefinite programming use the *random hyperplane* rounding technique, and are analyzed in a *local* manner (for instance [FG95, KZ97, Zwi99]). That is, for each edge a local expected approximation ratio is computed. This ratio then holds as an expected approximation ratio for the algorithm as a whole due to linearity of expectation.[1] In our algorithm for the Max-Cut problem on graphs with maximum degree three, we present an analysis in which we compute a local approximation ratio not on single edges but on clusters of edges. In particular we concentrate on pairs of edges which share a common vertex. Such an analysis allows us to evaluate the contribution of our additional improvement step as a function of the vector configuration obtained by the semidefinite relaxation.

[1]We would like to note that other techniques have been used in the analysis of approximation algorithms based on semidefinite programming. For example Nesterov [Nes98] presents a general framework for analyzing the quality of semidefinite relaxations to quadratic optimization problems. Applying his techniques on the semidefinite relaxation of Max-Cut results with a $2/\pi \simeq 0.6366$ approximation ratio.

The general approach We would like to note, that our approach applied on Max-Cut can be viewed as a general approach to enhance randomized algorithms for various problems. Namely, given a random algorithm A which does not necessarily produce locally optimal solutions, we suggest enhancing A by adding a local improvement step. Such a scheme has been used for example in [FKL00] in the approximation of certain restricted instances of Max-Sat.

The remainder of this chapter is organized as follows. In Section 4.2 we illustrate the general approach (mentioned above) on a very simple algorithm for the Max-Cut problem. Namely, we consider enhancing the trivial random algorithm which, given a graph $G = (V, E)$, constructs a partition $(U, V \setminus U)$ by independently choosing each vertex to be in U with probability $1/2$. In our main Section 4.3, we continue to study the Max-Cut problem and consider the [GW95] algorithm discussed above. In Section 4.3.1 we consider graphs of maximum degree three, and in Section 4.3.2 we present our results for graphs of maximum degree Δ. Finally, in Section 4.4 we briefly discuss how our results can be extended to graphs with edge weights.

4.2 The trivial algorithm for Max-Cut

Let $G = (V, E)$ be a graph of maximum degree Δ where $V = \{1, \dots, n\}$. For any set $U \subseteq V$ let $w(U)$ be the number of edges with one end point in U and another in $V \setminus U$. Denote the total number of edges by $|E|$; and the number of edges cut by the optimal partition of G as $Opt(G)$. $w(U)$ is referred to as the *value* or *weight* of the cut $(U, V \setminus U)$. For any natural number n we denote the set $\{1, \dots, n\}$ by $[n]$.

Consider the following trivial random algorithm for the Max-Cut problem on G which constructs a partition $(U, V \setminus U)$ by independently choosing each vertex to be in U with probability $1/2$. It is clear that the expected number of edges cut by the resulting partition is exactly half the total number of edges in G, thus this trivial algorithm has an expected approximation ratio of $1/2$. We now improve this algorithm by adding a simple local improvement step which moves vertices from one side of the partition to the other.

Consider a random partition $(U, V \setminus U)$ of the vertices of G obtained by the above trivial algorithm. We say that a vertex v is *misplaced* in $(U, V \setminus U)$ if it has more neighbors on its side of the partition than on the opposite side. Moving such a vertex from one side of the partition to the other will increase the value of the cut. Roughly speaking we will prove that in a random partition of the vertex set V the expected fraction of misplaced vertices is constant. Combining this with the fact that moving each misplaced vertex may affect at most Δ other vertices, we conclude that the expected contribution of the additional step to the trivial algorithm is a constant fraction of the total number of edges in E (assuming Δ is constant). This yields an approximation algorithm with an expected approximation ratio strictly greater than $1/2$ by some constant. Details follow.

We start by computing the probability that a vertex v will be misplaced in a random partition. Consider all neighbors of v. With probability at least $1/2$, the random partition partitions them into two sets of unequal size. Thereafter, with probability at least $1/2$, the vertex v is placed on the same side as the majority of its neighbors. Hence the probability that v is misplaced is at least $1/4$, and the expected number of misplaced vertices is at least $n/4$. Moving a misplaced vertex v from one side of the partition to the other contributes at least one edge to the partition. Such a movement can reduce the number of misplaced vertices by at most $\Delta + 1$, because only v and its neighbors are affected. We conclude that enhancing the trivial random algorithm described above by adding a step which moves misplaced vertices from one side of the partition to the other until no such vertices are left will achieve a cut of expected weight at least

$$\frac{|E|}{2} + \frac{n}{4(\Delta+1)} \geq |E|\left(\frac{1}{2} + \frac{1}{2\Delta(\Delta+1)}\right) \geq Opt(G)\left(\frac{1}{2} + \frac{1}{2\Delta(\Delta+1)}\right),$$

thus yielding an improved expected approximation ratio of $\frac{1}{2} + \frac{1}{2\Delta(\Delta+1)}$. We remark that the term $\frac{1}{2\Delta(\Delta+1)}$ can be significantly improved by tighter analysis, but we wish to keep the analysis simple so as to clearly illustrate the method.

4.3 Enhancing the [GW95] algorithm for Max-Cut

In the following section, we consider enhancing the GW algorithm [GW95] for Max-Cut. In Section 4.3.1 we consider graphs G of maximum degree 3, and in Section 4.3.2 we consider graphs G of maximum degree Δ. We will use the notation presented in Section 4.2. Furthermore, α will represent the expected approximation ratio of the GW algorithm for Max-Cut [GW95]. Throughout this section, we assume that there are no weights associated with the edge set E. Our results generalize to the case in which G has edge weights, we briefly discuss this in Section 4.4.

The GW algorithm As presented in Section 1.2, the Max-Cut problem on a graph $G = (V, E)$ with $|V| = n$ can be represented as a quadratic integer program:

(QI-MC)	Maximize	$\sum_{e_{ij}\in E} \frac{1-x_i\cdot x_j}{2}$	
	subject to:		
		$x_i \in \{-1, 1\}$	for $i \in [n]$

The requirement $x_i \in \{-1, 1\}$ can be relaxed by representing each variable x_i by a unit n-dimensional vector $v_i \in S^{n-1}$ (here S^{n-1} is the unit sphere) and the multiplication $x_i \cdot x_j$ by the inner product $\langle v_i, v_j\rangle$.

(SDP-MC)	Maximize	$\sum_{e_{ij}\in E} \frac{1-\langle v_i,v_j\rangle}{2}$	
	subject to:		
		$v_i \in S^{n-1}$	for $i \in [n]$

The algorithm suggested in [GW95] for the Max-Cut problem has two steps. In the first step the relaxation *(SDP-MC)* is solved. A solution to *(SDP-MC)* is a set of unit vectors $v_1, \ldots, v_n$ in R^n, rather than a cut of G. In the second step a cut $(U, V \setminus U)$ of G is obtained by *rounding* the set of optimal vectors $v_1, \ldots, v_n$ using the *random hyperplane* rounding technique. That is, a random vector $r \in S^{n-1}$ is chosen, and the set U is defined to be the set of vertices i with corresponding vectors v_i that have a positive inner product with r. The vector r can be viewed as the normal to a random hyperplane passing through the origin.

It is shown in [GW95] that the expected contribution of each edge $e_{ij} \in E$ to the cut $(U, V\setminus U)$ is $\frac{\theta_{ij}}{\pi}$, where θ_{ij} is the angle between the vectors v_i and v_j. As the contribution of each edge to the objective function of *(SDP-MC)* is $\frac{1-\cos\theta_{ij}}{2}$ we conclude that the expected approximation ratio achieved on each edge is $\frac{2\theta_{ij}}{\pi(1-\cos\theta_{ij})}$. This expected ratio is minimal when $\theta_{ij} = \theta_0 \simeq 2.3311$, and in this case obtains the value $\alpha \simeq 0.87856$. We conclude that the expected weight $w(U)$ of the partition $(U, V \setminus U)$ obtained by the random hyperplane rounding technique is at least $\alpha Opt(G)$, and can potentially be exactly $\alpha Opt(G)$ when for all edges e_{ij} we have that the angle θ_{ij} is θ_0 or zero.

4.3.1 Max-Cut on graphs of maximum degree 3

In this section we consider graphs G in which the degree of every vertex is bounded by 3. We enhance the GW algorithm, and show that on such graphs our enhanced algorithm obtains an approximation ratio of $r \geq 0.921$.

Our algorithm consists of two steps. In the first step an enhanced semidefinite relaxation of the Max-Cut problem on G is solved to obtain a set of n unit vectors $\{v_1, \ldots, v_n\}$. These vectors are then rounded into a cut $(U, V \setminus U)$ of G using the random hyperplane rounding technique of [GW95]. As mentioned in the Introduction, it may be the case that there are misplaced vertices in the cut $(U, V \setminus U)$ (*i.e.* vertices with more neighbors on their side of the cut than on the opposite side). In the second step of our algorithm, we move these vertices from one side to the other until no more misplaced vertices remain. This is done in a *greedy* manner. In what follows, we present the semidefinite relaxation used in the first step of our algorithm and define the greedy procedure used in the last step.

The semidefinite relaxation Let T be the set of triplets (i, j, k) in which i, j, k are vertices in V and both j and k are neighbors of i (for technical reasons we also assume $j < k$). The semidefinite relaxation we use in our algorithm is

(SDP-Cut)	Maximize	$\sum_{e_{ij} \in E} \frac{1 - \langle v_i, v_j \rangle}{2}$	
	subject to:		
	(1)	$v_i \in S^{n-1}$	for $i \in [n]$
	(2)	$\langle v_i, v_j \rangle + \langle v_i, v_k \rangle + \langle v_j, v_k \rangle \geq -1$	
		$\langle v_i, v_j \rangle - \langle v_i, v_k \rangle - \langle v_j, v_k \rangle \geq -1$	for all $i, j, k \in [n]$
	(3)	$\langle v_i, v_j \rangle + \langle v_i, v_k \rangle + \langle v_j, v_k \rangle = -1$	for all $(i, j, k) \in T$.

Notice that we have added two families of constraints to the original relaxation of [GW95]. The triangle constraints *(2)*, and the *tight* triangle constraints *(3)*. The triangle constraints are valid when the vectors $\{v_1, \ldots, v_n\}$ are restricted to be one dimensional and thus can be added to *(SDP-Cut)*. It is shown in [GW95] that the original semidefinite relaxation *(SDP-MC)* (without triangle constraints) has an integrality gap of at least $1/0.885$ even on 2 regular graphs (the 5 cycle). Hence these additional triangle constraints are crucial to our analysis which results in an approximation ratio of 0.921 on such graphs.

For the validity of the tight triangle constraints consider an **optimal** partition $(U, V \setminus U)$ of $G = (V, E)$. As G has maximum degree three, for every vertex v it cannot be the case that v and two of its neighbors lie on the same side of the partition (*i.e.* v is misplaced). In such a case moving v to the other side of the partition would increase the weight of the partition, which is a contradiction to its optimality. Hence we may add a corresponding constraint to *(SDP-Cut)* which rules out the possibility of misplaced vertices, when the corresponding vectors of *(SDP-Cut)* are restricted to be one dimensional. This constraint is the tight triangle constraint.

The greedy phase As mentioned earlier after the first step of our algorithm it might be the case that for some triplet $(i, j, k) \in T$, defined above, the vertices i, j and k lie on the same side of the partition. In such a case, moving the misplaced vertex i to the other side of the partition will increase its weight. We denote such a triplet $(i, j, k) \in T$ in which i is misplaced as a *good* triplet.

