Fractionally Total Colouring Most Graphs

Conor John Meagher

Masters of Science

School of Computer Science

McGill University

Montreal,Quebec

2004-08-31

A thesis submitted to McGill University in partial fulfilment of the requirements of the degree of Masters of Science.

ABSTRACT

A total colouring is the assignment of a colour to each vertex and edge of a graph such that no adjacent vertices or incident edges receive the same colour and no edge receives the same colour as one of its endpoints. If we formulate the problem of finding the total chromatic number as an integer program, we can consider the fractional relaxation known as fractional total colouring. In this thesis we present an algorithm for computing the fractional total chromatic number of a graph, which runs in polynomial time on average. We also present an algorithm that asymptotically almost surely computes the fractional total chromatic number of $G_{n,p}$ for all values of p.

ABRÉGÉ

Une coloration totale d'un graphe est le coloration des arêtes et des sommets telle que deux sommets adjacents ont des couleurs différentes, deux arêtes incidentes ont des couleurs différentes, et une arête a une couleur différente de celles des ses extrémités. Si nous formulons le problème de trouver le nombre chromatique total comme un programme linéaire entier, nous pouvons considérer la relaxation connue comme la coloration totale fractionnaire. Dans cette thèse nous présentons un algorithme pour calculer le nombre chromatique total d'un graphe en temps polynomial en moyenne. Nous présentons aussi un algorithme qui calcule asymptotiquement presque sûrement le nombre chromatique total de $G_{n,p}$ pour toute valeur de p.

ACKNOWLEDGEMENTS

I sincerely appreciate the opportunity to have studied under Bruce Reed. Bruce not only provided me with the background needed to complete this thesis, he also proposed many of the approaches used to solve the problems within it. I truly am in awe of how, after having been stuck on a problem for an extended period of time, I could pose it to him over dinner and a couple minutes later he would come up with a scheme that would undoubtedly solve it. With the amount of time Bruce has put into helping me, the time he spends helping other students, and the time he spends doing his own research, I often wonder when he finds the time to sleep. I would also like to thank Bruce for not only teaching me an abundance of math, but also for his lessons on writing. Bruce, I will forever be in your debt.

I would also like to thank everyone in the Discrete Mathematics Group at McGill. It is a pleasure to learn, study, and work along such talented and pleasant individuals. My office mates over the past year, Bohdan Kaluzny and Chris Wu, have made coming into school a delight rather than a chore. I thank them for putting up with my many jokes at their expense.

Brad Hammell helped get me a job during the middle of my M.Sc. that tremendously helped my financial situation. I would like to thank everyone in Secure Systems at Pitney Bowes for making my time there enjoyable.

My family has always been supportive of my career in academia, and have helped me realize that enjoying your work is an important part of life. My friends that have had to deal with my moody disposition over the last few months, I thank you for your patience.

TABLE OF CONTENTS

CHAPTER 1
Introduction

1.1 Introduction

Solving a *combinatorial optimization* problem involves finding an optimal solution to the problem amongst the set of possible candidates. Often the number of possible solutions is exponential in the size of the problem or even infinite, and hence running through all the solutions to find an optimal one is prohibitively expensive. Finding an optimal solution efficiently is the main focus of study in this area.

Determining the minimum number of colours needed to colour a graph is the main area of study in graph colouring. This is a combinatorial optimization problem. Graph colouring has applications to many real world problems, eg. telecommunications, scheduling, bioinformatics, and the internet. We will mainly focus on algorithms for computing optimal colourings.

Graph colouring dates back to at least 1852, when Francis Guthrie came up with the four colour conjecture. 'Every map can be coloured with four colours so that neighbouring countries that shared a common border receive a different colours'. Since then graph colouring has been one of the most studied areas of graph theory.

Vertex and edge colouring are two of the most popular areas of study in graph theory. They have been shown to be computationally difficult and at the same time practically important. While total colouring doesn't enjoy the long history or theory of the others, it too seems to be inherently difficult and interesting.

From the several different ways to approach graph colouring we choose to focus on the integer programming formulations of graph colouring and their LP relaxations. We present more formal definitions in section 1.4.

1.2 Point of View

Viewing fractional colouring as a linear program has many advantages. In particular, this approach allows the use of the dual of the LP formulation to aid in developing efficient algorithms. We will see how the algorithmic equivalence between optimal weighted fractional colouring and strong separation algorithms for the polytope of the dual LP can be useful, both combinatorially and with the help of the ellipsoid method.

When considering fractional total colouring, we exploit the fact that we have a polynomial time algorithm for fractional edge colouring. To do so, we fix a vertex colouring (not necessarily optimal) and combine it with optimal fractional edge colourings of subgraphs in an intelligent manner. Variants of this approach provide algorithms that will compute the fractional total chromatic number of a graph with certain structural properties. We will also see techniques that construct an auxiliary graph whose fractional edge colourings correspond to the fractional total colourings of the original graph.

The algorithms we present for colouring random graphs rely heavily on properties of their degree distributions. We combine probabilistic results on structural properties of random graphs with deterministic results on fractional total colouring graphs with these properties. We obtain polynomial time algorithms that asymptotically almost surely give an optimal fractional total colouring of $G_{n,p}$ for all values of p. Using these techniques we will also give an algorithm that computes the fractional total chromatic number in polynomial time on average.

1.3 Colouring

Before we discuss the different types of graph colourings, we need some formal definitions. A *graph* $G = (V, E)$ consists of a set $V(G)$ of vertices and a set $E(G)$ of edges. The edge (v_1, v_2) joins vertices v_1 and v_2. We will only consider loopless graphs, ie. $\forall v \in V(G)$, $(v, v) \notin E(G)$. For the most part we will only be considering graphs with one edge between any pair of vertices, but on occasion we will have multiple edges between vertices, ie. we sometime consider multi-graphs. Unless otherwise instructed you must assume that every graph is a *simple graph*, no loops or multi-edges. The complement $\bar{G}$ of a graph $G = (V, E)$, is a graph on the vertices $V(G)$ with the edge set $E' = \{e = (u, v) : e \notin E(G), u, v \in V(G)\}$. We will consider three main types of graph colouring in this thesis.

Vertex colouring is the assignment of a colour to each vertex of a graph so that no vertices joined by an edge receive the same colour. Two vertices joined by an edge are called *adjacent*. The minimum number of colours needed to vertex colour a graph G is known as the *chromatic number*, $\mathcal{X}(G)$. Unfortunately, vertex colouring for general graphs has been shown to be NP-complete. In fact, it was proven in [12] that approximating $\mathcal{X}(G)$ to within $|V(G)|^{1-\epsilon}$ is hard, unless $P = NP$.

The second type of graph colouring that we are going to consider is *edge colouring*. A proper edge colouring is an assignment of colours to the edges of a graph so that no edges sharing an endpoint receive the same colour. Two edges that share an endpoint are called *incident*. The *neighbourhood*, $N(v)$, of a vertex v is the set of adjacent vertices to v. The degree of a vertex is the size of its neighbourhood, or equivalently the number of edges that have v as an endpoint. The maximum vertex degree in a graph G, is denoted $\Delta(G)$ or just Δ. The minimum number of colours needed to edge colour a graph G is known as the *chromatic index*, and denoted $\mathcal{X}^e(G)$. Clearly we need at least $\Delta(G)$ colours to edge colour G, since if v has degree $\Delta(G)$ the edges around v will need $\Delta(G)$ distinct colours. In fact, Vizing [40] proved that you can always edge colour a simple graph using at most $\Delta + 1$ colours. Graphs with chromatic index of Δ are known as *Class 1* graphs and those with chromatic index of $\Delta + 1$ are known as *Class 2* graphs. So approximating the chromatic index within 1 is trivial. But Holyer [17] showed that determining whether a graph's chromatic index is Δ or $\Delta + 1$ is NP-complete. Nevertheless, certain structural properties allow us to find a $\Delta(G)$ edge colouring of G in polynomial time. Fournier [13] proved the following lemma on edge colouring.

Lemma 1 *A simple graph G whose vertices of maximum degree induce a stable set is $\Delta(G)$ edge colourable. Furthermore, such a colouring can be found in $O(n^4)$ time.*

We focus mainly on a third type of graph colouring, *total colouring*. A total colouring is an assignment of colours to the edges and the vertices of a graph so that no two adjacent vertices or incident edges are assigned the same colour, and no edge has the same colour as one of its endpoints. The minimum number of colours needed to total colour a graph G is known as the graph's *total chromatic number*, $\mathcal{X}^T(G)$. A total colouring of a graph G will require at least $\Delta(G) + 1$ colours, $\Delta(G)$ colours for the edges out of a vertex of maximum degree and one more colour for the vertex itself. Behzad [3] and Vizing [41] independently conjectured that $\mathcal{X}^T(G) \leq \Delta(G)+2$. While this has been shown to be true for many classes of graphs (see [42]), it is still open in general. Molloy and Reed [35] proved that $\mathcal{X}^T(G) \leq \Delta(G) + C$ where C is a large constant. Unfortunately, it was shown in [33] that determining the total chromatic number is NP-complete, even for bipartite 3-regular graphs.

Since this thesis focuses on polyhedral algorithms for computing optimal colourings, some background on the area is needed before we can proceed. We present an overview of some of the concepts needed in the following two sections.

1.4 Polyhedral Optimization

Many combinatorial optimization problems can be formulated as optimization problems over a polyhedron. Formally, a subset $P \subseteq \mathbb{R}^n$ is called a *polyhedron* if it can be defined by $P = \{x|Ax \leq b\}$, where A is a $m \times n$ matrix and $b \in \mathbb{R}^m$ is a vector. Often the added constraint that $x \geq 0$ will also be present. If the polyhedron is bounded we refer to it as a *polytope.* A *linear program* or *LP* is the maximization or minimization of a linear function over a polyhedron. We will typically represent them in the following form:

$$max\{c^T x : Ax \leq b, x \geq 0\}$$

We refer to $c^T x$ as the objective function, it is the linear equation we are trying to maximize or minimize. The rest of the inequalities are the constraints of the linear program. The following pivotal theorem is known as the *duality theorem of linear programming.*

Theorem 2 *If A is a $m \times n$ matrix, b is a vector in $\mathbb{R}^m$ and c is a vector in $\mathbb{R}^n$, then*

$$\max\{c^T x : Ax \leq b, x \geq 0\} = \min\{y^T b : y^T A \leq c^T, y \geq 0\}$$

if the optimums are finite.

We refer to these LPs as *duals.*

If we add the constraint that the solution is an integral vector (ie. has integer entries) we call the linear program an *integer program*. It's common to refer to the process of removing the constraint that the solution take integral values as "relaxing" the integer program to a linear program, its fractional relaxation.

We will abbreviate integer program as IP. The solution to the relaxation of an IP can often give useful bounds on the optimal value of the IP. If we are really lucky, the relaxation's solution can be proved to be the same as the IP solution. The value of the IP may also be equal to the value of the LP rounded up (or rounded down in a maximization problem).

1.5 Ellipsoid Method and Complexity

In 1979 Khachiyan [20] used the *Ellipsoid Method*, an algorithmic procedure, to show that solving LPs is in P. In contrast solving integer programs is NP-complete. Sometimes LPs are given in a compact representation. Then it isn't always clear if the ellipsoid method can solve the LP in polynomial time in terms of its input size. Every IP is a compact representation of an LP. The solutions of an IP lie on the convex hull of integral vectors that are feasible solutions to the IPs fractional relaxation LP. Thus, the IP is an LP whose feasible region is this convex hull. This LPs constraints may be difficult to find, given the IP, and there may be exponentially (in terms of the input LPs size) many of them.

The ellipsoid method also provided the means to show algorithmic equivalence of complexity for many problems. The following is known as *strong separation*,

Given a polytope P and a vector x. If x is in P output 'YES'. Otherwise, give a constraint that x violates but all points in P satisfy.

Similarly, *strong optimization* is,

For a given polytope P and objective function c. Determine $x \in P$ such that $c^T x$ is maximized.

It was shown in [28] and [18] by the use of the ellipsoid method that,

Theorem 3 *Strong separation over a polytope P is polynomial time solvable if and only if the problem of strong optimization over P is polynomial time solvable.*

The following lemma gives a bound on the runtime for the ellipsoid method to do strong optimization when given an algorithm for strong separation. This runtime analysis is from [4].

Lemma 4 *Suppose we are given a strong separation oracle that takes time T over polytope $P \in \mathbb{R}^d$. Suppose further that polytope P is contained in a sphere of radius R, and if P is non-empty it contains a sphere of radius r. Then the ellipsoid method can optimize a linear function over P in time $O(d^2 \log(\frac{R}{r})T + d^4 \log(\frac{R}{r}))$.*

If we have an LP where the polytope defined by the dual is P', and an algorithm for strong separation exists over P'. Then Theorem 3 implies we can optimize over the dual and hence determine the optimal objective value of our original LP in polynomial time.

Theorem 3 also implies that if a violated constraint (separating hyperplane) can't be found efficiently over a polytope P, then the ellipsoid method won't be able to do strong optimization over P efficiently.

An LP given in a compact representation may have an exponential number of constraints in terms of its input size. However, this does not imply that the LP is solvable in polynomial time nor does it imply that it is NP-hard to solve. For example, the IP for maximum weight stable set can be given in a compact representation and it is NP-hard to solve in general. On the other hand, fractional edge colouring may also be given in a compact representation but even though its polytope is defined by an exponential number of constraints, it can be solved in polynomial time. This is in part due to a polynomial time strong separation algorithm for its dual's polytope (we will discus this further in chapter 3). Thus, it doesn't always matter how many constraints define the polytope of an LP, what's more important is how quickly a violated one (separation) can be found over the polytope or the polytope of the dual. For a more thorough understanding of the ellipsoid method and complexity results related to it see [30].

The relation between an optimization problem and separation over the polytope of the dual problem is pivotal in understanding the complexity of graph colouring. In the next section we describe colouring as an integer program and show how we can apply these results.

CHAPTER 2
Relaxing Colouring IPs

2.1 Vertex Colouring

No edges exist between a set of vertices coloured the same colour in a proper colouring, such a set of vertices is called a *stable set*. Thus, the chromatic number of a graph G is the minimum number of stable sets whose union is $V(G)$. If we let $\mathcal{S}(G)$ be the family of all stable sets of a graph G, we can formulate vertex colouring as the following IP.

$$\min\{\sum_{S\in\mathcal{S}(G)} w_S : \forall S \in \mathcal{S}(G), w_S \in \{0,1\}; \forall v \in V(G), \sum_{\{S\in\mathcal{S}(G):S\ni v\}} w_S \geq 1\}$$

So vertex colouring is just a special case of integer programming. As is often the case when an IP is known to be NP-hard, it's worthwhile looking at the fractional relaxation of the problem. If we relax the constraint that $w_S \in \{0,1\}$ we get the following LP that defines *fractional vertex colouring*, $\mathcal{X}_f(G)$.

$$\min\{\sum_{S\in\mathcal{S}(G)} w_S : \forall S \in \mathcal{S}(G), w_S \geq 0; \forall u \in V(G), \sum_{\{S\in\mathcal{S}(G):S\ni u\}} w_S \geq 1\}$$

Unfortunately, it was shown in [24] that an optimal fractional vertex colouring approximates the chromatic number of a graph G within $\log(|V(G)|)$. Since approximating $\mathcal{X}(G)$ to within $|V(G)|^{1-\epsilon}$ is NP-hard, it follows that it is NP-hard to compute the fractional chromatic number in general [26].

The dual of fractional vertex colouring is known as the *fractional clique number*, $\omega_f(G)$, and is given by the following LP:

$$\max\{\sum_{v\in V(G)} x_v : \forall v \in V(G), x_v \geq 0; \forall S \in \mathcal{S}(G), \sum_{v\in S} x_v \leq 1\} \qquad \text{LP (1)}$$

This LP is the relaxation of an IP that finds the maximum size clique $\omega(G)$ in a graph. By the duality theorem for linear program $w_f(G)$ is NP-hard to compute.

Getting back to the theme of optimization and separation, we now consider the "harder" problem of weighted fractional vertex colouring. In this generalization of fractional vertex colouring, each vertex v has a specific weight b_v such that the sum of weights on stable sets containing v is at least b_v (instead of 1). This gives the following LP for weighted fractional chromatic number.

$$\min\{\sum_{S\in\mathcal{S}(G)} w_S : \forall S \in \mathcal{S}(G), w_S \geq 0; \forall u \in V(G), \sum_{\{S\in\mathcal{S}(G):S\ni u\}} w_S \geq y_v\}$$

Computing the weighted fractional chromatic number for general graphs is clearly NP-hard. If we set $\forall\ v \in V(G)$, $b_v = 1$ this problem is fractional vertex colouring.

If we consider the dual of the weighted fractional chromatic number LP, we get

$$\max\{\sum_{v\in V(G)} x_v y_v : \forall v \in V(G), x_v \geq 0; \forall S \in \mathcal{S}(G), \sum_{\{v\in V(G):v\in S\}} x_v \leq 1\} \qquad \text{LP (2)}$$

The *stable set polytope*, $STAB(G) \subseteq \mathbb{R}^{|V(G)|}$, is the convex hull of the incidence vectors of all stable sets of G. Clearly, strong optimization over $STAB(G)$ is maximum weight stable set. In fact, an algorithm for maximum weight stable set is a separation algorithm over the polytope of LP (2). This implies that a polynomial time strong optimization algorithm over $STAB(G)$ (maximum weight stable set) gives a polynomial time algorithm for solving LP (2) via the ellipsoid method. Hence, it solves weighted fractional chromatic number in polynomial time, this problem is NP-hard for graphs in general. These results follow from Theorem 3 and duality.

