Algorithms and Combinatorics 16

Springer
*Berlin
Heidelberg
New York
Barcelona
Budapest
Hong Kong
London
Milan
Paris
Singapore
Tokyo*

M. Habib C. McDiarmid
J. Ramirez-Alfonsin B. Reed
Editors

Probabilistic Methods for Algorithmic Discrete Mathematics

Springer

Michel Habib
LIRMM
161 rue Ada
34392 Montpellier Cedex 5
France
e-mail: habib@lirmm.fr

Colin McDiarmid
Department of Statistics
University of Oxford
1 South Parks Road
Oxford OX1 3TG
United Kingdom
e-mail: cmcd@stats.ox.ac.uk

Jorge Ramirez-Alfonsin
Equipe Combinatoire
Université Pierre et Marie Curie
Paris 6
Case 189
4 place Jussieu
75252 Paris Cedex 05
France
e-mail: ramirez@ecp6.jussieu.fr

Bruce Reed
Equipe Combinatoire
Université Pierre et Marie Curie
Paris 6
Case 189
4 place Jussieu
75252 Paris Cedex 05
France
e-mail: reed@ecp6.jussieu.fr

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Probabilistic methods for algorithmic discrete mathematics / M. Habib ... - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Singapore ; Tokyo : Springer, 1998
(Algorithms and combinatorics ; 16)
ISBN 3-540-64622-1

Partially supported by [illegible] financed by the CNRS.

Mathematics Subject Classification (1991):
[illegible]

ISSN 0937-5511
ISBN 3-540-64622-1 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera-ready copy produced from the editors' output file
SPIN [illegible] - Printed on acid-free paper

Preface

Leave nothing to chance. This cliché embodies the common belief that randomness has no place in carefully planned methodologies, every step should be spelled out, each i dotted and each t crossed. In discrete mathematics at least, nothing could be further from the truth. Introducing random choices into algorithms can improve their performance. The application of probabilistic tools has led to the resolution of combinatorial problems which had resisted attack for decades. The chapters in this volume explore and celebrate this fact.

Our intention was to bring together, for the first time, accessible discussions of the disparate ways in which probabilistic ideas are enriching discrete mathematics. These discussions are aimed at mathematicians with a good combinatorial background but require only a passing acquaintance with the basic definitions in probability (e.g. expected value, conditional probability). A reader who already has a firm grasp on the area will be interested in the original research, novel syntheses, and discussions of ongoing developments scattered throughout the book.

Some of the most convincing demonstrations of the power of these techniques are randomized algorithms for estimating quantities which are hard to compute exactly. One example is the randomized algorithm of Dyer, Frieze and Kannan for estimating the volume of a polyhedron. To illustrate these techniques, we consider a simple related problem. Suppose S is some region of the unit square defined by a system of polynomial inequalities: $p_i(x, y) \geq 0$. Then the area of S is equal to the probability that a random point is in S, where the point is chosen uniformly at random from the unit square. Furthermore, we can determine if a point is in S simply by evaluating each polynomial at this point. So we can estimate the area of S by the proportion of a sufficiently large set of random points which lie in S. For this problem, choosing a random sample point was straightforward, as was using the sample to estimate the area. Estimating the volume of a polyhedron is not so simple.

The central chapter in this volume was written by Jerrum. It discusses more sophisticated techniques for generating random sample points from a probability distribution and using them to develop randomized algorithms for approximate counting. In particular, he discusses techniques for showing that random walks on certain types allow us to generate random points in the sample space efficiently. This is the theory of rapidly mixing Markov chains. Jerrum uses a toy example (colourings of the empty graph) to illustrate the basic techniques of the area. He then presents some more interesting applications of these techniques, including one which has the same flavour as the result of Dyer, Frieze and Kannan. He rounds out his survey by discussing two exciting new developments in the area, Path Coupling and Coupling From The Past.

Some of the earliest applications of random sampling and approximate counting were in percolation theory. As its name suggests, this field is concerned with flow in random media. One standard model for studying these flows is an infinite lattice with a supply of fluid at the origin where each edge allows fluid to pass with some probability p, independently of the other edges. A classical question asks, for a particular lattice L, how big must we make p in order to ensure that the probability that an infinite number of points get wet exceeds zero? Indeed, determining this critical value for the 3-dimensional cubic lattice is an important open problem in statistical physics. A crucial first step towards solving this problem is to determine how to evaluate a related polynomial, known as the partition function.

Welsh's article, which follows on from Jerrum's, discusses percolation theory, focussing in particular on three models: the Ising model, the Potts model, and the random cluster model. Much of the discussion is devoted to methods for evaluating the partition functions of these models. One intriguing fact is that these polynomials were already well known to combinatorialists under another name. Indeed they are specific instances of the so-called Tutte polynomial of graphs. This permits us to apply a combinatorial analysis to show that evaluating partition functions is hard but that Markov chain techniques can often be applied to obtain approximate solutions. This strand in Welsh's chapter runs in counterpoint to the central theme of the book.

Welsh's chapter is not the only one in which combinatorial analysis is applied to obtain results in probability theory. An interesting result in the same vein can be found in the article of Devroye. He describes how McDiarmid, building on earlier work of Devroye and Reed, uses the simple combinatorial idea of "leading sequences" to simplify and strengthen much of the central theory of branching random walks. This is however, only one of the host of results that Devroye presents. Most of his article concerns the application of a probabilistic tool, branching processes, to the analysis of a combinatorial structure, trees. The first branching process model is due to Galton and Watson, who developed it in 1874 to explain the disappearance of certain family names in England. The process begins with an initial ancestor which has a random number of children, according to some fixed distribution on the non-negative integers. Each child then independently has a random number of children according to the same distribution. The process obviously constructs a family tree and it is therefore not surprising that it has many applications in the analysis of random trees.

Devroye's article presents many extensions of the simple Galton-Watson process and considers their applications to a wide range of different types of random trees, tree-like structures, and algorithms on trees. It is the most comprehensive of the chapters in the volume and contains much that will be new even to an expert in the field.

The probabilistic analysis of combinatorial structures is not limited to the study of random trees. In the chapter of Frieze and Reed, we see how an understanding of the structure of a random object (e.g. graph, linear programming problem) permits us to develop algorithms which are usually efficient. In particular, we discuss algorithms for three difficult problems: Hamilton Cycle, Graph Isomorphism, and Edge Colouring. These algorithms run in polynomial time on the overwhelming proportion of inputs. In contrast, we shall see that certain classical branch and bound algorithms, for e.g. Knapsack, almost always take superpolynomial time.

These are just some of the topics covered in their broad survey of the probabilistic analysis of algorithms. The goal of the chapter is to carry out as much of the analysis as possible using only the simplest of tools. Indeed most of the discussion requires only the First Moment Method and the Chernoff Bound. The first of these has a one line proof and the second is a classical result which bounds the deviation from the mean of the number of heads observed in n flips of the same coin.

Of course, these two tools are not omnipotent. In particular, the Chernoff Bound applies only to sums of independent identically distributed 0-1 random variables. Often, in undertaking the probabilistic analysis of algorithms, we require extensions of this result which handle functions that depend, in a limited way, on a number of independent random variables. One such extension, the Hoeffding-Azuma Inequality, was first brought to the attention of the combinatorics community in the mid 80s and gained prominence after Bollobás used it to tie down the asymptotics of the chromatic number of a random graph. Recently, Talagrand introduced an exciting new method for bounding deviations (from the median) which seems to be even more widely applicable.

In his chapter, Concentration, McDiarmid provides a thorough overview of these related concentration inequalities and a number of others. He discusses a variety of applications, including Bollobás' result mentioned above. He also derives these concentration inequalities, sometimes obtaining sharper results than those known previously. Although these results are of a more technical nature than most of the other results in this volume, the author has ensured his treatment is accessible to non-experts. A careful reading of this paper will be well rewarded.

The tools presented in McDiarmid's chapter have applications outside of the probabilistic analysis of algorithms, as we shall see in the very first chapter of the book. One of the topics discussed there is sum-free sets, i.e. sets of positive integers no two subsets of which sum to the same value. One can obtain bounds on the maximum cardinality of a sum free subset A of $\{1,\ldots,n\}$ using the fact that the sum of the elements of a random subset is highly concentrated around its expected value. This is an example of the probabilistic method, which is the subject of that chapter. The probabilistic method consists of proving the existence or non-existence of a combinatorial object with particular properties (a sum-free subset of k elements of $\{1,\ldots,n\}$) via a probabilistic analysis.

Molloy begins his chapter by introducing some of the basic tools needed in such an analysis. He then discusses a number of recent results about graph colouring obtained by a joint application of various concentration bounds and a very powerful probabilistic tool, the Lovász Local Lemma. This lemma permits one to prove the existence of structures with certain global properties via a local analysis. For example, one can prove the existence of colourings of certain kinds by examining each neighbourhood separately. To see the advantages of this approach, consider the following result obtained by this method: if the maximum degree of G, Δ, is sufficiently large and G has no Δ-clique then it has a $\Delta-1$ colouring. Clearly the existence of a $\Delta-1$ colouring of a neighbourhood (which has at most $\Delta+1$ vertices) is easy to demonstrate. The fact that many problems are easier to resolve locally than globally, is what gives the Local Lemma its power. Further, as Molloy discusses, not only does the lemma prove the existence of the desired colourings, it may also yield efficient randomized algorithms for constructing them.

As we have seen, many of the chapters in this volume discuss randomized algorithms. Raghavan's chapter is devoted to the topic. Informally, a randomized algorithm is one whose behaviour is influenced by a number of random coin flips. The expected running time of the algorithm on a given input is the average over all possible sequences of coin flips. Its expected running time on inputs of size n is the maximum of its expected running time over all inputs of size n. There are many problems for which the expected running time of some randomized algorithm is better than the running time of any possible deterministic algorithm. Raghavan presents such examples. He also discusses a duality result which links the running times of randomized algorithms for a problem with the expected running times of deterministic algorithms over random inputs, thereby linking his chapter to that of Frieze and Reed. The bulk of Raghavan's chapter is devoted to a discussion of randomized algorithms for electronic fingerprinting. This area is of particular importance due to the current developments in electronic communication. It seems appropriate to end our brief introduction with the demonstration that the field discussed here is evolving in step with the world around it (possibly!).

Table of Contents

The Probabilistic Method

Probabilistic Analysis of Algorithms

An Overview of Randomised Algorithms

Mathematical Foundations of the Markov Chain Monte Carlo Method

List of Contributors

Luc Devroye
School of Computer Science,
McGill University,
Montreal, H3A 2K6, Canada

Alan M. Frieze
Mathematical Sciences Department
Carnegie-Mellon University,
Pittsburgh, PA 15213, USA

Mark Jerrum
Department of Computer Science,
University of Edinburgh,
The King's Buildings
Edinburgh EH9 3JZ, UK

Colin McDiarmid
Statistics Department,
University of Oxford,
1 South Parks Road,
Oxford OX1 3TG, UK

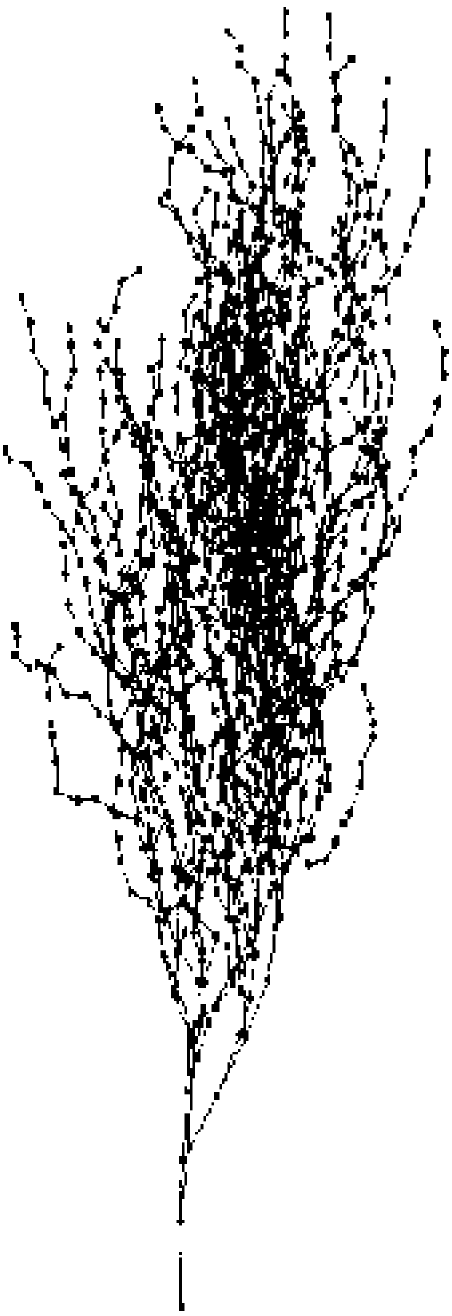
Michael Molloy
Department of Computer Science,
University of Toronto
Toronto, ON M5S 3G4, Canada

Rajeev Motwani
Department of Computer Science,
Stanford University,
CA 94305, USA

Prabhakar Raghavan
IBM Almaden Research Center,
650 Harry Road,
San Jose, CA 95120, USA

Bruce Reed
Equipe Combinatoire
Université Pierre et Marie Curie,
4 Place Jussieu
75252 Paris Cedex 5, France

Dominic Welsh
Mathematical Institute,
University of Oxford,
24-29 St. Giles',
Oxford OX1 3LB, UK



A Weyl sequence for θ is given by $\{\theta\}, \{2\theta\}, \{3\theta\}, \ldots$, where $\theta \in (0,1)$ is an irrational number, and $\{\cdot\}$ denotes "mod 1". Weyl showed that for all irrational θ the sequence is equidistributed[1]. A Weyl tree $T_n(\theta)$ is the binary search tree based upon the first n numbers in the Weyl sequence for θ[2]. Each datum is associated with a node of $T_n(\theta)$, and each node has the search tree property that is, all nodes in its left subtree have smaller values, and all nodes in its right subtree have larger values. $T_n(\pi)$ is presented on the front cover with height 36 where the branches are drawn according to the following predetermined properties. Firstly the branches are randomly rotated with respect to their parent branches. Secondly they are forced to be oriented towards the earth, facing the sun and finally the branches are assigned random lengths. This was done by a postscript program written by Luc Devroye.

[1] A sequence $x_1, x_2, \ldots$ is equidistributed if for all $0 \le a \le b \le 1$,
$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} 1_{[a \le x_i \le b]} = b - a.$$

[2] Weyl trees are a fundamental tool for the analysis of algorithms involving Weyl sequences as the input stream.

The Probabilistic Method

Michael Molloy

Department of Computer Science
University of Toronto
Toronto, Canada

Erdős is usually credited as being the pioneer of the probabilistic method, beginning with his seminal 1947 paper [21], although the probabilistic method had been used in at least two previous occasions by Turán in 1934 [64] and by Szele in 1943[58]. By now, it is widely recognized as one of the most important techniques in the field of combinatorics. In this short survey we will introduce a few of the basic tools and describe some of the areas in which the method has had impact.

The basic idea behind the probabilistic method is that in order to prove the existence of a combinatorial object satisfying certain properties (e.g. a graph with neither a large clique nor a large stable set, or a proper colouring of the vertices of a graph) we choose our object at random and prove that with positive probability it satisfies the desired properties. The two most fundamental tools used to show that this probability is positive are the First Moment Method and the Lovász Local Lemma. In order to apply these, we often need a few extra tools, most notably concentration bounds.

A common misconception regarding the probabilistic method is that one requires a deep knowledge of probability to use it. This is far from the truth – in fact, a very elementary knowledge of probability along with a familiarity with a handful of tools and some clever combinatorial reasoning will suffice. Thus, we do not assume that the readers have a strong background in probability, but we do assume that they are familiar with the basics, such as expected values. We also assume that the reader has a basic understanding of graph theory. We usually omit round-up and round-down signs when there is no chance of confusion. As is common with the probabilistic method, we rarely provide the best constant terms in our proofs, opting rather to present a simpler proof. The reader may often find it instructive to try to modify the proofs to obtain a stronger result.

1. The First Moment Method

The first tool that we will see is the First Moment[1] Method which is the most fundamental tool of the probabilistic method. The essence of the First Moment Method lies in these two simple and surprisingly powerful statements:

The First Moment Principle *If $\mathbf{E}(X) \le t$ then $\mathbf{Pr}(X \le t) > 0$.*

Proof. Intuitively, the expected value of X can be viewed as the average value of X over all possible outcomes of the random experiment. If every outcome is greater than t, then this average must be greater than t.

More formally, since $\mathbf{E}(X) = \sum_i i \times \mathbf{Pr}(X = i)$, then if $\mathbf{Pr}(X \le t) = 0$ we have $\mathbf{E}(X) = \sum_{i>t} i \times \mathbf{Pr}(X = i) > t \times \sum_{i>t} \mathbf{Pr}(X = i) = t$. □

Markov's Inequality *For any non-negative random variable X,*

$$\mathbf{Pr}(X \ge t) \le \frac{\mathbf{E}(X)}{t}.$$

Proof. Again using $\mathbf{E}(X) = \sum_i i \times \mathbf{Pr}(X = i)$, we have that since X is always non-negative, $\mathbf{E}(X) \ge \sum_{i \ge t} i \times \mathbf{Pr}(X = i) \ge t \times \mathbf{Pr}(X \ge t)$. □

Applying the First Moment Method requires a judicious choice of the random variable X, along with a (usually straightforward) expected value computation. Most often X is non-negative integer-valued and $\mathbf{E}(X)$ is shown to be less than 1, thus proving that $\mathbf{Pr}(X = 0)$ is positive. Markov's Inequality is frequently used when X is non-negative integer-valued and $\mathbf{E}(X)$ is less than 1, in which case we have $\mathbf{Pr}(X > 0) = \mathbf{Pr}(X \ge 1) \le \mathbf{E}(X)$.

Recalling that $\mathbf{E}(X) = \sum_i i \times \mathbf{Pr}(X = i)$, it may seem at first glance that you cannot compute $\mathbf{E}(X)$ without first computing $\mathbf{Pr}(X = i)$ for every value of i, which is in itself at least as difficult a task as computing $\mathbf{Pr}(X \le t)$ directly. The following fact allows us to compute $\mathbf{E}(X)$ without computing $\mathbf{Pr}(X = i)$ for any value of i, in effect by computing a different sum which has the same result.

Linearity of Expectation

$$\mathbf{E}(X_1 + \cdots + X_t) = \mathbf{E}(X_1) + \cdots + \mathbf{E}(X_t).$$

Proof. For any outcome ω of our random experiment, we denote by $X_i(\omega)$ the corresponding value of X_i. For this proof it is more convenient to express the expected value of X_i as $\sum_\omega \mathbf{Pr}(\omega) \times X_i(\omega)$. Linearity of Expectation follows immediately from this formulation as

$$\sum_\omega \mathbf{Pr}(\omega) \times (X_1(\omega) + \cdots + X_t(\omega)) = \sum_{i=1}^{t} \left(\sum_\omega \mathbf{Pr}(\omega) \times X_i(\omega) \right).$$

□

1.1 Satisfiability Problems

We first illustrate the First Moment Method with an application to Satisfiability problems.

A *boolean variable* is a variable which can take a value of either True or False. For any boolean variable x, there are two corresponding *literals*, x and $\bar{x}$, where $\bar{x}$ means "NOT x" and has the opposite value of x. A boolean formula in *Conjunctive Normal Form (CNF)* consists of a sequence of clauses joined by "$\wedge$" (AND), where each clause consists of a set of literals joined by "$\vee$" (OR). The formula is *satisfiable* if there is some assignment of values to its variables such that the entire formula equates to True, i.e. an assignment such that every clause contains at least one literal with the value True. For positive integer k, an instance of *k-SAT* is a CNF-formula where every clause has exactly k literals.

Theorem 1.1. *Any instance of k-SAT with fewer than 2^k clauses is satisfiable.*

Note that this theorem is best possible for every k, since it is straightforward to construct an unsatisfiable instance of k-SAT by taking each of the 2^k possible clauses on a fixed set of k variables.

Proof. Consider a random truth assignment generated by setting each variable to be True with probability $\frac{1}{2}$ and False with probability $\frac{1}{2}$. (Note that each truth assignment is equally likely to be chosen.) Let X be the number of unsatisfied clauses.

We will use Linearity of Expectation to compute $\mathbf{E}(X)$. To do this, we must express X as the sum of several variables each of whose expected value is easy to compute. The standard way to do this is as follows. For each clause C_i, set $X_i = 0$ if C_i is satisfied, and $X_i = 1$ if C_i is unsatisfied. Note that $X = \sum X_i$. Furthermore, for each i the expected value of X_i is simply the probability that C_i is unsatisfied, which is 2^{-k}. Since we have $m < 2^k$ clauses,

[1] The ith moment of a random variable X is $\mathbf{E}(X^i)$, and so the first moment is simply the expected value. We will examine the second moment in the next section.

$$\mathbf{E}(X) = \sum_{i=1}^{m} \mathbf{E}(X_i) = m \times 2^{-k} < 1$$

Therefore by the First Moment Principle, with positive probability $X < 1$, i.e. with positive probability the boolean formula is satisfied, and so there must be at least one satisfying assignment. □

More generally, the same argument proves the following:

Theorem 1.2 *Consider any CNF-formula $F = C_1 \wedge C_2 \wedge \ldots \wedge C_m$. If $\sum_{i=1}^{m} 2^{-|C_i|} < 1$, then F is satisfiable.*

It is well known that Satisfiability is an NP-complete problem. However, a simple corollary to the results of this section shows that any instance of Satisfiability where every clause is big enough can be solved in polytime. This may have been first noticed by Edmonds.

Corollary 1.3 *For any $\epsilon > 0$ there is a simple polytime algorithm which will solve Satisfiability for any CNF formula on n variables such that each clause has size at least ϵn.*

Proof. If the number of clauses is less than $2^{\epsilon n}$, then by Theorem 1.2 the formula must be satisfiable. Otherwise an exhaustive search of all 2^n possible truth assignments can be carried out in a time which is polynomial in the size of the input. □

1.2 Graphs with High Girth and High Chromatic Number

One of the earliest triumphs of the probabilistic method, was Erdős' proof that there are graphs with both no short cycles and arbitrarily high chromatic number [12]:

Theorem 1.4 *For any $g, k \geq 1$ there exist graphs with no cycles of length at most g and with chromatic number greater than k.*

Erdős proved the existence of such graphs using a random construction. The fact that no one was able to produce a non-probabilistic construction of such graphs for more than 10 years [36, 51] is a testament to the power of the First Moment Method. In presenting his proof here, we simplify the calculations a little by considering only the case where $g = 3$. The proof of the general case is nearly identical, and the calculations are only slightly more involved.

Theorem 1.5 *For any $k \geq 1$ there exist triangle-free graphs with chromatic number greater than k.*

Remark. *Zykov [72] was the first to prove this special case of Theorem 1.4 (and in fact did so without relying on the probabilistic method). However, his proof technique does not generalise to the more general case of arbitrary girth.*

Proof of Theorem 1.5 Choose a random graph G on n vertices by placing each of the $\binom{n}{2}$ potential edges into $E(G)$ with probability $p = n^{-\frac{2}{3}}$ (where, of course, these $\binom{n}{2}$ random choices are made independently).

In order to prove that $\chi(G) > k$, it suffices to prove that G has no stable sets of size $\frac{n}{k}$. In fact, for a delightful and elegant reason that will soon become apparent, we will show that with high probability G does not even have any stable sets of size $\frac{n}{2k}$.

We do this with a simple expected number calculation. Let I be the number of stable sets of size $\frac{n}{2k}$. For each subset S of $\frac{n}{2k}$ vertices, we define the random variable I_S to be 1 if S is a stable set and 0 otherwise. $\mathbf{E}(I_S)$ is simply the probability that S is a stable set, which is $(1-p)^{\binom{n/2k}{2}}$. Therefore by Linearity of Expectation:

$$\begin{aligned}\mathbf{E}(I) &= \sum_S \mathbf{E}(I_S)\\ &= \binom{n}{n/2k}(1-p)^{\binom{n/2k}{2}}\\ &< 2^n \times \mathbf{E}\left(-p\frac{n^2}{9k^2}\right)\\ &= 2^n \times \mathbf{E}(-\Omega(n^{4/3}))\\ &< \frac{1}{2}\end{aligned}$$

for n sufficiently large. Therefore, by Markov's Inequality, $\mathbf{Pr}(I > 0) < \frac{1}{2}$.

Our next step should be to show that the expected number of triangles is also very small. Unfortunately, this is not true. However, as we will see, by applying a clever trick it will suffice to show that with high enough probability the number of triangles is at most $\frac{n}{2}$.

To do this, we compute the expected value of T, the number of triangles. Each of the $\binom{n}{3}$ sets of 3 vertices forms a triangle with probability p^3. Therefore, by applying Linearity of Expectation as in the previous example,

$$\begin{aligned}\mathbf{E}(T) &= \binom{n}{3}p^3\\ &< \frac{n^3}{3!}(n^{-\frac{2}{3}})^3\\ &= \frac{n}{6}\end{aligned}$$

Therefore, by Markov's Inequality, $\Pr(Y \ge \frac{n}{2}) < \frac{1}{2}$.

Since $\Pr(I \ge 1) + \Pr(Y \ge \frac{n}{2}) < 1$, the probability that $I = 0$ and $Y < \frac{n}{2}$ is positive. Therefore, there exists a graph G for which $I = 0$ and $Y < \frac{n}{2}$.

And now for the elegant trick that we promised. Choose a set of at most $\frac{n}{2}$ vertices with at least one from each triangle of G and delete them to leave the subgraph G'. Clearly G' is triangle-free, and $|G'| \ge \frac{n}{2}$. Furthermore, G' has no independent set of size $\frac{n}{2k} \le \frac{|G'|}{k}$ and so $\chi(G') > k$ as desired. □

We invite the reader to now try to generalize this argument to prove Theorem 1.4. The first step should be to determine what p should be (it will depend on g).

2 The Second Moment Method

The variance of a random variable X is defined to be:

$$\mathrm{Var}(X) = \mathbf{E}((X - \mathbf{E}(X))^2).$$

Observing that the inner $\mathbf{E}(X)$ term can be treated as a constant, some simple manipulations yield

$$\begin{aligned} \mathrm{Var}(X) &= \mathbf{E}(X^2 - 2X\mathbf{E}(X) + \mathbf{E}(X)^2) \\ &= \mathbf{E}(X^2) - 2\mathbf{E}(X)\mathbf{E}(X) + \mathbf{E}(X)^2 \\ &= \mathbf{E}(X^2) - \mathbf{E}(X)^2. \end{aligned}$$

and so the variance of X is intimately related to its second moment. The second moment method refers to applications of the following, which is the most fundamental tool regarding the variance of a variable:

Chebyschev's Inequality For any $t > 0$,

$$\Pr(|X - \mathbf{E}(X)| \ge t) \le \frac{\mathrm{Var}(X)}{t^2}$$

Proof. $|X - \mathbf{E}(X)| \ge t$ iff $(X - \mathbf{E}(X))^2 \ge t^2$. The result now follows from Markov's Inequality. □

Chebyschev's Inequality is the simplest example of a concentration inequality, which means that it is mainly used to imply that, with high probability, a random variable is "concentrated" close to its expected value. We will see a few more concentration inequalities in a later section.

We illustrate the usefulness of Chebyschev's Inequality with an example from combinatorial number theory which can be found in [14].

Consider a set $A = \{a_1, \dots, a_k\}$ of positive integers. For any $I \subseteq A$ we define $s(I)$ to be the sum of the elements of I, and we define $S(A) = \{s(I) : I \subseteq A\}$ to be the set of all such sums. We say that A has distinct sums if all such sums are distinct, i.e. if $|S(A)| = 2^k$. For example, $A_1 = \{2,3,6,10\}$ has distinct sums, since $S(A_1) = \{0,2,3,5,6,8,9,10,11,12,13,15,16,18,19,21\}$, but $A_2 = \{3,5,8,10\}$ does not have distinct sums as $3 + 10 = 5 + 8 = 13$.

In terms of n, how large can a subset of $\{1, \dots, n\}$ with distinct sums be? It is not hard to construct one of size $k = \lfloor \log_2 n \rfloor + 1$ by setting $a_i = 2^{i-1}$ for $i = 1, \dots, k$. On the other hand, a simple counting argument shows that we cannot have a set of size k much bigger than $\log_2 n$ since every sum has size at most kn and so $2^k \le kn$ which yields $k \le \log_2 n + \log_2 \log_2 n + O(1)$. Erdős asked whether we can prove that in fact we cannot have a set of size larger than $\log_2 n + O(1)$, and this appears to be a very difficult question. Here, we will see how to apply Chebyschev's Inequality to cut our range of possible sizes in half.

Theorem 2.1 *If $A \subseteq \{1, \dots, n\}$ has distinct sums then $|A| \le \log_2 n + \frac{1}{2}\log_2 \log_2 n + O(1)$.*

Proof. The main idea is this. In order to achieve a set A of size k near the upper bound yielded by $2^k \le kn$, we would require that $S(A)$ be very close to $\{0, \dots, kn\}$ and in particular that the sums are spread very evenly amongst the first kn non-negative integers. In fact, as we will see, for any set A with distinct sums most of those sums tend to be clumped together near to the middle of the range $[0, s(A)]$, which will imply that the number of such sums must be much smaller than $s(A) \le kn$, and this will improve our upper bound on k.

Our first step is to formalize what we mean by "most of the sums tend to be clumped together near the middle of the range". What we will show is that if we were to pick a sum uniformly[2] at random, then with reasonably high probability it will be close to its expected value.

Since the sums are distinct, picking a uniformly random sum X from $S(A)$ is equivalent to picking a uniformly random subset $I \subseteq A$ and then taking $X = s(I)$. To do so, we can simply flip a fair coin for each a_i to decide whether to include a_i in I. In order to compute the expected value and the variance of X, it will be convenient to express X in terms of some indicator variables, so called because each variable X_i indicates whether $a_i \in I$. That is, for each $i = 1, \dots, k$ we set $X_i = 1$ if $a_i \in I$ and $X_i = 0$ otherwise. Thus $X = \sum_{i=1}^{k} a_i X_i$. By linearity of expectation we have

$$\mathbf{E}(X) = \sum_{i=1}^{k} a_i \mathbf{E}(X_i)$$

[2] "Uniformly" means that each sum is equally likely to be chosen.

$$= \frac{1}{2}\sum_{i=1}^{k} a_i$$

and

$$\begin{aligned}
E(X^2) &= E((\sum_{i=1}^{k} a_i X_i)^2) \\
&= E(\sum_{i=1}^{k} a_i^2 X_i^2 + 2\sum_{1\le i<j\le k} a_i a_j X_i X_j) \\
&= \sum_{i=1}^{k} a_i^2 E(X_i^2) + 2\sum_{1\le i<j\le k} a_i a_j E(X_i X_j) \\
&= \frac{1}{2}\sum_{i=1}^{k} a_i^2 + \frac{1}{2}\sum_{1\le i<j\le k} a_i a_j
\end{aligned}$$

where the last line uses the easily verified fact that $E(X_i^2) = E(X_i) = \frac{1}{2}$ while $E(X_i X_j) = \frac{1}{4}$. Using our expression for $E(X)$, we can calculate

$$E(X)^2 = \frac{1}{4}\sum_{i=1}^{k} a_i^2 + \frac{1}{2}\sum_{1\le i<j\le k} a_i a_j$$

and so

$$\mathrm{var}(X) = E(X^2) - E(X)^2 = \frac{1}{4}\sum_{i=1}^{k} a_i^2.$$

Thus we have $\mathrm{var}(X) < \frac{n^2 k}{4}$. Applying Chebyschev's inequality with $t = 2\sqrt{\mathrm{var}(X)}$ we have

$$\Pr(|X - E(X)| \ge 2\sqrt{\mathrm{var}(X)}) \le \frac{1}{4}$$

and so

$$\Pr(|X - E(X)| \ge n\sqrt{k}) < \frac{1}{4}.$$

In other words, at least $\frac{3}{4}$ of the members of $S(X)$ are contained in an interval of length less than $2n\sqrt{k}$ around $E(X)$. Therefore, $\frac{3}{4}2^k \le 2n\sqrt{k}$, which yields $k \ge \log_2 n + \frac{1}{2}\log_2\log_2 n - O(1)$. □

3. The Lovász Local Lemma

3.1 The Basic Form

In this section, we introduce one of the most powerful tools of the probabilistic method: The Lovász Local Lemma. We present the Local Lemma by reconsidering Satisfiability problems.

Recall that in Section 1, we showed that any instance of k-SAT with fewer than 2^k clauses is satisfiable because the expected number of false clauses in a uniformly random truth assignment is less than 1.

Now suppose that an instance of k-SAT has many more than 2^k clauses, say 2^{2k} clauses. Obviously, the First Moment Method will fail in this case. In fact, at first glance it appears that any attempt to apply the probabilistic method by simply selecting a uniformly random truth assignment is doomed since the chances of it being a satisfying assignment would typically be very remote indeed. Fortunately, we don't require a high probability of success, just a positive probability of success.

To be more precise, we will choose a uniformly random truth assignment, and for each clause C, we denote by A_C the event that C is false. Consider the extreme case where every variable appears in only one clause. In this case, the events A_C are independent, and so setting m to be the number of clauses, the probability that none of the clauses are false is exactly $(1-2^{-k})^m$ which is positive no matter how large m is. Therefore, the formula is satisfiable. (Of course, there is a much easier way to prove this fact!)

Now for general instances of k-SAT, these events are certainly not independent as typically there are many variables which each appear in several clauses. The Lovász Local Lemma is a remarkably powerful tool which says that in such situations, as long as there is a sufficiently limited amount of dependency, we can still claim a positive probability of success.

Here we state the Lovász Local Lemma in its simplest and most common form. Before doing so we need the following definition.

An event A is *mutually independent* of a set of events $\mathcal{E}$ if conditioning on whether or not some of the events in $\mathcal{E}$ hold does not affect the probability of A. More formally, for every $B_1, \ldots, B_l, C_1, \ldots, C_r \in \mathcal{E}$,

$$\Pr(A \mid B_1 \wedge \ldots \wedge B_l \wedge \overline{C_1} \wedge \ldots \wedge \overline{C_r}) = \Pr(A).$$

The Lovász Local Lemma [24]: *Consider a set $\mathcal{E}$ of (typically bad) events such that for each $A \in \mathcal{E}$*

(a) $\Pr(A) \le p < 1$, *and*

(b) A_i is mutually independent of a set of all but at most d of the other events.

If $4pd \le 1$ then with positive probability, none of the events in $\mathcal{E}$ occur.

Our first application of the Lovász Local Lemma is the following, which is a rewording of a well-known result of Erdős and Lovász regarding hypergraph colouring.

Theorem 3.4. *If F is an instance of k-SAT such that each variable lies in at most $2^{k-2}/k$ clauses, then F is satisfiable.*

Note that there is no restriction on the number of clauses here – there can be arbitrarily many!

Proof. We will select a uniformly random truth assignment, i.e. we set each variable to be True with probability $\frac{1}{2}$ and False with probability $\frac{1}{2}$.

Recall that for each clause C, A_C is the event that C is False. We also define N_C to be the set of clauses which share a variable with C. Note that since each variable lies in at most $2^{k-2}/k$ clauses, the size of N_C is less than 2^{k-2}.

Claim 3.5. *Each event A_C is mutually independent of the set of events $\{A_{C'} : C' \notin N_C\}$.*

Our theorem follows easily from this claim and the Lovász Local Lemma as $\Pr(A_C) = 2^{-k}$ and $4 \times 2^{-k} \times 2^{k-2} = 1$.

The claim seems intuitively clear, but we should take care to prove it, as looks can often be deceiving in this field.

Suppose that the variables are ordered $x_1, \ldots, x_n$, where C contains $x_1, \ldots, x_k$. There is a standard one-to-one correspondence between the set of truth assignments and the set of n-digit binary sequences, where digit i represents the value assigned to x_i.

Consider any clauses $C_1, \ldots, C_t \notin N_C$. Let Y be the set of binary sequences corresponding to colourings for which the event $B = A_{C_1} \cap \ldots \cap A_{C_t}$ holds.

For any $(n-k)$ digit sequence α, define T_α to be the set of 2^k different n-digit binary sequences which end with α. It is straightforward to verify that for each α, Y contains either all of T_α or none of T_α. In other words, Y is the disjoint union $T_{\alpha_1} \cup \ldots \cup T_{\alpha_r}$ for some $\alpha_1, \ldots, \alpha_r$.

Within each T_{α_i}, exactly 1 of the 2^k sequences correspond to colourings in which C is False, and so $\Pr(A_C \mid B) = 2^{-k} = \Pr(A_C)$ as claimed. □

The Claim in the preceding proof is a special case of a very useful principle concerning mutual independence. In fact, we appeal to the following fact nearly every time we wish to establish mutual independence.

The Mutual Independence Principle *Suppose that $X = X_1, \ldots, X_m$ is a sequence of independent random trials. Suppose further that $A_1, \ldots, A_n$ is a set of events, where each A_i is determined by $F_i \subseteq X$. If $F_i \cap (F_{j_1} \cup \ldots \cup F_{j_k}) = \emptyset$ then A_i is mutually independent of $\{A_{j_1}, \ldots, A_{j_k}\}$.*

The proof follows along the lines of that of the preceding Claim, and we leave the details to the reader.

3.2 Disjoint Cycles

We illustrate the Local Lemma in this section by proving a simple result regarding vertex-disjoint cycles in graphs. This type of application appears in a few places, such as [7, 1]. Here we will prove a simple weakening of one such lemma from [10]:

Theorem 3.6. *Every k-regular directed graph G has a collection of $\lfloor k/3\ln k \rfloor$ vertex-disjoint directed cycles.*

Proof. We will randomly partition $V(G)$ into $r = \lfloor k/3\ln k \rfloor$ parts $V_1, \ldots, V_r$ and show that with positive probability, each part contains a cycle. To do so, we will prove that with positive probability, every vertex has an outneighbour in the same part. In other words, each V_i induces a subgraph with minimum outdegree at least 1, and it is well known (and easy to prove) that any such subgraph contains a cycle.

So for each vertex v, we place v into a randomly chosen V_i, where each part is equally likely to be chosen. We let A_v be the event that v does not have any outneighbour in the same part.

$\Pr(A_v) = (1-\frac{1}{r})^k < e^{-k/r} \le e^{-3\ln k} = k^{-3}$. By the Mutual Independence Principle, each A_v is mutually independent of the events $\{A_u : (v \cup N^+(v)) \cap (u \cup N^+(u)) = \emptyset\}$ which is all but at most $(k+1)^2$ of the events. Therefore, by the Lovász Local Lemma, with positive probability none of these events hold as long as $4k^{-3}(k+1)^2 < 1$ which is true for $k \ge 6$, while for $k < 6$ the theorem is trivial since $r = 1$. □

Using the Semirandom Method, described in a later section, Theorem 3.6 can be improved to yield a linear number of vertex-disjoint cycles, more precisely $k/2^{64}$ of them (see [10]). In related work, Bermond and Thomassen [4] conjecture that if a digraph G has minimum outdegree k, then G has $\frac{k}{2}$ vertex disjoint cycles. Thomassen [15] showed that such a digraph has r disjoint cycles so long as $k \ge (r+1)!$. Alon[3] improved this result, showing that any digraph with minimum outdegree k has $k/64$ vertex disjoint cycles. Note that this also significantly improves the constant term from the aforementioned result from [10].

3.2 More General Forms

The most general form of the Local Lemma is as follows. We omit the proof as it is available in many places such as [11, 53].

The General Local Lemma *Consider a set $\mathcal{E} = \{A_1, \dots, A_n\}$ of (typically bad) events such that each A_i is mutually independent of $\mathcal{E} - (\mathcal{D}_i \cup A_i)$, for some $\mathcal{D}_i \subseteq \mathcal{E}$. If we have reals $x_1, \dots, x_n \in [0, 1)$ such that for each $1 \le i \le n$*

$$\Pr(A_i) \le x_i \prod_{A_j \in \mathcal{D}_i} (1 - x_j)$$

then the probability that none of the events in $\mathcal{E}$ occur is at least $\prod_{i=1}^n (1 - x_i) > 0$.

Most known applications of the General Local Lemma are essentially applications of either the simple form of the Local Lemma, or one of the following two more general forms.

The Asymmetric Local Lemma *Consider a set $\mathcal{E} = \{A_1, \dots, A_n\}$ of (typically bad) events such that each A_i is mutually independent of $\mathcal{E} - (\mathcal{D}_i \cup A_i)$, for some $\mathcal{D}_i \subseteq \mathcal{E}$. If for each $1 \le i \le n$*

a) $\Pr(A_i) \le \frac{1}{4}$, and

b) $\sum_{A_j \in \mathcal{D}_i} \Pr(A_j) \le \frac{1}{4}$

then with positive probability, none of the events in $\mathcal{E}$ occur.

The Weighted Local Lemma *Consider a set $\mathcal{E} = \{A_1, \dots, A_n\}$ of (typically bad) events such that each A_i is mutually independent of $\mathcal{E} - (\mathcal{D}_i \cup A_i)$, for some $\mathcal{D}_i \subseteq \mathcal{E}$. If we have integers $t_1, \dots, t_n \ge 1$ and a real $0 \le p \le \frac{1}{4}$ such that for each $1 \le i \le n$*

a) $\Pr(A_i) \le p^{t_i}$, and

b) $\sum_{A_j \in \mathcal{D}_i} (2p)^{t_j} \le \frac{t_i}{2}$

then with positive probability, none of the events in $\mathcal{E}$ occur.

It is straightforward to verify that these both follow from the General Local Lemma. For example, to prove the Asymmetric Local Lemma, we set $x_i = 2\Pr(A_i)$ for each i. Since $\Pr(A_i) \le \frac{1}{4}$, then $x_i \le \frac{1}{2}$ and so $(1 - x_i) \ge e^{-(2\ln 2)x_i}$.

$$\begin{aligned} x_i \prod_{A_j \in \mathcal{D}_i} (1 - x_j) &\ge x_i \prod_{A_j \in \mathcal{D}_i} e^{-(2\ln 2)x_j} \\ &\ge 2\Pr(A_i) \times e^{-(4\ln 2)\sum_{A_j \in \mathcal{D}_i} \Pr(A_j)} \\ &\ge 2\Pr(A_i) \times e^{-\ln 2} \\ &> \Pr(A_i) \end{aligned}$$

A proof of the Weighted Local Lemma follows in a similar manner, after setting $x_i = (2p)^{t_i}$. Clearly, the simple form of the Local Lemma follows from the Asymmetric Local Lemma (after observing that for the simple case of the Local Lemma we can assume $d \ge 1$ and so $\Pr(A_i) \le \frac{1}{4}$ for each i). We illustrate each of these latter two forms with an application, the first to graph colouring, and the second to expander graphs.

A proper *vertex colouring* of a graph is *β-frugal*, if for each vertex v and colour c, the number of times that c appears in the neighbourhood of v is at most β. This notion was introduced in [33] and it played an important role in the bound on the total chromatic number provided in [33].

Consider any constant $\beta \ge 1$. Alon (see [33]) has shown that for each Δ, there exist graphs with maximum degree Δ for which the number of colours required for a β-frugal colouring is at least of order $\Delta^{1+\frac{1}{\beta}}$. We prove here that this is best possible as shown by Hind, Molloy and Reed [33].

Theorem 3.4. *If G has maximum degree $\Delta \ge \beta^\beta$ then G has a β-frugal proper vertex colouring using at most $16\Delta^{1+\frac{1}{\beta}}$ colours.*

Proof. For $\beta = 1$ this is easy. We are simply trying to find a proper vertex colouring of the square of G, i.e. the graph obtained from G by adding an edge between any two vertices of distance 2 in G. It is straightforward to show that this graph has maximum degree less than Δ^2 and so by Brooks' Theorem it can be properly Δ^2-coloured.

For $\beta \ge 2$ we need the Asymmetric Local Lemma. Set $C = 16\Delta^{1+\frac{1}{\beta}}$. We assign to each vertex of G a uniformly random colour from $\{1, \dots, C\}$. For each edge (u, v) we define the Type A event $A_{u,v}$ to be the event that u, v both receive the same colour. For each $\{u_1, \dots, u_{\beta+1}\}$ all in the neighbourhood of one vertex, we define the Type B event $B_{u_1, \dots, u_{\beta+1}}$ to be the event that $u_1, \dots, u_{\beta+1}$ all receive the same colour. Note that if none of these events hold, then our random procedure has successfully found a β-frugal colouring of G.

The probability of any Type A event is at most $1/C$, and the probability of any Type B event is at most $1/C^\beta$. By the Mutual Independence Principle, each event is mutually independent of all events with which it does not have any common vertices, which is all but at most $(\beta+1)\Delta$ Type A events and $(\beta+1)\Delta\binom{\Delta}{\beta}$ Type B events.

$$(\beta+1)\Delta \times \frac{1}{C} + (\beta+1)\Delta\binom{\Delta}{\beta} \times \frac{1}{C^{\beta}} < \frac{(\beta+1)\Delta}{C} + \frac{(\beta+1)\Delta^{\beta+1}}{\beta! C^{\beta}}$$
$$= \frac{\beta+1}{10\Delta^{\frac{1}{\beta}}} + \frac{\beta+1}{\beta! 10^{\beta}}$$
$$< \frac{1}{4}$$

for $\Delta \geq 6$.

The proof now follows from the Asymmetric Local Lemma. □

Remark. *It is instructive to note here that if we had tried to use the Local Lemma in its simplest form, we would have had to take $p = 1/C$ and $d = (\beta+1)\Delta\binom{\Delta}{\beta}$. Thus pd would have been much bigger than 1 for large Δ and so the Local Lemma would not have applied.*

A graph G is a β-*expander* if for any subset $S \subseteq V(G)$ with $|S| \leq \frac{1}{2}|V(G)|$, we have $|E(S,\bar{S})| \geq \beta|S|$ (and so we are discussing edge-expansion rather than vertex-expansion). Expander graphs have many important applications, for example they can form the basis of good sorting algorithms, good routing networks, and the rate at which many Markov chains converge (see Chapter 4) is intimately related to the expansion properties of underlying graphs. Many of the most important types of expander graphs are regular. Here we will show that the edges of any regular β-expander can be partitioned into E_1, E_2 such that each E_i is the edgeset of a nearly $\frac{\beta}{2}$-expander on the same vertex set, as proved by Frieze and Molloy[17], who were answering a question from [20].

Theorem 3.5. *For every $\epsilon > 0$, $r \geq 3$, and β sufficiently large in terms of r, ϵ, if G is an r-regular β-expander then there is a partition $E(G) = E_1 \cup E_2$ such that each E_i induces a $\beta(\frac{1}{2}-\epsilon)$-expander on $V(G)$.*

Proof. We leave it to the reader to verify the easy fact that if $|E_i(S,\bar{S})| \geq \beta(\frac{1}{2}-\epsilon)|S|$ holds for every connected subset[1] $S \subseteq V(G)$, $|S| \leq \frac{1}{2}|V(G)|$, then it holds for every $S \subseteq V(G)$, $|S| \leq \frac{1}{2}|V(G)|$.

We will place each edge into E_1 or E_2, each with equal probability and of course the choices for different edges being independent. For each connected subset S of size at most $\frac{1}{2}|V(G)|$, we define A_S to be the event that either $E_1(S,\bar{S}) < \beta(\frac{1}{2}-\epsilon)|S|$ or $E_2(S,\bar{S}) < \beta(\frac{1}{2}-\epsilon)|S|$.

[1] i.e. a subset of the vertices which induces a connected subgraph of G.

Since $E(S,\bar{S}) \geq \beta|S|$, the probability of A_S is at most the probability that the binomial random variable[2] $BIN(\beta|S|, \frac{1}{2})$ differs from its expected value by more than $\epsilon\beta|S|$. By using either classical results regarding $BIN(n, \frac{1}{2})$ or the Chernoff Bound presented in the next section, it is straightforward to show that this probability is less than $2e^{-\epsilon^2\beta|S|/3}$ for ϵ sufficiently small.