Given a partition $(U, V \setminus U)$ we are interested in moving all misplaced vertices until none are left. In general at each step of this greedy process we could decide to move the one misplaced vertex that increases the weight of the partition by most. But as moving one misplaced vertex affects other vertices, we are also interested that the vertex moved does not *destroy* many good triplets (a good triplet is destroyed if it is no longer good). In order to combine these two interests, at each stage of our greedy process we move the vertex for which the ratio between the weight added to the partition by moving it from one side of the partition to the other and the number of good triplets destroyed by this act, is maximal. We denote the resulting algorithm as A_{cut}.

Algorithm A_{cut}:

1. Solve *(SDP-Cut)* to obtain an optimal vector configuration $\{v_1, \ldots, v_n\}$ of value Z. Round the vector configuration using the random hyperplane rounding technique from [GW95]. Denote the partition obtained by $(U, V \setminus U)$.

2. Greedily move misplaced vertices from one side of the partition to the other according to the procedure above.

Theorem 4.3.1 *A_{cut} has an expected approximation ratio of $r \geq 0.921$.*

Proof: First we note that w.l.o.g. we may assume that G does not have any vertices of degree 1. Otherwise we may run algorithm A_{cut} on the graph $\hat{G}$ obtained by iteratively removing all vertices of degree one in G. It can be seen that an approximation algorithm with ratio r on $\hat{G}$ yields an approximation algorithm with ratio at least r on G.

Assume G is as above, let $\{v_1, \ldots, v_n\}$ be the optimal vector configuration, and let $(U, V \setminus U)$ be the partition after step *(1)* of algorithm A_{cut}. Let $W = w(U)$ be the weight of this partition. Recall, using the analysis of [GW95] that in expectation $W \geq \alpha Z \geq \alpha Opt(G)$. Furthermore, the expected value of W is exactly $\alpha Opt(G)$ only if for each edge e_{ij} in G the angle between its corresponding vectors v_i and v_j is either 0 or $\theta_0 \simeq 2.3311$. Our main observation is the fact that in both these *worst* cases (and in the case in which we are *close* to being in the worst case) there is some constant probability that a vertex and two of its neighbors lie on the same side of the partition $(U, V \setminus U)$. In such cases moving the vertex from one side of the partition to the other will increase the weight of the partition.

Geometrically speaking, one can view our observation in the following way. Let v be some vertex in V, and y_1, y_2 be two of its neighbors. Denote the corresponding vectors in the optimal vector configuration as v, y_1, y_2 respectively. Assume that the angles between the vectors v and y_1, v and y_2 are exactly θ_0 (a similar analysis can be given for the case in which these angles are close to θ_0), then by constraint *(3)* of *(SDP-Cut)*, it **cannot** be the case that all three vectors lie on the same plane. Furthermore, this constraint implies that the angle between the vector v and the plane containing the vectors y_1 and y_2 is at least 0.57, which in turn implies that with probability at least 0.07 the vertex v will be misplaced (after the randomized rounding).

The upcoming lemmas extend this observation. We use a refinement of the set T. For each edge e_{ij} let d_{ij} be the number of triplets in T in which e_{ij} appears. It can be seen that in a graph of minimum degree 2 and maximum degree 3, $d_{ij} = 2$ if the degree of v_i and v_j are 2, $d_{ij} = 3$ if the degree of v_i differs from the degree of v_j, and $d_{ij} = 4$ otherwise. We refine the set T of triplets in G as follows. For $l = 4, 5, \ldots, 8$ denote the set of triplets $(i, j, k) \in T$ in which $d_{ij} + d_{ik} = l$ by T_l. Given a partition $(U, V \setminus U)$ we denote by S_l the number of good triplets in T_l.

Lemma 4.3.2 *Let $(U, V \setminus U)$ be some partition in G of weight W. Let T, T_l, and S_l for $l = 4, \ldots, 8$ be defined as above. Executing step (2) of algorithm A_{cut} on the partition $(U, V \setminus U)$ will yield a new partition of weight at least*

$$W + \frac{2}{3}S_4 + \frac{1}{2}S_5 + \frac{2}{5}S_6 + \frac{3}{8}S_7 + \frac{1}{3}S_8$$

Proof: Roughly speaking, we define the contribution of each triplet (i, j, k) that is good in $(U, V \setminus U)$ as the number of edges it adds to the partition the moment it is destroyed by the greedy phase of our algorithm. Let τ be one of the vertices i, j, or k. If (i, j, k) is destroyed by the movement of the vertex τ, and by this act p edges are added to the partition and q triplets (including (i, j, k)) are destroyed, we fix the contribution of the triplet (i, j, k) to be p/q. Hence, due to the nature of our greedy phase, we may bound the contribution of the good triplet (i, j, k) from below by computing the ratio between the number of edges added to the partition and the number of triplets destroyed by the act of moving any one of the vertices i, j, or k from one side of the partition to the other. By finding such a lower bound for each good triplet in T_l $(l = 4, \ldots, 8)$ we conclude our assertion.

Note that for any good triplet (i, j, k) and any vertex τ as above, the contribution of moving τ from one side of the partition to the other is at least one edge, while this act will destroy at most 9 triplets. It follows that we may trivially claim that executing step *(2)* of algorithm A_{cut} on the given partition $(U, V \setminus U)$ will yield a new partition of weight at least

$$W + \frac{1}{9}(S_4 + S_5 + S_6 + S_7 + S_8).$$

The following case analysis refines this trivial analysis and provides a full proof of the asserted lemma.

Case 1: Let (i, j, k) be a good triplet in T_4, *i.e.* the vertices i, j and k are of degree 2, and all lie on the same side of the partition $(U, V \setminus U)$. As the greedy step *(2)* of A_{cut} continues as long as there is some misplaced vertex, the triplet (i, j, k) will be destroyed sometime during our greedy procedure. Denote the vertex moved when (i, j, k) is destroyed by τ (τ is either i, j or k). Let p be the number of edges added to the partition when (i, j, k) is destroyed, and q be the number of good triplets destroyed along with (i, j, k).

As mentioned above the degrees of i, j and k are 2. Hence, if (i, j, k) is a good triplet, then by moving the vertex i we destroy at most 3 triplets and increase the weight of the partition by exactly 2 edges. Due to the nature of our greedy phase, we conclude that by moving τ from one side of the partition to the other, the ratio between the number of edges added to the partition and the number of triplets destroyed is at least 2/3. We conclude that when (i, j, k) is destroyed it contributes at least 2/3 to the partition.

Case 2: Let (i, j, k) be a good triplet in T_5; using the same line of analysis displayed in the previous case, it is enough to analyze the movement of the vertex i from one side of the partition to the other. Assume that the vertex j is of degree 3 and vertices i, k are of degree 2 (it cannot be the case that i is of degree 3). In Figure 4.1 a schematic view of our case is presented. Solid lines represent edges that **preserve** sides of the partition (for instance the vertices i, j and k lie on the same side of the partition), dotted lines represent edges that are **not known to preserve** sides of the partition, and bold lines represent edges that are known **not to preserve** sides of the partition. It can be seen that moving i from one side of the partition to the other we gain exactly 2 edges to the partition, but destroy at most 4 triplets. We conclude that (i, j, k) contributes at least 1/2 edges to the partition.

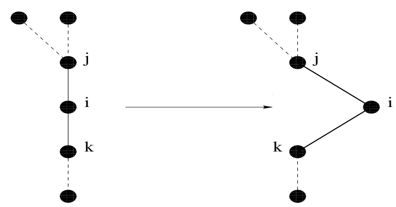

Figure 4.1: Case 2.

Case 3: Let (i, j, k) be a good triplet in T_6. There are two possibilities; either the vertex i is of degree 2 and the vertices j, k are of degree 3, or the vertex i is of degree 3 and j, k are of degree 2. Both cases are presented in Figure 4.2. In the first case *(a)*, moving i to the other side of the partition

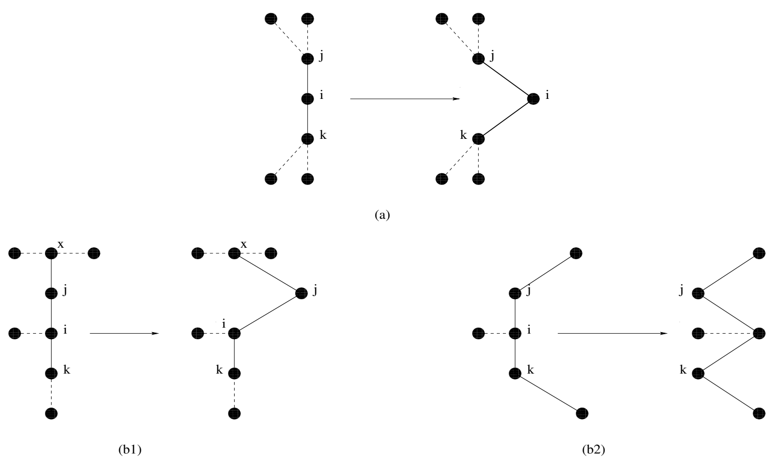

Figure 4.2: Case 3.

increases the weight of the partition by 2 edges and destroys at most 5 triplets. Hence, if we are in this case then (i, j, k) contributes 2/5 when it is destroyed.

In the second case we consider two sub-cases: *(b1)* the case in which one of the vertices j and k is on the same side as its additional neighbor x, *(b2)* the case in which the additional neighbors of j and k are both on the opposite side of the partition. In the first case we have that moving j from one side of the partition to the other will contribute 2 edges to the cut and destroy at most 5 triplets, yielding a ratio of 2/5, and in the second we obtain a ratio of 3/5 by moving i (note that in the case *(b2)* one must fix the side of the third neighbor of i and only then analyze); thus we conclude our assertion.

Case 4: Let (i, j, k) be a good triplet in T_7. It must be the case that the vertex i is of degree 3 and the vertices j, k are of different degree (we assume that j is of degree 2). We consider three cases that are presented in Figure 4.3. In the first case *(a)* all three neighbors of i are on the same side as i. In

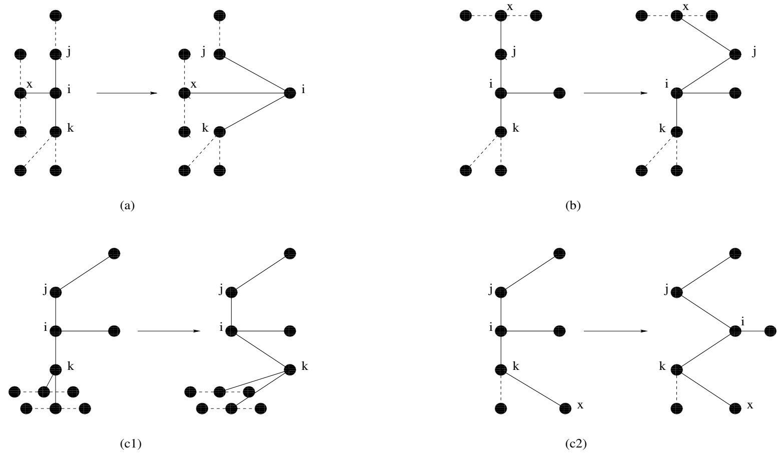

Figure 4.3: Case 4.

such a case by moving i one can see that a ratio of at least 3/8 is achieved. In the second and third cases we assume that the additional neighbor of i is on the opposite side of the partition.

In the second case *(b)* we assume that the additional neighbor x of j is on the same side as j. In this case moving j we achieve a ratio of at least 2/4.

At last in the third case *(c1)* and *(c2)* we assume that the additional neighbor of j is also on the opposite side of the partition, and we consider two situations: the first when all additional neighbors of k are on the same side of the partition as k, and the second when there is some neighbor x of k on a different side. It can be seen that moving k in the first case results in a ratio of at least 3/8, and moving i in the second results in a ratio of at least 1/2.