The following lemma, a corollary of the duality theorem, also relates fractional vertex colouring to the stable set problem. It will be used in chapter 5. A graph *automorphism* is a one-to-one mapping of the vertices of the given graph G back to the vertices of G. A graph is *vertex transitive* if for every choice of $v_1 \in V(G)$ and $v_2 \in V(G)$ there exists a valid automorphism mapping v_1 to v_2.

Lemma 5 *For any simple graph G, $\mathcal{X}_f(G) \geq \min\{\frac{|V(G)|}{|S|} : S \in \mathcal{S}(G)\}$ which holds with equality if G is vertex transitive.*

Proof. By the duality theorem of linear programming, we know that $\mathcal{X}_f(G) = \omega_f(G)$. Assume $S \subseteq V(G)$ is the largest stable set in G. Consider the LP (1) defining $\omega_f(G)$. Define $x_v \in \mathcal{R}^{|V(G)|}$ as the all $\frac{1}{|S|}$ vector (ie. $\forall\ v \in V(G), x_v = \frac{1}{|S|}$). Looking at the constraints of LP (1), we see that x is a feasible solution, since no stable set can have a total weight of more than 1. The objective value of x is $\frac{|V(G)|}{|S|}$, this proves $\mathcal{X}_f(G) \geq \min\{\frac{|V(G)|}{|S|} : S \in \mathcal{S}(G)\}$.

If G is vertex transitive, let A be the set of all automorphisms of G. Let $x \in \mathbb{R}^{|V(G)|}$ be an optimal solution to LP (1). Since every convex combination of optimal solutions of LP (1) is an optimal solution, it follows that $y = \frac{1}{|A|}\sum_{\pi \in A} \pi(x)$ is an optimal solution to LP (1). Since G is vertex transitive, it follows that each entry of y must have the same value (ie. $\forall v, u \in V(G), y_v = y_u$). If $y_v > \frac{1}{|S|}$ stable set S will have too much weight on it, it follows that $\mathcal{X}_f(G) = \min\{\frac{|V(G)|}{|S|} : S \in \mathcal{S}(G)\}$ when G is vertex transitive. ■

In the next section we present some classes of graphs for which maximum weight stable set can be solved in polynomial time. For these classes of graphs we therefore have polynomial time algorithms for separation on the polytope of the dual of weighted fractional colouring. Thus, we can solve weighted fractional chromatic number on these graphs in polynomial time.

2.1.1 Fractional Chromatic Number in Polynomial Time

While fractional vertex colouring in general is NP-hard, there do exists classes of graphs for which polynomial algorithms exists to compute the fractional chromatic number.

For any graph G,

$$\omega(G) \leq \omega_f(G) = \mathcal{X}_f(G) \leq \mathcal{X}(G)$$

where the equality is by LP duality. A graph is called *perfect* if $\omega(G') = \mathcal{X}(G')$ for all $G' \subseteq G$, and all four of these numbers are equal. If $\mathcal{C}(G)$ is the set of all cliques of a graph G, the polytope defined by

$$\{\forall v \in V(G), x_v \geq 0; \forall C \in \mathcal{C}(G), \sum_{\{v \in V(G): v \in C\}} x_v \leq 1\}$$

is a well studied polytope called the *fractional stable set polytope* or $QSTAB(G)$. A graph being perfect is equivalent to requiring that the max of $\sum_{v \in V(G)} c_v x_v$ is integral for all objective functions c and $x \in QSTAB(G)$ where $\forall v \in V(G)$, $c_v \in \{0,1\}$. Chvátal [9] proved, using results of Lovász [23], that in fact the max of $\sum_{v \in V(G)} c_v x_v$ is the same over $QSTAB(G)$ and $STAB(G)$ for any objective function c. Equivalently, he proved that $QSTAB(G) = STAB(G)$. While it's hard to optimize over these polytopes in general, a convex body known as the *theta* *body* $TH(G)$ has been shown to always lie between the two polytopes, ie. $STAB \subseteq TH(G) \subseteq QSTAB(G)$. This body has the nice property that we can optimize any linear function over it in polynomial time [25]. Hence, we can solve IPs and LPs in polynomial time over $STAB(G)$ and $QSTAB(G)$ for perfect graphs. Lovász [23] proved that a graph is perfect if and only if its complement is perfect. It was then shown in [29] that the weighted versions of optimal stable set, clique, chromatic number, and clique cover have polynomial algorithms if the graph is perfect. These algorithms motivated the use the ellipsoid method as a polynomial time separation algorithm. Although interesting from a mathematical programming viewpoint, the study of perfect graphs was actually motivated by a question of Shannon's on information theory. In 1960 Claude Berge conjectured that a graph is perfect if and only if the graph and its complement do not contain an odd induced hole of length at least 5. In 2002, Chudnovsky, Robertson, Seymour, and Thomas [27] proved Berge's conjecture to be true, and it's now known as the Strong Perfect Graph Theorem.

The *line graph* $L(G)$ of a graph G has vertex set $E(G)$ in which two vertices are adjacent if the edges they correspond to in G are incident. Given a graph G, we can check whether or not it is a line graph in polynomial time [22], ie. check if there is a graph G' such that $L(G') = G$. Since maximum weight stable set for a line graph $L(G)$ is equivalent to maximum weight matching for the graph G, we have a separation algorithm for the dual of fractional chromatic number. We will discuss maximum weight matching and a polynomial time combinatorial algorithm for fractional chromatic number further in chapter 3.

A graph is called *claw-free* if it does not contain an induced $K_{1,3}$ (the complete bipartite graph with one vertex in one partition and three in the other). Minty [34] gave a polynomial time algorithm to compute the maximum weight stable set for a claw-free graph. However, in 2001 Nakamura and Tamura [36] showed this algorithm fails on certain cases and gave a revised algorithm for computing the maximum weight stable set in polynomial time. With this algorithm we can do strong separation over the polytope of the dual of fractional vertex colouring. Therefore, we can compute the fractional chromatic number of a claw-free graph in polynomial time, this follows by duality. This is a generalization of the results on line graphs.

2.2 Edge Colouring

Consider a set $M \subset E(G)$ of edges with a given colour in a proper edge colouring of G. Note that no two edges in M are incident, we call a set of edges with this property a *matching*. If we let $\mathcal{M}$ be the family of all matchings of a graph G, we can formulate the chromatic index of a graph as the following IP:

$$\min\{\sum_{M\in\mathcal{M}(G)} w_M : \forall M \in \mathcal{M}(G), w_M \in \{0,1\}; \forall e \in E(G), \sum_{\{M\in\mathcal{M}(G):M\ni e\}} w_M \geq 1\}$$

If we consider the fractional relaxation of the IP formulation of the chromatic index we get the *fractional chromatic index*. The following linear program defines the fractional chromatic index,

$$\min\{\sum_{M\in\mathcal{M}(G)} w_M : \forall M \in \mathcal{M}(G), w_M \geq 0; \forall e \in E(G), \sum_{\{M\in\mathcal{M}(G):M\ni e\}} w_M \geq 1\}$$

We denote the fractional chromatic index as $\mathcal{X}_f^e$. We can define weighted fractional edge colouring in a similar fashion as we defined weighted fractional vertex colouring. The following LP is the dual of weighted fractional edge colouring.

$$\max\{\sum_{e\in E(G)} x_e y_e : \forall e \in E(G), x_e \geq 0; \forall M \in \mathcal{M}(G), \sum_{e\in E(G):e\in M} x_e \leq 1\} \qquad \text{LP (3)}$$

The *incidence vector* I_M of a matching M is the vector x in $\mathbb{R}^{|E(G)|}$ such that $x_e = 1$ if $e \in M$ and $x_e = 0$ otherwise. Here and elsewhere we index $\mathbb{R}^{|E(G)|}$ by the elements of $E(G)$. The *matching polytope* $\mathcal{M}(G)$ of a graph G is the convex hull of incidence vectors of all matchings of G. Strong optimization over the matching polytope is known as maximum weight matching.

An algorithm for maximum weight matching is actually a strong separation algorithm over the polytope of the dual of weighted fractional edge colouring (LP (3)). This implies that a polynomial time algorithm for maximum weight matching gives a polynomial time algorithm for computing the weighted fractional chromatic index via the ellipsoid method and duality. We will discus a combinatorial algorithm in chapter 3. The special case of maximum weight matching where all the weights are 1 is maximum matching. If the maximum matching covers all the vertices of the graph we call the matching, *perfect*. So a maximum weight matching algorithm not only solves fractional edge colouring, it also allows us to check if a graph has a perfect matching.

In some graphs finding a perfect matching is easier.

Definition 1 *We use $\Delta_{min}(G)$ to denote the minimum vertex degree in G.*

Lemma 6 *Every graph G with $n = |V(G)|$ even and $\Delta_{min}(G) \geq \frac{n}{2}$ has a perfect matching. Furthermore one can be found in $O(n^3)$ time.*

Proof. Let M be a maximum matching of G and suppose there exists u and v unmatched by M. Then $N(v)$ and $N(u)$ are contained in $V(M)$ by the maximality of M. Furthermore, for each edge e of M, $|N(v) \cap e| + |N(u) \cap e| \leq$

2 by the maximality of M. So $|N(v)| + |N(u)| \leq 2|M| < n$, a contradiction. Hence, since $|V(G)|$ is even M is perfect. ■

2.3 Total Colouring

The *total graph* $T(G)$ of a graph G has vertex set $V(G) \cup E(G)$ and two elements are adjacent if they are either adjacent vertices of G, incident edges of G or one element is an edge of G and the other is one of its endpoints. One can easily see that the fractional total chromatic number is equal to the fractional chromatic number of $T(G)$, ie. $\mathcal{X}_f^T(G) = \mathcal{X}_f(T(G))$. This relation between total colouring and vertex colouring makes natural an analogous definition of the total colouring IP from the vertex colouring IP.

Consider a set $T \subset E(G) \cup V(G)$ of edges and vertices assigned the same colour in a proper total colouring of a graph G. No two vertices of T have an edge between them, no two edges of T share an endpoint, and no edge has an endpoint in T (ie. a matching plus a disjoint stable set). We call a set of edges and vertices with this property a *total stable set.* Thus, the total chromatic number of a graph is the minimum number of total stable sets whose union is $V(G) \cup E(G)$. If we let $\mathcal{T}(G)$ be the family of all total stable sets of a graph G, we can formulate the total chromatic number as an optimal solution to the following IP.

$$\min\{\sum_{T\in\mathcal{T}(G)} w_T : \forall T \in \mathcal{T}(G), w_T \in \{0,1\}; \forall u \in E(G) \cup V(G), \sum_{\{T\in\mathcal{T}(G):T\ni u\}} w_T \geq 1\}$$

The fractional relaxation of this IP, obtained by modifying the constraint $w_T \in \{0,1\}$ to be $w_T \geq 0$ is the following LP:

$$\min\{\sum_{T\in\mathcal{T}(G)} w_T : \forall T \in \mathcal{T}(G), w_T \geq 0; \forall u \in E(G) \cup V(G), \sum_{\{T\in\mathcal{T}(G):T\ni u\}} w_T \geq 1\}$$

We refer to the optimal objective value of this LP as the graphs *fractional total chromatic number*, $\mathcal{X}_f^T$. It's easy to see that $\mathcal{X}_f^T(G) \geq$

$\Delta(G)+1$. It was shown in [21] by Kilakos and Reed that $\mathcal{X}_f^T(G) \leq \Delta(G)+2$, and this upper bound is the best possible. It is currently unknown if determining the fractional total chromatic number is NP-hard. By assigning a weight of 0 to the edges of the graph and a weight of 1 to the vertices we reduce fractional vertex colouring to fractional weight total colouring. Since the former problem is NP-hard, so is the latter.

An *edge dominating set* is a subset of edges $M \subset E(G)$ such that the removal of M and the vertices that are endpoints of edges in M leaves a stable set. Yannakakis and Gavril [14] show that computing a minimum edge dominating set is NP-hard. They also showed G has a total stable set of size $n - k$ if and only if it has an edge dominating set of size k. This implies that maximum total stable set is NP-hard.

Fractional total colouring is similar to fractional vertex colouring in the sense that the weighted versions are NP-hard. But in other ways fractional total colouring can be viewed as being "easier" than fractional vertex colouring since approximating the value within a constant value of 1 is trivial (by Kilakos and Reed's [21] result). In this way it is similar to fractional edge colouring.

The complexity equivalence between maximum weight total stable set and fractional total colouring implies that we can get a separation algorithm for the polytope of the dual of fractional total colouring from an algorithm for maximum weight total stable set. In section 4.2 we will see that this approach yields an $O(2^n n^9)$ algorithm for computing the fractional total chromatic number.

We will also be exploiting fractional total colouring's kinship with fractional edge colouring to get algorithms which work quickly on almost every graph. They also work in polynomial average time. We discuss the theory of fractional edge colouring in the next chapter. We show how we can exploit it to obtain algorithms and prove theorems on fractional total colouring in chapter 4. We apply these results to the study of random graphs in chapter 5.

CHAPTER 3
Matching and Fractional Edge Colouring

Edmond's characterization of the matching polytope provided the means for a polynomial time algorithm to find a maximum weight matching of a graph. This implies that strong separation is polynomial time solvable over the polytope of the dual of weighted fractional edge colouring. Hence, we have a polynomial time algorithm for solving weighted fractional chromatic index. We discuss this characterization in the following section. In section 3.2 we describe how these results can be used to compute the fractional chromatic index of a graph combinatorially. In section 3.3 we present some results on fractional matching.

3.1 The Matching Polytope

To simplify notation, for a subset of vertices H, we denote the set of edges leaving H going to vertices in $V(G) \setminus H$ as $\delta(H)$ and $E(H)$ denotes the induced edges of H. Edmond's [10] gave the following characterization for a vector $x \in \mathbb{R}^{|E(G)|}$ to be in the matching polytope.

$$(i) \quad \forall \quad v \in V(G), \sum_{e \ni v} x_e \leq 1.$$

$$(ii) \quad \forall \quad H \subseteq V(G),\ where\ |H| \geq 3\ and\ odd, \sum_{e \in E(H)} x_e \leq \frac{1}{2}(|H| - 1).$$

$$(iii) \quad \forall \quad e \in E(G),\ x_e \geq 0.$$

If we let $\mathcal{P}(G)$ be the polytope defined by inequalities (i),(ii), and (iii) we get the following theorem.

Theorem 7 *Inequalities (i),(ii), and (iii) define the matching polytope, ie.* $\mathcal{P}(G) = \mathcal{M}(G)$*.*

To prove that this theorem is correct we will first give a characterization of the perfect matching polytope, and then use it to prove Theorem 7. The *perfect matching polytope* $\mathcal{M}_p(G)$ is the convex hull of incidence vectors of all perfect matchings of G. To characterize it, we need the following variant of (i):

(i') $\forall\ v \in V(G)$, $\sum_{e \ni v} x_e = 1$.

We let $\mathcal{P}'(G)$ be the polytope characterized by (i'),(ii), and (iii).

Theorem 8 *Inequalities* (i')*,(ii), and (iii) define the perfect matching polytope, ie.* $\mathcal{M}_p(G) = \mathcal{P}'(G)$*.*

Proof. Let $\mathcal{M}_p(G)$ be the perfect matching polytope, and let $\mathcal{P}'(G)$ be the polytope defined by (i'), (ii), and (iii). We claim that $\mathcal{P}'(G) = \mathcal{M}_p(G)$. Because $\mathcal{P}'(G)$ is convex, to show that $\mathcal{M}_p(G) \subseteq \mathcal{P}'(G)$, we need only show that the vertices of $\mathcal{M}_p(G)$ are in $\mathcal{P}'(G)$. That is, that inequalities (i'), (ii), and (iii) hold for every incidence vector of a matching. Assume that $x \in \mathbb{R}^{|E(G)|}$ is the incidence vector of a matching M, clearly $x \geq 0$ so (iii) is satisfied. Also (i') is satisfied since there is exactly one edge of M incident to each vertex. Now for any odd subset H of vertices, $\sum_{e \in E(H)} x_e$ is the number of edges of M contained within H which is at most $1/2(|H| - 1)$ since these edges are disjoint. So, (ii) holds.

To prove $\mathcal{P}'(G) \subseteq \mathcal{M}_p(G)$ assume for a contradiction that $\mathcal{P}'(G) \nsubseteq \mathcal{M}_p(G)$ and G is a minimal counter example in terms of $|E(G)| + |V(G)|$. Consider a vertex z of $\mathcal{P}'(G)$ which is not in $\mathcal{M}_p(G)$. Then $\forall\ e \in E(G)$, $z_e > 0$, since if $z_e = 0$ we could remove edge e and $G - e$ is a smaller counter example.