By the Mutual Independence Principle, each A_S is mutually independent of all events A_T such that $S \cap T = \emptyset$. It is a standard fact (see for example [6]) that since G is r-regular, every vertex lies in at most $\binom{rk}{k} < (er)^k$ connected subsets of size k for any $k \geq 1$. It follows that D_S contains at most $(er)^k|S|$ events corresponding to a subset of size k.

Therefore, setting $a = 2e^{-\epsilon^2\beta/3}$ and $t_S = |S|$ for each S, we have:

(a) $\Pr(A_S) \leq a^{t_S}$, and
(b) $\sum_{A_T \in D_S}(2a)^{t_T} \leq t_S \times \sum_{k\geq 1}(2a)^k(er)^k < \frac{t_S}{2}$

as long as $4re^{1-\epsilon^2\beta/3} < \frac{1}{3}$, which is true as long as β is sufficiently large (a little larger than $\frac{3\ln r}{\epsilon^2}$ will do). Thus, the result follows from the Weighted Local Lemma. □

Remark. *It is instructive to attempt to use the simple version of the Local Lemma and the Asymmetric Local Lemma to prove Theorem 3.5 using the same events, to see why they do not apply.*

4. Concentration

The ultimate goal of nearly every application of the probabilistic method is to show that a particular "good event" occurs with positive probability, or equivalently to show that the probability of a particular "bad event" is less than 1. However, frequently an intermediate step requires us to prove that the probability of an intermediate bad event is very small, not merely less than 1. For example, in applications of the Local Lemma, in order to show that the probability of the union of a set of bad events is less than 1, we must show that each individual bad event has very small probability.

Concentration bounds are amongst the most important tools for showing that the probability of an event is extremely small. We have already seen Markov's Inequality, which is, in a sense, a one-sided concentration bound, as it bounds the probability that X is much larger than $\mathbf{E}(X)$, and Chebyshev's Inequality which is the most basic of the true concentration bounds. The strength of these two inequalities is that they are

[2] $BIN(n,p)$ is the number of heads obtained from a sequence of n coin flips where each coin comes up heads with probability p.

widely applicable, requiring only that X is non-negative. Unfortunately they provide relatively weak bounds. For example, Markov's Inequality yields $\Pr(X \geq 2E(X)) \leq \frac{1}{2}$, and Chebychev's Inequality, while usually a little stronger, is often not nearly powerful enough. We frequently require the very strong bound $\Pr(X \geq 2E(X)) < e^{-\Theta(E(X))}$, for which we need more powerful tools.

In this section, we will briefly list a few of the most useful concentration bounds in their simplest forms.

A more detailed discussion appears in Chapter 6 of this book.

Recall that $BIN(n,p)$ is the sum of n independent variables, each equal to 1 with probability p and 0 otherwise. Our first tool, the Chernoff Bound, bounds the probability that $BIN(n,p)$ is far from its expected value.

The Chernoff Bound[3] *For any $0 \leq a \leq np$:*

$$\Pr(|BIN(n,p) - np| > a) < 2e^{-a^2/3np}.$$

For example, in the proof of Theorem 3.2, we needed to bound the probability that $BIN(|S|, \frac{1}{2})$ differs from its expected value by more than $\epsilon|S|$. By applying the Chernoff Bound with $n = |S|$, $p = \frac{1}{2}$ and $a = \epsilon|S|$, we see that this probability is at most $2e^{-2\epsilon^2|S|^2/3|S|} = 2e^{-(\frac{2\epsilon^2}{3})|S|}$, as long as $\epsilon \leq \frac{1}{2}$.

Note: For $a > np$, it is usually a good enough bound to simply note $\Pr(BIN(n,p) - np > a) \leq \Pr(|BIN(n,p) - np| > np)$ and apply the Chernoff Bound.

The shortcoming of the Chernoff Bound is that it only applies to binomial random variables. The next tool gives a similar bound on the concentration of a wider class of random variables.

Simple Concentration Bound *Let X be a random variable determined by n independent trials $T_1, \ldots, T_n$, and satisfying*

changing the outcome of any one trial can affect X by at most c, (4.1)

then

$$\Pr(|X - E(X)| > t) \leq 2e^{-\frac{t^2}{2c^2n}}.$$

Typically, we take c to be a small constant.

Clearly if $X = BIN(n,p)$ then X satisfies the conditions of this theorem with $c = 1$. Note furthermore, that in the case that p is a constant, the bound provided by the Simple Concentration Bound is almost as tight as that provided by the Chernoff Bound.

Our next two tools are the two most powerful concentration bounds widely used in the probabilistic method. They can both be regarded as refinements of the Simple Concentration Bound.

In the first of these refinements, we replace condition (4.1) by a weaker condition. In particular, instead of requiring that the amount by which the outcome of any one trial can affect X is bounded, we only require that if we carry out the the trials in sequence then the amount by which the outcome of any one trial can affect *the conditional expected value of X* is bounded. Another feature of this next inequality is that we do not require the random trials to be independent.

In the following statement, we denote by $E(X|T_1, \ldots, T_i)$ the conditional expected value of X conditional on the outcomes of $T_1, \ldots, T_i$.

The Hoeffding-Azuma Inequality [15, 34] *Let X be a random variable determined by n trials $T_1, \ldots, T_n$, and satisfying for each i,*

$$\max |E(X|T_1, T_2, \ldots, T_{i+1}) - E(X|T_1, T_2, \ldots, T_i)| \leq c_i \qquad (4.2)$$

where the maximum is taken over all possible outcomes of $T_1, \ldots, T_{i+1}$. Then

$$\Pr(|X - E(X)| > t) \leq 2e^{-t^2/(2\sum c_i^2)}.$$

It is straightforward to show that condition (4.1) implies condition (4.2), and thus to verify that The Hoeffding-Azuma Inequality implies the Simple Concentration Bound. For a more detailed discussion of The Hoeffding-Azuma Inequality see Chapter 6 of this book, or [11, 40]. Some applications of The Hoeffding-Azuma Inequality can also be found in Chapter 2 of this book. We will not discuss this inequality further here, as it is not used in the remainder of this chapter, and we only mention it here as it is widely used in the literature and to compare it to Talagrand's Inequality.

The Simple Concentration Bound and The Hoeffding-Azuma Inequality perform much more weakly than the Chernoff Bound in the case $X = BIN(n,p)$, where $p = o(1)$. More generally, when $E(X) = o(n)$ and we take each c or c_i to be a constant then, for example, we obtain that for any constant $\alpha > 0$, $\Pr(|X - E(X)| > \alpha E(X)) < e^{-\Theta(E(X)^2/n)}$, when we often require that probability to be as small as $e^{-\Theta(E(X))}$. (Sometimes, by taking c_i to be sufficiently small, we can obtain this tighter bound using The Hoeffding-Azuma Inequality, but this is usually difficult and in many cases no such proof is known.) Our next tool is the most recent of our tools, and by generalizing

[3] This is somewhat of a misnomer, as this bound is actually a common strengthening of Chernoff's original bound. For a more detailed history of this result, see Chapter 6 of this book. This bound follows easily from Theorem 2.3 (b) and (c) in that chapter.

the Simple Concentration Bound in a different direction, allowing us to replace r by $\mathbf{E}(X)$ in the bound, thus overcoming this problem.

Talagrand's Inequality I [84] *Let X be a random variable determined by n independent trials $T_1, \ldots, T_n$, and satisfying*

1. *changing the outcome of any one trial can affect X by at most c, and*
2. *for any s, if $X \geq s$ then there are s trials $T_{i_1}, \ldots, T_{i_s}$ whose outcomes certify that $X \geq s$,*

then for any $0 \leq t \leq \mathbf{Med}(X)$,

$$\mathbf{Pr}(|X - \mathbf{Med}(X)| > t) \leq 4e^{-\frac{t^2}{8c^2\mathbf{Med}(X)}}.$$

More precisely, condition 2 says that changing the outcomes of all trials other than $T_{i_1}, \ldots, T_{i_s}$ cannot cause X to be less than s, and so in order to "prove" to someone that $X \geq s$ it is enough to show him just the outcomes of $T_{i_1}, \ldots, T_{i_s}$. For example, if each T_i is a binomial variable equal to 1 with probability p and 0 with probability $1-p$, then if $X \geq s$ we could take $T_{i_1}, \ldots, T_{i_s}$ to be s of the trials which came up "1".

Remark. *Again, in a typical application c is a small constant. Also, as with the Chernoff Bound, if we wish to apply Talagrand's Inequality with $t > \mathbf{Med}(X)$, it usually suffices to apply $\mathbf{Pr}(|X - \mathbf{Med}(X)| > t) < \mathbf{Pr}(|X - \mathbf{Med}(X)| > \mathbf{Med}(X))$.*

The fact that Talagrand's Inequality proves concentration around the median rather than the expected value is not a serious problem, as in the situation where Talagrand's Inequality applies, those two values are very close together, and so concentration around one implies concentration around the other:

Fact. *Under the conditions of Talagrand's inequality, $|\mathbf{E}(X) - \mathbf{Med}(X)| \leq 40c\sqrt{\mathbf{E}(X)}$.*

This fact allows us to reformulate Talagrand's Inequality in terms of $\mathbf{E}(X)$.

Talagrand's Inequality II *Let X be a random variable determined by n independent trials $T_1, \ldots, T_n$, and satisfying*

1. *changing the outcome of any one trial can affect X by at most c, and*
2. *for any s, if $X \geq s$ then there are s trials $T_{i_1}, \ldots, T_{i_s}$ whose outcomes certify that $X \geq s$,*

then, for any $0 \leq t \leq \mathbf{E}(X)$,

$$\mathbf{Pr}(|X - \mathbf{E}(X)| > t + 60c\sqrt{\mathbf{E}(X)}) \leq 4e^{-\frac{t^2}{16c^2\mathbf{E}(X)}}.$$

Remark. *In almost every application, c is a small constant and we take t to be asymptotically much larger than $\sqrt{\mathbf{E}(X)}$, and so the $60c\sqrt{\mathbf{E}(X)}$ term is negligible. For the cases in which a smaller value of t is required, further strengthenings of Talagrand's Inequality will apply, but these go beyond the scope of this survey.*

The reader should now verify that Talagrand's Inequality yields a bound on the concentration of $BIN(n, p)$ nearly as good as that obtained from the Chernoff Bound.

Remark. *This statement is probably the simplest useful version of Talagrand's Inequality, and does not express its full power. In fact, the reader might note that this version does not imply the Simple Concentration Bound. We refer the reader to Chapter 4 of this book, or to [69] for more powerful versions of Talagrand's Inequality, including some from which the Simple Concentration Bound, with some weakening of the constant multiple in the exponent, is an easy corollary. We also refer the reader to [70] for a derivation of this form of Talagrand's Inequality from the statement originally presented in [84].*

We illustrate Talagrand's Inequality with one of its most important simple applications. This application to random permutations was one of the original applications in [84].

Let $\pi = x_1, \ldots, x_n$ be a uniformly random permutation of $1, \ldots, n$, and let X be the length of the longest increasing subsequence[1] of π. A well-known theorem of Erdős and Szekeres [26] states that any permutation of $1, \ldots, n$ contains either a monotone increasing subsequence of length $\lceil\sqrt{n}\rceil$ or a monotone decreasing subsequence of length $\lceil\sqrt{n}\rceil$. It turns out that the expected value of X is approximately $2\sqrt{n}$, i.e. twice the minimum guaranteed by the Erdős-Szekeres Theorem (see [40, 67]). A natural question is whether X is highly concentrated. Prior to the development of Talagrand's Inequality, the best result in this direction was due to Frieze [33] who showed that with high probability, X is within a distance of roughly $\mathbf{E}(X)^{2/3}$ of its mean, somewhat weaker than our usual target of $\mathbf{E}(X)^{1/2}$.

At first glance, it is not clear whether Talagrand's Inequality applies here, since we are not dealing with a sequence of independent random trials. Thus,

[1] In other words, a subsequence $x_{i_1} < x_{i_2} < \ldots < x_{i_k}$, where, of course, $i_1 < \ldots < i_k$.

we need to choose our random permutation in a more straightforward manner. We choose n uniformly random real numbers, $y_1, \ldots, y_n$ from the interval [0, 1]. Now arranging $y_1, \ldots, y_n$ in increasing order induces a permutation of $1, \ldots, n$ in the obvious manner[2].

It is easy to verify that changing the value of any one y_i can affect X by at most one. Furthermore, if $X \geq s$, then if there is an increasing subsequence of length s, then the s corresponding random reals clearly certify the existence of that increasing subsequence and so certify that $X \geq s$. Therefore Talagrand's Inequality implies that $\Pr(|X - \mathbf{E}(X)| > t + 60\sqrt{\mathbf{E}(X)}) \leq 4e^{-t^2/8\mathbf{E}(X)}$.

5. The Semirandom Method

Suppose that we wished to prove that the vertices of a graph could be partitioned into 2^k sets satisfying a particular property, P. The most straightforward probabilistic approach would be to generate a uniformly random partition, i.e. to individually place each of the vertices into a random part where each part is equally likely, and then prove that with positive probability this partition satisfies property P. Unfortunately this approach often does not work, but in many cases we can succeed by choosing a partition via a sequence of many random choices.

Our first step is to consider a uniformly random partition of the vertices into 2 sets, and to prove that with positive probability this partition satisfies an intermediary property P_1. This implies that there is at least one partition satisfying P_1, so we take that partition. Next, we prove that we can find a 2-partition of each of our parts satisfying property P_2, by considering a uniformly random partition of each part and using the fact that the first partition satisfies P_1 to prove that with positive probability the random refinement satisfies P_2. Repeating this process k times, we prove the existence of a 2^k-partition satisfying P_k, which of course we choose to be property P. Examples of this technique can be found in [9, 10, 23].

At first glance, it appears that our argument just reduces to a complicated way to take a uniformly random 2^k-partition. It is important to note that this is not the case. If we had simply taken a sequence of k uniformly random 2-partitions, then we would have formed a uniformly random 2^k partition. However, at each step we do not take a uniformly random 2-partition – we merely consider a uniformly random 2-partition in order to prove the existence of a particular partition which satisfies our intermediary property. For example, if we apply the Local Lemma at each step, then the probability that a uniformly random 2-partition satisfies our intermediary property might be exponentially small, and so the partition that we take doesn't resemble a uniformly random partition at all.

[2] Because these are uniformly random real numbers, it follows that with probability 1, they are all distinct.

This technique is an example of what is known as the semirandom method, which is the term used when we prove the existence of something by generating it through many iterations, applying the probabilistic method at each iteration. The semirandom method is often referred to as the Rödl Nibble, because many applications were inspired by a series of refinements of the arguments in [66].

One area of graph theory where the semirandom method has had the greatest impact is graph colouring. In fact, many of the strongest results in graph colouring over the past decade are examples of this method, including [45, 58, 28, 40, 41, 54, 27, 49]. In this section, we will briefly discuss some of these applications. For a more thorough discussion, we refer the reader to [63] or [58].

In the most basic type of application, we wish to show that a graph has a proper vertex colouring using only C colours. We prove that such a colouring exists through several iterations of colouring a few vertices each time, showing that eventually we can find a proper colouring of the entire graph. For the first iteration, we consider assigning to each vertex a random colour. Of course with high probability many pairs of adjacent vertices will have the same colour. We address this problem as follows: If any vertex receives the same colour as a neighbour, then we uncolour that vertex. Clearly, the set of vertices which retain their colours form a proper partial colouring. During each subsequent iteration, we consider assigning to each uncoloured vertex a random colour chosen from amongst those colours which were not retained by any of its neighbours during an earlier iteration, and then we uncolour some vertices as before. Our goal is to show that after each iteration, the partial colouring satisfies a particular property with positive probability, thus showing that we can choose a partial colouring satisfying that property. After several iterations, the final property will imply that the partial colouring can be completed to a valid proper colouring of the graph.

This method also applies well to list colouring problems[3]. At each iteration, we assign to each uncoloured vertex a colour chosen uniformly at

[3] The basic list colouring problem is to find a proper vertex colouring of a graph G where every vertex has a list of permissible colours. The tricky part is that the vertices typically have different lists. If G has the property that we can always succeed for any set of lists, so long as they each contain at least k colours, then we say that G is k-choosable. The list chromatic number of G, denoted by $\chi_l(G)$ is the smallest k such that G is k-choosable. Note that $\chi_l(G) \geq \chi(G)$ by considering the case where all the lists are equal. List edge colouring problems are defined similarly and the list chromatic index of G, $\chi_l'(G)$, is the corresponding extension of the chromatic index (also known as the edge chromatic number), see [8].

random from its list. If a vertex retains its colour then we delete that colour from the lists of its neighbours.

At each iteration, our proof usually consists of: (1) computing the expected values of a few variables, (2) proving that those variables are concentrated by applying the tools in Section 4, and (3) applying the Local Lemma.

5.1 Triangle-free Graphs

It is well-known that the chromatic number of any graph with maximum degree Δ is at most $\Delta+1$, and in fact such a colouring can be obtained via a simple greedy colouring algorithm. Johansson [36] used the semi-random method to prove that if G is triangle-free and has maximum degree Δ then $\chi(G) = O(\frac{\Delta}{\log\Delta})$, which is best possible up to a constant multiple. (Independently, Kim [41] obtained the same bound for the chromatic number of graphs with girth at least 5.) Johansson [37] subsequently refined his argument to show that for any constant r, if G is K_r-free and has maximum degree Δ then $\chi(G) = O(\frac{\Delta}{\log\Delta} \times \log\log\Delta)$.

Here, we will indicate why the semi-random colouring procedure described earlier should work so well on triangle-free graphs by describing how, using only a single iteration of that procedure, one can prove that the chromatic number of such a graph is a constant multiple less than Δ. We remark that this proof is presented merely to illustrate the technique, and the result is by no means best possible. In fact, there are much simpler proofs which yield slightly stronger results (see for example [55, 65]), and as mentioned above, there are more complicated proofs which yield much stronger results.

Theorem 5.1. *If G is triangle-free and has maximum degree Δ sufficiently large, then $\chi(G) \le (1-\frac{1}{2e^6})\Delta$.*

In fact, what we show is that if we carry out a single iteration of our procedure, using only $\frac{\Delta}{2}$ colours, then with positive probability the resulting partial colouring will be such that every vertex v has several colours which appear at least twice in its neighbourhood, which we call repeated colours (for v).

Lemma 5.2. *If G is triangle-free and has maximum degree Δ sufficiently large, then G has a partial colouring such that for each vertex v, N_v contains at least $\frac{\Delta}{2e^6}+1$ repeated colours.*

It is straightforward to show that the partial colouring guaranteed by Lemma 5.2 can be completed to a $(1-\frac{1}{2e^6})\Delta$ colouring of the entire graph using a simple greedy procedure, and so Lemma 5.2 implies Theorem 5.1.

The outline of the proof is as follows. We can assume that G is Δ-regular since it is easy to show that any graph with maximum degree Δ can be embedded in a Δ-regular graph.

For each vertex v, we let Z_v denote the number of colours retained by exactly two vertices in N_v (the neighbourhood of v). Because G is triangle-free, no two vertices in N_v are adjacent and so any such pair is eligible to retain the same colour (obviously if two vertices are adjacent then they cannot retain the same colour). The probability that two vertices retain the same colour and that no other vertex in N_v retains it is $\frac{2}{\Delta}(1-(\frac{2}{\Delta}))^{3\Delta-3}$ which is at least $\frac{2}{e^6\Delta}$, and so by linearity of expectation, $\mathbf{E}(Z_v) \ge \binom{\Delta}{2} \times \frac{2}{e^6\Delta} \approx \frac{\Delta}{e^6}$. Using either a straightforward application of Talagrand's Inequality or a clever application of Azuma's Inequality, we can show that $\Pr(Z_v \le \frac{1}{2}\mathbf{E}(Z_v)+1) < e^{-\Theta(\Delta)}$.

We let A_v be the event that $Z_v \le \frac{1}{2}\mathbf{E}(Z_v)+1$. It follows from the Mutual Independence Principle that each A_v is mutually independent of all but at most Δ^4 other events. Thus by the Local Lemma with positive probability A_v does not hold for any vertex v, and so Lemma 5.2 follows.

To obtain stronger results, such as those in [36, 37, 41], we must apply several iterations of this procedure, at each step keeping careful track of the number of neighbours of v which retain a colour, the number of colours appearing on the neighbourhood of v, and one or two other variables. To obtain the results in [38, 47] we must use a more sophisticated variant of this semirandom colouring procedure, but we will not go into such details here.

5.2 Sparse Graphs

It is straightforward to show that the argument used in the proof of Lemma 5.2 applies to a wider class of graphs than triangle-free graphs. In particular it will apply as long as for each vertex v, N_v does not have too many edges. For $\epsilon > 0$, if $|E(N_v)| \le (1-\epsilon)\binom{\Delta}{2}$ then we say that v is ϵ-sparse. If every vertex of a graph is ϵ-sparse then that graph is said to be ϵ-sparse.

Lemma 5.3. *If for some constant $\epsilon > 0$, G is ϵ-sparse and has maximum degree Δ sufficiently large, then $\chi(G) \le (1-\frac{\epsilon}{e^6})\Delta$.*

This was a key lemma for the bound on the strong chromatic index in [49]. Lemma 5.3 still holds for some values of $\epsilon = o(1)$. We leave it as an exercise for the reader to determine how small ϵ can be. It is not hard to verify that Lemma 5.3 also holds when we replace χ by χ_ℓ, the list chromatic number.

Applying the aforementioned theorem of Johansson concerning triangle-free graphs, Alon, Krivelevich and Sudakov[6] provided an extension of that extension to graphs which are merely very sparse, showing that for any $\varepsilon > 0$, if G has maximum degree Δ sufficiently large, and is $(1 - \Delta^{-\varepsilon})$-sparse (i.e. if the neighbourhood of any vertex v contains at most $\frac{1}{2}\Delta^{2-\varepsilon}$ edges), then $\chi(G) \le O(\frac{\Delta}{\varepsilon \ln \Delta})$. This result also applies to the list chromatic number.

In general, if a graph is sufficiently sparse then by performing several iterations of our semirandom colouring procedure, we can often obtain even stronger results. The most well known of these results is probably the following theorem of Kahn [38], which proved that the well-known List Colouring Conjecture (see e.g. [36]) that the list chromatic index of a graph is equal to its chromatic index, is asymptotically correct.

Theorem 5.4. If G has maximum degree Δ, then $\chi_l'(G) = \Delta + o(\Delta)$.

Häggkvist and Janssen [34], using a different technique, which involved an application of the Local Lemma, tightened this to $\Delta + O(\Delta^{2/3}\text{polylog}\,\Delta)$. By analyzing the semirandom procedure more precisely, Molloy and Reed [50] improved it further to $\Delta + O(\Delta^{1/2}\text{polylog}\,\Delta)$. The bounds of Kahn and of Molloy and Reed also apply to hypergraphs, yielding for example that for any constant k, the list chromatic index of a linear k-uniform hypergraph with maximum degree Δ is at most $\Delta + O(\Delta^{1-1/k}\text{polylog}\,\Delta)$. For similar bounds regarding non-linear hypergraphs see [38, 50].

5.3 Dense Graphs

If a graph is not very sparse, for example if for some vertex v, N_v is very close to being a Δ-clique, then it is easy to see that our basic semirandom procedure will not work very well, as with high probability N_v will not contain many repeated colours. Suppose for example that G is a $(\Delta + 1)$-clique with a perfect matching removed. Here, $\chi(G) = \frac{\Delta+1}{2}$, but our argument will only yield the far from satisfactory bound $\chi(G) \le \Delta - d$ for some $d = o(\Delta)$.

Reed developed a variation of our procedure which works well in such situations. The main step is to show that a graph can be partitioned into a sparse region and several dense regions such that there are very few edges between any two regions. This allows us to essentially colour each region separately.

The Reed Decomposition[14]: *For any $\varepsilon > 0$ and any graph G with maximum degree Δ, G can be decomposed into $S, D_1, \ldots, D_l$ such that*

a) each vertex in S is ε-sparse,

b) each D_i very closely resembles a clique

c) for each i, the number of edges from D_i to $G - D_i$ is at most $4\varepsilon\Delta^2$.

It can also be shown that each D_i satisfies a handful of other conditions which often change slightly by application, as does the precise sense in which each D_i resembles a clique.

Given this decomposition, we modify our semirandom procedure as follows. We assign to each vertex of S a random colour as usual. For each D_i, we take a specific proper colouring of D_i and permute its colours at random.

Reed's first application was the following:

Theorem 5.5. There exists some constant $\varepsilon > 0$ such that for every graph G with maximum degree Δ and maximum clique size ω, $\chi(G) \le \lceil \varepsilon\omega + (1 - \varepsilon)(\Delta + 1) \rceil$.

Reed conjectures that for Δ sufficiently large, this theorem holds with $\varepsilon = \frac{1}{2}$ (he shows that it does when ω is sufficiently close to Δ). It cannot hold for any $\varepsilon < \frac{1}{2}$.

By applying the Reed Decomposition with $\varepsilon = o(1)$, Reed [57] proved the similar theorem:

Theorem 5.6. If G has maximum degree Δ sufficiently large and no clique of size Δ then $\chi(G) \le \Delta - 1$.

This was conjectured to be true for $\Delta \ge 9$ by Borodin and Kostochka [8] and for Δ sufficiently large by Beutelspacher and Hering [6].

Another application of the Reed decomposition is the following bound on the total chromatic number due to Molloy and Reed [49], which is the best progress thus far on the conjecture of Vizing [66] and Behzad [5] that the total chromatic number of a graph is at most its maximum degree plus two.

Theorem 5.7. If G has maximum degree Δ sufficiently large then $\chi_T(G) \le \Delta + 10^{26}$.

6. Ramsey Theory

The Probabilistic Method has arguably had a greater impact on Ramsey Theory than on any other field of combinatorics, with the possible exceptions of graph colouring and combinatorial number theory. Erdős' proof that $R(k,k) \ge \Omega(k \times 2^{k/2})$ is probably the best known classical result of the First Moment Method. (We invite the reader to try to prove this, and, after having done so, to improve the constant term by using the Local Lemma.) More recently, some exciting new work has been done towards establishing the asymptotic value of $R(3,k)$. We outline some of the milestones here.

6.1 An Upper Bound

Using what is probably the earliest application of the semirandom method, Ajtai, Komlós and Szemerédi[1, 2] were the first to show that $R(3,k) \leq O(k^2/\ln k)$. Shearer [59, 60] reduced the constant term and simplified the proof significantly. We present here a refinement of Shearer's proof due to Alon [3]. The main step is the following:

Theorem 6.1. *If G is triangle-free and has maximum degree Δ, then G has a stable set of size at least $|V(G)| \times \frac{\ln \Delta}{8\Delta}$.*

Corollary 6.2. $R(3,k) \leq 8\frac{k^2}{\ln k}$

Proof. Set $n = 8\frac{k^2}{\ln k}$. We wish to show that any graph G on n vertices has either a triangle or a stable set of size k. If G has a vertex of degree greater than k, then clearly this must hold. Otherwise, apply Theorem 6.1 with $\Delta \leq k$.

Proof of Theorem 6.1. Let I be a stable set chosen uniformly at random from amongst all stable sets of G. Unlike most other random choices discussed in this survey, there is no obvious efficient way to actually choose I. Nevertheless, we will be able to show that $\mathbf{E}(|I|) \geq |V(G)| \times \frac{\ln \Delta}{8\Delta}$, thus proving our theorem.

For each vertex v define Z_v as follows. $Z_v = \Delta$ if $v \in I$, and $Z_v = |N(v) \cap I|$ otherwise. Since $\sum_{v \in V(G)} Z_v \leq 2\Delta |I|$, it will suffice to show that $\mathbf{E}(Z_v) \geq \frac{1}{4}\ln \Delta$ for every v.

Set $I' = I \cap V(G - (v \cup N(v)))$. We will show that for any possible choice of I', the conditional expected value $\mathbf{E}(Z_v | I')$ is at least $\frac{1}{4}\ln \Delta$, which clearly establishes that $\mathbf{E}(Z_v) \geq \frac{1}{4}\ln \Delta$.

Upon specifying I', set W to be the neighbours of v which are not adjacent to any vertex of I'. Any independent set of $v \cup W$ is equally likely to be the completion of I' in I. Since G is triangle-free, W contains no edges, and so there are $1 + 2^{|W|}$ such independent sets - one which only contains v, and the $2^{|W|}$ subsets of W. Clearly, the average size of the latter group of sets is $\frac{1}{2}|W|$. Therefore,

$$\mathbf{E}(Z_v | I') = \frac{\Delta + \frac{1}{2}|W| \times 2^{|W|}}{1 + 2^{|W|}}$$

which one can compute to be at least $\frac{1}{4}\ln \Delta$ for any $0 \leq |W| \leq \Delta$. To do this, if $\frac{1}{2}\ln \Delta \leq |W| \leq \Delta$ then we can apply $\mathbf{E}(Z_v) \geq \frac{1}{4}|W|$, while if $|W| < \frac{1}{2}\ln \Delta$ then we can apply $\mathbf{E}(Z_v) \geq \Delta/(1 + 2^{|W|})$. □

6.2 A Weak Lower Bound

Erdős[33] was the first to prove that $R(3,k)$ was at least $\Omega(\frac{k^2}{\ln^2 k})$. Subsequently, the proof was simplified and/or the constant term was improved in [51, 57, 25, 49]. We present here a short proof of Krivelevich[47], showing:

Theorem 6.3. *For k sufficiently large, $R(3,k) \geq (\frac{k}{10000 \ln k})^2$.*

Remark. *The constant term can be improved significantly by using a stronger version of the Chernoff Bound, amongst other things.*

Proof. Our goal is to prove that there exists a triangle-free graph on $n = (\frac{k}{10000 \ln k})^2$ vertices with no independent set of size k. We will do so by constructing such a graph randomly.

We first choose a random graph G on n vertices where each of the $\binom{n}{2}$ edges is chosen to be present with probability $p = \frac{1}{\sqrt{n}}$. Next, we choose any maximal set T of edge-disjoint triangles in G and we let G' be the graph formed by removing the edges of T from G. Clearly, G' has no triangles and so it will suffice to show that with positive probability G' has no stable set of size at least k.

Consider any set S of k vertices. Let X be the number of the edges within S, and let Y be the number of triangles of T which have at least one edge in S. Since deleting T from G removes at most $3Y$ edges from S, the probability that S is a stable set in G' is at most the probability that $X \leq 3Y$, which we will show is very small.

First, we bound the probability that X is small. $\mathbf{E}(X) = \binom{k}{2}p = \frac{k(k-1)}{2} / \frac{k}{10000 \ln k} = 5000(k-1)\ln k$. Therefore, it follows from the Chernoff Bound that $\Pr(X < 4000k\ln k) < e^{-100k\ln k} = k^{-100k}$.

Now we bound the probability that Y is large. For any t, if $Y \geq t$ then there must be some collection of t triples of vertices $(a_1, b_1, c_1), \ldots, (a_t, b_t, c_t)$ such that (1) no pair of vertices lies in two triples, (2) for each i we have $a_i, b_i \in S$, and (3) each triple forms a triangle in G. The expected number of such collections is at most

$$\binom{\binom{k}{2}}{t} \times n^t p^{3t} < \frac{(5000k\ln k)^t}{t!}$$

Thus, by Markov's Inequality, $\Pr(Y \geq t) \leq (5000k\ln k)^t/t!$, and it follows that:

$$\Pr(Y \geq 1333k\ln k) \leq (\frac{e}{4})^{1333k\ln k} < k^{-3k}.$$

Therefore, the probability that S is a stable set in G' is at most $2k^{-3k}$, and so the expected number of stable sets of size k is at most

$$\binom{n}{k} \times 2e^{-\frac{k^2}{8}} < n^{2k} \times 2e^{-\frac{k^2}{8}} < 1$$

for n sufficiently large. Therefore, by the First Moment Principle, with positive probability, G has no stable sets of size k, thus proving the theorem. ∎

6.3 A Tight Lower Bound

One of the most celebrated combinatorial results of the last few years was Kim's proof that $R(3,k) \geq \Omega(\frac{k^2}{\log k})$ [48], thus establishing the correct asymptotic value of $R(3,k)$ up to a constant multiple. This was inspired in part by Spencer's proof[68] that $R(3,k)$ is asymptotically of a higher order than $\frac{k^2}{\log^2 k}$. Kim's proof consisted of a very delicate application of the semi-random method, which we briefly outline here.

Our goal is to construct a triangle-free graph G on $n = \frac{k^2}{\log k}$ vertices with no stable set of size k. We actually build two graphs, G and H, and we keep track of a set E of permissible edges.

Initially, $G = H = \emptyset$, and E is the set of all possible edges on the n vertices. At each iteration, each edge $e \in E$ is added to H with probability p. We call these added edges new edges. We remove from E every new edge along with any edge e such that e forms a triangle with two edges from H.

Note that this does not ensure that H is triangle-free, as it is possible that 2 or 3 edges of a triangle could enter H during the same iteration. In this case, we call such a pair or triple of edges bad. From the set of new edges we remove a maximal edge-disjoint collection of bad pairs and triples, and we add the remaining edges to G. Note that G will remain triangle free.

The reader might now protest that this procedure is slightly wasteful. For example, it was not necessary to remove from E any edge which formed a triangle with two edges from H – it would have sufficed to remove an edge only if it did so with two edges from G. However, by being wasteful in this way the analysis is simplified significantly.

The main work lies in bounding the stability number of G. We do this using the First Moment Method. Consider any set I of k vertices. Kim shows that the probability of I being a stable set in G is smaller than $\binom{n}{k}^{-1}$, and so with positive probability G does not have a stable set of size k.

To do so, he shows that after each iteration, with very high probability, several parameters remain close to their expected values, including a few which control the number of potential edges from I which are in G, H and E. Unlike other applications of the semi-random method that we have discussed, at each step he uses the First Moment Method, not the Local Lemma. For details, see [47] or [25].

7. Algorithms

In its purest form, the probabilistic method merely proves the existence of a combinatorial object, such as a satisfying assignment or a colouring of a graph, without indicating how to find the object efficiently. An application of the First Moment Method will often prove that if we choose the object at random, it will meet our requirements with high probability, and this generally yields a simple efficient randomized algorithm (a formal definition of a randomized algorithm is given in Chapter 5 of this book – we will not need it here). On the other hand, when applying the Local Lemma, usually the object meets our requirements with exponentially low probability and so there is no obvious algorithm to construct it, not even a randomized one.

In this section, we will discuss general procedures to obtain deterministic algorithms from applications of the First Moment Method and both randomized and deterministic algorithms from applications of the Local Lemma.

7.1 The First Moment Method

The most common technique for derandomizing an application of the First Moment Method is the so-called Method of Conditional Probabilities due to Erdős and Selfridge [33]. We begin by presenting a deterministic algorithm for finding the satisfying assignment guaranteed by Theorem 1.2.

Recall that we are given a boolean formula $\mathcal{F}$ in conjunctive normal form on the variables $x_1,\ldots,x_n$ such that if we were to set each x_i to be True with probability $\frac{1}{2}$ and False with probability $\frac{1}{2}$, then the expected value of X, the number of unsatisfied clauses in $\mathcal{F}$ is less than 1. We will use this fact to deterministically assign truth values to each variable in sequence.

First, we consider x_1. Suppose that we assign x_1 = True. This reduces $\mathcal{F}$ to a smaller boolean formula $\mathcal{F}_T$ as follows: (i) every clause in $\mathcal{F}$ which contains the literal x_1 is removed from $\mathcal{F}$ since that clause is now satisfied, and (ii) every clause which contains the literal $\overline{x_1}$ is shortened by removing that literal since that clause can no longer be satisfied by setting x_1 = False (if a clause shrinks to size 0 then $\mathcal{F}_T$ is unsatisfiable). Similarly, if we assign x_1 = False, then $\mathcal{F}$ reduces to $\mathcal{F}_F$.

Now consider taking a random truth assignment of $x_2,\ldots,x_n$ where each variable is set to True with probability $\frac{1}{2}$ and False with probability $\frac{1}{2}$. It is easy to deterministically calculate the expected number of unsatisfied clauses in $\mathcal{F}_T$ or in $\mathcal{F}_F$. Note that these expected values are equal to the conditional expected values $\mathbf{E}(X|x_1 = \text{True})$ and $\mathbf{E}(X|x_1 = \text{False})$ respectively. The important fact is that one of these two values is no bigger than $\mathbf{E}(X)$, since $\mathbf{E}(X) = \frac{1}{2}\mathbf{E}(X|x_1 = \text{True}) + \frac{1}{2}\mathbf{E}(X|x_1 = \text{False})$. Therefore, at least one of these expected values is less than 1, and we set x_1 accordingly.

We now repeat this process, setting each variable one at a time, so that at each step the resulting formula has the property that if we were to take a random truth assignment for the remaining variables, the expected number of unsatisfied clauses is less than 1. After all variables have been set, this expected value is simply the number of unsatisfied clauses in the truth assignment that we have formed. Since it is less than 1, it must be equal to 0 and so we have found a satisfying assignment.

This technique generalizes in an obvious manner. Its general setting is as follows. X is a random variable determined by a sequence of random trials $T_1, \ldots, T_t$. Our problem is to find a set of outcomes $t_1, \ldots, t_t$ such that $X \leq \mathbf{E}(X)$.

Of all the possible outcomes of T_1, at least one of them, t_1, must be such that the conditional expected value $\mathbf{E}(X|T_1 = t_1)$ is at most $\mathbf{E}(X)$. We select this outcome, and then repeat this step on each T_i in order, each time choosing t_i such that

$$\mathbf{E}(X|T_1 = t_1, \ldots, T_i = t_i) \leq \mathbf{E}(X) \tag{7.1}$$

By the time we have selected t_t, there are no more random choices to be made, and so $\mathbf{E}(X|T_1 = t_1, \ldots, T_t = t_t)$ is just the value of X determined by $t_1, \ldots, t_t$. Thus we have found a set of outcomes for which $X \leq \mathbf{E}(X)$, as desired.

In order for this approach to succeed, we simply require that (a) the number of trials is not too large, and (b) at each step we can choose an outcome satisfying eq (7.1) efficiently. For example, it suffices that the following conditions hold:

1. The number of trials is a polynomial in the size of the input.
2. The number of possible outcomes of each trial is a polynomial in the size of the input.
3. We can compute any conditional expected value in polytime.

If these three conditions hold, then the running time of this deterministic algorithm will be at most the product of these three polynomials.

7.2 The Lovász Local Lemma

Beck [13] introduced a constructive version of Theorem 3.1 (actually of a variant of Theorem 3.1) with some weakening of the constant terms (see also [4]). In particular, he provided a polynomial expected time randomized algorithm to find a satisfying assignment for any instance of k-SAT in which each variable lies in at most $2^{k/48}$ clauses. We will briefly outline his algorithm for the case where k is a large constant.

Suppose that we are given such a CNF formula F with n variables and m clauses.

During Phase 1 of the algorithm, we assign a random value to each variable, one at a time. Naturally, we expect that most clauses will be satisfied. However, if there are an enormous number of clauses, it is inevitable that a few might have all of their literals set the wrong way. If a clause ever has $\frac{k}{2}$ of its literals set without first becoming satisfied, then we call that clause dangerous and we freeze its remaining literals, i.e. we will not assign any values to them until after the end of Phase 1, at which time they can be dealt with more carefully.

At the end of Phase 1, with high probability most of the clauses will be satisfied. The only unsatisfied clauses are the dangerous clauses along with some clauses which did not become dangerous but which had some of their literals frozen because they intersect dangerous clauses. For example, it is possible that every variable in a clause appears in some other clause which becomes dangerous, and so that clause might not have any of its variables set at all. It is important to note that, by definition, every unsatisfied clause contains at least $\frac{k}{2}$ frozen variables.

Thus, if we consider the formula F_1 induced by the unsatisfied clauses and the frozen variables, every clause will have size at least $\frac{k}{2}$. Since $4 \times 2^{-\frac{k}{2}} \times (k \times 2^{k/48}) < 1$, the Local Lemma guarantees that F_1 is satisfiable. Note that a satisfying assignment for F_1 will complete the partial assignment made during Phase 1 into a satisfying assignment of F.

The main part of the proof is to show that with high probability F_1 is the union of many disjoint formulas, each consisting of at most $O(\log n)$ clauses. Therefore, we can process each of them separately, and at first we can do so by using exhaustive search of all the possible $2^{O(\log n)} = \mathrm{poly}(n)$ truth assignments to find the one guaranteed by the Local Lemma.

If we wish to speed this algorithm up, we can repeat Phase 1 on F_1. By a similar analysis, with high probability this will reduce F_1 to a set of disjoint formulas each of size $O(\log \log n)$ which can be processed by exhaustive search in $\mathrm{poly}(\log n)$ time each, thus yielding a $O(n\,\mathrm{poly}(\log n))$ time randomized algorithm. Every property which we have claimed to hold with high probability can be shown to do so by the First Moment Method, thus the Method of Conditional Probabilities described in the previous section applies to produce a polytime deterministic algorithm.

For details of the proof that the components of F_1 are all small with high probability, we refer the reader to [13], [6], [5], or [64]. The intuition is as follows. As long as each clause intersects at most $d = k \times 2^{k/48}$ other clauses, one can show that any connected subformula of F_1 on X variables must contains at least X/d^5 disjoint dangerous clauses, all relatively close together (we do not define this precisely here). The probability that any particular set of X/d^5 disjoint clauses all become dangerous is at most $2^{-\frac{k}{2}\cdot\frac{X}{d^5}}$. For each vertex v, one can show that there are at most $(4d^3)^{X/d^5}$ sets of disjoint clauses which are relatively close together and such that at least one of them contains v. Applying the First Moment Method with $X = \beta \log n$ yields the desired result.

More generally, we can apply this approach whenever our underlying probability space is a sequence of independent random trials (here p and d are probability and dependency bounds as before). It works well provided that d is constant, and p sufficiently small $pd^9 < \frac{1}{8}$ (for details see [13]). If d is not constant then we can often show that the algorithm still works. We can also lower the constant "9" somewhat. However, this procedure will not work when p is of order $\frac{1}{d}$.

Recall that the Local Lemma only requires that $pd < \frac{1}{e}$. However, in many applications, the stronger condition $pd^9 < \frac{1}{8}$ still applies. Consider, for example, the case where every bad event is determined by exactly t random trials for some t, and where each trial helps to determine at most r bad events. In this case, it follows from the Mutual Independence Principle that each event is independent of all but at most tr other events. Frequently, the probability of each bad event is at most $p = e^{-\alpha t}$ for some constant α, for example when we bound this probability by using one of the concentration inequalities of Section 4. Thus, as long as r is not much bigger than t, for example, if r is a polynomial in t, then $pd^9 < \frac{1}{8}$ for any constant α so long as t is sufficiently large.

Molloy and Reed [51] modified Beck's procedure to work on a wider class of problems which seems to cover almost all applications of the Local Lemma, including the General Local Lemma, so long as d does not grow very large with the size of the input and so long as some of the parameters are sufficiently large. This includes applications where p is of order $\frac{1}{d}$, for which Beck's technique does not apply. Again, in many cases when d does grow quickly the technique of [51] will still apply. For more details, see [51] or [52].

It should be noted that with both of these techniques, the running time of the algorithm is polynomial in the number of random trials and the number of bad events. Thus, in applications of the Local Lemma where the number of bad events is not polynomial in the size of the input, for example Theorem 3.3, this does not always result in a polytime algorithm.

References

1. Ajtai M., Komlós J. and Szemerédi E. (1980) A note on Ramsey numbers, J. Comb. Th. A 29, 354 - 360.

2. Ajtai M., Komlós J. and Szemerédi E. (1981) A dense infinite Sidon sequence, Eur. J. Comb. 2, 1 - 11.
3. Alon N. (1996) Disjoint directed cycles, J. Comb. Th. (B) 68, 167 - 178.
4. Alon N. (1996) Independence numbers of locally sparse graphs and a Ramsey type problem, Rand. Struct. Alg. 9, 271 - 278.
5. Alon N. (1992) The strong chromatic number of a graph, Random Structures and Algorithms, 3, 1 - 7.
6. Alon N. (1991) A parallel algorithmic version of the Local Lemma, Random Structures and Algorithms, 2, 367 - 378.
7. Alon N. (1988) The linear arboricity of graphs, Isr. J. Math., 62, 311 - 325.
8. Alon N., Krivelevich M. and Sudakov B. (1998) List colouring of random and pseudo-random graphs, preprint.
9. Alon N. and Linial N. (1989) Cycles of length 0 modulo k in directed graphs, J. Comb. Th. (B) 47, 114 - 119.
10. Alon N., McDiarmid C. and Molloy M. (1996) Edge-disjoint cycles in regular directed graphs, J. Graph Th. 22, 231 - 237.
11. Alon N. and Spencer J. (1992) The Probabilistic Method. Wiley.
12. Azuma K. (1967) Weighted sums of certain dependent random variables, Tôhoku Math. Journal 19, 357 - 367.
13. Beck J. (1991) An algorithmic approach to the Lovász Local Lemma, Random Structures and Algorithms, 2, 343 - 365.
14. Bermond J. and Thomassen C. (1981) Cycles in digraphs - a survey, J. Graph Th. 5, 1 - 43.
15. [illegible] M. (1987) Graphs and Their Chromatic Numbers, Ph.D. thesis, Michigan State University.
16. Beutelspacher A. and Hering P. (1984) Minimal graphs for which the chromatic number equals the maximal degree, Ars Combinatoria 18, 201 - 216.
17. Bollobás B. (1985) Random Graphs. Academic Press, London.
18. Bollobás B. and Harris A. (1985) List colourings of graphs, Graphs and Comb. 1, 115 - 127.
19. Borodin O. and Kostochka A. (1977) On an upper bound on a graph's chromatic number, depending on the graph's degree and density, JCTB (B) 23, 247 - 250.
20. Broder A.Z., Frieze A. and Upfal E. (1996) Static and dynamic path selection on expander graphs: a random walk approach, STOC.
21. Erdős P. (1947) Some remarks on the theory of graphs, Bull. Amer. Math Soc. 53, 292 - 294.
22. Erdős P. (1959) Graph theory and probability, Canadian J. of Math. 11, 34 - 38.
23. Erdős P. (1961) Graph theory and probability II, Canadian J. of Math. 13, 346 - 352.
24. Erdős P. and Lovász L. (1975) Problems and results on 3-chromatic hypergraphs and some related questions, in "Infinite and Finite Sets" (A. Hajnal et. al. Eds), Colloq. Math. Soc. J. Bolyai 11, North Holland, Amsterdam, 609 - 627.
25. Erdős P., Rubin A. and Taylor H. (1979) Choosability in graphs, Congr. Num. 26, 125 - 157.
26. Erdős P. and Selfridge J. (1973) On a combinatorial game, J. Comb. Th. (A) 14, 298 - 301.
27. Erdős P., Suen S. and Winkler P. (1995) On the size of a random maximal subgraph, Random Structures and Algorithms 6, 309 - 318.
28. Fernandez de la Vega W. (1983) On the maximum cardinality of a consistent set of arcs in a random tournament, J. Comb. Th. (B) 35, 328 - 332.

29 Frieze A. (1991): On the length of the longest monotone subsequence in a random permutation, Ann. Appl. Prob. 1, 301 - 305.

30 Frieze A. and Molloy M. (1996): Splitting expander graphs, in preparation.

31 Häggkvist R. and Janssen J. (1997): New bounds on the list chromatic index of the complete graph and other simple graphs, Combin. Prob. and Comp. 6 295 - 313.

32 Hind H., Molloy M. and Reed B. (1996): Colouring a graph frugally, Combinatorica, to appear.

33 Hind H., Molloy M. and Reed B. (1998): Total colouring with $\Delta + poly(\log \Delta)$ colours, SIAM J. of Computing, to appear.

34 Hoeffding W. (1963): Probability inequalities for sums of bounded random variables, J. Amer. Statist. Assoc., 58, 13 - 30.