Case 5: Let (i, j, k) be a good triplet in T_8. It must be the case that all three vertices are of degree 3. In Figure 4.4 three sub-cases are considered. In case *(a)* all three neighbors of i are on the same

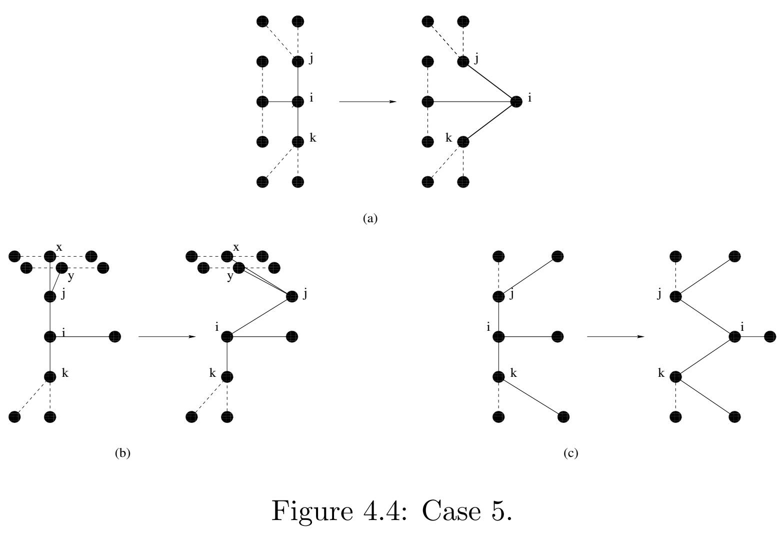

Figure 4.4: Case 5.

side of the partition as i; in this case moving i we obtain a ratio of 3/9. In the remaining cases *(b), (c)* we assume that the additional neighbor of i is on the opposite side of the partition. In case *(b)* we assume that for one of the vertices j or k, all three of its neighbors are on the same side of the partition as it is (in Figure 4.4 we assume the above for j). In such a case moving j results in a ratio $\geq 3/8$. In the last case *(c)* both i, j, and k have at least one neighbor on an opposite side. In this case moving i we have a ratio $\geq 1/3$. $\Box$

Corollary 4.3.3 *Let W be the weight of the partition $(U, V \setminus U)$ obtained by step (2) of algorithm A_{Cut}, and let S_l ($l = 4, \ldots, 8$) be the number of good triplets in $(U, V \setminus U)$ according to the above definition. The expected weight of the partition received by algorithm A_{Cut} is:*

$$\begin{aligned} E[w(A_{Cut})] &= E\left[W + \frac{2}{3}S_4 + \frac{1}{2}S_5 + \frac{2}{5}S_6 + \frac{3}{8}S_7 + \frac{1}{3}S_8\right] = E[W] + \sum_{l=4}^{8} \alpha_l E[S_l] \\ &= \sum_{e_{ij}} \Pr(e_{ij} \text{ is cut}) + \sum_{l=4}^{8} \sum_{(i,j,k)\in T_l} \alpha_l \Pr(v_i, v_j, v_k \text{ are not separated}) \\ &= \sum_{l=4}^{8} \sum_{(i,j,k)\in T_l} \frac{\Pr(e_{ij} \text{ is cut})}{d_{ij}} + \frac{\Pr(e_{ik} \text{ is cut})}{d_{ik}} + \alpha_l \Pr(v_i, v_j, v_k \text{ are not separated}), \end{aligned}$$

where $(\alpha_4, \alpha_5, \ldots, \alpha_8) = (2/3, 1/2, 2/5, 3/8, 1/3)$.

Proof: All probabilities above are taken over random hyperplanes cutting the unit sphere of R^n. The second and third equalities are due to linearity of expectation, and the fourth is due to the fact that each edge e_{ij} is in exactly d_{ij} triplets in T. $\Box$

Lemma 4.3.4 *Let $l \in \{4, \ldots, 8\}$. For every triplet $(i, j, k) \in T_l$ we have that:*

$$\frac{\Pr(e_{ij} \text{ is cut})}{d_{ij}} + \frac{\Pr(e_{ik} \text{ is cut})}{d_{ik}} + \alpha_l \Pr(v_i, v_j, v_k \text{ are not separated}) \geq r\left(\frac{1 - \langle v_i, v_j\rangle}{2d_{ij}} + \frac{1 - \langle v_i, v_k\rangle}{2d_{ik}}\right)$$

for $r = 0.921$.

Proof: Let $l = d_{ij} + d_{ik}$. Define the following functions:

$$\begin{aligned} f(\theta_{ij}, \theta_{ik}) &= \frac{\theta_{ij}}{d_{ij}\pi} + \frac{\theta_{ik}}{d_{ik}\pi} + \alpha_l\left(1 - \frac{1}{2\pi}(\theta_{ij} + \theta_{ik} + \arccos(-1 - \cos(\theta_{ij}) - \cos(\theta_{ik})))\right), \\ g(\theta_{ij}, \theta_{ik}) &= \frac{1 - \cos(\theta_{ij})}{2d_{ij}} + \frac{1 - \cos(\theta_{ik})}{2d_{ik}}, \\ h(\theta_{ij}, \theta_{ik}) &= \frac{f(\theta_{ij}, \theta_{ik})}{g(\theta_{ij}, \theta_{ik})}, \end{aligned}$$

where θ_{ij} (θ_{ik}) is the angle between v_i and v_j (v_i and v_k).

In [GW95] it is shown that the probability that three unit vectors v_i, v_j, and v_k are not separated by the random hyperplane rounding technique is $1 - \frac{1}{2\pi}(\theta_{ij} + \theta_{ik} + \theta_{jk})$. Furthermore recall that for every triplet $(i, j, k) \in T$ it is the case that $\cos(\theta_{ij}) + \cos(\theta_{ik}) + \cos(\theta_{jk}) = -1$ (constraint *(3)* of *(SDP-Cut)*). We conclude that the function $f(\theta_{ij}, \theta_{ik})$ represents the expected contribution of the triplet (i, j, k) to the cut. Similarly, the function $g(\theta_{ij}, \theta_{ik})$ represents the contribution of each triplet (i, j, k) to the objective function of *(SDP-Cut)*.

In order to prove the above claim it is suffice to show that $h(\theta_{ij}, \theta_{ik}) > r$ in the range $\theta_{ij} \in [0, \pi]$, $\theta_{ik} \in [\pi - \theta_{ij}, \pi]$ (the tight triangle constraints added to *(SDP-Cut)* imply that $\cos(\theta_{ij}) + \cos(\theta_{ik}) \leq 0$). By evaluating the value of h over a number of points in the above range and bounding the derivative of h over the whole range our lemma is proven (we use the facts that $d_{ij}, d_{ik} \in \{2, 3, 4\}$ and $l = d_{ij} + d_{ik} \in \{4, \ldots, 8\}$). □

By Corollary 4.3.3 and Lemma 4.3.4 we have that the expected weight of the partition obtained by algorithm A_{Cut} is

$$E[w(A_{Cut})] \geq r \sum_{l=4}^{8} \sum_{(i,j,k) \in T_l} \left(\frac{1 - \langle v_i, v_j \rangle}{2d_{ij}} + \frac{1 - \langle v_i, v_k \rangle}{2d_{ik}} \right) = rZ \geq rOpt(G),$$

for $r = 0.921$.

This completes the proof of Theorem 4.3.1. The same line of analysis yields a 0.924 approximation ratio on 3 regular graphs. □

4.3.2 Max-Cut on graphs of maximum degree Δ

In the following section we consider the Max-Cut problem on graphs of maximum degree Δ. We present a slightly different algorithm and analysis than the ones presented in the case where $\Delta = 3$, and achieve an improved approximation ratio of $\alpha + \Omega(1/\Delta^4)$.

As in the previous section, the starting point of our enhanced algorithm is the following semidefinite relaxation of the Max-Cut problem on a given graph $G = (V, E)$.

(SDP-Cut)	Maximize	$\sum_{e_{ij} \in E} \frac{1 - \langle v_i, v_j \rangle}{2}$	
	subject to:		
	(1)	$v_i \in S^{n-1}$	for $i \in [n]$
	(2)	$\langle v_i, v_j \rangle + \langle v_i, v_k \rangle + \langle v_j, v_k \rangle \geq -1$	
		$\langle v_i, v_j \rangle - \langle v_i, v_k \rangle - \langle v_j, v_k \rangle \geq -1$	for all $i, j, k \in [n]$

Note that constraint *(3)* used in the previous section is no longer valid. As earlier, solving *(SDP-Cut)* on a given graph G, we obtain a set of n vectors corresponding to the vertices of G. We will concentrate on the case in which for all edges e_{ij} in G the angle between the corresponding vectors v_i and v_j is equal to $\theta_0 \simeq 2.3311$, for in this case the random hyperplane technique will yield a partition $(U, V \setminus U)$ of expected weight no better than α times the value of *(SDP-Cut)*. [2]

In this *worst* case, we will show that there is some probability (depending on Δ alone) that a vertex and more than half its neighbors lie on the same side of the partition obtained by the random hyperplane rounding technique. As earlier, we denote such vertices as misplaced, and by adding an additional step which moves misplaced vertices from one side of the partition to the other, we are able to achieve an improved approximation ratio. In Theorem 4.3.7 we show that improving the approximation ratio for this worst case scenario suffices to obtain an improved approximation ratio when the angles θ_{ij} corresponding to edges e_{ij} are not necessarily all of value θ_0.

Consider the following algorithm for the Max-Cut problem on graphs with bounded degree Δ.

[2] As noted earlier, there are other worst cases for the [GW95] algorithm, namely when for some edges e_{ij} the angles between the corresponding vectors is zero. In Chapter 3 it is shown how to deal with these cases by the use of an improved rounding technique. We deal with such cases in a different manner which will be explained in the proof of Theorem 4.3.7.

Algorithm A_{Cut}:

1. Solve the semidefinite relaxation *(SDP-Cut)* and round the resulting optimal vector configuration using the random hyperplane rounding technique. Denote the partition obtained by $(U, V \setminus U)$.

2. Arbitrarily move misplaced vertices in $(U, V \setminus U)$ from one side of the partition to the other until no such vertices are left.

Let $\{v_1, \ldots, v_n\}$ be the vector configuration obtained after solving the semidefinite relaxation *(SDP-Cut)* on a given graph G. Let $(U, V \setminus U)$ be the partition obtained after rounding this vector configuration using the random hyperplane rounding technique ([GW95]). Define an edge e_{ij} to be *bad* if the inner product $\langle v_i, v_j \rangle$ is equal to $\cos(\theta_0) \simeq -0.688$ (*i.e.* $\theta_{ij} = \theta_0$). Define a vertex i to be *all bad* if for all neighbors j of i we have that the edge e_{ij} is bad. Denote by $N(i)$ the set of vertices adjacent to i. In what follows, we show that an 'all bad' vertex will be misplaced in $(U, V \setminus U)$ with probability $\Omega(1/\Delta^2)$. We will need the following lemma.

Lemma 4.3.5 *Let v be some vector in R^n of norm $\|v\|$, and let $r = (r_1, \ldots, r_n) \in R^n$ be the normal of a random hyperplane in R^n (i.e. each r_i has standard normal distribution $N(0,1)$). The size of the projection of v on r can be bounded as follows:*

$$\Pr[|\langle v, r \rangle| \geq \varepsilon \|v\|] = \Pr[|r_1| \geq \varepsilon] \geq 1 - \varepsilon,$$

$$\Pr[\langle v, r \rangle \in [0, \varepsilon]\|v\|] = \Pr[r_1 \in [0, \varepsilon]] \in [\varepsilon/8, \varepsilon/2],$$

where $\varepsilon \in [0, 1]$.

