Graph Embeddings and Approximate Graph Coloring

by

François Labelle

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science.

School of Computer Science
McGill University
Montréal, Canada

May 2000

0-612-64386-7

Abstract

We describe how to embed a graph into a small n-dimensional hypersphere such that the endpoints of any edge are placed exactly one unit of distance apart. As an application, we show how such an embedding can be used to find large independent sets in sparse 3-colorable graphs. By combining this algorithm with another which is specialized in dense graphs, we obtain a randomized, polynomial time algorithm that can color any 3-colorable graph using at most $O(n^{1/4}\log^{1/2} n)$ colors.

Résumé

Nous décrivons comment inclure un graphe dans une petite hypersphère en n dimensions de sorte que la distance entre les extremités de chaque arête soit de exactement une unité. Comme application, nous montrons comment une telle inclusion peut être employée pour trouver de grands ensembles indépendants dans les graphes creux qui sont 3-colorables. En combinant cet algorithme avec un autre qui est specialisé dans les graphes denses, nous obtenons un algorithme randomisé qui peut colorer n'importe quel graphe 3-colorable en temps polynomial, en utilisant au plus $O(n^{1/4}\log^{1/2} n)$ couleurs.

Contents

Chapter 1

Introduction

Suppose that given a graph, we build a physical model of it where edges are unit-length rods and vertices are flexible joints. To give more freedom, we suppose that this linkage lies in many dimensions and that edges and vertices can cross. A graph constructed in this manner is likely to be very flexible, and therefore can take a variety of shapes. These shapes are called *unit-distance embeddings* of a graph in $\mathbf{R}^d$.

Having made this definition, we can ask for a unit-distance embedding that is *optimal* in some respect. One such optimization will be investigated: given a graph, find the most "compact" unit-distance embedding. That is, the linkage should lie in a hypersphere of radius r where r is minimum. Finding such a compact embedding has applications to the problem of graph coloring. We will show how it can be used to find an independent set of size $\geq 0.006\, n^{4/3} m^{-1/3} \left(\ln \frac{4m}{n}\right)^{-1/2}$ in a 3-colorable graph with n vertices and m edges such that $m \geq 2.61\, n$. This will be added to a toolbox of algorithms that also find large independent sets in 3-colorable graphs. It will be shown how these can be combined to give an algorithm that can color 3-colorable graphs using $O(n^{1/4} \log^{1/2} n)$ colors, and how most of the approximation algorithms of the literature for this problem can be obtained as a combination of some independent set algorithms from our tool-box.

1.1 Statement of originality

The main result is not new (it can be found in [KMS94]), however, the approach taken here is different. For instance, we consider an embedding that is equivalent to but different from the one in [KMS94]. This allows us to replace the concept of *vector-coloring* by the known concept of ***unit-distance embedding***, which is also more intuitive. Although the connection with unit-distance embeddings is simple and obvious, it is neither mentioned in [KMS94], nor in the rewrites [MNR97] and [KMS98]. In fact, all of the results of Chapter 2 have been rediscovered by myself, and the proof supplied is mine.

Finding independent sets as an intermediate step toward a coloring is an old idea [Joh74a], but it is not explicitly stated in recent papers like [BK97] and [KMS98]. Here, the idea is fully exploited, and leads to the diagram of Figure 5.2, which not only explains graphically where the number 1/4 of the performance analysis comes from, but also contains in itself an historical account of most of the previous approximation algorithms for coloring 3-colorable graphs. Such a diagram has never appeared before.

1.2 Previous work on general graph coloring

Most previous results on approximate graph coloring can be classified into 4 categories, depending on whether they apply to *all graphs* or only to *k-colorable graphs*, and whether they are *approximation algorithms* or *inapproximability results*.

We will denote by A an algorithm that colors an n-vertex graph G in time polynomial in n, and write $A(G)$ for the number of colors used by the algorithm A on the graph G. We also write $\chi(G)$ for the chromatic number of G (the minimum number of colors necessary to color the graph G). See Definitions 2.1 and 2.6 for formal definitions of a graph, of a coloring, and of the function χ.

Definition 1.1 *Write $f = O(g(n))$ to mean that $f \leq cg(n)$ for some constant c, and $f = \tilde{O}(g(n))$ to mean that $f = O(g(n) \log^a n)$ for some constant exponent a.*

In the case of general graphs, the results are often given as a bound on the *approximation ratio* $\frac{A(G)}{\chi(G)}$ in terms of n. (Note that $\frac{A(G)}{\chi(G)} \geq 1$ by definition of $\chi(G)$.) For example, the trivial algorithm *"Color G by using one different color for each vertex"* is an example of an algorithm A which satisfies:

$$\frac{A(G)}{\chi(G)} = \frac{n}{\chi(G)} \leq n \quad \text{for every graph } G \tag{1.1}$$

In [Joh74b], Johnson analyses the worst case performance of many heuristics and shows that in this respect they are comparable to the trivial algorithm. Furthermore, the best known algorithms are in some sense not significantly better, as shown in Table 1.1.

On the other hand, by using Probabilistically Checkable Proof systems, many inapproximability results have been found. That is, assuming various computational complexity conjectures, it is shown that some approximation ratios cannot be attained by any polynomial time algorithm. A list is given in Table 1.2.

The result of [FK96] is inspired by zero-knowledge proof systems and almost matches the performance of the best approximation algorithm.

reference	performance
[Joh74b]	$\frac{A(G)}{\chi(G)} = O(\frac{n}{\log n})$
[Wig83]	$\frac{A(G)}{\chi(G)} = O(\frac{n(\log\log n)^2}{\log^2 n})$
[BR90]	$\frac{A(G)}{\chi(G)} = O(\frac{n(\log\log n)^3}{\log^3 n})$
[Hal93]	$\frac{A(G)}{\chi(G)} = O(\frac{n(\log\log n)^2}{\log^3 n})$

Table 1.1: Approximation algorithms for general graph coloring

reference	impossible for any $\epsilon > 0$	assumption
[BS94]	$\frac{A(G)}{\chi(G)} \leq n^{\frac{1}{14}-\epsilon}$ for every graph G	P $\neq$ NP
[BS94]	$\frac{A(G)}{\chi(G)} \leq n^{\frac{1}{10}-\epsilon}$ for every graph G	co-R$\bar{\text{P}}$ $\neq$ N$\bar{\text{P}}$
[BGS95]	$\frac{A(G)}{\chi(G)} \leq n^{\frac{1}{7}-\epsilon}$ for every graph G	P $\neq$ NP
[Für95]	$\frac{A(G)}{\chi(G)} \leq n^{\frac{1}{5}-\epsilon}$ for every graph G	co-RP $\neq$ NP
[FK96]	$\frac{A(G)}{\chi(G)} \leq n^{1-\epsilon}$ for every graph G	co-RP $\neq$ NP

Table 1.2: Inapproximability results for general graph coloring

1.3 Previous work on restricted graph coloring

In the previous section it was shown that finding good colorings is a hard problem. but this is due to the presence of some graphs with relatively large chromatic number $\chi(G)$. If we restrict ourselves to graphs with $\chi(G) \leq k$ (called k-colorable graphs). then more refined results become possible.

In a classic paper. Karp [Kar72] showed that it is NP-complete to color a 3-colorable graph using 3 colors. More recently. it has been shown [KLS92] that it is also NP-complete to color a 3-colorable graph using 4 colors. Therefore. assuming that P $\neq$ NP. it is impossible to find a polynomial time algorithm A such that:

$$A(G) < 5 \quad \text{for every 3-colorable graph } G \tag{1.2}$$

For $k \geq 6$. Garey and Johnson [GJ76]. [GJ79] exhibit a family of k-colorable graphs such that it is NP-complete to color them using less that $2k - 4$ colors. Therefore. assuming that P $\neq$ NP, the following is impossible:

$$A(G) < 2k - 4 \quad \text{for every } k\text{-colorable graph } G \quad (k \geq 6) \tag{1.3}$$

More recently it was shown [KLS92]. [LY94] that for any $c > 1$. there exists a constant k_c such that the following is impossible (again. assuming that P $\neq$ NP):

$$A(G) < ck \quad \text{for every } k\text{-colorable graph } G \quad (k \geq k_c) \tag{1.4}$$

On the positive side. the literature contains many approximation algorithms for the problem (which are described and classified in this thesis). shown in Table 1.3.

reference	$A(G)$ for 3-colorable G	$A(G)$ for k-colorable G
[Wig83]	$O(\sqrt{n})$	$O(n^{\frac{k-2}{k-1}})$
[BR90]	$O(\sqrt{n/\log n})$	$O((n/\log n)^{\frac{k-2}{k-1}})$
[Blu89]	$\tilde{O}(n^{\frac{2}{5}})$	$\tilde{O}(n^{\frac{3k-7}{3k-4}})$
[Blu94]	$\tilde{O}(n^{\frac{3}{8}})$	$\tilde{O}(n^{\frac{3k-7}{3k-4}})$
[KMS94]	$\tilde{O}(n^{\frac{1}{4}})$	$\tilde{O}(n^{\frac{k-2}{k+1}})$
[BK97]	$\tilde{O}(n^{\frac{2}{9}})$	
[BK97]	$\tilde{O}(n^{\frac{3}{14}})$	

Table 1.3: Approximation algorithms for restricted graph coloring

We note that for k-colorable graphs there currently exists a large gap between the performance of the best known approximation algorithm and the best known inapproximability result. Namely for 3-colorable graphs, the best performance $A(G)$ theoretically achievable in polynomial time can be anything between 5 and $n^{\frac{1}{14}}$. This is unlike the case of general graphs (Section 1.2) where the gap is practically closed, assuming that co-RP $\neq$ NP.

Finally, by using improvement techniques, Linial and Vazirani [LV89] have shown that if an approximation algorithm ever achieves $A(G) \leq n^{\epsilon} \quad \forall \epsilon > 0$, then there exists an approximation algorithm that achieves $A(G) \leq \log^{1+\epsilon} n \quad \forall \epsilon > 0$. In other words, for fixed k, the best possible approximation algorithm for k-colorable graphs uses either more that n^c colors (for some c), or closer to $\log n$ colors, but not in between.

Chapter 2

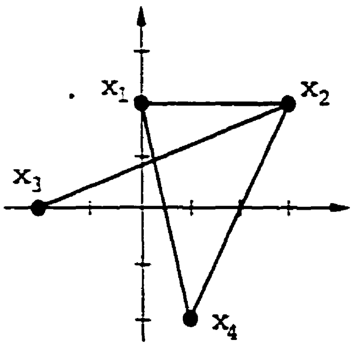
Unit-Distance Embeddings of Graphs

2.1 General embeddings

Since we will only consider graphs that are simple, finite, undirected and unweighted, we will use the following definition.