35 Jensen T. and Toft B. (1995): Graph colouring problems, Wiley, New York.

36 Johansson A. (1996): Asymptotic choice number for triangle free graphs, DIMACS Technical Report 91-5.

37 Johansson A. (1996): The choice number of sparse graphs, manuscript.

38 Kahn J. (1996): Asymptotically good list-colorings, J. Combinatorial Th. (A) 73, 1 - 59.

39 Kahn J. (1996): Asymptotics of the chromatic index for multigraphs, J. Combinatorial Th. (B), 68, 233 - 255.

40 Kahn J. (1996): Asymptotics of the list-chromatic index for multigraphs, manuscript.

41 Kim J.H. (1995): On Brooks' Theorem for sparse graphs, Combinatorics, Probability and Computing 4, 97 - 132.

42 Kim J.H. (1995): The Ramsey number R(3,t) has order of magnitude $t^2/\log t$, Random Structures and Algorithms 7, 173 - 207.

43 Krivelevich M. (1995): Bounding Ramsey numbers through large deviation inequalities, Random Struct. and Alg. 7, 145 - 155.

44 Lawrence J. (1978): Covering the vertex set of a graph with subgraphs of smaller degree, Disc. Math 21, 61 - 68.

45 Logan B. and Shepp L. (1977): A variational problem for random Young tableaux, Adv. Math. 26, 206 - 222.

46 Lovász L. (1968): On chromatic number of finite set-systems, Acta Math. Acad. Sci. Hung. 19, 59 - 67.

47 McDiarmid C. (1989): On the method of bounded differences, Surveys in Combinatorics, Proceedings of the Twelfth British Combinatorial Conference 148 - 188.

48 Molloy M. and Reed B. (1997): A bound on the strong chromatic index of a graph, J. of Comb. Th. (B) 69, 103 - 109.

49 Molloy M. and Reed B. (1998): A bound on the total chromatic number, Combinatorica, to appear.

50 Molloy M. and Reed B. (1996): Asymptotically better list colourings, manuscript.

51 Molloy M. and Reed B. (1998): Further algorithmic aspects of the Local Lemma, to appear in the proceedings of the 30th ACM Symposium on Theory of Computing.

52 Molloy M. and Reed B. (1998): Graph Colouring via the Probabilistic Method, preprint.

53 Molloy M. and Reed B.: Graph Colouring and the Probabilistic Method, a book in preparation.

54 Nesetril J. and Rödl V. (1979): A short proof of the existence of highly chromatic hypergraphs without short cycles, J. Comb. Th. (B) 27, 225 - 227.

55 Pippenger N. and Spencer J. (1989): Asymptotic behaviour of the chromatic index for hypergraphs, J. Combinatorial Th. (A) 51, 24-42.

56 Reed B. (1997): ω, Δ, and χ, Journal of Graph Theory, to appear.

57 Reed B. (1998): A strengthening of Brooks' Theorem, manuscript.

58 Rödl V. (1985): On a packing and covering problem, Europ. J. Combinatorics 6, 69-78.

59 Shearer J. (1983): A note on the independence number of triangle-free graphs, Disc. Math. 46, 83 - 87.

60 Shearer J. (1995): On the independence number of sparse graphs, Rand. Struc. Alg. 7, 269 - 271.

61 Spencer J. (1977): Asymptotic lower bounds for Ramsey functions, Disc. Math. 20, 69 - 76.

62 Spencer J. (1994): Maximal triangle-free graphs and the Ramsey number $R(3,k)$, manuscript.

63 Steele T. (1986): [illegible] Mat. Fis. [illegible] 60, 327 - 356.

64 Talagrand M. (1995): Concentration of measure and isoperimetric inequalities in product spaces, Institut Des Hautes Études Scientifiques, Publications Mathématiques 81, 73 - 205.

65 Thomassen C. (1983): Disjoint cycles in digraphs, Combinatorica 3, 393 - 396.

66 Turán P. (1934): On a theorem of Hardy and Ramanujan, J. London Math Soc. 9, 274 - 276.

67 Vershik A. and Kerov C. (1977): Asymptotics for the Plancherel measure of the symmetric group and a limiting form for Young tableaux, Dokl. Akad. Nauk USSR, 233, 1024 - 1027.

68 Vizing V.G. (1968): Some unsolved problems in graph theory, Russian Math Surveys 23, 125 - 141.

69 Vizing V. (1976): Colouring the vertices of a graph with prescribed colours, Diskret. Analiz 29, 3 - 10.

70 Zykov A. (1949): On some properties of linear complexes, Mat. Sbornik N.S. 24, 163 - 188. English translation in Amer. Math. Soc. Transl. 79 (1952).

Probabilistic Analysis of Algorithms

Alan M. Frieze[1] and Bruce Reed[2]

[1] Department of Mathematical Sciences, Carnegie-Mellon University, Pittsburgh PA 15213, USA. E-mail: alan@random.math.cmu.edu

[2] Equipe Combinatoire, Univ. de Paris VI, 4 Place Jussieu, Paris 75005, France. E-mail: reed@ecp6.jussieu.fr

1. Introduction

Rather than analysing the worst case performance of algorithms, one can investigate their performance on typical instances of a given size. This is the approach we investigate in this paper. Of course, the first question we must answer is: what do we mean by a typical instance of a given size?

Sometimes, there is a natural answer to this question. For example, in developing an algorithm which is typically efficient for an NP-complete optimization problem on graphs, we might assume that an n vertex input is equally likely to be any of the $2^{\binom{n}{2}}$ labelled graphs with n vertices. This allows us to exploit any property which holds on almost all such graphs when developing the algorithm.

There is no such obvious choice of a typical input to an algorithm which sorts n numbers $x_1, \ldots, x_n$, for, e.g., it is not clear how big we want to permit them to become. One of many possible approaches is to impose the condition that each number is a random element of $[0,1]$, where each such element is equally likely. Another is to note that in analysing our algorithm, we may not need to know the values of the variables but simply their relative sizes. We can then perform our analysis assuming that the x_i are a random permutation of $y_1 < y_2 < \ldots < y_n$ with each permutation equally likely.

More generally, we will choose some probability distribution on the inputs of a given size and analyse the performance of our algorithm when applied to a random input drawn from this distribution. Now, in general, probability distributions are complicated objects which must be formally described and analysed using much messy measure theory. Fortunately, we will be concerned only with relatively simple distributions which will be much easier to deal with.

We often consider finite distributions in which our probability space is a finite set S, and for each $x \in S$ there is a p_x such that the $\sum_{x \in S} p_x = 1$ and the probability that the outcome is x is p_x. If all the p_x are the same then we are choosing a *uniform* member of S. For example, we discussed above choosing uniformly a random labelled graph on n vertices.

We may also consider choosing reals uniformly in $[a, b]$. Thus the probability our random real is between c and d for $a \le c \le d \le b$ is $\frac{d-c}{b-a}$.

Alternatively, we may consider analysing probability distributions by imposing conditions on the random objects chosen without specifying any further the underlying distribution. One example of such a distribution independent analysis was mentioned earlier when we suggested studying sorting under the assumption that all $n!$ permutations of n numbers are equally likely to be the input.

Finally, we may consider combining the above three possibilities. For example, we may consider a uniformly chosen graph on n vertices whose edges have been assigned uniform random weights from $[0,1]$, or a set S of random vectors in R^m where each vector consists of m independent uniform elements of $[0,1]$.

Focusing on these simple distributions allows us to dispense with the development of a rigorous measure theoretical foundation of probability theory. It is also quite natural.

One of our goals in this paper is to develop exact algorithms which work efficiently on the overwhelming majority of random inputs. A related goal is to try and find algorithms whose expected running time is small. We examine these approaches in Sections 2 and 3. A different technique is to consider algorithms which are guaranteed to run quickly but do not necessarily find the optimal solution, and show they are typically optimal, very close to optimal, or at least reasonably close to optimal. This is the approach taken in Sections 4 and 5.

Alternatively, we can show that an algorithm almost always behaves poorly on random instances. For example, we might prove that an algorithm almost always takes exponential time. This is a much more damning condemnation of its performance than the pathological examples constructed to provide lower bounds on worst-case complexity. We discuss this approach in Section 6. Finally, we note that how an algorithm performs on a random input depends heavily on the probability distribution we are using. In Section 7, we compare the analysis of various probability distributions for some specific problems.

We stress that we are interested in providing the reader with a gentle introduction to some of the most important topics in this area. Our survey is neither comprehensive nor up to date. Readers may turn to the survey articles [53],[80],[76] and the books [84],[99],[104] for more in-depth discussions of this area.

[1] Supported in part by NSF grant CCR9530974.

Finally we remark that from the third section on, the subsections are basically independent so a reader who lacks the necessary background for one may safely skip it.

1.1 Some Basic Notions

We begin with two simple but powerful probabilistic tools.

The First Moment Method/Markov Inequality. If X is a random non-negative integer valued variable then

$$\Pr(X>0) \le \mathbf{E}(X).$$

Proof. $\Pr(X>0) = \sum_{i=1}^{\infty}\Pr(X=i) \le \sum_{i=1}^{\infty} i\Pr(X=i) = \mathbf{E}(X)$. □

Moreover, $\mathbf{E}(X)$ is often easier to compute than $\Pr(X>0)$. If this is the case, then we may compute $\mathbf{E}(X)$ and use it as a bound on $\Pr(X>0)$. This technique is known as the First Moment Method.

The Chernoff Bound. Suppose X is the sum of n independent random variables each of which is 1 with probability p and 0 with probability $1-p$ (hence $\mathbf{E}(X)=pn$). Then

$$\Pr(|X-\mathbf{E}(X)|>a) < 2e^{-a^2/3pn}.$$

This is one of many inequalities which bound the extent to which a variable deviates from its expected value. Chapter 6 of this volume is dedicated to the study of such inequalities and contains a proof of the above result (obtained by combining Theorem 2.3 (b) and (c) of that chapter).

We recall that we use $BIN(n,p)$ to denote a random variable which is the sum of n random 0-1 variables each of which is 1 with probability p and 0 with probability $1-p$.

We say that a property defined in terms of n holds whp if it holds with probability $1-o(1)$ as $n \to \infty$.

By $G_{n,p}$ we mean a random graph with vertex set $V_n = \{1,\dots,n\}$ where each edge is present with probability p independently of the presence of the other edges. Thus, for each graph H with vertex set V_n and m edges the probability that $G_{n,p} = H$ is $p^m(1-p)^{\binom{n}{2}-m}$. In particular, $G_{n,\frac{1}{2}}$ is a uniformly chosen random graph with vertex set V_n.

We note that the expected number of edges in $G_{n,p}$ is $p\binom{n}{2}$. Further the Chernoff Bound can be used to show that unless $p = O(1/n^2)$, $|E(G_{n,p})|$ is whp $(1+o(1))p\binom{n}{2}$. Thus, if we analyze $G_{n,p}$, then typical graphs have about $p\binom{n}{2}$ edges. $G_{n,m}$ is the random graph on n vertices whose edge set $E_{n,m}$ is a uniformly chosen random set of m of the $\binom{n}{2}$ unordered pairs contained within $\{1,\dots,n\}$.

Finally we note that if we have an algorithm A for an optimization problem and we run it on a random instance I of size n drawn from some probability distribution, then the running time of this algorithm on this instance, $R_A(I)$, is a random variable which depends on I. We let its expected value be $r_{A,n}$. The expected running time of algorithm A with respect to the specified distribution is a function ER_A such that $ER_A(n) = r_{A,n}$.

2. Exact Algorithms for Hard Problems

NP-complete problems are natural candidates for probabilistic analysis, as the traditional worst-case approach has failed to provide efficient algorithms for such problems. In this section, we focus on two such problems, Edge Colouring, and Hamilton cycle. We shall also discuss Graph Isomorphism, another problem which although not known to be NP-complete, also is not known to be solvable in polynomial time. As we shall see, it makes little sense to speak of approximation algorithms for any of these problems, as they are essentially yes-no questions. Thus, the failure to find efficient algorithms to solve them means that from a traditional viewpoint we are completely at sea. Our first step is to find efficient algorithms which solve these problems whp on uniform random instances, we then present algorithms which have polynomial expected running time.

Some may criticise as unrealistic the assumption that a typical input is a uniformly chosen graph. However, this is no more unrealistic than the belief that studying the pathological examples constructed in NP-completeness proofs yields information about typical instances. Furthermore, a standard paradigm for constructing algorithms which run in polynomial time whp (though by no means the only one), is to provide an algorithm which works provided that the input graph has a certain structure and then prove that $G_{n,p}$ has the required structure whp. Such proofs are valuable because they add to our understanding of what it is that makes the problem difficult. For example, Arora's famous [3] approximation scheme for the Euclidean TSP [1] stemmed from Karp's analysis of the Euclidean TSP for random inputs which we present in Section 4.2.

2.1 Algorithms Which Almost Always Succeed

2.1.1 Hamilton Cycles. A Hamilton cycle in a graph G is one passing through all its vertices. Determining if a graph has a Hamilton cycle was one of the first six NP-complete problems reduced to SAT by Karp in his seminal paper [73]. In this section we show that $G_{n,\frac{1}{2}}$ have Hamilton cycles whp and present a polynomial-time algorithm which whp constructs such a cycle.

Definition. We call a graph tractable, if the following conditions hold

(i) every vertex has between $\frac{n}{2} - \frac{n}{40}$ and $\frac{n}{2} - \frac{n}{40}$ neighbours,

(ii) for every pair $\{u, v\}$ of vertices we have $\frac{n}{4} - \frac{n}{40} \le |N(u) \cap N(v)| \le \frac{n}{4} + \frac{n}{40}$,

(iii) for every triple $\{u, v, w\}$ of vertices we have

$$\frac{n}{8} - \frac{n}{50} \le |N(u) \cap N(v) \cap N(w)| \le \frac{n}{8} + \frac{n}{50}.$$

We need

Lemma 3.1. $G_{n,\frac{1}{2}}$ is tractable whp.

Proof. For each pair of vertices $\{u, v\}$ of $G_{n,\frac{1}{2}}$, $|N(u) \cap N(v) - u - v|$ is the sum of $n - 2$ independent random variables each of which is 1 with probability $\frac{1}{4}$ and 0 with probability $\frac{3}{4}$. Thus, applying the Chernoff Bound, we obtain that with probability at least $1 - 2e^{[illegible]}$, (ii) holds. Thus, (ii) holds whp. Similar techniques apply for (i) and (iii), we leave the details to the reader. ∎

We now present a polynomial time algorithm for constructing a hamilton cycle in a tractable graph, which by the above lemma works whp on $G_{n,\frac{1}{2}}$. The algorithm has three phases. While discussing it, we sometimes find it convenient to consider a path and its reverse.

Phase 1: Path Construction

Construct a path P by iteratively applying the following two rules, until this is no longer possible.

(i) If some vertex x not on P sees an endpoint a of P, add the edge ax to P.

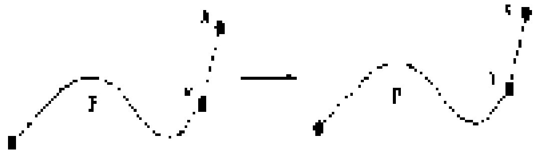


Fig. 2.1

(ii) If there are vertices $x \notin P$, $y, z \in P$ such that $P = aP'yzP''$ and $xy, az \in E(G)$ then replace P by the path $xyP'azP''$.

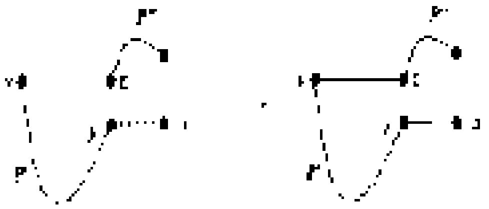


Fig. 2.2

We leave it as an exercise for the reader to show that, in a tractable graph, the final path has at least $\frac{n}{2} - \frac{n}{50}$ vertices.

Phase 2: Cycle Construction

Construct a path C by applying one of the following two rules.

(i) If there are vertices $x, y \in P$, such that $P = aP'xyP''b$ and $ay, xb \in E(G)$ then let C be the cycle $aP'xbP''ya$.

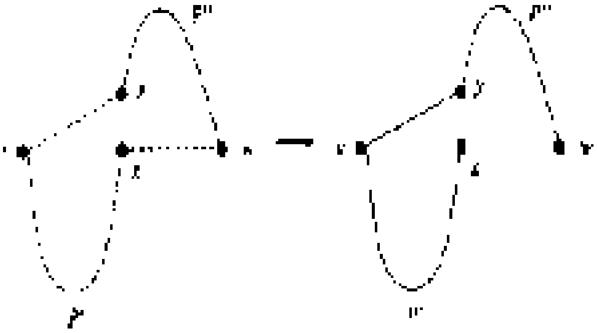


Fig. 2.3

(ii) If there are vertices $x, y \in P$ such that $P = aP'xyP''w$ and $ay, xw \in E(G)$ then let C be the cycle $aP'xwP''ya$.

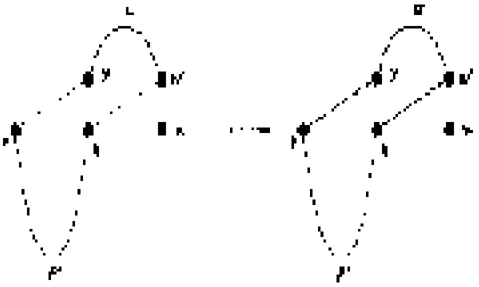


Fig. 2.4

We leave it as an exercise for the reader to show that in a tractable graph, this phase is always possible. We note that $|C| \geq \frac{n}{3} - \frac{n}{32}$.

Phase 3: Cycle Extension

We add the vertices of $V - C$ to C, one or two at a time, until $V(C) = V$, according to the following three rules

(i) If some vertex x not on C sees two consecutive vertices y and z of C then replace C by $C - yz + yx + xz$.

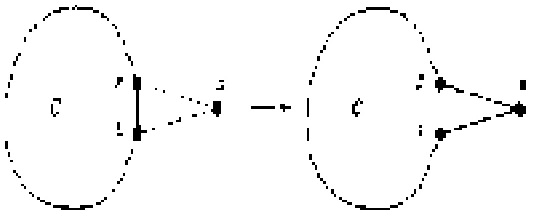


Fig. 2.5

(ii) If there are adjacent vertices $x, y \notin C$, and consecutive vertices u, v of C such that $ux, vy \in E(G)$ then replace C by the cycle $C - uv + ux + xy + yv$.

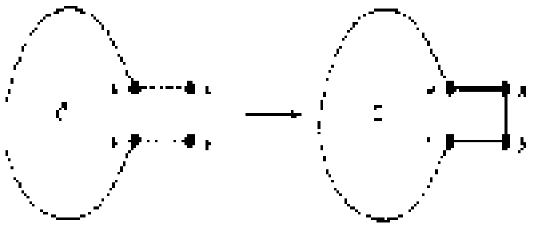


Fig. 2.6

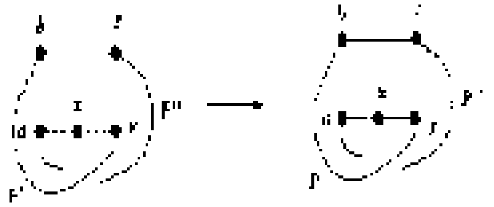


Fig. 2.7

(iii) If there are vertices $x \notin C$ and vertices $y, z, u, k \in C$ such that $C = ukP'yzP''u$ and $xu, xy, kz \in E(G)$ then replace C by the cycle $uxyP'kzP''u$.

We leave it as an exercise for the reader to show that in a tractable graph this step is always possible (that is, if $V - C$ is not a stable set (i.e. if there are any edges with both endpoints in this set) then we can apply (i) or (ii)).

It is easy to see that each phase of the algorithm can be implemented in $O(n^4)$ time and so is indeed a polynomial time algorithm as claimed.

Exercise: Show that the above algorithm can actually be implemented in $O(n^2)$ time on tractable graphs (which is linear in the number of edges).

2.1.2 Edge Colouring. An edge colouring of a graph G is an assignment of colours to its edges so that no two edges which share an endpoint receive the same colour, i.e. each colour class is a matching, that is, a graph all of whose vertices have degree at most one. Clearly, if a graph has maximum degree Δ then every edge colouring uses at least Δ colours. Vizing proved that every such graph has a $\Delta + 1$ colouring. So determining the chromatic index of a graph G, i.e. the minimum number of colours used in an edge colouring, boils down to determining if G has a Δ colouring. Vizing [109] also proved that if the maximum degree vertices of G form a stable set, then G has a Δ colouring. Berge and Fournier [10] developed a polynomial time algorithm for constructing a $\Delta + 1$ colouring of G. The algorithm provides a Δ colouring provided the vertices of maximum degree in G form a stable set. In contrast, Holyer [64] has shown that determining the chromatic index of a graph is NP complete.

In this section, we present the following result due to Erdős and Wilson [41].

Theorem 2.2. *$G_{n,\frac{1}{2}}$ has a unique vertex of maximum degree whp.*

Thus, we obtain:

Corollary 2.3. *Berge and Fournier's algorithm is a polynomial time algorithm which edge colours $G_{n,\frac{1}{2}}$ whp.*

Proof of Theorem 2.2. To prove the theorem, we need to analyse the probability distribution on the degrees of the vertices in $G_{n,\frac{1}{2}}$. Now, the degree of a vertex of $G_{n,\frac{1}{2}}$ is the sum of $n - 1$ variables each of which is 0 with probability $\frac{1}{2}$ and 1 with probability $\frac{1}{2}$. Thus, the expected degree of a vertex of $G_{n,\frac{1}{2}}$ is $\frac{n-1}{2}$ and

$$\Pr(d(v) = t) = \frac{\binom{n-1}{t}}{2^{n-1}}. \tag{2.1}$$

It follows easily (e.g. from the Chernoff Bounds) that if we let $t = t(n)$ be the smallest integer such that $\Pr(d(v) > t) < n^{-1}$ then provided n is large enough, $\frac{n}{2} \leq t \leq \frac{n}{2} + \sqrt{n \log n}$. Then using (2.1) we obtain

$$\Pr(d(v) \ge t) \ge \frac{1}{2}\Pr(d(v) \ge t-1) \ge 2n^{-3/4} \qquad (2.2)$$

Thus we expect at least $\frac{n^{1/4}}{2}$ vertices of $G_{n,\frac{1}{2}}$ to have degree greater than t. So, the following result, which we prove in the next section, is not surprising.

Whp there is a vertex of $G_{n,\frac{1}{2}}$ whose degree exceeds t. (2.3)

Now, a simple but tedious First Moment calculation using (2.1) will allow us to show:

Whp there is no $i > t$ such that two vertices of $G_{n,\frac{1}{2}}$ have degree i. (2.4)

Combining (2.3) with (2.4) yields the theorem. It remains only to prove (2.4).

To do so, we note that, by (2.1), for s between t and $t+\frac{\sqrt{n}}{8\sqrt{\log n}}$, we have:

$$\frac{\Pr(d(v)=s)}{\Pr(d(v)=t)} = \frac{t!(n-1-t)!}{s!(n-1-s)!} = 1-o(1).$$

Thus

$$\Pr(d(v) \ge t) \ge \sum_{i=t+1}^{t+\frac{\sqrt{n}}{8\sqrt{\log n}}} \Pr(d(v)=i) \ge \frac{\sqrt{n}}{16(\log n)^{1/2}}\Pr(d(v)=t).$$

So, we obtain that $\Pr(d(v)=t) = O(n^{-5/4}(\log n)^{1/2})$.

We can now bound the expected number of pairs of vertices N in $G_{n,\frac{1}{2}}$ both of which have the same degree i which exceeds t. Let $\hat{d}(v)$ denote the degree of v in $G_{n,\frac{1}{2}} - u$. Let $\hat{d}(u)$ denote the degree of u in $G_{n,\frac{1}{2}} - v$. Then

$$\Pr(d(u) = d(v) = i) \le \Pr(\hat{d}(v) \in \{i-1,i\})\Pr(\hat{d}(u) \in \{i-1,i\})$$

$$= \Pr(\hat{d}(u) \in \{i-1,i\})^2 \le \Pr(d(u) \in \{i-1,i,i+1\})^2$$

$$\le 9\Pr(d(u) = i-1)^2.$$

Hence,

$$E(N) \le 9\binom{n}{2}\sum_{i=t}^{n-1}(\Pr(d(u)=i))^2$$

$$\le 9\binom{n}{2}\sum_{i=t}^{t+8\sqrt{n\log n}-1}(\Pr(d(u)=i))^2 + 9\binom{n}{2}\sum_{i=t+8\sqrt{n\log n}}^{n-1}(\Pr(d(u)=i))^2$$

$$\le 36\binom{n}{2}\sqrt{n\log n}\,\Pr(d(v)=t)^2 + 9\binom{n}{2}\sum_{i=t+8\sqrt{n\log n}}^{n-1}(\Pr(d(u) \ge i))^2$$

Applying our bound on the probability that $d(u) = t$ to the first term and the Chernoff Bound to the second, we obtain

$$E(N) = O(n^{-1/10}(\log n)^{3/2}) + O(n^{-1}) = o(1).$$

Thus, the probability that for some $i \ge t$ there are two vertices of degree i is also $o(1)$, i.e. (2.4) holds. ∎

A similar but easier First Moment computation yields the following result which we state without proof as we need it later:

For $j \le \sqrt{n}$ the probability that there are j disjoint pairs of vertices $\{x_1,y_1\},\dots,\{x_j,y_j\}$ such that for some $d_1 > t$, $d_i = d(x_i) \le d(y_i) \le d_i - 4$ is $O(n^{-1/20})$. (2.5)

As we discuss in Section 2.2.3, Frieze, Jackson, McDiarmid and Reed [63] showed that the probability that $G_{n,\frac{1}{2}}$ does not have a Δ edge colouring is between $(n^{-c_1 n})$ and $(n^{-c_2 n})$ for some positive constants c_1 and c_2 (see [64]).

2.1.5 Graph Isomorphism. The input to the decision problem Graph Isomorphism is two graphs G_1 and G_2. The problem is to determine if there is an isomorphism between them. That is, a bijection f from $V(G_1)$ to $V(G_2)$ such that xy is an edge of G_1 if and only if $f(x)f(y)$ is an edge of G_2. This problem is neither known to be in P nor known to be NP-complete.

In a probabilistic analysis of Graph Isomorphism, we do not want to consider an input consisting of two random graphs as they will whp be obviously non isomorphic because, e.g., they have a different number of edges or different degree sequences. There are (at least) two ways of dealing with the problem. The first is to assume that the input consists of a graph G drawn from the uniform distribution on the n vertex graphs and a second graph H about which we have no information (the reader may wish to think of H as chosen by an adversary who has seen G). The second (more studied) approach is to consider canonical labelling algorithms. A canonical labelling algorithm assigns to a graph G on vertex set $\{1,\dots,n\}$ a permutation Π_G such that if two graphs G and H are isomorphic then $\Pi_H^{-1}\Pi_G$ is an isomorphism from G to H. That is a canonical labelling algorithm relabels graphs so that if two original graphs were isomorphic then the relabelled graphs coincide.

As an example, a canonical labelling algorithm might choose to order the vertices of the graph so that if $\Pi(i) < \Pi(j)$ then i is in more triangles than j. We note that if no two vertices of G are in the same number of triangles then there is a unique Π_G satisfying this condition. Furthermore, if H is isomorphic to G then there is a unique Π_H satisfying this condition and $\Pi_G(G)$ and $\Pi_H(H)$ are the same graph. Of course our canonical labelling algorithm must also have a way of dealing with graphs in which some pairs of vertices are in the same number of triangles.

We leave the reader to show that there is a canonical labelling algorithm that runs in $O(n^2 2^n)$ time. We also discuss canonical labelling algorithms which relabel some but not all graphs. In this case, if the algorithm relabels G it should also relabel all graphs isomorphic to G.

In this section we prove a result of Babai, Erdős and Selkow [6] (for strengthenings see Karp [73]).

Theorem 2.4. *There is a canonical labelling algorithm which labels $G_{n,\frac{1}{2}}$ whp.*

One such canonical labelling algorithm is to order the vertices in non-increasing order of degree and to order the vertices of the same degree so that vertices in more triangles come first. We shall not treat this algorithm here (however the reader is invited to show that it succeeds whp by showing that the expected number of pairs of vertices with the same degree and in the same number of triangles is $o(1)$). Instead, we treat an algorithm which orders the vertices in non-increasing order of degree but chooses the order in the set of vertices of the same degree in a slightly different way.

We need

Definition. We call a degree unique if there is precisely one vertex with this degree. We call a vertex solitary if it has unique degree.

Lemma 2.5. *Whp, the highest $\lceil 3\log n\rceil$ degrees of $G_{n,\frac{1}{2}}$ are unique and no two vertices have the same neighbourhood on the $\lceil 3\log n\rceil$ vertices of highest degree.*

Now, the canonical labelling algorithm we consider orders vertices of the same degree so that if $\pi(i) < \pi(j)$ then the highest degree vertex which sees exactly one of $\{i,j\}$ sees i but not j. Lemma 2.5 ensures that this algorithm succeeds whp. Thus the lemma implies the theorem. We prove the lemma below.

Proof of Lemma 2.5 Let $\ell = \lceil 3\log n\rceil$. The key to proving the lemma is to show:

Whp the $\ell+1$ highest degrees in $G_{n,\frac{1}{2}}$ are unique and the difference between two consecutive degrees is at least five. (2.6)

We prove this result below. Combining it with the following result proves the lemma.

The probability that the $\ell+1$ highest degrees in $G_{n,\frac{1}{2}}$ are unique and differ by at least five and two vertices have the same neighbourhood on the ℓ vertices of highest degree is $o(1)$. (2.7)

To prove (2.7), we compute the expected number of sets $v_1,\dots,v_\ell,u_1,u_2$ in $G_{n,\frac{1}{2}}$ such that (i) $v_1,\dots,v_\ell$ are solitary vertices with the highest degrees, the $\ell+1$ highest degrees all differ by at least five, and (ii) u_1 and u_2 have the same neighbourhood on $W = \{v_1,\dots,v_\ell\}$. We show that the expected number of such sets is $o(1)$ hence the probability one exists is $o(1)$ and (2.7) holds.

Now, there are $\binom{n}{\ell}\binom{n-\ell}{2}$ choices for W, u_1, u_2. For each choice, we determine the edges of $G_1 = G_{n,\frac{1}{2}} - u_1 - u_2$. That is, we take a copy of $G_{n-2,\frac{1}{2}}$ with vertex set $V - u_1 - u_2$. If the ℓ vertices of highest degree in G_1 are not distinct then (i) cannot hold, for adding u_1 and u_2 changes each degree by at most two and the difference between two degrees by at most four. If the ℓ vertices of highest degree in this graph are unique, then for (i) to hold the vertices with these degrees must be those in W which by symmetry occurs with probability $\binom{n-2}{\ell}^{-1}$. Given that W is the set of high degree vertices in this graph we see, by considering the edges from u_1 and u_2, that the probability that (ii) holds is $2^{-\ell} \le \frac{1}{n^3}$. Thus the expected number of W, u_1, u_2 such that (i) and (ii) holds is $\binom{n}{\ell}\binom{n-\ell}{2}\binom{n-2}{\ell}^{-1}n^{-3} = o(1)$. So, (2.7) holds as claimed, we turn now to (2.6).

To prove (2.6), we consider the $c (= c(\epsilon))$ defined in our discussion of edge-colouring. As promised in that discussion, we will show that whp $G_{n,\frac{1}{2}}$ has a vertex of degree greater than $\frac{n}{2}$. In fact, we will prove that whp it has at least $l+1$ such vertices, which combined with (2.5) for $j=1$, proves (2.6). We actually prove a much stronger result which we will need later, to wit:

The probability that there are fewer than $n/7$ vertices of degree greater than t is $O(e^{-n^{1/5}})$. (2.9)

To prove this result, we use "the method of deferred decisions" as described in Knuth, Motwani and Pittel [illegible]. Imagine that we have an assistant and when we want to know whether an edge uv exists, he flips a fair coin and if it comes down heads the edge exists, otherwise it does not. We only do this at most once for each possible pair u,v. The order in which we flip the edges is as described in the following procedure:

(1) Set $i=1$, choose some vertex v_1. Determine which edges incident to v_1 are present.

(2) If $i=n-1$ stop, otherwise choose the vertex v_{i+1} in $V-\{v_1,\ldots,v_i\}$ which has the most neighbours in $V_i=\{v_1,\ldots,v_i\}$ and determine which edges between v_{i+1} and $V-V_i-v_{i+1}$ are present.

(3) Increment i and return to Step 2.

By analyzing this procedure, we can show:

The probability that there is some $i<\frac{n}{2}$ such that v_{i+1} has fewer than $\frac{i}{2}-\sqrt{n}$ neighbours in V_i is $O(2^{-n^{1/2}})$. (2.8)

Proof. By our choice of v_{i+1}, if this occurs, then there are fewer than $\frac{i(n-i)}{2}-(n-i)\sqrt{n}$ edges between V_i and $V-V_i$. However, we expect $\frac{i(n-i)}{2}$ edges between the two sets. Using the Chernoff Bound, it is easy to show that expected number of sets S of $i<\frac{n}{2}$ vertices such that there are fewer than $\frac{i(n-i)}{2}-(n-i)\sqrt{n}$ edges between S and $V-S$ is $O(2^{-n^{1/2}})$ (we leave the details to the interested reader). The result follows. □

The probability that there are fewer than $n/8$ values of i which are less than $\frac{n}{2}$ such that v_{i+1} has more than $\frac{n-i}{2}+(1-\frac{i}{n})\sqrt{n}$ neighbours in $V-V_i-v_{i+1}$ is $O(2^{-n^{1/2}})$. (2.10)

Proof. For $i<\frac{n}{2}$ let E_i be the event that v_{i+1} has more than $\frac{n-i}{2}+(1-\frac{i}{n})\sqrt{n}$ neighbours in $V-V_i-v_{i+1}$. In the first i iterations, we flip coins only for edges from V_i. Thus, after we choose v_{i+1}, the coins for the edges from v_{i+1} to $V-V_i-v_{i+1}$, which determine if E_i are yet to be flipped, and in fact are those flipped in the next iteration. It follows that for distinct i and j, E_i and E_j are independent for they are determined by disjoint sets of edges (the coins for which are flipped in different iterations of our procedure for generating $G_{n,1/2}$). Furthermore, by the Chernoff Bound, the probability of the event E_i is close to e^{-2} and is certainly greater than $p=e^{-3}$. Applying the Chernoff Bound once more, we obtain that the number of i for which E_i holds is less than $\frac{pn}{4}$ with a probability which is $O(2^{-n^{1/2}})$.

Combining (2.8) and (2.10) yields (2.9) thereby completing the proof of the lemma. □

We close this section by remarking that combining (2.9) and (2.4) yields the following result, which we shall find useful:

The probability that there are fewer than $\frac{pn}{8}$ solitary vertices of G with degree greater than t is $O(2^{-n^{1/2}})$. (2.11)

2.2 Polynomial Expected Time

2.2.1 Graph Isomorphism. We now present a polynomial expected time algorithm for graph isomorphism. The input to the algorithm is a graph G drawn from uniform distribution on n-vertex graphs and a graph H about which we have no information.

As a last resort, our algorithm uses the brute force $O(n^2 n!)$ procedure of testing each of the $n!$ bijections between $V(G)$ and $V(H)$.

Our algorithm also uses two sub-algorithms both of which are reminiscent of the canonical labeling procedure in the last section. In the canonical labeling procedure, we essentially knew the bijection on some subset S of V (the high degree solitary vertices) and this allowed us to determine the rest of the bijection simply by considering $N(v)\cap S$ for each $v\in V-S$.

To ease our discussion of extending partial bijections in this manner, we need some definitions. Let $S\subseteq V(G)$; we say a vertex v in $V-S$ is determined by S if there is no $w\in V-S$ with $N(w)\cap S=N(v)\cap S$. We let det(S) be the set of vertices determined by S. We need the following deterministic result:

Lemma 2.6. *If $S\subseteq V(G)$ and f is a bijection from S to some subset of $V(H)$, then for any isomorphism f' extending f and for any $v\in$ det(S), we have only one candidate for $f'(v)$ and in $O(n^2)$ time, we can either*

(a) determine that there is no isomorphism from G to H extending f, or

(b) find a bijection g from det$(S)\cup S$ to a subset of $V(H)$ such that any isomorphism f' extending f corresponds with g on det$(S)\cup S$.

Proof. We leave this as an exercise for the reader. □

We need to take this idea one step further. To this end, we say a vertex v in $V - S$ is *fixed by* S if $v \in \det(S) \cup \det(\det(S))$. We let $fix(S)$ be the set of vertices fixed by S. Applying Lemma 2.6 twice, we obtain

Lemma 2.7. *If $S \subseteq V(G)$ and f is a bijection from S to some subset of $V(H)$, then for any isomorphism f' extending f and for any $v \in fix(S)$, we have only one candidate for $f'(v)$, and in $O(n^2)$ time, we can either*

(i) *determine that there is no isomorphism from G to H extending f, or*

(ii) *find a bijection g from $fix(S) - S$ to a subset of $V(H)$ such that any isomorphism f' extending f corresponds with g on $fix(S) \cup S$.*

The probabilistic results we need are

Lemma 2.8. *With probability $1 - O(2^{-n^{1/3}})$, the solitary vertices fix V.*

Lemma 2.9. *With probability $1 - O(2^{-2n\log n})$, every set S of $\lceil 20\log n\rceil$ vertices fixes all but at most $\lceil 20\log n\rceil$ vertices of G.*

We prove these results in a moment. First, we show that they imply the existence of the desired polynomial expected time algorithm.

We will use an algorithm A_1 which computes the degree sequence of G and H, chooses the characteristic sets S to be the set of solitary vertices of G and S' to be the set of solitary vertices of H, and lets f be the bijection from S to S' such that $d_G(v) = d_H(f(v))$. It then determines if S fixes $V(G)$. If not it halts. Otherwise, applying the algorithm of Lemma 2.7, it either determines and outputs that G is not isomorphic to H or extends f to a bijection g from $V(G)$ to $V(H)$ such that the only possible isomorphism from G to H is g. If it returns such a bijection g, it then checks whether or not g is in fact an isomorphism. If so, it outputs this isomorphism, otherwise it outputs the fact that G and H are not isomorphic. By Lemma 2.7, the answer returned by the algorithm is correct. By Lemma 2.8, the probability that A_1 does not give an answer is $O(2^{-n^{1/3}})$. It is straightforward to verify that the algorithm can be implemented in $O(n^2)$ time.

We will also use an algorithm A_2 which first chooses an arbitrary set S of $\lceil 20\log n\rceil$ vertices of G. The algorithm then checks if S fixes all but at most $\lceil 20\log n\rceil$ vertices of G. If not it halts. The algorithm next determines for each set S' of $|S|$ vertices of H and bijection f from S to S' whether or not there is isomorphism extending f. If it finds for some S' and f that there is an isomorphism extending f, it returns with the information that G and H are isomorphic. If it determines that for each S' and f there is no isomorphism extending f then it outputs that G and H are not isomorphic.

For a given S' and f, applying the procedure of Lemma 2.7, A_2 either determines and outputs that no isomorphism from G to H extends f or extends f to a bijection g from $fix(S) \cup S$ to a subset of $V(H)$ such that the only possible isomorphisms from G to H extending f also extend g. If it returns such a bijection g, it then checks whether or not any of the at most $|V - fix(S) - S|! \le \lceil 20\log n\rceil!$ extensions of g to bijections from $V(G)$ to $V(H)$ are isomorphisms. If any of these are isomorphisms, the algorithm returns that there is an isomorphism extending f, otherwise it returns that no such isomorphism exists. By Lemma 2.7 the answer returned by the algorithm is correct. By Lemma 2.9, the probability that A_2 does not give an answer is $O(2^{-2n\log n})$. It is straightforward to show that the algorithm can be implemented so that it spends $O(n^2\lceil 20\log n\rceil!)$ time on each pair (S', f) and hence uses at most $O(n^{20\log n} n^2\lceil 20\log n\rceil!) = O(n^{40\log n})$ time in total.

Now, our global algorithm applies A_1, then applies A_2 if A_1 terminates without a response, and finally applies our brute force algorithm if A_2 fails to provide an answer. By the above remarks, the expected running time of this algorithm is $O(n^2) + O(2^{-n^{1/3}}n^{40\log n}) + O(2^{-2n\log n}n!n) = O(n^2)$. Since a random graph has $\Theta(n^2)$ edges clearly the algorithm has optimal expected running time. We can actually obtain a canonical labelling algorithm whose expected running time is $O(n^2)$ using similar techniques (see Babai and Kučera [9] for a result in this vein).

With our description of the algorithm complete, it remains only to prove our two probabilistic lemmas.

We need the following auxiliary results, all of which can be proven using simple First Moment calculations:

The probability that there is a set S of $\lceil 20\log n\rceil$ vertices which determines fewer than $\frac{9n}{10}$ vertices is $O(2^{-2n\log n})$. (2.12)

The probability that there is a set S of $\frac{9n}{10}$ vertices which determines fewer than $\frac{n}{10} - 20\log n$ vertices is $O(2^{-2n\log n})$. (2.13)

The probability that there is a set S of $\frac{9n}{10}$ vertices which does not determine $V - S$ is $o(2^{-n^{1/3}})$. (2.14)

Now, Lemma 2.9 follows from (2.12) and (2.13). Lemma 2.8 follows from (2.12) and (2.14), and (2.11).

2.3.2 Hamilton Cycles. We now present an algorithm **DENSEHAM** for Hamilton cycle that has expected running time which is $O(n^4)$. The algorithm uses two subalgorithms. One, A_1, solves Hamilton cycle on any graph in $O(n^2 2^n)$ time and actually finds the cycle if it exists. It is the Dynamic Programming algorithm of Held and Karp [44]. The other, A_2 runs in $O(n^4)$ time. It attempts to construct a Hamilton cycle in the input graph. The probability that it fails to return a Hamilton cycle when applied to $G_{n,\frac{1}{2}}$ is $O(2^{-n}n^2)$. DENSEHAM first applies A_1 and then applies A_2 if A_1 fails to find a Hamilton cycle. Clearly, DENSEHAM does indeed solve Hamilton Cycle, and in fact outputs a Hamilton cycle if one exists. Furthermore, its expected running time is $O(n^4)+O(2^{-n}n^2)O(2^n n^2) = O(n^4)$, as claimed. It remains only to describe and analyze A_1 and A_2.

A_2 is a simple dynamic programming algorithm which determines, for each subset S of V with $|S| \geq 2$, and for each pair of vertices $\{u, v\}$ of S, whether or not there is a Hamilton path through S with endpoints u and v. To determine if G has a Hamilton cycle we need then only check if for any edge uv of G there is a Hamilton path through $S = V$ with endpoints u and v. A_2 considers the subsets of V in increasing order of size. To determine if there is a Hamilton path of S with endpoints u and v, it simply checks whether there is some neighbour w of u in S such that there is a Hamilton path of $S-u$ with endpoints w and v. Since the algorithm has already considered $S-u$, this can be done via a simple table lookup. We spend $O(n)$ time on each triple S, u, v so the the claimed running time bound on A_2 holds. With a little extra bookkeeping we can also construct the Hamilton cycle; we omit the details.

A_1 is reminiscent of the algorithm for Hamilton Cycle presented in the last section. We will show:

Lemma 2.10. *Let G be a sufficiently large graph such that*

(i) there exists a set S of at most 12000 vertices such that $G - S$ is tractable,

(ii) the minimum degree of G is at least 2, and

(iii) at most one vertex of G has degree less than 4000.

Then G has a Hamilton cycle. Furthermore, given S we can find the Hamilton cycle in $O(n^4)$ time.

We will also show that the probability that $G_{n,\frac{1}{2}}$ satisfies conditions (i)-(iii) of Lemma 2.10 is $1-O(\frac{n^2}{2^n})$. Actually we will prove a slightly stronger result which permits us to use a greedy procedure for finding S.

Definition. A *bad sequence of length l* is a sequence $\{X_1, \ldots, X_l\}$ of disjoint subsets of G such that letting $G^i = G - \cup_{j \leq i} X_j$, we have that for each i between 0 and $l-1$, either

(a) X_{i+1} is a vertex v such that $|d_{G^i}(v) - \frac{|V(G^i)|}{2}| > \frac{|V(G^i)|}{50}$,

(b) X_{i+1} is a pair uv such that $||N_{G^i}(u) \cap N_{G^i}(v)| - \frac{|V(G^i)|}{4}| > \frac{|V(G^i)|}{50}$,

(c) X_{i+1} is a triple u, v, w such that $||N_{G^i}(u) \cap N_{G^i}(v) \cap N_{G^i}(w)| - \frac{|V(G^i)|}{8}| > \frac{|V(G^i)|}{50}$.

Lemma 2.11. *With probability $1 - O(\frac{n^2}{2^n})$, $G_{n,\frac{1}{2}}$ has minimum degree 2, has at most one vertex of degree less than 4000, and has no bad sequence of length 4000.*

Now algorithm A_1 proceeds as follows. It first ensures that G has minimum degree at least two and at most one vertex of degree less than 4000. If this is not true, the algorithm terminates with no output. Otherwise, it generates a maximal bad sequence $\{X_1, \ldots, X_l\}$ of length at most 4000 (i.e. the sequence either has length 4000 or cannot be extended). This can be done in $O(n^4)$ time because having found $\{X_1, \ldots, X_i\}$ we can search for X_{i+1} simply by checking whether any of the $\binom{n}{3} + \binom{n}{2} + n$ sets of size at most 3 in G^i satisfy one of conditions (a)-(c) in the definition of bad sequence. If the bad sequence A_1 finds has length 4000, it terminates without output. Otherwise, it sets $S = \cup_{i=1}^{l} X_i$ and applies the algorithm of Lemma 2.10 to construct a Hamilton cycle in G in $O(n^4)$ time (we note that $G - S$ is tractable by the maximality of the bad sequence). By Lemma 2.11, the probability that A_1 fails to return a Hamilton cycle is $O(\frac{n^2}{2^n})$ as claimed. This completes our description of A_1 and *DENSEHAM*. It remains only to prove the two lemmas.

Proof of Lemma 2.11. The probability that a vertex v of $G_{n,\frac{1}{2}}$ has degree 0 or 1 is $\frac{n}{2^{n-1}}$. Thus, the probability that the minimum degree of $G_{n,\frac{1}{2}}$ is 0 or 1 is $O(\frac{n^2}{2^n})$. The probability that there are two vertices of $G_{n,\frac{1}{2}}$ of degree less than 4000 is $O(\binom{n}{2}(\frac{n^{4000}}{2^n})^2) = o(2^{-n})$.

Finally, the probability that some $\{X_1, \ldots, X_{4000}\}$ is a bad sequence is, by an application of the Chernoff Bound, $O(e^{-[illegible]})$. Hence, the expected number of bad sequences of length 4000 is $o(2^{-n})$. The result follows. ∎

Proof of Lemma 2.10. The key to the proof is the following auxiliary result.

> Let H be a graph which is the union of a tractable graph G and a matching $M \subseteq G$ with fewer than 12000 edges. Then provided H is sufficiently large it has a Hamilton cycle C such that $M \subseteq E(C)$. Furthermore, we can find such a Hamilton cycle in $O(n^4)$ time. (2.16)

Proof. The first step in the proof of (2.16) is to find a path Q in H with $M \subseteq E(Q)$ and such that Q has at most $3|M|$ edges. This can be done greedily because every two vertices of G have more than $\frac{n}{5}$ common neighbours. We then apply Phases 1-3 of the algorithm for constructing a Hamilton cycle presented in the last section initializing with $P = Q$, and ensuring that we never delete an edge of Q from the path or cycle we create (this is possible because Q has only a bounded number of edges; we note that in Phase 2 we will be able to use an endpoint of P which is not in Q). □

We turn now to the proof of Lemma 2.10. We enumerate S as $s_1, \ldots, s_k$ (with $k \le 1000$) so that s_1 is the lowest degree vertex of S. We first consider the case in which s_1 has exactly one neighbour x in $V - S$. In this case, we know that s_1 must have a neighbour in S, w.l.o.g. s_2. Since for $i > 1$, s_i has at least 4000 neighbours, we can find distinct vertices $y_2, \ldots, y_k, z_2, \ldots, z_k$ of $V - S$ such that for $i \ge 2$ $s_i y_i, s_i z_i \in E(G)$, $z_2 = x$, and $y_2 z_2 \in E(G)$. We set $M = \{y_2 z_2, \ldots, y_k z_k\}$ and apply the algorithm of (2.15) to $H = (G - S) \cup M$. We let C be the output Hamilton cycle in H with $M \subseteq E(H)$. We let C' be the Hamilton cycle in G with edge set $E(C) - M \cup (\bigcup_{i \ge 2}\{s_i y_i, s_i z_i\}) \cup \{s_2 s_1, s_1 x\}$.