Proof: As the distribution of r is spherically symmetric ([Ren70] Theorem **IV**.16.3), we may assume that v is the vector $(\|v\|, 0, \ldots, 0)$, thus $\langle v, r \rangle$ is exactly $r_1 \|v\|$. □

Lemma 4.3.6 *Let $(U, V \setminus U)$ be the partition obtained after step (1) of algorithm A_{Cut}. Let i be an* **all bad** *vertex of degree d. With probability at least $\Omega(1/d^2)$ it is the case that* **more** *than half of the vertices in $N(i)$ lie on the same side of the partition $(U, V \setminus U)$ as the vertex i.*

Proof: Let $N(i) = \{u_1, \ldots, u_d\}$. Denote the vector v_i corresponding to the vertex i as v. Recall that $N(i)$ is defined to be a set of vertices adjacent to the vertex i. To simplify our notation, we would like to use the same notation for the set of vectors corresponding to the vertices of $N(i)$. Namely we denote the vectors corresponding to each vertex u_i in $N(i)$ as u_i, and the set of vectors $\{u_1, \ldots, u_d\}$ as $N(i)$. In what follows, it will be clear from the context whether we are referring to the vertex u_i or its corresponding vector.

As the vectors $v, \{u_1, \ldots, u_d\}$ lie in a common $d+1$ dimensional space, the following representation of $v, \{u_1, \ldots, u_d\}$ may be assumed without loss of generality. Let $v = (1, 0, \ldots, 0) \in R^{d+1}$ and $u_j = (\alpha_j, \beta_j, \gamma_j)$ for $j \in [d]$. Where $\alpha_j \in R$, $\beta_j \in R$, $\gamma_j \in R^{d-1}$, and α_j, β_j, γ_j are vectors in mutually orthogonal subspaces. Furthermore, we may assume for the vector u_1 that $\gamma_1 = 0$. Let $r = (r_1, \ldots, r_{d+1})$ be a $d+1$ dimensional random variable representing the normal to a random hyperplane, *i.e* every r_i is an independent standard normal random variable. Let r be defined as $(\alpha_r, \beta_r, \gamma_r)$ where $\alpha_r = r_1$, $\beta_r = r_2$, and $\gamma_r = (r_3, \ldots, r_{d+1})$.

In general we intend to prove that with probability $\delta = \Omega(1/d^2)$ (over r), the vector v and more than half its neighbors lie on the same side of the hyperplane corresponding to r. Let δ_1 be the probability that the vector v lies very *close* but *above* the random hyperplane corresponding to r, *i.e.* the inner product $\langle v, r\rangle$ is small but positive. Note that this probability is dependent on r_1 alone, and corresponds to a positive r_1 of low magnitude. Conditioning on such r (*i.e* r_1), we are interested in the probability that more than half of the vectors in $N(i)$ also lie above the hyperplane corresponding to r. As r_1 is small, this is roughly the probability that more than half of the projected vectors $(\beta_j, \gamma_j) \in R^d$ for $j \in [d]$ will be above the random hyperplane corresponding to the random vector (r_2, r_3).

If the number of vectors in $N(i)$ is odd (*i.e.* d is odd), then the above will happen with probability $1/2$, and we may conclude that δ is roughly $\delta_1/2$. Otherwise we condition on an additional projection. Let δ_2 be the probability that the vector u_1 is also very close but above the random hyperplane corresponding to a random vector r. As before this probability is dependent on r_1 and r_2 alone (recall that $u_1 = (\alpha_1, \beta_1, 0)$). Now conditioning the random vector r on r_1 and r_2 (in order to assure that both v and u_1 lie close to and above the random hyperplane corresponding to r), we are interested in the probability that more than half of the remaining vectors in $N(i)$ also lie above the hyperplane corresponding to r. As above, due to the fact that r_1 and r_2 are of small magnitude this probability is roughly the probability that more than half of the projected vectors $\gamma_j \in R^{d-1}$ for $j \in \{2, \ldots, d\}$ will be above the random hyperplane corresponding to the random vector r_3. We are now in a case similar to the case in which d is odd, and conclude that δ is roughly $\delta_1\delta_2/2$. Detailed proof follows.

The vertex i is **all bad**, thus for all $j \in [d]$ we have that the inner product $\langle v, u_j\rangle = \alpha_j$ equals $\cos(\theta_0) \simeq -0.688$, and that the vector $(\beta_j, \gamma_j) \in R^d$ satisfies $\|(\beta_j, \gamma_j)\| \simeq 0.725$ (for u_1 this implies that $\beta_1 \simeq 0.725$). For $i, j \in [d]$ the triangle constraints $(\langle v, u_i\rangle + \langle v, u_j\rangle + \langle u_i, u_j\rangle \geq -1)$ of *(SDP-Cut)* yield $\langle u_i, u_j\rangle = \alpha_i\alpha_j + \beta_i\beta_j + \langle \gamma_i, \gamma_j\rangle \geq 0.376$. Hence $\beta_i\beta_j + \langle \gamma_i, \gamma_j\rangle \geq -0.097$ meaning that the angle between the vectors $(\beta_i, \gamma_i), (\beta_j, \gamma_j) \in R^d$ is at most 1.763 (bounded away from π).

Let $\varepsilon > 0$ be some small value which will be fixed later in the proof. By Lemma 4.3.5 we have that with probability at least $\varepsilon/16$ (over $\alpha_r \in R$) it is the case that $\langle v, r\rangle = \alpha_r \in [\varepsilon/8, \varepsilon]$, and with probability at least $\varepsilon/2$ (over $\beta_r \in R$) we have that $\beta_r \in [\varepsilon, 8\varepsilon]$ (we assume that $\varepsilon < 1/8$). We conclude that with probability at least $\frac{\varepsilon^2}{2^5}$ over the random vector r we have that both v and u_1 lie above the hyperplane corresponding to r.

Furthermore, considering a vector γ_j, we conclude using Lemma 4.3.5 that with probability at least $1 - \frac{1}{4d}$ (over γ_r) the value of $|\langle \gamma_j, \gamma_r\rangle|$ is at least $\frac{1}{4d}\|\gamma_j\|$. Hence with probability at least 3/4 (over γ_r) we have for all vectors γ_j that $|\langle \gamma_j, \gamma_r\rangle| \geq \frac{1}{4d}\|\gamma_j\|$.

Finally, consider the probability over γ_r that **more** than half the inner products $\langle \gamma_j, \gamma_r\rangle$ are non-negative ($j \in [d]$). As described above, this probability is at least half, due to the fact that for any γ_r we have $\langle \gamma_1, \gamma_r\rangle = 0$. We conclude, that with probability 1/4 over γ_r, we are in the case in which $\langle \gamma_r, \gamma_j\rangle \geq \frac{1}{4d}\|\gamma_j\|$ for **more** than half of the vectors γ_j. Denote the set of these vectors as $\hat{N}(i)$. Note that $u_1 \in \hat{N}(i)$.

Let r be a random vector $r = (\alpha_r, \beta_r, \gamma_r)$ with $\alpha_r,\ \beta_r,\ \gamma_r$ as above (this happens with probability at least $\frac{\varepsilon^2}{2^7}$). We now claim that choosing an appropriate ε, we obtain for all vectors u_j in $\hat{N}(i)$ that the inner product $\langle u_j, r\rangle > 0$, as $|\hat{N}(i)| > d/2$ this completes the proof of our lemma.

Let $u_j = (\alpha_j, \beta_j, \gamma_j)$ be some vector in $\hat{N}(i)$. If $\beta_j \geq 0.7$ then we have that

$$\langle u_j, r\rangle = \alpha_j\alpha_r + \beta_j\beta_r + \langle \gamma_j, \gamma_r\rangle \geq (\beta_j - 0.688)\varepsilon + \frac{1}{4d}\|\gamma_j\| > 0.$$

Otherwise assume that for $u_j \in \hat{N}(i)$ we have $\beta_j < 0.7$. As we have shown earlier, $\|(\beta_j, \gamma_j)\| \simeq 0.725$ for all vectors in $N(i)$, thus we conclude that in this case $\|\gamma_j\|$ must be at least 1/8. Furthermore, for any vector $u_j \in N(i)$ the inner product $\langle u_1, u_j\rangle = \alpha_1\alpha_j + \beta_1\beta_j$ is at least 0.376, thus implying that β_j must be at least -0.14 (recall that $u_1 = (\alpha_1, \beta_1, 0)$). We conclude that

$$\langle u_j, r\rangle = \alpha_j\alpha_r + \beta_j\beta_r + \langle\gamma_j, \gamma_r\rangle \geq (-1.12 - 0.688)\varepsilon + \frac{1}{4d}\|\gamma_j\| > \frac{1}{2^5 d} - 2\varepsilon.$$

Setting ε to be less than $\frac{1}{2^6 d}$ the above inner product is positive.

All in all, the proof of our lemma is complete with $\delta = \frac{\varepsilon^2}{2^7} = \Omega(1/d^2)$. □

Theorem 4.3.7 *There exists a semidefinite based algorithm that for every $\Delta > 0$ approximates the Max-Cut problem on graphs with bounded degree Δ within an expected approximation ratio of $\alpha + \Omega(1/\Delta^4)$.*

Proof: Consider the optimal vector configuration $\{v_1, \ldots, v_n\}$, and the partition $(U, V \setminus U)$ achieved after step *(1)* of algorithm A_{Cut}. Let Z be the value of the above optimal vector configuration, and $w(U)$ the weight of the partition $(U, V \setminus U)$. Using the analysis of [GW95] it can be seen that the expected value of $w(U)$ is at least αZ. Define an edge e_{ij} to be *bad* if the inner product $\langle v_i, v_j\rangle$ is *close* to $\cos(\theta_0) \simeq -0.688$, *i.e.* $\langle v_i, v_j\rangle \in [-0.688 - 0.01, -0.688 + 0.01]$.

Let Z_{good} be the contribution of the edges in G which are good (*i.e.* not bad) to the value Z defined above, and let $\varepsilon > 0$ be some small constant that will be determined later. We consider two cases. If Z_{good} is greater than εZ, using the standard analysis of [GW95], we have that

$$E[w(U)] \geq \alpha(Z - Z_{good}) + \left(\alpha + \frac{1}{2^{12}}\right) Z_{good} \geq \left(\alpha + \frac{\varepsilon}{2^{12}}\right) Z.$$

Otherwise, the contribution of the good edges, Z_{good}, is less that εZ. In such a case consider the graph $\hat{G}$ obtained by removing these good edges from G. In $\hat{G}$ all vertices are all bad and have degree at most Δ. Furthermore, $\hat{G}$ has at least $\frac{(1-\varepsilon)Z}{\Delta}$ vertices of degree ≥ 1. Using Lemma 4.3.6, we have that each such vertex in $\hat{G}$ is misplaced in $(U, V \setminus U)$ (with respect to $\hat{G}$) with probability at least $\delta = \Omega(1/\Delta^2)$. Note that in Lemma 4.3.6 we dealt with the case in which e_{ij} was assumed to be bad if $\langle v_i, v_j\rangle$ was exactly $\cos(\theta_0) \simeq -0.688$. Our analysis in Lemma 4.3.6 was slack, thus it can be seen that we obtain the same results for the above definition of a bad edge as well. As each vertex moved may effect at most Δ other vertices, we conclude that running the greedy step of our algorithm A_{Cut} on $\hat{G}$ has an expected contribution of $\frac{(1-\varepsilon)Z}{2^{19}\Delta^3(\Delta+1)}$ edges. Therefore in such a case, we obtain a final cut of $\hat{G}$ (which is also a cut of G) of expected value at least $\left(\alpha + \frac{1}{2^{19}\Delta^3(\Delta+1)}\right)(1-\varepsilon)Z$. Setting ε to be $\frac{1}{2^{20}\Delta^3(\Delta+1)}$ our proof is complete. □

4.4 Discussion

Throughout our work, we have considered the unweighted version of the Max-Cut problem. Our analysis can be extended to graphs with edge weights as well. Namely, it can be shown that the Max-Cut problem on graphs with edge weights restricted to graphs of maximum degree Δ can be approximated within a ratio of $\alpha + \varepsilon$; where α is the expected approximation ratio of the Max-Cut algorithm of Goemans and Williamson presented in [GW95] and $\varepsilon > 0$ depends on Δ alone.