Definition 2.1 *A graph $G = (V, E)$ is composed of two parts: a set $V = \{v_1, v_2, \ldots, v_n\}$ of n elements called vertices, and a set $E = \{e_1, e_2, \ldots, e_m\} \subseteq V \times V$ of m unordered pairs of distinct vertices (called edges).*

Definition 2.2 *An embedding of an n-vertex graph $G = (V, E)$ into a d-dimensional Euclidean space is simply a function $\varphi : V \to \mathbf{R}^d$ mapping vertices into points in $\mathbf{R}^d$. Let $x_i = \varphi(v_i)$ for all i. Then the matrix of an embedding is a $d \times n$ matrix $X = [x_1, x_2, \ldots, x_n]$, where each x_i is viewed as a column vector.*



$$V = \{v_1, v_2, v_3, v_4\}$$
$$E = \{(v_1, v_2), (v_1, v_4), (v_2, v_3), (v_2, v_4)\}$$
$$X = \begin{bmatrix} 0 & 3 & -2 & 1 \\ 2 & 2 & 0 & -2 \end{bmatrix}$$

Figure 2.1: An embedding of a graph in $\mathbf{R}^2$

The following result shows that not too many dimensions are necessary.

Lemma 2.1 *Embedding an n-vertex graph never requires more than n dimensions. If we have an embedding $\varphi : V \to \mathbf{R}^d$ with $d > n$, then it is equivalent to another embedding $\psi : V \to \mathbf{R}^n$ up to an orthogonal transformation.*

Proof. The n vectors $\varphi(v_1), \varphi(v_2), \ldots, \varphi(v_n)$ span a subspace $S \subseteq \mathbf{R}^d$ of dimension at most n. Let k be the dimension of S and choose an orthonormal basis for S. For each v_i, let $\psi(v_i)$ be the coordinate vector of $\varphi(v_i)$ in the new basis. Then $\psi(v_i) \in \mathbf{R}^k \subseteq \mathbf{R}^n \quad \forall i, 1 \leq i \leq n$.

Note. The careful reader will notice that only $n - 1$ dimensions are necessary if we allow translations. (For example, any triangle has 3 vertices, but it can be drawn on a piece of paper in only 2 dimensions.) A systematic way of achieving this is by translating the set of points to make its "center of mass" coincide with the origin before we apply Lemma 2.1. However, this "$(n-1)$-Lemma" will not be used later because the position of the origin will be important (it cannot be moved).

One reason for embedding a graph into an Euclidean space is that doing so defines a distance between vertices.

Definition 2.3 *The Euclidean distance between two points $x_i, x_j \in \mathbf{R}^d$ is defined to be* $\|x_i - x_j\|_2 = \sqrt{(x_i - x_j)^T(x_i - x_j)}$.

2.2 Small distortion embeddings

Now that these basic concepts have been defined many possibilities can be investigated. For instance, given a connected graph G, it is not always possible to find an embedding φ such that $dist(v_i, v_j) = \|\varphi(v_i) - \varphi(v_j)\|_2$, where $dist(v_i, v_j)$ denotes the graph distance between two vertices. An example of this is the star $K_{1,3}$. However, Bourgain [Bou85] has shown that it is possible to find an embedding with small distortion and small dimension. More precisely, for any n-vertex graph G, there exists an embedding $\varphi : V \to \mathbf{R}^d$ satisfying:

$$\frac{c_1}{\log n} dist(v_i, v_j) \le \|\varphi(v_i) - \varphi(v_j)\|_2 \le dist(v_i, v_j) \quad \forall v_i, v_j \in V$$
$$d \le c_2 \log n$$

where $c_1, c_2 > 0$ are some fixed constants.

In [LLR95], it is shown how such an embedding can be found in polynomial time (by semidefinite programming), and how it can be used to find approximate solutions to a variety of problems.

2.3 Unit-distance embeddings

The previous section was a digression because we will not use small distortion embeddings. However, we will use the following:

Definition 2.4 *A unit-distance embedding of a graph $G = (V, E)$ is an embedding $\varphi : V \to \mathbf{R}^d$ such that edges act like unit-length rods. That is:*

$$\|\varphi(v_i) - \varphi(v_j)\|_2 = 1 \quad \forall (v_i, v_j) \in E$$

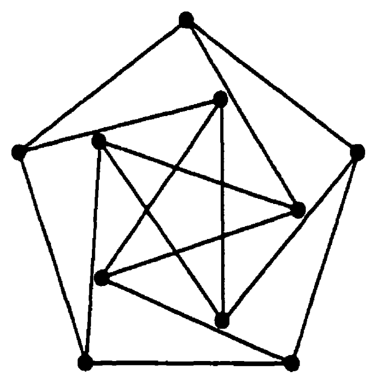


Figure 2.2: The Petersen graph, drawn as a unit-distance graph in $\mathbf{R}^2$

The following definition will help in visualizing many proofs geometrically, as opposed to merely checking the algebra as it is done in, say, [KMS94].

Definition 2.5 *A regular simplex is the generalization of the* equilateral triangle *and of the* regular tetrahedron *to many dimensions. If a regular simplex has k vertices, then the distance between all $\frac{k(k-1)}{2}$ possible pairs of distinct vertices must be the same.*

A regular simplex with k vertices and unit-length sides is easily constructed. Let $z_1, z_2, \ldots, z_k \in \mathbf{R}^k$ be the vertices of the simplex. Then one possibility is represented by the following $k \times k$ matrix:

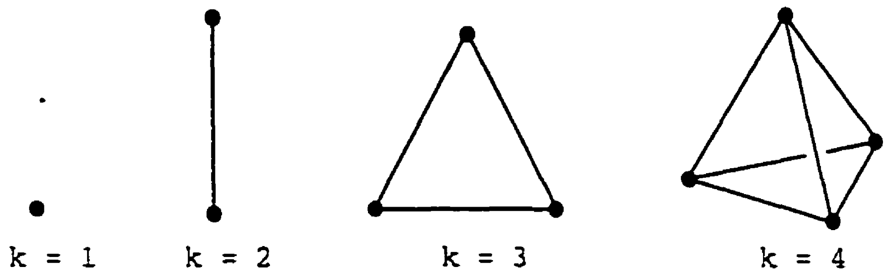


Figure 2.3: Some regular simplices with k vertices

$$Z = [z_1, z_2, \ldots, z_k] = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & 0 & \ldots & 0 \\ 0 & \frac{\sqrt{2}}{2} & 0 & \ldots & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & \frac{\sqrt{2}}{2} \end{bmatrix} \tag{2.1}$$

Clearly, $(z_i - z_j)^T(z_i - z_j) = 1 \quad \forall i \neq j,\ 1 \leq i, j \leq k$.

However, the center of the simplex described above is not the origin. To obtain a *centered* regular simplex with k vertices and unit-length sides, we simply subtract $\frac{\sqrt{2}}{2}(\frac{1}{k})$ from every entry in the above matrix to obtain:

$$Z = \begin{bmatrix} \frac{\sqrt{2}}{2}(\frac{k-1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & \ldots & -\frac{\sqrt{2}}{2}(\frac{1}{k}) \\ -\frac{\sqrt{2}}{2}(\frac{1}{k}) & \frac{\sqrt{2}}{2}(\frac{k-1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & \ldots & -\frac{\sqrt{2}}{2}(\frac{1}{k}) \\ -\frac{\sqrt{2}}{2}(\frac{1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & \frac{\sqrt{2}}{2}(\frac{k-1}{k}) & \ldots & -\frac{\sqrt{2}}{2}(\frac{1}{k}) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\frac{\sqrt{2}}{2}(\frac{1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & -\frac{\sqrt{2}}{2}(\frac{1}{k}) & \ldots & \frac{\sqrt{2}}{2}(\frac{k-1}{k}) \end{bmatrix} \tag{2.2}$$

From this explicit representation we can derive an important lemma:

Lemma 2.2 *In a regular simplex with k vertices and unit-length sides. the distance between a vertex and the center of the simplex is always* $\sqrt{\frac{k-1}{2k}}$.

Proof. Let r be the distance in question. In the representation given by the matrix (2.2). the center of the simplex is the origin. By symmetry, r can be computed from any column of the matrix Z. $r^2 = z_1^T z_1 = \frac{1}{2}(\frac{k-1}{k})^2 + (k-1)\frac{1}{2}(\frac{1}{k})^2 = \frac{k-1}{2k}$. Therefore. $r = \sqrt{\frac{k-1}{2k}}$.

Corollary 2.3 *In a regular simplex with k vertices and unit-length sides. the distance between a vertex and the center of the simplex for* $k = 1, 2, 3, 4, \ldots$ *is* $0, \frac{1}{2}, \sqrt{\frac{1}{3}}, \sqrt{\frac{3}{8}}, \ldots$ *and this tends to* $\sqrt{\frac{1}{2}}$ *as* $k \to \infty$.

Proposition 2.4 *Every graph G has a unit-distance embedding.*

Proof. Suppose that G has n vertices. Simply map the vertices of G to an n-vertex regular simplex. That is. let φ be represented by the matrix (2.1) or (2.2) with $k = n$. Every pair of distinct vertices (in particular the endpoints of every edge) is mapped by φ to points that are 1 unit apart. Therefore, this in a unit-distance embedding of G.

2.4 Spherical embeddings, colorings and cliques

Definition 2.4 can be interpreted in an interesting way. If we imagine that the graph is a *linkage* where vertices are joints and edges are unit-length rods between the joints. then any configuration of the linkage is a unit-distance embedding, and vice-versa.

One question that we can ask is *"can we fold the linkage into a small ball?"* More formally. we ask the question:

Given $r \geq 0$. can we find an embedding $\varphi : V \to \mathbf{R}^n$ which satisfies:

•

$$\|\varphi(v_i) - \varphi(v_j)\|_2 = 1 \quad \forall\, (v_i, v_j) \in E \tag{2.3}$$

$$\|\varphi(v_i)\|_2 = r \quad \forall\, v_i \in V \tag{2.4}$$

Such an embedding will be called a spherical unit-distance embedding because all vertices lie on a hypersphere of radius r centered at the origin.

We will now make important connections between possible values of r and some standard graph theoretical numbers.