Suppose $z_e = 1$ for some edge $e = (u, v)$, then $z_{e'} = 0, \forall\, e' \neq e$ incident to u or v. So (i'), (ii), and (iii) hold for the restriction y of z to the edges of $G - u - v$. Thus, y can be expressed as a convex combination of perfect matchings of $G - u - v$ (by the minimality of G). That is, $y = \sum_{i=1}^{j} \alpha_i I_{M'_i}$, where $\forall i$, M'_i is a perfect matching of $G - u - v$ and the α_i's are nonnegative reals summing to one. Now setting $M_i = M'_i \cup (u, v)$ we have $z = \sum_{i=1}^{j} \alpha_i I_{M_i}$. This contradicts the fact that $z \notin \mathcal{M}_p(G)$. So, $\forall\, e \in E(G), z_e < 1$.

Every vertex of G must have degree at least 2, since (i') holds but $\forall e \in E(G), z_e \neq 1$. If every vertex of G had degree 2 then $\mathcal{P}'(G) = \mathcal{M}_p(G)$ trivially, so it follows that $\Delta(G) \geq 3$ which, along with the fact that every vertex has degree at least two, implies $|E(G)| > |V(G)|$. Now since z is a vertex of $\mathcal{P}'(G)$ it follows that at least $|E(G)|$ constraints of (i'),(ii), and (iii) hold with equality. This implies that there exists at least one odd subset of vertices H with $|H| \geq 3$ and $\sum_{e \in E(H)} z_e = \frac{1}{2}(|H| - 1)$ (since none of the constraints in (iii) are tight).

Now let G_1 be obtained from the subgraph of G induced by H by adding a new vertex v_2 with edges $\{(v, v_2) : (u, v) \in \delta(H), u \in V(G) \setminus H\}$. Similarly define G_2 to be the vertex induced graph of $V(G) \setminus H$, plus a new vertex v_1 and edges $\{(v, v_1) : (v, u) \in \delta(H), u \in H\}$. This may make G_1 and G_2 multi-graphs, in fact this characterization is valid for multi-graphs as well. By the minimality of G we know that the perfect matching polytopes for G_1 and G_2 are characterized by Edmond's constraints. Let z^1 and z^2 be the vectors indexed by the edges of G_1 and G_2 respectively, with $z^1_{e^1} = z_e$, where $\forall e^1 \in E(G_1)$, e is the corresponding edge in G. Similarly let $z^2_{e^2} = z_e$, where $\forall e^2 \in E(G_2)$, e is the corresponding edge in G. We can express z^1 and z^2 as convex combinations of perfect matchings of G_1 and G_2 (by the minimality of G). Let $M_{1,1}, ..., M_{1,j}$ be the matchings of G_1

such that $z^1 = \sum_{i=1}^{j} \alpha_{M_{1,i}} I_{M_{1,i}}$, similarly let $z^2 = \sum_{i=1}^{k} \alpha_{M_{2,i}} I_{M_{2,i}}$. Where $\sum_{i=1}^{j} \alpha_{M_{1,i}} = 1$ and $\sum_{i=1}^{k} \alpha_{M_{2,i}} = 1$.

For each edge $e \in \delta(H)$, the sum of $\sum_{M_{1,i} \ni e^1} \alpha_{M_{1,i}} = z_e$. Similarly $\sum_{M_{2,i} \ni e^2} \alpha_{M_{2,i}} = z_e$. Let $\mathcal{N}_{1,e}$ be the set of matchings of G_1 that contain edge e^1 having an associated non-zero α. Similarly, let $\mathcal{N}_{2,e}$ be the set of matchings containing e^2 of G_2 with a non-zero associated α. Define, $\forall N_1 \in \mathcal{N}_{1,e}$, $\forall N_2 \in \mathcal{N}_{2,e}$ a perfect matching $N = (N_1 - e^1) \cup (N_2 - e^2) \cup \{e\}$ of G and assign it a weight of $\alpha_N = \frac{\alpha_{N_1} \alpha_{N_2}}{z_e}$. It's an easy matter to verify that we have expressed z as a convex combination of these perfect matchings of G. This is a contradiction to $z \notin \mathcal{M}_p(G)$, so $\mathcal{P}'(G) = \mathcal{M}_p(G)$. ■

We will now show how we can use the characterization of the perfect matching polytope to characterize the matching polytope.

Proof of Theorem 7. To show that $\mathcal{M}(G) \subseteq \mathcal{P}(G)$ we mimic the proof that $\mathcal{M}_p(G) \subseteq \mathcal{P}'(G)$ above. To prove that $\mathcal{P}(G) \subseteq \mathcal{M}(G)$, we will construct an auxiliary graph G^*. Take a copy of G, call it G' with $V(G') = \{v' | v \in V(G)\}$ and $E(G') = \{e' = (u', v') | e = (u, v) \in E(G)\}$. Define $V(G^*) = V(G) \cup V(G')$, and $E(G^*) = E(G) \cup E(G') \cup \{(v, v') : \forall v \in V(G)\}$.

Suppose there is a vector $z \in \mathcal{P}(G) - \mathcal{M}(G)$. Let $y \in \mathbb{R}^{|E(G^*)|}$ be the vector indexed by the edges of G^* defined as follows:

- $\forall e \in E(G), y_e = z_e$ and $y_{e'} = z_e$.
- $\forall (v, v') \in E(G^*), y_{(v,v')} = 1 - \sum_{e \in E(G), e \ni v} z_e$.

We claim that $z \in \mathcal{P}(G) - \mathcal{M}(G)$ if and only if $y \in \mathcal{P}'(G^*) - \mathcal{M}_p(G^*)$.

If z satisfies (i) and (iii), then clearly y satisfies (i') and (iii). Similarly if y satisfies (i') and (iii) then z satisfies (i) and (iii). If z doesn't satisfy (ii) for some subgraph $H \subset G$, then y doesn't satisfy (ii) either. Since by the definition of y,

$$\sum_{e\in E(H)} y_e \;=\; \sum_{e\in E(H)} z_e. \tag{3.1}$$

Using equation (3.1), we find that if (ii) isn't satisfied by y for some odd subgraph completely contained in $V(G)$ or $V(G')$ then (ii) isn't satisfied by z.

It only remains to consider the case where (ii) isn't satisfied by y when the odd subgraph H has vertices from G and G'. Let $H_1 = \{v|v \in V(H), v' \notin V(H)\}$ and $H_2 = \{v'|v \notin V(H), v' \in V(H)\}$. Now let $S = V(H) - H_1 - H_2$. S has an even number of vertices, since for every vertex $v \in S$, S also contains v' by the definition of H_2. Similarly, for every $v' \in S$, $v \in S$ by the definition of H_1. This means that either H_1 or H_2 is odd, without loss of generality assume it's H_1. Any edge going from a vertex $v \in H_1$ to $V(H)$ must go to a vertex $u \in S$, since there are no edges between H_1 and H_2 in H. The corresponding edge (u', v') must be in $\delta(V(H)) \setminus \delta(H_1)$ and $y_{(u',v')} = y_{(u,v)}$. It now follows that,

$$\sum_{e\in\delta(H_1)} y_e \;\le\; \sum_{e\in\delta(H)} y_e.$$

Now, since $\sum_{e\in\delta(G)} y_e + 2\sum e \in E(G) y_e = |V(H)|$ and y violates (ii) for H, it follows that,

$$\sum_{e\in\delta(H_1)} y_e \le \sum_{e\in\delta(G)} y_e < 1.$$

Since $\sum_{e\in\delta(H_1)} y_e < 1$, we have $\sum_{e\in E(H_1)} y_e > \frac{1}{2}(|V(H_1)| - 1)$. Thus, (ii) is violated by a set of vertices completely contained in $V(G)$ so by equation (3.1) z violates (ii). ■

The following lemma is a consequence of this characterization of the matching polytope,

Lemma 9 *For a simple graph G, the fractional chromatic index of G is given by the following formula,*

$$\mathcal{X}_f^e(G) \;=\; \max\{\Delta(G), \max\{\frac{2|E(H)|}{|V(H)|-1} : H \subseteq G, |V(H)| \text{ is odd}, |V(H)| \geq 3\}\}$$

Proof. Clearly $(1/\beta, 1/\beta, ..., 1/\beta)$ is in the matching polytope precisely if $(1, 1, ..., 1)$ is a convex combination of β matchings. So, the fractional chromatic index is the minimum β s.t. $(1/\beta, 1/\beta, ..., 1/\beta)$ is in $\mathcal{M}(G)$. The constraints of (i) are satisfied for $(1/\beta, 1/\beta, ..., 1/\beta)$ if and only if $\beta \geq \Delta$. The constraints of (ii), for a specific H, are satisfied for $(1/\beta, 1/\beta, ..., 1/\beta)$ if and only if $\beta \geq \frac{2|E(H)|}{|V(H)|-1}$. ■

This lemma implies that a graph G with a subgraph H such that $|V(H)|$ is odd, $|V(H)| \geq 3$, and $\frac{2|E(H)|}{|V(H)|-1} > \Delta(G)$ has $\mathcal{X}_f^e > \Delta$. Hence $\mathcal{X}^e(G) = \Delta + 1$ for such graphs. We refer to H as an *overfull* subgraph if $\frac{2|E(H)|}{|V(H)|-1} > \Delta(G)$.

Since we can compute $\mathcal{X}_f^e(G)$ in polynomial time, the following conjecture of Chetwynd and Hilton[8] implies that we can compute $\mathcal{X}^e$ for any graph with $\Delta(G) > \frac{|V(G)|}{3}$ in polynomial time.

Conjecture 10 (The Overfull Conjecture) *A graph G with n vertices and $\Delta(G) > \frac{|V(G)|}{3}$ is Class 1 if and only if G has no odd overfull subgraph.*

There is a similar conjecture by Chetwynd, Hilton, and Hind [15] on total colouring. The *deficiency* of a vertex v, $def(v)$ or $def_G(v)$, is the difference between the maximum degree of G and the degree of v, so $def(v) = \Delta - deg(v)$. The *deficiency of a graph*, $def(G)$, is the sum of the deficiencies of each vertex, $def(G) = \sum_{v \in V(G)}(\Delta - deg(v))$. A graph G is *conformable* if it has a vertex colouring with $\Delta(G) + 1$ colours such that number of classes of parity different from that of $|V(G)|$ is at most the deficiency of G.

Conjecture 11 (The Conformability Conjecture) *For a graph G with $\Delta(G) \geq \frac{1}{2}(|V(G)|+1)$. Then $\mathcal{X}^T(G) \geq \Delta + 2$ if and only if G contains a subgraph H with $\Delta(G) = \Delta(H)$ which either isn't conformable or when $\Delta(G)$ is even, consists of $K_{\Delta(G)+1}$ with one edge subdivided.*

3.2 A Fractional Edge Colouring Algorithm

Lemma 9 implies that to determine $\mathcal{X}_f^e(G)$, we just need to check whether or not G contains an odd overfull subgraph, and find the "most" overfull one.

If G is Δ-regular (every vertex has degree Δ) and $H \subseteq V(G)$ with $|H|$ odd, then less than Δ edges leave H precisely if $|E(H)| > \Delta\frac{(|V(H)|-1)}{2}$. So, for regular graphs, finding an overfull odd subgraph is equivalent to finding an *odd cut* (V_1, V_2) (a partition of the vertices such that at least one partition contains an odd number of vertices) such that $|\delta(V_1)| < \Delta$.

In the case of general graphs we will construct an auxiliary graph G' from G with the property that the vertices of G' that correspond to vertices in G all have degree Δ. We do this by adding a new vertex v^* to G and adding edges between v^* and $V(G)$ until all the vertices in $V(G)$ have degree Δ. Note, this may make G' a multi-graph. Label all the vertices of $V(G')$ 'odd' if $|V(G')|$ is even. If $|V(G')|$ is odd we label v^* 'even' and the remaining vertices 'odd'.

We now define an odd cut in this auxiliary graph to be a partition of $V(G')$ into (V_1, V_2) such that both V_1 and V_2 contain an odd number of vertices labelled 'odd'. A minimum odd cut in G' has either V_1 or V_2 having an odd number of vertices of $V(G)$. It is easy to see that finding the "most" overfull odd subgraph is equivalent to finding the minimum odd cut in G'.

In [37] Padberg and Rao present an algorithm for finding minimum odd cuts, and hence overfull odd subgraphs of regular graphs in $O(n^4)$ time. We fix some 'odd' vertex s and for each other 'odd' vertex t, run a max-flow min-cut algorithm to find the minimum s-t cut of G.

Let (V_1, V_2) be the minimum cut found. If (V_1, V_2) is an odd cut we are done. If not consider two new auxiliary graphs G_1 and G_2, defined in the same manner as in the proof of Theorem 8. So, G_1 is the induced graph V_1 plus a new vertex v_2 with edges between V_1 and v_2 so that the degrees of the vertices V_1 in G_1 are preserved. Similarly we define G_2 to be the graph induced by the vertices of V_2 and a new vertex v_1 and edges between v_1 and V_2 in G_2 to preserve the degrees of the vertices. Label v_1 and v_2 as 'even'. We claim that a minimum odd cut (V_1^*, V_2^*) of G' exists such that either $V_1^* \subset V_1$ or $V_1^* \subset V_2$, so we only have to consider the problem of finding the minimum odd cuts in G_1 and G_2. We can continue recursively splitting our graphs into smaller and smaller graphs. We don't recurse on a problem if the minimum cut found is an odd cut or the graph has 2 vertices.

It only remains to prove that there is a minimum odd cut lying in one of the two auxiliary graphs. Let (S, T) is a minimum odd cut in G'. We can assume that both S and T intersect both V_1 and V_2 otherwise the odd cut would lie in one of our subgraphs. Also note that each of V_1, V_2, T, and S have at least one vertex labelled 'odd'. Consider $V_1 \cap T$ and $V_2 \cap S$, the edges of cuts $(V_1 \cap T, V_2 \cup S)$ and $(V_2 \cap S, V_1 \cup T)$ clearly lie in the union of our two original cuts. Also any edge that lies in both our new cuts must lie in both our original cuts, so

$$|\delta(V_1 \cap T)| + |\delta(V_2 \cap S)| \leq |\delta(V_1)| + |\delta(S)|. \tag{3.2}$$

Also one of our new cuts must be odd, since if $V_1 \cap T$ had an even number of 'odd' labelled vertices then $V_2 \cap T$ would have to have an odd number of 'odd' labelled vertices, which means that $V_2 \cap S$ would also need an odd number of ‘odd’ labelled vertices. Suppose by symmetry that $(V_1 \cap T, V_2 \cup S)$ is an odd cut, then if $|\delta(V_1 \cap T)| > |\delta(S)|$ by (3.2) this would mean that $|\delta(V_2 \cap S)| < |\delta(V_1)|$, but this is a contradiction since (V_1, V_2) was a minimum cut in all of G'. So $|\delta(V_1 \cap T)| \leq |\delta(S)|$, and we now have a minimum odd cut completely contained in one of our two subproblems.

We can find a max-flow min-cut in $O(n^3)$ [19] and the maximum level of the recursive tree of splitting up G' is $n-1$ so we get a total run-time of $O(n^4)$.

The above procedure can be modified to provide an algorithm for computing an actual optimal fractional edge colouring combinatorially in polynomial time (as opposed to just the optimal objective value). Due to the complexity of the changes, we’ve omitted this algorithm.

The ellipsoid method can also be used to compute an optimal fractional edge colouring. However, due to the complexity in presenting all the details involved we don’t include it here either. Lemma 4 can’t be applied to the variant of the ellipsoid method needed, since the polytope we are optimizing over is in $\mathbb{R}^{|\mathcal{M}(G)|}$ where $\mathcal{M}(G)$ is the set of all matchings of G and usually exponentially large. Instead we will refer to the runtime of a polynomial time implementation of an algorithm giving an optimal fractional edge colouring as $O(FEC)$. See [30] section (6.6.5) for the details of a polynomial time algorithm based on the ellipsoid method.

3.3 Fractional Matching

While we are on the subject of matchings we present some results on fractional matchings of graphs that we will use in section 4.5. These results will help us characterize the fractional total chromatic number of graphs with high degree.

The *maximum fractional matching* of a graph G is defined to be an optimal solution to the following linear program:

$$max\{\sum_{e \in E(G)} w_e : \forall e \in E(G), w_e \geq 0; \forall v \in V(G), \sum_{e \ni v} w_e \leq 1\}$$

Balinski [2] proved that there always exists an optimal fractional matching where the weights on the edges are either 0, $\frac{1}{2}$, or 1 (this is known as *half integral*). He also showed that a maximum fractional matching can be partitioned into a matching M with weight 1 on the edges and a set of odd cycles, vertex disjoint and disjoint from M, with weight $w_e = 1/2$ for each edge e in any of the odd cycles.

An optimal fractional matching can be computed in polynomial time by the ellipsoid method, since the fractional matching polyhedron is defined by a polynomial number of constraints and we can check to see if any are violated in polynomial time. But as is often the case when a problem has a polynomial time algorithm using the ellipsoid method, faster combinatorial algorithms can be found. Bourjolly and Pulleyblank [7] present an algorithm for computing an optimal fractional matching where the edges of weight 1 form a matching M and the edges of weight 1/2 form disjoint odd cycles, disjoint from M.