Our extended analysis is very similar to that presented throughout this chapter, although the results we obtain are weaker (that is the parameter ε above is smaller). In Section 4.2 such an extension can be obtained by considering for every misplaced vertex v, the ratio between the total weight of edges adjacent to v and the weight gained by moving the vertex v from one side of the cut to the other. In Section 4.3.1, when considering graphs without edge weights, a vertex i is said to be misplaced if any triplet of vertices (i, j, k) (where j and k are adjacent to i) lay on the same side of the cut. In the case in which we allow edge weights, a vertex i and two of its neighbors may be on the same side of the cut without i being misplaced, thus in this case we consider only the triplet (i, j, k) in which the edges e_{ij} and e_{ik} are the heaviest edges adjacent to i. In Section 4.3.2, for every vertex v regardless of its degree d, use the part of Lemma 4.3.6 addressing the case in which the degree d is even, and choose the neighbor u_1 of v to be such that the edge (v, u_1) is the heaviest edge adjacent to v.

In general, when considering graphs with edge weights, we stop the greedy placement step after a polynomial number of steps and not necessarily at a local optimum (as obtaining the local optimum for graphs of large maximum degree may take exponential time [SY91]). Our extended analysis holds under this additional restriction.

Chapter 5

Max-Bisection on regular graphs

5.1 Introduction

Let $G = (V, E)$ be an undirected graph. The maximum cut of G is a partition of the vertex set V into two arbitrarily sized sets (X, Y) such that the number of edges with one end point in X and the other in Y is maximal. The maximum bisection of $G = (V, E)$ is a partition of V into two equally sized sets (X, Y) (*i.e.* a *bisection* of V) that maximizes the number of edges between X and Y (here we assume that $|V|$ is even). Given a graph G, we say that an algorithm A approximates the Max-Cut (Max-Bisection) of G within an approximation ratio of r, if by running A on the graph G we obtain a partition (bisection) (X, Y) such that the number of edges between X and Y is at least r times the number of edges between the two sides of an optimal partition (bisection). Note that $r \leq 1$.

A graph G is said to be *regular* if all its vertices have equal degree. In this chapter we analyze the ratio between the maximum bisection of any given regular graph G, and its maximum cut. For general graphs it is not hard to see that this ratio can be arbitrarily close to 1/2. For regular graphs we show that this ratio is at least $\theta \simeq 0.9027$, and that there are infinitely many regular graphs which obtain a ratio arbitrarily close to θ. We then use this property to present a 0.795 approximation algorithm for the Max-Bisection problem restricted to regular graphs. In the case of three regular graphs our results imply an approximation ratio of 0.834.

Previous work To the best of our knowledge, the best known approximation ratios for Max-Cut and Max-Bisection on regular graphs are those obtained on general graphs. For Max-Cut this ratio is $\alpha \simeq 0.87856$ (presented in the work of Goemans and Williamson [GW95]). For Max-Biseciton, this ratio is 0.7028 (presented in Chapter 3 of this thesis). On the negative side, it has been shown by [Hås01] that approximating the Max-Cut and Max-Bisection problems on general graphs beyond the ratio of $\frac{16}{17}$ is NP-hard. Furthermore, [BK98] show that approximating Max-Cut (and thus also Max-Bisection) on 3-regular graphs beyond some explicit constant factor r strictly less than one is also NP-hard.

In this chapter we are interested in the restriction of Max-Bisection to regular graphs. As mentioned above, on such graphs we prove that the ratio between the maximum bisection and maximum cut is bounded by $\simeq 0.9028$. Moreover, our proof is constructive. That is, given a graph G and a partition (X, Y) that cuts w edges, one may efficiently find a bisection $(\hat{X}, \hat{Y})$ of G that cuts at least $\simeq 0.9027w$ edges. Using this observation, one may approximate the maximum bisection on regular graphs by first approximating the Max-Cut problem, and then converting the cut obtained to a bisection. Observe that such a scheme will result in an approximation ratio of $\simeq 0.87856 \cdot 0.9027 \simeq 0.793$. The approximation ratio for Max-Bisection on regular graphs obtained in this chapter slightly improves over this.

The remainder of this chapter is organized as follows. In Section 5.2 we analyze the ratio between the maximum bisection of any given regular graph G, and its maximum cut. In Section 5.3 we present and analyze an 0.795 approximation algorithm for the Max-Bisection problem restricted to regular graphs. In Section 5.4 we show that the ratio obtained in Section 5.2 is tight. Finally, in Section 5.5 we present the proof a few technical lemmas that appear throughout this chapter.

5.2 Max-Cut vs. Max-Bisection

Let $G = (V, E)$ be a Δ regular graph, where $V = \{v_1, \ldots, v_n\}$ and n is even. For every $X \subseteq V$ and $Y = V \setminus X$, let $w(X)$ be the number of edges cut by the partition (X, Y). We call $w(X)$ the *value* of the partition (X, Y). Let $Cut(G)$ be the value of the Max-Cut of G, and $Bis(G)$ be the value of the Max-Bisection of G. Given a partition (X, Y), the *out-degree* of any $v \in V$ is the number of neighbors that v has on the opposite side of the partition, and the *in-degree* is the number of neighbors that v has on its own side of the partition. We say that a partition (X, Y) is *locally optimal* if there is no vertex v with out-degree smaller than in-degree. Clearly if (X, Y) is not locally optimal one may obtain a partition of value strictly greater than $w(X)$ by moving a vertex with out-degree smaller than its in-degree from one side of the partition to the other.

Theorem 5.2.1 *Given a regular graph G and any partition (X, Y) of G with value $w(X)$, one can efficiently find a bisection $(\hat{X}, \hat{Y})$ of G of value $\theta w(X)$, where $\theta \simeq 0.9027$. Specifically we have that $Bis(G) \geq \theta Cut(G)$.*

Proof: Let $G = (V, E)$ be a Δ regular graph and (X, Y) be any cut in G of value $w(X)$. As a preliminary step we would like to turn the partition (X, Y) into a locally optimal one. This is done by iteratively moving vertices with out-degree strictly smaller than their in-degree from one side of the partition to the other. Note that the new partition is of value greater than or equal to the original partition. We keep our original notation and denote this improved partition as (X, Y). Let cn be the size of X where $c \geq 1/2$, and $x|E|$ be the value of the partition (X, Y), *i.e.* $w(X) = x|E|$.

From the definition of c, we conclude that $cn\Delta/2 \leq w(X) \leq \Delta(1-c)n$ as there are only $(1-c)n$ vertices in Y, and the partition (X, Y) is locally optimal. Hence $c \leq x \leq 2 - 2c$ (and $c \leq 2/3$). We would now like to move vertices from X to Y in order to obtain a bisection of G, *i.e.* a partition $(\hat{X}, \hat{Y})$ for which $|\hat{X}| = |\hat{Y}| = n/2$.

Let $\hat{X}$ be a random subset of X of size $n/2$, and $\hat{Y}$ be $V \setminus \hat{X}$. Denote the value of the bisection $(\hat{X}, \hat{Y})$ by $w(\hat{X})$. In the following, we compute the expected value of $w(\hat{X})$. Afterwards, using a greedy derandomization scheme, we show that a bisection of G with this expected value can be efficiently obtained from (X, Y).

Define the function $\theta(x)$ to be the expected ratio between the values $w(\hat{X})$ and $w(X)$ given $w(X) = x|E|$, and θ to be a lower bound of this ratio.

Lemma 5.2.2 *Let X and $\hat{X}$ be as above (in particular $w(X) = x|E|$). The expected ratio between $w(\hat{X})$ and $w(X)$ is at least $\theta(x) = \frac{1-x}{(2-x)x} + \frac{1}{(2-x)^2} \geq 0.9027$.*

Proof: Let $p = \frac{1}{2c}$. Recall that $\hat{X}$ is a random subset of X of size exactly $n/2$. The expected number of edges that were cut by the partition (X, Y) that are still cut in $(\hat{X}, \hat{Y})$ is $w(X)\binom{cn-1}{n/2-1} / \binom{cn}{n/2} = w(X)p = xp|E|$. As each vertex in G is of degree Δ, we have that the number of edges in the subgraph induced by X is $(\Delta cn - x|E|)/2 = (2c - x)|E|/2$. We conclude that the expected number of these edges cut by the partition $(\hat{X}, \hat{Y})$ is $(2c - x)|E|\binom{cn-2}{n/2-1} / \binom{cn}{n/2} \geq (2c - x)|E|p(1 - p)$.

Thus, given any partition (X, Y) of value $w(X)$ we may obtain a partition $(\hat{X}, \hat{Y})$ such that

$$E\left[\frac{w(\hat{X})}{w(X)}\right] \geq \frac{xp|E| + (2c-x)|E|p(1-p)}{x|E|} = \frac{x + (2c-x)\left(1 - \frac{1}{2c}\right)}{2cx} = \frac{2c-1}{2cx} + \frac{1}{4c^2}.$$

The expression above, as a function of x is decreasing, and is thus minimal when $x = 2 - 2c$. Replacing c in the above expression with $1 - x/2$ we conclude that

$$E\left[\frac{w(\hat{X})}{w(X)}\right] \geq \frac{1-x}{(2-x)x} + \frac{1}{(2-x)^2}.$$

Using basic computations that are presented in Section 5.5, it can be seen that the above expression obtains a minimal value of $\theta \simeq 0.9027$ when $x^* \simeq 0.7932$ ($c^* = 1 - x^*/2 \simeq 0.6034 \leq x^*$). $\square$

It is left to show that given a partition (X, Y) of value $w(X)$ a bisection $(\hat{X}, \hat{Y})$ with value at least the expected can be obtained efficiently. Consider the random process analyzed above, that fixes a bisection by setting $\hat{X}$ to be a random subset of X of size $n/2$. Note that choosing a random subset $\hat{X} \subseteq X$ of size $n/2$ is equivalent to removing a random subset of size $|X| - n/2$ from X and setting $\hat{X}$ to be the remaining vertices of X. Furthermore, the latter is equivalent to the random process which iteratively removes one vertex in X at a time, uniformly at random, until X is of size $n/2$. For $i \in [0, |X| - n/2]$ let (X_i, Y_i) be the partition obtained by this last random process at step i (the size of X_i is $|X| - i$), and set $(\hat{X}, \hat{Y})$ to be the bisection $(X_{|X|-n/2}, Y_{|X|-n/2})$. Let $E[w(X_i)]$ be the expected value of the partition (X_i, Y_i). Given the value $w(X_i)$, the value of $E[w(X_{i+1})]$ can be explicitly computed. A random vertex in X_i has expected out-degree of $d = w(X_i)/|X_i|$ and expected in-degree $\Delta - d$ thus

$$E[w(X_{i+1}) \mid w(X_i)] = w(X_i)\left(1 - \frac{2}{|X_i|}\right) + \Delta.$$

We conclude that

$$E[w(X_{i+1})] = E[E[w(X_{i+1}) \mid w(X_i)]] = E[w(X_i)]\left(1 - \frac{2}{|X_i|}\right) + \Delta.$$

In the following, we prove that the greedy process which at each step removes the vertex in X with lowest out-degree until the set X is of size exactly $n/2$, will obtain a bisection of value at least $E[w(X_i)]$ for each $i \in [0, |X| - n/2]$. As $\hat{X}$ is set to be X_i for $i = |X| - n/2$, this completes our proof.