Definition 2.6 *Let $G = (V, E)$ be a graph. A k-coloring of G is a function $f : V \to \{1, 2, ..., k\}$ mapping vertices of G into k numbers called "colors". A coloring f is said to be proper iff $f(v_i) \neq f(v_j)$ $\forall\,(v_i, v_j) \in E$. The chromatic number of G is defined to be $\chi(G) = \min\{k \geq 1 :$ there exists a proper k-coloring of $G\}$.*

Lemma 2.5 *Let $G = (V, E)$ be a graph. Let $k = \chi(G)$. Then there exists a spherical unit-distance embedding of G of radius $r = \sqrt{\frac{k-1}{2k}}$.*

Proof. The idea is to map V to the vertices of a simplex according to a coloring of G. See Figure 2.4 for an illustration. Let $f : V \to \{1, 2, ..., k\}$ be a proper k-coloring of G. Let $Z = [z_1, z_2, \ldots, z_k]$ be the representation matrix of a centered regular simplex with k vertices and unit-length sides. Z is given by equation (2.2). We define an embedding $\varphi : V \to \mathbf{R}^k$ by $\varphi(v_i) = z_{f(v_i)}$. This is a unit-distance embedding since $(v_i, v_j) \in E \Rightarrow f(v_i) \neq f(v_j) \Rightarrow \|\varphi(v_i) - \varphi(v_j)\|_2 = 1$. By Lemma 2.2. $r = \sqrt{\frac{k-1}{2k}}$.

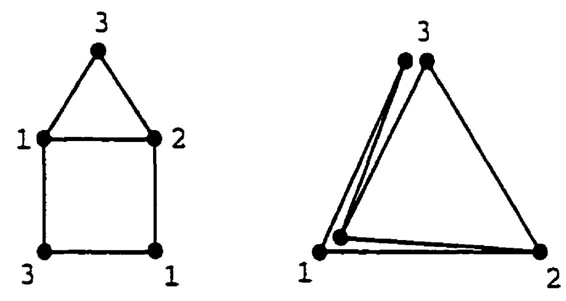


Figure 2.4: A colored graph, and its corresponding embedding onto a simplex (the points should actually overlap on the right)

Definition 2.7 *Let $G = (V, E)$ be a graph. A subset $S \subseteq V$ is said to be a clique of size $|S|$ in G iff $(v_i, v_j) \in E\ \forall v_i, v_j \in S,\ v_i \neq v_j$. Define the clique number of G to be $\omega(G) = \max\{|S| : S \subseteq V$ and S is a clique in $G\}$.*

Lemma 2.6 *Let $G = (V, E)$ be a graph. Let $k = \omega(G)$. Then for every spherical unit-distance embedding of G, the radius must satisfy $r \geq \sqrt{\frac{k-1}{2k}}$.*

Proof. The intuition in this case is that a regular simplex is a "rigid" object that requires a large enclosing radius. Since any embedding maps cliques to regular simplices with unit sides, if G contains a large clique, then the embedding must have a large radius because it contains a large simplex. A formal proof is as follows. Let $\varphi : V \to \mathbf{R}^n$ be any spherical unit-distance embedding. Since the embedding is spherical, we have:

$$\|\varphi(v_i)\|_2 = r \quad \forall v_i \in V \tag{2.5}$$

Let $S = \{s_1, s_2, \ldots, s_k\}$ be a maximum clique in G. For conciseness, write $z_i = \varphi(s_i)$. Since S is a clique, the z_i's are vertices of a regular simplex with k vertices and unit sides. Let $w = \frac{1}{k}\sum_{i=1}^{k} z_i$ be the center of the simplex. We have the following equation, which is similar to the basic equality $Var(X) = \mathbf{E}(X^2) - (\mathbf{E}(X))^2$ of probability theory:

$$\begin{aligned}\frac{1}{k}\sum_{i=1}^{k}\|z_i - w\|_2^2 &= \frac{1}{k}\sum_{i=1}^{k}(z_i - w)^T(z_i - w)\\ &= \frac{1}{k}\sum_{i=1}^{k} z_i^T z_i - \frac{2}{k}\sum_{i=1}^{k} z_i^T w + \frac{1}{k}\sum_{i=1}^{k} w^T w\\ &= \frac{1}{k}\sum_{i=1}^{k}\|z_i\|_2^2 - 2\, w^T w + w^T w\\ &= \frac{1}{k}\sum_{i=1}^{k}\|z_i\|_2^2 - \|w\|_2^2 \qquad (2.6)\end{aligned}$$

From Lemma 2.2, $\|z_i - w\|_2 = \sqrt{\frac{k-1}{2k}} \quad \forall\, i$, and from (2.5) $\|z_i\|_2 = r \quad \forall\, i$. Therefore, (2.6) now reads:

$$\begin{aligned}&\frac{k-1}{2k} = r^2 - \|w\|_2^2\\ \Rightarrow\ &\frac{k-1}{2k} \le r^2\\ \Rightarrow\ &r \ge \sqrt{\frac{k-1}{2k}}\end{aligned}$$

2.5 The Sandwich Theorem

Let $G = (V, E)$ be an n-vertex graph. By Lemma 2.5, at least one spherical unit-distance embedding of G exists. So the following definition makes sense:

Definition 2.8 *Let $r(G)$ be the minimum $r \ge 0$ such that there exists a spherical unit-distance embedding $\varphi : V \to \mathbf{R}^n$ of radius r.*

Then Lemma 2.5 and Lemma 2.6 combine to give the following result:

Theorem 2.7 *Let G be any graph. Let $\chi = \chi(G)$ and $\omega = \omega(G)$ be the chromatic and clique numbers of G. Then $\sqrt{\frac{\omega-1}{2\omega}} \le r(G) \le \sqrt{\frac{\chi-1}{2\chi}}$.*

To get a nicer result, we make a substitution:

Definition 2.9 (The Lovász number of a graph) *Let G be any graph. We define*

$$\bar{\vartheta}(G) = \frac{1}{1 - 2(r(G))^2} \tag{2.7}$$

Solving for r gives $r(G) = \sqrt{\frac{\bar{\vartheta}-1}{2\bar{\vartheta}}}$. Since the function $f(x) = \sqrt{\frac{x-1}{2x}}$ is increasing on the interval $[1, \infty)$ on which it is used. Theorem 2.7 becomes:

Theorem 2.8 (The Sandwich Theorem) *Let G be any graph. Then*

$$\omega(G) \leq \bar{\vartheta}(G) \leq \chi(G) \tag{2.8}$$

The Lovász number of a graph is a graph parameter that has surprisingly many different characterizations (see [Knu94] for instance). It is proven in [KMS94] that the characterization given here is equivalent to other accepted definitions.

Chapter 3

Finding a Minimum Radius Embedding

The number $r(G)$ of Definition 2.8 is interesting theoretically, but even better is the fact that $r(G)$ can be approximated in polynomial time. A spherical unit-distance embedding φ of the claimed radius can also be found. This may be surprising because by Theorem 2.7, $r(G)$ is "sandwiched" between two numbers that are known to be hard to compute (computing any of them is an NP-complete problem).

The technique uses *semidefinite programming*, which unfortunately is too complex to describe here. However, in this chapter we will give enough details so that it will be possible for the reader to compute $r(G)$ numerically by using a semidefinite programming package as a black box. The same basic idea is used in [LLR95] to compute the *small distortion embedding* of Section 2.2.

3.1 Positive semidefinite matrices

Definition 3.1 *Let A be a real symmetric $n \times n$ matrix. A is said to be positive semidefinite iff* $u^T Au \geq 0 \quad \forall u \in \mathbf{R}^n$

We now make a correspondence between embeddings and positive semidefinite matrices.

Fact 3.1 *Let X be the matrix of an embedding, as described in Definition 2.2. Let $A = X^T X$. Then A is real symmetric positive semidefinite.*

Proof. This is straightforward. A is symmetric because $A^T = (X^T X)^T = X^T X^{TT} = X^T X = A$. A is positive semidefinite because for all $u \in \mathbf{R}^n$, $u^T Au = u^T X^T Xu = (Xu)^T (Xu) = \|Xu\|_2^2 \geq 0$.

The converse will be more important to us:

Fact 3.2 *Let A be a real symmetric positive semidefinite $n \times n$ matrix. Then we can find a real $n \times n$ matrix X such that $A = X^T X$.*

Proof. Since A is a real symmetric matrix, we can write $A = P^T DP$ where P is orthogonal and D is diagonal. Since A is positive semidefinite, the diagonal elements of D will be non-negative. So we put $X = D^{1/2}P$. Alternatively, X can be found by the Cholesky decomposition, slightly modified so that it will not divide by zero if A is not of full rank. See a book on matrix computations such as [GVL83] for details.

3.2 Reformulating spherical unit-distance embeddings

Let $V = (G, E)$ be a graph and let $\varphi : V \to \mathbf{R}^n$ be an embedding of G. By letting $X = [\varphi(v_1), \varphi(v_2), \ldots, \varphi(v_n)]$ and $A = [a_{ij}] = X^T X$, we can rewrite the constraints (2.3) and (2.4) of a *spherical unit-distance embedding* as:

$$\begin{aligned} 1 &= \|\varphi(v_i) - \varphi(v_j)\|_2^2 = \|x_i - x_j\|_2^2 = (x_i - x_j)^T(x_i - x_j) \\ &= x_i^T x_i - 2x_i^T x_j + x_j^T x_j = a_{ii} - 2a_{ij} + a_{jj} \quad \forall\, (v_i, v_j) \in E \\ r^2 &= \|\varphi(v_i)\|_2^2 = \|x_i\|_2^2 = x_i^T x_i = a_{ii} \quad \forall\, v_i \in V \end{aligned}$$

Hence the problem of finding a minimum radius spherical unit-distance embedding of a graph (recall Definition 2.8) can be rewritten as the following *semidefinite program.*

Theorem 3.3 *Let $G = (V, E)$ be graph. Suppose that we can find $A = [a_{ij}]$ such that*

$$A \text{ is real symmetric positive semidefinite} \tag{3.1}$$
$$a_{ii} - 2a_{ij} + a_{jj} = 1 \quad \forall\, (v_i, v_j) \in E$$
$$a_{ii} - a_{11} = 0 \quad \forall\, i,\ 2 \le i \le n$$
$$a_{11} \text{ is minimum}$$

Then,

$$r(G) = \sqrt{a_{11}} \tag{3.2}$$
$$\bar{\vartheta}(G) = \frac{1}{1 - 2a_{11}} \tag{3.3}$$
$$\text{an embedding matrix } X \text{ can be found by solving } A = X^T X \tag{3.4}$$

3.3 Solving the semidefinite program

Definition 3.2 *In general, a semidefinite program is an optimization problem with variables a_{ij} that includes the constraint (3.1) and some linear equality constraints of the form $\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}a_{ij} = c$. The objective function to minimize (or maximize) is a linear combination of the a_{ij}'s.*

Theorem 3.4 *A semidefinite program of size n with m equality constraints can be solved within ϵ of the optimum in time polynomial in n, m and $\log\frac{1}{\epsilon}$ by interior point methods [Ali95] or by the ellipsoid method [GLS88], provided that a bound on the value of the optimum is known in advance.*

Theorem 3.4 will not be proved, but we will show that the search space of a semidefinite program is *convex*. Since optimizing a linear function over a convex set is in general much easier than over an arbitrary set, Theorem 3.4 is not too surprising.