The basic idea is to construct an auxiliary bipartite graph $B(G)$. Label the vertices of G as $v_1, ..., v_n$, and define the vertices of $B(G)$ to be $\{y_1, ..., y_n, z_1, ..., z_n\}$. Make (y_i, z_j) an edge of $B(G)$ precisely if (v_i, v_j) is an edge of G. Take a maximum matching M of $B(G)$. Construct a fractional matching vector $x \in \mathbb{R}^{|E(G)|}$ of G by assigning $x_{(v_i,v_j)}$ a value of 1 if both (y_i, z_j) and (y_j, z_i) are in M, a value of 1/2 if only one of (y_i, z_j) or (y_j, z_i) are in M and the value 0 otherwise. If we consider the edges of G that correspond to non-zero entries in x, they form disjoint paths or cycles (since each vertex has degree at most 2). If we have an even cycle or even path with edges $e_1, e_2, ..., e_{2k}$, for $i = 1...k$ we can set the value of $x_{e_{2i}}$ to zero and $x_{e_{2i-1}}$ to one. We still have a fractional matching and now we no longer have any even cycles, or even paths. We can't have an odd path of length at least three or we could have found a larger matching in $B(G)$. Now we have a fractional matching consisting of a matching of weight one edges along with a set of disjoint odd cycles of weight 1/2 edges. This algorithm runs in $O(n^3)$ time. For a proof that the size of a maximum matching of $B(G)$ is twice the size of a maximum fractional matching of G see [39]. (It relies on the fact that for bipartite graphs the maximum size of a matching is equal to the maximum size of a fractional matching.)

CHAPTER 4
Fractional Total Colouring

In this chapter we discuss some algorithms for determining the total chromatic number and fractionally total chromatic number of graphs. As discussed in section 2.3, we will use our polynomial time algorithm for determining the fractional chromatic index to obtain algorithms that determine $\mathcal{X}_f^T$ for certain types of graphs. We also use the fact that an algorithm for maximum weight total stable set immediately gives an algorithm for fractional total chromatic number via the ellipsoid method and duality. We also show how to compute the fractional total chromatic number of a graph in polynomial time on average by combining two of these algorithms.

We then give some deterministic results on sparse and dense graphs that will be needed in chapter 5.

4.1 Some Fractional Total Colouring Algorithms

One of the main techniques we use to determine the fractional total chromatic number has three steps. We first fix a vertex colouring. Then we use our polynomial time algorithm for determining the fractional chromatic index. Lastly, we combine the colourings to get a fractional total colouring. We will give a few examples of algorithms that use this approach.

In [21], Kilakos and Reed give an algorithm that fractionally $\Delta + 2$ total colours all simple graphs. This once again uses the technique of fixing a vertex colouring then exploiting our knowledge of fractional edge colouring to get the desired colouring. We present a simplified version of this algorithm that shows that all graphs have a $\Delta + 3$ fractional total colouring.

Lemma 12 *For any simple graph G, $\mathcal{X}_f^T(G) \leq \Delta + 3$.*

Proof. Vertex colour G using colours $\{1, ..., \Delta + 3\}$ greedily. This is always possible since any uncoloured vertex v has at most Δ neighbours and hence at least three free colours will be available to be used on v. For $i = 1, ..., \Delta + 3$ let S_i be the set of vertices coloured i. Now consider the vertex induced subgraphs $G_i = G \setminus S_i$. We know that G_i is $\Delta + 1$ edge colourable; let $M_{i,1}, ..., M_{i,\Delta+1}$ be the matchings of a $\Delta + 1$ edge colouring of G_i. For $i = 1, ..., \Delta + 3$ and $j = 1, .., \Delta + 1$ let $T_{i,j}$ be the total stable set $S_i \cup M_{i,j}$. Assign each such total stable set a weight of $\frac{1}{\Delta+1}$ and every other total stable set a weight of 0.

We claim this yields a $\Delta + 3$ fractional total colouring. One can easily see that the sum of the weights assigned to all the total stable sets is $\Delta + 3$. To verify our claim we need only ensure that the constraints of the LP are satisfied.

For every vertex $v \in V(G)$, v is in exactly $\Delta + 1$ total stable sets which have weight $\frac{1}{\Delta+1}$ so the constraints for each vertex are satisfied.

For every edge $e \in E(G)$, let $e = (u, v)$ and let $u \in S_{i'}$ and $v \in S_{j'}$, then e is in exactly one $M_{i,j}$ for all $j \neq j', i'$. So the sum of the weights of all total stable sets containing e is $\sum_{l=1..\Delta+3:l\notin\{i',j'\}} \frac{1}{\Delta+1} = 1$. ■

If the graph has the property that vertices of maximum degree Δ and vertices of degree $\Delta - 1$ are far apart, the above approach can be modified to efficiently find a fractional $\Delta + 1$ total colouring. The *length of a path* between two vertices in G is the minimum number of edges in $E(G)$ needed to keep them connected.

Lemma 13 *If a graph G contains neither a path of length at most* 3 *between two vertices of degree $\Delta(G)$, nor an edge between two vertices of* *degree at least $\Delta(G)-1$, then G has a $\Delta(G)+1$ fractional total colouring. This colouring can be found in $O(n^5)$ time.*

Proof. We construct a special vertex colouring of G using colours $\{1, ..., \Delta+1\}$. We then use this vertex colouring to get a fractional total colouring of G. For each vertex v of degree $\Delta(G)$ colour v and $N(v)$ using all $\Delta(G)+1$ colours (ie. each neighbour of v gets a different colour). Since every two vertices of degree Δ are at distance at least four, no edge has two endpoints with the same colour. We can greedily extend this partial vertex colouring of G to a complete $\Delta(G)+1$ vertex colouring C of G. We use S_i to denote the set of vertices of $V(G)$ assigned colour i by C . It takes $O(|E(G)|)$ time to get our $\Delta+1$ vertex colouring.

For all $i = 1, ..., \Delta(G)+1$, define G_i to be the vertex induced subgraph of $V(G) \setminus S_i$. We will bound the chromatic index of $G_i, \forall i$ using Lemma 1. For all v with $deg_G(v) = \Delta(G)$ and all i if $v \notin S_i$ then it has a neighbour in S_i so $\forall i$, $\Delta(G_i) \leq \Delta(G)-1$. If $\Delta(G_i) < \Delta(G)-1$ then G_i has a $\Delta(G)-1$ edge colouring by the fact that $\mathcal{X}^e(G) \leq \Delta(G)+1$. Otherwise $\Delta(G_i) = \Delta(G)-1$, and so by the hypothesis, the vertices of $\Delta(G)-1$ in G_i are a stable set. By Lemma 1, G_i has a $\Delta(G)-1$ edge colouring in this case as well. So $\forall i, \mathcal{X}^e(G_i) \leq \Delta(G)-1$. We can use Fournier's algorithm to get the edge colouring of each subgraph in $O(n^4)$ time, so it takes $O((\Delta+1)(n^4)) = O(n^5)$ time to edge colour all the subgraphs.

We are going to combine the $\Delta(G)-1$ edge colourings of the G_i's with the vertex colouring C to get a total fractional colouring of G, using the approach of [21].

Let $\mathcal{M}_i = \{M_{i,1}, ..., M_{i,\Delta(G)-1}\}$ be the set of matchings in a $\Delta(G)-1$ edge colouring of G_i. For $1 \leq i \leq \Delta(G)+1$ and j between 1 and $\Delta(G)-1$, we let $T_{i,j}$ be the total stable set $S_i \cup M_{i,j}$ We assign weights of $w_{T_{i,j}} = 1/(\Delta(G) - 1)$ to each $T_{i,j}$ and a weight of zero to all the other total stable sets of G (this takes $O((\Delta + 1)(\Delta - 1))$ time). We now claim that w is a feasible solution to the fractional total colouring of G and $\sum_{T \in \mathcal{T}(G)} w_T = \Delta(G) + 1$.

To prove that the inequalities of the fractional total LP are satisfied, we need to show that every element of $V(G) \cup E(G)$ is in $\Delta(G) - 1$ of the $T_{i,j}$'s. This is clear for $v \in V(G)$ as v is in some S_i and hence in $T_{i,j}$ for $1 \leq j \leq \Delta(G) - 1$. For each $e \in E(G)$ with one end in S_k and the other in S_l, for all $1 \leq i \leq \Delta + 1$ with $i \notin S_k \cup S_l$, there is some j such that $e \in M_{i,j}$ and hence in $T_{i,j}$ so e is in at least $\Delta - 1$ of the $T_{i,j}$'s.

The second part of the claim obviously holds, as our objective function has the following value,

$$\begin{aligned} \sum_{T \in \mathcal{T}(G)} w_T &= \sum_{i=1}^{\Delta(G)+1} \sum_{M_{i,j} \in \mathcal{M}_i} (1/(\Delta(G) - 1)) \\ &= \sum_{i=1}^{\Delta(G)+1} 1 = \Delta(G) + 1 \end{aligned}$$

We've shown that we have a $\Delta(G) + 1$ fractional total colouring of G as required. The algorithm runs in $O(n^5)$ time. ■

4.2 Fractional Total Chromatic Number in $O(2^n n^9)$

To develop our algorithm for fractional total chromatic number, we will need an algorithm for maximum weight total stable set. Given a graph G with weights $w_l, \forall l \in E(G) \cup V(G)$, consider all possible 2^n subsets of $V(G)$, and label them $V_i \subseteq V(G)$ for $1 \leq i \leq 2^n$. If V_i is a stable set then let G_i be the induced graph on $V(G) \setminus V_i$. We can then use a maximum weight matching algorithm to compute a maximum matching M_i of G_i in $O(n^3)$ time. Let T_i be the total stable set $V_i \cup M_i$. The sum of the weights associated with the elements of T_i is $w_{T_i} = \sum_{l \in T_i} w_l$. Now the T_i with that largest w_{T_i} is clearly a maximum weight total stable set of G. We can therefore solve maximum weight total stable set in $O(2^n n^3)$ time.

Let $\mathcal{T}(G)$ be the set of all total stable sets of G and let Q be the polytope defined by:

$$\begin{aligned} x_u &\geq 0 \quad ,\forall\ u \in E(G) \cup V(G) \\ \sum_{u \in T_i} x_u &\leq 1 \quad ,\forall\ T_i \in \mathcal{T}(G) \end{aligned}$$

Then Q is the polytope of the dual of fractional total chromatic number. A strong optimization algorithm over the total stable set polytope (ie. maximum weight total stable set) is in fact a strong separation algorithm over Q. This implies that we have a $O(2^n n^3)$ time algorithm for strong separation over Q. We can therefore optimize the dual of fractional total colouring via ellipsoid method, hence compute the fractional total chromatic number.

Polytope Q can be bounded by a sphere of radius $\sqrt{|E(G)| + |V(G)|}$ and must contain a sphere of radius $\frac{1}{\sqrt{2^{|E(G)|+|V(G)|}}}$. By Lemma 4, the ellipsoid method takes $O(|E(G) \cup V(G)|^2 2^n n^3 \log(\sqrt{2^{|E(G)|+|V(G)|}(|E(G)| + |V(G)|)}) = O(2^n n^9)$ time to optimize over Q. We can therefore compute the fractional total chromatic number of any graph in $O(2^n n^9)$ time.

4.3 $\Delta + 1$ Fractionally Total Colouring Most Graphs

McDiarmid and Reed in [32] gave an algorithm that $\Delta + 1$ total colours almost all graphs. We give an outline of a simplified version of this algorithm that gives a fractional $\Delta + 1$ total colouring of almost all graphs.

Assume $\Delta(G) \geq \frac{49n}{100}$.

$\lceil \frac{n}{100} \rceil$ vertex colour G using colour classes $S_1, ..., S_{\lceil \frac{n}{100} \rceil}$ of size at most 200.

Set $G_1 = G$.

For $i = \{1, ..., \lceil \frac{n}{100} \rceil\}$ do:

- Choose a special vertex v_i of $G_i - S_i$ of degree between $\frac{47n}{100}$ and $\Delta(G) - 2$ in G which was not special in any previous iteration.
- Find a matching $M_i \in G_i - S_i$ hitting every vertex of degree at least $\frac{47n}{100}$ in G except possibly v_i.
- Set $T_i = M_i \cup S_i$ and $G_{i+1} = G_i - M_i$.

Fractionally $\Delta(G) - \lceil n/100 \rceil + 1$ edge colour $G_{\lceil n/100 \rceil + 1}$.

Combining this fractional edge colouring with $T_1, ..., T_{\lceil n/100 \rceil}$ yields the desired colouring.

If each step of this algorithm is successful, it's not difficult to see we end up with a valid fractional total colouring. Note, $G_{\lceil n/100 \rceil + 1}$ has maximum degree $\Delta(G) - \lceil n/100 \rceil + 1$ because each vertex v of degree Δ or $\Delta - 1$ in G is hit by all of the M_i except for the i with $v \in S_i$. Similarly if v has degree between $\Delta - 2$ and $\Delta(G) - \lceil n/100 \rceil - 1$ it is hit by every M_i except the i such that $v \in S_i$ and possibly the i where $v_i = v$ (ie. v was chosen as the special vertex). Thus $\Delta(G_{\lceil n/100 \rceil} + 1)$ is at most $\Delta(G) - \lceil n/100 \rceil + 1$ and its maximal degree vertices had degree $\Delta(G)$ or $\Delta(G) - 1$ in G. So in the last step we are looking for a $\Delta(G_{\lceil n/100 \rceil + 1})$ fractional edge colouring of $G_{\lceil n/100 \rceil + 1}$.

We don't show it here, but this is approach will work on almost all graphs. In fact, in [32] it is shown that the proportion of graphs for which any of the following properties fail is at most $n^{-(\frac{1}{8} + o(1))n}$.

(A) The graph has $\Delta \geq \frac{49n}{100}$.

(B) The desired vertex colouring of G exists.

(C) The number of vertices of degree between $\Delta - 2$ and $\frac{49n}{100}$ is at least $\frac{2n}{100}$. So, regardless of our choices so far at each iteration there is a valid choice of special vertex.

(D) For every valid colouring and choice of special vertices and choice of matchings $M_1, ..., M_i$, the desired M_{i+1} exists.

(E) For every valid colouring, choice of special vertices, and matchings $M_1, ..., M_i$, $G_{\lceil n/100 \rceil + 1}$ has the desired fractional edge colouring.

It also not too hard to show by analyzing a greedy colouring algorithm that the following property:

(B') A $O(|E(G)|)$ greedy colouring algorithm gives us our desired vertex colouring of G.

fails on at most $n^{-(\frac{1}{8}+o(1))n}$ proportion of graphs. We can check for property (A) in $O(|E(G)|)$ time. We can select our special vertex v_i assuming (C) holds in $O(n)$ time by choosing any element of the set. We can also get our desired matchings (D) covering vertices of large degree in polynomial time by finding a maximum matching in the graph obtained from G_i by adding a clique of $|V(G_i)|$ vertices each of which is adjacent to all the vertices of G_i which do not need to be covered (this takes $O(n^3)$ time). We can check that $G_{\lceil n/100 \rceil + 1}$ has a $\Delta(G_{\lceil n/100 \rceil + 1})$ fractional total colouring in $O(n^4)$ time (by the algorithm of section 3.3). We can therefore check if a graph can be $\Delta + 1$ fractional total coloured by this algorithm in $O(n^4)$ time.

4.4 Polynomial Average Time Fractional Total Colouring

A direct result of combining McDiarmid and Reed's algorithm with our $O(2^n n^9)$ algorithm for computing the fractional total chromatic number on the at most $n^{-(\frac{1}{8}+o(1))n}$ proportion of graphs for which McDiarmid and Reed's algorithm fails is a polynomial average time algorithm for computing the fractional total chromatic number.

4.5 A Deterministic Result on $\mathcal{X}_f^T(G)$ for Sparse Graphs

By sparse we mean a forest plus an edge.

Lemma 14 *A connected graph G which contains at most one cycle has a $\Delta(G)+1$ total colouring, unless*

(a) it is a single edge, in which case $\mathcal{X}^T(G) = \mathcal{X}_f^T(G) = \Delta + 2$

(b) it is a cycle of length $3k+1$, then $\mathcal{X}_f^T(G) = \Delta + 1 + \frac{1}{k}$.

(c) it is a cycle of length $3k+2$, then $\mathcal{X}_f^T(G) = \Delta + 1 + \frac{1}{2k+1}$.

Proof. If G is a single edge then trivially we need 3 colours, one for each vertex and a third for the edge. Since each element of $V(G) \cup E(G)$ is only in one total stable set, we get $\mathcal{X}^T(G) = \mathcal{X}_f^T(G) = \Delta + 2 = 3$. We can get this colouring in $O(n)$ time.

To prove the result for trees, we proceed by induction on the number of vertices. For a tree of 3 vertices it's easy to see that $\mathcal{X}^T(G) = \Delta + 1$. Assume it's true for trees with k vertices. Given a tree G with $k+1$ vertices remove a leaf $v \in V(G)$ where $u \in V(G)$ is adjacent to v. Let $e \in E(G)$ be the edge incident to v. If $\Delta(G-v) < \Delta(G)$ we can $\Delta(G)$ total colour $G-v$. Since we want to $\Delta(G)+1$ total colour G and we have a $\Delta(G)$ total colouring of $G-v$ we have a free colour to use on e. Now we just have to colour v, its colour can't be the same as e or u but this leaves $\Delta(G)-1$ colours available, so G has a $\Delta(G)+1$ total colouring. If $\Delta(G-v) = \Delta(G)$ then edge e only has at most $\Delta(G)-1$ incident edges and one vertex u that it can't conflict with, so there is a free colour to give to e. Similarly we can colour v since it only need not conflict with e and u. This proof gives an algorithmic procedure to $\Delta+1$ total colour trees in $O(n^2)$ time.