The cases in which s_1 has 0 or more than 1 neighbours in $V - S$ are similar, we omit the details. □

Exercise: Combine this algorithm with our earlier algorithm to develop an algorithm for Hamilton cycle whose expected running time on $G_{n,\frac{1}{2}}$ is in $O(n^2)$ time (and hence is linear in the size of the input).

2.2.8 Edge Colouring. Perkovic and Reed [95] recently developed a polynomial expected time algorithm for edge colouring. Their algorithm is much too complicated to explain in detail here. The complexity is due to the fact that the fastest known edge colouring algorithm which succeeds on all graphs has a worst-case running time bound which is $O(3^{n^2})$ on n vertex graphs for some $c > 0$. We will briefly outline their algorithm, to do so we need a few auxiliary results.

We use $\Delta(G)$ for the maximum degree in G.

Definition. *H is an l-reduction of G if $\Delta(H) = \Delta(G) - l$ and there exist matchings $M_1, \ldots, M_l$ in G such that $H = G - \bigcup_{i=1}^{l} M_i$. H is a reduction of G if it is an l-reduction for some l.*

Remark: *If a reduction H of G has a $\Delta(H)$ edge colouring then G has a $\Delta(G)$ edge colouring.*

Definition. *A subgraph H of G is overfull if $|V(H)|$ is odd and $E(H) > \Delta(G)\frac{|V(H)|-1}{2}$.*

Fact. *If G contains an overfull subgraph then it has no Δ edge colouring.*

Proof. If H has $2k + 1$ vertices then the largest matching in H has k edges. □

Theorem 2.12. [Padberg and Rao] [94] *There is a polynomial time algorithm which determines if G has an overfull subgraph.*

Theorem 2.13. [62] *The probability that $G_{n,\frac{1}{2}}$ has a reduction H whose vertices of maximum degree form a stable set is $1 - O(2^{-n^{c_1}})$ for some $c_1 > 0$. Furthermore, there is a polynomial time algorithm which finds such a reduction and corresponding matchings $M_1, \ldots, M_l$ with this probability.*

Corollary 2.14. *There is a polynomial time algorithm which Δ edge colours $G_{n,\frac{1}{2}}$ with probability $1 - O(2^{-n^{c_1}})$ for some $c_1 > 0$.*

Proof. We attempt to find a reduction H of G whose vertices form a stable set using the algorithm of the theorem. If we succeed, we apply Berge and Fournier's algorithm to edge colour H and then use the matchings $M_1, \ldots, M_l$ to colour the remaining edges of G. □

As an aside, we mention the following complementary result:

Theorem 2.15. [87] *There exists a $c_2 > 0$ such that for $n > 3$, the probability that $G_{n,\frac{1}{2}}$ has an overfull subgraph is at least $n^{-c_2 n}$.*

Definition. A graph is *bipartite* if it can be partitioned into two stable sets. A graph G is *near-bipartite* if for some vertex v, $G - v$ is bipartite.

Theorem 2.16. [97] *A near bipartite graph G is Δ edge colourable if and only if it contains no overfull subgraph. Furthermore, there is a polynomial time algorithm which given a near-bipartite graph either finds an overfull subgraph or a Δ edge colouring.*

Perkovic and Reed's algorithm first applies the polynomial time algorithm of Corollary 2.14, which fails with probability $O(2^{-n^{c_1}})$ for some constant c_1. They then apply the algorithm of Theorem 2.12 to determine if the input graph has an overfull subgraph. If it does they use the algorithm of Berge and Fournier to optimally $\Delta + 1$ colour it. There are two more algorithms which might be applied. The first, $Colour_1$, runs in $O(2^n)$ time and attempts to find a Δ edge colouring of a graph with no overfull subgraph. It fails with probability $O(2^{-n^{c}})$ for some c. The second $Colour_2$ is a dynamic programming algorithm which optimally colours every graph and has running time which is smaller than the inverse of the probability that $Colour_1$ fails. It follows that applying the four algorithms in the given order yields a polynomial expected time algorithm. We omit the description of $Colour_2$. $Colour_1$ more or less finds a near-bipartite reduction H of the input graph and applies the algorithm of Theorem 2.16 to find a $\Delta(H)$ edge colouring of H. Actually, the algorithm finds a reduction of a graph which is derived from the input graph and may have multiple edges. We omit any further description.

2.3 Further Results

Hamilton Cycles for Sparse Graphs. As we have seen, finding a Hamilton cycle in a dense graph is relatively easy. The analysis for sparse graphs is more intricate but still based on the two procedures used in Phase 1 of our algorithm for tractable graphs. That is, extension of the path by adding a neighbour of an endpoint, and rotation of the path $P = uP'yP''$ to obtain $P'yP''$. By iteratively applying rotations before extending, Bollobás, Fenner and Frieze [16] develop a polynomial time algorithm HAM with the property that for all $m = m(n)$

$$\lim_{n\to\infty} \Pr(\text{HAM finds a Hamilton cycle}) = \lim_{n\to\infty} \Pr(G_{n,m} \text{ is Hamiltonian}).$$

Frieze [40] proved a similar result for random digraphs.

Research Problem: Develop an algorithm which runs in polynomial expected time on $G_{n,m}$ for every m.

Graph Colouring. As we shall see in Section 6.1, there is no known polynomial time algorithm which optimally vertex colours $G_{n,p}$ with high probability. There has been some success in designing algorithms that whp optimally vertex colour randomly generated k-colourable graphs, for small k. The strongest current results stem from the spectral approach of Alon and Kahale [7]. Chen and Frieze [26] used this approach to colour random hypergraphs. The backtracking algorithm of Dyer and Frieze [33] optimally colours in polynomial expected time.

Min Bisection. We are given a graph G and asked to divide the vertices into two sets of equal size so as to minimise the number of edges between them. Most analysis has been concerned with the case where there is a fixed planted bisection with many fewer edges than expected. Bui, Chaudhuri, Leighton and Sipser [24] considered random regular graphs and showed how to find the planted cut in polynomial time whp. Dyer and Frieze [38] did the same for $G_{n,p}$ p constant. The strongest results on this problem have been obtained by Boppana [17] using spectral techniques. Jerrum and Sorkin [67] analysed a version of simulated annealing on $G_{n,p}$.

3. Faster Algorithms for Easy Problems

In this section, we discuss the probabilistic analysis of algorithms for which polynomial time algorithms are known to exist. Typically, we analyse a simple algorithm for the problem and show that its expected running time is much better than its worst-case running time. Our three representative examples, shortest paths, matchings and linear programming, are the foundations on which the field of combinatorial optimisation is built.

3.1 Perfect Matchings

Recall that a matching is a set of edges no two of which are incident. A vertex v is covered by a matching M if it is in an edge of M, otherwise it is uncovered. A matching is *perfect* if it covers all the vertices. The fastest algorithm for determining if a graph with n vertices and m edges has a perfect matching has a worst-case running time of $O(n^{1/2}m)$ [80]. In this section we describe an algorithm which runs in linear expected time on $G_{n,1/2}$, n even. There are two phases. Phase 1 greedily chooses edges and finds a matching of size $n/2 - O(\log n)$ whp. Phase 2 uses augmenting paths of length 3 (that is repeatedly replaces an edge xy of the matching by two edges ux and yz where u and z were previously uncovered) to produce a perfect matching whp.

Recall that $V(G_{n,p}) = \{1,\ldots,n\}$.

Phase 1

In this procedure S will denote the vertices not covered by the matching M produced so far.

In iteration i, we choose the minimum x_i of S and find the smallest numbered vertex y_i it can be matched to (i.e. the smallest y_i which is still uncovered and is adjacent to x_i). If there is no such $y_i \in S$ we terminate Phase 1, else we add x_iy_i to M and repeat.

Suppose Phase 1 produces $M = \{x_1y_1, x_2y_2, \ldots, x_ty_t\}$ and that M leaves $Z = \{z_1, z_2, \ldots, z_{2q}\}$, $q = \frac{1}{2}n - t$ unmatched. Note that for each i, $x_i < y_i$. We set $X = \{x_1, \ldots, x_t\}$. We set $s' = \max X$.

Phase 2

In this phase we take the members of Z in pairs z_{2i-1}, z_{2i}, $i = 1, 2, \ldots, q$ and try to find x_jy_j such that $z_{2i-1}x_j$ and $z_{2i}y_j$ are both edges. In which case we delete edge x_jy_j from M and add the edges $z_{2i-1}x_j, z_{2i}y_j$. For each i we go sequentially through values of j, starting the ith search at $x_{i,1}$. If we fail for some i then the whole algorithm fails.

We now discuss the probability that we fail to find a perfect matching in $G_{n,1/2}$ this way. Our analysis fits the notion of "the method of deferred decisions" described in Section 2.1.1.

First consider Phase 1. We claim that in this phase we need only examine the presence of each edge once. To see this note that in iteration i, we only examine edges from x_i to $S - x_i$. But any edge examined in a previous iteration has an endpoint x_j with $j < i$ and x_j is no longer in S: the claim follows. Furthermore, if we fix the outcome of an edge xv incident to some vertex v in this iteration and find it exists then we add xv to M and will flip no more coins for edges incident to v in this Phase. Thus if we test for the presence of t edges incident to v and find none of them exist then these must be the first t edges incident to v examined and so this occurs with probability $\left(\frac{1}{2}\right)^t$. Fu

$\xi \in Z \cup X$ and $K > 0$ we define the event

$$\mathcal{E}_\xi = \{|\{i : x_i < \xi < y_i\}| \geq K n \log_2 n\}.$$

Then we have

1. $\mathbf{Pr}\left(\bigcup_{\xi \in Z \cup X} \mathcal{E}_\xi\right) \leq n^{1-K}$

Proof. For for each i with $x_i < \xi < y_i$, we failed to find the edge $x_i\xi$. □

2. $\mathbf{Pr}(\exists i \leq p : y_i - x_i \geq 2K \log_2 n) \leq 2n^{1-K}$

Proof. For each such i either $\mathcal{E}_{x_i}$ occurs or the first $K \log_2 n$ edges examined in the ith iteration are not present. □

3. $\mathbf{Pr}(z^* \leq n - 2K \log_2 n) \leq 2n^{1-K}$

Proof. If this occurs then either $\mathcal{E}_{z^*}$ occurs or the first $K \log_2 n$ edges examined in the final iteration are not present. □

Assume now that none of the events described in 1,2,3 above occur and consider Phase 2. We observe that for any edge $x_i y_i$ of M we have not flipped the coin for the edges $x_i y_k$ for $k \geq y_i$, as if $y_i < z^*$ we have not flipped the coin for the edges $y_i z$ for any $z \in Z$. Since $y_i < 2n$, it follows from 2 and 3 that we have not flipped the coins for $x_i z$ or $y_i z$ where $z \in Z$ and $i \leq n/3$. So when we search for an alternating path of length 3 for the pair z_1, z_2, the probability that we need $3K \log_2 n$ attempts is $\left(\frac{3}{4}\right)^{3K \log_2 n} = o(n^{-K})$. Similarly the probability that when patching z_{2i-1}, z_{2i} we need to examine more than $3K \log_2 n$ pairs $\{x_i, y_i\}$ is $o(n^{-K})$. Thus Phase 2 fails with (conditional) probability $o(n^{-K} K \log_2 n)$.

In summary, this algorithm finds a perfect matching with probability at least $1 - O(n^{-K})$ after flipping at most $3Kn \log_2 n$ coins.

9.2 Linear Programming

It was observed early on that the simplex algorithm and its variants worked remarkably well in practice. A theoretical explanation was sought for this through probabilistic analysis, especially as Klee and Minty [83] had shown that a standard variant did not run in worst-case polynomial time.

The first average-case results were due to Borgwardt [18] and Smale [101, 102]. The model chosen in [18] is not the most obvious and [101, 102] requires that the number of constraints be small. Blair [12] later gave a simplified explanation for the results of [101, 102] — see Section 9.2.1. Further work on this problem came through another type of probabilistic model where randomness is introduced through a random choice of $\leq$ or $\geq$ for a particular constraint. See Haimovich [61], Adler and Megiddo [2], Adler, Karp and Shamir [1] and Adler, Megiddo and Todd [3]. A recent book by Borgwardt [19] covers the subject in detail.

There are still outstanding questions in this area. For example, can one find a reasonable model plus a proof that the algorithm which always chooses a variable of largest reduced cost to enter the basis runs in polynomial expected time.

9.2.1 Blair's Analysis. In this section we prove a simple result based on the ideas of Blair [12]. The result given here is not as strong but has a much simpler analysis.

In Blair's model we have a linear program

$$\begin{aligned} &\text{Maximize } cx \\ &\text{Subject to } Ax \leq b \\ &\qquad\qquad\quad x \geq 0 \end{aligned}$$

Here A is an $(m-1) \times n$ matrix.

We use the following notation. For a matrix M, $M_{(i)}$ denotes the ith row and $M^{(j)}$ denotes the jth column.

It is assumed that b is non-negative but arbitrary ($x=0$ is a feasible solution) and A, c are produced as follows: let $\hat{A} = \begin{bmatrix} c \\ A \end{bmatrix}$ have rows indexed by $\{0, 1, \ldots, m-1\}$. We have an $m \times n$ matrix B in which no two elements in the same row are the same. $\hat{A}_{(i)}$ is an independent random permutation of the corresponding row $B_{(i)}$.

Column $\hat{A}^{(j)}$ dominates column $\hat{A}^{(k)}$ if $\hat{A}(i,j) > \hat{A}(i,k)$ for $i = 0, 1, \ldots, m-1$. It is easy to see that an optimal solution will have $x_k = 0$ if $\hat{A}^{(k)}$ is dominated by some other column.

Several versions of the simplex algorithm have the following property: No variable corresponding to a dominated column of $\hat{A}$ enters the basis at any iteration.

As examples:

- Try to choose a surplus variable to enter, otherwise choose the entering variable with the largest reduced cost.
- Delete dominated columns at the start.
- The path following algorithms of [101, 102]

So if we let L be the number of undominated columns of $\hat{A}$, then these algorithms require at most $\binom{L+m-1}{m}$ iterations. Below we sketch a proof of

Lemma 3.1. whp $L \le m^{2m\log\log n+16m}$.

If this bound on L holds then

$$\binom{L+m-1}{m-1} \le 2L^m \le e^{(2+o(1))m^2\log\log n\log m}$$

So if m is small i.e. $O((\log n)^{1/2}/\log\log n)$ the algorithm uses a polynomial number of iterations whp.

Proof. We actually prove

$$E(L) \le m^{2m\log\log n+16}, \qquad (3.1)$$

from which the result follows. Let $a = \left(\frac{2\log n}{n}\right)^{1/m}$. Consider $i = 0$ and let I_i be the index set of the $\lceil an\rceil$ largest elements in $A_{i,\cdot}$. Let $I = \bigcap_{i=0}^{m-1} I_i$. Then

$$E(|I|) \ge \lceil a^m n\rceil \ge 2\log n.$$

Exercise: show that $\Pr(I = \emptyset) \le \frac{1}{n}$ (this is easy if m is 2, the general case requires iterative applications of the Hoeffding-Azuma inequality discussed in Chapters 6 and 7).

Any column not in $I_0 \cup \cdots \cup I_{m-1}$ is dominated by a column with index in I. So, using the result of the exercise, the expected number of undominated columns exceeds the sum of the number of undominated columns in each I_j by at most 1. Letting $f(m,n)$ be the expected number of undominated columns in a matrix with m columns and n rows each of which is uniformly randomly permuted, we obtain

$$f(m,n) \le mf(m,\lceil an\rceil) + 1.$$

Checking inductively that $f(m,n) \le m^{(2m\log\log n+16)}$ yields the desired result. (The 16 in the exponent allows us to assume n is at least 2^{16}).

3.3 Shortest Paths

Most work in this area has been restricted to that of finding shortest paths between all pairs of nodes in a complete digraph with independently chosen random non-negative edge weights. More generally, one considers distributions which are endpoint independent. Loosely, this means that if the edges leaving a vertex are sorted according to their cost, then the associated endpoints occur in random order. Spira [104] showed that using a heap in a version of Dijkstra's algorithm [33] gave a solution in $O(n^2(\log n)^2)$ expected time. This was improved by Bloniarz [13] and Frieze and Grimmett [51]. Moffat and Takaoka [93] subsequently reduced the expected running time to $O(n^2\log n)$. Recently, Mehlhorn and Priebe [89] show this algorithm runs in time $O(n^2\log n)$ whp and not just in expectation. They also give an $\Omega(n\log n)$ lower bound for the single source problem under a class of distributions.

Luby and Ragde [85] consider the problem of finding a single shortest path between a source s and a sink t. They show that searching simultaneously from both s and t can be efficient on average. For example they give a $\Theta(\sqrt{n}\log n)$ time bound assuming sorted edge lists and edge lengths chosen independently from "reasonable" distributions.

Spira's Algorithm

For each $v \in V$ we keep a list L_v of the edges $(v,w): w \ne v$, sorted in increasing order of length. It takes $O(n^2\log n)$ time to produce these lists. By the assumption of endpoint independence these orderings are random and independent of each other. We keep pointers $p_v, v \in V$ which are initialised to point to a dummy element preceding the first real element of L_v.

The algorithm consists of n single source problems, one for each $s \in V$. Consider one such problem for some $s \in V$. As usual the algorithm incrementally produces a set S (initially $S = \{s\}$) containing those vertices v for which a shortest path from s to v has been calculated. For each $v \in S$ we keep a value $\delta(v)$. When v is added to S we have

$$\delta(v) = dist(s,v) + \min_{w \notin S} \ell(v,w). \qquad (3.2)$$

We do not immediately update $\delta(v)$ each time we update S. This saves time on average.

The algorithm needs an auxiliary data structure Q called a priority queue. Q admits the following operations: insert an item, delete an item and determine the item of minimum value. Each such operation takes $O(\log n)$ time.

An iteration of Spira's algorithm consists of

```
1. a) Determine the minimum value $\delta(v) = dist(s,v) + \ell(v,w)$ in $Q$;
      If $w \notin S$ then
        i. Add $w$ to $S$.
        ii. $dist(s,w) = \delta(v)$
        iii. goto 2.
   b) Otherwise move $p_v$ one position to the next vertex $w'$ on $L_v$;
   c) Replace $\delta(v)$ by $dist(s,v) + \ell(v,w')$ and update $Q$, goto 1
2. Currently $p_w$ is pointing to a dummy element of $L_w$. Let $x$ be the first
   element of $L_w$.
3. Put $\delta(w) = dist(s,w) + \ell(w,x)$ and insert this value into $Q$.
```

It is straightforward to show that this algorithm solves the all-pairs shortest path problem.

Time Analysis

We argue that if $|S| = k$ then the expected number of times we find $u \in S$ in Step 1 is $O(n/(n-k))$. Thus the total expected running time for each single source shortest path problem is of the order

$$\sum_{k=1}^{n-1} \frac{n}{n-k}\log n = O(n(\log n)^2).$$

To explain the bound $O(n/(n-k))$ we need to apply the method of deferred decisions. In particular, for each vertex u we expose the $n-1$ distances from u without exposing the other endpoints. By the endpoint independence assumption, every bijection between the other endpoints and the distances is equally likely. Thus, at Step 1 (resp. 1(b)), we do not actually expose the vertex v (resp. w); we simply expose the next distance. It is only in Step 1(c) that we expose the actual vertex name associated with the distance. Suppose in Step 1(a) p_u points to the tth member of L_u. We have already exposed the names of the first $t-1$ vertices on L_u and they are all in S. By the endpoint independence assumption the tth vertex is equally likely to be any of the remaining $n-t$ vertices. Thus, the probability that the tth vertex is in S is at most $\frac{k}{n-1}$, conditional on the history of the process so far. The next execution of Step 1(a) may move to a different value for u, but this probability bound remains true. Thus if X is the random number of moves needed to find a vertex not in S, then

$$\Pr(X > r) \le \left(\frac{k}{n-1}\right)^r$$

and

$$E(X) \le \sum_{r=0}^{\infty}\left(\frac{k}{n-1}\right)^r = \frac{n-1}{n-k-1}.$$

There are only two papers we know of that deal with arbitrary, as opposed to non-negative weights. Kolliopoulos and Stein [82] modify the Bellman-Ford dynamic programming algorithm and show that a single source problem can be solved in $O(n^2\log n)$ expected time when the distribution is endpoint independent. Their model allows negative cycles. Cooper, Frieze, Mehlhorn and Priebe [35] consider a model in which the arc costs c_{ij} are generated from

$$c_{ij} = -a_i + a_j + b_{ij}$$

where $b_{ij} \ge 0$. It is assumed that the b_{ij}'s are independent, identically distributed, bounded and their common probability function F satisfies $F'(0) > 0$. The a_i's are arbitrary and of size $O(n/(\log n)^2)$. The algorithm does not see the a's and b's, only the values c_{ij}. They show that a single source shortest path problem can be solved in $O(n^2)$ expected time and an all pairs shortest path problem can be solved in $O(n^2\log n)$ expected time.

4. Asymptotic Optimality and Approximation

In this chapter, we change the focus of our probabilistic analysis. We examine polynomial time algorithms which do not necessarily output optimal solutions and examine how well they perform on typical instances. We discuss Bin Packing, the Euclidean and Asymmetric TSP, and disjoint paths problems.

4.1. Bin Packing

In its simplest form we are given $x_1, x_2, \ldots, x_n \in [0,1]$ and are asked to partition $[n] = \{1, 2, \ldots, n\}$ into $S_1, S_2, \ldots, S_k$ such that $\sum_{i \in S_j} x_i \le 1$ for $j = 1, 2, \ldots, k$ and such that k is as small as possible. The elements $i \in S_j$ are thought of as being placed in bin j which has capacity 1. Then k is the number of bins used.

The analysis of bin packing algorithms has proved to be very challenging. There are many deep results and the reader is referred to a survey by Coffman and Johnson [33] for further reading.

We now give an accessible result, essentially due to Frederickson [47]. Suppose that $x_1, x_2, \ldots, x_n$ are independent uniform [0,1] random variables. It is clear that the expected number of bins required is at least $E(\sum_{i=1}^n x_i)$ which is $\frac{n}{2}$. We describe an algorithm FFD for which the expected number of bins used is at most $\frac{n}{2} + O(\sqrt{n\log n})$. (Frederickson proved the bound $\frac{n}{2} + 2n^{2/3}$ with a similar analysis, we make no attempt to optimise the constants.)

Let $\alpha = 1 - \frac{(\log n)^{1/2}}{n^{1/2}}$.

1. Place each element $x_i \ge \alpha$ into a bin on its own. Suppose there are B_1 such.
2. Let $N = n - B_1$ be the number of bins remaining to be packed.
3. Order the items so that $x_1 \le x_2 \le \cdots \le x_N \le \alpha$.
4. For $i = 1, 2, \ldots, \lfloor N/2 \rfloor$
   (a) Put x_i, x_{N-i+1} into one bin if $x_i + x_{N-i+1} \le 1$.
   (b) Put x_i, x_{N-i+1} into separate bins if $x_i + x_{N-i+1} > 1$.
   Put item $\lceil N/2 \rceil$ into a separate bin if N is odd.

The desired bound on the expected number of bins used by FOLD is implied by:

Theorem 4.1. For n sufficiently large, the expected number of bins packed by FOLD is at most $\frac{n}{2} + 7\log n\sqrt{n}$.

Proof. Each item has size greater than i with probability $\frac{n-i}{n}$, so $\mathbf{E}(B_i) = 6\log n\sqrt{n}$. We show that for $i = 1, 2, \ldots, \lceil N/2\rceil$,

$$\Pr(x_i + x_{N-i+1} > 1) \le \frac{1}{n}. \qquad (4.1)$$

Thus, the expected number of bins used in step 2 is less than $\frac{N}{2} - 2$ and the theorem follows. To prove (4.1), we show that:

$$\Pr\left(x_i > \frac{i + 3\log n\sqrt{n}}{n}\right) \le \frac{1}{2n} \qquad (4.2)$$

and

$$\Pr\left(x_{N-i+1} > \frac{n - i - 3\log n\sqrt{n}}{n}\right) \le \frac{1}{2n}. \qquad (4.3)$$

To prove (4.2) we note that $x_i > p = \frac{i+3\log n\sqrt{n}}{n}$ if and only if there are at most i items of size less than p. But each item has size less than p with probability p and so we can apply the Chernoff Bound to obtain the desired result. We obtain (4.3) via a similar but slightly messier computation. □

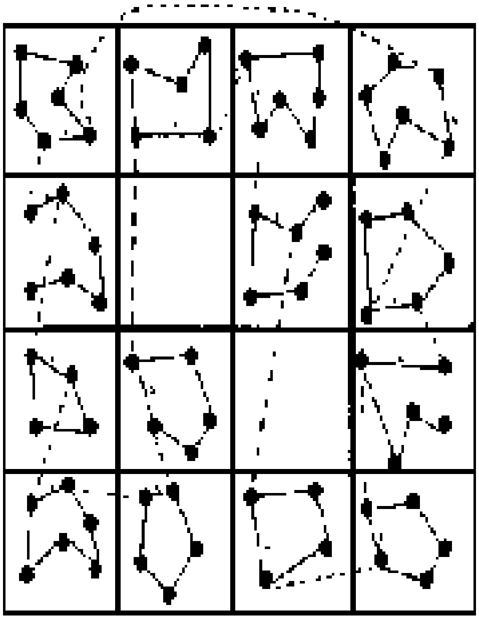
4.2 Euclidean Travelling Salesman Problem

One of the earliest and most influential results in the probabilistic analysis of combinatorial optimization problems was Karp's partitioning algorithm [?] for the travelling salesman problem in the unit square $C = [0,1]^2$. Here we have n points $X_1, X_2, \ldots, X_n$ chosen uniformly at random in C and the problem is to find the minimum length tour (i.e. Hamilton cycle) through them, using Euclidean distance to define the distance between points.

We let $\ell(T)$ be the length of a tour T and let $L^* = L^*(X_1, X_2, \ldots, X_n)$ be the minimum length of a tour. We give an outline of a simplified version of Karp's algorithm. First we mention the equally important results of Beardwood, Halton and Hammersley [?]. Their results are stronger and more general, but in any case they imply that there exists an (unknown) constant $\beta > 0$ such that for any $\epsilon > 0$

$$\lim_{n\to\infty} \Pr\left(\left|\frac{L^*}{\sqrt{n}} - \beta\right| > \epsilon\right) = 0.$$

In other words we expect that $L^* \approx \beta\sqrt{n}$. Consider the following heuristic:



Patch by adding broken edges and deleting edges marked with a x

Fig. 4.1

Partitioning Algorithm

(a) Divide C into $M = m^2$ squares $C_1, C_2, \ldots, C_M$ of size $\frac{1}{m} \times \frac{1}{m}$ where $m = \epsilon\sqrt{n}$ for some small $\epsilon > 0$.

(b) Find an optimal tour T_i through the points A_i in each C_i.

(c) Patch these tours together to make a tour $\hat{T}$ as indicated in Figure 4.1.

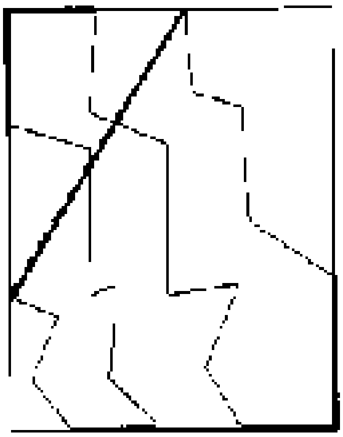
Let T^* be the optimum tour and let L_i^* be the length of the edges and parts of edges of T^* which lie in C_i. One can patch these edges into a tour of A_i, see Figure 4.2, at an additional cost of at most the perimeter of C_i. Therefore

$$L_i^* \ge \ell(T_i) - \frac{4}{m} \qquad 1 \le i \le M. \qquad (4.4)$$

The length of the tour $\hat{T}$ obtained by the patching satisfies

$$\ell(\hat{T}) \le \sum_{i=1}^{M} \ell(T_i) + 6m. \qquad (4.5)$$

It follows from (4.4) and (4.5) that



Edge of optimal tour

Added edge

Fig. 4.2

$$T^* \le |\hat{T}| \le T^* + 16\epsilon\sqrt{n}$$

Since $T^* \approx \beta\sqrt{n}$ whp we see that $\hat{T}$ is asymptotically optimal.

How long does it take to compute $\hat{T}$? Each tour T_i can be computed in time $O(|A_i|^2 2^{|A_i|})$ by dynamic programming. Now $|A_i|$ has distribution $B = B(n, 1/M)$ and so the expected running time for computing all the T_i's is of order

$$E\left(\sum_{i=1}^{M} |A_i|^2 2^{|A_i|}\right) = ME(B^2 2^B)$$

$$= M\sum_{k=0}^{n}\binom{n}{k}k^2 2^k M^{-k}\left(1-\frac{1}{M}\right)^{n-k}$$

$$\le 2M\left(1-\frac{1}{M}\right)^{n}\sum_{k=2}^{n}\binom{n}{k}k(k-1)\left(\frac{2}{M-1}\right)^{k} + 2e^{-1}n$$

$$\le \frac{2}{e}M\sum_{k=2}^{n}n(n-1)\binom{n-2}{k-2}\left(\frac{2}{M-1}\right)^{k} + 2e^{-1}n$$

$$= \frac{8n(n-1)}{(M-1)^2}\left(1+\frac{2}{M-1}\right)^{n-2} + 2e^{-1}n$$

$$\le 3e^{-2}\epsilon^{-1}n$$

This constitutes the main amount of work and so in expected time $O(\epsilon^{-2}n)$ we can find a solution which is likely to be within $1 + O(\epsilon)$ of optimal.

Since the appearance of [73] and [16] there has been a great amount of research effort directed the analysis of optimization problems in Euclidean space. A recent book by Steele [104] is an excellent source for this material.

4.3 Asymmetric Travelling Salesman Problem

The *Assignment Problem* (AP) is the problem of finding a minimum-weight perfect matching in an edge-weighted bipartite graph. An instance of the AP can be specified by an $n \times n$ matrix $M = (m_{ij})$; here m_{ij} represents the weight of the edge between x_i and y_j, where $X = \{x_1, x_2, \ldots, x_n\}$ is the set of "left vertices" in the bipartite graph, and $Y = \{y_1, y_2, \ldots, y_n\}$ is the set of "right vertices." The AP can be stated in terms of the matrix M as follows: find a permutation $\sigma^* = \sigma^*(M)$ of $\{1, 2, \ldots, n\}$ that minimizes $\sum_{i=1}^{n} m_{i\sigma(i)}$. Let $AP(M)$ be the optimal value of the instance of the AP specified by M.

The *Asymmetric Traveling-Salesman Problem* (ATSP) is the problem of finding a Hamiltonian circuit of minimum weight in an edge-weighted directed graph. An instance of the ATSP can be specified by an $n \times n$ matrix $M = (m_{ij})$ in which m_{ij} denotes the weight of edge $\langle i, j\rangle$. The ATSP can be stated in terms of the matrix M as follows: find a cyclic permutation $\pi^* = \pi^*(M)$ of $\{1, 2, \ldots, n\}$ that minimizes $\sum_{i=1}^{n} m_{i\pi(i)}$. Here the cycle structure of a permutation π is just the set of cycles formed by the arcs $\langle i, \pi(i)\rangle$ and a cyclic permutation is one whose cycle structure consists of a single cycle. Let $ATSP(M)$ be the optimal value of the instance of the ATSP specified by M.

It is evident from the parallelism between the above two definitions that $AP(M) \le ATSP(M)$. The ATSP is NP-hard, whereas the AP is solvable in time $O(n^3)$.

Karp [74] studied the relationship between AP and ATSP when entries of the matrix M are independent [0,1] uniform random variables. He proved the rather surprising result that

$$E(ATSP(M)) \le E(AP(M)) + o(1).$$

The proof was quite involved and later on Karp and Steele [76] simplified the argument and improved the error term. Subsequently, Dyer and Frieze [40] reduced the error term to $O((\log n)^4/(n\log\log n))$. We give an outline of the approach from [76]. The first important observation is that the solution of $AP(M)$ will be a random permutation.

$$\Pr(\sigma^*(M) = \sigma_1) = \Pr(\sigma^*(\sigma_2 M) = \sigma_2\sigma_1) = \Pr(\sigma^*(M) = \sigma_2\sigma_1)$$

where M' is the matrix obtained by permuting the columns of M by π. Note that M and M' have the same distribution. Thus whp the optimal solution σ^* will have $O(\log n)$ cycles. See e.g. Bollobás [14].

Karp and Steele then argue that whp the optimal solution to $AP(M)$ does not contain any edges of length greater than $\lambda = K(\log n)^2/n$ for some suitably large constant $K > 0$. Thus if we remove the edges of length greater than λ from the problem before solving $AP(M)$ then whp we will get the same solution. This means that we can pessimistically consider the edges not in the optimal assignment solution to independently have length uniform in $[\lambda, 1]$ as we delay specifying their exact length until after solving the AP.

Suppose that the solution to $AP(M)$ consists of cycles $C_1, C_2, \ldots, C_k$ where $|C_1| \geq |C_2| \geq \cdots \geq |C_k|$ where $|C_1| = \Omega(n/\log n)$. The idea is to iteratively patch C_{i+1} into a cycle $\hat{C}_i$ formed on the vertices of $C_1 \cup C_2 \cup \cdots \cup C_i$.

A patch involves deleting an edge e_1 of C_{i+1} and an edge e_2 of $\hat{C}_i$ and replacing them by the edges e_3, e_4 to create a single cycle. The algorithm chooses the patch which minimises the cost $c_{e_3} + c_{e_4}$. If $|\hat{C}_i| = a$ and $|C_{i+1}| = b$ and Z_i denotes the cost of the best patch, then for any $\xi > 0$

$$\Pr(Z_i \geq 2(\xi + \lambda)) \leq (1 - \xi^2)^{ab}$$

This is because if $Z_i \geq 2(\xi + \lambda)$ then for every relevant e_1, e_2 it is not the case that $c_{e_3} \leq \xi + \lambda$ and $c_{e_4} \leq \xi + \lambda$. The ab possible events of this form can be considered independent as they deal with disjoint sets of edges. Now by assumption $ab = \Omega(n/\log n)$ and so

$$\Pr(\exists i : Z_i \geq (\log n)^2/n^{1/2}) = o(1).$$

Whp there are $O(\log n)$ cycles and so whp the total patching cost is $O((\log n)^3/n^{1/2})$.

4.4 Disjoint Paths

Suppose we are given a graph $G = (V, E)$ and a set of pairs (a_i, b_i), $1 \leq i \leq K$ of vertices. In the Edge Disjoint Paths Problem (EDPP) we want to find paths P_i joining vertex a_i to b_i, for $1 \leq i \leq K$ which are edge disjoint, or prove it is not possible. In the Vertex Disjoint Paths Problem (VDPP), the vertices are all distinct and we want vertex disjoint paths. Both problems are solvable in polynomial time if K is fixed, independent of the input, Robertson and Seymour [78], but NP-hard if K varies. The problem is interesting for theoretical and practical reasons, the latter interest comes from its use as a model for some communication problems.

For random graphs $G_{n,m}$ the VDPP was considered by Shamir and Upfal [80] who gave a linear time algorithm which whp succeeds in finding paths provided $m \geq 2n \log n$ and $K = O(\sqrt{n})$. It should be remarked that here the two sets of vertices are fixed before the random graph is constructed. The problem was also considered by Hochbaum [54] who gave an $O(m)$ time algorithm when $K = O(\sqrt{d}/\log n)$, where here and in what follows $d = 2m/n$ is the average degree. Both algorithms are based on growing disjoint forests rooted at the sources and sinks until the corresponding trees are large enough so that for each i the tree rooted at a_i can be joined to the tree rooted at b_i.

The above approach is simple and efficient, but does not address the problem when the random graph is constructed first and then the sources and sinks are chosen by an adversary. Suppose $2m/n - \log n \to \infty$ so that $G_{n,m}$ is connected whp. Let D be the median distance between pairs of vertices in $G_{n,m}$. Then $D = O(\log n/\log d)$ whp. Clearly it is not possible to connect more than $O(m/D)$ pairs of vertices by edge-disjoint paths, for all choices of pairs, since some choice would require more edges than all the edges available. Also, some restriction on the number of times a vertex can be a source or sink is necessary. Thus the following theorem of Broder, Frieze, Suen and Upfal [22] is optimal up to constant factors.

Theorem 4.2. *Suppose $2m/n - \log n \to \infty$. Then there exist positive constants α and β such that whp, for all $A = \{a_1, a_2, \ldots, a_K\}$, $B = \{b_1, b_2, \ldots, b_K\} \subseteq [n]$ satisfying*

(i) $K = \lceil \alpha m \log d/\log n \rceil$,

(ii) for each vertex v, $|\{i : a_i = v\}| + |\{i : b_i = v\}| \leq \min\{d_G(v), \beta d\}$,

there exist edge-disjoint paths in $G_{n,m}$ joining a_i to b_i, for each $i = 1, 2, \ldots, K$. Furthermore, there is an $O(nm^2)$ time randomized algorithm for constructing these paths.

The strategy for proving Theorem 4.2 is quite different from [80] and [54]. First of all the sources and sinks are joined, by a network flow algorithm to randomly chosen $\tilde{a}_i, \tilde{b}_i$, $1 \leq i \leq K$. This has a spreading out effect, similar to that achieved by the method of Valiant and Brebner [88] for routing messages in the n-cube. The new sources and sinks are then joined up by utilising random walks.

Frieze and Zhao [41] have extended the above ideas to deal with random r-regular graphs where r is considered to be constant.

The VDPP is discussed in [23]. Using similar ideas to those above it is shown that:

Theorem 4.3. *Suppose $2m/n - \log n \to \infty$. Then there exist positive constants α, β such that whp, for all $A = \{a_1, a_2, \ldots, a_K\}$, $B = \{b_1, b_2, \ldots, b_K\} \subseteq [n]$ satisfying*

(i) $A \cap B = \emptyset$,

(a) $|A| = |B| = K \le \frac{\alpha n \log d}{\log n}$,

(b) $|N(v) \cap (A \cup B)| \le \beta |N(v)|, \quad \forall v \in V$,

then there are vertex disjoint paths P_i from a_i to b_i for $1 \le i \le K$. Furthermore, there is an $O(n^3)$ time randomized algorithm for constructing these paths.

Here $N(v)$ is the neighbour set of vertex v. This is again optimal up to the constant factors α, β.

5. Greedy Algorithms

In this chapter, we continue to focus on the average performance guarantees of algorithms which are sure to run in polynomial time. In particular, we focus on the expected behaviour of greedy algorithms. These algorithms are appealing because they are usually fast and easy to implement. We consider three examples: a greedy algorithm for constructing a stable set, a greedy algorithm for constructing a matching, and a greedy algorithm for the Knapsack Problem.

5.1 Cliques, Stable Sets, and Colourings

We consider the following greedy algorithm for constructing a stable set. Pick a vertex x, determine which vertices are not adjacent to x, recursively apply the algorithm to find a stable set S in the graph induced by these vertices, and return $S + x$.

We prove:

Whp the above algorithm finds a stable set of size at least $\log_2 n - 3\log_2\log_2 n$ in $G_{n,\frac{1}{2}}$. (5.1)

Proof. The algorithm terminates with a stable set S such that every vertex of $G - S$ sees a vertex of S. But it is easy to compute that the number of such sets (stable or otherwise) with fewer than the given number of vertices is $o(1)$. □

For a sharper analysis see [6]. Now, a classic result, see [4] states that

Whp the largest stable set in $G_{n,\frac{1}{2}}$ has $2\log_2 n - 2\log_2\log_2 n - O(1)$ elements (5.2)

Thus the algorithm typically constructs a stable set which is about half the size of the largest stable set.

We can analyse our algorithm using the method of deferred decisions. We note that in constructing the stable set we need only examine edges which have an endpoint in the stable set. It follows that $G_{n,\frac{1}{2}} - S$ is a uniformly chosen random graph on vertex set $V_n - S$. So, we can re-apply our algorithm to find a stable set, disjoint from S. Repeating this procedure allows us to colour G with $(1 + o(1))\frac{n}{\log_2 n}$ colours. A beautiful analysis due to Bollobás [3], which can be found in the third section of the sixth chapter of this book shows:

Whp the chromatic number of $G_{n,\frac{1}{2}}$ is $(1 + o(1))\frac{n}{2\log_2 n}$. (5.3)

Thus our colouring algorithm uses about twice the optimal number of colours. To close this section, we mention two open problems.

Research Problem Develop a polynomial-time algorithm which finds a stable set of size $(\frac{1}{2} + \epsilon)\log_2 n$ in $G_{n,\frac{1}{2}}$ whp, for some constant $\epsilon > 0$.

Research Problem Develop a polynomial-time algorithm which finds a colouring of $G_{n,\frac{1}{2}}$ using $(1 - \epsilon)\frac{n}{\log_2 n}$ colours whp, for some constant $\epsilon > 0$.

5.2 Greedy Matchings

In this section we consider finding large matchings in sparse random graphs. Recall that the random graph $G_{n,m}$ has vertex set $[1, 2, \ldots, n]$ and m random edges. The graph is considered to be sparse if $m = \lfloor cn \rfloor$ for some constant $c > 0$. In this case $G_{n,m}$ has no perfect matching whp. We leave it as an exercise to show that, in fact, whp there are a large number of isolated vertices. This is an interesting case, because as we have seen it is easy to find a perfect matching when there are many more edges. For such a sparse random graph the interest is in using a simple heuristic to find a large matching which is close to optimal whp. Researchers have concentrated in the main on the analysis of greedy heuristics:

```
GREEDY
    begin
        $M \leftarrow \emptyset$;
        while $E(G) \neq \emptyset$ do
        begin
            A:  Choose $e = \{u, v\} \in E$
                $G \leftarrow G \setminus \{u, v\}$;
                $M \leftarrow M \cup \{e\}$
        end;
    Output $M$
    end
```

($G \setminus \{u,v\}$ is the graph obtained from G by deleting the vertices u, v and all edges incident with them, together with any vertices which become isolated.)

The average performance of GREEDY when the input is random was first analysed by Tinhofer [89]. He considered its performance on the random graph $G_{n,p}$ in the dense case where p is fixed independent of n. In this case it is fairly easy to show that the algorithm produces a matching of size $n/2 - O(\log n)$ whp. In fact the analysis in Section 3.1 essentially yields this result.

Let $X = X(n,m)$ be the random number of edges in the matching produced by GREEDY applied to $G_{n,m}$ when the edge choice in statement A is uniformly random. Dyer, Frieze and Pittel [40] were able to establish the asymptotic distribution of this variable when $m = \lfloor cn \rfloor$. In particular they showed that $E(X) \approx \phi(c)n$ where $\phi(c) = \frac{c}{2c+1}$ (and that this variable is asymptotically normal).

It is possible to modify this algorithm without considerable complications, so as to improve its likely performance. Perhaps the simplest modification is to first choose a vertex v at random and then to randomly choose an edge incident with it. We refer to this as MODIFIED GREEDY. Dyer, Frieze and Pittel also analysed the performance of MODIFIED GREEDY in the same setting as for GREEDY. Let $\hat{X} = \hat{X}(n,m)$ be the random number of edges in the matching produced by MODIFIED GREEDY on $G_{n,m}$. Now the asymptotic expectation increases to $E(\hat{X}) \approx \hat{\phi}(c)n$ where $\hat{\phi}(c) = \frac{1}{2} - \frac{\log(2-e^{-2c})}{4c} > \phi(c)$.

GREEDY and MODIFIED-GREEDY both find matchings which are less than the maximum by a constant factor. Karp and Sipser [77] considered a similar greedy type of algorithm which we will call KSGREEDY. Their algorithm (a) chooses an edge incident to a vertex of degree 1 while there is one and otherwise (b) chooses a random edge. The algorithmic change is tiny, but the improvement in performance is spectacular. They show that this algorithm is asymptotically optimal in the sense that with high probability it finds a matching which is within $o(n)$ of the optimum size! They also prove that if $c \le e$ then KSGREEDY spends almost all of its time in case (a). The algorithm is considered to run in two phases. Phase 1 ends when the minimum degree of the graph that remains is at least two. Note that during Phase 1 the algorithm makes correct choices in the sense that the edges chosen are a subset of some maximum matching.

Aronson, Frieze and Pittel [6] have undertaken a further analysis of this algorithm.

- If $c < e$ then at the end of Phase 1, all that is left of the graph is a few vertex disjoint cycles.

- If $c > e$ then in Phase 2, KSGREEDY will match all but about $n^{1/5}$ of those vertices which remain at the end of Phase 1. More precisely, there exist positive constants c_1, c_2, a, b such that if L denotes the number of vertices which become isolated in Phase 2, then
$$c_1 n^{1/5}(\log n)^{-a} \le E(L) \le c_2 n^{1/5}(\log n)^{b}. \qquad (5.4)$$
- Analysis of the algorithm gives an asymptotic expression for the size of the maximum matching in $G_{n,m}$.

Another possible version of GREEDY is MINGREEDY where in Step A one chooses a (random) vertex of minimum degree and then a random neighbour of this vertex. Frieze, Radcliffe and Suen [50] considered the performance of MINGREEDY on random cubic graphs (a graph is cubic if every vertex has degree three). They proved

Theorem 5.1. *Let L_n denote the number of vertices left exposed by the matching constructed by running MINGREEDY on a random cubic graph with n vertices. Then there exist constants $d_1, d_2 > 0$ such that*
$$d_1 n^{1/5} \le E(L_n) \le d_2 n^{1/5} \log n. \qquad (5.5)$$

We note that a random cubic graph has a perfect matching whp, see for example Bollobás [14].

Thus MINGREEDY usually does very well. Note the common exponent 1/5 in (5.4) and (5.5). This can be explained to some extent by the fact that near the end of KSGREEDY, when most available vertex deletions are made, the maximum degree is bounded whp.

In computational experiments MINGREEDY left an average of just over 10 vertices unmatched when run on random cubic graphs with 10^6 vertices.

5.3 Knapsack Problems

In this section we consider the 0-1 Knapsack problem in which we have n items $I_1, \ldots, I_n$, some subset of which we shall put in a knapsack. Each item I_j has an associated weight w_j and profit p_j. Our restriction is that the knapsack can hold total weight at most W and our objective is to maximize the profit. That is, we solve

$$\text{Maximize} \quad \sum_{j=1}^{n} p_j x_j \qquad (5.6)$$

$$\text{Subject to} \quad \sum_{j=1}^{n} w_j x_j \le W \qquad (5.7)$$

$$x_j = 0/1 \qquad 1 \le j \le n$$

Here we analyse a random instance in which the coefficients $p_1,\ldots,p_n$, $w_1,\ldots,w_n$ are independently chosen from the unit interval $[0,1]$. For the constraint (5.7) to be active but not too strong we let $W = \beta n$ where $0 < \beta < 1/2$. The following greedy algorithm is likely to have a good asymptotic average performance.

```
    Greedy
begin
    Order the variables in increasing order of value $w_j/p_j$.
    $S := 0$; $x_j := 0$ for $j = 1$ to $n$;
    For $j = 1$ to $n$ do
    begin
        If $w_j \le W - S$ then $x_j := 1$; $S := S + w_j$
    end
end
```

The algorithm is known to produce at least a 1/2 optimal solution, but is likely to do much better. Let Z^* denote the optimal value in (5.6), Z_{LP} the optimal solution to the Linear Programming relaxation and Z_G the value of the solution produced by Greedy. It is easy to see that to obtain an optimal solution to the linear programming relaxation, we simply take the solution obtained by Greedy and put into the knapsack as much as we can of the item not in the knapsack which maximizes $\frac{p_j}{w_j}$. Thus,

$$Z^* \ge Z_G \ge Z_{LP} - 1 \ge Z^* - 1. \tag{5.8}$$

It is easy to derive, as the reader may wish to do, that $Z_{LP} = \Omega(n)$ and hence by the above equation is a very good approximation to Z^* (by e.g. using the Chernoff Bound to show that there are about $\frac{n}{4}$ items whose profit is greater than $\frac{1}{2}$ and whose weight is less than $\frac{1}{2}$). We present a more complicated analysis which allows us to calculate Z_{LP} more precisely. Assuming $w_1 + w_2 + \cdots + w_n > W$ (and this is true whp):

$$Z_{LP} = \sum_{j=1}^{r} p_j + \alpha p_{r+1}$$

where $0 \le \alpha < 1$ and

$$\sum_{j=1}^{r} w_j + \alpha w_{r+1} = W < \sum_{j=1}^{r+1} w_j.$$

There is a geometric interpretation:

The pairs (w_j, p_j) are chosen uniformly from the unit square OABC. We sweep the line OX clockwise starting at OA until we have swept over points whose w sum exceeds W. Then we stop with OX through a point (w_r, p_r) where $\theta_r = \theta$.