Proposition 3.5 *The search space of a semidefinite program is convex. Let A and B be matrices. Let $0 \le \lambda \le 1$. If A and B satisfy the constraints of a semidefinite program, then $C = \lambda A + (1-\lambda)B$ also satisfies these constraints.*

Proof.

1. C is real symmetric. $C^T = \lambda A^T + (1-\lambda)B^T = \lambda A + (1-\lambda)B = C$
2. C is positive semidefinite. Let $u \in \mathbf{R}^n$. $u^TCu = u^T(\lambda A + (1-\lambda)B)u = \lambda u^TAu + (1-\lambda)u^TBu \ge 0 + 0 = 0$.
3. C satisfies the linear equality constraints. Suppose that one of the linear constraint is $\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}a_{ij} = c$. Then:
   $\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}c_{ij} = \lambda\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}a_{ij} + (1-\lambda)\sum_{i=1}^{n}\sum_{j=1}^{n} t_{ij}b_{ij} = \lambda c + (1-\lambda)c = c$.

To simplify the rest of the exposition, we will disregard the error ϵ. assuming that ϵ is chosen so that the precision will be satisfactory.

3.4 An example: the 5-cycle

Let G be a graph. If somehow we know that

$$\omega(G) = \chi(G) \tag{3.5}$$

Then for this graph. Theorem 2.8 becomes $\omega(G) = \bar{\vartheta}(G) = \chi(G)$. This means that for *this* graph. $\omega(G)$ and $\chi(G)$ can be found in polynomial time by computing $\bar{\vartheta}(G)$. Furthermore. the embedding given by Lemma 2.5 will have minimal radius.

The smallest graph G with $\omega(G) < \chi(G)$ is the 5-cycle C_5. Finding a spherical unit-distance embedding of minimum radius for C_5 should prove to be interesting because by Theorem 2.7. we know that $\frac{1}{2} \leq r(C_5) \leq \sqrt{\frac{1}{3}}$.

In this case. solving the semidefinite program given by Theorem 3.3 gives:

$$A = \begin{bmatrix} a & b & c & c & b \\ b & a & b & c & c \\ c & b & a & b & c \\ c & c & b & a & b \\ b & c & c & b & a \end{bmatrix} \qquad \text{where} \quad \begin{aligned} a &= \frac{1}{2} - \frac{1}{10}\sqrt{5} = 0.276393\ldots \\ b &= -\frac{1}{10}\sqrt{5} = -0.223606\ldots \\ c &= \frac{3}{20}\sqrt{5} - \frac{1}{4} = 0.085410\ldots \end{aligned}$$

Interestingly. $rank(A) = 2$, therefore the spherical unit-distance embedding of C_5 of minimum radius lies in two dimensions. It is shown in Figure 3.1.

$$r(C_5) = \sqrt{a} = 0.525731\ldots$$

$$\bar{\vartheta}(C_5) = \frac{1}{1-2a} = \sqrt{5} = 2.236067\ldots$$

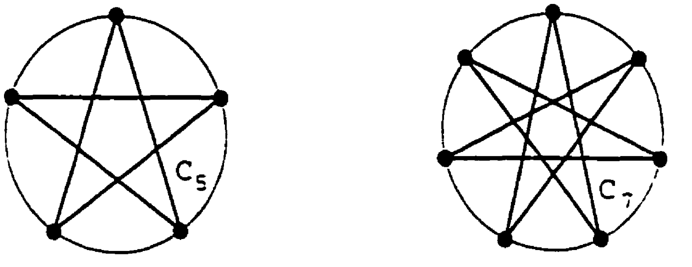


Figure 3.1: Optimal spherical unit-distance embeddings of C_5 and of C_7

3.5 Other applications of semidefinite programming

Semidefinite programming has been used by Lovász [Lov79] to compute the Shannon capacity of a graph. It is also the only known way of computing a maximum clique and a minimum coloring in perfect graphs in polynomial time [GLS81].

Historically, the first use of semidefinite programming in combinatorial optimization was by Goemans and Williamson [GW94], [GW95] to obtain improved approximability results for the MAX CUT and MAX 2SAT problems.

These results inspired Karger, Motwani and Sudan [KMS94] to adapt the technique to GRAPH COLORING, which in this case corresponds to finding a unit-distance spherical embedding of minimum radius, as described in Chapter 2.

To date, semidefinite programming has also been successfully applied to MAX k-CUT [FJ97], MAX BISECTION [FJ97] and GRAPH PARTITIONING [WZ96]. Furthermore, for MAX CUT and GRAPH COLORING, one can use the specialized procedure of [PL96] that runs significantly faster that a general purpose semidefinite program solver, if the tolerance ϵ is not too small.

Chapter 4

Finding Large Independent Sets

In this chapter, we will see how the graph embedding of the previous chapter can be used to find a large independent set in a k-colorable graph. This first step is already interesting because in general it is hard to find large independent sets.

Sections 4.4 and 4.5 each present an additional independent set algorithm that does not require any result from the previous chapter. These algorithms have different properties, and they nicely complement the independent set algorithms that are based on graph embeddings.

4.1 Generating random n-dimensional vectors

The algorithms in the next two sections require random n-dimensional vectors. They are *randomized algorithms*. A good way to generate a random n-dimensional vector in practice is through the *multidimensional normal distribution*.

Theorem 4.1 *Let $z_1, z_2, \dots, z_n \sim N(0,1)$ be independent, normally distributed random variables with mean 0 and variance 1. Then the distribution of the random vector $z = (z_1, z_2, \dots, z_n)^T$ is spherically symmetric.*

Proof. Write $\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ for the density of the standard normal distribution. The joint density of the n-dimensional standard normal distribution is:

$$\begin{aligned} f(x_1, x_2, \dots, x_n) &= \prod_{i=1}^{n} \phi(x_i) = \left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x_1^2}{2}}\right) \cdots \left(\frac{1}{\sqrt{2\pi}} e^{-\frac{x_n^2}{2}}\right) \\ &= \frac{1}{(2\pi)^{\frac{n}{2}}} e^{-\frac{x_1^2+x_2^2+\dots+x_n^2}{2}} = \frac{1}{(2\pi)^{\frac{n}{2}}} e^{-\frac{\|x\|^2}{2}} \end{aligned}$$

Note that $f(x_1, x_2, \dots, x_n)$ depends *only* on the length $\|x\|$ of the vector x, therefore the distribution must be spherically symmetric.

As a practical note, there is a very nice way to obtain two independent standard normal random variables $z_1, z_2 \sim N(0,1)$ from two independent uniform random variables $u_1, u_2 \sim UNIF(0,1)$ on a computer:

$$\begin{aligned} \text{Let} \quad t &= \sqrt{-2 \ln u_1} \\ a &= 2\pi u_2 \\ \text{Put} \quad z_1 &= t \cos a \\ z_2 &= t \sin a \end{aligned}$$

4.2 Hyperplane partitions

Let $G = (V, E)$ be a graph with n vertices and let $\vartheta = \vartheta(G)$. Apply Theorem 3.3 to get a spherical unit-distance embedding $\varphi : V \to \mathbf{R}^n$ of radius $r = \sqrt{\frac{\vartheta - 1}{2\vartheta}}$. This embedding is compact, yet pairs of vertices that are joined by an edge are forced to be "far" apart. This section and the following exploit this observation to construct large independent sets. We start with the *hyperplane partitions* approach, which gives a weaker result, but is easier to explain.

Definition 4.1 *Given a graph G and an embedding φ as above, a random hyperplane partition of the vertex set V is defined as follows. Let $z = (z_1, z_2, \dots, z_n)^T$ be a random n-dimensional vector. Then z partitions the vertex set as $V = V_1 \cup V_2$ in a natural way:*

$$V_1 = \{v \in V : \varphi(v) \cdot z > 0\}$$
$$V_2 = \{v \in V : \varphi(v) \cdot z < 0\}$$

Note that the set $\{v \in V : \varphi(v) \cdot z = 0\}$ is empty with probability 1. If we imagine the hyperplane orthogonal to z and going through the origin, then this classify vertices according to which side of the hyperplane they lie on. This explains the name hyperplane partition.

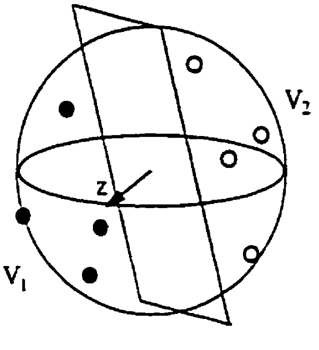


Figure 4.1: The partition $V = V_1 \cup V_2$ of the vertex set by z

As an aside, there is an application (Theorem 4.3) of hyperplane partitions to *clique breaking*. This is interesting because it is a special case of the NP-complete problem SET SPLITTING.

Proposition 4.2 (Clique breaking) *Let $G = (V, E)$ be a k-colorable graph. There exists a partition $V = V_1 \cup V_2$ such that the graphs induced by V_1 and V_2 are both k-clique free.*

Proof. Let $f : V \to \{1, 2, \ldots, k\}$ be a proper k-coloring of G. There are many possible solutions. One of them is to put

$$V_1 = \{v \in V : f(v) = 1\}$$
$$V_2 = \{v \in V : f(v) = 2, 3, \ldots, \text{or}\, k\}$$

The induced subgraphs are respectively 1 and $k-1$ colorable, therefore they must be k-clique free.

Theorem 4.3 *A "clique breaking" partition as described in Proposition 4.2 can be found in polynomial time.*

Proof. Compute $\bar{\vartheta}(G)$. By Theorem 2.8, $\bar{\vartheta}(G) \leq k$. If $\bar{\vartheta}(G) < k$ then by Theorem 2.8, G contains no k-clique, so no partition is necessary. Else $\bar{\vartheta}(G) = k$. Find a spherical unit-distance embedding φ of radius $r = \sqrt{\frac{k-1}{2k}}$. Let z be a random n-dimensional vector and compute an hyperplane partition $V = V_1 \cup V_2$ from z as in Definition 4.1. We claim that V_1 and V_2 contain no k-clique. We proceed by contradiction. Suppose without loss of generality that V_1 contains a k-clique $K \subseteq V_1$. Since $\varphi(v) \cdot z > 0 \quad \forall v \in K$, we have that $\frac{1}{k} \sum_{v \in K} \varphi(v) \cdot z > 0$. But this is impossible because the center of a regular simplex with k vertices and unit side must be the origin when it is embedded on the surface of a hypersphere of radius $r = \sqrt{\frac{k-1}{2k}}$.