Let (A_i, B_i) be the partition obtained by the above greedy process at step i, and let $w(A_i)$ be the value of the partition (A_i, B_i). Clearly $w(A_0) = E[w(X_0)] = w(X)$. Furthermore, using the fact that for each i there is a vertex in A_i with out-degree at most $w(A_i)/|A_i|$ we have that

$$w(A_{i+1}) \geq w(A_i)\left(1 - \frac{2}{|A_i|}\right) + \Delta.$$

We now conclude our proof using induction on i:

$$w(A_{i+1}) \geq w(A_i)\left(1 - \frac{2}{|A_i|}\right) + \Delta \geq E[w(X_i)]\left(1 - \frac{2}{|X_i|}\right) + \Delta = E[w(X_{i+1})].$$

□

In Section 5.4 we show that our result is tight, namely:

Proposition 5.2.3 *There are infinitely many regular graphs G for which the ratio between $Bis(G)$ and $Cut(G)$ is arbitrarily close to $\theta \simeq 0.9027$.*

5.3 Approximating Max-Bisection on regular graphs

As mentioned in Chapter 1, Goemans and Williamson [GW95] present a $\alpha \simeq 0.87856$ approximation algorithm based on semidefinite programming for the general Max-Cut problem. Their work is extended in [FJ97, Ye01, HZ01] and in Chapter 3 of this thesis to obtain a 0.7028 approximation algorithm for the general Max-Bisection problem. On the negative side, it has been shown by [Hås01] that approximating the Max-Cut and Max-Bisection problems on general graphs beyond the ratio of $\frac{16}{17}$ is NP-hard. Furthermore, [BK98] show that approximating Max-Cut (and thus Max-Bisection) on 3-regular graphs beyond some explicit constant factor r strictly less than one is also NP-hard.

In the following, we use the work of [GW95] with the results of the previous section to achieve an approximation ratio of 0.795 on Max-Bisection restricted to regular graphs.

Theorem 5.3.1 *The Max-Bisection problem on regular graphs can be approximated within a ratio of* 0.795.

Proof: Consider the Max-Cut algorithm based on semidefinite programming presented in [GW95]. In this algorithm, given a graph $G = (V, E)$, a semidefinite relaxation of the Max-Cut problem on G is solved yielding an embedding of G on the n dimensional unit sphere. This embedding is then rounded using the random hyperplane rounding technique, into a partition of G. In general, it is shown in [GW95] that the expected value of this partition is at least $\alpha \simeq 0.87856$ times the value of the optimal cut in G.

Given a Δ regular graph $G = (V, E)$, using the algorithm of [GW95] one may obtain a partition (X, Y) of G of value $w(X) \geq \alpha Cut(G)$. Applying Theorem 5.2.1 on this partition naively, a bisection $(\hat{X}, \hat{Y})$ of value at least $\theta w(X) \geq \alpha\theta Cut(G) \geq \alpha\theta Bis(G) \simeq 0.793 Bis(G)$ may be obtained. We conclude that a 0.793 approximation algorithm for Max-Bisection on regular graphs is achieved by combining the algorithm of [GW95] and Theorem 5.2.1. A slight improvement in this ratio may be achieved by noticing that the *worst case* value of θ is obtained when $w(X)$ is of value $x^*|E| = 0.7932|E|$, while the *worst case* approximation ratio α of the [GW95] algorithm is obtained when $w(X)$ is of value $0.742|E|$. Details follow.

Denote the value of the semidefinite relaxation of G as $\delta|E|$. It is shown in [GW95] that the value $w(X)$ of the partition (X, Y) is at least $x(\delta)|E|$ where

$$x(\delta) = \begin{cases} \frac{acos(1-2\delta)}{\pi} & \delta \geq 0.8445 \\ \delta \cdot 0.87856 & \delta < 0.8445 \end{cases}$$

Assume that $w(X)$ is exactly of value $x(\delta)|E|$. Recall, using Lemma 5.2.2, that we may obtain a bisection $(\hat{X}, \hat{Y})$ of value at least $\theta(x(\delta))w(X)$ where

$$\theta(x(\delta)) = \frac{1 - x(\delta)}{(2 - x(\delta))x(\delta)} + \frac{1}{(2 - x(\delta))^2}.$$

We conclude that in such a case, the value of the bisection $(\hat{X}, \hat{Y})$ is at least $x(\delta)\theta(x(\delta))|E| \geq \frac{x(\delta)\theta(x(\delta))}{\delta} Bis(G)$. Using basic calculations which are described in Section 5.5 it can be seen that the above is minimal when $\delta \simeq 0.8748$, yielding an approximation ratio of 0.7953.

It is left to show that if the partition (X, Y) is of value greater than that promised by the analysis of [GW95], we obtain a strictly higher approximation ratio. Assume that $w(X)$ is of value $y|E|$ for some y greater than $x(\delta)$. In such a case we may obtain a bisection of G of value $\frac{y\theta(y)}{\delta} Bis(G)$. We conclude that in order to prove our claim it is enough to show that the function $x\theta(x)$ is increasing. Using basic calculations, which are described in Section 5.5, it can be seen that this is true. $\square$

Two remarks regarding the result and proof of Theorem 5.3.1 are in place. The result above holds for regular graphs of arbitrary degree. Using the work presented in Chapter 4, further improved approximation ratios for the Max-Bisection problem can be achieved when we assume the degree is constant. For instance, in Chapter 4 we show that the Max-Cut problem on 3-regular graphs can be approximated within an approximation ratio of 0.924. Thus, combining this result with the result of Theorem 5.2.1, we conclude an 0.834 approximation ratio on the Max-Bisection problem restricted to 3-regular graphs [1].

Regarding the proof of Theorem 5.3.1, we use the results of [GW95] which are based on a semidefinite relaxation for the Max-Cut problem. As we are interested in approximating the maximum bisection, one may add additional constraints to this semidefinite relaxation as is done in [FJ97]. It would be interesting to see if such an addition can improve our results.

5.4 Upper bound

Proposition 5.2.3 *There are infinitely many regular graphs G for which the ratio between $Bis(G)$ and $Cut(G)$ is arbitrarily close to $\theta \simeq 0.9027$.*

Proof: We construct a constant degree regular graph $G = (V, E) = (X, Y; E)$ where X and Y are a partition of V, X is of size cn, and the ratio between $Bis(G)$ and $Cut(G)$ is arbitrarily close to $\theta \simeq 0.9027$. In general, our construction is random. We start by constructing a random regular graph H_x on the vertex set X, and a random regular bipartite graph H_{xy} on the vertex sets X and Y. Afterwards we show that for their union H the ratio between $Bis(H)$ and $Cut(H)$ is close to θ. Using the notation of Theorem 5.2.1 let $x \simeq 0.7932$ and $c = 1 - x/2$.

[1] Using a recent work of Halperin, Livnat and Zwick [HLZ02] this ratio can further be improved to 0.8418 when considering 3 regular graphs.

We now randomly construct a Δ_1 regular multi-graph H_x on the vertices of X. Consider the graph $\hat{H}_x$ consisting of $|X| = cn$ disjoint sets $\{S_1, \ldots, S_{|X|}\}$ of Δ_1 vertices each, *i.e.* a set of Δ_1 vertices corresponding to each vertex of H_x. Define the edge set of $\hat{H}_x$ to be a random perfect matching on its vertices. Note that $\hat{H}_x$ has exactly $\Delta_1|X|/2$ edges. Define H_x to be the multi-graph obtained by *shrinking* each set S_i of vertices in $\hat{H}_x$ into a single vertex i of H_x. That is, H_x is the graph with a single vertex i corresponding to each set S_i in which each edge connecting S_i and S_j in $\hat{H}_x$ is expressed as an edge (i, j) in H_x. In what follows, we analyze some properties of H_x.

Lemma 5.4.1 *For every constant $\varepsilon > 0$ there exists a constant Δ_1 such that with constant probability the following holds.* (a) *H_x will not include self loops or multiple edges, and* (b) *the number of edges in every partition $(A, X \setminus A)$ of H_x is at most $\frac{\Delta_1|A|(cn-|A|)}{cn} + \varepsilon\Delta_1 cn$.*

Proof: Part *(a)* of Lemma 5.4.1 is proven in [BM86b] where it is shown that with some constant probability (depending on Δ_1) H_x will not include self loops or multiple edges. For part *(b)*, let A be some subset of X of size at most $|X|/2$ and define B to be $X \setminus A$. The probability that the cut $(A, X \setminus A)$ has value k is exactly

$$\Pr(w(A) = k) \;=\; \frac{k! \begin{pmatrix} \Delta_1|A| \\ k \end{pmatrix} \begin{pmatrix} \Delta_1|B| \\ k \end{pmatrix} M(\Delta_1|A| - k)M(\Delta_1|B| - k)}{M(\Delta_1|X|)}$$

Where $M(i) = \frac{i!}{(\frac{i}{2})!\; 2^{\frac{i}{2}}}$ is the number of perfect matchings in a graph with i vertices. Using basic calculations which are described in Section 5.5 it can be seen that $\Pr(w(A) = k)$ is at most $\delta^{\Delta_1 cn}$ for some constant $\delta < 1$ (dependent on ε) for any $\Delta_1|A| \geq k \geq \frac{\Delta_1|A|(cn-|A|)}{cn} + \varepsilon\Delta_1 cn$. As there are at most 2^{cn} subsets A of X and the range of k is polynomial in n, we conclude by choosing Δ_1 large enough that with overwhelming probability $(1 - \delta^n)$ part *(b)* of our lemma holds. Hence both properties *(a)* and *(b)* hold for the random graph H_x with some constant probability. $\square$

We now construct the multi-graph H_{xy}, a bipartite graph on the vertex sets X and Y in which the degree of each vertex in Y is Δ and the degree of each vertex in X is $\Delta(1/c - 1)$. The construction is similar to the construction of H_x presented above. Consider the graph $\hat{H}_{xy}$ consisting of $|X| = cn$ disjoint sets $\{S_1, \ldots, S_{|X|}\}$ each of $\Delta(1/c-1)$ vertices, and $|Y| = (1-c)n$ disjoint sets $\{R_1, \ldots, R_{|Y|}\}$ each of Δ vertices. Define the edge set of $\hat{H}_{xy}$ to be a random bipartite perfect matching between the vertices in $\{S_1, \ldots, S_{|X|}\}$ and $\{R_1, \ldots, R_{|Y|}\}$. Define H_{xy} to be the multi-graph obtained by shrinking each vertex set S_i into a single vertex $i \in X$ and each vertex set R_i into a single vertex $i \in Y$.