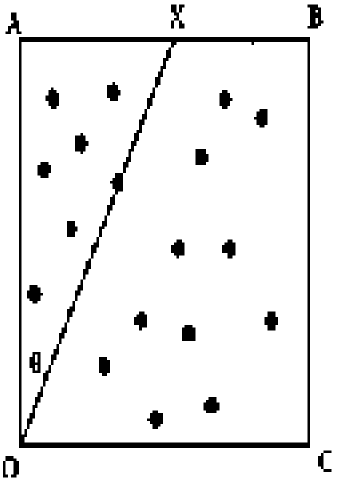


Fig. 5.1

Now consider a fixed θ and let A_θ denote the area of the region T_θ to the left of OX.

$$A_\theta = \begin{cases} \frac{\tan\theta}{2} & 0 \le \theta \le \pi/4 \\ 1 - \frac{\cot\theta}{2} & \pi/4 \le \theta \le \pi/2 \end{cases}$$

Next let w_θ denote the expected w coordinate of a point chosen uniformly at random within T_θ and let p_θ be the corresponding expected p coordinate.

$$w_\theta = \begin{cases} \frac{\tan\theta}{3} & 0 \le \theta \le \pi/4 \\ \frac{c^2-3c+3}{3(2-c)} & \pi/4 \le \theta \le \pi/2 \end{cases} \qquad c = \cot\theta$$

and

$$p_\theta = \begin{cases} \frac{2}{3} & 0 \le \theta \le \pi/4 \\ \frac{3c^2-c^3+3}{3(2-c)} & \pi/4 \le \theta \le \pi/2 \end{cases}$$

The expected weight $w(T_\theta)$ of points falling in T_θ is $nA_\theta w_\theta$. Define θ_0 by $A_{\theta_0} w_{\theta_0} = \beta$. Applying a simple standard concentration result (e.g. the Hoeffding-Azuma Inequality, see Chapter 6) we obtain that for any t

$$\Pr(|w(T_\theta) - nA_\theta w_\theta| \ge t) \le 2e^{-2t^2/n}$$

and

$$\Pr(|p(T_\theta) - nA_\theta p_\theta| \ge t) \le 2e^{-2t^2/n}$$

It follows that whp

$$Z_{LP} = nA_{\theta_0} p_{\theta_0} + O(\omega n^{1/2}) \tag{5.9}$$

for any $\omega \to \infty$.

It follows from (5.6) and (5.8) that whp Z_G is a good approximation to Z^*.

This is fairly simple. Lueker [76] proved a much deeper result.

$$E(Z_{LP} - Z^*) = O((\log n)^2/n).$$

He did this basically by showing that whp there exists a good integer solution obtainable by changing a few ($O(\log n)$) values of x_j in the optimal linear program. Goldberg and Marchetti-Spaccamela [48] used this to define a simple enumerative search with the following property: for any $\epsilon > 0$ there is an $O(n^{1/\epsilon})$ time algorithm which solves this model of a knapsack problem exactly with probability at least $1-\epsilon$.

Subsequently Dyer and Frieze [36, 41] extended this approach to multi-dimensional knapsack problems and generalised assignment problems with a bounded number of constraints.

Mamer and Schilling [87] established probabilistic approximation results for multi-dimensional knapsack problems with the number of constraints growing with n.

Related problems
In the Subset-Sum problem we are given $a_1, a_2, \ldots, a_n, b$ and asked to decide if there exists a subset $S \subseteq \{1, 2, \ldots, n\}$ such that $a(S) = \sum_{i \in S} a_i = b$. This has some cryptographic applications. Lagarias and Odlyzko [82] gave a lattice based algorithm for solving this problem when the a_i are chosen independently from $\{1, 2, \ldots, 2^{n^2}\}$ and $b = \sum_{i \in S^*} a_i$ for some unknown set S^*. Frieze [50] gave a simplified analysis of their result.

In the Partition problem we are given $a_1, a_2, \ldots, a_n$ and asked to find the set S which minimises $|a(S) - a(\bar{S})|$. Assume that $a_1, a_2, \ldots, a_n$ are chosen independently and uniformly from $[0,1]$. It is known that whp this minimum is of order $n2^{-n}$, see Karmarkar, Karp, Lueker and Odlyzko [71]. On the other hand, Karmarkar and Karp [70] gave an algorithm which whp finds a set S with $|a(S) - a(\bar{S})| \leq (\log n)^{-c \log n}$ for some constant $c > 0$. They gave another more elegant and natural algorithm and conjectured that it has the same performance. This was recently verified in a lovely paper by Yakir [111].

6 Negative Results

In this chapter, we focus on results which show that algorithms are typically inefficient or that problems are usually hard. Actually, we devote almost all of our discussion to the first of these topics. To begin we present a proof that a certain branch and bound algorithm for the knapsack problem takes super-polynomial time whp on a random example drawn from a specific probability distribution. We then present less detailed discussions of similar results for the quadratic assignment problem and the k-median problem. Finally we survey some other results in this vein.

Showing that problems are difficult on average is much harder than showing that a certain algorithm is typically inefficient. In particular, if we show that an NP-complete problem is difficult on average then we can deduce that $P \neq NP$. The best we can hope for is to prove "on average" completeness results analogous to those developed for NP. This theory is outside the scope of this paper and uses a very different notion of "average". For these reasons, we content ourselves with giving the address of a website dedicated to the theory and a quote from some introductory material posted on the web-site. The website is:

http://www.uncg.edu/mat/avg.html

The quote is:

> Despite many years of intensive effort, there are no known efficient algorithms for NP-complete problems, where by efficient we mean algorithms that are fast in the worst case. Due to this striking gap in our knowledge, the search for algorithms that are "efficient" according to various more modest criteria has attracted increasing attention.
>
> One particularly interesting criterion is that of requiring problems be solvable quickly "on average." That is, one can solve NP-complete problems via algorithms that, although possibly very slow on some inputs, are fast on average with respect to some underlying probability distribution on instances. Algorithms that are fast on average have been found for several NP-complete problems, such as the vertex coloring problem and the Hamiltonian path problem, under commonly used distributions on graphs.
>
> However, there also are NP-complete problems that have so far resisted such "average case" attacks. Are these problems difficult on average? What does it mean for a problem to be difficult on average, and how is one to know whether a problem is difficult on average? In his seminal paper [84], Levin initiated the study of these questions. Two fundamental and robust notions were defined along lines similar to (standard, worst-case) NP-completeness theory. Namely, he introduced the notion of average polynomial time for measuring "easiness" on average and the notion of average case NP completeness for measuring "hardness" on average. Levin then showed that a tiling problem is average-case NP-complete if each parameter of an instance is randomly selected. This framework has been studied and enhanced by a number of researchers and several more average case NP complete problems have been found. Such average-case completeness results, as indicated by Levin [84], may not only save misguided "positive" efforts such as trying to find fast-on-average algorithms for problems

that probably exist them—but might also be used in areas (like cryptography) where hardness on average of some problems is a frequent assumption.

6.1 Knapsack

The simplest method for solving a 0-1 knapsack problem is to compute the weight and profit of each subset of the items and choose the highest profit subset that fits in the knapsack. We can enumerate all these possible solutions in a systematic way with the aid of a complete binary tree of height n as shown in Figure 6.1. Each path of the tree from the root to the leaves corresponds to a partial solution where if we branch right at height i then item i is to be in the solution and if we branch left at height i it is not.

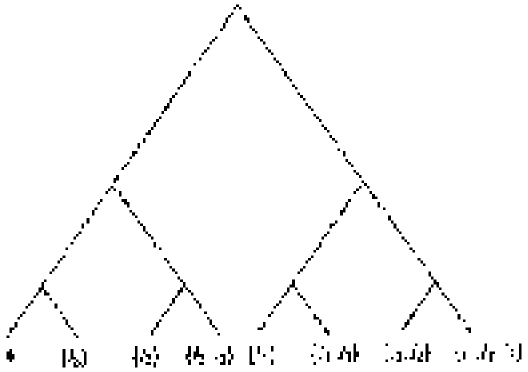


Fig. 6.1
A complete enumeration tree.

More generally we can construct an enumeration tree T which is a complete binary tree of height n such that

(i) every node v corresponds to a partial solution consisting of a subset S_v of the items and a partition of S_v into two sets P_v, those which we intend to put into the knapsack, and Q_v, those which we do not intend to put in the knapsack.

(ii) For r the root of the tree S_r is empty and for each non-leaf node v with right child v' and left child v'' there is an item I_v not in S_v such that $S_{v'} = S_{v''} = S_v + I_v$, $P_{v''} = P_v$, $P_{v'} = P_v + I_v$.

See Figure 6.2 for an example. Thus, in our original enumeration tree we insisted that if two nodes u and v have the same level then $I_u = I_v$, a condition we now drop without losing the bijection between the leaves and the subsets of the items.

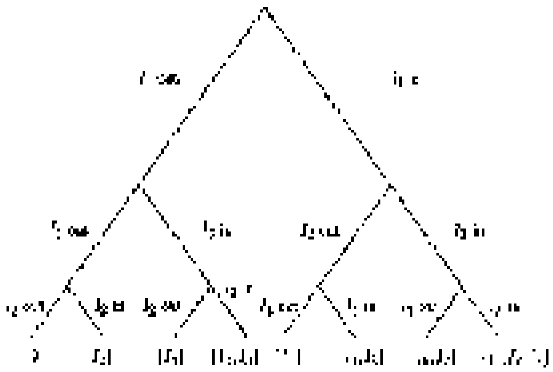


Fig. 6.2

Now, in generating all the candidate solutions, we do not need to construct the whole tree. For example, if there is a node v such that $\sum_{i \in P_v} w(i) > B$ then for every leaf u in the subtree T_v underneath v, since $P_v \subseteq P_u$, P_u does not fit in the knapsack. So there is no point exploring T_v. More generally, there is no point in exploring the subtree underneath a node if we know there is no optimal solution underneath this node.

In a branch and bound algorithm for the 0-1 knapsack problem, we generate some partial subtree of a complete enumeration tree whilst ensuring that one of its leaves corresponds to an optimal solution. We begin with the root, and repeatedly branch out from the tree constructed so far by adding two children to some leaf l. Throughout the algorithm, we have a set of active leaves of the current tree, which are those underneath which we intend to search. We must ensure that at all times, there is some optimal solution lying in a subtree underneath an active leaf. Initially, the root is active, and when we branch (from an active leaf), the two new leaves become active. We may make a leaf l inactive for either of the following two reasons:

(i) An already explicitly computed solution has at least as good an objective value as the best solution in T_l, or

(ii) there is another active leaf l' such that for any solution corresponding to a leaf of T_l there is a leaf of $T_{l'}$ which corresponds to a solution which is at least as good.

We continue growing the partial enumeration tree, as long as there are any active leaves which are not also leaves of the complete enumeration tree.

making nodes inactive whenever we can. Obviously, the best solution corresponding to a leaf of our partial tree is an optimal solution to the knapsack problem. Our hope is that the pruning due to (A)–(C), and a clever choice of the items on which we choose to branch, will restrict the partial tree to a reasonable size.

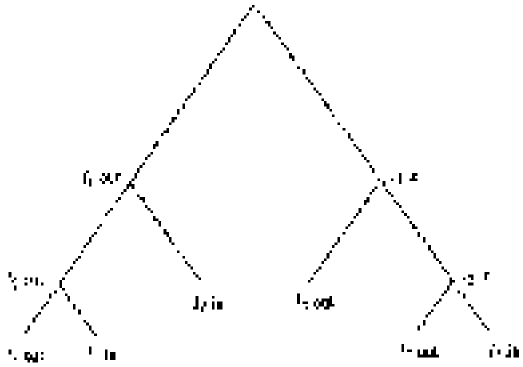


Fig. 6.3
A partial enumeration tree.

We remark that this technique clearly generalizes to other optimization problems. In particular, it is often applied to 0-1 programming problems, in which case to compute a bound on the best possible solution in T_s we usually consider the fractional relaxation of the integer program. For example, we remark that in our knapsack problem, for any node s of the partial tree, a solution corresponding to a leaf of T_s has profit at most $B_s = \sum_{i \in P_s} p_i + (B - \sum_{i \in P_s} w_i) \max_{i \in F_s} (\frac{p_i}{w_i})$, because any fractional solution with $x_i = 1$ for each $I_i \in P_s$ will generate at most this much profit. Thus if B_s is less than the profit of the optimal solution found so far, then we can make s inactive. The results in Section 6.3 can be reinterpreted as stating that using this pruning procedure and always branching so as to maximize $\frac{p_i}{w_i}$ for the item I_i on which we branch, for sufficiently small ϵ, we obtain the optimal solution in polynomial time with probability $1 - \epsilon$.

We turn now to a specific 0-1 knapsack problem and a refinement of this branch and bound algorithm. We insist that the weights and costs are all integers. We note that, in this case we can improve the above remark and obtain:

For any node s of the partial tree, let d_s be the greatest common divisor of the weights of the items not in P_s. Then a solution corresponding to a leaf of T_s has profit at most:

$$C_s = \sum_{i \in P_s} p_i + d_s \left\lfloor \frac{(B - \sum_{i \in P_s} w_i)}{d_s} \right\rfloor \max_{i \in F_s} \left(\frac{p_i}{w_i}\right) \qquad (6.1)$$

We denote by OPT the best solution found to date by the algorithm. We will make a node s inactive if:

(A) $\sum_{i \in P_s} w_i > B$, or

(B) $C_s \le$ OPT, or

(C) there is an active leaf t such that $F_t = F_s$, $\sum_{i \in P_t} p_i \ge \sum_{i \in P_s} p_i$ and $\sum_{i \in P_t} w_i \le \sum_{i \in P_s} w_i$.

We remark that for any s, t as in (C), if $P_s + X$ is the set of items put in the knapsack for some feasible solution corresponding to a leaf of T_s, then $P_t + X$ is at least as good a solution and corresponds to a leaf of T_t. This justifies our making s inactive.

We apply this algorithm to knapsack problems in which the costs and weights are equal and B is the sum of the weights divided by two and rounded down. Thus, we are considering a generalization of the partition problem, and an optimal solution can have profit at most B. Now, since $\frac{p_i}{w_i} = 1$ for all i, we only apply (B) at a node if the corresponding d exceeds 1, or we find a solution of value B. Further we only apply (C) at a node s if there is another node t such that $F_t = F_s$ and $\sum_{i \in P_t} w_i = \sum_{i \in P_s} w_i$ (note that by construction if $F_t = F_s$, we must have $P_t \neq P_s$).

We choose a random knapsack instance of this type by choosing each $w_i = p_i$ to be a uniform integer between 1 and 10^n, and then setting $B = \frac{\sum_{i=1}^n w_i}{2}$. We prove a theorem of Chvátal, originally proven in [C].

Theorem 6.1. *Whp none of the $2^{n/10}$ nodes in the first $\frac{n}{10}$ levels of the tree are made inactive. Hence whp the algorithm takes exponential time.*

Proof. Whp the following properties hold:

Property 1: there does not exist a set of $\frac{n}{10}$ items the sum of whose weights exceed B.

Property 2: there do not exist two distinct sets of items with the same weight.

Property 3: there does not exist a set of items the sum of whose weights is B.

Property 4: no integer d greater than 1 divides more than $\frac{4n}{5}$ of the items.

Now, if Property 1 holds then we never apply (A) to a node in the first $\frac{n}{10}$ levels. Similarly, if Properties 3 and 4 hold then we never apply (B) to a node in the first $\frac{n}{10}$ levels. Finally, if Property 2 holds then we never apply (C) to a node in the first $\frac{n}{10}$ levels. So this result implies the theorem; we leave its proof as an exercise in applying the First Moment Method. □

6.2 k-Median

We have a set X of n points $\{X_1, X_2, \ldots, X_n\}$ with distance $d_{i,j}$ between X_i and X_j. The k-median problem is to find a set $S \subseteq X$, $|S| = k$ which minimises $\sum_{i=1}^{n} d(X_i, S)$ where $d(X_i, S)$ is the minimum of $d_{i,j}$ over $j \in S$. As an integer program this can be expressed:

$$\begin{array}{lll}
\text{Minimise} & \sum_{i=1}^{n} \sum_{j=1}^{n} d_{i,j} x_{i,j} & \\
\text{Subject to} & \sum_{j=1}^{n} x_{i,j} = 1 & 1 \le i \le n \\
 & \sum_{j=1}^{n} y_j = k & \\
 & 0 \le x_{i,j} \le y_j \le 1 & 1 \le i,j \le n \\
 & y_j \in \{0,1\} & 1 \le j \le n
\end{array}$$

The strong linear programming relaxation is obtained by removing the integrality constraint on the y_j's. In practice this has been very useful a linear programming relaxation for branch and bound algorithms. Nevertheless a probabilistic analysis by Ahn, Cooper, Cornuéjols and Frieze [4] shows that in several probabilistic models, including points chosen uniformly in the unit square, the number of branches needed in such a branch and bound algorithm is whp at least $n^{\alpha k}$ for some constant α, provided $k/\log n \to \infty$ and $k = o((n/\log n)^{1/2})$. Thus in this case a probabilistic analysis does not agree with computational experience.

6.3 Quadratic Assignment

Here we have n items which have to be placed in n positions, one item to a position. There is a cost $a_{i,j,p,q}$ associated with placing item i in position p and item j in position q. The total cost is the sum of these costs and the problem is to:

$$\begin{array}{lll}
\text{Minimise} & \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{p=1}^{n} \sum_{q=1}^{n} a_{i,j,p,q} x_{i,p} x_{j,q} & \\
\text{Subject to} & \sum_{p=1}^{n} x_{i,p} = 1 & 1 \le i \le n \\
 & \sum_{i=1}^{n} x_{i,p} = 1 & 1 \le p \le n \\
 & x_{i,p} = 0/1 & 1 \le i,p \le n
\end{array}$$

This is a rather difficult problem and many branch and bound algorithms are based on (i) replacing the terms $x_{i,p}x_{j,q}$ by new 0/1 variables $y_{i,j,p,q}$ and adding suitable linear constraints to make a linear integer program and then (ii) relaxing the integrality of the $y_{i,j,p,q}$ to give a linear program (often this is only done approximately).

Assume that the $a_{i,j,p,q}$ are independent uniform [0,1] random variables. The expected optimum value then becomes $n^2/2$ – see Section 7.2. Dyer, Frieze and McDiarmid [44] show that the expected value of the linear relaxation considered above is at most $5n + O(1)$, i.e. there is a severe duality gap problem. Not unexpectedly they go on to show that as a consequence, any branch and bound algorithm based on using the LP relaxation for a bound will whp require an exponential number of branches to solve the problem.

6.4 Further Results

The first result giving bounds on the average-case complexity of branch and bound type algorithms are due to Chvátal and concern the maximum stable set problem [29]. Further results on this problem are given in Jerrum [67] and in Pittel [94]. McDiarmid [88] obtained difficulty results for vertex colouring. Perhaps the most impressive result of this type concerns the well-known resolution rule for Satisfiability. Chvátal and Szemerédi [32] showed that it will take exponential time whp for an appropriate probability distribution.

7. Non-Algorithmic Issues

The performance of some of our algorithms may be highly sensitive to the probability distribution which we use. We present two examples here concerning the asymmetric TSP and SAT. We also present results in the opposite direction, which show that for some problems an algorithm's performance is essentially independent of which input it is given. Here we may show that under some probability distributions, the algorithm will get close to the same answer on all but a tiny fraction of the inputs. As an example we consider the quadratic assignment problem.

7.1 Thresholds

7.1.1 Satisfiability. Given a boolean formula in conjunctive normal form the satisfiability problem (SAT) is to determine whether there is a truth assignment that satisfies it (see Chapter 1 for a longer definition). Since SAT is NP-complete one is interested in efficient heuristics that perform well "on average," or with high probability. The choice of the probabilistic space is crucial for the significance of such a study. In particular it is easy to decide SAT in probabilistic spaces that generate formulas with large clauses [59]. To circumvent this problem, recent studies have focused on formulas with exactly k literals per clause (the k-SAT problem). Of particular interest is the case $k = 3$, since this is the minimal k for which the problem is NP-complete.

Let V_n be a set of n variables. We define a uniform probability space $\Omega_{n,m}$ on the set of all $m = \lceil cn \rceil$ clause formulas over the variables which have exactly k literals per clause.

Most practical algorithms for the satisfiability problem (such as the well-known Davis-Putnam algorithm [36]) work iteratively. At each iteration, the algorithm selects a literal and assigns it the value 1. All clauses containing this literal are erased from the formula and the complement of the chosen literal is erased from the remaining clauses. Algorithms differ in the way they select the literal for each iteration. The following three rules are the most common ones:

1. The unit clause rule: If a clause contains only one literal, that literal must have the value 1.
2. The pure literal rule: If a formula contains a literal but does not contain its complement, this literal is assigned the value 1.
3. The smallest clause rule: Give value 1 to a (random) literal in a (random) smallest clause.

Broder, Frieze and Upfal [21] analysed an algorithm based solely on the pure literal rule. They showed that when $k = 3$ the pure literal rule alone is sufficient to find, with high probability, a satisfying assignment for a random formula $I \in \Omega^{(3)}_{n,m}$, for $c = m/n \le 1.63$. On the other hand, if $c > 1.7$, then the pure literal rule by itself does not suffice. The gap between 1.63 and 1.7 has been closed by Brightwell, Broder, Frieze, Mitzenmacher and Upfal [20]. In fact, if z is the solution to

$$(1-z)^{3/2} + \exp\left(\frac{1}{2((1-z)^{-1/2}-1)}\right) - 1 = 0,$$

and

$$c_* = \frac{1}{3(1-z)^{1/2}-\sqrt{(1-z)}}$$

then then the pure literal rule is sufficient whp when $c < c_*$ and the pure literal rule will almost surely be insufficient when $c > c_*$.

Chao and Franco [29],[30], Chvátal and Reed [31] and Frieze and Suen [56] analysed based on the small clause rule

```
begin
    repeat
        choose a literal $x$
        remove all clauses from $I$ that contain $x$ and remove $\bar{x}$ from any
        remaining clause
        if a clause becomes empty - HALT, FAILURE
    until no clauses left
    HALT, SUCCESS
end
```

In particular, in the case of 3-SAT Frieze and Suen showed that if $c_3 = 3.003$ is the solution to the equation

$$3c - 2\log c = 6 - 2\log(2/3),$$

then a small clause rule combined with some limited backtracking is enough to find a satisfying assignment whp, whenever $c < c_3$. From the other end it is easy to show that if c is sufficiently large then whp there is no satisfying assignment. There have been several attempts to estimate how large is large. Kamath, Motwani, Palem and Spirakis [68] showed that 4.758 is large enough for 3-SAT and subsequently Kirousis, Kranakis and Krizanc [79] reduced this to 4.598. Experimental evidence [93] strongly suggests that there exists a threshold γ, such that formulas are almost surely satisfiable for $c < \gamma$ and almost surely unsatisfiable for $c > \gamma$, where γ is about 4.2. This has not been proven rigorously, but such a threshold (namely $c=1$) is known to exist for 2-SAT formulas [58, 31]. On the other hand, Friedgut [46] has shown that there is a sharp threshold c_n for each n. We refer the reader to the paper for an explanation of what this means. Basically, the question now is as to whether c_n tends to a limit as $n \to \infty$.

7.1.2 The Asymmetric TSP. In this section, we consider the ATSP where each cost is a uniform integer between 0 and L_n for some integer L_n. If $L_n \le \frac{cn}{\log n}$ then a variant of Karp and Steele's algorithm can be used to show that some optimal AP solution can be patched to an optimal ATSP solution using only zero cost edges. Frieze, Karp and Reed [55] using a more involved argument showed

$$ATSP - AP = \begin{cases} 0 & \text{whp} & \text{if } L_n/n \to 0 \\ 0 & \text{with prob} \ge \epsilon > 0 & \text{if } L_n = cn \\ > 0 \text{ whp} & & \text{if } L_n/n \to \infty \end{cases}$$

These work was partially motivated by computational results of Miller and Pekny [91].

Research problem: Determine the relationship between the optimal solutions for AP and ATSP when $L_n = cn$.

Research Problem: Show that for L_n sufficiently large, the Branch and Bound procedure of Miller and Pekny which is based on Karp and Steele's algorithm, takes exponential time whp.

7.2 Concentration

Concentration inequalities generalizing the Chernoff Bound are discussed in Chapter 6 (particularly useful is the Hoeffding Azuma Inequality). They can be used to show that for many optimization problems, the optimal solution values of the instances of size n are heavily concentrated around the expected value of the optimal solution. In Section 3 of Chapter 6, such a result is presented for Bin Packing. Section 4 of that chapter presents similar results for the Euclidean TSP and another geometric problem, Minimum Cost Steiner Tree.

There are cases where such an analysis can lead to counter-intuitive results which make near optimization a trivial exercise whp. We close this chapter with one such result.

Consider the Quadratic Assignment Problem (QAP) defined in Section 6.3. As we have seen any branch and bound algorithm based on a natural linear programming relaxation will take exponential time whp. On the other hand, we suggest that whp one cannot avoid finding a solution which is near optimal.

Fix an assignment $x = [x_{ij}]$ and let

$$Z_x = \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\sum_{q=1}^{n} a_{ijpq}x_{ip}x_{jq}.$$

The values a_{ijpq} are independent uniform $[0,1]$. Hence, for a fixed x, the random variable Z_x has mean

$$\mathbf{E}(Z_x) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\sum_{q=1}^{n} x_{ip}x_{jq} = \frac{n^2}{2}.$$

Z_x is the sum of n^2 independent random variables $\{a_{ijpq} : x_{ip} = x_{jq} = 1\}$ and so a standard analysis (in fact a straightforward application of the Hoeffding-Azuma inequality) yields:

$$\Pr(|Z_x - n^2/2| \geq t) \leq e^{-2t^2/n^2}$$

for any $t > 0$. In particular, if $t = \alpha n^{3/2}\sqrt{\log n}$ where $\alpha = \alpha(n) \to \infty$ then we have

$$\Pr(|Z_x - n^2/2| \geq \alpha n^{3/2}\sqrt{\log n}) \leq e^{-2\alpha^2 n\log n}.$$

Now there are only $n!$ solutions to QAP and so

$$\Pr(\exists x : |Z_x - n^2/2| \geq \alpha n^{3/2}\sqrt{\log n}) \leq n!e^{-2\alpha^2 n\log n} \to 0.$$

Our conclusion therefore is that whp every solution to QAP has an objective value in the interval $[n^2/2 - \alpha n^{3/2}\sqrt{\log n}, n^2/2 + \alpha n^{3/2}\sqrt{\log n}]$ and taking say $\alpha = o((n/\log n)^{1/2})$ we see that any solution is within $1 + o(1)$ of the optimum.

This was first observed by Burkard and Fincke [34]. More recent examples of this phenomenon are given by Barvinok [11] and Szpankowski [105].

References

1. Adler I., Karp R.M. and Shamir R. (1986): A family of simplex variants solving an m × d linear program in O(min(m², d²)) expected number of pivot steps, University of California, Computer Science Division, Berkeley.
2. Adler I. and Megiddo N. (1983): A simplex algorithm whose average number of steps is bounded between two quadratic functions of the smaller dimension, Department of Industrial Engineering and Operations Research, University of California, Berkeley.
3. Adler I., Megiddo N. and Todd M.J. (1984): New results on the average behavior of simplex algorithms Bulletin of the American Mathematical Society 11, 378-382.
4. Ahn S., Cooper C., Cornuéjols G. and Frieze A.M. (1988): Probabilistic analysis of a relaxation for the k-median problem, Mathematics of Operations Research 13, 1-31.
5. Alon N. and Kahale N. (1994): A spectral technique for coloring random 3-colorable graphs, Proceedings of the 26th Annual ACM Symposium on Theory of Computing, 346-355.
6. Aronson J., Frieze A.M. and Pittel B.G. (1998): Maximum matchings in sparse random graphs: Karp-Sipser revisited, Random Structures and Algorithms 12, 111-177.
7. Arora S. (1996): Polynomial time approximation schemes for Euclidean TSP and other geometric problems, Proceedings of the 37th Annual Symposium on Foundations of Computer Science 2-11.
8. Babai L., Erdős P. and Selkow S.M. (1980): Random graph isomorphism, SIAM Journal on Computing 9, 628-635.
9. Babai L. and Kučera L. (1979): Canonical labelling of graphs in linear average time, Proceedings of the 20th Annual IEEE Symposium on the Foundations of Computer Science 39-46.
10. Beardwood J., Halton J.H. and Hammersley J.M. (1959): The shortest path through many points, Proceedings of the Cambridge Philosophical Society 55, 299-327.
11. Barvinok A. (1997): Measure concentration in optimization, Mathematical Programming Series B, 79 (Lectures on Mathematical Programming, ISMP 97, T.M. Liebling and D. de Werra eds.), 33-53.
12. Blair C. (1986): Random linear programs with many variables and few constraints, Mathematical Programming 34, 62-71.
13. Bloniarz P. (1983): A shortest-path algorithm with expected time $O(n^2\log n\log^* n)$, SIAM Journal on Computing 12, 588-600.
14. Bollobás B. (1985): Random Graphs, Academic Press.
15. Bollobás B. (1988): The chromatic number of random graphs, Combinatorica 8, 49-55.
16. Bollobás B., Fenner T.I. and Frieze A.M. (1987): An algorithm for finding hamilton paths and cycles in random graphs, Combinatorica 7, 327-341.
17. Boppana R. (1987): Eigenvalues and graph bisection: an average-case analysis, Proceedings of the 28th Annual IEEE Symposium on the Foundations of Computer Science 280-285.
18. Borgwardt K.H. (1982): The average number of pivot steps required by the simplex method is polynomial, Zeitschrift für Operations Research 26, 157-177.
19. Borgwardt K.H. (1987): The simplex method, a probabilistic analysis, Springer-Verlag.

20. Brightwell G., Broder A.Z., Frieze A.M., Mitzenmacher M. and Upfal E., On the satisfiability and maximum satisfiability of random 3-CNF formulas, to appear.
21. Broder A.Z., Frieze A.M. and Upfal E. (1993), On the satisfiability and maximum satisfiability of random 3-CNF formulas, Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms 341-351.
22. Broder A.Z., Frieze A.M., Suen S. and Upfal E. (1994), Optimal construction of edge-disjoint paths in random graphs, Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms 603-612.
23. Broder A.Z., Frieze A.M., Suen S. and Upfal E. (1996), Optimal construction of vertex-disjoint paths in random graphs, Proceedings of the Seventh Annual ACM-SIAM Symposium on Discrete Algorithms 261-268.
24. Burkard R.E. and Fincke U. (1983), The asymptotic probabilistic behaviour of quadratic sum assignment problems, Zeitschrift für Operations Research 27, 73-81.
25. Bui T., Chaudhuri S., Leighton T. and Sipser M. (1987), Graph bisection with good average case behaviour, Combinatorica 7, 171-191.
26. Chao M.T. and Franco J. (1986), Probabilistic analysis of two heuristics for the 3-satisfiability problem, SIAM Journal on Computing 15, 1106-1118.
27. Chao M.T. and Franco J. (1990), Probabilistic analysis of a generalization of the unit-clause literal selection heuristics for the k satisfiability problem, Information Sciences 51, 289-314.
28. Chen H. and Frieze A. (1996), Coloring bipartite hypergraphs, Proceedings of IPCO 1996, 345-358.
29. Chvátal V. (1977), Determining the stability number of a graph, SIAM Journal on Computing 6, 643-662.
30. Chvátal V. (1980), Hard knapsack problems, Operations Research 28, 1402-1411.
31. Chvátal V. and Reed B. (1992), Mick gets some (the odds are on his side), Proceedings of the 33rd IEEE Symposium on Foundations of Computer Science, 620-627.
32. Chvátal V. and Szemerédi E. (1988), Many hard examples for Resolution, Journal of the Association for Computing Machinery 35, 759-768.
33. Coffman E.G. and Johnson D.S. (1997), Approximation algorithms for bin packing: a survey, in Approximation Algorithms for NP-hard Problems, D.S. Hochbaum (Ed.), PWS.
34. Coffman E.G. and Lueker G.S. (1991), Probabilistic analysis of packing and partitioning algorithms, John Wiley and Sons, New York.
35. Cooper C., Frieze A.M., Mehlhorn K. and Priebe V. (1997), Average-case analysis of shortest-paths algorithms in the vertex potential model, Randomization and approximation techniques in Computer Science (Proceedings of RANDOM 97) Lecture Notes in Computer Science 1269, 15-26.
36. Davis M. and Putnam H. (1960), A computing procedure for quantification theory, Journal of the ACM 7, 201-215.
37. Dijkstra E. (1959), A note on two problems in connexion with graphs, Numerische Mathematik 1, 269-271.
38. Dyer M.E. and Frieze A.M. (1989), Fast algorithms for some random NP-hard problems, Journal of Algorithms 10, 451-489.
39. Dyer M.E. and Frieze A.M. (1989), Probabilistic analysis of random m-dimensional knapsack problems, Mathematics of Operations Research 14, 162-176.
40. Dyer M.E. and Frieze A.M. (1990), On patching algorithms for random asymmetric travelling salesman problems, Mathematical Programming 46, 361-378.
41. Dyer M.E. and Frieze A.M. (1992), Probabilistic analysis of the generalised assignment problem, Mathematical Programming 55, 169-181.
42. Dyer M.E., Frieze A.M. and McDiarmid C.J.H. (1986), Linear programs with random costs, Mathematical Programming 35, 3-16.
43. Dyer M.E., Frieze A.M. and Pittel B.G. (1993), On the average performance of the greedy algorithm for finding a matching in a graph, Annals of Applied Probability 3, 526-552.
44. Erdős P. and Wilson R.J. (1977), On the chromatic index of almost all graphs, Journal of Combinatorial Theory B 23, 255-257.
45. Flajolet P. and Sedgewick R. (1996), An introduction to the analysis of algorithms, Addison-Wesley, New York.
46. Fournier J.C. (1973), Colorations des arêtes d'un graphe, Cahiers Centre Études Rech. Opér. 15, 311-314.
47. Frederickson G. (1980), Probabilistic analysis for simple one and two dimensional bin packing algorithms, Information Processing Letters 11, 156-161.
48. Friedgut E. (1998), Necessary and Sufficient conditions for Sharp Thresholds of Graph Properties, and the k-sat Problem, to appear.
49. Frieze A.M. (1988), An algorithm for finding Hamilton cycles in random digraphs, Journal of Algorithms 9, 181-204.
50. Frieze A.M. (1986), On the Lagarias-Odlyzko algorithm for the subset sum problem, SIAM Journal on Computing 15, 536-539.
51. Frieze A.M. and Grimmett G.R. (1985), The shortest path problem for graphs with random arc-lengths, Discrete Applied Mathematics 10, 57-77.
52. Frieze A.M., Jackson W., McDiarmid C.J.H. and Reed B. (1988), Edge-colouring random graphs, Journal of Combinatorial Theory B 45, 135-149.
53. Frieze A.M. and McDiarmid C.J.H. (1997), Algorithmic theory of random graphs, Random Structures and Algorithms 10, 5-42.
54. Frieze A.M., Radcliffe J. and Suen S. (1993), Analysis of a simple greedy matching algorithm on random cubic graphs, Proceedings of the Fourth Annual ACM-SIAM Symposium on Discrete Algorithms 341-351.
55. Frieze A.M., Karp R.M. and Reed B.A. (1995), When is the assignment bound asymptotically tight for the asymmetric traveling-salesman problem? SIAM Journal on Computing 24, 484-493.
56. Frieze A.M. and Suen S. (1996), Analysis of two simple heuristics on a random instance of k-SAT, Journal of Algorithms 20, 312-355.
57. Frieze A.M. and Zhao L., Optimal Construction of Edge-Disjoint Paths in Random Regular Graphs, in preparation.
58. Goerdt A. (1992), A threshold for unsatisfiability, Fifth International Symposium on Mathematical Foundations of Computer Science, Springer-Verlag LNCS 629, 264-274.
59. Goldberg A. (1979), Average case complexity of the satisfiability problem, Proceedings of the Workshop on Automated Deduction, 1-6.
60. Goldberg A.V. and Marchetti-Spaccamela A. (1984), On finding the exact solution of a 0,1 knapsack problem, Proceedings of the 16th Annual ACM Symposium on the Theory of Computing 359-368.
61. Grimmett G.R. and McDiarmid C.J.H. (1975), On colouring random graphs, Proceedings of the Cambridge Philosophical Society 77, 313-324.
62. Gurevich Y. and Shelah S. (1987), Expected computation time for Hamilton path problem, SIAM Journal on Computing 16, 486-502.
63. Haimovich M. (1983), The simplex algorithm is very good! – on the expected number of pivot steps and related properties of random linear programs, Columbia University, New York.

64. Held M. and Karp R.M. (1962): A Dynamic Programming approach to sequencing problems, SIAM Journal of Applied Mathematics 10, 196-210.
65. Hochbaum D.S. (1992): An exact sublinear algorithm for the max-flow, vertex disjoint paths and communication problems on random graphs, Operations Research 40, 923-935.
66. Holyer I. (1981): The NP-completeness of edge colouring, SIAM Journal of Computing 10, 718-720.
67. Jerrum M.R. (1992): Large cliques elude the Metropolis process, Random Structures and Algorithms 3, 347-359.
68. Jerrum M.R. and Sorkin G. (1993): Simulated Annealing for Graph Bisection, Proceedings of the 34th Annual IEEE Symposium on the Foundations of Computer Science 94-103.
69. Kamath A., Motwani R., Palem K. and Spirakis P. (1994): Tail bounds for occupancy and the satisfiability threshold conjecture, Proceedings of the 35th IEEE Symposium on Foundations of Computer Science, 592-603.
70. Karmarkar N. and Karp R.M. (1982): The differencing method of set partitioning, Technical Report UCB/CSD 82/113 Computer Science Division (EECS) University of California, Berkeley.
71. Karmarkar N., Karp R.M., Lueker G.S. and Odlyzko A.M. (1986): Probabilistic analysis of optimum partitioning, Journal of Applied Probability 23 626-645.
72. Karp R.M. (1976): Probabilistic analysis of a canonical numbering algorithm for graphs, Proceedings of Symposia in Pure Mathematics 34 American Mathematical Society (1979) 365-378, [illegible]
73. Karp R.M. (1977): Probabilistic Analysis of Partitioning Algorithms for the Traveling-Salesman Problem in the Plane, Mathematics of Operations Research 2 209-224.
74. Karp R.M. (1979): A patching algorithm for the non-symmetric traveling salesman problem, SIAM Journal on Computing 8, 561-573.
75. Karp R.M. (1972): Reducibility amongst combinatorial problems in R.E. Miller and J.W. Thatcher(Eds.) Complexity of computer computations, Plenum Press, New York.
76. Karp R.M., Lenstra J.K., McDiarmid C. and Rinnooy Kan A.H.G. (1985): Probabilistic analysis of combinatorial algorithms: an annotated bibliography, in Combinatorial Optimization: Annotated Bibliographies, (eds. M. O'hEigeartaigh, J.K. Lenstra and A.H.G. Rinnooy Kan), Wiley Chichester.
77. Karp R.M. and Sipser M. (1981): Maximum matchings in sparse random graphs, Proceedings of the 22nd Annual IEEE Symposium on the Foundations of Computer Science 364-375.
78. Karp R.M. and Steele J.M. (1985): Probabilistic analysis of heuristics in The traveling salesman problem: a guided tour of combinatorial optimization, E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan and D.B. Shmoys Eds.
79. Kirousis L.M., Kranakis E. and Krizanc D. (1996): Approximating the unsatisfiability threshold of random formulas, Proceedings of the 4th Annual European Symposium on Algorithms 27-38.
80. Klee V. and Minty G. (1972): How good is the simplex algorithm?, in Inequalities III, O. Shisha ed., Academic Press 159-175.
81. Knuth D.E., Motwani R. and Pittel B.G. (1990): Stable husbands, Random Structures and Algorithms 1, 1-14.
82. Kolliopoulos S.G. and Stein C. (1996): Finding real-valued single source shortest paths in $o(n^3)$ expected time, Proceedings of the 5th Conference on Integer Programming and Combinatorial Optimization 94-104.

83. Lagarias J.C. and Odlyzko A.M. (1985): Solving low density subset sum problems, Journal of ACM 32, 229-246.
84. Levin L. (1986): Average case complete problems, SIAM Journal of Computing 15 285-286.
85. Luby M. and Ragde P. (1989): Bidirectional search is $O(\sqrt{n})$ faster than Dijkstra's shortest path algorithm, Algorithmica 4, 551-567.
86. Lueker G.S. (1982): On the average distance between the solutions to linear and integer knapsack problems, Applied Probability - Computer Science, The Interface 1, 489-504.
87. Mamer J. and Schilling K. (1990): On the growth of random knapsacks, Discrete Applied Mathematics 28.
88. McDiarmid C. (1979): Determining the chromatic number of a graph, SIAM Journal on Computing 8, 1-14.
89. Mehlhorn K. and Priebe V. (1997): On the all pairs shortest path algorithm of Moffat and Takaoka, Random Structures Algorithms 10 205-220.
90. Micali S. and Vazirani V.V. (1980): An $O(\sqrt{|V|}|E|)$ algorithm for finding maximum matching in general graphs, Proceedings of the 21st Annual Symposium on Computer Science, IEEE, New York, 17-27.
91. Miller D.L. and Pekny J.F. (1991): Exact solution of large asymmetric traveling salesman problems, Science 251, 754-762.
92. Mitchell D., Selman B. and Levesque H. (1992): Hard and easy distributions of SAT problems, AAAI, 459-465.
93. Moffat A. and Takaoka T. (1985): An all pairs shortest path algorithm with expected time $O(n^2 \log n)$, Proceedings of the 26th Annual IEEE Symposium on the Foundations of Computer Science 101-105.
94. Padberg M. and Rao M. (1982): Odd minimum cut-sets and b-matchings, Mathematics of Operations Research 7, 67-80.
95. Perkovic L. and Reed B.A. (1998): Edge coloring random graphs in polynomial expected time, to appear.
96. Pittel B. (1982): On the probable behaviour of some algorithms for finding the stability number of a graph, Mathematical Proceedings of the Cambridge Philosophical Society 92, 511-526.
97. Reed B. (1997): Edge coloring nearly bipartite graphs, manuscript.
98. Robertson N. and Seymour P.D. (1995): Graph minors-XIII: The disjoint paths problem, Journal of Combinatorial Theory B 52 153-190.
99. Sedgewick R. and Flajolet P. (1996): An introduction to the analysis of algorithms, Addison Wesley, New York.
100. Shamir E. and Upfal E. (1985): A fast construction of disjoint paths in networks, Annals of Discrete Mathematics 24, 141-154.
101. Smale S. (1983): On the average number of steps of the simplex method of linear programming, Mathematical Programming 27, 241-262.
102. Smale S. (1983): The problem of the average speed of the simplex method, in Mathematical Programming: The State of the Art, A. Bachem, M. Grötschel and B. Korte 530-539.
103. Spira P.M. (1973): A new algorithm for finding all shortest paths in a graph of positive arcs in average time $O(n^2 \log^2 n)$, SIAM Journal on Computing 2, 28-32.
104. Steele J.M. (1997): Probability theory and combinatorial optimization, CBMS-NSF Regional Conference Series in Applied Mathematics 69.
105. Szpankowski W. (1995): Combinatorial optimization problems for which almost every algorithm is asymptotically optimal, Optimization 33, 359-367.
106. Thomason A. (1989): A simple linear expected time algorithm for Hamilton cycles, Discrete Mathematics 75, 373-379.

107. Tinhofer G. (1984): A probabilistic analysis of some greedy cardinality matching algorithms, Annals of Operations Research 1, 239-254.
108. Valiant L.G. and and Brebner G.J. (1981): Universal schemes for parallel communication. Proceedings of the 13th Annual ACM Symposium on Theory of Computing, 263-277.
109. Vizing V.G. (1964): On an estimate of the chromatic class of a p-graph (Russian) Diskret. Analiz 3, 25-30.
110. Yakir B. (1996): The differencing algorithm LDM for partitioning: a proof of a conjecture of Karmarkar and Karp, Mathematics of Operations Research 21, 85-99.

An Overview of Randomized Algorithms

Rajeev Motwani*1 and Prabhakar Raghavan2

1 Department of Computer Science, Stanford University, CA 94305
2 IBM Almaden Research Center, 650 Harry Road, San Jose CA 95120

1 Introduction and Terminology

A randomized algorithm makes random choices during its execution. The behavior of such an algorithm may thus be random even on a fixed input. The process of designing and analyzing a randomized algorithm focuses on establishing that it is likely to behave "well" on every input. The likelihood in such a statement depends only on the probabilistic choices made by the algorithm during execution and not on any assumptions about the input. It is especially important to distinguish a randomized algorithm from the average-case analysis of algorithms, where one analyzes an algorithm assuming that its input is drawn from a fixed probability distribution. With a randomized algorithm, in contrast, no assumption is made about the input.

Two benefits of randomized algorithms have made them popular: simplicity and efficiency. For many applications, a randomized algorithm is the simplest algorithm available, or the fastest, or both. Below we make these notions concrete through a number of illustrative examples. We assume that the reader has had undergraduate courses in Algorithms and Complexity, and in Probability Theory. A comprehensive source for randomized algorithms is the book by the authors [51]. The articles by Karp [39], Maffioli, Speranza, and Vercellis [49] and Welch [75] are good surveys of randomized algorithms. The book by Mulmuley [57] focuses on randomized geometric algorithms.

Throughout this chapter we assume the RAM model of computation, in which we have a machine that can perform the following operations involving registers and main memory: input-output operations, memory-register transfers, indirect addressing, branching and arithmetic operations. Each register or memory location may hold an integer which can be accessed as a unit, but an algorithm has no access to the representation of the number. The arithmetic instructions permitted are $+, -, \times, /$. In addition, an algorithm can

* Supported by an Alfred P. Sloan Research Fellowship, an IBM Faculty Partnership Award, an ARO MURI Grant DAAH04-96-1-0007, and NSF Young Investigator Award CCR-9357849, with matching funds from IBM, Schlumberger Foundation, Shell Foundation, and Xerox Corporation.

compare two numbers, and evaluate the square root of a positive number. In this article E[X] will denote the expectation of a random variable X, and Pr[A] will denote the probability of an event A.

1.1 Organization of This Survey

One of the principal ways of classifying randomized algorithms is to think of them as either *Monte Carlo* algorithms or as *Las Vegas* algorithms. A Las Vegas algorithm must terminate with the correct answer on every instance; the random choices it makes only influence its running time. We consider a Las Vegas algorithm to be efficient if its expected running time is polynomial in the size of the input. A Monte Carlo algorithm, on the other hand, can err on a given execution. Typically, we are interested in Monte Carlo algorithms that run for a number of steps that is polynomial in the size of the input. The key is to give an upper bound on the probability that the Monte Carlo algorithm errs; this bound should hold for every input. Thus, a Monte Carlo algorithm errs only because of "unlucky" random choices it makes. Moreover, independent repetitions of a Monte Carlo algorithm can be used to make the probability of error on all repetitions be very small.

The sorting algorithm of Section 2 as well as the game tree evaluation algorithm of Section 3 are Las Vegas algorithms. The fingerprinting algorithms of Section 6, on the other hand, are Monte Carlo algorithms. Section 4 considers the issue of proving lower bounds for randomized algorithms; the general technique introduced there borrows from game theory. A common technique for proving the existence of combinatorial objects with desired properties is the probabilistic method; this is described in Section 5.