Note. If G is 2-colorable, then the partition found by Theorem 4.3 is 2-clique free, that is, it is *edge* free. By assigning one color to V_1 and one color to V_2, we have found a 2-coloring of G in polynomial time. (There is no surprise here, but it is nice to see that our complex approach can solve this basic problem.)

Now, the application of hyperplane partitions to finding independent sets in 3-colorable graphs.

Definition 4.2 *Let $G = (V, E)$ be a graph. A subset $S \subseteq V$ is said to be an independent set of size $|S|$ in G iff $(v_i, v_j) \notin E \quad \forall v_i, v_j \in S$, $v_i \neq v_j$. Note that this definition is in some sense the* opposite *of the Definition 2.7 of a* clique.

Lemma 4.4 *In a spherical unit-distance embedding of radius $\frac{1}{\sqrt{3}}$, the probability that the two endpoints of an edge are on the same side of a random hyperplane is $\frac{1}{3}$.*

Proof. Let $e = (v_i, v_j)$ be an edge of the embedding. Consider the two-dimensional plane going through $x_i = \varphi(v_i)$, $x_j = \varphi(v_j)$, and the origin. The intersection of this plane with a random hyperplane is a line L going through the origin. Furthermore, since the hyperplane is random, the orientation of L on the two-dimensional plane must also be uniform. See Figure 4.2. The probability that the two endpoints are on the same side of the hyperplane can be computed geometrically: it is $\frac{1}{3}$.

Theorem 4.5 (Markov's inequality) *Let X be a random variable that takes only non-negative values. Then for any $a > 0$,*

$$P(X \geq a) \leq \frac{E(X)}{a}$$

When a graph is *very* sparse, it is very easy to find an independent set directly. The following proposition will be useful as the final step of other methods. It also gives our first independent set algorithm.

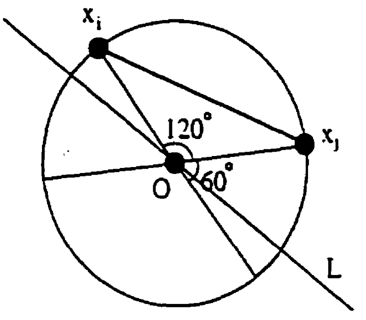


Figure 4.2: The probability that a random line through the origin does *not* cut a segment spanning 120° is $\frac{1}{3}$

Proposition 4.6 *Let $G = (V, E)$ be a graph with n vertices and m edges such that $m \le \frac{1}{2}n$. We can find an independent set of size $\ge \frac{1}{2}n$ in polynomial time.*

Proof. For every edge, remove one of its endpoint from the vertex set. Since there are at most $\frac{1}{2}n$ edges, at most $\frac{1}{2}n$ vertices will be removed. In the process, every edge will be destroyed and at least $\frac{1}{2}n$ vertices will remain. Therefore, we get an independent set of size at least $\frac{1}{2}n$.

Theorem 4.7 (Karger-Motwani-Sudan [KMS94] [KMS98]) *Let $G = (V, E)$ be a 3-colorable graph with n vertices and m edges such that $m \ge \frac{1}{4}n$. We can find an independent set of size $\ge 0.104\, n^{1.630} m^{-0.631}$ in polynomial time with probability at least $\frac{1}{2}$.*

Proof. Find a unit-distance embedding φ of G of radius $r = \frac{1}{\sqrt{3}}$ in $\mathbf{R}^n$. This is possible since G is 3-colorable. Let $c = \lceil \log_3 \frac{4m}{n} \rceil$. Use c independent random hyperplanes to partition V into 2^c subsets. By Lemma 4.4, the probability that an edge "resists" these c cuts and have both endpoints in the same subset is $\left(\frac{1}{3}\right)^c$. Let m' be the number of such edges. The expected value of m' is:

$$E(m') = m\left(\frac{1}{3}\right)^c \leq m\left(\frac{1}{3}\right)^{\log_3 \frac{4m}{n}} = m\frac{n}{4m} = \frac{n}{4}.$$

By Markov's inequality, $P(m' < \frac{n}{2}) \geq \frac{1}{2}$. If $m' \geq \frac{n}{2}$ then exit with a failure. Otherwise. build a subgraph H of G by removing one endpoint from each edge. By construction. H has at least $\frac{n}{2}$ vertices, and every edge of H has endpoints in different subsets. By the pigeonhole principle, one of the 2^c subsets must contain at least $\frac{n}{2}/2^c$ vertices. This subset will be our independent set. It has size $\geq \frac{n}{2}\left(\frac{1}{2}\right)^c \geq \frac{n}{4}\left(\frac{1}{2}\right)^{\log_3 \frac{4m}{n}} = \frac{n}{4}\left(\frac{1}{2}\right)^{\log_2 \left(\frac{4m}{n}\right)^{\log_3 2}} = \frac{n}{4}\left(\frac{n}{4m}\right)^{\log_3 2} = \frac{1}{4 \cdot 4^{\log_3 2}} n^{1+\log_3 2} m^{-\log_3 2} \geq 0.104\, n^{1.630} m^{-0.631}$.

Note that by repeating the above randomized algorithm t times. the probability of failure drops to $\left(\frac{1}{2}\right)^t$.

In this section we considered 3-colorable graphs only. but the algorithm and its analysis can be generalized to k-colorable graphs in a straightforward way.

4.3 Vector projection

Instead of using many hyperplane cuts through the origin. the following method uses only *one* hyperplane which is at some carefully chosen distance to the origin. The method considers only the vertices lying in the *small* half of the hypersphere.

This is the most complex section of the thesis, but the main result of the section (Theorem 4.11) is key to get good colorings later, so proving it is justified.

Lemma 4.8 *Let $x \in \mathbf{R}^n$ be a unit vector. Let $z = (z_1, z_2, \ldots, z_n)^T$ be a random variable having the n-dimensional standard normal distribution. Then $x \cdot z \sim \mathcal{N}(0, 1)$.*

Proof. Note that since $\|x\| = 1$, $x \cdot z$ is the length of the projection of z along x. Since the distribution of z is spherically symmetric, the distribution of $x \cdot z$ does not depend on the direction of x. Therefore, $x \cdot z \sim (1, 0, \ldots, 0)^T \cdot z = z_1 \sim N(0, 1)$.

Definition 4.3 *Write $N(x)$ for the tail of the standard normal distribution. i.e.*

$$N(x) = \int_x^\infty \phi(t)dt$$

Lemma 4.9 (folklore) *For every $x > 0$, $\left(\frac{1}{x} - \frac{1}{x^3}\right)\phi(x) < N(x) < \frac{1}{x}\phi(x)$.*

Proof.

$$\begin{aligned}
& \left(1 - \frac{3}{t^4}\right)\phi(t) < \phi(t) < \left(1 + \frac{1}{t^2}\right)\phi(t) \quad \forall t > 0 \\
\Rightarrow \quad & \int_x^\infty \left(1 - \frac{3}{t^4}\right)\phi(t)dt < \int_x^\infty \phi(t)dt < \int_x^\infty \left(1 + \frac{1}{t^2}\right)\phi(t)dt \quad \forall x > 0 \\
\Rightarrow \quad & \left(\frac{1}{x} - \frac{1}{x^3}\right)\phi(x) < N(x) < \frac{1}{x}\phi(x) \quad \forall x > 0
\end{aligned}$$

Lemma 4.10 *Fix any $\alpha < 1$. Then $\frac{1}{x} - \frac{1}{x^3} \geq \frac{\alpha}{x}$ for $x \geq \sqrt{\frac{1}{1-\alpha}}$*

Proof. This is embarrassingly straightforward.

$$\begin{aligned}
x \geq \sqrt{\frac{1}{1-\alpha}} \quad & \Rightarrow \quad x > 0 \quad \text{and} \quad x^2 \geq \frac{1}{1-\alpha} \\
& \Rightarrow \quad \frac{1-\alpha}{x} \geq \frac{1}{x^3} \quad \left(\text{multiplying each side by } \frac{1-\alpha}{x^3}\right) \\
& \Rightarrow \quad \frac{1}{x} - \frac{1}{x^3} \geq \frac{\alpha}{x}
\end{aligned}$$

The constant 0.006 that is derived in the next theorem is in no way fundamental. For instance, it depends on the number 2.61 which is somewhat arbitrary. Deriving a constant is not usually done in theoretical works but it makes the result much more concrete.

Theorem 4.11 (Karger-Motwani-Sudan [KMS98]) *Let $G = (V, E)$ be a graph with n vertices and m edges such that $m \geq 2.61\, n$. Let $\bar{\vartheta} \geq \max(\bar{\vartheta}(G), 3)$. We can find an independent set of size $\geq b = 0.006\, n^{2-\frac{3}{\bar{\vartheta}}} m^{\frac{2}{\bar{\vartheta}}-1} \left(\ln \frac{4m}{n}\right)^{-\frac{1}{2}}$ in polynomial time with probability at least $\frac{b}{n}$.*

Proof. Find a spherical unit-distance embedding φ of G of radius $r = \sqrt{\frac{\bar{\vartheta}-1}{2\bar{\vartheta}}}$ in $\mathbf{R}^n$. In this case it is important that $z = (z_1, z_2, \ldots, z_n)^T$ have the n-dimensional standard normal distribution.

$$\text{Define} \quad S = \left\{ v_i \in V : \frac{1}{r}\varphi(v_i) \cdot z \geq c \right\}$$

$$\text{where} \quad c = \sqrt{\frac{2(\bar{\vartheta}-2)}{\bar{\vartheta}} \ln \frac{4m}{n}} \quad \text{is a "magic" number}$$

Note that $\bar{\vartheta} \geq 3$ and $m \geq 2.61\, n \Rightarrow c \geq 1.25$. Let n' and m' be the number of vertices and edges in the subgraph induced by S. As a first step, we derive a lower bound for $E(n')$. Since $\frac{1}{r}\varphi(v_i)$ is a unit vector, by Lemma 4.8 $\frac{1}{r}\varphi(v_i) \sim N(0, 1)$. So for any $v_i \in V$,

$$\begin{aligned}
P[v_i \in S] &= P\left[\frac{1}{r}\varphi(v_i) \cdot z \geq c\right] = N(c) \\
&\geq \left(\frac{1}{c} - \frac{1}{c^3}\right) \phi(c) \\
&\geq \frac{0.36}{c} \frac{1}{\sqrt{2\pi}} e^{-\frac{c^2}{2}} \quad \text{since} \quad c \geq 1.25 = \sqrt{\frac{1}{1-0.36}} \\
&= \frac{0.36}{c} \frac{1}{\sqrt{2\pi}} e^{-\frac{\bar{\vartheta}-2}{\bar{\vartheta}} \ln \frac{4m}{n}} \\
&= \frac{0.36}{c} \frac{1}{\sqrt{2\pi}} \left(\frac{4m}{n}\right)^{\frac{2}{\bar{\vartheta}}-1}
\end{aligned}$$

$$\Rightarrow E(n') = nP[v_i \in S] \geq \frac{0.09}{c} \frac{1}{\sqrt{2\pi}} 4^{\frac{2}{\bar{\vartheta}}} n^{2-\frac{2}{\bar{\vartheta}}} m^{\frac{2}{\bar{\vartheta}}-1}$$