The total graph of a cycle is vertex transitive, it follows that the inequality of Lemma 5 holds with equality. Therefore, if $n = 3k$ then $\mathcal{X}_f^T(G) = \Delta + 1$. This follows from the fact that $T(G)$ has a maximum stable set of size $2k$. If $n = 3k+1$ then $T(G)$ has a maximum stable set of size $2k$ and it follows that $\mathcal{X}_f^T(G) = \frac{2(3k+1)}{2k} = 3 + 1/k$. Similarly if $n = 3k+2$, then the maximum stable set of $T(G)$ has size $2k+1$, and it follows that $\mathcal{X}_f^T(G) = \frac{2(3k+2)}{2k+1} = 3 + \frac{1}{2k+1}$.

If G is unicyclic but isn't a cycle, then it must have $\Delta(G) \geq 3$. We can trivially get a total 4-colouring of the cycle. Then the induction argument from the proof of the case where G is a tree can be applied to show that unicyclic G has a $\Delta + 1$ total colouring since $\Delta(G) \geq 3$. Once again, this proof produces an optimal total colouring in $O(n^2)$ time. ■

4.6 Deterministic Results on $\mathcal{X}_f^T(G)$ for Dense Graphs

Hilton [16] proved the following theorem,

Theorem 15 *Let J be a subgraph of the complete graph K_{2k}, let $e' = |E(J)|$, and let j be the size of a maximum matching of J, then $\mathcal{X}^T(K_{2k} \setminus E(J)) \leq 2k$ if and only if $e' + j > k - 1$*

We now treat the fractional total chromatic number of such even cliques.

Lemma 16 *If K_n is the complete graph with n vertices where n is even, then $\mathcal{X}_f^T(K_n) = \Delta(K_n) + 2 = n + 1$.*

Proof. We show that $\mathcal{X}_f^e(K_{n+1}) = \mathcal{X}_f^T(K_n)$. To this end we define e_i to be the edge of K_{n+1} between v_{n+1} and v_i. For each matching M of K_{n+1} if M contains some e_i we let T_M be the total stable set of G consisting of v_i and $M - e_i$. Otherwise we set $T_M = M$. This defines a bijection between the family $\mathcal{T}(K_n)$ of total stable sets of K_n, and the family $\mathcal{M}(K_{n+1})$ of matchings of K_{n+1}. There is a corresponding bijection between fractional edge colourings of K_{n+1} and fractional total colourings of K_n. So we can conclude that $\mathcal{X}_f^T(K_n) = \mathcal{X}_f^e(K_{n+1})$. Now for n even we know from Lemma 9 that $\mathcal{X}_f^e(K_{n+1}) = \max\{\Delta(K_{n+1}), max_{H \subseteq K_{n+1}} \frac{2|E(H)|}{|V(H)|-1}\}$ where $|V(H)|$ is odd. If we take $H = K_{n+1}$ we have $|V(H)|$ odd and we get that

$\mathcal{X}_f^e(K_{n+1}) \geq \frac{2|E(H)|}{|V(H)|-1} = \frac{2(n+1)n}{2n} = n+1$. By Vizing's theorem for edge colouring we know $\mathcal{X}_f^e(K_{n+1}) \leq \Delta(K_{n+1}) + 1 = n+1$. So we can conclude that, for n even, $\mathcal{X}_f^T(K_n) = \Delta(K_n) + 2 = n+1$. ∎

Lemma 17 *Let J be a subgraph of K_{2n} with e' edges whose maximum fractional matching has value j. Then $\mathcal{X}_f^T(K_{2n} \setminus E(J)) = 2n + 1 - \frac{e'+j}{n}$ when $\frac{e'+j}{n} \leq 1$.*

Proof. Let $\epsilon = 1 - \frac{e'+j}{n}$.

Proof of necessity (ie. if $\mathcal{X}_f^T(K_{2n} \setminus E(J)) \geq 2n + \epsilon$ then $\epsilon \geq 1 - \frac{e'+j}{n}$).

This part of the proof follows Hilton's [16] proof of Theorem 15, with necessary modifications because we are treating fractional total colouring, not total colouring.

Suppose $G = K_{2n} \setminus E(J)$ has a fractional $2n + \epsilon$ total colouring. We can assume that $\forall v \in V(G) \sum_{T \ni v} w_T = 1$. This follows since if $\sum_{T \ni v} w_T = b > 1$ for some $v \in V(G)$, we can assume removing v from some non-zero weight total stable set will drop $\sum_{T \ni v} w_T$ below 1 otherwise we would do so, we could now split a total stable set T_i of weight $w_i > 0$ into two total stable sets $T_i' = T_i$ and $T_i'' = T_i - \{v\}$ of weights $w_i' = w_i - (b-1)$ and $w_i'' = b - 1$ respectively, by repeating this process for each vertex we get a $2n + \epsilon$ fractional total colouring where $\forall v \in V(G)$, $\sum_{T \ni v} w_T = 1$. Let $T_1, ..., T_l$ be those total stable sets with non-zero weights $w_1, ..., w_l$ in the $2n+\epsilon$ fractional total colouring which contain at least one vertex. Let S_i be $T_i \cap V(G)$ and M_i be $T_i \cap E(G)$. Let $x_i = |S_i|$ then,

$$w_1 x_1 + w_2 x_2 + ... + w_l x_l = 2n$$

There is a matching of J, whose vertices are in S_i, of size $\lfloor \frac{x_i}{2} \rfloor$. Since j is the size of a largest fractional matching of J we see:

$$w_1 \lfloor \frac{x_1}{2} \rfloor + w_2 \lfloor \frac{x_2}{2} \rfloor + ... + w_l \lfloor \frac{x_l}{2} \rfloor \leq j \tag{4.1}$$

Let $z = \sum_{\{i:x_i \ is \ odd\}} w_i$. Then it follows that,

$$w_1 x_1 + w_2 x_2 + ... + w_l x_l \leq 2j + z$$

Therefore,

$$z \geq 2n - 2j. \tag{4.2}$$

Call a pair (T_i, v) a *vertex total stable set pair* if either $v \in S_i$ or v is an endpoint of an edge of M_i. We associate the weight w_i of total stable set S_i with each (S_i, v) pair. We will consider the sum of the weights of all vertex total stable set pairs.

A total stable set with an odd number of vertices is in at most $2n - 1$ vertex total stable set pairs. A total stable set with an even number of vertices is in at most $2n$ vertex total stable set pairs. It follows from (4.2) that the sum of the weights of all vertex total stable set pairs is at most,

$$\begin{aligned} (2n-1)z + (2n)(2n + \epsilon - z) &= 4n^2 + 2n\epsilon - z \\ &\leq 4n^2 - 2n + 2j + 2n\epsilon \end{aligned} \tag{4.3}$$

Since the sum of the weights of all vertex total stable sets containing a vertex v is $1 + deg(v)$, it follows that the sum of the weights of all vertex total stable sets is,

$$\begin{aligned} 2n + \sum_{v \in V(G)} deg(v) &= 2n + 2|E(G)| \\ &= (2n)^2 - 2e' \end{aligned} \tag{4.4}$$

Combining equation (4.3) with equation (4.4) we get that,

$$4n^2 - 2e' \leq 4n^2 - 2n + 2j + 2n\epsilon \tag{4.5}$$

Rearranging gives us our desired result that $\epsilon \geq 1 - \frac{e'+j}{n}$.

Proof of sufficiency (ie. if $\epsilon = 1 - \frac{e'+j}{n}$ then $\mathcal{X}_f^T(K_{2n} \setminus E(J)) \leq 2n + \epsilon$).

Let $G = K_{2n} \setminus E(J)$ and let R be the set of edges in a maximal fractional matching of J which is half integral and let $r = |R|$. Clearly $e' \geq r$ and $j \geq \frac{r}{2}$, so $r \leq \frac{2}{3}(n - n\epsilon)$. More strongly, for every vertex v letting $r_v = |\{e \in R : v \in e\}|$ we have $e' \geq r + (def_G(v) - r_v)$, so $r \leq \frac{2}{3}(n - n\epsilon - def_G(v) + r_v)$. We enumerate R as $\{e_1 = (x_1, y_1), ..., e_r = (x_r, y_r)\}$ and let $w_i \in \{0, \frac{1}{2}, 1\}$ be the weight of e_i in our optimal fractional matching. We find disjoint matchings $M_1, ..., M_r$ in G such that M_i is a perfect matching in $G_i = G - x_i - y_i - \bigcup_{j<i} M_j$. By Lemma 6, to prove we can do so it is enough to show that every vertex v of G_i, $def_{G_i}(v) \leq \frac{|V(G_i)|}{2} = n - 1$.

Now,

$$\begin{aligned} def_{G_i}(v) &\leq def_G(v) + |\{j : j < i, \exists e \in M_j \ s.t. \ v \in e\}| \\ &\leq def_G(v) + r - r_v \\ &\leq def_G(v) + \frac{2}{3}(n - n\epsilon - def_G(v) + r_v) - r_v \\ &\leq \frac{1}{3}def_G(v) + \frac{2}{3}n - \frac{2}{3}n\epsilon - \frac{1}{3}r_v \end{aligned}$$

Since $def_G(v) \leq n - 1$ we get $def_{G_i}(v) < n$ as required.

Having constructed $M_1, ..., M_r$ we define total stable sets $T_1, ..., T_r$ where $T_i = M_i \cup \{x_i, y_i\}$. We give T_i the same weight, w_i, that e_i has in our optimal fractional matching of J. Since our fractional matching was half-integral and every edge with weight 1/2 was in an odd cycle, it follows that for any vertex $v \in V(G)$, $\sum_{T_i \ni v} w_i \in \{0, 1\}$. So every vertex is either completely covered or not covered yet at all.

We construct an auxiliary graph G' from G by adding a vertex v^* adjacent to all of $V(G) - \cup_{e_i \in R}\{x_i, y_i\}$. We weight the edges of G' as follows:

$\forall\ x = v^*v \in E(G')$	$f(x) = 1.$
$\forall\ x \in E(G) - \cup_{i=1}^{r} M_i$	$f(x) = 1.$
$\forall\ x \in M_i$	$f(x) = 1 - w_i.$

We claim that we can find matchings $\{N_1, ..., N_l\}$ of G' and weights $\{z_1, ..., z_l\}$ s.t. $\sum z_i = 2n - j + \epsilon$ and $\forall x \in G'$, $\sum_{\{i: x \in N_i\}} z_i = f(x)$.

Having done so, for $1 \leq i \leq l$ we define a total stable set N'_i of G as follows:

- If $v^* \notin V(N_i)$ then $N'_i = N_i$.
- If $\exists\ u$ s.t. $v^*u \in N_i$ then $N'_i = N_i - v^*u + u$.

It is an easy matter to verify that giving T_i weight w_i and N'_i weight z_i and all other total stable sets weight 0 yields a fractional $2n + \epsilon$ total colouring of G. So it remains only to prove our claim.

By the characterization of the matching polytope, we know that if the claim does not hold then either:

(i) $\exists\ v \in V(G')$ s.t. $\sum_{v \in x} f(x) > 2n - j + \epsilon$, or

(ii) $\exists\ H \subseteq G'$, $|V(H)|$ odd s.t. $\sum_{x \in E(H)} f(x) > (2n - j + \epsilon)(|V(H)| - 1)/2$.

Because, each M_i is a perfect matching of $G - x_i - y_i$, $\forall\ v \in V(G)$, $\sum_{\{x \in E(G'), x \ni v\}} f(x) = deg_G(v) + 1 - j \leq 2n - j$. Clearly $\sum_{x \ni v^*} f(x) = 2n - 2j$. So (i) doesn't hold.

It remains to show that $\forall\ H \subseteq G$, where $|V(H)| \geq 3$ and $|V(H)|$ is odd the following inequality holds $\sum_{x \in E(H)} f(x) \leq (2n - j + \epsilon)(|V(H)| - 1)/2$.

Assume that $H = G'$, then we get,

$$\begin{aligned}\sum_{x \in E(H)} f(x) &= |E(G)| - (n-1)j + (2n - 2j) \\ &= (2n(2n-1)/2 - e') - (n-1)j + (2n - 2j)\end{aligned}$$

$$= \quad 2n^2 - (e' + j) - jn + n$$

But, the right hand side of (ii) is, $(2n - j + \epsilon)(|V(H)| - 1)/2 = 2n^2 - jn + n\epsilon$ and since $(e' + j) \geq n - n\epsilon$ it follows that (ii) doesn't hold for $H = G'$.

Now consider $H \subsetneq G'$ and assume property (ii) holds. A simple property of a subgraph satisfying (ii) is that the sum of the weights on edges leaving H is at most $(2n - j + \epsilon) - 2$. We are going to abuse notation a little and let $deg'(v) = \sum_{x \ni v} f(x)$ and $\Delta'_{min}(G') = \min\{deg'(v) : \forall\ v \in V(G')\}$.

For v^*, each x around it has $f(x) = 1$, so $deg'(v^*) = deg(v^*) = 2n - 2j$. For $v \in V(G)$, $deg'(v) = \sum_{x \ni v} f(x) = deg_G(v) - j + 1$.

So $\Delta'_{min}(G') = \min\{2n - 2j, \Delta_{min}(G) - j + 1\}$. Since $j \leq e'$ and $\Delta_{min}(G) \geq 2n - 1 - e'$ it follows that $\Delta'_{min}(G') \geq 2n - e' - j$.

Let B be the of vertices in G' not in H and let $b = |B|$, (ie. $b = |V(G') \setminus V(H)|$). Then for H to satisfy (ii) the sum of the weights on edges leaving H going to B must be at most $(2n - j + \epsilon) - 2$. This implies that the sum on edges leaving B is at most $(2n - j + \epsilon) - 2$. Assume that the vertices of B form a clique, and each edge x in this clique has $f(x) = 1$ (this minimizes the possible sum of weights on edges H). The sum on edges leaving B must be at least $b\delta'(G') - b(b-1)$. But since $2 \leq b \leq j \leq \frac{n}{2}$ it easy to verify that,

$$b\delta'(G') - b(b-1) \quad > \quad (2n - j + \epsilon) - 2. \tag{4.6}$$

This implies that no such H exists such that (ii) holds. This implies we have our matchings $\{N_1, ..., N_l\}$ with weights $\{z_1, ..., z_l\}$ as required.

Since we can find a fractional matching in $O(n^3)$ time, it follows that we can compute $\mathcal{X}_f^T(K_{2n} \setminus E(J))$ in $O(n^3)$. The second part of this proof is also algorithmic and finds an actual optimal fractional total colouring. It takes $O(n^4)$ time to find all the M_i matchings, we can find our matchings

$\{N_1, ..., N_l\}$ using a modified fractional edge colouring algorithm in $O(FEC)$ time (we don't discuss the details here, but we can construct a multi-graph so that a fractional edge colouring algorithm gives us our desired matchings and weights). So the whole procedure takes $O(n^4 + FEC)$ time. $\blacksquare$

Lemma 18 *For any real $c > 1$, there is an n_c such that for all odd n bigger than n_c the following holds: Suppose G is a graph which has n vertices and $n - \Delta(G)$ disjoint stable sets of size 3. If $\Delta_{min}(G) \geq \frac{n}{2} + n - \Delta(G) + 2$, $\Delta(G) \geq n - (c-1)\log(n)$, $|E(G)| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$, then G has a fractional $\Delta(G) + 1$ total colouring.*

Proof. Consider the $a = n - \Delta(G)$ vertex disjoint stable sets of size 3, call them $S_1, ..., S_a$ and $S_i \cap S_j = \emptyset$ for $i \neq j$.

Consider a perfect matching, M_1 of $G \setminus S_1$. Since $\Delta_{min}(G) \geq n/2 + 3$ we know that $G \setminus S_1$ has a perfect matching by Lemma 6. Define M_i to be a perfect matching of G_i where $V(G_i) = (V(G) \setminus S_i)$ and $E(G_i) = E(G \setminus S_i) \setminus (M_1 \cup M_2 \cup ... \cup M_{i-1})$. G_i has a perfect matching M_i by Lemma 6, the fact that $|V(G_i)|$ is even and $\Delta_{min}(G_i) \geq \Delta_{min}(G) - 3 - (i-1) > \frac{|V(G_i)|}{2}$. Let T_i be the total stable set $S_i \cup M_i$.

Let $V' = S_1 \cup S_2 \cup ... \cup S_a$ and $E' = M_1 \cup M_2 \cup ... \cup M_a$. Consider the auxiliary graph H where $V(H) = V(G)$ plus a new vertex v^* and $E(H) = E(G) \setminus E'$ plus edges $\{(u, v^*) : \forall u \in V(G) \setminus V'\}$.

Now, $\Delta(H) = \Delta(G) - a + 1$ since $\forall v \in V'$, $deg_H(v) = deg_G(v) - a + 1$ this follows since we've removed one edge adjacent to v in each M_j except one. For all $v \in V(G) \setminus V'$, $deg_H(v) = deg_G(v) - a + 1$ since we've removed a edges incident to v and added one adjacent to v^*. Finally $deg(v^*) < \Delta(G) - 2a + 1$ since $deg(v^*) = n - 3a$. So $\Delta(H) = \Delta(G) - a + 1 = n - 2a + 1$. Also, $\Delta_{min}(H) = \min\{deg_H(v^*), \Delta_{min}(G) - a + 1\} \geq n/2 + 1$. We want to show that H has a $\Delta(H)$ fractional edge colouring. By Lemma 9 it suffices to show H doesn't contain an odd overfull subgraph.