Lemma 5.4.2 *For every constant $\varepsilon > 0$ there exists a constant Δ such that with constant probability the following holds.* (a) *H_{xy} will not include multiple edges, and* (b) *the number of edges in H_{xy} between every two subsets $A \subseteq X$ and $B \subseteq Y$ is at most $\frac{\Delta|A||B|}{cn} + \varepsilon\Delta(1-c)n$.*

Proof: As in Lemma 5.4.1, part *(a)* of the lemma is stated in [BM86b] (with constant probability). For part *(b)*, let $N = (1-c)\Delta n = \Delta|Y|$. Let A be some subset of X and B be some subset of Y. Let $\Delta(1/c-1)|A| = \alpha N$, and $\Delta|B| = \beta N$. The probability that there are k edges between A and B in H_{xy} is exactly

$$\Pr(w(A, B) = k) \;=\; \frac{\begin{pmatrix} \alpha N \\ k \end{pmatrix} \begin{pmatrix} \beta N \\ k \end{pmatrix} k! \begin{pmatrix} N - \alpha N \\ \beta N - k \end{pmatrix} (\beta N - k)!(N - \beta N)!}{N!}$$

Similarly to Lemma 5.4.1, basic calculations which are described in Section 5.5 yield that $\Pr(w(A,B) = k)$ is at most δ^N for some constant $\delta < 1$ (dependent on ε) for any $k \geq \frac{\Delta|A||B|}{cn} + \varepsilon\Delta(1-c)n$. As there are at most 2^{2n} subsets A,B of X,Y respectively and the range of k is polynomial in n, we conclude that part *(b)* of our lemma holds with overwhelming probability. Hence both properties *(a)* and *(b)* hold for the random graph H_{xy} with constant probability. □

Set Δ_1 to be $(2-1/c)\Delta$ and define H to be the union of the two graphs H_x and H_{xy}. Assume that H_x and H_{xy} have the properties stated in Lemma 5.4.1 and 5.4.2 (this happens with constant probability). It is not hard to verify that H is a Δ regular graph, and that the value of the partition (X,Y) in H is $\Delta(1-c)n$. Thus $Cut(H)$ is at least this value. In what follows, we show that the maximum bisection of H is at most $(\theta + 8\varepsilon)Cut(H)$, thus completing our proof.

Let (U,V) be some bisection of H, where $U = X_1 \cup Y_1$, $V = X_2 \cup Y_2$, the sets X_i are some partition of X, and the sets Y_i are some partition of Y. Denote the size of the set X_1 as γn, the size of X_2 as $(c-\gamma)n$, the size of Y_1 as $(1/2-\gamma)n$, and the size of Y_2 as $(1/2-c+\gamma)n$ for $\gamma \in [c/2, 1/2]$. Let p_x be $\frac{\Delta(2-1/c)}{cn}$ and p_{xy} be $\frac{\Delta}{cn}$. We have that the value of the bisection (U,V) is at most

$$n^2\left(\gamma(c-\gamma)(p_x - 2p_{xy}) + \frac{c \cdot p_{xy}}{2}\right) + 3\varepsilon n\Delta.$$

Which is maximal when $\gamma = 1/2$. Thus we conclude using the fact that $c = 1 - x/2$ that

$$\frac{Bis(H)}{Cut(H)} \leq \left(\frac{1 - \frac{2c-1}{2c^2}}{x}\right) + 8\varepsilon = \left(\frac{1-x}{(2-x)x} + \frac{1}{(2-x)^2}\right) + 8\varepsilon \simeq \theta + 8\varepsilon.$$

□

5.5 Proof of claims

Recall that

$$\begin{aligned} x(\delta) &= \begin{cases} \frac{acos(1-2\delta)}{\pi} & \delta \geq 0.8445 \\ \delta \cdot 0.87856 & \delta < 0.8445 \end{cases} \\ \theta(x) &= \frac{1-x}{(2-x)x} + \frac{1}{(2-x)^2}. \end{aligned}$$

Bounding $\theta(x)$ (Lemma 5.2.2):

$$\theta'(x) = -\frac{x^3 - 2x^2 + 6x - 4}{x^2(x-2)^3}.$$

Using computer assisted analysis, it can be seen that $\theta'(x)$ is zero only when $x^* \simeq 0.7932$, yielding a lower bound of approximately 0.9027. Note that $\theta(x)$ is decreasing when $x \leq 0.7932$ and increasing otherwise.

Monotonicity of $x\theta(x)$ (Theorem 5.3.1):

$$(x\theta(x))' = \frac{2x}{(2-x)^3}.$$

It can be easily seen that $(x\theta(x))'$ is positive for every $x \in [0.5, 1]$ (the range of our interest).

Bounding $\frac{x(\delta)\theta(x(\delta))}{\delta}$ (Theorem 5.3.1): We consider three cases, the first in which $\delta \in [0.5, 0.8445]$, the second in which δ is in the range $[0.8445, 0.8981]$, and the last in which $\delta \in [0.8981, 1]$. In the first case we have that $\frac{x(\delta)\theta(x(\delta))}{\delta}$ is equal to $0.87856 \cdot \theta(\delta \cdot 0.87856)$. As $\delta \cdot 0.87856 \in [0.4392, 0.7419]$, and $\theta(x)$ is monotone decreasing when $x \leq 0.7932$, we conclude that $\frac{x(\delta)\theta(x(\delta))}{\delta}$ is decreasing in this range and

$$\frac{x(\delta)\theta(x(\delta))}{\delta} \geq 0.87856 \cdot \theta(0.7419) \geq 0.7983.$$

In the third case, in which $\delta \in [0.8981, 1]$, we have that $x(\delta) \in [0.7932, 1]$. Using the fact that $x'(\delta) \geq 0$ we have

$$\left(\frac{x(\delta)}{\delta}\right)' = \left(\frac{2x(\delta)}{1-\cos(\pi x(\delta))}\right)' = \frac{2(1-\cos(\pi x(\delta)) - \pi x(\delta)\sin(\pi x(\delta)))}{(1-\cos(\pi x(\delta)))^2} x'(\delta) \geq 0.$$

As $\theta(x)$ is monotone increasing when $x \geq 0.7932$, we conclude that $\frac{x(\delta)\theta(x(\delta))}{\delta}$ is increasing in the range $[0.8981, 1]$ and is of a minimal value of 0.7972 when $\delta = 0.8981$.

The final case in which $\delta \in [0.8445, 0.8981]$ is proven using computer assisted analysis. A plot of $\frac{x(\delta)\theta(x(\delta))}{\delta}$ in the above range is displayed in Figure 5.1. It can be seen that the function obtains a minimal value of approximately 0.7953 when $\delta \simeq 0.8748$. We thus conclude that $\frac{x(\delta)\theta(x(\delta))}{\delta}$ is bounded by 0.7953 in the range $\delta \in [0.5, 1]$.

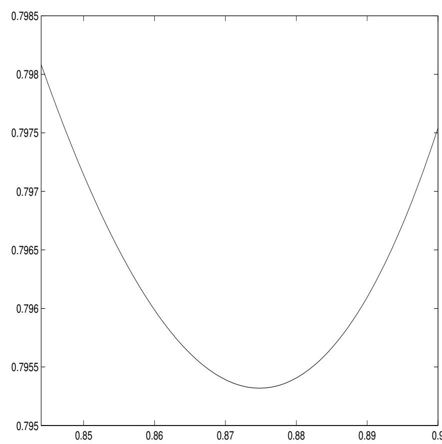

Figure 5.1: The function $\frac{x(\delta)\theta(x(\delta))}{\delta}$ in the range $\delta \in [0.8445, 0.8981]$.

Lemma 5.4.1: Recall that $|X|$ is of size cn, $A \subseteq X$, $k \geq \frac{\Delta_1|A|(cn-|A|)}{cn} + \varepsilon\Delta_1 cn$, and

$$\Pr(w(A) = k) \;=\; \frac{k!\begin{pmatrix}\Delta_1|A|\\ k\end{pmatrix}\begin{pmatrix}\Delta_1|B|\\ k\end{pmatrix}M(\Delta_1|A|-k)M(\Delta_1|B|-k)}{M(\Delta_1|X|)}$$

Where $M(i) = \frac{i!}{(\frac{i}{2})!\;2^{\frac{i}{2}}}$. Let $N = \Delta_1 cn$, $\Delta_1|A| = \alpha N$ and $k = \beta N$. The condition above on k implies that $\beta \geq \alpha(1-\alpha)+\varepsilon$. Ignoring factors which are polynomial in n we conclude using *Stirling's formula* that

$$\Pr(w(A)=k) \simeq \left(\frac{\alpha^{\alpha}(1-\alpha)^{(1-\alpha)}}{\beta^{\beta}(\alpha-\beta)^{\frac{\alpha-\beta}{2}}(1-\alpha-\beta)^{\frac{1-\alpha-\beta}{2}}} \right)^{N}$$

The above formula is decreasing in β (as long as $\beta \geq \alpha(\alpha-1)$). Furthermore, using computer assisted analysis, it can be seen that by setting β to be $\alpha(1-\alpha)+\varepsilon$ the resulting formula is increasing in α (as long as $\alpha \leq 1/2$). We conclude that

$$\Pr(w(A)=k) \leq \left(2\left(\frac{1}{4}+\varepsilon\right)^{\frac{1}{4}+\varepsilon}\left(\frac{1}{4}-\varepsilon\right)^{\frac{1}{4}-\varepsilon} \right)^{-N} \leq e^{-4\varepsilon^2 N}$$

where the second inequality is proven in [Hoe63].

Lemma 5.4.2: Recall that X is of size cn, Y is of size $(1-c)n$, $A \subseteq X$, $B \subseteq Y$, and $k \geq \frac{\Delta|A||B|}{cn} + \varepsilon\Delta(1-c)n$. Let $N = (1-c)\Delta n$, $\Delta(1/c-1)|A| = \alpha N$, $\Delta|B| = \beta N$, and $k = \gamma N$. The condition above on k implies that $\gamma \geq \alpha\beta+\varepsilon$. Ignoring factors which are polynomial in n we conclude using *Stirling's formula* that

$$\Pr(w(A,B)=k) \simeq \frac{\alpha^{\alpha}\beta^{\beta}(1-\beta)^{(1-\beta)}(1-\alpha)^{(1-\alpha)}}{\gamma^{\gamma}(\alpha-\gamma)^{(\alpha-\gamma)}(\beta-\gamma)^{(\beta-\gamma)}(1-\alpha-\beta+\gamma)^{(1-\alpha-\beta+\gamma)}}$$

Using computer assisted analysis it can be seen that the above formula is maximal when $\alpha = \beta = 1/2$ and $\gamma = \alpha\beta + \varepsilon$. In such a case we obtain

$$\Pr(w(A,B)=k) \leq \left(2\left(\frac{1}{4}+\varepsilon\right)^{\frac{1}{4}+\varepsilon}\left(\frac{1}{4}-\varepsilon\right)^{\frac{1}{4}-\varepsilon} \right)^{-2N} \leq e^{-8\varepsilon^2 N}.$$

Bibliography

[AK98] N. Alon and N. Kahale. Approximating the independence number via the θ function. *Mathematical Programming*, 80(2):253–264, 1998.

[AK02] N. Alon and M. Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Mathematics*, 15(2):211–227, 2002.

[Ali95] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):13–51, 1995.

[ALM+98] S. Arora, C. Lund, R. Motowani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 3(45):501–555, 1998.

[Aro98] S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *Journal of the ACM*, 45(5):753–782, 1998.

[AS98] S. Arora and S. Safra. Probabilistic checking of proofs: A new characterization of NP. *Journal of the ACM*, 1(45):70–122, 1998.

[ASZ02] N. Alon, B. Sudakov, and Uri Zwick. Constructing worst case instances for semidefinite programming based approximation algorithms. *SIAM Journal on Discrete Mathematics*, 15(1):58–72, 2002.

[AW02] T. Asano and D.P. Williamson. Improved approximation algorithms for MAX SAT. *Journal of Algorithms*, 42(1):173–202, 2002.

[Ben84] Y. Benyamini. Two point symmetrization, the isoperimetric inequality on the shpere and some applications. *Longhorn Notes, Univ. of Texas, Texas Funct. Anal. Seminar*, pages 53–76, 1983-1984.