Next, we derive an upper bound for $E(m')$. Let $(v_i, v_j) \in E$. Since we are using the standard normal distribution, the calculation will reduce to a 2-dimensional problem in the plane that goes through $x_i = \varphi(v_i)$, $x_j = \varphi(v_j)$ and the origin.

$$\begin{aligned}
P[v_i \in S \text{ and } v_j \in S] &= P\left[\frac{1}{r}x_i \cdot z \geq c \quad \text{and} \quad \frac{1}{r}x_j \cdot z \geq c\right] \\
&= P[z \text{ falls in region R}] \quad \text{(see Figure 4.3)} \\
&\leq P[z \text{ falls in region U}] \quad \text{(see Figure 4.3)} \\
&= N(ac) \quad \text{where} \quad a = \sqrt{\frac{2(\bar\vartheta - 1)}{\bar\vartheta - 2}} \geq \sqrt{2} \\
&\leq \frac{1}{ac}\phi(ac) \leq \frac{0.72}{c}\phi(ac) \\
&= \frac{0.72}{c}\frac{1}{\sqrt{2\pi}}e^{-\frac{a^2(\bar\vartheta-2)}{\bar\vartheta}\ln\frac{4m}{n}} \\
&= \frac{0.72}{c}\frac{1}{\sqrt{2\pi}}\left(\frac{4m}{n}\right)^{\frac{2}{\bar\vartheta}-2}
\end{aligned}$$

$$\Rightarrow E(m') = mP[v_i \in S \text{ and } v_j \in S] \leq \frac{0.045}{c}\frac{1}{\sqrt{2\pi}}4^{\frac{2}{\bar\vartheta}}n^{2-\frac{3}{\bar\vartheta}}m^{\frac{2}{\bar\vartheta}-1}$$

This upper bound is exactly *half* the lower bound that we derived for $E(n')$.

$$\begin{aligned}
E(n') - E(m') &\geq (0.09 - 0.045)\frac{1}{c}\frac{1}{\sqrt{2\pi}}4^{\frac{2}{\bar\vartheta}}n^{2-\frac{3}{\bar\vartheta}}m^{\frac{2}{\bar\vartheta}-1} \\
&= 0.045\sqrt{\frac{\bar\vartheta}{2(\bar\vartheta-2)}}\left(\ln\frac{4m}{n}\right)^{-\frac{1}{2}}\frac{1}{\sqrt{2\pi}}4^{\frac{2}{\bar\vartheta}}n^{2-\frac{3}{\bar\vartheta}}m^{\frac{2}{\bar\vartheta}-1} \\
&\geq 0.012\, n^{2-\frac{3}{\bar\vartheta}}m^{\frac{2}{\bar\vartheta}-1}\left(\ln\frac{4m}{n}\right)^{-\frac{1}{2}}
\end{aligned}$$

We will use Markov's inequality to show that $n' - m'$ will be within half of its expected value with some reasonable probability. Let $b = 0.006\, n^{2-\frac{3}{\bar\vartheta}}m^{\frac{2}{\bar\vartheta}-1}\left(\ln\frac{4m}{n}\right)^{-\frac{1}{2}}$ (half of the bound above). Define the random variable $X = n + m' - n' \geq 0$. Note that our result above translates into $E(X) \leq n - 2b$.

$$\begin{aligned}
P[n' - m' > b] &= P[m' - n' < -b] = P[X < n - b] \\
&= 1 - P[X \geq n - b] \geq 1 - \frac{E(X)}{n - b} \\
&\geq 1 - \frac{n - 2b}{n - b} = \frac{b}{n - b} \geq \frac{b}{n}
\end{aligned}$$

If $n' - m' \leq b$ then exit with a failure. Otherwise, build a subgraph H starting from S and removing one endpoint from each edge of the graph induced by S. By construction, H will be an independent set with at least b vertices.

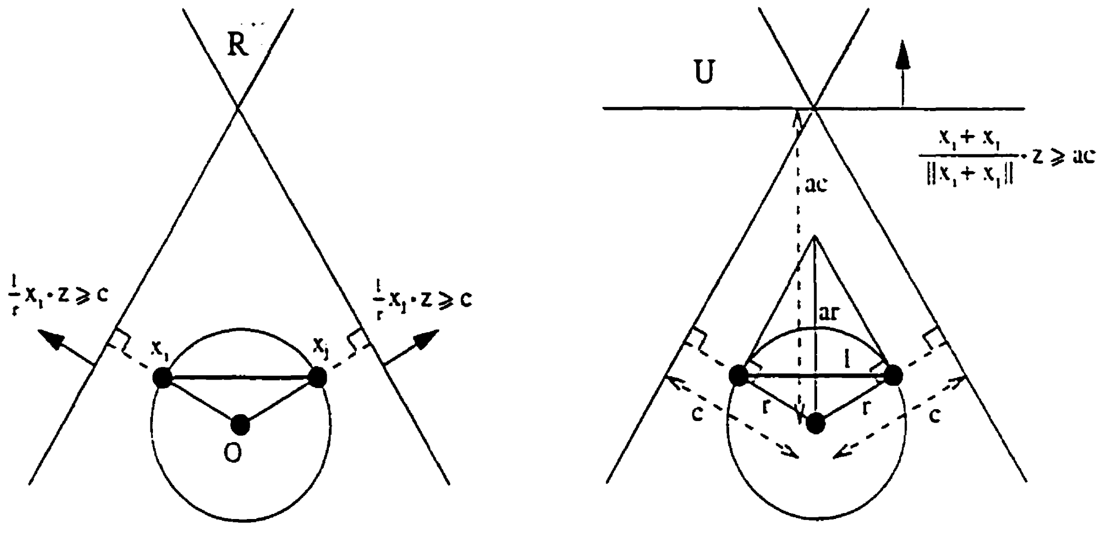


Figure 4.3: Finding both endpoints of an edge in S is surprising because it implies that the projection of z onto the plane going through x_i, x_j and O falls into region R. It is easier to evaluate the (greater) probability that the projection of z falls into region U. The computation of a (omitted) can be done geometrically from the drawing on the right, and using Definition 2.9.

Note that by repeating the above randomized algorithm a polynomial number of times, the probability of failure drops exponentially.

We have the following specialization of the theorem to 3-colorable graphs:

Corollary 4.12 *Let $G = (V, E)$ be a 3-colorable graph with n vertices and m edges such that $m \geq 2.61\,n$. We can find an independent set of size* $b = 0.006\, n^{4/3} m^{-1/3} \left(\ln \frac{4m}{n}\right)^{-1/2}$ *in polynomial time with probability at least $\frac{b}{n}$.*

4.4 Greedy coloring

Definition 4.4 *Let $G = (V, E)$ be a graph. For every vertex $v \in V$, the degree $d(v)$ of the vertex v is the number of neighbors of v in G. The maximum degree in a graph is defined to be $\Delta(G) = \max_{v \in V} d(v)$.*

Greedy coloring is a very simple coloring algorithm: At every step, we pick an arbitrary uncolored vertex v and color it using the smallest-indexed color that is not already used by some neighbor of v. Since every vertex has at most $\Delta(G)$ neighbors, greedy coloring will use at most $\Delta(G) + 1$ colors.

We show how the greedy coloring algorithm can be turned into an independent set algorithm.

Theorem 4.13 *In any graph $G = (V, E)$ with n vertices and m edges such that $m \geq \frac{1}{2}n$, we can find an independent set of size $\geq \frac{1}{12}n^2m^{-1}$.*

Proof. Partition V into two sets.
Let $S = \{v \in V : d(v) < \frac{4m}{n}\}$.
Let $T = \{v \in V : d(v) \geq \frac{4m}{n}\}$.
Then $2m = \sum_{v \in V} d(v) \geq \sum_{v \in T} d(v) \geq |T|\frac{4m}{n}$
$\Rightarrow |T| \leq \frac{1}{2}n$, so $|S| = n - |T| \geq \frac{1}{2}n$
Let H be the graph induced by S. $\Delta(H) < \frac{4m}{n}$, so the greedy coloring algorithm will color H by using at most $\frac{4m}{n} + 1$ colors. By the pigeonhole principle, at least $\frac{\frac{1}{2}n}{\frac{4m}{n}+1}$ vertices will get the same color. This gives us an independent set of size $\geq \frac{n^2}{2(4m+n)} \geq \frac{n^2}{2(4m+2m)} = \frac{1}{12}n^2m^{-1}$

4.5 Wigderson's technique

We present Wigderson's algorithm, specialized for 3-colorable graphs.

Theorem 4.14 (Wigderson [Wig83]) *Let $G = (V, E)$ be a 3-colorable graph with n vertices and m edges. We can find an independent set of size $\geq n^{-1}m$.*

Proof. $2m = \sum_{v \in V} d(v)$, so by the pigeonhole principle, $\exists v \in V$ such that $d(v) \geq \frac{2m}{n}$. Let $N(v)$ be the neighborhood of v. Since G is 3-colorable, the subgraph H induced by $N(v)$ is 2-colorable. Use a polynomial time algorithm to 2-color H. By the pigeonhole principle, one of the color classes will contain at least $\frac{1}{2}\frac{2m}{n} = \frac{m}{n}$ vertices. This gives our independent set.

Note that if $m < n$ then the argument still goes through, but the result becomes "we can find an independent set of size 1", which is trivial. So we might as well require that $m \geq n$.

Chapter 5

Approximate Graph Coloring

5.1 Heuristics vs approximation algorithms

Let G be a graph. Finding a proper coloring of G with the minimum number $\chi(G)$ of colors is a hard problem (it is NP-complete [Kar72], [GJ79]). However, in practical applications, a proper coloring with slightly more than $\chi(G)$ colors can still be useful. Practical applications of graph coloring are mostly resource allocation problems. For example, if V is a set of equally long tasks and E is the set of pairs of conflicting tasks, then a small coloring of the graph $G = (V, E)$ will give a way of scheduling the tasks that avoids conflicts and minimizes the time needed to run all the tasks.