2. Randomized Sorting

Consider sorting a set S of n numbers. The main idea behind these algorithms is the use of random sampling: a randomly chosen member of S is unlikely to be one of the largest or smallest elements; rather, it is likely to be "near the middle."

Algorithm **RandomQS** is inspired by the Quicksort algorithm due to Hoare [14]. We assume that the random choice in Step 1 can be made in unit time. We now analyze the expected number of comparisons in an execution of **RandomQS**. Comparisons are performed in Step 2, in which we compare a randomly chosen element to the remaining elements. For $1 \leq i \leq n$, let $S_{(i)}$ denote the element of rank i (the ith smallest element) in the set S. Define X_{ij} to assume the value 1 if $S_{(i)}$ and $S_{(j)}$ are compared in an execution, and the value 0 otherwise. Thus the total number of comparisons is $\sum_{i=1}^{n} \sum_{j>i} X_{ij}$. By linearity of expectation the expected number of comparisons is

```
Algorithm RandomQS
Input: A set of numbers S.
Output: The elements of S sorted in increasing order.
1. Choose an element y uniformly at random from S: every element in S has
   equal probability of being chosen.
2. By comparing each element of S with y, determine the set S_1 of elements
   smaller than y and the set S_2 of elements larger than y.
3. Recursively sort S_1 and S_2. Output the sorted version of S_1, followed by y,
   and then the sorted version of S_2.
```

$$\mathbf{E}\Big[\sum_{i=1}^{n} \sum_{j>i} X_{ij}\Big] = \sum_{i=1}^{n} \sum_{j>i} \mathbf{E}[X_{ij}]. \tag{2.1}$$

Let p_{ij} denote the probability that $S_{(i)}$ and $S_{(j)}$ are compared during an execution. Then

$$\mathbf{E}[X_{ij}] = p_{ij} \times 1 + (1 - p_{ij}) \times 0 = p_{ij}. \tag{2.2}$$

To compute p_{ij}, we view the execution of **RandomQS** as a labeled binary tree T. Each node of T is labeled with a distinct element of S. The root of the tree is labeled with the element y chosen in Step 1, the left subtree of y contains the elements in S_1 and the right subtree of y contains the elements in S_2. The structures of the two subtrees are determined recursively by the executions of **RandomQS** on S_1 and S_2. The root y is compared to the elements in the two subtrees, but no comparison is performed between an element of the left subtree and an element of the right subtree. Thus, there is a comparison between $S_{(i)}$ and $S_{(j)}$ if and only if one of these elements is an ancestor of the other.

Consider the permutation π obtained by visiting the nodes of T in increasing order of the level numbers, and in a left-to-right order within each level; recall that the ith level of the tree is the set of all nodes at distance exactly i from the root. The following two observations are the crux of the analysis:

1. There is a comparison between $S_{(i)}$ and $S_{(j)}$ if and only if $S_{(i)}$ or $S_{(j)}$ occurs earlier in the permutation π than any element $S_{(l)}$ such that $i <$

$i < j$. To see this, let $S_{(k)}$ be the earliest in π from among all elements of rank between i and j. If $k \notin \{i, j\}$, then $S_{(i)}$ will belong to the left subtree of $S_{(k)}$ while $S_{(j)}$ will belong to the right subtree of $S_{(k)}$, implying that there is no comparison between $S_{(i)}$ and $S_{(j)}$. Conversely, when $k \in \{i, j\}$, there is an ancestor-descendant relationship between $S_{(i)}$ and $S_{(j)}$, implying that the two elements are compared by RandomQS.

2. Any of the elements $S_{(i)}, S_{(i+1)}, \ldots, S_{(j)}$ is equally likely to be the first of these elements to be chosen as a partitioning element and hence to appear first in π. Thus, the probability that this first element is either $S_{(i)}$ or $S_{(j)}$ is exactly $2/(j-i+1)$.

Thus, $p_{ij} = 2/(j-i+1)$. By (2.1) and (2.2), the expected number of comparisons is given by

$$\sum_{i=1}^{n}\sum_{j>i} p_{ij} = \sum_{i=1}^{n}\sum_{j>i}\frac{2}{j-i+1}$$
$$\le \sum_{i=1}^{n-1}\sum_{k=1}^{n-i}\frac{2}{k+1}$$
$$\le 2\sum_{i=1}^{n}\sum_{k=1}^{n}\frac{1}{k}.$$

It follows that the expected number of comparisons is bounded above by $2nH_n$, where H_n is the nth Harmonic number, defined by $H_n = \sum_{k=1}^{n} 1/k$.

Theorem 2.1. *The expected number of comparisons in an execution of RandomQS is at most* $2nH_n$.

Now $H_n = \ln n + \Theta(1)$, so that the expected running time of RandomQS is $O(n \log n)$. Note that this expected running time holds for every input. It is an expectation that depends only on the random choices made by the algorithm, and not on any assumptions about the distribution of the input.

3. Foiling an Adversary

A common paradigm in the design of randomized algorithms is that of *foiling an adversary*. Whereas an adversary might defeat a deterministic algorithm with a carefully constructed "bad" input, it is difficult for an adversary to defeat a randomized algorithm in this fashion. The random choices made by the randomized algorithm prevent the adversary, while constructing the input, from predicting the precise behavior of the algorithm. An alternative view of this process is to think of the randomized algorithm as first picking a series of random numbers which it then uses in the course of execution as needed. In this view, we may think of the random numbers chosen at the start as "selecting" one of a family of deterministic algorithms. In other words a randomized algorithm can be thought of as a probability distribution on deterministic algorithms. We illustrate these ideas in the setting of AND-OR tree evaluation; the following algorithm is due to Snir [86].

An AND-OR tree is a rooted complete binary tree in which internal nodes at even distance from the root are labeled AND and internal nodes at odd distance are labeled OR. Associated with each leaf is a Boolean value. The evaluation of the game tree is the following process. Each leaf returns the value associated with it. Each OR node returns the Boolean OR of the values returned by its children, and each AND node returns the Boolean AND of the values returned by its children. At each step an evaluation algorithm chooses a leaf and reads its value. We do not charge the algorithm for any other computation. We study the number of such steps taken by an algorithm for evaluating an AND-OR tree, the worst case being taken over all assignments of Boolean values to the leaves.

Let T_k denote an AND-OR tree in which every leaf is at distance $2k$ from the root. Thus, any root-to-leaf path passes through k AND nodes (including the root itself) and k OR nodes, and there are 2^{2k} leaves. An algorithm begins by specifying a leaf whose value is to be read at the first step. Thereafter, it specifies such a leaf at each step, based on the values it has read on previous steps. In a deterministic algorithm, the choice of the next leaf to be read is a deterministic function of the values at the leaves read so far. For a randomized algorithm, this choice may be randomized. It is not hard to show that for any deterministic evaluation algorithm, there is an instance of T_k that forces the algorithm to read the values on all 2^{2k} leaves.

We now give a simple randomized algorithm and study the expected number of leaves it reads on any instance of T_k. The algorithm is motivated by the following simple observation. Consider a single AND node with two leaves. If the node were to return 0, at least one of the leaves must contain 0. A deterministic algorithm inspects the leaves in a fixed order, and an adversary can therefore always "hide" the 0 at the second of the two leaves inspected by the algorithm. Reading the leaves in a random order foils this strategy. With probability 1/2, the algorithm chooses the hidden 0 on the first step, so its expected number of steps is 3/2, which is better than the worst case for any deterministic algorithm. Similarly, in the case of an OR node, if it were to return a 1 then a randomized order of examining the leaves will reduce the expected number of steps is 3/2. We now extend this intuition and specify the complete algorithm.

To evaluate an AND node v, the algorithm chooses one of its children (a subtree rooted at an OR node) at random and evaluates it by recursively invoking the algorithm. If 1 is returned by the subtree, the algorithm proceeds to evaluate the other child (again by recursive application). If 0 is returned, the algorithm returns 0 for v. To evaluate an OR node, the procedure is the same with the roles of 0 and 1 interchanged. We establish by induction on k that the expected cost of evaluating any instance of T_k is at most 3^k.

The basis ($k = 0$) is trivial. Assume now that the expected cost of evaluating any instance of T_{k-1} is at most 3^{k-1}. Consider first a tree T whose root is an OR node, each of whose children is the root of a copy of T_{k-1}. If the root of T were to evaluate to 1, at least one of its children returns 1. With probability 1/2 this child is chosen first, incurring (by the inductive hypothesis) an expected cost of at most 3^{k-1} in evaluating T. With probability 1/2 both subtrees are evaluated, incurring a net cost of at most $2 \times 3^{k-1}$. Thus the expected cost of determining the value of T is

$$\le \frac{1}{2} \times 3^{k-1} + \frac{1}{2} \times 2 \times 3^{k-1} = \frac{3}{2} \times 3^{k-1}. \qquad (3.1)$$

If on the other hand the OR were to evaluate to 0 both children must be evaluated, incurring a cost of at most $2 \times 3^{k-1}$.

Consider next the root of the tree T_k, an AND node. If it evaluates to 1, then both its subtrees rooted at OR nodes return 1. By the discussion in the previous paragraph and by linearity of expectation, the expected cost of evaluating T_k to 1 is at most $2 \times (3/2) \times 3^{k-1} = 3^k$. On the other hand, if the instance of T_k evaluates to 0, at least one of its subtrees rooted at OR nodes returns 0. With probability 1/2 it is chosen first, and so the expected cost of evaluating T_k is at most

$$2 \times 3^{k-1} + \frac{1}{2} \times \frac{3}{2} \times 3^{k-1} \le 3^k.$$

Theorem 3.1. *Given any instance of T_k, the expected number of steps for the above randomized algorithm is at most 3^k.*

Since $n = 4^k$ the expected running time of our randomized algorithm is $n^{\log_4 3}$, which we bound by $n^{0.793}$. Thus the expected number of steps is smaller than the worst case for any deterministic algorithm. Note that this is a Las Vegas algorithm and always produces the correct answer.

4. The Minimax Principle and Lower Bounds

The Las Vegas randomized algorithm of the preceding section has an expected running time of $n^{0.793}$ on any uniform binary AND-OR tree with n leaves. Can we establish that no randomized algorithm can have a lower expected running time? We first introduce a standard technique for proving such lower bounds. The technique derives from classical game theory; its application to lower bounds for randomized algorithms is due to Yao [49]. This technique applies only to algorithms that terminate in finite time on all inputs and on all random choices.

The key idea is to relate the running times of randomized algorithms for a problem to the running times of deterministic algorithms for the problem when faced with randomly chosen inputs. Consider a problem where the number of distinct inputs of a fixed size is finite, as is the number of distinct (deterministic, terminating and always correct) algorithms for solving that problem. Let us define the *distributional complexity* of the problem at hand as the expected running time of the best deterministic algorithm for the worst distribution on the inputs. Thus we envision an adversary choosing a probability distribution on the set of possible inputs, and study the best deterministic algorithm for this distribution. Let p denote a probability distribution on the set $\mathcal{I}$ of inputs. Let the random variable $C(I_p, A)$ denote the running time of deterministic algorithm $A \in \mathcal{A}$ on an input chosen according to p. Viewing a randomized algorithm as a probability distribution q on the set $\mathcal{A}$ of deterministic algorithms, we let the random variable $C(I, A_q)$ denote the running time of this randomized algorithm on the worst-case input.

Proposition 4.1 (Yao's Minimax Principle). *For all distributions p over $\mathcal{I}$ and q over $\mathcal{A}$,*

$$\min_{A \in \mathcal{A}} E[C(I_p, A)] \le \max_{I \in \mathcal{I}} E[C(I, A_q)].$$

Stated alternatively, the expected running time of the optimal deterministic algorithm for an arbitrarily chosen input distribution p is a lower bound on the expected running time of the optimal (Las Vegas) randomized algorithm for Π. Thus, to prove a lower bound on the randomized complexity, it suffices to choose any distribution p on the input and prove a lower bound on the expected running time of deterministic algorithms for that distribution. The power of this technique lies in the flexibility in the choice of p and, more importantly, the reduction to a lower bound on deterministic algorithms. It is important to remember that the deterministic algorithm "knows" the chosen distribution p.

The above discussion dealt only with lower bounds on the performance of Las Vegas algorithms. We briefly discuss Monte Carlo algorithms with error probability $\epsilon \in [0, 1/2]$. Let us define the distributional complexity with error ϵ, denoted $\min_{A\in\mathcal{A}} E[C_\epsilon(I_p, A)]$, to be the minimum expected running time of any deterministic algorithm that errs with probability at most ϵ under the input distribution p. Similarly, we denote by $\max_{I\in\mathcal{I}} E[C_\epsilon(I, A_q)]$ the expected running time (under the worst input) of any randomized algorithm that errs with probability at most ϵ (again, the randomized algorithm is viewed as a probability distribution q on deterministic algorithms). Analogous to Proposition 4.1, we then have:

Proposition 4.2. *For all distributions p over $\mathcal{I}$ and q over $\mathcal{A}$ and any $\epsilon \in [0, 1/2]$,*

$$\frac{1}{2}\left(\min_{A\in\mathcal{A}} E[C_{2\epsilon}(I_p, A)]\right) \le \max_{I\in\mathcal{I}} E[C_\epsilon(I, A_q)].$$

4.1 Lower Bound for Game Tree Evaluation

We now apply Yao's Minimax Principle to the AND-OR tree evaluation problem. A randomized algorithm for AND-OR tree evaluation can be viewed as a probability distribution over deterministic algorithms, because the length of the computation as well as the number of choices at each step are both finite. We may as well imagine that all of these coins are tossed before the beginning of the execution.

The tree T_k is equivalent to a balanced binary tree all of whose leaves are at distance $2k$ from the root, and all of whose internal nodes compute the NOR function: a node returns the value 1 if both inputs are 0, and 0 otherwise. We proceed with the analysis of the tree of NORs of depth $2k$.

Let $p = (3 - \sqrt{5})/2$; each leaf of the tree is independently set to 1 with probability p. If each input to a NOR node is independently 1 with probability p, its output is 1 with probability

$$\left(\frac{\sqrt{5}-1}{2}\right)^2 = \frac{3-\sqrt{5}}{2} = p.$$

Thus the value of every node of the NOR tree is 1 with probability p, and the value of a node is independent of the values of all the other nodes on the same level. Consider a deterministic algorithm that is evaluating a tree furnished with such random inputs; let v be a node of the tree whose value the algorithm is trying to determine. Intuitively, the algorithm should determine the value of one child of v before inspecting any leaf of the other subtree. An alternative view of this process is that the deterministic algorithm should inspect leaves visited in a depth-first search of the tree, except of course that it ceases to visit subtrees of a node v when the value of v has been determined. Let us call such an algorithm a *depth-first pruning algorithm*, referring to the order of traversal and the fact that subtrees that supply no additional information are "pruned" away without being inspected. The following result is due to Tarsi [41].

Proposition 4.3. *Let T be a NOR tree each of whose leaves is independently set to 1 with probability q for a fixed value $q \in [0, 1]$. Let $W(T)$ denote the minimum, over all deterministic algorithms, of the expected number of steps to evaluate T. Then, there is a depth-first pruning algorithm whose expected number of steps to evaluate T is $W(T)$.*

Proposition 4.3 tells us that for the purposes of our lower bound, we may restrict our attention to depth-first pruning algorithms. Let $W(h)$ be the expected number of leaves inspected by a depth-first pruning algorithm in determining the value of a node at distance h from the leaves, when each leaf is independently set to 1 with probability $(3 - \sqrt{5})/2$. Clearly

$$W(h) = W(h-1) + (1-p) \times W(h-1),$$

where the first term represents the work done in evaluating one of the subtrees of the node, and the second term represents the work done in evaluating the other subtree (which will be necessary if the first subtree returns the value 0, an event occurring with probability $1 - p$). Letting h be $\log_2 n$ and solving, we get $W(h) \ge n^{0.694}$.

Theorem 4.4. *The expected running time of any randomized algorithm that always evaluates an instance of T_k correctly is at least $n^{0.694}$, where $n = 2^{2k}$ is the number of leaves.*

Why is our lower bound of $n^{0.694}$ less than the upper bound of $n^{0.793}$ that follows from Theorem 3.1? The reason is that we have not chosen the best possible probability distribution for the values of the leaves. Indeed, in the NOR tree if both inputs to a node are 1, no reasonable algorithm will read leaves of both subtrees of that node. Thus, to prove the best lower bound we have to choose a distribution on the inputs that precludes the event that both inputs to a node will be 1; in other words, the values of the inputs are chosen at random but not independently. This stronger (and considerably harder) analysis can in fact be used to show that the algorithm of Section 3 is optimal; the reader is referred to the paper of Saks and Wigderson [34] for details.

5. The Probabilistic Method

As we saw in the last chapter, the probabilistic method is a technique for proving the existence of combinatorial objects satisfying a set of desired properties. The idea is to set up a probability space and show that an object drawn from this space will satisfy all the specified properties with nonzero probability. We exemplify this technique using a result on *conference scheduling* due to [illegible] and Raghavan [illegible].

Consider a conference in which n talks are organized into two "parallel sessions" of $n/2$ talks each. An attendee wishing to see cn random talks is likely to encounter a number of conflicts — times at which the two concurrent talks are both of interest to her — whose expectation is c^2n. When c is a constant, this represents a loss of a constant fraction of talks of interest to the attendee. Consider instead the following alternative proposal. Suppose instead of two parallel sessions we have *four* sessions, with each talk given twice. We show (using the probabilistic method) that for any number of attendees up to n^2, each wishing to see no more αn talks (for α a sufficiently small constant), there is a scheduling of talks into four sessions such that every attendee will be able to see all their desired talks.

Suppose in fact that we have as many as n^2 attendees, each with a list of αn talks they wish to see. Now consider a random conference schedule with four parallel tracks, designed as follows. Sessions 1 and 3 each have $n/2$ talks (and thus contain one rendition of each of the n talks) and are designed by the Program Committee in any manner at all (even adversarially, knowing what the attendees want to see). Session 2 is a random permutation of session 1, and session 4 is a random permutation of session 3. (So, the n talks are still being given over a period of $n/2$ time-slots.) We argue that with probability $1 - o(1)$ for this schedule, every one of the n^2 attendees will be able to see all their desired talks. Since a random schedule is good by this measure with positive probability, we conclude that for any set of up to n^2 attendees, there is a schedule that is good by this measure. Indeed, since this probability is close to 1, it follows that almost all schedules from this probability space are good.

A convenient way to view a conference schedule is as a bipartite graph. Each talk is represented by a node on the left, each time-slot is represented by a node on the right, and there is an edge between a talk and a time-slot if that talk is being presented in that time-slot.

We will say that a set of talks S suffers a compression if $|N(S)| < |S|$ where $N(S)$ represents the neighborhood of the nodes in S. Note that by Hall's Theorem, a set of talks S has no conflicts if and only if no $S' \subseteq S$ suffers a compression. We state our main theorem in more general terms than above: the number of attendees in the statement is only bounded by some polynomial function of n. The specialization to the case of n^2 attendees is straightforward, and yields a concrete lower bound on the constant α. We leave its calculation as an exercise for the reader.

Theorem 5.1. *For any polynomial $p(n)$ there exists a constant $\alpha > 0$ such that if $p(n)$ attendees each wish to see αn talks, then with probability $1 - o(1)$, the randomized scheduling method described above allows all attendees to see all their desired talks.*

The analysis proceeds in two steps. We first consider small sets of talks, showing that with "reasonably" high probability, all sets of at most $\frac{1}{6}n$ talks can be seen without conflict. We then consider large sets, and show that for any *fixed* set of at most αn talks, with high probability no subset of it suffers a compression. These together give our desired result.

Lemma 5.2. *Let E_k be the event that some set of at most k talks suffers a compression.* Then $\Pr[E_k] \le$ [illegible]

Proof. Consider a fixed set S of k talks, with k_1 talks in session 1 and $k_2 = k - k_1$ talks in session 3. Let c_1 be the number of time-slots occupied by these talks in sessions 1 and 2 combined. (So, [illegible].) Then,

$$\Pr[S \text{ is compressed}] \le \text{[illegible]} \qquad (5.1)$$

The number of different sets of talks S occupying c_1 time-slots in sessions 1 and 2 is at most [illegible]. Therefore (using that for a given k_1 our bound is increasing with c_1)

$$\Pr[E_k] \le \sum \text{[illegible]}$$

where the last step uses the inequality $a^a b^b \ge ((a+b)/2)^{a+b}$. This gives us our desired bound. □

Lemma 5.3. *For a fixed set S of size k, the probability that some subset of S of size at least k suffers a compression is at most* $\frac{1}{k}(16\epsilon^4)^{k/2}\left(\frac{1}{1-16\epsilon^4}\right)$.

Proof. The probability that a fixed set $S \subseteq T$ of k talks suffers a compression, given that the talks of S use up k_1 timeslots in sessions 1 and 2, is at most the quantity given in Equation (5.1). The number of sets $S \subseteq T$ using k_1 timeslots in sessions 1 and 2 is at most $\binom{m}{k_1}2^{2k_1}$. Therefore the probability that some set $S \subseteq T$ using k_1 timeslots in sessions 1 and 2 (and having at most $2k_1$ talks total) suffers a compression is at most $\frac{1}{k}\left(16\epsilon^4\right)^{k_1}$. Thus, the probability that any $S \subseteq T$ with at least k talks suffers a compression is at most

$$\sum_{k_1=k/2}^{m} \frac{(16\epsilon^4)^{k_1}}{k} \le \frac{(16\epsilon^4)^{k/2}}{k}\left(\frac{1}{1-16\epsilon^4}\right). \qquad \Box$$

Proof of Theorem 5.1. Lemma 5.2 implies that with probability $1 - o(1)$, no set of size $\le \frac{1}{6}m$ is compressed. Now, set $p(m) = O(m^\beta)$ for some constant β. Choose $\epsilon = \frac{1}{16}e^{-4-2\beta}$ so that $(16e\epsilon^4)^{(\log m)/2} = o(m^{-\beta})$. Lemma 5.3 implies that with probability $1 - o(1)$, no subset of size $\ge \frac{1}{6}m$ in any of the $p(m)$ sets of desired talks suffers a compression either. □

One might hope to improve on Theorem 5.1 (and Lemma 5.2) by producing a schedule such that every set of k talks can be seen without conflicts, for $k \ge \log n$. However, the following simple argument shows that this is not possible.

Theorem 5.4. *For any schedule of n talks into 2 sessions such that each talk is given twice, there exists a set S of O(log n) talks that conflict under a compression.*

Proof. Consider a graph with a vertex for each time slot, and where a talk scheduled in time-slots i and j is represented as an edge from i to j. This graph has degree 4. Pick some arbitrary vertex in the graph and grow a breadth-first search tree from that node until at least two back-edges are observed. (An edge from a node to itself — i.e., a talk given in only one time-slot — counts as a back-edge.) This must occur by the time the tree has grown to depth $\lg n$ because the degree of the graph is at least 3. Consider now the two cycles induced by these two back edges. If the cycles touch (or overlap), then the union of the two cycles is our desired set S. If the cycles do not touch, then the two cycles together with the path in the tree between them (which has length at most $2\lg n$) is our desired set. □

What if we allow each talk to be given 3 times? In this case, standard arguments (along the lines of the proof of Lemma 5.2) show that the bipartite graph will with high probability be an expander, and therefore all sets of $o(n)$ talks have the property that they can be seen without conflict. Once we have created a schedule at random, how do we verify whether it is good for a set of attendees? And how does each attendee decide which of the two renditions of each interesting talk to see in order to ensure that she sees all the talks of interest to her? These questions, and other extensions, can be found in [5].

6. Algebraic Methods and Randomized Fingerprints

We now turn to a discussion of the randomized fingerprinting technique, due to Freivalds [13], for the verification of identities involving matrices, polynomials, and integers. We also describe how this generalizes to the so-called Schwartz-Zippel technique for identities involving multivariate polynomials (independently due to Schwartz [33] and Zippel [37]; see also DeMillo and Lipton [8]). Finally, following Lovász [22], we apply the technique to the problem of detecting the existence of perfect matchings in graphs.

The *fingerprinting* technique has the following general form. Suppose we wish to check the equality of two elements x and y drawn from some "large" universe U. Under any reasonable model of computation, this problem has a deterministic complexity $\Omega(\log |U|)$. Employing randomization, an alternative approach is to choose a random function from U into a smaller space V such that with high probability x and y are identical if and only if their images in V are identical. These images of x and y are said to be their *fingerprints*, and the equality of fingerprints can be verified in time $O(\log |V|)$.

The obvious problem with the fingerprinting technique is that the average number of elements of U mapped to an element of V is $|U|/|V|$. Given this, it seems difficult, if not impossible, to find good fingerprint functions that work for arbitrary or worst-case choices of x and y. However, as we will show below, when the identity-checking is only required to be correct for x and y chosen from a small subspace S of U, particularly a subspace with some well-defined algebraic structure, it is possible to choose good fingerprint functions without any a priori knowledge of the subspace, provided the size of V is chosen to be comparable to the size of S.

Throughout this section we will be working over some unspecified field $\mathcal{F}$. Since the randomization will involve uniform sampling from a finite subset of the field, we do not even need to specify whether the field is finite or not. The reader may find it helpful in the infinite case to assume that $\mathcal{F}$ is the field $\mathbb{Q}$ of rational numbers, and in the finite case to assume that $\mathcal{F}$ is Z_p, the field of integers modulo some prime number p.

6.1 Freivalds' Technique and Matrix Product Verification

We begin with the problem of verifying the correctness of matrix product identities. Currently, the fastest algorithm for matrix multiplication (Coppersmith and Winograd [7]) has running time $O(n^{2.376})$, improving significantly on the obvious $O(n^3)$ time algorithm; however, the fast matrix multiplication algorithm has the disadvantage of being extremely complicated. Suppose we have an implementation of the fast matrix multiplication algorithm and, given its complex nature, are unsure of its correctness. Since program verification appears to be an intractable problem, we consider the more reasonable goal of verifying the correctness of the output produced by executing the algorithm on specific inputs. This notion of verifying programs on specific inputs is the basic tenet of the theory of program checking recently formulated by Blum and Kannan [4].

Suppose we are given three $n \times n$ matrices X, Y and Z over a field $\mathcal{F}$ and would like to verify that $XY = Z$. Clearly, it does not make sense to use a simpler but slower matrix multiplication algorithm for the verification, as that would defeat the whole purpose of using the fast algorithm in the first place. In fact, there is no need to recompute Z; indeed, we are merely required to verify that the product of X and Y is equal to Z. Freivalds' technique gives an elegant solution that leads to an $O(n^2)$ time randomized algorithm with bounded error probability.

We choose a random vector $r \in \{0,1\}^n$, i.e., each component of r is chosen independently and uniformly at random from the set $\{0,1\}$ consisting of the additive and multiplicative identities of the field $\mathcal{F}$. Then, in $O(n^2)$ time, we can compute $y = Yr$, $x = Xy = XYr$, and $z = Zr$. We would like to claim that the identity $XY = Z$ can be verified by merely checking that $x = z$. Quite clearly, if $XY = Z$ then $x = z$; unfortunately, the converse is not true in general. However, given the random choice of r, we can show that for $XY \neq Z$, the probability that $x \neq z$ is at least $1/2$. Note that the fingerprinting algorithm errs only if $XY \neq Z$ but x and z turn out to be equal, and this has a bounded probability.

Theorem 6.1 *Let X, Y and Z be $n \times n$ matrices over some field $\mathcal{F}$ such that $XY \neq Z$; further, let r be chosen uniformly at random from $\{0,1\}^n$ and define $x = XYr$ and $z = Zr$. Then,*

$$\Pr[x = z] \leq 1/2.$$

Proof: Let $W = XY - Z$ and observe that W is not the all-zeroes matrix. Since $Wr = XYr - Zr = x - z$, the event $x = z$ is equivalent to the event that $Wr = 0$. Assume, without loss of generality, that the first row of W has a non-zero entry and that the non-zero entries in that row precede all the zero entries. Define the vector w as the first row of W, and assume that the first $k > 0$ entries in w are non-zero. Since the first component of Wr is $w^T r$, giving an upper bound on the probability that the inner product of w and r is zero will give an upper bound on the probability that $x = z$.

Clearly, $w^T r = 0$ if and only if

$$r_1 = \frac{-\sum_{i=2}^{k} w_i r_i}{w_1}. \tag{6.1}$$

Assume, without loss of generality, that in choosing the random vector r we select $r_2, \ldots, r_n$ before picking r_1. Once the values for $r_2, \ldots, r_n$ have been determined, the right hand side of (6.1) is fixed at some value $v \in \mathcal{F}$. If $v \notin \{0,1\}$, then r_1 will never equal v; conversely, if $v \in \{0,1\}$, then the probability that $r_1 = v$ is $1/2$. Clearly, the probability that $w^T r = 0$ is at most $1/2$, which gives us the desired result. □

In essence, the fingerprinting technique reduces the matrix multiplication verification problem to that of verifying the equality of two vectors. The reduction itself can be performed in $O(n^2)$ time and vector equality can be checked in $O(n)$ time, giving an overall running time of $O(n^2)$ for this Monte Carlo procedure. The error probability can be reduced to $1/2^k$ via k independent iterations of the Monte Carlo algorithm. There was nothing sacrosanct about choosing the components of the random vector r from $\{0,1\}$, since any two distinct elements of $\mathcal{F}$ would have done equally well. This suggests an alternative approach towards reducing the error probability, as follows: each component of r is chosen independently and uniformly at random from some subset S of the field $\mathcal{F}$; then, it is easily verified that the error probability is no more than $1/|S|$.

In general, Freivalds' technique can be applied to the verification of any matrix identity $A = B$. Of course, given A and B, just comparing their entries takes only $O(n^2)$ time. But there are many situations where, just as in the case of matrix product verification, computing A explicitly is either too expensive or possibly even impossible, whereas computing Ar is easy. The random fingerprint technique is an elegant solution in such settings.

6.2 Extension to Identities of Polynomials

Freivalds' fingerprinting technique is quite general and can be applied to many different versions of the identity verification problem. We show that it can be applied to identity verification for symbolic polynomials, where two polynomials $P_1(x)$ and $P_2(x)$ are deemed identical if they have identical coefficients for corresponding powers of x. Observe that verifying integer or string equality is a special case, since we can represent any string of length n as a polynomial of degree n by using the kth element in the string to determine the coefficient of the kth power of a symbolic variable.

We define the polynomial product verification problem as follows: given three polynomials $P_1(x), P_2(x), P_3(x) \in \mathcal{F}[x]$, we are required to verify that $P_1(x) \times P_2(x) = P_3(x)$. We will assume that $P_1(x)$ and $P_2(x)$ are of degree at most n, implying that $P_3(x)$ has degree at most $2n$. It is well-known that degree n polynomials can be multiplied in $O(n \log n)$ time via Fast Fourier Transforms, and that the evaluation of a polynomial requires only $O(n)$ time.

We present a randomized algorithm for polynomial product verification which is similar in spirit to the matrix product verification algorithm. First, fix a set $S \subseteq \mathcal{F}$ of size at least $2n+1$ and choose $r \in S$ uniformly at random. Then, after evaluating $P_1(r)$, $P_2(r)$ and $P_3(r)$ in $O(n)$ time, the algorithm declares the identity $P_1(x)P_2(x) = P_3(x)$ to be correct if and only if $P_1(r)P_2(r) = P_3(r)$. The algorithm errs only in the case where the polynomial identity is false but the value of the three polynomials at r indicates otherwise. We will show that the error event has bounded probability.

Let us define the degree $2n$ polynomial $Q(x) = P_1(x)P_2(x) - P_3(x)$. We say that a polynomial $Q(x)$ is identically zero, denoted by $Q(x) \equiv 0$, if each of its coefficients equals zero. The polynomial identity $P_1(x)P_2(x) = P_3(x)$ is valid if and only if $Q(x) \equiv 0$. It remains to establish that if $Q(x) \not\equiv 0$, then with high probability $Q(r) = P_1(r)P_2(r) - P_3(r) \neq 0$. By elementary algebra we know that $Q(x)$ has at most $2n$ distinct roots. Clearly, unless $Q(x) \equiv 0$, no more that $2n$ different choices of $r \in S$ will cause $Q(r)$ to evaluate to 0. Thus, the error probability is at most $2n/|S|$. We may reduce the error probability either by using independent iterations of this algorithm, or by choosing a larger set S.

It turns out that the above verification technique can be easily extended to a generic procedure for testing any polynomial identity of the form $P_1(x) = P_2(x)$ by converting it into the identity $Q(x) = P_1(x) - P_2(x) \equiv 0$. Certainly, when P_1 and P_2 are explicitly provided, the identity can be deterministically verified in $O(n)$ time by comparing corresponding coefficients. Our randomized technique will take just as long to merely evaluate $P_1(x)$ and $P_2(x)$ at a random value. But, as in the case of verifying matrix identities, the randomized algorithm is very useful in situations where the polynomials are implicitly specified, e.g., when we only have a "black box" for computing the polynomials with no information about their coefficients, or when they are provided in a form where computing the actual coefficients is expensive. One example of the latter situation is provided by the following problem of verifying the determinant of a symbolic matrix. As will soon become obvious, the determinant problem will in fact require a technique for the verification of polynomial identities of multivariate polynomials and therefore we will need to provide a generalization to that setting.

Let M be an $n \times n$ matrix. The determinant of the matrix M is defined as follows:

$$\det(M) = \sum_{\pi \in S_n} \mathrm{sgn}(\pi) \prod_{i=1}^{n} M_{i,\pi(i)}, \qquad (6.2)$$

where S_n is the symmetric group of permutations of order n, and $\mathrm{sgn}(\pi)$ is the sign[1] of a permutation π. While the determinant is defined as a summation with $n!$ terms, it turns out that it is easily evaluated in polynomial time provided the matrix entries M_{ij} are explicitly specified. The situation is more complicated when the matrix entries are not explicit constants, as we illustrate next.

Consider the Vandermonde matrix $M(x_1,\ldots,x_n)$ which is defined in terms of the indeterminates $x_1,\ldots,x_n$ such that $M_{ij} = x_i^{j-1}$, i.e.,

$$M = \begin{pmatrix} 1 & x_1 & x_1^2 & \ldots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \ldots & x_2^{n-1} \\ & & & \vdots & \\ 1 & x_n & x_n^2 & \ldots & x_n^{n-1} \end{pmatrix}.$$

It is known that for the Vandermonde matrix, $\det(M) = \prod_{i<j}(x_i - x_j)$. Consider the problem of verifying this identity without actually devising a formal proof for a fixed value of n. Computing the determinant of a symbolic matrix is infeasible as it requires dealing with a summation over $n!$ terms. However, we can formulate the identity verification problem as the problem of verifying that the polynomial $Q(x_1,\ldots,x_n) = \det(M) - \prod_{i<j}(x_i - x_j)$ is identically zero. Based on our discussion of Freivalds' technique, it is natural to consider the substitution of random values for each x_i. Since the determinant can be computed in polynomial time for any specific assignment of values to the symbolic variables $x_1,\ldots,x_n$, it is easy to evaluate the polynomial Q for random values of the variables. The only issue is that of bounding the error probability for this randomized test.

[1] The sign function is defined to be $\mathrm{sgn}(\pi) = (-1)^t$, where t is the number of pairwise exchanges required to convert the identity permutation into π.

We now turn to the extension to the multivariate case of the analysis of Freivalds' technique as applied to univariate polynomials. Note that in a multivariate polynomial $Q(x_1,\ldots,x_n)$, the degree of a term is the sum of the exponents of the variable powers that define it, and the total degree of Q is the maximum over all terms of the degrees of the terms.

Theorem 6.2. *Let $Q(x_1,\ldots,x_n) \in F[x_1,\ldots,x_n]$ be a multivariate polynomial of total degree m. Let S be a finite subset of the field F, and let $r_1,\ldots,r_n$ be chosen uniformly and independently from S. Then*

$$\Pr[Q(r_1,\ldots,r_n)=0 \mid Q(x_1,\ldots,x_n)\not\equiv 0] \le \frac{m}{|S|}.$$

Proof. The proof is by induction on the number of variables n. The base case of the induction is $n = 1$, which reduces to verifying the theorem for a univariate polynomial $Q(x_1)$ of degree m. But we have already seen for $Q(x_1)\not\equiv 0$, the probability that $Q(r_1)=0$ is at most $m/|S|$, taking care of the base.

Suppose now that the induction hypothesis holds for multivariate polynomials with at most $n-1$ variables, where $n > 1$. In the polynomial $Q(x_1,\ldots,x_n)$ we can factor out the variable x_1 and thereby express Q as

$$Q(x_1,\ldots,x_n)=\sum_{i=0}^{k} x_1^i P_i(x_2,\ldots,x_n),$$

where $k\le m$ is the largest exponent of x_1 in Q. Given our choice of k, the coefficient $P_k(x_2,\ldots,x_n)$ of x_1^k cannot be identically zero. Note that the total degree of P_k is at most $m-k$. Thus, by the induction hypothesis, we conclude that the probability that $P_k(r_2,\ldots,r_n)=0$ is at most $(m-k)/|S|$.

Let us now turn to the case where $P_k(r_2,\ldots,r_n)$ is not equal to 0. Consider the following univariate polynomial over x_1 obtained by substituting the random values for the other variables in Q:

$$q(x_1)=Q(x_1,r_2,r_3,\ldots,r_n)=\sum_{i=0}^{k} x_1^i P_i(r_2,\ldots,r_n).$$

The resulting polynomial $q(x_1)$ has degree k and is not identically zero (since the coefficient of x_1^k is assumed to be non-zero). As in the base case, we conclude that the probability that $q(r_1)=Q(r_1,r_2,\ldots,r_n)$ evaluates to 0 is bounded by $k/|S|$.

We have established the following two inequalities:

$$\Pr[P_k(r_2,\ldots,r_n)=0]\le\frac{m-k}{|S|};$$

and

$$\Pr[Q(r_1,r_2,\ldots,r_n)=0 \mid P_k(r_2,\ldots,r_n)\ne 0]\le\frac{k}{|S|}.$$

Observe that for any two events $\mathcal{E}_1$ and $\mathcal{E}_2$, $\Pr[\mathcal{E}_1]\le\Pr[\mathcal{E}_1\mid\overline{\mathcal{E}_2}]+\Pr[\mathcal{E}_2]$. Consequently, we obtain that the probability that $Q(r_1,r_2,\ldots,r_n)=0$ is no more than the sum of the two probabilities on the right hand side of the two inequalities displayed above, and this turns out to be $m/|S|$. □

There is one major disadvantage to the randomized verification procedure just discussed. In large (or possibly infinite) fields, the evaluation of the polynomials could involve large intermediate values, leading to inefficient implementation. To deal with this problem in the case of integers, we perform all computations modulo a random prime number chosen from a suitable range. It is easy to verify that this does not have any adverse effect on the error probability.

6.3 Detecting Perfect Matchings in Graphs

We now present an interesting application of the techniques from the preceding section. Consider a bipartite graph $G(U,V,E)$ with two independent sets of vertices $U=\{u_1,\ldots,u_n\}$ and $V=\{v_1,\ldots,v_n\}$, such that the edges in E have one end-point each in U and V. A matching in G is a collection of edges $M\subseteq E$ such that each vertex is an end-point of at most one edge in M. A perfect matching is a matching of size n, i.e., where each vertex occurs as an end-point of exactly one edge in M. Perfect matchings are in a 1-to-1 correspondence with the permutations in S_n, where the matching corresponding to a permutation $\pi\in S_n$ is given by the collection of edges $\{(u_i,v_{\pi(i)}) \mid 1\le i\le n\}$. It turns out that there is an intimate relationship between matchings in a graph and the determinant of a matrix obtained from the graph.

Theorem 6.3. *For any bipartite graph $G(U,V,E)$, define a corresponding $n\times n$ matrix A as follows:*

$$A_{ij}=\begin{cases}x_{ij} & (u_i,v_j)\in E\\ 0 & (u_i,v_j)\notin E\end{cases}$$

Let the multivariate polynomial $Q(x_{11},x_{12},\ldots,x_{nn})$ denote the determinant $\det(A)$. Then, G has a perfect matching if and only if $Q\not\equiv 0$.

Proof. The determinant of A may be represented as follows:

$$\det(A) = \sum_{\pi \in S_n} \mathrm{sgn}(\pi) A_{1,\pi(1)} A_{2,\pi(2)} \cdots A_{n,\pi(n)}$$

There cannot be any cancellation of the terms in the summation since each x_{ij} occurs at most once in A. It follows that the determinant is not identically zero if and only if there exists some permutation π for which the corresponding term in the summation is non-zero. The term corresponding to a permutation π is non-zero if and only if $A_{i,\pi(i)} \neq 0$ for each i, $1 \leq i \leq n$; this is equivalent to the presence in G of the perfect matching corresponding to π. ∎

The matrix of indeterminates is the *Edmonds matrix* of a bipartite graph. The above result can be extended to the case of non-bipartite graphs, and the corresponding matrix of indeterminates is called the Tutte matrix. Tutte [42] was the first to point out the relationship between matchings and determinants, while the simpler relation between bipartite matchings and determinants was given by Edmonds [9].

The result described above leads to a simple randomized procedure for testing the existence of perfect matchings in a bipartite graph (due to Lovász [22]): using the algorithm from Section 6.2, determine whether the determinant is identically zero or not. The running time of this procedure is dominated by the cost of computing a determinant, which is essentially the same as the time required to multiply two matrices. Of course, there are algorithms for constructing a maximum matching in a graph with m edges and n vertices in time $O(m\sqrt{n})$ (see Hopcroft and Karp [15], Micali and Vazirani [24, 44], and Feder and Motwani [10]). Given that the time required to compute the determinant exceeds $m\sqrt{n}$ for small m, the benefit in using this randomized decision procedure appears marginal at best. But this technique was extended by Rabin and Vazirani [32, 33] to obtain simple algorithms for the actual construction of maximum matchings; although their randomized algorithms for matchings are simple and elegant, they are still slower than the deterministic $O(m\sqrt{n})$ time algorithms known earlier. Perhaps more significantly, this randomized decision procedure proved to be an essential ingredient in devising fast parallel algorithms for computing maximum matchings [20, 29].

7. Further Reading

We conclude by giving some pointers to the (large) number of randomized algorithms not covered here. It should be noted that the examples we discussed here are but a mere sampling of the many randomized algorithms for each of the problems considered. The algorithms covered were chosen to illustrate the ideas rather than to represent the state of the art for these problems. The interested reader is referred to the book [27] for a discussion of other algorithms for these problems.

Randomized algorithms have found application in a large number of areas: in load-balancing [43], approximation algorithms and combinatorial optimization [13, 19, 25], graph algorithms [1, 37], data structures [2], counting and enumeration [28], parallel algorithms [21, 31], distributed algorithms [36], geometric algorithms [27], online algorithms [3, 6] and number-theoretic algorithms [30, 17]. The interested reader should consult these articles or the book [27].

References

1. Aleliunas R., Karp R.M., Lipton R.J., Lovász L., and Rackoff C. (1979): Random walks, universal traversal sequences, and the complexity of maze problems, in Proceedings of the 20th Annual Symposium on Foundations of Computer Science, pages 218–223, San Juan, Puerto Rico, October.
2. Aragon C.R. and Seidel R.G. (1989): Randomized search trees, in Proceedings of the 30th Annual IEEE Symposium on Foundations of Computer Science, pages 540–545.
3. Ben-David S., Borodin A., Karp R.M., Tardos G., and Wigderson A. (1994): On the power of randomization in on-line algorithms. Algorithmica 11, (1), 2–14.
4. Blum M. and Kannan S. (1989): Designing programs that check their work, in Proceedings of the 21st Annual ACM Symposium on Theory of Computing, pages 86–97. ACM.
5. Blum A. and Raghavan P. (1998): [illegible]. [illegible] Algorithms.
6. Borodin A. and El-Yaniv R. (1998): On-line Algorithms, Cambridge University Press.
7. Coppersmith D. and Winograd S. (1990): Matrix multiplication via arithmetic progressions. Journal of Symbolic Computation 9, 251–280.
8. DeMillo R.A. and Lipton R.J. (1978): A probabilistic remark on algebraic program testing. Information Processing Letters 7, 193–195.
9. Edmonds J. (1967): Systems of distinct representatives and linear algebra. Journal of Research of the National Bureau of Standards, 71B, 4, 241–245.
10. Feder T. and Motwani R. (1991): Clique partitions, graph compression, and speeding up algorithms, in Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, pages 123–133.
11. Floyd R.W. and Rivest R.L. (1975): Expected time bounds for selection. Communications of the ACM 18, 165–172.
12. Freivalds R. (1977): Probabilistic machines can use less running time, in B. Gilchrist, editor, Information Processing 77, Proceedings of IFIP Congress 77, pages 839–842, Amsterdam, August. North-Holland Publishing Company.

13. Goemans M.X. and Williamson D.P. (1994): .878-approximation algorithms for MAX-CUT and MAX-2SAT. In Proceedings of the 26th Annual ACM Symposium on Theory of Computing, pages 422–431.
14. Hoare C.A.R. (1962): Quicksort. Computer Journal 5, 10–15.
15. Hopcroft J.E. and Karp R.M. (1973): An $n^{5/2}$ algorithm for maximum matching in bipartite graphs. SIAM Journal on Computing 2, 225–231.
16. Karger D.R. (1993): Global min-cuts in RNC, and other ramifications of a simple min-cut algorithm, in Proceedings of the 4th Annual ACM-SIAM Symposium on Discrete Algorithms.
17. Karger D.R., Klein P.N. and Tarjan R.E. (1995): A randomized linear-time algorithm for finding minimum spanning trees. Journal of the ACM 42, 321–328.
18. Karger D., Motwani R. and Sudan M. (1994): Approximate graph coloring by semidefinite programming, in Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science, 2–13.
19. Karp R.M. (1991): An introduction to randomized algorithms, Discrete Applied Mathematics 34, 165–201.
20. Karp R.M., Upfal E. and Wigderson A. (1986): Constructing a perfect matching is in random NC. Combinatorica 6, 35–48.
21. Karp R.M., Upfal E. and Wigderson A. (1988): The complexity of parallel search, Journal of Computer and System Sciences 36, 225–253.
22. Lovász L. (1979): On determinants, matchings and random algorithms, in L. Budach, editor, Fundamentals of Computing Theory. Akademia-Verlag, Berlin.
23. Maffioli F., Speranza M.G. and Vercellis C. (1985): Randomized algorithms, in M. O'hEigeartaigh, J.K. Lenstra and A.H.G. Rinnooy Kan, editors, Combinatorial Optimization: Annotated Bibliographies, 89–105. John Wiley and Sons, New York.
24. Micali S. and Vazirani V.V. (1980): An $O(\sqrt{|V|}|E|)$ algorithm for finding maximum matching in general graphs, in Proceedings of the 21st Annual IEEE Symposium on Foundations of Computer Science, 17–27.
25. Motwani R., Naor J. and Raghavan P. (1996): Randomization in approximation algorithms, in D. Hochbaum, editor, Approximation Algorithms, PWS.
26. Motwani R. and Raghavan P. (1995): Randomized Algorithms. Cambridge University Press, New York.
27. Mulmuley K. (1993): Computational Geometry: An Introduction Through Randomized Algorithms. Prentice Hall, New York.
28. Mulmuley K., Vazirani U.V. and Vazirani V.V. (1987): Matching is as easy as matrix inversion. Combinatorica 7, 105–113.
29. Pugh W. (1990): Skip lists: A probabilistic alternative to balanced trees. Communications of the ACM 33(6), 668–676.
30. Rabin M.O. (1980): Probabilistic algorithm for testing primality. Journal of Number Theory 12, 128–138.
31. Rabin M.O. (1983): Randomized Byzantine generals, in Proceedings of the 24th Annual Symposium on Foundations of Computer Science, 403–409.
32. Rabin M.O. and Vazirani V.V. (1984): Maximum matchings in general graphs through randomization. Technical Report TR-15-84, Aiken Computation Laboratory, Harvard University.
33. Rabin M.O. and Vazirani V.V. (1989): Maximum matchings in general graphs through randomization. Journal of Algorithms 10, 557–567.
34. Saks M. and Wigderson A. (1986): Probabilistic Boolean decision trees and the complexity of evaluating game trees. In Proceedings of the 27th Annual IEEE Symposium on Foundations of Computer Science, 29–38, Toronto, Canada.