To get a contradiction assume $H' \subset H$ is an odd overfull subgraph. By definition this means,

$$\begin{aligned} |E(H')| &> \Delta(H)\frac{(|V(H')|-1)}{2} \\ &= (n-2a+1)\frac{(|V(H')|-1)}{2} \end{aligned}$$

This implies that $|V(H')| > \Delta(H) = n - 2a + 1$. Let $B = V(H) \setminus V(H')$, then $|B| = n + 1 - |V(H')| < 2a$. Therefore,

$$|E(H')| > (n-2a+1)(n-|B|)/2 \tag{4.7}$$

Since, $|E(H)| = |E(G)| - (n-3)a/2 + n - 3a$, we get

$$\begin{aligned} |E(H')| \leq\ & |E(G)| - (n-3)a/2 + n - 3a \\ & -|B|\Delta_{min}(H) + |B|(|B|-1)/2 \end{aligned} \tag{4.8}$$

Combining inequalities (4.7) and (4.8) we find that if

$$|E(G)| \leq \binom{n}{2} - nB/2 - an/2 + aB + 3a/2 + B\Delta_{min}(H) - B^2/2 \tag{4.9}$$

no overfull subgraph can exists. Substituting in the bound on $\Delta_{min}(H)$ we get that no overfull subgraph exists if

$$|E(G)| \leq \binom{n}{2} - an/2 + aB + 3a/2 + B - B^2/2 \tag{4.10}$$

Since $|B| < 2a$ it follows that if

$$|E(G)| \leq \binom{n}{2} - an/2 + 3a/2 \tag{4.11}$$

no overfull subgraph exists. Since $|E(G)| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$ and $a \leq (c-1)\log(n)$ it follows that inequality (4.11) holds for all n at least as large as some n_c and no overfull subgraph H' exists. This means that H has a $\Delta(H)$ fractional edge colouring.

We can now show we have a fractional $\Delta(G)+1$ total colouring of G in a similar fashion as we did in the proof of Lemma 17. Give each total stable set T_i a weight of 1. Now every edge and vertex in T_i is covered. For all matchings M_j of H where M_j has weight w_j in our $\Delta(H)$ fractional edge colouring, if $(v^*, u) \in M_j$ for some vertex u then set total stable set M'_j of G to be $M_j - (v^*, u) \cup \{u\}$. Otherwise set $M'_j = M_j$. Assign M'_j the weight w_j. Now, the total stable sets $\forall i\ T_i$ and $\forall j\ M'_j$ together with their weights form a $\Delta + 1$ fractional total colouring of G.

If we are given our $n - \Delta$ vertex disjoint stable sets of size 3, then we can construct this $\Delta + 1$ fractional total colouring in polynomial time. This follows since it takes at most $O(n^4)$ time to compute our perfect matchings $M_1, ..., M_a$, then we just need to compute our fractional edge colouring of H which we can do in polynomial time. Then combine the two sets of total stable sets. This whole procedure takes $O(n^4 + FEC)$ where FEC is the time needed to compute an optimal fractional edge colouring (discussed in section 3.3). ■

Lemma 19 *For any real $c > 1$, there is an n_c such that for all even n bigger than n_c the following holds: Suppose G is a graph which has n vertices and $n-\Delta(G)$ disjoint stable sets of size 2. If $\Delta_{min}(G) \geq \frac{n}{2} + (n-\Delta(G))+1$, $\Delta(G) \geq n-(c-1)\log(n)$, $|E(G)| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$. Then G has a fractional $\Delta(G)+1$ total colouring.*

Proof. The proof is the same as that of Lemma 18, except we combine $n - \Delta(G)$ stable sets of size 2 with perfect matchings of the rest of the graph. Then we create the auxiliary graph H in the same fashion, and we can show it has a $\Delta(H)$ fractional edge colouring. Finally we can use the total stable sets and the fractional edge colouring of H to show we have a fractional $\Delta(G) + 1$ total colouring of G as before. Once again we can compute this total colouring in polynomial time. In fact we can find the maximum number of vertex disjoint stable sets of size 2 by running a perfect matching algorithm on the complement of G, this takes $O(n^3)$ time. So this algorithm runs again in $O(n^4 + FEC)$. ■

Lemma 20 *If G has $n > 50$ vertices, $\Delta(G) = n - 2$, $\Delta_{min}(G) \geq n/2 + 2$ and $|E(G)| \leq \binom{n}{2} - 9n$, then $\mathcal{X}_f^T(G) = \Delta + 1$.*

Proof. Assume that edges (v_1, v_2) and (v_3, v_4) are in $E(\bar{G})$. Let v_5 and v_6 be two vertices of degree less than $n - 2$. Create two total stable sets $T_1 = \{v_1, v_2\} \cup M_1$, where M_1 is a matching hitting every vertex of $G - v_1 - v_2$ except possibly v_5. Similarly, let T_2 be the total stable set $\{v_3, v_4\} \cup M_2$ where M_2 is a matching hitting every vertex of $G - v_3 - v_4$ except possibly v_6. By Lemma 6 it's not difficult to see that matchings M_1 and M_2 exist.

Construct an auxiliary graph H from G with a new vertex v^* adjacent to all vertices in G except for v_1, v_2, v_3, and v_4. It's not difficult to check that $\Delta(H) = n - 4$ and H can't contain an odd overfull subgraph. This implies we have a $\Delta(H)$ fractional edge colouring of H. We can convert the matchings of the fractional edge colouring to total stable sets of G and combine them with T_1 and T_2 to get a $\Delta + 1$ fractional total colouring of G in the usual manner.

This proof is algorithmic and would take $O(n^3)$ time to find the matchings and $O(FEC)$ for the fractional edge colouring, so $O(n^3 + FEC)$ in total. ■

CHAPTER 5
Colouring Random Graphs

While in many cases finding optimal colourings of simple graphs may be computationally difficult, a typical graph may be easy to colour. Indeed for many standard probability distributions, we can find efficient algorithms to fractionally total colour which asymptotically almost surely work.

An analysis of the structural properties of random graphs allows us to show that using the algorithms of the last chapter we can asymptotically almost surely compute the fractional total chromatic number of the random graph $G_{n,p}$ for all values of $0 \leq p \leq 1$ in polynomial time.

5.1 Probability

Before we can give our algorithm that computes the fractional total chromatic number of a random graph, we need to give a brief introduction to some basic probability.

We consider the $G_{n,p}$ model of random graphs. A graph from the $G_{n,p}$ model is a graph with n vertices in which each of the $\binom{n}{2}$ possible edges is independently present with probability p.

It follows that the degree of a vertex in a random graph has a binomial distribution. So the probability that a vertex v has degree k is the probability that of the $n-1$ possible edges incident to v, k are present, and $n-1-k$ are missing. Now, $b(k; n-1, p) = \binom{n-1}{k} p^k (1-p)^{n-1-k}$ is the probability of this event. We will let $X_k(n, p)$ or simply X_k be the random variable counting the number of vertices of degree k in $G_{n,p}$. For a property A of a graph, we denote the probability that property A holds for $G_{n,p}$ as $\mathbf{Pr}(A)$. We will use $\mathbf{Exp}[A]$ to denote the expected value of the random variable A.

We will use $\mathbf{Var}[X]$ to denote the variance of the random variable X. If the possible values of random variable X are $\{x_1, ..., x_l\}$, then the expectation of X is defined as:

$$\mathbf{Exp}[X] = \sum_{i=1}^{l} x_i \mathbf{Pr}(X = x_i)$$

Similarly, the variance of X is:

$$\mathbf{Var}[X] = \sum_{i=1}^{l} (x_i - \mathbf{Exp}[X])^2 \mathbf{Pr}(X = x_i).$$

So $\mathbf{Exp}[X_k]$ is just $nb(k; n-1, p)$ we will also use $\alpha_k(n)$ to represent this expectation since we will use if often. We say that an property A occurs asymptotically almost surely (a.a.s.) for a random graph $G_{n,p}$ if $\mathbf{Pr}(A) \to 1$ as $n \to \infty$.

We will need to use some basic probabilistic tools to prove that properties of $G_{n,p}$ a.a.s. hold. Two useful tools will be Markov's and Chebychev's inequalities

Lemma 21 (Markov's Inequality) *For a non negative random variable X and positive constant a,*

$$\mathbf{Pr}(X \geq a) \quad \leq \quad \frac{\mathbf{Exp}[x]}{a}$$

Lemma 22 (Chebychev's Inequality) *For a random variable X,*

$$\mathbf{Pr}(|X - \mathbf{Exp}[X]| \geq \epsilon) \quad \leq \quad \frac{\mathbf{Var}[X]}{\epsilon^2}.$$

5.2 Some Previous Results on Colouring $G_{n,p}$

It is known that $G_{n,1/2}$ a.a.s. has a maximum stable set of size $\leq 2\log(n)$ (see **??**). So by Lemma 5 it follows that a.a.s. $\mathcal{X}(G_{n,1/2}) \geq \mathcal{X}_f(G_{n,1/2}) \geq \frac{n}{2\log(n)}$. Bollobás [5], building on work of Matula [31], proved that in fact a.a.s. $\mathcal{X}(G_{n,1/2}) = \frac{n(1+o(1))}{2\log(n)}$. However, there currently are no known algorithms that will a.a.s. colour (or fractionally colour) $G_{n,p}$

with $\frac{n(1+o(1))}{2\log(n)}$ in polynomial time. This stems from the fact that there are currently no known algorithms that will a.a.s. find a stable set of size $(1+\epsilon)\log(n)$ in polynomial time.

We turn to edge colouring random graphs. We denote by p'_n the proportion of graphs on n vertices with $\mathcal{X}^e(G) = \Delta + 1$. In the $G_{n,p}$ model with $p = 1/2$, Erdös and Wilson [11] showed that $p'_n \to 0$ as $n \to \infty$. Frieze, Jackson, McDiarmid, and Reed [1] strengthened these results to show that $n^{-(1/2+o(1))n} \leq p'_n \leq n^{-(1/8+o(1))n}$ as $n \to \infty$.

To develop algorithms for fractionally total colouring $G_{n,p}$ we need to combine some deterministic results on fractional total colouring with some probabilistic results on the structure of $G_{n,p}$. The latter are found in section 5.3, the former in chapter 4. We combine them in section 5.4.

5.3 Structural Properties of $G_{n,p}$

We now analyze the degree sequence of $G_{n,p}$ and other structural properties of it. The following three theorems of Bollobás [6] will be useful:

Theorem 23 *Let $\epsilon > 0$ be fixed, $\epsilon n^{-3/2} \leq p(n) \leq 1 - \epsilon n^{-3/2}$, let $k = k(n)$ be a natural number and set $\alpha_k = \alpha_k(n) = nb(k; n-1, p)$. Let X_k be the random variable representing the number of vertices of degree k. Then the following assertions hold,*

(i) If $\lim \alpha_k(n) = 0$, then a.a.s. $\mathbf{Pr}(X_k = 0) = 1$.

(ii) If $\lim \alpha_k(n) = \infty$, then a.a.s. $\mathbf{Pr}(X_k \geq t) = 1$ for every fixed t.

(iii) If $0 < \underline{lim}\alpha_k(n) \leq \overline{lim}\alpha_k(n) < \infty$ then X_k has asymptotically Poisson distribution with mean α_k:

$$\mathbf{Pr}(X_k = r) \approx e^{-\alpha_k}\alpha_k^r/r!$$

for every fixed r.

Here $\underline{lim}\alpha_k(n) = h$ exists if $|\alpha_k(n) - h| < \epsilon|$ for all $\epsilon > 0$ for infinitely many values of n and no number less than h has this property ($\overline{lim}\alpha_k(n) = h$ is defined similarly where no number larger than h has the property). Note, we will also use an analogous result that a.a.s. $\Delta_{min}(G_{n,p}) \leq k$, if

$\lim \frac{1}{n^{k-1}} \sum_{l \leq k} \alpha_k(n) \to \infty$.

Theorem 24 *If* $\frac{pn}{\log(n)} \to \infty$ *and* $\frac{(1-p)n}{\log(n)} \to \infty$ *then a.a.s.* $G_{n,p}$ *has a unique vertex of maximum degree and a unique vertex of minimum degree.*

Remark: Bollobás proved this for $p \leq 1/2$ and $\frac{pn}{\log(n)} \to \infty$ but this version follows by symmetry.

Theorem 25 *Given a labelling of vertices* $x_1, x_2, ...x_n \in V(G_{n,p})$ *and respectively corresponding degrees* $d_1, d_2, ..., d_n$*, such that* $d_1 \geq d_2 \geq ... \geq d_n$*. If* $m = o(\frac{p(1-p)n}{logn})^{\frac{1}{4}}$*,* $m \to \infty$*, and* $\alpha(n) \to 0$ *then a.a.s.*

$$d_i - d_{i+1} \geq \frac{\alpha(n)}{m^2}\left(\frac{p(1-p)n}{\log n}\right)^{\frac{1}{2}} \qquad \forall i < m$$

Corollary 26 *If* $\frac{pn}{\log(n)} \to \infty$ *and* $\frac{(1-p)n}{\log(n)} \to \infty$ *then a.a.s.* $G_{n,p}$ *has a unique vertex of degree* Δ *and all other vertices have degree* $\leq \Delta - 2$*.*

Proof. This is a direct result of Theorem 24 and Theorem 25. Let $m = (\frac{p(1-p)n}{\log(n)})^{\frac{1}{8}}$, since p satisfies both $np/\log(n) \to \infty$ and $(1-p)n/\log(n) \to \infty$ it follows that $m \to \infty$. Let $\alpha(n) = 2m^2(\frac{\log(n)}{p(1-p)n})^{\frac{1}{2}}$, then it's easy to see that $\alpha(n) \to 0$. We can now use the Theorem 25 to show that asymptotically almost surely,

$$d_i - d_{i+1} \geq \frac{2m^2(\frac{\log(n)}{p(1-p)n})^{\frac{1}{2}}}{m^2}\left(\frac{p(1-p)n}{\log(n)}\right)^{\frac{1}{2}} \qquad \forall i < m$$

Simplifying we get that,

$$d_i \geq d_{i+1} + 2 \qquad \forall i < m$$

In particular this gives us our desired result that $d_1 \geq d_2 + 2$. Therefore if $v^* \in V(G_{n,p})$ has degree $\Delta(G_{n,p})$ then $\forall u \in V(G_{n,p}) - v^*$, a.a.s. $deg(u) \leq \Delta(G_{n,p}) - 2$. ■

Lemma 27 *If $p = o(n^{-2})$ then a.a.s. $G_{n,p}$ has no edges.*

Proof. The expected number of edges is pn^2 which is $o(1)$ in this range. ■

Simple calculations also yield:

Lemma 28 *If $p = (c + o(1))n^{-2}$ for some constant c then a.a.s. every component of $G_{n,p}$ contains at most one edge, and the probability $G_{n,p}$ has an edge is bounded away from both 0 and 1.*

Lemma 29 *If $p = \omega(n^{-2})$ and $p = o(n^{-3/2})$ then a.a.s. $|E(G_{n,p})| \geq 1$ and a.a.s. every component of $G_{n,p}$ contains at most one edge.*

Lemma 30 *If $p = (c + o(1))n^{-3/2}$ then a.a.s. every component of $G_{n,p}$ has at most two edges and the probability $G_{n,p}$ has a two edge component is bounded away from both 0 and 1.*

Proof. Let C be the random variable for the number of components with more than 2 edges. We can bound the expectation of C as,

$$\begin{aligned} \mathbf{Exp}[C] &\leq \binom{n}{3}p^3 + 16\binom{n}{4}p^3 \\ &= o(1) \end{aligned}$$

So a.a.s. every component of $G_{n,p}$ has at most 2 edges. The expected number of vertices of degree 2 is given by,

$$\begin{aligned} \mathbf{Exp}[X_2] &= n(n-1)(n-2)p^2(1-p)^{n-3} \\ &\approx (c + o(1))^2 \end{aligned}$$

By Theorem 23 property (iii) this implies the probability we have a component with two edges is bounded away from zero and one. ■

Lemma 31 *If $p = \Omega(n^{-3/2})$ and $p = o(n^{-1})$ then a.a.s. $\Delta(G_{n,p}) \geq 2$ and every component of $G_{n,p}$ is a tree.*

Proof. By Theorem 23 we know that $\mathbf{Pr}(X_2 \geq 1) \to 1$ when $\lim \mathbf{Exp}[X_2] \to \infty$. For our possible values of p, $\lim \mathbf{Exp}[X_2] = \lim n\binom{n-1}{2}p^2(1-p)^{n-3} \to \infty$. This implies that a.a.s. $\Delta(G_{n,p}) \geq 2$. The expected number of cycles is,

$$\begin{aligned}\mathbf{Exp}[\#\ of\ cycles] &= \sum_{k=3}^{n}\binom{n}{k}\frac{(k-1)!}{2}p^k \\ &< \sum_{k=3}^{n}\frac{(np)^k}{2k}\end{aligned}$$

So $\mathbf{Pr}(\#\ cycles\ \geq 1) \leq \mathbf{Exp}[\#\ of\ cycles] = o(1)$ when $p = o(n^{-1})$, and a.a.s. only trees occur. ■

Lemma 32 *If $p = \Omega(n^{-1})$ and $p < \frac{1-\epsilon}{n}$ for some $\epsilon > 0$ then a.a.s. $\Delta(G_{n,p}) \geq 3$ and every component of $G_{n,p}$ is unicyclic.*

Proof. Simple calculations show $\lim \mathbf{Exp}[X_3] = \lim \binom{n-1}{3}p^3(1-p)^{n-4} \to \infty$ in our range of p. If follows from Theorem 23 that a.a.s. $\Delta(G_{n,p}) \geq 3$.