It is not hard to invent a *heuristic* that will color most graphs with few colors. However, it is usually not hard either to find an example for which the heuristic will fail to give a good coloring. (Such examples can be found in [Joh74b].) It is much more interesting (at least theoretically) when a heuristic comes with a performance guarantee. When an algorithm has such a guarantee and runs in polynomial time it is called an *approximation algorithm*.

5.2 From independent sets to colorings

From any algorithm that finds large independent sets, we can derive a graph coloring algorithm as follows:

```
while G is non-empty
  find an independent set I of vertices in G
  color these vertices using one new color
  replace G with G - I
end while
```

The following proposition gives a guarantee on the quality of the coloring in terms of the quality of the independent set algorithm.

Proposition 5.1 *Let G be a graph with n vertices. Suppose that we have an algorithm that can find an independent set of size $\geq cN^a$ for any N-vertex subgraph of G (for some constants $0 < a, c < 1$). Then the technique above will find a proper coloring of G with at most $\frac{1}{c(1-a)}n^{1-a}$ colors.*

Proof. Note that if at some iteration G has n vertices, then at the next iteration G will have at most $\lfloor n - cn^a \rfloor$ vertices. Since $\lfloor n - cn^a \rfloor \leq n - 1$, the algorithm will terminate. Let $s(n)$ be the maximum number of iterations the algorithm can possibly take when its input graph has n vertices. We prove by strong induction on n that $s(n) \leq \frac{1}{c(1-a)}n^{1-a}$.

Basis: if $n = 0$ then $s(n) = 0 \leq \frac{1}{c(1-a)}0^{1-a}$ so the proposition holds.

Induction step: The algorithm provides a recurrence relation:

$$s(n) = 1 + s(\lfloor n - cn^a \rfloor) \qquad \text{the recurrence relation}$$

$$
\begin{aligned}
&\leq 1+\frac{1}{c(1-a)}(\lfloor n-cn^a\rfloor)^{1-a} \qquad \text{by induction}\\
&\leq 1+\frac{1}{c(1-a)}(n-cn^a)^{1-a}\\
&= 1+\frac{1}{c(1-a)}n^{1-a}(1-cn^{a-1})^{1-a}\\
&\leq 1+\frac{1}{c(1-a)}n^{1-a}(1-(1-a)cn^{a-1}) \quad \text{since } 1-a<1\\
&= 1+\frac{1}{c(1-a)}n^{1-a}-\frac{c(1-a)}{c(1-a)}\\
&= \frac{1}{c(1-a)}n^{1-a}
\end{aligned}
$$

Since we use exactly one color per iteration, an n-vertex graph will be colored with at most $\frac{1}{c(1-a)}n^{1-a}$ colors.

5.3 Combining independent set algorithms

All the independent set algorithms that we have so far for 3-colorable graphs (randomized or not) are reviewed below.

method	reference	condition	size of independent set
Direct	Proposition 4.6	$m \leq \frac{1}{2}n$	$\frac{1}{2}n$
Hyperplane partitions	Theorem 4.7	$m \geq \frac{1}{4}n$	$0.104\, n^{1.630} m^{-0.631}$
Vector projection	Corollary 4.12	$m \geq 2.61\, n$	$0.006\, n^{4/3} m^{-1/3} \left(\ln \frac{4m}{n}\right)^{-1/2}$
Greedy coloring	Theorem 4.13	$m \geq \frac{1}{2}n$	$\frac{1}{12}n^2 m^{-1}$
Wigderson's technique	Theorem 4.14	$m \geq n$	$n^{-1}m$

Some algorithms are good for sparse graphs, others are good for dense graphs. To make this idea precise, consider a graph with $m = n^a$ edges. The above algorithms all return an independent set of size $\tilde{O}(n^b)$ for some b. The relation between a and b

is as follows (shown graphically in Figure 5.1):

method	condition	relation
Direct	$0 \leq a \leq 1$	$b = 1$
Hyperplane partitions	$1 \leq a \leq 2$	$b = 1.630 - 0.631\,a$
Vector projection	$1 \leq a \leq 2$	$b = \frac{4}{3} - \frac{1}{3}a$
Greedy coloring	$1 \leq a \leq 2$	$b = 2 - a$
Wigderson's technique	$1 \leq a \leq 2$	$b = -1 + a$

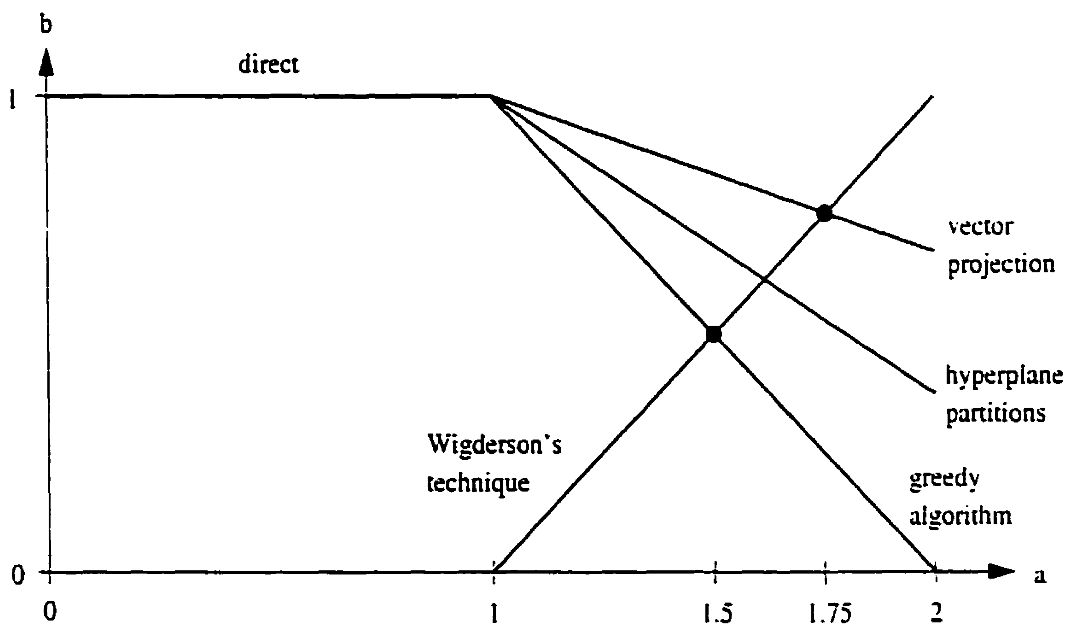


Figure 5.1: For $m = n^a$, the independent set found is of size $\tilde{\Omega}(n^b)$

We can see from Figure 5.1 that

- The *direct* method is the only method available in the range $0 \leq a \leq 1$.
- *Vector projection* is the best method in the range $1 \leq a \leq 1.75$
- *Wigderson's technique* is the best method in the range $1.75 \leq a \leq 2$

- *Hyperplane partitions* and *greedy coloring* are strictly inferior to *vector projection*.

So by combining the algorithms properly, we obtain:

Proposition 5.2 *Let $G = (V, E)$ be any 3-colorable graph with n vertices and m edges. We can find an independent set of size $\Omega(n^{0.75}(\ln n)^{-1/2})$.*

Proof.

- if $m \leq 0.5\,n$, use the *direct* method to obtain an independent set of size $\Omega(n)$.
- if $0.5\,n < m \leq 2.61\,n$, use *greedy coloring* to obtain an independent set of size $\Omega(n)$.
- if $2.61 < m \leq n^{1.75}$, use *vector projection* to obtain an independent set of size $\Omega(n^{0.75}(\ln n)^{-\frac{1}{2}})$.
- if $n^{1.75} < m$, use *Wigderson's technique* to obtain an independent set of size $\Omega(n^{0.75})$.

In every case, the size of the independent set is $\Omega(n^{0.75}(\ln n)^{-\frac{1}{2}})$, therefore the result follows. Note that we use *greedy coloring* only to fill the small gap $0.5\,n < m \leq 2.61\,n$, which cannot be seen in Figure 5.1.

Corollary 5.3 (Karger-Motwani-Sudan [KMS98]) *Let $G = (V, E)$ be any 3-colorable graph with n vertices. We can color G using $O(n^{1/4}\ln^{1/2} n)$ colors.*

Proof. This follows by combining Proposition 5.2 and Proposition 5.1. In fact, the $(\ln n)^{-1/2}$ factor in Proposition 5.2 is a small nuisance, and technically, we need to apply a generalization of Proposition 5.1.

Historically, there was a time [Wig83] when only the *direct* method, the *greedy algorithm* method, and *Wigderson's technique* were known. It's easy to see from Figure 5.1 that in that case:

- The *direct* method was the only method available in the range $0 \leq a \leq 1$.
- *Greedy coloring* was the best method in the range $1 \leq a \leq 1.5$
- *Wigderson's technique* was the best method in the range $1.5 \leq a \leq 2$

So together, these three methods can be used to find an independent set of size $\Omega(n^{1/2})$ in any 3-colorable graph. By Proposition 5.1, this allows us to color any 3-colorable graph with $O(n^{1/2})$ colors, hence repeating the result of Wigderson [Wig83].

5.4 Blum's improvement

We see from Figure 5.1 that *hyperplane partitions* and *vector projection* are in some sense improvements to *greedy coloring*. Blum attacked the problem from the other direction, trying to improve *Wigderson's technique*. He found two interesting improvements which, unfortunately, are not as simple as adding two new independent set algorithms to our tool-box. However, these improvement can still be represented as new lines in our figure.

Theorem 5.4 (Blum 1 [Blu94] [BK97]) *If there is a constant "a" and an algorithm that finds independent sets of size* $\tilde{\Omega}(n^{-\frac{1}{3}+\frac{2}{3}a})$ *for 3-colorable graphs with* $m \leq n^a$, *then there is an algorithm which, for any 3-colorable graph G, will output either:*
- *2 vertices which must have the same color in any legal 3-coloring of G, or:*
- *an independent set of size* $\tilde{\Omega}(n^{-\frac{1}{3}+\frac{2}{3}a})$.

Theorem 5.5 (Blum 2 [Blu94] [BK97]) *Same as above, but with* $\tilde{\Omega}(n^{-\frac{1}{5}+\frac{3}{5}a})$.

Note that finding two vertices which must have the same color in any legal 3-coloring of G is in fact much more desirable than finding independent sets, because we can collapse the two vertices together and start again with one less vertex.