35. Schrijver A. (1986): Theory of Linear and Integer Programming. John Wiley, New York.
36. Schwartz J.T. (1980): Fast probabilistic algorithms for verification of polynomial identities. Journal of the ACM 27(4), 701–717, October.
37. Seidel R.G. (1991): Small-dimensional linear programming and convex hulls made easy. Discrete and Computational Geometry 6, 423–434.
38. Sinclair A. (1992): Algorithms for Random Generation and Counting: A Markov Chain Approach. Progress in Theoretical Computer Science. Birkhauser, Boston.
39. Snir M. (1985): Lower bounds on probabilistic linear decision trees. Theoretical Computer Science 38, 69–82.
40. Solovay R. and Strassen V. (1977): A fast Monte-Carlo test for primality. SIAM Journal on Computing 6(1), 84–85, March; see also SIAM Journal on Computing 7, 1 February 1978, 118.
41. Tarsi M. (1983): Optimal search on some game trees. Journal of the ACM 30, 389–396.
42. Tutte W.T. (1947): The factorization of linear graphs. J. of the London Math. Soc. 22, 107–111.
43. Valiant L.G. (1982): A scheme for fast parallel communication. SIAM Journal on Computing 11, 350–361.
44. Vazirani V.V. (1994): A theory of alternating paths and blossoms for proving correctness of $O(\sqrt{V}E)$ graph maximum matching algorithms. Combinatorica 14(1), 71–109.
45. Welsh D.J.A. (1983): Randomized algorithms. Discrete Applied Mathematics 5, 133–145.
46. Yao A.C-C. (1977): Probabilistic computations: Towards a unified measure of complexity. In Proceedings of the 17th Annual Symposium on Foundations of Computer Science, pages 222–227.
47. Zippel R.E. (1979): Probabilistic algorithms for sparse polynomials, in Proceedings of EUROSAM 79, 72 of Lecture Notes in Computer Science, pages 216–226, Marseille.

Mathematical Foundations of the Markov Chain Monte Carlo Method

Mark Jerrum*

Department of Computer Science, University of Edinburgh, The King's Buildings, Edinburgh EH9 3JZ, United Kingdom

Summary. The Markov chain Monte Carlo (MCMC) method exploits the idea that information about a set of combinatorial objects may be obtained by performing an appropriately defined random walk on those objects. In the area of statistical physics, MCMC algorithms have been in use for many years for the purpose of estimating various quantities of physical interest, often expectations of random variables on "configurations" of a statistical model. The running time of MCMC algorithms depends on the rate at which the random walk converges to equilibrium; only when a condition of near-equilibrium has been achieved can the algorithm discover what "typical" objects are like. In the past decade or so it has become possible to derive a priori bounds on the rate of convergence to equilibrium of random walks underlying MCMC algorithms of practical interest. In cases where a priori bounds cannot be derived, it may still be possible to conduct rigorously grounded experiments. Many of the main ideas and techniques are set out here, with the recent developments being discussed at greater length.

1. Introduction

The classical *Monte Carlo method* is an approach to estimating quantities that are hard to compute exactly. The quantity z of interest is expressed as the expectation $z = \mathrm{E}\,Z$ of a random variable (r.v.) Z for which some efficient sampling procedure is available. By taking the mean of some sufficiently large set of independent samples of Z, one may obtain an approximation to z. For example, suppose

$$S = \{(x,y) \in [0,1]^2 : p_i(x,y) \le 0, \text{ for all } i\}$$

is some region of the unit square defined by a system of polynomial inequalities $p_i(x,y) \le 0$. Let Z be the r.v. defined by the following experiment or trial: choose a point (x,y) uniformly at random from $[0,1]^2$; let $Z = 1$ if $p_i(x,y) \le 0$ for all i, and $Z = 0$ otherwise. Then the area a of S is equal to $\mathrm{E}\,Z$, and an estimate of a may be obtained from the sample mean of a sufficiently long sequence of trials. In this example, the use of the Monte Carlo method is perhaps avoidable, at the expense of a more complex algorithm; for more essential uses, see, for example, Knuth's proposal [48] for estimating the size of a tree by taking a random path from the root to a leaf, or Rasmussen's [56] for estimating the permanent of a 0,1-matrix.

The *Markov chain Monte Carlo* (MCMC) method is a development of the foregoing approach, which is sometimes applicable when Z cannot be sampled "directly." Computer scientists approaching this subject with only the most basic probabilistic tools can, for the moment, think of a Markov chain $\mathfrak{M}$ as being a kind of finite automaton, in which the transitions from any state are labelled, not by letters from some alphabet, but by non-negative real numbers (probabilities) summing to 1. The Markov chain $\mathfrak{M}$ starts in a distinguished state x_0 at time 0, and makes a sequence of transitions at successive time-steps, resulting in $\mathfrak{M}$ passing through a sequence of states $X_0 = x_0, X_1, X_2, \ldots$. The transitions are guided by the specified probabilities: if $X_t = x_t$, i.e., $\mathfrak{M}$ is in state x_t after the tth transition, then the probability that $X_{t+1} = x_{t+1}$ is just the number assigned to the transition from state x_t to state x_{t+1}.

Suppose Ω denotes the (finite) state space of $\mathfrak{M}$. The Markov chain $\mathfrak{M}$ will be completely specified if we give the matrix of transition probabilities $(P(x,y) : x,y \in \Omega)$, where, for all pairs of states $x,y \in \Omega$,

$$P(x,y) = \Pr(X_{t+1} = y \mid X_t = x)$$

is the probability that the Markov chain is in state y at time $t+1$, conditioned on it being in state x at time t. Note the crucial "forgetting property" of Markov chains: the state at time $t+1$ depends probabilistically on the state at time t, but not on the state at any earlier time.

Provided a certain technical condition—here called *ergodicity*—is met, $\mathfrak{M}$ will converge to a well-defined *stationary distribution* π. More precisely, there is a probability distribution π on Ω such that $\Pr(X_t = y \mid X_0 = x) \to \pi(y)$ as $t \to \infty$, for all pairs of states $x,y \in \Omega$. Note that the initial state x is "forgotten" by $\mathfrak{M}$ over a sufficiently large number of steps.

So suppose we have a r.v. Z for which no obvious direct sampling procedure exists. The idea behind MCMC is to construct an ergodic Markov chain $\mathfrak{M}$ whose state space is the range of Z (or at least includes the range of Z) and whose stationary distribution matches the probability distribution of Z. Then the required samples are obtained by simulating $\mathfrak{M}$ for sufficiently many steps T from some fixed initial state, and returning the final state. Of course, what we obtain is not a perfect sample from the probability distribution of Z, but if T is large the error will be negligible. Naturally, the determination of a suitable T is a significant concern in rigorous applications of MCMC.

As an example of the approach, we consider the problem of estimating the number of (proper) q-colourings of a graph G. In Section 2 we consider how

* Supported in part by Esprit Working Group No. 21726, "RAND2".

sample q-colourings of G, generated independently and u.a.r., can be used to obtain an estimate for the number of q-colourings of G. This step of the MCMC programme—how samples are used—is often (though not always) rather routine. We therefore leave graph colouring as our representative example and turn from the use of samples to their generation. In Section 3, we show how to design a simple Markov chain on colourings that, given a certain condition on the graph G and the number of colours q, is ergodic and has uniform stationary distribution. Again this step—the design of the Markov chain—is often rather routine.

We then turn to what is the crux, often the sticking point, of the method, namely determining good upper bounds on the "mixing time", i.e., the number of steps before the Markov chain is "close" to the stationary distribution. Section 4 presents three methods for bounding the mixing time in the context of a toy example, namely a Markov chain on q-colourings of the empty graph. Obviously, the toy example is of no practical value, but its very simplicity brings the various techniques into sharp relief. Section 5 applies the same three methods to some more realistic and challenging applications. Most of the material of Sections 2 to 5 can be followed in greater detail (though sometimes with different examples) in the survey article of Jerrum and Sinclair [37].

The remainder of the article deals in greater depth with a topic, namely the coupling method, which has grown in perceived importance since the survey article [37] was written. Coupling is a classical (and elementary) technique for bounding the convergence rate of a Markov chain, but some of us working in the analysis of MCMC algorithms had been guilty of thinking it too weak in practice to be applied to interesting examples. Two recent developments—"coupling from the past" and "path coupling"—are beginning to correct this perception.

2. Approximate Counting, Uniform Sampling and Their Relationship

What do we mean precisely by (efficient) approximate counting and uniform sampling?

Suppose $N : \Sigma^* \to \mathbb{N}$ is a function mapping problem instances (encoded as words over some convenient alphabet Σ) to natural numbers. For example, N might map (an encoding of) a graph G to the number $N(G)$ of perfect matchings in G. It should be clear that any combinatorial enumeration problem can be cast in this framework. A *randomised approximation scheme* for N is a randomised algorithm that takes as input a word (instance) $x \in \Sigma^*$ and an error bound $\varepsilon > 0$, and produces as output a number Y (a random variable) such that[1]

$$\Pr\big((1-\varepsilon)N(x) \le Y \le (1+\varepsilon)N(x)\big) \ge \frac{3}{4}. \qquad (2.1)$$

A randomised approximation scheme is said to be *fully polynomial* [39] if it runs in time polynomial in $|x|$ (the input length) and ε^{-1}. We shall abbreviate the rather unwieldy phrase "fully polynomial randomised approximation scheme" to FPRAS.

Suppose now that $S \subseteq \Sigma^* \times \Sigma^*$ is a relation between (encodings of) problem instances and (encodings of) feasible solutions to that instance. Thus, S might assign to each graph G the set $S(G)$ of perfect matchings in G. We insist that the set $S(x)$ is finite for all x. (The relationship we envisage between S and the counting function N considered earlier is, of course, that $N(x) = |S(x)|$ for all meaningful encodings $x \in \Sigma^*$ of problem instances.) For any probability distribution π on a finite set Ω, we define the total variation distance between π and the uniform as

$$d_{\mathrm{TV}}(\pi) = \max_{A \subseteq \Omega}\left|\pi(A) - \frac{|A|}{|\Omega|}\right| = \frac{1}{2}\sum_{\sigma\in\Omega}\left|\pi(\sigma) - \frac{1}{|\Omega|}\right|.$$

An *almost uniform sampler* for S is a randomised algorithm that takes as input a word (instance) $x \in \Sigma^*$ and a tolerance $\delta > 0$, and produces a feasible solution $Z \in S(x)$ (a random variable) such that the probability distribution of Z is within variation distance δ of the uniform distribution on $S(x)$. An almost uniform sampler is said to be *fully polynomial* if it runs in time polynomial in $|x|$ (the input length) and $\log \delta^{-1}$.

There is a close connection between almost uniform sampling and approximate counting, which has been discussed at some length by Jerrum, Valiant, and Vazirani [40]. In brief, provided a certain technical condition known as self-reducibility is met, almost uniform sampling is possible in polynomial time if and only if approximate counting is. Here is a possible way to make the connection concrete in the case of graph colourings.

Proposition 2.1. *Suppose we have an almost uniform sampler for q-colourings of a graph, which works for graphs G with maximum degree bounded*

[1] There is no significance in the constant $\frac{3}{4}$ appearing in the definition, beyond its lying strictly between $\frac{1}{2}$ and 1. Any success probability greater than $\frac{1}{2}$ may be boosted to $1-\delta$ for any desired $\delta > 0$ by performing a small number of trials and taking the median of the results; the number of trials required is $O(\ln \delta^{-1})$ [38].

$2\Delta < q$, and suppose that the sampler has time complexity $T(n,\delta)$, where n is the number of vertices in G, and δ the allowed deviation from uniformity in the sampling distribution. Then we may construct a randomised approximation scheme for the number of q-colourings of a graph, which works for graphs G with maximum degree bounded by Δ, and which has time complexity

$$O\left(\frac{m^2}{\varepsilon^2}\,T\left(n,\frac{\varepsilon}{6m}\right)\right),$$

where m is the number of edges in G, and ε the specified error bound.

At this point we merely indicate the key algorithmic technique underlying Proposition 3.1. A full proof, including a detailed statistical analysis, can be found in the last section.

Denote by $\Omega(G)$ the set of all q-colourings of G. Let $G = G_m > G_{m-1} > \cdots > G_1 > G_0 = (V,\emptyset)$ be any sequence of graphs in which each graph G_{i-1} is obtained from the previous graph G_i by removing a single edge e_i. We may express the quantity we wish to estimate as a product of ratios:

$$|\Omega(G)| = \frac{|\Omega(G_m)|}{|\Omega(G_{m-1})|}\times\frac{|\Omega(G_{m-1})|}{|\Omega(G_{m-2})|}\times\cdots\times\frac{|\Omega(G_1)|}{|\Omega(G_0)|}\times|\Omega(G_0)|, \qquad (3.1)$$

where, it will be observed, $|\Omega(G_0)| = q^n$. Our strategy is to estimate the ratio

$$\varrho_i = \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|}$$

for each i in the range $1 \le i \le m$, and by substituting these quantities into identity (3.1), obtain an estimate for the number of q-colourings of G:

$$|\Omega(G)| = q^n \varrho_1 \varrho_2 \cdots \varrho_m .$$

To estimate the ratio ϱ_i we use the almost uniform sampler to obtain a sufficiently large sample of q-colourings from $\Omega(G_{i-1})$ and compute the proportion of samples that lie in $\Omega(G_i)$ (i.e., for which the end points of e_i have different colours). The analysis presented in the last section places a bound on the sample size required.

For background material on approximate counting, refer to Welsh's survey article [59].

4. Sampling by Markov Chain Simulation

Let G be an undirected graph on vertex set $V = [n] = \{0,\ldots,n-1\}$ whose maximum degree is bounded by $\Delta = \Delta(G)$, and let $Q = [q]$ be a set of q colours. Let $X_0 : V \to Q$ be a proper colouring of the vertices of G, i.e., one in which every edge has endpoints of different colours. Such a colouring always exists if $q \ge \Delta + 1$, as can be appreciated by considering a simple sequential colouring algorithm. Indeed Brooks' theorem asserts that a colouring exists when $q \ge \Delta$, provided $\Delta \ge 3$ and G does not contain $K_{\Delta+1}$ as a connected component [7, 9].

For a discussion of strengthenings of Brooks' theorem via the probabilistic method, see Chapter 1 of this book, in particular Section 1.6.

Consider the Markov chain (X_t) whose state space $\Omega = \Omega(G,q)$ is the set of all q-colourings of G, and whose transition probabilities from state (colouring) X_t are given by the following procedure:

(1) Select a vertex $v \in V$ uniformly at random (u.a.r.), and then a colour $c \in Q$ u.a.r. from the set of legal colours for v. (A colour c is legal if it is different from the colour of any neighbour of v.)

(2) Recolour vertex v with colour c, and let the resulting colouring be X_{t+1}.

This procedure describes what would be termed, by the statistical physics community, the "heat-bath" dynamics of an antiferromagnetic q-state Potts model at zero temperature. Readers unfamiliar with this terminology need not worry; we do not use it in the sequel.

For $t \in \mathbb{N}$, let $P^t : \Omega^2 \to [0,1]$ denote the t-step transition probabilities[1] arising from this procedure, so that $P^t(x,y) = \Pr(X_t = y \mid X_0 = x)$ for all $x,y \in \Omega$.

Assume now that $q \ge \Delta + 2$. As we now verify, the Markov chain (X_t)—which we refer to in the sequel as $\mathfrak{M}_{\mathrm{LG}}(G,q)$ or simply $\mathfrak{M}_{\mathrm{LG}}$—is (a) irreducible, i.e., for all $x,y \in \Omega$ there is a t such that $P^t(x,y) > 0$, and (b) aperiodic, i.e., $\gcd\{t : P^t(x,x) > 0\} = 1$ for all $x \in \Omega$. Irreducibility of $\mathfrak{M}_{\mathrm{LG}}$ follows from the observation that any colouring x may be transformed to any other colouring y by sequentially assigning new colours to the vertices V in ascending sequence; before assigning a new colour c to vertex v, it is necessary to recolour all neighbouring vertices $u > v$ that have colour c, but there is always at least one "free" colour to allow this to be done, provided $q \ge \Delta + 2$. Aperiodicity follows from the fact that the loop probabilities $P(x,x)$ are non-zero for all $x \in \Omega$, thus if $P^t(x,y) > 0$ so is $P^{t+1}(x,y)$.

A finite Markov chain that is irreducible and aperiodic is ergodic, i.e., there is a unique stationary distribution $\pi : \Omega \to [0,1]$ such that for all

[1] We drop the superscript t in the case $t = 1$.

$x, y \in \Omega$, $\lim_{t\to\infty} P^t(x,y) = \pi(y)$. The use of the word "stationary" is justified by the fact that $\sum_{x\in\Omega}\pi(x)P(x,y) = \pi(y)$, for all $y \in \Omega$; loosely speaking, a Markov chain that is started in the stationary distribution remains in the stationary distribution for all time. In the case of $\mathcal{M}_{\mathrm{COL}}$, the stationary distribution is actually the uniform distribution on Ω, which can be deduced from the fact that $P(x,y) = P(y,x)$ for all x, y using the following simple but useful fact.

Lemma 3.1 *Let $\mathcal{M}$ be an ergodic Markov chain with finite state space Ω and transition probabilities $P(\cdot,\cdot)$. If $\pi' : \Omega \to [0,1]$ is any function satisfying "detailed balance"*

$$\pi'(x)P(x,y) = \pi'(y)P(y,x), \quad \text{for all } x, y \in \Omega, \tag{3.1}$$

and the normalisation condition $\sum_{x\in\Omega}\pi'(x) = 1$, then π' is indeed the stationary distribution of $\mathcal{M}$.

Proof. For all $y \in \Omega$,

$$\sum_{x\in\Omega}\pi'(x)P(x,y) = \sum_{x\in\Omega}\pi'(y)P(y,x) = \pi'(y);$$

i.e., π' is a stationary distribution of $\mathcal{M}$. But $\mathcal{M}$ is ergodic, so π' is the unique stationary distribution of $\mathcal{M}$. □

A Markov chain whose stationary distribution satisfies the detailed balance condition is said to be *time-reversible*.

In Section 5.3 we demonstrate that $\mathcal{M}_{\mathrm{COL}}$ is "rapidly mixing," i.e., the t-step distribution closely approaches the stationary distribution in time polynomial in n, provided $q \ge 2\Delta + 1$. To make this statement precise we need to explain what is meant by "closely" here.

To do so, we must generalise our definition of total variation distance. To wit, for any probability distributions π and π' on a countable set Ω we define the total variation distance between π and π' to be

$$D_{\mathrm{tv}}(\pi,\pi') = \max_{A\subseteq\Omega}|\pi(A) - \pi'(A)| = \frac{1}{2}\sum_{x\in\Omega}|\pi(x) - \pi'(x)|.$$

(This definition extends to uncountable probability spaces with the maximum replaced by a supremum over measurable sets A, and the sum by an integral.)

It seems natural to measure convergence to stationarity in terms of the variation distance. For $t \in \mathbb{N}$ define

$$\Delta_x(t) = D_{\mathrm{tv}}(P^t(x,\cdot),\pi) = \max_{A\subseteq\Omega}|P^t(x,A) - \pi(A)|,$$

where x is the initial state and $P^t(x,A) = \sum_{y\in A}P^t(x,y)$. The rate of convergence to stationarity from initial state x may be measured by the mixing time, i.e., the function

$$\tau_x(\varepsilon) = \min\{t : \Delta_x(t') \le \varepsilon \text{ for all } t' \ge t\}.$$

When making statements about rates of convergence that are independent of the initial state, the appropriate version of mixing time is $\tau(\varepsilon) = \max_x \tau_x(\varepsilon)$, where the maximum is over all $x \in \Omega$. By rapid mixing, we mean that $\tau(\varepsilon) \le \mathrm{poly}(n, \log \varepsilon^{-1})$.

The rapid mixing result of Section 5.3 provides us with a simple almost uniform sampler for q-colourings of G: simulate the Markov chain $\mathcal{M}_{\mathrm{COL}}$, starting at an arbitrary state, for a sufficiently large (but polynomial in n) number of steps, and return the current state as result. As a corollary we obtain, via Proposition 2.1, an FPRAS for the number of q-colourings of a graph in the case $q \ge 2\Delta + 1$.

As a warm-up we consider first the rather trivial case of an empty graph (i.e., $\Delta = 0$).

4 A Toy Example: Colourings of the Empty Graph

In this section we survey those techniques for proving rapid mixing that have shown themselves to have some degree of general applicability. The three techniques described here — which might be titled "canonical paths", "geometric" and "coupling" — cover the majority of applications. Nevertheless, some ingenious special techniques have been introduced to handle specific problems, most notably Feder and Mihail's inductive argument to demonstrate rapid mixing of the basis-exchange random walk on a "balanced" matroid [21].

The three techniques will be illustrated by applying each in turn to the graph-colourings Markov chain $\mathcal{M}_{\mathrm{COL}}(G,q)$ of Section 3, specialised to the empty graph $G_0 = (V,\emptyset)$, where, as usual, $|V| = n$. Since the state space in this case is simply $\Omega = Q^V$, it would be a trivial matter to sample from Ω directly. On the other hand, the very triviality of the situation will allow us to concentrate on the methods without getting bogged down in calculation or technical detail. Section 5 will consider some more realistic applications.

Sections 4.1–4.3 are largely independent of one another, as are Sections 5.1–5.3. Readers whose main goal is to follow the newer developments in coupling need only read Sections 4.1 and 5.1 before proceeding to Sections 6 and 7. In particular, an understanding of the geometric notions introduced in Section 4.2 is not required in the later sections. However, geometric arguments are of wider importance, particularly in the all-important application of the MCMC to volume estimation (see the discussion at the end of Section 3.2).

4.1 Canonical Paths

Let $\mathfrak{M}$ be an ergodic Markov chain with finite state space Ω, transition probabilities $P(\cdot,\cdot)$, and stationary distribution π. Any description of the canonical path argument is considerably simplified if we assume $\mathfrak{M}$ to be time-reversible. In the light of the detailed balance condition (3.1), we may view $\mathfrak{M}$ as an undirected graph (Ω, E) with vertex set Ω and edge set

$$E = \{\{x,y\} \in \Omega^{(2)} : \hat{P}(x,y) > 0\}, \qquad (4.1)$$

where

$$\hat{P}(x,y) := \pi(x)P(x,y) = \pi(y)P(y,x). \qquad (4.2)$$

For each (ordered) pair $(x,y) \in \Omega^2$, we specify a canonical path γ_{xy} from x to y in the graph (Ω, E); the canonical path γ_{xy} corresponds to a sequence of legal transitions in $\mathfrak{M}$ that leads from initial state x to final state y. Denote by $\Gamma = \{\gamma_{xy} : x,y \in \Omega\}$ the set of all canonical paths. For the method to yield good bounds, it is important to choose a set of paths Γ that avoids the creation of "hot spots:" edges of the graph that carry a particularly heavy burden of canonical paths. The degree to which an even loading has been achieved is measured by the quantity

$$\rho = \rho(\Gamma) := \max_t \frac{1}{\hat{P}(t)} \sum_{\gamma_{xy} \ni t} \pi(x)\pi(y)|\gamma_{xy}|,$$

where the maximum is over oriented edges (transitions) t of (Ω, E), and $|\gamma_{xy}|$ denotes the length of the path γ_{xy}.

If a Markov chain is to be rapidly mixing then clearly there is no small subset S of the state space such that the probability that we leave S after a transition, given we begin a randomly chosen element of S, is very small. In order to prove that a reversible ergodic chain is rapidly mixing we essentially have to prove that no such obstruction exists (a precise statement of this result is given in the next section). In this section we discuss doing so using canonical paths. Intuitively if a Markov chain has an obstruction S then the canonical paths between S and $\Omega \setminus S$ will overload the edges of E leaving S. Thus we expect a Markov chain to be rapidly mixing if it contains no "bottlenecks," i.e., if it admits a choice of paths Γ for which $\rho(\Gamma)$ is not too large.

This intuition is formalised in the following result, derived from Sinclair [56], which is a development of a theorem of Diaconis and Stroock [19].

Theorem 4.1. *Let $\mathfrak{M}$ be a finite, time-reversible, ergodic Markov chain with loop probabilities $P(x,x) \geq \frac{1}{2}$ for all states x. Let Γ be a set of canonical paths with maximum edge loading $\rho = \rho(\Gamma)$. Then the mixing time of $\mathfrak{M}$ satisfies $\tau_x(\varepsilon) \leq \rho(\ln \pi(x)^{-1} + \ln \varepsilon^{-1})$, where x is the initial state.*[2]

Proof. Combine [56, Prop. 1] and [56, Thm. 5]. □

We demonstrate the canonical path method by applying it to the toy example. For convenience, we shall work with a slightly modified version of the Markov chain $\mathfrak{M}_{\mathrm{rc}}$ of Section 3. The transitions will be defined as before, except for the addition of a preliminary step:

(0) with probability $\frac{1}{2}$ let X_{t+1} equal X_t and halt this transition, otherwise, progress to step (1).

This modification has the effect of adding an additional loop probability $\frac{1}{2}$ to every state (and reducing all other transition probabilities by a similar factor). Let us refer to the modified Markov chain with increased loop probabilities as $\mathfrak{M}'_{\mathrm{rc}}$. Note that $\mathfrak{M}'_{\mathrm{rc}}(Q_n, q)$ satisfies the conditions of Theorem 4.1.

Let $x = (x_0, \ldots, x_{n-1})$ and $y = (y_0, y_1, \ldots, y_{n-1})$ be arbitrary colourings in $[1..q]^n$. To obtain the canonical path γ_{xy} from x to y, first consider the path obtained by composing the n edges (transitions) t_i for $0 \leq i \leq n-1$, where

$$t_i = \big((y_0, \ldots, y_{i-1}, x_i, x_{i+1}, \ldots, x_{n-1}), (y_0, \ldots, y_{i-1}, y_i, x_{i+1}, \ldots, x_{n-1})\big),$$

i.e., t_i is the transition that changes the ith colour from x_i to y_i. Now remove any loops. To compute ρ, fix attention on a particular (oriented) edge

$$t = (w, w') = \big((w_0, \ldots, w_{i-1}, w_i, \ldots, w_{n-1}), (w_0, \ldots, w'_i, \ldots, w_{n-1})\big)$$

and consider the number of canonical paths γ_{xy} that include t. The number of possible choices for x is q^i, as the final $n-i$ positions are determined by $x_j = w_j$ for $j \geq i$, and by a similar argument the number of possible choices for y is q^{n-i-1}. Thus the total number of canonical paths using a particular edge t is q^{n-1}. Furthermore, $\hat{P}(t) = \pi(w)P(w,w') \geq q^{-n}(2qn)^{-1}$, and the length of every canonical path is at most n. Plugging all these bounds into the definition of ρ yields $\rho \leq 2n^2$. Thus, by Theorem 4.1, the mixing time

[2] This Theorem also has a slightly weaker converse; see [56, Thm 8].

of $\mathcal{MC}_{\mathrm{cube}}(Q_n)$ is $\tau(\varepsilon) \le 2n^2(n \ln 2 + \ln \varepsilon^{-1})$. Note that the mixing time of $\mathcal{MC}_{\mathrm{cube}}(Q_n)$ grows only polynomially with the input size n, even though the size of the state space is exponential in n: i.e., $\mathcal{MC}_{\mathrm{cube}}(Q_n)$ is "rapidly mixing" in the sense of Section 3. The bound on mixing time we have derived is some way off the exact answer [21], which is $\tau(\varepsilon) = O(n(\log n + \log \varepsilon^{-1}))$, and the slackness we see here is typical of the method.

On reviewing the canonical path argument, we perceive what appears to be a major weakness. In order to compute the key quantity ρ, we needed to compute quantities such as $\tilde{P}(t)$ that depend crucially on the size of the state space Ω. In the current example this does not present a problem, but in more interesting examples we do not know the size of the state space: indeed, our ultimate goal will often be to estimate this very quantity. Fortunately, it is possible to finesse this obstacle by implicit counting using a carefully constructed injective map. The idea will be illustrated by application to the Markov chain $\mathcal{MC}_{\mathrm{cube}}(Q_n)$.

Let edge $t = (u, u')$ be as before, and denote by $\mathrm{cp}(t) = \{(x,y) : \gamma_{xy} \ni t\}$ the set of all (endpoints of) canonical paths that use edge t. Define the map $\eta_t : \mathrm{cp}(t) \to \Omega$ as follows: if $(x,y) = ((x_0,\ldots,x_{n-1}),(y_0,\ldots,y_{n-1})) \in \mathrm{cp}(t)$ then

$$\eta_t(x,y) = (w_0,\ldots,w_{n-1}) = (x_0,\ldots,x_{i-1},y_i,\ldots,y_{n-1}).$$

The crucial feature of the map η_t is that it is injective. To see this, observe that x and y may be unambiguously recovered from $(w_0,\ldots,w_{n-1}) = \eta_t(x,y)$ through the explicit expressions

$$x = (w_0,\ldots,w_{i-1},u_i,u_{i+1},\ldots,u_{n-1})$$

and

$$y = (u_0,\ldots,u_{i-1},w_i,w_{i+1},\ldots,w_{n-1}).$$

Using the injective map η_t it is possible to evaluate ρ without recourse to explicit counting. Noting[4] that $\pi(x)\pi(y) = \pi(u)\pi(\eta_t(x,y))$, we have

$$\begin{aligned}
\frac{1}{\tilde{P}(t)} \sum_{\gamma_{xy} \ni t} \pi(x)\pi(y)|\gamma_{xy}| &= \frac{1}{\pi(u)P(u,u')} \sum_{\gamma_{xy} \ni t} \pi(u)\pi(\eta_t(x,y))|\gamma_{xy}| \\
&= \frac{n}{P(u,u')} \sum_{\gamma_{xy} \ni t} \pi(\eta_t(x,y)) \\
&\le \frac{n}{P(u,u')} \le 2n^2,
\end{aligned}$$

[4] This is a trivial observation when the stationary distribution is uniform, as it is here, but it is sometimes possible, by judicious choice of η_t, to contrive such an equality even when the stationary distribution is non-uniform. See Section 5.1 for an example.

where the penultimate inequality follows from the facts that η_t is injective and that π is a probability distribution. Since the above argument is valid uniformly over the choice of t, we deduce $\rho \le 2n^2$. The factor of 2 as compared with the direct argument was lost to redundancy in the encoding: the map η_t was not a bijection.

4.2 Geometry

As before, suppose $\mathcal{M}$ is a finite, time-reversible ergodic Markov chain with stationary distribution π, and recall definitions (4.2) and (4.1) of $\tilde{P}$ and T from the previous section. The conductance [86] of $\mathcal{M}$ is defined by

$$\Phi = \Phi(\mathcal{M}) = \min_{\substack{S \subset \Omega \\ 0 < \pi(S) \le 1/2}} \frac{\tilde{P}(S, \bar{S})}{\pi(S)}, \tag{4.5}$$

where $\tilde{P}(S,\bar{S})$ denotes the sum of $\tilde{P}(x,y)$ over edges $\{x,y\} \in T$ with $x \in S$ and $y \in \bar{S} = \Omega \setminus S$. The conductance may be viewed as a weighted version of edge expansion of the graph (Ω, T) associated with $\mathcal{M}$. Alternatively, the quotient appearing in (4.5) can be interpreted as the conditional probability that the chain in equilibrium escapes from the subset S of the state space in one step, given that it is initially in S; thus Φ measures the readiness of $\mathcal{M}$ to escape from any small enough region of the state space, and hence to make rapid progress towards equilibrium. This intuitive connection can be given a precise quantitative form as follows. (Related results may be found in the work of Aldous [2] and Alon [3].)

Theorem 4.2. *Sinclair. Let $\mathcal{M}$ be a finite, reversible, ergodic Markov chain with loop probabilities $P(x,x) \ge \frac{1}{2}$ for all states x. Let Φ be the conductance of $\mathcal{M}$ as defined in (4.5). Then the mixing time of $\mathcal{M}$ satisfies $\tau_x(\varepsilon) \le 2\Phi^{-2} \times (\ln \pi(x)^{-1} + \ln \varepsilon^{-1})$, where x is the initial state.*

Proof. Combine [85, Prop. 1] and [86, Thm 2]. ∎

One approach in this section to bounding the conductance of a Markov chain $\mathcal{M}$ is to give $\mathcal{M}$ a geometric interpretation, in which states of $\mathcal{M}$ are identified with certain polytopes, and transitions with their common faces. A lower bound on conductance then follows from an isoperimetric inequality.[5] This was the approach pioneered by Dyer, Frieze and Kannan in the analysis of a random walk in a convex body [32], and Karzanov and Khachiyan in the context of a Markov chain on linear extensions of a partial order [60] (see also Section 5.1). The following isoperimetric inequality of Dyer and Frieze, is particularly well suited to this purpose. To state the inequality we need the concept of the dual of a norm. If $\|\cdot\|$ is a norm, then the norm $\|\cdot\|^*$ dual to $\|\cdot\|$ is defined by

$$\|x\|^* = \sup\{x \cdot y : \|y\| = 1\}.$$

The symbol ∂ denotes "boundary of."

Theorem 4.9. [Dyer and Frieze] *Suppose $K \subseteq \mathbb{R}^m$ is a convex body and f a log-concave function on int K. For a set $S \subset K$ such that $\sigma = \partial S \setminus \partial K$ is a piecewise smooth surface, define $\mu(S) = \int_S f(x)\,dx$ and $\nu(\sigma) = \int_\sigma f(x)\|u(x)\|^*\,dx$, where $u(x)$ is the Euclidean unit normal to σ at $x \in \sigma$. If $\mu(S) \le \frac{1}{2}\mu(K)$, then $\nu(\sigma)/\mu(S) \ge 2/\operatorname{diam} K$, where the diameter $\operatorname{diam} K$ is measured with respect to the (primal) norm $\|\cdot\|$.*

Proof. See [27], Thm. 3 and preliminary remarks. □

We illustrate the utility of Theorem 4.9 by applying it to the toy example. We again work with the modified Markov chain $\mathcal{MC}'_{\mathrm{col}}(O_n, q)$, with inflated loop probabilities, applied to the empty graph O_n. We view colourings of Ω as functions $V \to Q$, where $V = [n]$ and $Q = [q]$. For each colouring $c \in \Omega$ define a corresponding polytope (a closed, bounded region formed by the intersection of half-spaces) in $\mathbb{R}^{nq}$ by

$$R(c) = \{x = (x_{ij}) \in \mathbb{R}^{n \times q} : 0 \le x_{ij} \le 1 \text{ and } x_{i,c(i)} \ge x_{ij} \text{ for all } i, j\}.$$

For any $S \subseteq \Omega$, let $R(S) = \bigcup_{c \in S} R(c)$, and observe that $K := R(\Omega) = \frac{1}{2}B_\infty$, where $\frac{1}{2}B_\infty$ denotes the ℓ_∞-ball of radius $\frac{1}{2}$, or unit cube. Clearly, $\operatorname{diam} K = 1$, where diameter is measured with respect to ℓ_∞-norm. Note that, by symmetry, $\operatorname{vol}_{nq} R(c) = |\Omega|^{-1}$ for any $c \in \Omega$, and hence

$$\operatorname{vol}_{nq} R(S) = \frac{|S|}{|\Omega|}. \tag{4.4}$$

Recall the definitions of $\overline{P}$ (3.2) and of conductance (4.3). A transition is available between colourings c and c' (we say the colourings are adjacent) iff they differ at exactly one vertex; equivalently, iff $R(c)$ and $R(c')$ share a common facet (i.e., $(nq-1)$-dimensional face). By calculus, the $(nq-1)$-dimensional volume of such a facet is

$$\operatorname{vol}_{nq-1}(R(c) \cap R(c')) = \frac{\sqrt{2}}{q^{n-1}(q-1)}. \tag{4.5}$$

(See the last section for a proof of this claim.) Thus the number of transitions $(c, c') \in (S, \overline{S})$ from a state in S to one in $\overline{S}$ is

$$\operatorname{vol}_{nq-1}(\partial R(S) \setminus \partial K) \times \frac{q^{n-1}(q-1)}{\sqrt{2}},$$

and, since the $\overline{P}(c, c') = (2nq|\Omega|)^{-1}$ for any pair of adjacent colourings c, c'

$$\overline{P}(S, \overline{S}) = \frac{q^{n-1}(q-1)\operatorname{vol}_{nq-1}(\partial R(S) \setminus \partial K)}{2\sqrt{2}nq|\Omega|}. \tag{4.6}$$

Furthermore the unit vector u normal to any facet has ℓ_1-norm $\|u\|_1 = \sqrt{2}$. Taking f identically 1 in Theorem 4.9, we have, for $|S| \le \frac{1}{2}|\Omega|$,

$$\frac{\operatorname{vol}_{nq} R(S)}{\sqrt{2}\operatorname{vol}_{nq-1}(\partial R(S) \setminus \partial K)} \le \frac{\operatorname{diam} K}{2},$$

which, in the light of (4.4), is equivalent to

$$\operatorname{vol}_{nq-1}(\partial R(S) \setminus \partial K) \ge \frac{\sqrt{2}|S|}{|\Omega|}.$$

Combining this inequality with (4.6) yields

$$\overline{P}(S, \overline{S}) \ge \frac{(q-1)|S|}{2nq^2|\Omega|},$$

whence, by definition of conductance (4.3),

$$\Phi \ge \frac{q-1}{2nq^2}.$$

Thus, by Theorem 4.3, the mixing time of $\mathcal{MC}'_{\mathrm{col}}(O_n, q)$ is

$$\tau(\varepsilon) \le 8n^2q^4(q-1)^{-2}(n \ln q + \ln \varepsilon^{-1}).$$

Again, we have demonstrated that $\mathcal{MC}'_{\mathrm{col}}(O_n, q)$ is rapidly mixing, though the bound is worse by a factor of order q^2 than the one we had already obtained using the canonical paths argument.

4.3 Coupling

Suppose $\mathcal{M}$ is a countable, ergodic (though not necessarily time-reversible) Markov chain with transition probabilities $P(\cdot, \cdot)$ and stationary distribution π. As usual, the assumption of countability is for expository convenience only, and the ideas easily extend to uncountably infinite state spaces. In its basic form, the coupling technique was introduced by Doeblin in the 1930s. The word "coupling" in probability theory is applied to a variety of related notions, and it would be difficult to provide a general definition. In the current context, we mean by coupling a Markov process (X_t, Y_t) on $\Omega \times \Omega$

such that each of the processes (X_t) and (Y_t), considered in isolation, is a faithful copy of $\mathfrak{M}$. More precisely, we require that

$$\Pr(X_{t+1} = x' \mid X_t = x \wedge Y_t = y) = P(x, x') \tag{4.7}$$

and

$$\Pr(Y_{t+1} = y' \mid X_t = x \wedge Y_t = y) = P(y, y'), \tag{4.8}$$

for all $x, x', y, y' \in \Omega$. This condition is consistent with (X_t) and (Y_t) being independent evolutions of $\mathfrak{M}$, but does not imply it. In fact, we shall use the possibility that

$$\Pr(X_{t+1} = x' \wedge Y_{t+1} = y' \mid X_t = x \wedge Y_t = y) \neq P(x,x')P(y,y')$$

to encourage (X_t) and (Y_t) to coalesce rapidly, so that $X_t = Y_t$ for all sufficiently large t. (Note that it is easy to design the coupling so that, if t is the first time step such that $X_t = Y_t$, then $X_{t'} = Y_{t'}$ for all $t' > t$.)

If it can be arranged that coalescence occurs rapidly, independently of the initial states X_0 and Y_0, we may deduce that $\mathfrak{M}$ is rapidly mixing. The key result we use here is the "Coupling Lemma", which apparently makes its first explicit appearance in the work of Aldous [1, Lemma 3.6] (see also Diaconis [17, Chap. 4, Lemma 5]).

Lemma 4.4. *Suppose that $\mathfrak{M}$ is a countable, ergodic Markov chain with transition probabilities $P(\cdot,\cdot)$, and let $((X_t, Y_t) : t \in \mathbb{N})$ be a coupling, i.e., a Markov process satisfying (4.7) and (4.8). Suppose further that $t : (0,1] \to \mathbb{N}$ is a function such that $\Pr(X_{t(\varepsilon)} \neq Y_{t(\varepsilon)}) \le \varepsilon$ for all $\varepsilon \in (0,1]$, uniformly over the choice of initial state (X_0, Y_0). Then the mixing time $\tau(\varepsilon)$ of $\mathfrak{M}$ is bounded above by $t(\varepsilon)$.*

Proof. Let $X_0 = x \in \Omega$ be arbitrary, and choose Y_0 according to the stationary distribution π. Fix $\varepsilon \in (0,1]$ and for convenience abbreviate $t(\varepsilon)$ to t. Let $A \subseteq \Omega$ be an arbitrary event. Then

$$\begin{aligned}\Pr(X_t \in A) &\ge \Pr(Y_t \in A \wedge X_t = Y_t)\\ &\ge 1 - \Pr(Y_t \notin A) - \Pr(X_t \neq Y_t)\\ &\ge \Pr(Y_t \in A) - \varepsilon\\ &= \pi(A) - \varepsilon,\end{aligned}$$

with a similar inequality holding for the complementary event $\Omega \setminus A$. Since A was chosen arbitrarily, $d_{TV}(P^t(x, \cdot), \pi) \le \varepsilon$, i.e., the total variation distance between the t-step distribution and the stationary distribution is bounded by ε. □

For the toy example, the coupling may be very simple indeed. The transition $(X_t, Y_t) \mapsto (X_{t+1}, Y_{t+1})$ in the coupling is defined by the following experiment:

(1) Select a vertex $v \in V$ u.a.r.

(2) Select a colour $c \in Q$ u.a.r., and recolour vertex v in X_t (respectively Y_t) with colour c and let the resulting colouring be X_{t+1} (respectively Y_{t+1}).

Note that (X_t) and (Y_t) are both faithful copies of $\mathfrak{M}$; specifically, (4.7) and (4.8) are satisfied. Nevertheless it is also clear that (X_t) and (Y_t) are "highly coupled" and we can expect rapid coalescence.

As before, regard states (colourings) as functions $V \to Q$. Denote by D_t the random variable

$$D_t = \{v \in V : X_t(v) \neq Y_t(v)\},$$

i.e., the set of vertices on which the two colourings X_t and Y_t disagree. If step (1) of the coupling selects a vertex v in D_t then $D_{t+1} = D_t \setminus \{v\}$; otherwise $D_{t+1} = D_t$. Since v is selected u.a.r.,

$$\mathbb{E}(|D_{t+1}| \mid D_t) = \left(1 - \frac{1}{n}\right)|D_t|$$

and hence

$$\mathbb{E}(|D_t| \mid D_0) = \left(1 - \frac{1}{n}\right)^t |D_0|.$$

Since $|D_t|$ is a non-negative integer r.v., we obtain

$$\begin{aligned}\Pr(|D_t| > 0 \mid D_0) &\le \mathbb{E}(|D_t| \mid D_0)\\ &\le n\left(1 - \frac{1}{n}\right)^t\\ &\le n e^{-t/n},\end{aligned}$$

which is bounded by ε, provided $t \ge n \ln n\varepsilon^{-1}$. Invoking the Coupling Lemma, we obtain $\tau(\varepsilon) \le n(\ln n + \ln \varepsilon^{-1})$, independent of the starting state x; the correct asymptotic result.

5. Some More Challenging Applications

We now explore the three techniques for proving rapid mixing in the context of three more realistic problems. In each case, the chosen solution technique will be "natural" for the application. Indeed, for our first example, we are forced to use the canonical paths method, as it provides the only known solution technique.

5.1 Monomer-Dimer Coverings Via Canonical Paths

The presentation of this topic is condensed from Jerrum and Sinclair [31], which in turn is an improved version of the original source [34]. See also Sinclair [57].

We shall be concerned with the classical monomer-dimer model from statistical physics. A monomer-dimer system is defined by a graph $G = (V, E)$ and a positive real parameter λ. A configuration of the system is just a *matching* in G, that is to say, a subset $M \subseteq E$ such that no two edges in M share an endpoint. In physical terms, the pairs of matched vertices are dimers and the uncovered vertices monomers. Thus a matching of cardinality k, or *k-matching*, corresponds precisely to a monomer-dimer configuration with k dimers and $2(n-k)$ monomers, where $2n = |V|$ is the number of vertices in G. (The assumption that the number of vertices in G is even is inessential and is made for notational convenience.) Typically, G is a regular lattice in some fixed number of dimensions, but we shall make no such assumption in what follows. For a detailed account of the history and significance of monomer-dimer systems, the reader is referred to the seminal paper of Heilmann and Lieb [28] and the references given there.

To each matching M, a weight $w(M) = \lambda^{|M|}$ is assigned; thus the parameter λ reflects the contribution of a dimer to the energy of the system. The *partition function* of the system is defined as

$$Z = Z(G, \lambda) = \sum_{M} w(M) = \sum_{k=0}^{n} m_k \lambda^k, \tag{5.1}$$

where $m_k = m_k(G)$ is the number of k-matchings in G. For a physical interpretation of (5.1), see [28]. The partition function may be efficiently approximated (in the FPRAS sense) using the method of Section 2, provided we can efficiently sample matchings from the distribution that assigns probability

$$\pi(M) = \frac{w(M)}{Z} \tag{5.2}$$

to matching M (see [31] for details). We therefore concentrate on the sampling problem.

Following an idea of Broder [8], we construct a Markov chain $\mathfrak{M}_{\text{match}} = \mathfrak{M}_{\text{match}}(G, \lambda)$, parameterised by the underlying graph G and the edge weight λ. The state space, Ω, is the set of all matchings in G, and the transitions are constructed so that the chain is ergodic with stationary distribution π given by (5.2). In other words, the stationary probability of each matching (monomer-dimer configuration) is proportional to its weight in the partition function (5.1). The Markov chain $\mathfrak{M}_{\text{match}}$, if simulated for sufficiently many steps, provides a method of sampling matchings from the distribution π.

It is not hard to construct a Markov chain $\mathfrak{M}_{\text{match}}$ with the right asymptotic properties. Let the state of $\mathfrak{M}_{\text{match}}$ at time t be X_t. The probability distribution of the next state X_{t+1} is defined by the following experiment:

(1) With probability $\frac{1}{2}$ let $X_{t+1} := X_t$ and halt.

(2) Otherwise (with the remaining probability $\frac{1}{2}$), select an edge $e = \{u, v\} \in E$, u.a.r., and set

$$M' := \begin{cases} M - e & \text{if } e \in M; \\ M + e & \text{if both } u \text{ and } v \text{ are unmatched in } M; \\ M + e - e' & \text{if exactly one of } u \text{ and } v \text{ is matched in } M \\ & \quad \text{and } e' \text{ is the matching edge;} \\ M & \text{otherwise.} \end{cases}$$

(3) With probability $\min\{1, w(M')/w(M)\}$ let $X_{t+1} := M'$; otherwise (with the complementary probability) let $X_{t+1} := M$.

It is helpful to view this chain as follows. There is an underlying graph defined on the set of matchings Ω in which the neighbours of matching M are all matchings M' that differ from M via one of the following local perturbations: an edge is removed from M (a $\downarrow$-transition), an edge is added to M (a $\uparrow$-transition), or a new edge is exchanged with an edge in M (a $\leftrightarrow$-transition). Transitions from M are made by first selecting a neighbour M' u.a.r., and then actually making, or *accepting* the transition with probability $\min\{1, w(M')/w(M)\}$. Note that the ratio appearing in this expression is easy to compute: it is just λ^{-1}, λ or 1 respectively, according to the type of the transition.

As the reader may easily verify, this acceptance probability is constructed so that the transition probabilities $P(M, M')$ of $\mathfrak{M}_{\text{match}}$ satisfy the detailed balance condition (3.1) for the distribution π of (5.2). Furthermore, $\mathfrak{M}_{\text{match}}$ is irreducible (i.e., all states communicate via the empty matching) and aperiodic (by step 1, the self-loop probabilities $P(M, M)$ are all non-zero), and hence ergodic. Thus, by Lemma 3.1, the distribution π defined in (5.2) is indeed the stationary distribution of $\mathfrak{M}_{\text{match}}$.[5]

[5] The device of performing random walk on a connected graph with acceptance probabilities of this form is well known in computational physics under the name of the "Metropolis process" [58]. Clearly, it can be used to achieve any desired stationary distribution π for which the ratio $\pi(u)/\pi(v)$ for neighbours u, v can be computed easily.