For the proof that every component of $G_{n,p}$ is unicyclic see either [6] or [38]. ■

Lemma 33 *If $p = o(\log(n)/n)$ then a.a.s $\Delta(G_{n,p}) \leq \log(n)$.*

Proof. Let $p = \frac{d(n)}{n}$ where $d(n) = o(\log(n))$. Let Y be number of $\log(n)$ sets of edges around a fixed vertex v. Then $\mathbf{Exp}[Y] = \binom{n-1}{\log(n)}p^{\log(n)}$. If we let Z be the number of vertices of degree at least $\log(n)$ we get,

$$\begin{aligned}\mathbf{Exp}[Z] &\leq n\mathbf{Exp}[Y] \\ &= n\binom{n-1}{\log(n)}(\frac{d(n)}{n})^{\log(n)} \\ &\leq n(\frac{e(n-1)}{\log(n)})^{\log(n)}(\frac{d(n)}{n})^{\log(n)} \\ &= o(1)\end{aligned}$$

Since Z is integer valued, it follows that

$$\begin{aligned} \mathbf{Pr}(Z > 0) &\leq \mathbf{Exp}[Z] \\ &= o(1). \end{aligned}$$

So a.a.s. $\Delta(G_{n,p}) \leq \log(n)$. ■

Lemma 34 *For $p = (c + o(1)) \log(n)/n$ for some constant c with $0 < c < \infty$, $G_{n,p}$ a.a.s. has $\frac{c}{5} \log(n) \leq \Delta(G_{n,p}) \leq 3ec \log(n)$.*

Proof. Standard concentration bounds on the binomial distribution ($Bin(\binom{n}{2}, p)$) tell us that a.a.s. $G_{n,p}$ has $(c + o(1))n \log(n)(1 - o(1))$ edges which yields our lower bound.

The upper bound can be proved by following same procedure as the proof of Lemma 33. We let Y be the number of $\lceil 3ec \log(n) \rceil$ sets of edges around a fixed vertex v. If Z is the number of vertices of degree at least $\lceil 3ec \log(n) \rceil$, we can easily show that $\mathbf{Exp}[Z] \leq n\mathbf{Exp}[Y] = o(1)$.

Since Z is integer valued, it follows that

$$\begin{aligned} \mathbf{Pr}(Z > 0) &\leq \mathbf{Exp}[Z] \\ &= o(1). \end{aligned}$$

So a.a.s. $\Delta(G_{n,p}) \leq 3ec \log(n)$. ■

Lemma 35 *If $p = 1 - \frac{(c+o(1)) \log(n)}{n}$ for some constant c with $1 \leq c < \infty$ then a.a.s. $\Delta(G_{n,p}) \geq n - (c - 1) \log(n)$.*

Proof. For $\bar{G}_{n,p}$, $\lim \frac{1}{n^{k-1}} \sum_{l \leq k} \mathbf{Exp}[X_k] \to \infty$ when $k = \lfloor (c - 1) \log(n) \rfloor - 1$. By the analogous result of Theorem 23 it follows that a.a.s. $\Delta_{min}(\bar{G}_{n,p}) \leq (c - 1) \log(n) + 1$. So a.a.s. $\Delta(G_{n,p}) \geq n - (c - 1) \log(n)$. ■

Let A be the random variable counting the number of pairs of vertices such that either both have maximum degree and have a path of length ≤ 3

between them, or an edge exists between the pair and both have degree $\geq \Delta(G_{n,p}) - 1$. Our interest in A is explained by Lemma 13.

Lemma 36 *If* $\frac{1}{n} \leq p = \frac{o(log(n))}{n}$ *then* $\mathbf{Pr}(A \neq 0) = o(1)$.

Proof. In [6] Bollobás proved that if $\frac{1}{n} \leq p = o(\log(n)/n)$ then there is some $k = k(n)$ which is between $\frac{\log(n)}{\log\log(n)}$ and $\log(n)$ such that a.a.s. $\Delta(G_{n,p}) = k$ and furthermore $\mathbf{Exp}[X_{k+1}] = o(1)$. Now,

$$\mathbf{Exp}[X_{k+1}] = n\binom{n-1}{k+1}p^{k+1}(1-p)^{n-2-k}$$

and,

$$\mathbf{Exp}[X_k] = n\binom{n-1}{k}p^{k}(1-p)^{n-1-k}$$

So, $\mathbf{Exp}[X_k] = \frac{(k+1)p^{-1}(1-p)}{n-k-2}\mathbf{Exp}[X_{k+1}]$. Thus since $\mathbf{Exp}[X_{k+1}] = o(1)$ and $k = O(\log(n))$ we know $\mathbf{Exp}[X_k] = O(\log(n))$.

Since $\mathbf{Exp}[X_k] = n\binom{n-1}{k}p^k(1-p)^{n-1-k} = O(\log(n))$ it follows that

$$\binom{n-1}{k}p^{k}(1-p)^{n-1-k} = O(\frac{\log(n)}{n}) \tag{5.1}$$

The probability $G_{n,p}$ has two vertices of degree Δ with an edge between them is given by,

$$\begin{aligned}
&\binom{n}{2}\mathbf{Pr}(deg(u) = deg(v) = \Delta \ and \ (u,v) \in E(G_{n,p})) \\
=\ &\binom{n}{2}(\binom{n-2}{k-1}p^{k-1}(1-p)^{n-1-k})^2 p \\
=\ &\binom{n}{2}O(\log^2(n)/n)^2 p && (5.2) \\
=\ &o(1) && (5.3)
\end{aligned}$$

equation (5.2) follows from (5.1), since $\binom{n-2}{k-1} = \frac{k}{n-1}\binom{n-1}{k}$.

Similarly we can bound the probability we that a vertex of degree Δ is adjacent to a vertex of degree $\Delta - 1$ or a vertex of degree $\Delta - 1$ is adjacent to a vertex of degree $\Delta - 1$ by,

$$\begin{aligned}
& \binom{n}{2}p((\binom{n-2}{k-1}p^{k-1}(1-p)^{n-1-k})(\binom{n-2}{k-2}p^{k-2}(1-p)^{n-k}) \\
& +(\binom{n-2}{k-2}p^{k-2}(1-p)^{n-k})^2) \\
= & \binom{n}{2}p(O(\log^5(n)/n^2) + O(\log^6(n)/n^2)) \qquad (5.4) \\
= & o(1) \qquad (5.5)
\end{aligned}$$

The probability that we have a path of length two between vertices of degree Δ can be bounded by,

$$\begin{aligned}
\binom{n}{3}p^2(\binom{n-3}{k-1}p^{k-1}(1-p)^{n-2-k})^2 &= \binom{n}{3}p^2(O(\log^2/n)^2) \qquad (5.6) \\
&= o(1) \qquad (5.7)
\end{aligned}$$

The probability that we have a path of length three between vertices of degree Δ can be bounded by,

$$\begin{aligned}
\binom{n}{4}p^3(\binom{n-4}{k-1}p^{k-1}(1-p)^{n-3-k})^2 &= \binom{n}{4}p^3(O(\log^2/n)^2) \qquad (5.8) \\
&= o(1) \qquad (5.9)
\end{aligned}$$

Combining equations (5.3), (5.5), (5.7) and (5.9) we get $\mathbf{Pr}(A \neq 0) = o(1)$. ■

Lemma 37 *If* $p = \frac{(c+o(1))\log(n)}{n}$ *for fixed* c *and* $0 < c < \infty$ *then* $\mathbf{Pr}(A \neq 0) = o(1)$.

Proof. Bollobás proved in [6] that for $p = (c + o(1))\log(n)/n$, where c is a constant, a.a.s. $\Delta(G_{n,p})$ can't be confined to a finite set of values. By Lemma 34 we know a.a.s. $\frac{c}{5}\log(n) \leq \Delta(G_{n,p}) \leq 3ec\log(n)$. Let j be the largest value such that $\lim \mathbf{Exp}[X_j] = \infty$, we know from Lemma 34 that $j \geq \frac{c}{5}\log(n)$. By Theorem 23 we know that for $k > j$, $\lim \mathbf{Exp}[E_k] = C$ where $0 \leq C < \infty$. Now, $\mathbf{Exp}[X_j] = \mathbf{Exp}[X_{j+1}]\frac{(j+1)p^{-1}(1-p)}{n-1-j}$ since

$\lim \mathbf{Exp}[X_{j+1}] = C$ and $j = O(\log(n))$ it follows that $\mathbf{Exp}[X_j] = O(\log(n))$. So for $k \geq j$ we have,

$$\binom{n-1}{k} p^k (1-p)^{n-1-k} \quad = \quad O(\log(n)/n)$$

Now we can bound the probability that two vertices have degree Δ and are adjacent by equation (5.2) summed over $O(\log(n))$ possible values of k.

$$O(log(n)) \binom{n}{2} O(\log^2(n)/n)^2 p \quad = \quad o(1)$$

Similarly we can sum equations (5.4), (5.6) and (5.8) over the $O(\log(n))$ possible values of Δ to get, $\mathbf{Pr}(A \neq 0) = o(1)$. ■

Lemma 38 *For* $\frac{pn}{\log(n)} \to \infty$ *and* $\frac{(1-p)n}{\log(n)} \to \infty$ *a.a.s.* $\mathbf{Pr}(A \neq 0) = o(1)$.

Proof. By Corollary 26 we know that a.a.s. $G_{n,p}$ has a unique vertex of degree $\Delta(G_{n,p})$ and all other vertices have degree $\leq \Delta(G_{n,p}) - 2$, it follows that $\mathbf{Pr}(A \neq 0) = o(1)$. ■

For large values of p we will need the following lemma,

Lemma 39 *If* $p = 1 - \frac{(c+o(1))\log(n)}{n}$ *for some constant* c *with* $c \geq 1$ *a.a.s.* $G_{n,p}$ *has at least* $c\log^2(n)$ *vertex disjoint stable sets of size* 3.

Proof. Consider $G_{n,q}$ with $q = 1 - p = (c + o(1))\log(n)/n$ we wish to show that a.a.s. there exists $c\log(n)$ vertex disjoint triangles in $G_{n,q}$. Label all possible triples of vertices in $G_{n,q}$ from t_1 to $t_{\binom{n}{3}}$. Let Z_i be an indicator random variable such that $Z_i = 1$ if all three edges appear between the triple t_i, and 0 otherwise. Let Z be the random variable counting the total number of triangles in $G_{n,q}$, ie. $Z = \sum_{i=1}^{\binom{n}{3}} Z_i$. We get that,

$$\mathbf{Exp}[Z] = \binom{n}{3} q^3 \geq \frac{(c + o(1))^3 \log^3(n)}{6}$$

Similarly, let Y_i be an indicator random variable such that if there exists $j \neq i$ whose $t_i \cap t_j \neq \emptyset$, such that $Z_i = 1$ and $Z_j = 1$ then $Y_i = 1$, and

$Y_i = 0$ otherwise. Also let $Y = \sum_{i=1}^{\binom{n}{3}} Y_i$ be the random variable counting the number of triangles in $G_{n,q}$ that intersect at least one other triangle.

$$\mathbf{Exp}[Y] = \binom{n}{3} q^3 (\binom{n-3}{2} 3q^3 + (n-3)3q^2) \quad (5.10)$$

$$= o(1) \quad (5.11)$$

If Z' is the number of triangles that aren't intersected by any other triangles in $G_{n,q}$, we get that $Z' = Z - Y$ and by linearity of expectation that $\mathbf{Exp}[Z'] = \mathbf{Exp}[Z] - \mathbf{Exp}[Y]$, so

$$\mathbf{Exp}[Z'] = \binom{n}{3} q^3 (1 - (\binom{n-3}{2} 3q^3 + (n-3)3q^2)) \quad (5.12)$$

$$= \Omega(\log^2 n) \quad (5.13)$$

Now we just have to show that with high probability Z' is concentrated about its mean.

First we will show that Z is concentrated around its mean. By Chebychev's inequality we have that,

$$\begin{aligned} \mathbf{Pr}(|Z - \mathbf{Exp}[Z]| \geq \log^2(n)) &\leq \frac{1}{\log^4(n)} \mathbf{Exp}[(Z - \mathbf{Exp}[Z])^2] \\ &= \frac{1}{\log^4(n)} (\mathbf{Exp}[Z^2] - \mathbf{Exp}^2[Z]) \quad (5.14) \end{aligned}$$

Let A be the set of all pairs of 3-tuples of vertices such that the pairs don't share any vertices. Let B be the set of all pairs of 3-tuples of vertices such that the two 3-tuples share exactly one vertex. Let C be the set of all pairs of 3-tuples such that the 3-tuples share exactly two vertices. Let D be the pair of 3-tuples such that the 3-tuples are identical.

$$\begin{aligned} \mathbf{Exp}[Z^2] &= \sum_{\{Z_i, Z_i\} \in D} \mathbf{Exp}[Z_i Z_i] + \sum_{\{Z_i, Z_j\} \in A} \mathbf{Exp}[Z_i Z_j] \\ &+ \sum_{\{Z_i, Z_j\} \in B} \mathbf{Exp}[Z_i Z_j] + \sum_{\{Z_i, Z_j\} \in C} \mathbf{Exp}[Z_i Z_j] \end{aligned}$$

$$
\begin{aligned}
&= \binom{n}{3}q^3 + \binom{n}{3}\binom{n-3}{3}q^6 + 3\binom{n}{3}\binom{n-3}{2}q^6 \\
&\quad +3\binom{n}{3}(n-3)q^5 \qquad (5.15)
\end{aligned}
$$

Substituting (5.15) into (5.14) and using $\mathbf{Exp}^2[Z] = (\binom{n}{3}q^3)^2$ we get that,

$$
\begin{aligned}
\mathbf{Pr}(|Z - \mathbf{Exp}[Z]| \geq \log^2(n)) &\leq \frac{1}{\log^4(n)}\binom{n}{3}q^3(1 + (\binom{n-3}{3} - \binom{n}{3})q^3 \\
&\quad +3\binom{n-3}{2}q^3 + 3(n-3)q^2) \\
&= o(1) \qquad (5.16)
\end{aligned}
$$

Since Y is integer valued,

$$
\begin{aligned}
\mathbf{Pr}(Y > 0) &\leq \mathbf{Exp}[Y] \\
&= o(1) \qquad (5.17)
\end{aligned}
$$

By equation (5.16) we know that a.a.s. $Z \geq \frac{(c+o(1))^3 \log^3(n)}{6} - \log^2(n)$, and by (5.17) a.a.s. $Y = 0$. Since $Z' = Z - Y$, it follows that $\mathbf{Pr}(Z' < c\log^2(n)) = o(1)$. This implies that a.a.s. $G_{n,p}$ has at least $c\log^2(n)$ vertex independent stable sets of size three if $p = 1 - \frac{(c+o(1))\log(n)}{n}$. ■

Lemma 40 *If $p = 1 - \frac{(c+o(1))\log(n)}{n}$ where c is a constant then a.a.s.*
$|E(G_{n,p})| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n.$

Proof. We know $\mathbf{Exp}[|E(G)|] = \binom{n}{2}p = \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2}$. Using Chebychev's inequality we get,

$$
\begin{aligned}
\mathbf{Pr}(||E(G)| - \mathbf{Exp}[|E(G)|]| \geq \epsilon) &\leq \frac{1}{\epsilon^2}(\mathbf{Exp}[|E(G)|^2] - \mathbf{Exp}^2[|E(G)|]) \\
&= \frac{1}{\epsilon^2}(\binom{n}{2}p + \binom{n}{2}(\binom{n}{2} - 1)p^2 - p^2\binom{n}{2}^2) \\
&= \frac{\binom{n}{2}p(1-p)}{\epsilon^2}
\end{aligned}
$$

If we take ϵ to be n we get that a.a.s. $|E(G)| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$. ■

5.4 Computing $\mathcal{X}_f^T(G_{n,p})$

Theorem 41 *We can a.a.s. compute $\mathcal{X}_f^T(G_{n,p})$ in $O(n^3)$ time.*

The following lemmas deal with small values of p and go towards proving Theorem 41.