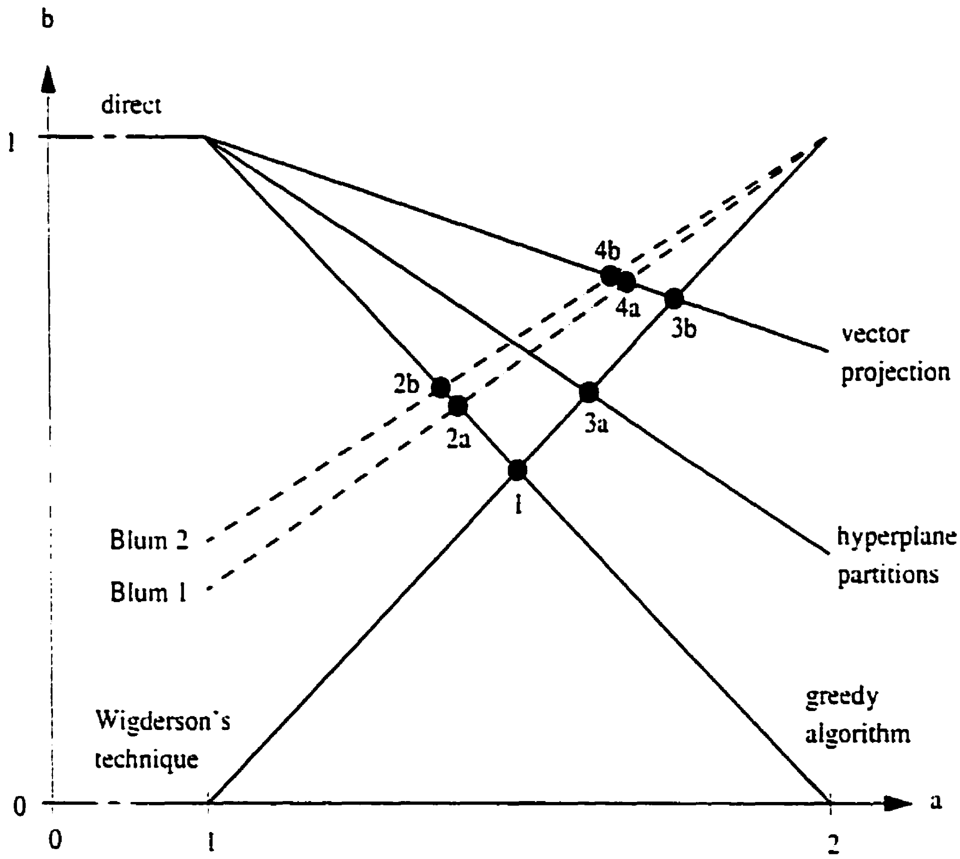


Figure 5.2: For $m = n^a$, the independent set found is of size $\tilde{\Omega}(n^b)$. This is not technically true for Blum 1 and Blum 2, but these algorithms behave similarly.

Each marked intersection point in Figure 5.2 corresponds to a coloring algorithm that appeared in the literature as the result of combining two methods (whose names are given by the labels on the two lines creating the intersection).

intersection	coloring obtained	reference
1	$O(n^{1/2})$	[Wig83]
2a	$\tilde{O}(n^{2/5})$	[Blu94]
2b	$\tilde{O}(n^{3/8})$	[Blu94]
3a	$O(n^{0.386})$	[KMS94]
3b	$\tilde{O}(n^{1/4})$	[KMS94]
4a	$\tilde{O}(n^{2/9})$	[BK97]
4b	$\tilde{O}(n^{3/14})$	[BK97]

Basically, all the components were present in 1994, but the way in which these components should be combined to give the best coloring wasn't published until 3 years later. In our framework this amounts to noticing the two intersections 4a and 4b at the top of Figure 5.2.

Chapter 6

Conclusion

As stated in Section 1.3, the best result on the *inapproximability* of coloring 3-colorable graphs is a result by Khanna, Linial, and Safra [KLS92], which says that it is NP-complete to color a 3-colorable graph with 4 colors. Thus a polynomial time algorithm that would color any 3-colorable graph with 5 colors is *still* possible in our current state of knowledge.

Hence there is the obvious open problem:
Can we improve the best bound for 3-colorability, currently $\tilde{O}(n^{3/14})$? In particular, can we color any 3-colorable graph using a constant number of colors in polynomial time? Or maybe using a constant times $\log n$ *colors? Or is there an inapproximability result that would forbid one or both of these possibilities?*

On the other hand, it is also possible that progress in the near future will be made by devising better independent set algorithms that insert new cuts in the diagram of Figure 5.2. In that case, the framework laid out in Section 5.3 will prove to be useful.

Spherical unit-distance embeddings have nice properties, thus one additional question would be:

Can compact embeddings of graphs be used for applications other than approximate graph coloring?

Also, I find that an algorithm that uses semidefinite programming is less attractive than a discrete algorithm because the former involves computing with imprecise real numbers, and the implementations of semidefinite programming solvers are still slow. Therefore:

Can we prove Corollary 4.12 (or something at least as strong) without using semidefinite programming?

6.1 Acknowledgments

I would like to thank Sue Whitesides for introducing me to NP-completeness, approximation algorithms and graph theory, and for her flexibility concerning the choice of the thesis topic. Also, I would like to thank the School of Computer Science and the Mathematics department of McGill University for the very fine environment during my stay there. Finally, I would like to thank Donald Knuth for inventing TEX, which made typing this thesis a pleasant and efficient enterprise.

This research was supported by NSERC.

Bibliography

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

[BGS95] M. BELLARE, O. GOLDREICH, M. SUDAN. *Free bits, PCPs and non-approximability - towards tight results.* Proceedings of the 36th Annual Symposium on Foundations of Computer Science. 1995. 422-431.

[BK97] A. BLUM. D. KARGER. *An $\tilde{O}(n^{3/14})$-coloring algorithm for 3-colorable graphs.* Information Processing Letters 61. 1997. 49–53.

[Blu89] A. BLUM. *An $O(n^{0.4})$ approximation algorithm for 3-coloring.* Proceedings of the 21st Annual Symposium on the Theory of Computing. 1989. 535–542.

[Blu94] A. BLUM. *New approximation algorithms for graph coloring.* Journal of the ACM. 41(3), 1994. 470–516.

[Bou85] J. BOURGAIN. *On Lipschitz embedding of finite metric spaces in Hilbert space.* Israel Journal of Mathematics 52, 1985, 46–52.

[BR90] B. BERGER, J. ROMPEL. *A better performance guarantee for approximate graph coloring,* Algorithmica 5(4), 1990, 459–466.

[BS94] M. BELLARE, M. SUDAN. *Improved non-approximability results*. Proceedings of the 26th Annual Symposium on the Theory of Computing, 1994. 184–193.

[FJ97] A. FRIEZE, M. JERRUM. *Improved approximation algorithms for MAX k-CUT and MAX BISECTION*, Algorithmica 18, 1997. 67–81.

[FK96] U. FEIGE, J. KILIAN. *Zero knowledge and the chromatic number*. Proceedings of the 11th Annual IEEE Conference on Computational Complexity, 1996, 278–287.

[Für95] M. FÜRER. *Improved hardness results for approximating the chromatic number*. Proceedings of the 36th Annual Symposium on Foundations of Computer Science, 1995. 414–421.

[GJ76] M. GAREY, D. JOHNSON. *The complexity of near-optimal graph coloring*. Journal of the ACM. 23(1), 1976, 43–49.

[GJ79] M. GAREY, D. JOHNSON. Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman. New York. 1979.

[GLS81] M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER. *The ellipsoid method and its consequences in combinatorial optimizations*. Combinatorica 1. 1981. 169–197.

[GLS88] M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER. Geometric algorithms and combinatorial optimization. Springer-Verlag, Berlin. 1988.

[GVL83] G. GOLUB, C. VAN LOAN. Matrix computations. The Johns Hopkins University Press, Baltimore, 1983.

[GW94] M. GOEMANS, D. WILLIAMSON. *.878-approximation algorithms for MAX CUT and MAX 2SAT.* Proceedings of the 26th Annual Symposium on the Theory of Computing, 1994, 422–431.

[GW95] M. GOEMANS AND D. WILLIAMSON. *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming.* Journal of the ACM, 42(6) 1995, 1115–1145.

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

[Joh74a] D. JOHNSON *Approximation algorithms for combinatorial problems.* Journal of Computer and System Sciences, 9. 1974, 256–278.

[Joh74b] D. JOHNSON. *Worst case behavior of graph coloring algorithms.* Proceedings of the 5th Southeastern Conference on Combinatorics. Graph Theory and Computing. Congressus Numerantium X. 1974. 513-527.

[Kar72] R. KARP. *Reducibility among combinatorial problems.* In R. E. Miller and J. W. Thatcher (editors). Complexity of Computer Computations. Plenum Press. New York, 1972. 85–103.

[KLS92] S. KHANNA. N. LINIAL. S. SAFRA. *On the hardness of approximating the chromatic number,* Proceedings of the 2nd Israeli Symposium on Theory and Computing Systems. 1. 1992. 250–260.

[KMS94] D. KARGER, R. MOTWANI. M. SUDAN. *Approximate graph coloring by semidefinite programming,* Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 1994, 2–13.

[KMS98] D. KARGER, R. MOTWANI, M. SUDAN. *Approximate graph coloring by semidefinite programming,* Journal of the ACM, 45(2). 1998. 246–265.

[Knu94] D. KNUTH. *The sandwich theorem*, The Electronic Journal of Combinatorics, 1, 1994, 1–48.

[LLR95] N. LINIAL, E. LONDON, Y. RABINOVICH. *The geometry of graphs and some of its algorithmic applications*, Combinatorica, 15(2), 1995, 215–245.

[Lov79] L. LOVÁSZ *On the Shannon capacity of a graph*, IEEE Transactions on Information Theory, IT-25(1), 1979, 1–7.

[LV89] N. LINIAL, U. VAZIRANI. *Graph products and chromatic numbers*, Proceedings of the 30th Annual Symposium on Foundations of Computer Science, 1989, 124–128.

[MNR97] R. MOTWANI, J. NAOR, P. RAGHAVAN *Randomized approximation in combinatorial optimization*, In D. Hochbaum (editor), Approximation algorithms for NP-hard problems, PWS Publishing Company, 1997, section 11.3.2, 462–475.

[PL96] P. KLEIN, H. LU. *Efficient approximation algorithms for semidefinite programs arising from maxcut and coloring*, Proceedings of the 28th Annual Symposium on the Theory of Computing, 1996, 338–347.

[Wig83] A. WIGDERSON. *Improving the performance guarantee for approximate graph coloring*, Journal of the ACM, 30(4) 1983, 729–735.

[WZ96] H. WOLKOWICZ, Q. ZHAO. *Semidefinite programming relaxations for the graph partitioning problem*, Technical report, Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, N2L 3G1 Canada, October 1996.
`http://orion.math.uwaterloo.ca/~hwolkowi/henry/reports/graphpart.ps.gz`