[Bez02] S. Bezrukov. Edge-isoperimetric problems of graphs. *To appear in J'anos Bolyai Math. Series*, 2002.

[BK97] A. Blum and D. Karger. An $o(n^{3/14})$-coloring for 3-colorable graphs. *Information Processing Letters*, 61(6):49–53, 1997.

[BK98] P. Berman and M. Karpinski. On some tighter inapproximability results, further improvements. *ECCC*, TR98-065, 1998. Extended abstract appears in *ICALP* 1999, pages 200-209.

[BM86a] F. Barahona and A.R. Mahjoub. On the cut polytope. *Mathematical Programming*, 36:157–173, 1986.

[BM86b] B. Bollobás and B.D. McKay. The number of matchings in random graphs and bipartite graphs. *J. of Combinatorial Theory, Ser. B.*, 41:80–91, 1986.

[BT76] A. Baernstein and A. Taylor. Spherical rearrangments, subharmonic functions, and *-functions in n-space. *Duke Math J.*, 43:245–268, 1976.

[CFMP99] T. Calamoneri, I. Finocchi, Y. Manoussakis, and R. Petreschi. Approximation algorithm for the Max Cut problem on cubic graphs. *Advances in Computing Science - ASIAN'99, Lect. Notes in Comp. Science 1742, Springer Verlag Ed.*, pages 27–36, 1999.

[Cha02] M. Charikar. On semidefinite programming relaxations for graph coloring and vertex cover. *Proceedings of SODA*, pages 616–620, 2002.

[CK95] P. Crescenzi and V. Kann. A compendium of NP optimization problems. *Technical report SI/RR-95/02, Dipartimento di Scienze dell'Informazione, Univetsitá di Roma La Sapienza. Uptaded version can be found at http://www.nada.kth.se/ viggo/problemlist*, 1995.

[CS98] B. Chor and M. Sudan. A geometric approach to betweenness. *SIAM Journal on Discrete Mathematics*, 11(4):511–523, 1998.

[DKRS99] I. Dinur, G. Kindler, R. Raz, and S. Safra. An improved lower bound for approximating CVP. *Manuscript*, 1999.

[DKS98] I. Dinur, G. Kindler, and S. Safra. Approximating-CVP to within almost-polynomial factors is NP-hard. *In Proceedings of 39th Symposium on the Foundations of Computer Scienc*, pages 99–111, 1998.

[dlVL81] W. F. de la Vega and G. Lueker. Bin packing can be solved within $1+\varepsilon$ in linear time. *Combinatorica*, 1:349–355, 1981.

[Fei97] U. Feige. Randomized graph products, chromatic numbers, and the Lovász theta function. *Combinatorica*, 17(1):79–90, 1997.

[Fei98] U. Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998.

[Fei02] U. Feige. Approximating maximum clique by removing subgraphs. *Manuscript*, 2002.

[Fel66] W. Feller. *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons, 1966.

[FG95] U. Feige and M.X. Goemans. Approximating the value of two prover proof systems with applications to Max-2-Sat and Max-Dicut. *Proc. of the 3rd Israel Symposium on Theory of Computing and Systems*, pages 182–189, 1995.

[FGL+96] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximation. *Journal of ACM*, 43(2):268–292, 1996.

[FJ97] A. Frieze and M. Jerrum. Improved approximation algorithms for Max-k-Cut and Max-Bisection. *Algorithmica*, 18:67–81, 1997.

[FK98] U. Feige and J. Kilian. Zero knowledge and the chromatic number. *Journal of Computer and System Sciences*, 57:187–199, 1998.

[FKL00] U. Feige, M. Karpinski, and M. Langberg. Improved approximation of Max-Cut on graphs of bounded degree (extended version). *ECCC TR00-021*, 2000.

[FKL01] U. Feige, M. Karpinski, and M. Langberg. A note on approximating Max-Bisection on regular graphs. *Information Processing Letters*, 79:181–188, 2001.

[FKL02] U. Feige, M. Karpinski, and M. Langberg. Improved approximation of Max-Cut on graphs of bounded degree. *Journal of Algorithms*, 43(2):201–219, 2002.

[FKP01] U. Feige, G. Kortsarz, and D. Peleg. The Dense k-subgraph problem. *Algorithmica*, 29(3):410–421, 2001.

[FL01a] U. Feige and M. Langberg. Approximation algorithms for maximization problems arising in graph partitioning. *Journal of Algorithms*, 41:174–211, 2001.

[FL01b] U. Feige and M. Langberg. The RPR^2 rounding technique for semidefinite programs. *In proceedings of ICALP*, pages 213–224, 2001.

[FLS02] U. Feige, M. Langberg, and G. Schechtman. Graphs with tiny vector chromatic numbers and huge chromatic numbers. *In proceedings of 43rd annual Symposium on Foundations of Computer Science*, pages 283–292, 2002.

[FS02] U. Feige and G. Schechtman. On the optimality of the random hyperplane rounding technique for MAX CUT. *Random Structures and Algorithms*, 20(3):403–440, 2002.

[GGR98] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of ACM*, 45(4):653–750, 1998.

[GJ79] M.R. Garey and D.S. Johnson. Computers and intractability: A guide to the theory of NP-completenes. *W.H. Freeman*, 1979.

[GK98] M. Goemans and J. Kleinberg. The Lovász theta function and a semidefinite programming relaxation of Vertex Cover. *SIAM Journal on Discrete Mathematics*, 11:196–204, 1998.

[GL89] G.H. Golub and C.F. Van Loan. Matrix computation (second edition). *The Johns Hopkins University Press, Baltimore, Maryland*, 1989.

[GLS88] M. Grötschel, L. Lovász, and A. Schrijver. Geometric algorithms and combinatorial optimization. *Springer-Verlag*, 1988.

[GW94] M.X. Goemans and D.P. Williamson. New 3/4-approximation algorithms for the maximum satisfiability problem. *SIAM Journal on Discrete Mathematics*, 7(4):656–666, 1994.

[GW95] 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.

[GW01] M.X. Goemans and D.P. Williamson. Approximation algorithms for MAX-3-CUT and other problems via complex semidefinite programming. *In Proceedings of 33rd Symposium on Theory of Computing*, pages 443–452, 2001.

[Hal93] M. Halldorsson. A still better performance guarantee for approximate graph coloring. *Information Processing Letters*, 45:19–23, 1993.

[Hal00] E. Halperin. Improved approximation algorithms for the vertex cover problem in graphs and hypergraphs. *In Proceedings of the 11th SODA*, pages 329–337, 2000.

[Hås99] J. Håstad. Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Math*, 182(1):105–142, 1999.

[Hås01] J. Håstad. Some optimal inapproximability results. *Journal of ACM*, 48:798–859, 2001.

[HLZ02] E. Halperin, D. Livnat, and U. Zwick. Max-Cut in cubic graphs. *In Proceedings of SODA*, pages 506–513, 2002.

[HNZ01] E. Halperin, R. Nathaniel, and U. Zwick. Coloring k-colorable graphs using smaller palettes. *Proceedings of SODA*, pages 319–326, 2001.

[Hoc82] D.S. Hochbaum. Approximation algorithms for the set covering and vertex cover problem. *SIAM J. on Computing*, 11(3):555–556, 1982.

[Hoc97] D.S. Hochbaum. Approximation algorithms for NP-hard problems. *PWS Publishing Company*, 1997.

[Hoe63] W. Hoeffding. Probability inequalities for sums of bounded random variables. *American Statistical Association Journal*, 58(301):13–30, 1963.

[HYZ02] Q. Han, Y. Ye, and J. Zhang. An improved rounding method and semidefinite programming relaxation for graph partition. *Mathematical Programming*, 92(3):509–535, 2002.

[HZ99] E. Halperin and U. Zwick. Approximation algorithms for Max-4-Sat and rounding procedures for semidefinite programming. *In Procedures of IPCO*, page 202, 1999.

[HZ01] E. Halperin and U. Zwick. A unified framework for obtaining improved approximation algorithms for maximum graph bisection problems. *Proceedings of 7th IPCO*, pages 202–217, 2001.

[IK75] O.H. Ibarra and C.E. Kim. Fast approximation for the knapsack and sum of subset problems. *Journal of the ACM*, 22:463–468, 1975.

[Joh74] D. S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Comput. and Sys. Sci.*, 9:256–278, 1974.

[Kar96] H. J. Karloff. How good is the Goemans-Williamson MAX CUT algorithm? *In Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 427–434, 1996.

[Kha79] L.G. Khachiyan. A polynomial algorithm for linear programming (translation from Russian). *Soviet Mathematics Doklady*, 20:191–194, 1979.

[Kho01] S. Khot. Improved inapproximability results for max clique, chromatic number and approximate graph coloring. *In Proceedings of 42nd IEEE Symposium on Foundations of Computer Science*, 2001.

[KKL88] J. Kahn, G. Kalai, and N. Linial. The influence of variables on boolean functions. *In proceedings of 29th Symposium on the Foundations of Computer Science*, pages 68–80, 1988.

[KMS98] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. *Journal of ACM*, 45(2):246–265, 1998.

[KZ97] H. Karloff and U. Zwick. A 7/8-approximation algorithm for Max-3-Sat? *In Proceedings of the 38th Annual IEEE Symposium on Foundations of Computer Science*, pages 406–415, 1997.

[LLZ02] M. Lewin, D. Livnat, and U. Zwick. Improved rounding techniques for the MAX 2-SAT and MAX DI-CUT problems. *In Proceedings of IPCO*, pages 67–82, 2002.

[Lov79] L. Lovász. On the Shannon capacity of a graph. *IEEE Transactions on Information Theory*, 25:2–13, 1979.

[LY94] C. Lund and M. Yannakakis. On the hardness of approximation minimization problems. *Journal of ACM*, 41(5):960–981, 1994.

[MM01] S. Matuura and T. Matsui. 0.863-approximation algorithm for MAX DICUT. *In Proceedings of RANDOM-APPROX*, pages 138–146, 2001.

[Nes98] Y. E. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software*, 9:141–160, 1998.

[Pol95] S. Poljak. Local search for Max-Cut. *SIAM Journal on Computing*, 24:822–839, 1995.

[PY91] C.H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991.

[Ren70] A. Renyi. Probability theory. *Elsevier, New York*, 1970.

[RT87] P. Raghavan and C.D. Thompson. Randomized rounding : A technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7(4):365–374, 1987.

[Sch85] C.P. Schnorr. A hierarcy of polynomial-time basis reduction algorithms. *In Proceedings of Conference on Algorithms, Pécs (Hungary)*, pages 375–386, 1985.

[Sku01] M. Skutella. Convex quadratic and semidefinite programming relaxations in scheduling. *Journal of ACM*, 48(2):206–242, 2001.

[SY91] A.A. Schäffer and M. Yannakakis. Simple local search problems that are hard to solve. *SIAM Journal on Computing*, 20:56–87, 1991.

[Sze94] M. Szegedy. A note on the θ number of Lovász and the generalized Delsarte bound. *In proceedings of 35th Annual Symposium on Foundations of Computer Science*, pages 36–39, 1994.

[Tur41] P. Turán. On an extremal problem in graph theory. *Mat. Fiz. Lapok*, 48:436–452, 1941.

[Vaz01] V.V. Vazirani. Approximation algorithms. *Springer-Verlag, Berlin*, 2001.

[Ye01] Y. Ye. A 0.699-approximation algorithm for Max-Bisection. *Mathematical Programming*, 90:101–111, 2001.

[Zwi99] U. Zwick. Outward rotations: a new tool for rounding solutions of semidefinite programming relaxations, with application to Max-Cut and other problems. *In Proceedings of the 31th ACM Symposium on Theory of Computing*, pages 679–687, 1999.