Lemma 42 *If $p = o(n^{-2})$ a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 1$.*

Proof. By Lemma 27 we know that a.a.s. $G_{n,p}$ doesn't have any edges, $\Delta(G_{n,p}) = 0$ and all the vertices of $G_{n,p}$ form a stable set. We can check if $G_{n,p}$ is in fact a stable set in $O(n^2)$ time. ■

Lemma 43 *If $p = (c + o(1))n^{-2}$ then the probability that $G_{n,p}$ has a $\Delta(G_{n,p}) + 1$ fractional total colouring is bounded away from 0 and 1. But we can a.a.s. compute $\mathcal{X}_f^T(G_{n,p})$ in polynomial time.*

Proof. By Lemma 28 we know that a.a.s. every component of $G_{n,p}$ contains at most one edge. If an edge exists then $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 2$ and if no edge exists $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 1$. But, the probability that at least one edge exists is bounded away from 0 and 1. We can check if an edge exists in $O(n^2)$ time. ■

Lemma 44 *If $p = \omega(n^{-2})$ and $p = o(n^{-3/2})$ then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 2$.*

Proof. By Lemma 29 we know that a.a.s. $\Delta(G_{n,p}) = 1$. This implies that a.a.s. $\mathcal{X}^T(G_{n,p}) = \mathcal{X}_f^T(G_{n,p}) = 3$ by Lemma 14. Checking that one edge exists and $\Delta(G_{n,p}) = 1$ takes $O(n^2)$ time. ■

Lemma 45 *If $p = (c + o(1))n^{-3/2}$ then the probability $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 1$ is bounded away from 0 and 1. However we can a.a.s. compute $\mathcal{X}_f^T(G_{n,p})$ in polynomial time.*

Proof. By Lemmas 29 and 30 we know that a.a.s. each component has at most two edges, and at least one edge exists, so it follows that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = 3$ by Lemma 14. But, Lemma 30 implies that the probability

$\Delta(G_{n,p}) = 2$ is bounded away from 0 and 1. Again, it's trivial to get a total three colouring of $G_{n,p}$ is this case. Checking if $\Delta(G) \geq 2$ takes $O(n^2)$ time.

■

Lemma 46 *If $p = \Omega(n^{-3/2})$ and $p = o(n^{-1})$ then a.a.s. $G_{n,p}$ has a fractional $\Delta + 1$ total colouring.*

Proof. By Lemma 31 we know that a.a.s. every component of $G_{n,p}$ is a tree and $\Delta(G_{n,p}) \geq 2$. So by Lemma 14 a.a.s. $G_{n,p}$ has a fractional $\Delta + 1$ total colouring. We can check that $G_{n,p}$ is a tree with a depth first search in $O(n)$ time. ■

Lemma 47 *If $p = \Omega(n^{-1})$ and $p < (1 - \epsilon)/n$ for some $\epsilon > 0$ then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$.*

Proof. By Lemma 32 we know that a.a.s. only unicyclic components occur, and $\Delta(G_{n,p}) \geq 3$. By Lemma 14 it follows that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$. We can check if $G_{n,p}$ is in fact unicyclic with a depth first search in $O(n)$ time.

■

The following lemma deals with p not too close to zero or one.

Lemma 48 *If $1/n \leq p \leq (1 - d(n)/n)$ where $\frac{d(n)}{\log(n)} = \infty$ then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$.*

Proof. By Lemma 36, Lemma 37, and Lemma 38 we know that $\mathbf{Pr}(A \neq 0) = o(1)$ it follows from Lemma 13 that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$. We can easily check for paths of length at most three between vertices of maximum degree and edges between vertices of degree at least $\Delta - 1$ in $O(n^3)$ time. ■

What follows deals with p close to one.

Lemma 49 *If $p = 1 - \frac{(c+o(1))\log(n)}{n}$ for constant c and $1 < c < \infty$ then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$.*

Proof. If $p = 1 - \frac{(c+o(1))\log(n)}{n}$ and $c > 1$ then the requirements of Lemma 18 or Lemma 19 are a.a.s. satisfied. Since, if we let $a = n - \Delta(G_{n,p})$ we know by

Lemma 35 that a.a.s. $\Delta(G_{n,p}) \geq n-(c-1)\log(n)$ and $a \leq (c-1)\log(n)$. It now follows by Lemma 39 that a.a.s. $G_{n,p}$ has at least a vertex independent stable sets of size 3 (clearly then it a.a.s. has a vertex independent stable sets of size 2 if n is even). Lemma 34 states that a.a.s. $\Delta(\bar{G}_{n,p}) \leq 3ec\log(n)$ this implies that a.a.s. $\Delta_{min}(G_{n,p}) \geq n-1-3ec\log(n)$. Finally by Lemma 40, a.a.s. $|E(G_{n,p})| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$. So if n is even, the requirements of Lemma 18 are a.a.s. satisfied and if n is odd the requirements of Lemma 19 are a.a.s. satisfied. We can conclude that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p}) + 1$.

■

Lemma 50 *If $p = 1 - \frac{(c+o(1))\log(n)}{n}$ for constant c and $\frac{1}{3} \leq c \leq 1$ then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$.*

Proof. If $p = 1 - \frac{(c+o(1))\log(n)}{n}$ and $\frac{1}{3} \leq c \leq 1$ then a.a.s. $|E(G_{n,p})| \leq \binom{n}{2} - \frac{(n-1)(c+o(1))\log(n)}{2} + n$ by Lemma 40. We also know that a.a.s. $\Delta(G_{n,p}) \geq n-2$ since $\lim \mathbf{Exp}[X_{n-2}] \rightarrow \infty$.

If $\Delta = n-1$ and n is even it follows by Lemma 17 that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$, since a.a.s. $\bar{G}_{n,p}$ has more than $n/2$ edges. If $\Delta = n-1$ and n is odd, then a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta + 1$ trivially, since $\mathcal{X}_f^T(K_n) = \Delta + 1$ when n is odd.

If $\Delta(G_{n,p}) = n-2$ the conditions of Lemma 20 are a.a.s. satisfied, since a.a.s. $\Delta_{min}(G_{n,p}) \geq n-1-3ec\log(n)$ by Lemma 34. Therefore, a.a.s. $\mathcal{X}(G_{n,p}) = \Delta + 1$. ■

Lemma 51 *If $1 - \frac{\log(n)}{3n} \leq p < 1$ then we can a.a.s. compute $\mathcal{X}_f^T(G_{n,p})$ in $O(n^3)$ time.*

Proof. If $p \geq 1 - log(n)/3n$ then for $k = n-1$,

$$\begin{aligned} \lim \mathbf{Exp}[X_{n-1}] &= \lim np^{n-1} \\ &\rightarrow \infty \end{aligned}$$

By Theorem 23 we know that $G_{n,p}$ a.a.s. has a vertex of degree $n-1$.

If n is odd K_n has a total colouring of size n, it follows that a.a.s. $G_{n,p}$ has a $\Delta+1$ total colouring.

If n is even let $n' = n/2$ and consider $G_{2n',p}$ (to be consistent with the notation of Lemma 17). To compute $\mathcal{X}_f^T(G_{2n',p})$, we just need to compute an optimal fractional matching of the complement of $G_{2n',p}$. Let e be the number of edges missing in $G_{2n',p}$ and j be size of a maximum fractional matching of the complement. By Lemma 17 if $e+j \geq n$, a.a.s. $\mathcal{X}_f^T(G_{2n',p}) = \Delta+1$, otherwise a.a.s. $\mathcal{X}_f^T(G_{2n',p}) = 2n'+\epsilon$, where $\epsilon = 1-\frac{e+j}{n'}$. Lemma 17 also states that we can compute j in $O(n^3)$ time.

■

Lemma 52 *For $p = 1-o(n^{-2})$ if n is odd then a.a.s. $\mathcal{X}_f^T = \Delta(G)+1$, if n is even then a.a.s. $\mathcal{X}_f^T = \Delta(G)+2$.*

Proof. By symmetry and Lemma 27 we know that a.a.s. $G_{n,p}$ contains all possible $\binom{n}{2}$ edges. If n is even, by Lemma 16 we know that a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p})+2$. If n is odd it's well known that K_n has a $\Delta(K_n)+1$ total colouring. So a.a.s. $\mathcal{X}_f^T(G_{n,p}) = \Delta(G_{n,p})+1$. We can trivially check if all possible edges are present in $O(n^2)$ time. ■

Proof of Theorem 41. This follows from Lemmas 42 through 52. ■

5.5 Finding a Fractional Total Colouring of $G_{n,p}$

In the previous section we showed how we could a.a.s. determine $\mathcal{X}_f^T(G_{n,p})$ in $O(n^3)$ time. We can actually a.a.s. determine an optimal fractional total colouring of $G_{n,p}$ in polynomial time as opposed to just the value of $\mathcal{X}_f^T(G_{n,p})$.

For $p \leq \frac{(1-\epsilon)}{n}$ a.s.s. $G_{n,p}$ is unicyclic and by the algorithmic proof of Lemma 14 we can a.a.s. compute a fractional total colouring in $O(n^2)$ time.

For $1/n \leq p \leq (1-d(n))/n$ where $\frac{d(n)}{\log(n)} = \infty$ we know that $\mathbf{Pr}(A \neq 0) = o(1)$. So the algorithmic proof of Lemma 13 a.a.s. gives us our optimal fractional total colouring in $O(n^5)$ time.

For $p = 1 - \frac{(c+o(1))\log(n)}{n}$ where c is a constant greater and $c > 1$, we know the conditions of Lemma 18 or 19 are a.a.s. satisfied. Moreover, since we a.a.s. have at least $c\log^2(n)$ vertex independent stable sets of size 3 (by Lemma 39), and we want $n - \Delta \leq (c-1)\log(n)$ of them, we can do this in a greedy manner. Choosing our sets in this fashion is guaranteed to be within 1/3 of optimal. Since each independent set of size 3 eliminates at most 3 other possible better choices. We can always a.a.s. get enough stable sets of size 3 because $\frac{c}{3}\log^2(n) > (c-1)\log(n) \geq n - \Delta$ and this greedy approach takes $O(n^2)$ time. Therefore, we can a.a.s. compute a fractional total colouring in $O(n^4 + FEC)$ by Lemmas 18 and 19.

For $p = 1 - \frac{c+o(1))\log(n)}{n}$ where c is a constant and $\frac{1}{3} \leq c \leq 1$, we know from Lemma 50 that a.a.s. $\Delta(G_{n,p}) \geq n-2$. If $\Delta = n-1$ we can use Lemma 17 to a.a.s. get a $\Delta + 1$ fractional total colouring in $O(n^4 + FEC)$ time. If $\Delta = n-2$ we can use Lemma 20 to a.a.s. get a $\Delta + 1$ fractional total colouring in $O(n^3 + FEC)$ time.

For $1 - \frac{\log(n)}{3n} \leq p \leq 1$, we know a.a.s. that $\Delta(G_{n,p}) = n-1$. Then if n is odd $G_{n,p}$ has a trivial $\Delta + 1$ total colouring that is well known. For n even we can use the second part of the proof of Lemma 17 to compute an optimal fractional total colouring in $O(n^4 + FEC)$ time.

CHAPTER 6
Conclusion

It is still unknown whether the problem of fractional total colouring is polynomial time solvable. It shares many properties with fractional edge colouring and fractional vertex colouring. It's easy to approximate within 1, like fractional edge colouring. But finding a maximum size total stable set is NP-hard, in that way it's similar to fractional vertex colouring (finding a maximum size stable set is NP-hard). While no attempt was made to optimize the algorithms presented, with a minimal amount of effort one could implement them to run quickly in practise. It is the authour's belief that fractional total colouring is probably NP-hard, but as we have seen efficient algorithms exists for most graphs.

We also would like to present a fractional version of the conformability conjecture (Conjecture 11).

Definition 2 *A graph G is fractionally conformable if it has a $\Delta(G) + 1$ fractional vertex colouring, such that the sum of weights of stable sets having parity different from $|V(G)|$ is at most the deficiency of G.*

Conjecture 53 (Fractional Conformability Conjecture) *A simple graph G has $\mathcal{X}_f^T(G) > \Delta + 1$ if and only if G contains a subgraph H with $\Delta(G) = \Delta(H)$ which is not fractionally conformable.*

Note, Lemma 17 proves this conjecture to be true for G with $\Delta(G) = |V(G)| - 1$.

References

[1] C.J.H. McDiarmid A.M. Frieze, B. Jackson and B. Reed. Edge-colouring random graphs. *Journal of Combinatorial Theory B*, 45:135–149, 1988.

[2] M.L. Balinski. Integer programming: Methods, uses, computation. *Management Science Series A*, 12:253–313, 1965.

[3] M. Behzad. Graphs and their chromatic numbers. *Ph.D. Thesis, Michigan State University*, 1965.

[4] D. Bertsimas and J.N. Tsitsiklis. Introduction to linear optimization. 1997.

[5] B. Bollobás. The chromatic number of random graphs. *Combinatorica*, 8:49–55, 1988.

[6] B. Bollobás. Random graphs, second edition. 2001.

[7] J.-M. Bourjolly and W.R. Pulleyblank. König-egerváry graphs, 2-bicritical graphs and fractional matchings. *Discrete Applied Mathematics*, 24:63–82, 1989.

[8] A.J. Chetwynd and A.J.W. Hilton. Starmultigraphs with three vertices of maximum degree. *Math. Proc. Cambridge Philos. Soc.*, 100:303–317, 1986.

[9] V. Chvátal. On certain polytopes associated with graphs. *Journal of Combinatorial Theory B*, 18:138–154, 1975.

[10] J. Edmonds. Paths, trees and flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965.

[11] P. Erdös and R.J. Wilson. On the chromatic index of almost all graphs. *Journal of Combinatorial Theory B*, 23:255–257, 1977.

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

[13] J.-C. Fournier. Colorations des arêtes d'un graphe. *Cahiers Centre Études Rech. Opér*, 15:311–314, 1973.

[14] F. Gavril and M. Yannakakis. Edge dominating sets in graphs. *SIAM J. APPL. MATH.*, 38:364–372, 1980.

[15] A.J.W. Hilton G.M. Hamilton and H.R. Hind. Updating the conformability conjecture. *Congr. Numer.*, 113:153–169, 1996.

[16] A.J.W. Hilton. A total-chromatic number analogue of plantholt's theorem. *Discrete Mathematics*, 79:169–175, 1989.

[17] I. Holyer. The np-completeness of edge-colouring. *SIAM Journal on Computing*, 10:718–720, 1981.

[18] R.M. Karp and C.H. Papadimitriou. On linear characterizations of combinatorial optimization problems. *SIAM Journal on Computing*, 11:620–632, 1982.

[19] A.V Karzanov. Determining the maximal flow in a network by the method of preflows. *Soviet Mathematics Doklady*, 15:434–437, 1974.

[20] L.G. Khachiyan. A polynomial algorithm in linear programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[21] K. Kilakos and B. Reed. Fractionally colouring total graphs. *Combinatorica*, 13 (4):435–440, 1993.

[22] P.G.H. Lehot. An optimal algorithm to detect a line graph and output its root graph. *J. Assoc. Comput. Mach.*, 21:569–575, 1974.

[23] L. Lovász. Normal hypergraphs and the perfect graph conjecture. *Discrete Math.*, 2:253–267, 1972.

[24] L. Lovász. On the ratio of optimal integral and fractional covers. *Discrete Mathematics*, 13:383–390, 1975.

[25] L. Lovász. On the shannon capacity of a graph. *IEEE Transactions on Information Theory*, 25:1–7, 1979.

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

[27] P. Seymour M. Chudnovsky, N. Robertson and R. Thomas. The strong perfect graph theorem. *Submitted for publication*, 2002.

[28] A. Schrijver M. Grötschel, L. Lovász. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.

[29] A. Schrijver M. Grötschel, L. Lovász. Polynomial algorithms for perfect graphs. *Topics on perfect graphs, North-Holland Math. Stud.*, 88:325–356, 1984.

[30] A. Schrijver M. Grötschel, L. Lovász. Geometric algorithms and combinatorial optimization. 1988.

[31] D. Matula. Expose-and-merge explration and the chromatic number of a random graph. *Combinatorics*, 7:275–284, 1987.

[32] C. McDiarmid and B. Reed. On total colourings of graphs. *Journal of Combinatorial Theory B*, 57:122–130, 1993.

[33] C.J.H. McDiarmid and A. Sánchez-Arroyo. Total colouring regular bipartite graphs is np-hard. *Discrete Mathematics*, 124:155–162, 1994.

[34] G.J. Minty. On maximal independent sets of vertices in claw free graphs. *Journal of Combinatorial Theory B*, 28:284–304, 1980.

[35] M. Molloy and B. Reed. A bound on the total chromatic number. *Combinatorica*, 18,(2):241–280, 1998.

[36] D. Nakamura and A. Tamura. A revision of minty's algorithm for finding a maximum weight stable set of a claw-free graph. *J. Oper. Res. Soc. Japan*, 44:194–204, 2001.

[37] M. W. Padberg and M. R. Rao. Odd minimum cut-sets and b-matchings. *Mathematics of Operations Research*, 7:67–80, 1982.

[38] E. Palmer. Graphical evolution: An introduction to the theory of random graphs. 1985.

[39] E. R. Scheinerman and D. H. Ullman. Fractional graph theory: A rational approach to the theory of graphs. 1997.

[40] V. Vizing. On the estimate of the chromatic class of a p-graph. *Diskret. Analiz.*, 3:25–30, 1964.

[41] V. Vizing. Some unsolved problems in graph theory. *Russian Math Surveys*, 23:125–141, 1968.

[42] H. P. Yap. Total colourings of graphs. *Lecture Notes In Mathematics*, 1623, 1996.