Proposition 5.1. *The mixing time of the Markov chain $\mathcal{M}_{\mathrm{match}}$ satisfies*

$$\tau(\varepsilon) \le 4|E|n\bar{\lambda}\bigl(n(\ln n + \ln\bar{\lambda}) + \ln\varepsilon^{-1}\bigr),$$

where $\bar{\lambda} = \max\{1, \lambda\}$.

Proof Sketch. Our strategy will be to carefully choose a collection of canonical paths $\Gamma = \{\gamma_{XY} : X, Y \in \Omega\}$ in the Markov chain $\mathcal{M}_{\mathrm{match}}$ for which the "bottleneck" measure $\bar{\rho}(\Gamma)$ of Section 4.1 is small. We can then appeal to Theorem 4.1 to bound the mixing time. Specifically, we shall show that our paths satisfy

$$\bar{\rho}(\Gamma) \le 4|E|n\bar{\lambda}^2. \tag{5.3}$$

Since the number of matchings in G is certainly bounded above by $(2n)!$, the stationary probability $\pi(X)$ of any matching X is bounded below by $\pi(X) \ge 1/(2n)!\,\bar{\lambda}^n$. Using (5.3) and the fact that $\ln n! \le n \ln n$, the bound on the mixing time in Proposition 5.1 can now be read off Theorem 4.1.

It remains for us to find a set of canonical paths Γ satisfying (5.3). For every pair of matchings X, Y in Ω we construct a canonical path γ_{XY} from X to Y as indicated in Figure 5.1. (A rigorous description of the canonical paths together with all other details missing from this sketch proof may be found in [24].)

The interpretation of Figure 5.1 is as follows. Consider the symmetric difference $X \oplus Y$. A moment's reflection should convince the reader that this consists of a disjoint collection of paths in G (some of which may be closed cycles), each of which has edges that belong alternately to X and to Y. Now suppose that we have fixed some arbitrary ordering on the set of all simple paths in G, and designated in each of them a so-called "start vertex," which is arbitrary if the path is a closed cycle but must be an endpoint otherwise. This ordering induces a unique ordering $P_1, P_2, \ldots, P_m$ on the paths appearing in $X \oplus Y$. The canonical path from X to Y involves "unwinding" each of the P_i in turn. In Figure 5.1 the path P_i (which happens to be a cycle) is the one currently being unwound; the paths $P_1, \ldots, P_{i-1}$ to the left have already been processed, while the ones $P_{i+1}, \ldots, P_m$ are yet to be dealt with.

Unwinding a cycle is done by removing the edge adjacent to the start vertex using a ↓-transition; then moving round the cycle using ↔-transitions to swap Y-edges for X-edges; and finally completing the cycle with a single ↑-transition. A path is processed similarly, working from one end to the other using a sequence of ↔-transitions to swap Y-edges for X-edges, starting or finishing with the path with single ↑- or ↓-transitions as required.

We now proceed to bound the "bottleneck" measure $\bar{\rho}(\Gamma)$ for these paths using the injective mapping technology introduced in Section 4.1. Let t be an arbitrary edge in the Markov chain, i.e., a transition from M to $M' \ne M$, and let $\mathrm{cp}(t) = \{(X,Y) : \gamma_{XY} \ni t\}$ denote the set of all canonical paths that use t. Just as in Section 4.1, we shall obtain a bound on the total weight of all paths that pass through t by defining an injective mapping $\eta_t : \mathrm{cp}(t) \to \Omega$. By analogy with the toy example in Section 4.1, what we would like to do is to set $\eta_t(X,Y) = X \oplus Y \oplus (M \cup M')$; the intention is that $\eta_t(X,Y)$ should agree with X on paths that have already been unwound, and with Y on paths that have not yet been unwound (just as $\eta_t(x,y)$ agreed with x on positions $1, \ldots, i-1$ and with y on positions $i+1, \ldots, n-1$). This will not quite do, since the set of edges $\eta_t(X,Y)$ defined in this way may fail to be a matching; however, the problem is a small one and can be rectified by removing a single offending edge. Figure 5.2 illustrates the encoding $\eta_t(X,Y)$ that would result from the transition t on the canonical path sketched in Figure 5.1.

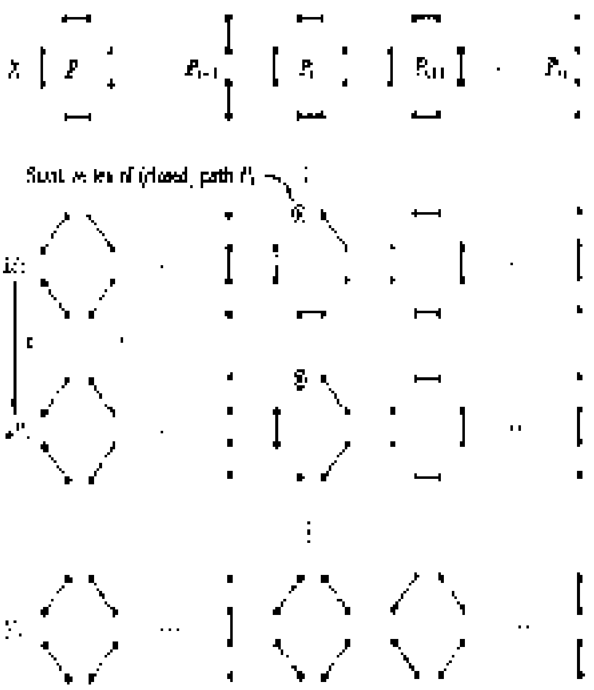


Fig. 5.1. A transition t in the canonical path from X to Y

We now have to check that η_t is injective, which amounts to demonstrating that X and Y can be unambiguously reconstructed from a knowledge of $t = (M, M')$ and $\eta_t(X,Y)$. Roughly, the way this is done is to note that, modulo the single offending edge,

$$X \oplus Y = \eta_t(X,Y) \oplus (M \cup M'),$$

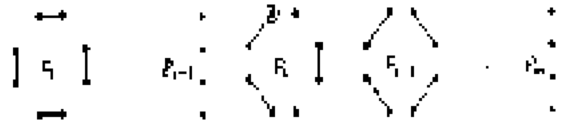


Fig. 5.2. The corresponding encoding $\eta_t(X,Y)$

so that, given $t = (M, M')$ and $\eta_t(X,Y)$, we may compute the path decomposition $P_1, \ldots, P_m$. The path P_i being processed during the transition t is immediately apparent from an examination of $M \oplus M'$. From these, it is a straightforward matter to apportion edges in $P_1 \cup \cdots \cup P_m$ to X or Y as appropriate. Finally, edges in $\eta_t(X,Y) \cap M$ are the ones which are common to X and Y.

We are almost done. However, the fact that η_t is injective is not sufficient in this case because, in contrast to the toy example, the stationary distribution π is highly non-uniform. What we require in addition is that η_t be "weight-preserving," in the sense that $P(t)\pi(\eta_t(X,Y))$ is reasonably close to $\pi(X)\pi(Y)$. Roughly speaking, this occurs because each edge $e \in E$ (with a couple of exceptions) contributes an equal factor (1, λ or λ^2) to the two terms $\pi(M)\pi(\eta_t(X,Y))$ and $\pi(X)\pi(Y)$. Specifically it can be shown that

$$\pi(X)\pi(Y) \leq 2|E|\bar\lambda^2 P(t)\pi(\eta_t(X,Y)). \tag{5.4}$$

It is not too difficult to achieve a looser version of (5.4) with $\bar\lambda^4$ replacing $\bar\lambda^2$ on the right hand side, but the inequality as given requires a little care. The full calculation can be found in [37].

A bound on $\bar\varrho$ follows easily from (5.4). We have

$$\begin{aligned}\bar\varrho(\Gamma) &:= \frac{1}{P(t)}\sum_{\gamma_{XY}\ni t}\pi(X)\pi(Y)|\gamma_{XY}| && (5.5)\\ &\leq 2|E|\bar\lambda^2 \sum_{\gamma_{XY}\ni t}\pi(\eta_t(X,Y))|\gamma_{XY}| \\ &\leq 4|E|n\bar\lambda^2 \sum_{\gamma_{XY}\ni t}\pi(\eta_t(X,Y)) \\ &\leq 4|E|n\bar\lambda^2, && (5.6)\end{aligned}$$

where the second inequality follows from the fact that the length of any canonical path is bounded by $2n$, and the last inequality from the facts that η_t is injective and π is a probability distribution. The claimed bound on mixing time follows quickly from (5.6) and Theorem 4.2, as described at the outset. □

Aside from the monomer-dimer example presented in this section, applications of the canonical path method include counting dimer coverings (perfect matchings) of lattice graphs (Kenyon, Randall and Sinclair [47]), evaluating the partition function of the ferromagnetic Ising model (Jerrum and Sinclair [38]) and counting configurations in the "six-point ice model" (Mihail and Winkler [55]). All these applications share similarities with the monomer-dimer one. The reader will learn more about Monte Carlo methods for computing partition functions for statistical physics models in the next chapter.

An application which is further removed from the monomer-dimer example is to the "basis-exchange" random walk for graphic matroids. The state space here is the set of spanning trees of a graph, and a transition from tree T to T' is possible if the symmetric difference of T and T' consists of just two edges. The canonical paths argument for spanning trees has not, as far as I am aware, appeared explicitly in the literature, but Cordovil and Moreira have presented a construction (see [16, Thm 1.6]) for paths between pairs of spanning trees that is ideally suited to this purpose. However, there are many other approaches to proving rapid mixing in this instance (see Aldous [3], Dyer and Frieze [21] and Feder and Mihail [27]). Refer to Section 6 for a related open problem.

5.2 Linear Extensions of a Partial Order Via Geometry

In this example we essentially follow Karzanov and Khachiyan [44], though we achieve a sharper bound by invoking an enhanced isoperimetric inequality due to Dyer and Frieze [21].

We are given a partially ordered set $(V, \prec)$, where $V = [n]$. Denote by $\mathrm{Sym}\,V$ the symmetric group on V. We are interested in sampling, u.a.r., a member of the set

$$\Omega = \{g \in \mathrm{Sym}\,V : g(i) \prec g(j) \Rightarrow i \leq j, \text{ for all } i, j \in V\}$$

of linear extensions of $\prec$. In forming a mental picture of the state space Ω, the following characterisation may be helpful: $g \in \Omega$ if the linear order

$$g(0) \sqsubset g(1) \sqsubset \cdots \sqsubset g(n-1) \tag{5.7}$$

extends, or is consistent with, the partial order $\prec$.

As usual, we propose to sample from Ω by constructing an ergodic Markov chain on state space Ω, whose stationary distribution is uniform. Transitions from a linear extension $g \in \Omega$ are generated by composing g with a random transposition $(p, p+1)$[6] equivalently, by swapping adjacent elements in the linear order (5.2). Formally, transition probabilities from state $X_t \in \Omega$ are defined by the following experiment:

(1) Select $p \in [n-1]$ and $r \in \{0,1\}$, u.a.r.

(2) If $r = 1$ and $X_t \circ (p, p+1) \in \Omega$ then $X_{t+1} := X_t \circ (p, p+1)$; otherwise $X_{t+1} := X_t$.

Here, the operator $\circ$ denotes function composition (read right to left). Let us refer to this Markov chain as $\mathcal{M}_{LE}$. As in Section 4.2, the loop probabilities are artificially raised to permit convenient application of Theorem 4.2.

Proposition 5.4. *The mixing time of the Markov chain $\mathcal{M}_{LE}$ satisfies*

$$\tau(\varepsilon) \le 8n^2(n-1)^2(\ln \varepsilon^{-1} + \ln n!) = O(n^4(n \log n + \log \varepsilon^{-1})).$$

We shall see in Section 6 that this bound can be tightened considerably.

Proof. We adopt the notation introduced in Section 4.2. To each permutation $g \in \operatorname{Sym} V$ associate the simplex

$$R(g) = \{z = (z_i) \in \mathbb{R}^n : 0 \le z_{g(1)} \le z_{g(2)} \le \cdots \le z_{g(n)} \le 1\}.$$

For any $S \subseteq \operatorname{Sym} V$, let $R(S) = \bigcup_{g \in S} R(g)$, and observe that $R(\operatorname{Sym} V) = \frac{1}{2}B_\infty$, where $\frac{1}{2}B_\infty$ denotes the ℓ_∞-ball of radius $\frac{1}{2}$, or unit cube. Define $K := R(\Omega)$, and observe that K is a convex set. (Take any two points in K and join them by a straight line segment. It is routine to check that every intermediate point is contained in a simplex $R(g)$, where g is a linear extension of $\prec$.) Clearly, $\operatorname{diam} K \le \operatorname{diam}(R(\operatorname{Sym} V)) = 1$, where diameter is measured with respect to ℓ_∞-norm. Note that, by symmetry, $\operatorname{vol}_n R(g) = |\operatorname{Sym} V|^{-1} = 1/n!$ for any $g \in \Omega$ and hence

$$\operatorname{vol}_n R(S) = \frac{|S|}{n!}. \qquad (5.4)$$

A transition is available between linear extensions g and g' (we say that g and g' are *adjacent*) if they differ in an adjacent transposition, equivalently, if $R(g)$ and $R(g')$ share a common $(n-1)$-dimensional face. By an argument very similar to that used in Section 4.2 (see also the next section), if g and g' are adjacent,

$$\operatorname{vol}_{n-1}(R(g) \cap R(g')) = \frac{\sqrt{2}}{(n-1)!},$$

[6] The transposition is to be performed first, followed by the permutation g.

so the number of transitions $(g, g') \in (S, \bar S)$ from a state in S to one in $\bar S$ is

$$\operatorname{vol}_{n-1}(\partial R(S) \setminus \partial K) \times \frac{(n-1)!}{\sqrt{2}}$$

and

$$P(S, \bar S) = \frac{(n-1)!\, \operatorname{vol}_{n-1}(\partial R(S) \setminus \partial K)}{2\sqrt{2}(n-1)\,|\Omega|}. \qquad (5.5)$$

Furthermore, the unit vector normal to any facet has ℓ_1-norm $\|u\|_1 = \sqrt{2}$. Taking f identically 1 in Theorem 4.3, we have, for $|S| \le \frac{1}{2}|\Omega|$,

$$\frac{\operatorname{vol}_n R(S)}{\sqrt{2}\operatorname{vol}_{n-1}(\partial R(S) \setminus \partial K)} \le \frac{\operatorname{diam} K}{2},$$

which, in the light of (5.4), is equivalent to

$$\operatorname{vol}_{n-1}(\partial R(S) \setminus \partial K) \ge \frac{\sqrt{2}|S|}{n!}.$$

Combining this inequality with (5.5) yields

$$P(S, \bar S) \ge \frac{|S|}{2n(n-1)|\Omega|}$$

whence

$$\Phi \ge \frac{1}{2n(n-1)}.$$

The claimed bound on mixing time now follows from Theorem 4.2. □

By far the most important application of the techniques we've seen in Section 4.2 is to the analysis of random walks in convex bodies. The groundbreaking work on this topic was done by Dyer, Frieze and Kannan [23], who showed that a certain natural random walk in a convex body $K \subseteq \mathbb{R}^n$ is rapidly mixing. As a consequence, they were able to exhibit the first FPRAS for approximating the volume of a convex body. (The significant point here is that the running time of the algorithm is polynomial in the dimension n, whereas all previous approaches were exponential in n.) In this application the state space comes ready equipped with a geometric interpretation, so the isoperimetric argument is a natural candidate.

The random walk employed in [23] was akin to a traditional unbiased random walk on a (sufficiently fine) n-dimensional lattice, but restricted to the interior of the body. The time complexity of the resulting sampling procedure was a high-degree polynomial in the dimension n. The perceived importance of the volume estimation problem spurred various authors to improve on Dyer et al.'s proposal in various directions: widening the range of applicability, refining the algorithmic techniques and sharpening the analysis.

tools. Applegate and Kannan [9] extended the method to cover integration of log-concave functions; Lovász and Simonovits [50] replaced the grid walk with a kind of 'idealised' continuous random walk known as the 'ball walk'; and Dyer and Frieze [20] introduced an improved isoperimetric inequality. Refer to Kannan [41] for an overview of the area, and Kannan, Lovász and Simonovits [42] to learn the state of the art.

5.3 Colourings of a Low-Degree Graph via Coupling

We return to the Markov chain $\mathcal{M}_{\rm col}(G,q)$ of Section 3, and use the coupling method to analyse its mixing time for graphs G of low degree.

Lemma 5.3. *Let G be a graph of maximum degree Δ on n vertices. Assuming $q \geq 2\Delta + 1$, the mixing time $\tau(\varepsilon)$ of the Markov chain $\mathcal{M}_{\rm col}(G,q)$ is bounded above by*

$$\tau(\varepsilon) \leq \frac{q-\Delta}{q-2\Delta}\, n \ln\left(\frac{n}{\varepsilon}\right) \leq \Delta n \ln\left(\frac{n}{\varepsilon}\right)$$

In order to define an appropriate coupling in this instance, the following easy technical lemma is useful.

Lemma 5.4. *Let U be a finite set, A, B be subsets of U, and Z_A, Z_B be random variables, taking values in U, such that*

(i) for all $a \in A$, $\Pr(Z_A = a) = \frac{1}{|A|}$

(ii) for all $b \in B$, $\Pr(Z_B = b) = \frac{1}{|B|}$.

Then there is a joint sample space for Z_A and Z_B such that

$$\Pr(Z_A(\omega) = Z_B(\omega)) = \frac{|A \cap B|}{\max\{|A|,|B|\}}.$$

The proof of Lemma 5.4 is left as an easy exercise.

Proof of Lemma 5.3. The proof is adapted from [38]; note however that the proof there applies to a Metropolis-style Markov chain, rather than the heat-bath dynamics version considered here.

We construct a coupling as in section 4.2, but now taking account of the constraints imposed by the edges of G. For all $v \in V$ denote by $\Gamma(v) \subseteq V$ the set of all neighbours of v in G, and by $X_t(v)$ (respectively, $Y_t(v)$) the colour of vertex v in colouring X_t (respectively, Y_t). Further, for all $U \subseteq V$, let $X_t(U) = \{X_t(u) : u \in U\}$. The transition $(X_t, Y_t) \to (X_{t+1}, Y_{t+1})$ in the coupling is defined by the following experiment.

(1) Select a vertex $v \in V$, u.a.r.

(2) Choose a colour $c_X \in Q \setminus X_t(\Gamma(v))$ and a colour $c_Y \in Q \setminus Y_t(\Gamma(v))$ u.a.r., using the joint sample space of Lemma 5.4.

(3) In the colouring X_t (respectively Y_t), recolour vertex v with colour c_X (respectively c_Y) to obtain a new colouring X_{t+1} (respectively Y_{t+1}).

Let $A = A_t \subseteq V$ be the set of vertices on which the colourings X_t and Y_t agree, and $D = D_t \subseteq V$ be the set on which they disagree. Let $d'(v)$ denote the number of edges incident at vertex v that have one endpoint in A and one in D. Observe that

$$\sum_{v \in A} d'(v) = \sum_{v \in D} d'(v) = m', \tag{5.10}$$

where m' is the number of edges of G that span A and D.

It is clear that $|D_{t+1}| - |D_t| \in \{-1, 0, 1\}$. Consider first the probability that $|D_{t+1}| = |D_t| + 1$. For this event to occur, the vertex v selected in step (1) must lie in A, and the new colours c_X and c_Y selected in step (2) must be unequal. Fix a vertex $v \in A$, and denote by $\xi = |Q \setminus X_t(\Gamma(v))|$ (respectively $\eta = |Q \setminus Y_t(\Gamma(v))|$) the number of possible values for c_X (respectively c_Y), and by $\zeta = |Q \setminus (X_t(\Gamma(v)) \cup Y_t(\Gamma(v)))|$ the number of possible common values. By Lemma 5.4, conditional on vertex v being selected in step (1), the probability that the same colour is selected for vertex v in both X_{t+1} and Y_{t+1} is

$$\Pr(c_X = c_Y) = \frac{\zeta}{\max\{\xi, \eta\}}. \tag{5.11}$$

A moment's reflection reveals that the quantities ξ, η and ζ satisfy the following linear inequalities:

$$\xi - \zeta \leq d'(v), \tag{5.12}$$

$$\eta - \zeta \leq d'(v) \tag{5.13}$$

and

$$\zeta \geq q - \Delta - d'(v). \tag{5.14}$$

Thus, starting from (5.11),

$$\Pr(c_X = c_Y) \geq \frac{\zeta}{d'(v) + \zeta} \geq 1 - \frac{d'(v)}{q - \Delta}. \tag{5.15}$$

where the first inequality is from (5.12) and (5.13), and the second from (5.14). Hence

$$\Pr(|D_{t+1}| = |D_t| + 1) \le \frac{1}{n}\sum_{v \in D_t} \frac{d'(v)}{q-\Delta} = \frac{m'}{(q-\Delta)n}, \qquad (5.16)$$

where the equality is by equation (5.11).

Now consider the probability that $|D_{t+1}| = |D_t| - 1$. For this event to occur, the vertex v selected in line (1) must lie in D_t, and the new colours c_0 and c_1 selected in step (2) must be equal. Equation (5.12) continues to hold with ξ, η and ζ defined as before. The analogues of inequalities (5.13)–(5.15) for the case $v \in D_t$ are

$$\xi - \zeta \le \Delta - d'(v),$$
$$\eta - \zeta \le \Delta - d'(v)$$

and

$$\zeta \ge q - 2\Delta + d'(v).$$

By reasoning similar to that leading to (5.16)

$$\Pr(c_0 = c_1) \ge \frac{\zeta}{\max\{\xi,\eta\}} \ge \frac{q-2\Delta}{q-\Delta} + \frac{d'(v)}{q-\Delta},$$

conditioned on v being selected in step (1). Hence

$$\Pr(|D_{t+1}| = |D_t| - 1) \ge \frac{1}{n}\sum_{v \in D_t}\left(\frac{q-2\Delta}{q-\Delta} + \frac{d'(v)}{q-\Delta}\right)$$
$$= \frac{q-2\Delta}{(q-\Delta)n}|D_t| + \frac{m'}{(q-\Delta)n}. \qquad (5.17)$$

Define

$$a = \frac{q-2\Delta}{(q-\Delta)n} \quad \text{and} \quad b = b(m') = \frac{m'}{(q-\Delta)n},$$

so that $\Pr(|D_{t+1}| = |D_t| + 1) \le b$ and $\Pr(|D_{t+1}| = |D_t| - 1) \ge a|D_t| + b$. Provided $a > 0$, i.e., $q > 2\Delta$, the size of the set D_t tends to decrease with t, and hence, intuitively at least, the event $D_t = \emptyset$ should occur with high probability for some $t \le T$ with T not too large. Since $D_t = \emptyset$ is precisely the event that coalescence has occurred, it only remains to confirm this intuition, and quantify the rate at which D_t converges to the empty set. From equations (5.16) and (5.17),

$$\mathbb{E}(|D_{t+1}| \mid D_t) \le b(|D_t| + 1) + (a|D_t| + b)(|D_t| - 1)$$
$$+ (1 - a|D_t| - 2b)|D_t|$$
$$= (1 - a)|D_t|.$$

Thus $\mathbb{E}|D_t| \le (1-a)^t |D_0| \le n(1-a)^t$, and, because $|D_t|$ is a non-negative integer random variable, $\Pr(|D_t| \ne 0) \le n(1-a)^t \le ne^{-at}$. Note that $\Pr(D_t \ne \emptyset) \le \varepsilon$, provided $t \ge a^{-1}\ln(n\varepsilon^{-1})$, establishing the result. □

Observe that this result, combined with Proposition 5.1, implies the existence of an FPRAS for q-colourings in graphs of maximum degree Δ, provided $q \ge 2\Delta + 1$. With a little care, the argument can be pushed to $q \ge 2\Delta$, though the bound on mixing time worsens by a factor of about n^2.

The (direct) coupling technique described here has been used in a number of other applications, such as approximately counting independent sets in a low-degree graph (Luby and Vigoda [31]), and estimating the volume of a convex body (Bubley, Dyer and Jerrum [14]).[7] In practice, the versatility of the approach is limited by our ability to design couplings that work well in situations of algorithmic interest. The next section reports on a new technique that promises to extend the effective range of the coupling argument by providing us with a powerful design tool.

6. A New Technique: Path Coupling

The coupling technique described and illustrated in Sections 4.3 and 5.3 is conceptually very simple and appealing. Unfortunately, it may be very difficult or indeed virtually impossible to design couplings appropriate to specific situations of practical interest. The problem, which began to surface even in Section 5.3, is one of engineering: how do we encourage (X_t) and (Y_t) to coalesce, while satisfying the demanding constraints (4.7) and (4.8)? Path coupling is an engineering solution to this problem, invented by Bubley and Dyer [10, 11]. Their idea is to define the coupling only on pairs of 'adjacent' states, for which the task of satisfying (4.7) and (4.8) is relatively easy, and then to extend the coupling to arbitrary pairs of states by composition of adjacent couplings along a path. The approach is not entirely distinct from classical coupling, and the Coupling Lemma (Lemma 4.4) still plays a vital role.

We illustrate path coupling in the context of the Markov chain $\mathcal{M}_{\mathrm{lin}}$ of Section 5.2 on linear extensions of a partial order. Our treatment will closely follow that of Bubley and Dyer [12]. For convenience, we work with a slightly modified version of $\mathcal{M}_{\mathrm{lin}}$. The transition from one linear extension to another is still obtained by pre-composing with a random transposition $(p, p+1)$; however, instead of selecting $p \in [n-1]$ uniformly, we select p from a probability

[7] The latter application draws inspiration from Lindvall and Rogers's [30] idea of coupling diffusions by reflection.

distribution f on $[n-1]$ that gives greater weight to values near the centre of the range. It is possible that this refinement actually reduces the mixing time, but in any case it leads to a simplification of the proof. Formally, transition probabilities from state X_t are defined by the following experiment:

(1) Select $p \in [n-1]$ according to the distribution f, and $r \in \{0,1\}$ u.a.r.

(2) If $r = 1$ and $X_t \circ (p, p+1) \in \Omega$, then $X_{t+1} := X_t \circ (p, p+1)$; otherwise, $X_{t+1} := X_t$.

Let us refer to this Markov chain as $\mathfrak{M}^f_{\mathrm{LE}}$. Provided the probability distribution f is supported on the whole interval $[n-1]$, the Markov chain $\mathfrak{M}^f_{\mathrm{LE}}$ is irreducible and aperiodic. It is easy to verify, for example using Lemma 3.1, that the stationary distribution of $\mathfrak{M}^f_{\mathrm{LE}}$ is uniform. As in Section 5.2, the explicit loop probability of $\frac{1}{2}$ is introduced mainly for convenience in the proof. However, some such mechanism for destroying periodicity is necessary in any case if we wish to treat the empty partial order consistently.

To apply path coupling, we need first to decide on an adjacency structure for the state space Ω. In this instance we decree that two states g and g' (linear extensions of $\prec$) are adjacent iff $g' = g \circ (i, j)$ for some transposition (i, j) with $0 \le i < j \le n-1$; in this case, the distance $d(g, g')$ from g to g' is defined to be $j - i$. Note that the notions of adjacency and distance are symmetric with respect to interchanging g and g', so we can regard this imposed adjacency structure as a weighted, undirected graph on vertex set Ω; let us refer to this structure as the adjacency graph. It is easily verified that the shortest path in the adjacency graph between two adjacent states is the direct one using a single edge. Thus d may be extended to a metric on Ω by defining $d(g, h)$, for arbitrary states g and h, to be the length of a shortest path from g to h in the adjacency graph.

Next we define the coupling. We need to do this just for adjacent states, as the extension of the coupling via shortest paths to arbitrary pairs of states will be automatic. Suppose the current pair of states is (X_t, Y_t) and that $Y_t = X_t \circ (i, j)$ for some transposition (i, j) with $0 \le i < j \le n-1$; then the transition to (X_{t+1}, Y_{t+1}) is defined by the following experiment:

(1) Select $p \in [n-1]$ according to the distribution f, and $r_X \in \{0,1\}$ u.a.r. If $j - i = 1$ and $p = i$, set $r_Y := 1 - r_X$; otherwise, set $r_Y := r_X$.

(2) If $r_X = 1$ and $X_t \circ (p, p+1) \in \Omega$ then set $X_{t+1} := X_t \circ (p, p+1)$; otherwise, set $X_{t+1} := X_t$.

(3) If $r_Y = 1$ and $Y_t \circ (p, p+1) \in \Omega$ then set $Y_{t+1} := Y_t \circ (p, p+1)$; otherwise, set $Y_{t+1} := Y_t$.

We need to show:

Lemma 6.1. *For adjacent states X_t and Y_t,*

$$E[d(X_{t+1}, Y_{t+1}) \mid X_t, Y_t] \le \rho\, d(X_t, Y_t), \qquad (6.1)$$

where $\rho < 1$ is a constant depending on f. For a suitable choice for f, one has $\rho = 1 - \alpha$, where $\alpha = 6/(n^3 - n)$.

Before proceeding with the proof of Lemma 6.1, let us pause to consider why it is sufficient to establish (6.1) just for adjacent states.

Lemma 6.2. *Suppose a coupling (X_t, Y_t) has been defined for $\mathfrak{M}^f_{\mathrm{LE}}$ on adjacent pairs of states, and suppose that the coupling satisfies the contraction condition (6.1) on adjacent pairs. Then the coupling can be extended to all pairs of states in such a way that (6.1) holds unconditionally.*

Proof (sketch). For notational convenience set $X = X_t$ and $Y = Y_t$, where $X_t, Y_t \in \Omega$ are now arbitrary. Denote by $P(\cdot,\cdot)$ the transition probabilities of $\mathfrak{M}^f_{\mathrm{LE}}$. Let $X = Z_0, Z_1, \ldots, Z_\ell = Y$ be a shortest path from X to Y in the adjacency graph. (Assume a deterministic choice rule for resolving ties.) First select $X' = Z_0' \in \Omega$ according to the probability distribution $P(X, \cdot)$. Now select Z_1' according to the distribution induced by the pairwise coupling of the adjacent states Z_0 and Z_1, conditioned on the choice of Z_0'; then select Z_2' using the pairwise coupling of Z_1 and Z_2, and so on, ending with $Z_\ell' = Y'$. Let $X_{t+1} = X'$ and $Y_{t+1} = Y'$. It is routine to verify, by induction on path length ℓ, that Y_{t+1} has been selected according to the (correct) distribution $P(Y_t, \cdot)$. Moreover, by linearity of expectation and (6.1),

$$\begin{aligned} E[d(X_{t+1}, Y_{t+1}) \mid X_t, Y_t] &\le \sum_{i=0}^{\ell-1} E[d(Z_i', Z_{i+1}') \mid Z_i, Z_{i+1}] \\ &\le \rho \sum_{i=0}^{\ell-1} d(Z_i, Z_{i+1}) \\ &= \rho\, d(X_t, Y_t). \end{aligned}$$

□

Proof of Lemma 6.1. If $p \notin \{i-1, i, j-1, j\}$ then the moves made in steps (2) and (3) either both succeed or both fail. Thus $Y_{t+1} = X_{t+1} \circ (i, j)$ and $d(X_{t+1}, Y_{t+1}) = j - i = d(X_t, Y_t)$. Summarising,

$$d(X_{t+1}, Y_{t+1}) = d(X_t, Y_t), \quad \text{if } p \notin \{i-1, i, j-1, j\}. \qquad (6.2)$$

Next suppose $p = i - 1$ or $p = j$. These cases are symmetrical, so we consider only the former. With probability at least $\frac{1}{2}$, the tests made in steps (2) and (3) both fail, since $\Pr(r_X = r_Y = 0) = \frac{1}{2}$. If this happens, clearly, $d(X_{t+1}, Y_{t+1}) = j - i = d(X_t, Y_t)$. Otherwise, with probability at most $\frac{1}{2}$, one or other test succeeds. If they both succeed, then

$$\begin{aligned} Y_{t+1} &= Y_t \circ (i-1, i) \\ &= X_t \circ (i,j) \circ (i-1,i) \\ &= X_{t+1} \circ (i-1,i) \circ (i,j) \circ (i-1,i) \\ &= X_{t+1} \circ (i-1,j), \end{aligned}$$

and $d(X_{t+1}, Y_{t+1}) = j - i + 1 = d(X_t, Y_t) + 1$. If only one (say the one in step 2) succeeds, then $Y_{t+1} = Y_t = X_t \circ (i,j) = X_{t+1} \circ (i-1,i) \circ (i,j)$, and $d(X_{t+1}, Y_{t+1}) \leq j - i + 1 = d(X_t, Y_t) + 1$. Summarising:

$$\mathrm{E}[d(X_{t+1}, Y_{t+1}) \mid X_t, Y_t, p = i - 1 \vee p = j] \leq d(X_t, Y_t) + \frac{1}{2}. \qquad (6.3)$$

Finally suppose $p = i$ or $p = j - 1$. Again, by symmetry, we need only consider the former. There are two subcases, depending on the value of $j - i$. The easier subcase is $j - i = 1$. If $r_X = 1$ then $r_Y = 0$ and

$$X_{t+1} = X_t \circ (i, i+1) = Y_t \circ (i, i+1) \circ (i, i+1) = Y_t = Y_{t+1},$$

with a similar conclusion when $r_X = 0$. Thus $d(X_{t+1}, Y_{t+1}) = 0 = d(X_t, Y_t) - 1$. The slightly harder subcase is the complementary $j - i \geq 2$. The crucial observation is that $X_t \circ (i, i+1), Y_t \circ (i, i+1) \in \Omega$ and hence the tests in steps (2) and (3) either both succeed or both fail, depending only on the value of $r_X = r_Y$. To see this, observe that

$$X_t(i) \not\prec X_t(i+1) = Y_t(i+1) \not\prec Y_t(i) = X_t(j),$$

from which we may read off the fact that $X_t(i)$ and $X_t(i+1)$ are incomparable in $\prec$. The same argument applies equally to $Y_t(i)$ and $Y_t(i+1)$. If $r_X = 0$ there is no change in state; otherwise, if $r_X = 1$,

$$\begin{aligned} X_{t+1} &= X_t \circ (i, i+1) \\ &= Y_t \circ (i,j) \circ (i, i+1) \\ &= Y_{t+1} \circ (i, i+1) \circ (i,j) \circ (i, i+1) \\ &= Y_{t+1} \circ (i+1, j) \end{aligned}$$

and $d(X_{t+1}, Y_{t+1}) = j - i - 1 = d(X_t, Y_t) - 1$. Summarising both the $j - i = 1$ and $j - i \geq 2$ subcases:

$$\mathrm{E}[d(X_{t+1}, Y_{t+1}) \mid X_t, Y_t, p = i \vee p = j - 1] \leq \delta(X_t, Y_t), \qquad (6.4)$$

where

$$\delta(X_t, Y_t) = \begin{cases} 0, & \text{if } d(X_t, Y_t) = 1; \\ d(X_t, Y_t) - \frac{1}{2}, & \text{otherwise.} \end{cases}$$

Note that, in the case $j - i = 1$, inequality (6.4) covers just one value of p, namely $p = i = j - 1$, instead of two; however, the effect is exactly counterbalanced by an expected reduction in distance of 1 instead of just $\frac{1}{2}$. Combining (6.2)–(6.4) we obtain

$$\begin{aligned} &\mathrm{E}[d(X_{t+1}, Y_{t+1}) \mid X_t, Y_t] \\ &\qquad \leq d(X_t, Y_t) - \frac{-f(i-1) + f(i) + f(j-1) - f(j)}{2}. \end{aligned}$$

Specialising the probability distribution $f(\cdot)$ to be $f(i) = \alpha(i+1)(n-i-1)$, where $\alpha = 6/(n^3 - n)$ is the appropriate normalising constant, we have, by direct calculation, $-f(i-1) + f(i) + f(j-1) - f(j) = 2\alpha(j - i)$. Since $d(X_t, Y_t) = j - i$, we obtain (6.1) with $\beta = 1 - \alpha$. □

From Lemmas 6.1 and 6.2 it is now a short step to

Proposition 6.3 *The mixing time of the Markov chain (M_t) is bounded by*

$$\tau(\varepsilon) \leq (n^3 - n)(2 \ln n + \ln \varepsilon^{-1})/6.$$

Proof. By iteration, $\mathrm{E}[d(X_t, Y_t) \mid X_0, Y_0] \leq \beta^t d(X_0, Y_0)$. For any pair of linear extensions X and Y, there is a path in the adjacency graph using only adjacent transpositions (i.e., length one edges) that swaps each incomparable pair at most once. Thus $d(X_0, Y_0) \leq \binom{n}{2} \leq n^2$, and

$$\Pr(X_t \neq Y_t) \leq \mathrm{E}[d(X_t, Y_t)] \leq (1 - \alpha)^t n^2.$$

The latter quantity is less than ε, provided $t \geq (n^3 - n)(2 \ln n + \ln \varepsilon^{-1})/6$. The result follows directly from Lemma 4.4. □

David Wilson has recently derived a similar $O(n^3 \log n)$ upper bound on mixing time when f is uniform, i.e., when the transposition $(p, p+1)$ is selected u.a.r.

New applications of path coupling are regularly being discovered. Bubley, Dyer and Greenhill [13] have presented an FPRAS for q-colourings of a low degree graph that extends the range of applicability of the one described earlier. They were able, for example, to approximate in polynomial time the number of 5-colourings of a graph of maximum degree 3, thus 'beating the 2Δ bound' that appeared to exist following the result described in Section 5.3. It is fair to say that this improvement would not have been possible without the aid of path coupling. Dyer and Greenhill have also considered independent sets in a low degree graph [28], and obtained a result similar to, but apparently incomparable with, that of Luby and Vigoda [51]. One further example was place. Cooper and Frieze [15] have applied path coupling to analyse the "Swendsen-Wang process," which is commonly used to sample configurations of the "random cluster" or ferromagnetic Potts model in statistical physics.

7. Exact Sampling by Coupling From the Past (CFTP)

The previous section perhaps struck an overly optimistic note. In the majority of cases we do not have good a priori bounds, using any of the techniques in the previous sections, on the mixing time of the Markov chains used in actual MCMC applications. When analytical bounds are weak or non-existent, we can sometimes use coupling as an algorithmic (as opposed to proof) technique. Propp and Wilson's remarkable contribution is to demonstrate that in certain circumstances, "algorithmic coupling" may be used to obtain samples from the exact stationary distribution, rather than just a close approximation. This section is based on Propp and Wilson's seminal article on exact sampling [64], and a paper of Kendall that describes an extension to their technique [45].

Suppose $\mathfrak{M}$ is an ergodic (irreducible, aperiodic) Markov chain on finite state space Ω and with transition probabilities $P : \Omega \times \Omega \to [0,1]$. (The finiteness assumption is for ease of presentation only, and plays no crucial role in what follows.) Suppose $\mathcal{F}$ is a probability distribution on functions $f : \Omega \to \Omega$ that is consistent with P in the sense that

$$\Pr(f(x) = y) = P(x,y) \quad \text{for all } x, y \in \Omega. \tag{7.1}$$

A special example of this situation arises when $\mathcal{F}$ is constructed as a product distribution from P. Thus, to sample $f \in \mathcal{F}$: (i) sample, independently for each $x \in \Omega$, a state y_x from the distribution $P(x,\cdot)$, and then (ii) let $f : \Omega \to \Omega$ be the function mapping x to y_x for all $x \in \Omega$. But just as with the trivial coupling in Section 4.3, we are in practice interested in distributions $\mathcal{F}$ that encourage coupled evolutions of f at different states (elements in the domain).

If $s < t$, and $f_s, \dots, f_{t-1} : \Omega \to \Omega$ is a indexed sequence of functions (usually the f_i will be sampled independently from $\mathcal{F}$) we denote by $F_s^t : \Omega \to \Omega$ the iterated function composition

$$F_s^t = f_{t-1} \circ f_{t-2} \circ \cdots \circ f_{s+1} \circ f_s. \tag{7.2}$$

We may perform a rather perverse t-step simulation of $\mathfrak{M}$ from some initial state $x_0 \in \Omega$ by the following procedure: (i) select $f_0, \dots, f_{t-1}$ independently from distribution $\mathcal{F}$, (ii) compute the composition $F_0^t = f_{t-1} \circ f_{t-2} \circ \cdots \circ f_1 \circ f_0$ as in (7.2), and (iii) return $F_0^t(x_0)$ as the required sample from the t-step distribution. Of course, this would be a very inefficient way of simulating $\mathfrak{M}$, requiring about $|\Omega|$ times the work of a direct simulation of a single trajectory. However, this view of proceedings will be convenient to bear in mind in what follows.

As hinted at earlier, for fixed transition probabilities $P(\cdot,\cdot)$ there is considerable flexibility in the choice of the distribution $\mathcal{F}$, allowing us to encode uniform couplings over the entire state space. The Coupling Lemma—at least an important special case of it—may be stated in this setting. Suppose $f_0, \dots, f_{t-1}$ are sampled independently from $\mathcal{F}$, and let F_0^t be as before. If there exists a function $t : (0,1] \to \mathbb{N}$ such that

$$\Pr(F_0^{t(\varepsilon)}(\cdot) \text{ is not a constant function}) \le \varepsilon,$$

then the mixing time $\tau(\varepsilon)$ of $\mathfrak{M}$ is bounded by $t(\varepsilon)$. In principle, this observation permits us to estimate the mixing time of $\mathfrak{M}$ empirically, by observing the coalescence time of the coupling defined by $\mathcal{F}$. We could then obtain samples from an approximation to the stationary distribution of $\mathfrak{M}$ by simulating $\mathfrak{M}$ for a number of steps comparable with the empirically observed mixing time. In practice, as we have already observed, the explicit evaluation of F_0^t would be computationally infeasible.

The first of the two ideas that underlie Propp and Wilson's proposal is completely original and surprising: by working with F_{-t}^0 in place of F_0^t, i.e., by "coupling from the past," (CFTP) it is possible to obtain samples from the exact stationary distribution.

Theorem 7.1. *Suppose that $f_{-1}, f_{-2}, \dots$ is a sequence of independent samples from $\mathcal{F}$. Let the stopping time T be defined as the smallest number t for which $F_{-t}^0(\cdot)$ is a constant function, and assume that $\mathrm{E}\,T < \infty$. Denote by $\hat{F}_{-\infty}^0$ the unique value of F_{-T}^0 (which is defined with probability 1). Then $\hat{F}_{-\infty}^0$ is distributed according to the stationary distribution of $\mathfrak{M}$.*

Note that the constant function F_{-t}^0 is the same constant function for all sufficiently large t, specifically for all $t \ge T$. Thus, coupling from time $-T$ is equivalent to "coupling from time $-\infty$," which is the rationale behind both the choice of notation $\hat{F}_{-\infty}^0$ and the CFTP method itself.

Proof of Theorem 7.1. Let π_0 be the distribution of the random variable $\hat{F}_{-\infty}^0$. Take one further independent sample f_0 from $\mathcal{F}$, and let $T' \le T$ be the smallest number such that $F_{-T'}^1$ is a constant function. Let $\hat{F}_{-\infty}^1$ denote the unique value of $F_{-T'}^1$, and let π_1 denote the distribution of the random variable $\hat{F}_{-\infty}^1$. By translational symmetry $\pi_1 = \pi_0$. But $\hat{F}_{-\infty}^1 = f_0(\hat{F}_{-\infty}^0)$, which implies that $\pi_0 = \pi$, i.e., is a stationary distribution for $\mathcal{M}$. ($\hat F^{0}_{-\infty}$ is obtained from $\hat F^{-1}_{-\infty}$ by effecting a single transition of $\mathcal{M}$.) But $\mathcal{M}$ is ergodic. □

Note that we did not really need to assume that $\mathcal{M}$ is ergodic, since the condition $E(T) < \infty$ implies the existence of a stationary distribution, as we commented [illegible] and it is easily verified that the stationary distribution must be unique.

The crucial idea underlying Propp and Wilson's proposal—independently discovered by others, e.g., Johnson [43]—is that in certain circumstances, specifically when the coupling F is "monotone," it is possible to evaluate F^0_{-T} without explicitly computing the function composition $f_{-1} \circ f_{-2} \circ \cdots \circ f_{-T+1} \circ f_{-T}$. Suppose that the state space Ω is partially ordered by $\preceq$, with a unique maximal element $\top$ and a unique minimum element $\bot$. We say that the coupling F is *monotone* if, for every $x, y \in \Omega$ and $f : \Omega \to \Omega$ in the support of F, the condition $x \preceq y$ entails $f(x) \preceq f(y)$. When F is monotone, the test for F^0_{-T} being a constant function is equivalent to the test $F^0_{-T}(\bot) = F^0_{-T}(\top)$. Moreover, if equality holds between $F^0_{-T}(\bot)$ and $F^0_{-T}(\top)$ then their common value is just $\hat F^0_{-\infty}$. Roughly speaking, rather than tracking $|\Omega|$ trajectories of $\mathcal{M}$, in the monotone case we just need to track two, namely the ones starting at $\bot$ and $\top$.

```
T ← 1;
repeat
    lower ← ⊥;
    upper ← ⊤;
    for t ← −T to −1:
        lower ← $f_t$(lower);
        upper ← $f_t$(upper);
    T ← 2T
until lower = upper
return lower
```

Fig. 7.1. Coupling from the past: the monotone case

Note that to compute $\hat F^0_{-\infty}$ it is not necessary to know T exactly, only an upper bound. Rather than iteratively computing F^0_{-t} for $t = 0, 1, 2, 3, \ldots$ until convergence, it is much more efficient to iterate according to the doubling scheme $t = 1, 2, 4, 8, 16, \ldots$. A general procedure for (monotone) CFTP, incorporating this algorithmic refinement, is presented as Figure 7.1.

7.1 A Monotone Example: the Random Cluster Model

The random cluster model arises in statistical physics as a dual (in some sense) of the ferromagnetic Potts model (this model is defined in great detail in the next chapter). An instance of the random cluster model is defined by an undirected graph $G = (V, E)$, and real numbers $0 \le p \le 1$ and $q \ge 0$. A configuration (state) of the model is a subset $X \subseteq E$; denote by $\Omega = 2^E$ the set of all configurations. Each configuration X is assigned a weight $w(X) := p^{|X|}(1-p)^{m-|X|}q^{c(X)}$, where $m = |E|$ and $c(X)$ is the number of connected components of the graph $H = (V, X)$. Let $Z := \sum_{X \subseteq E} w(X)$. Then the random cluster model specifies a probability distribution (Gibbs distribution) $\pi : \Omega \to [0, 1]$ on the set of configurations, where

$$\pi(X) := w(X)/Z, \tag{7.3}$$

for all $X \subseteq E$. In the special case $q = 1$ and $G = K_n$ (the complete graph on n vertices), the random cluster model reduces to the standard random graph model $\mathcal{G}_{n,p}$ [7]. When q is a positive integer, the random cluster model is equivalent (in a strong sense) to the ferromagnetic q-state Potts model, as was first observed by Fortuin and Kasteleyn [29]. For more on this, see e.g., Edwards and Sokal [25].

Suppose we wish to obtain random samples from the Gibbs distribution with the aim, for example, of estimating the average size of a "cluster" (connected component) of the graph (V, X). We construct a Markov chain $\mathcal{M}_{RC} = \mathcal{M}_{RC}(G, p, q)$ on the set of configurations Ω by defining transition probabilities according to the following trial:

(1) Suppose the current state is $X \subseteq E$. Select $e \in E$, u.a.r., and let

$$\theta_{X,e} := \frac{w(X + e)}{w(X - e) + w(X + e)}.$$

(2) Select $u \in [0, 1]$ u.a.r. If $u < \theta_{X,e}$, set $X' := X + e$; otherwise, set $X' := X - e$. The next state is X'.

It is easily to verify that $\mathcal{M}_{RC}$ is ergodic and, using Lemma 5.1, that its stationary distribution is the Gibbs distribution (7.3).

The threshold $\theta_{X,e}$ can be interpreted as the probability, in the Gibbs distribution, that edge e is present in a random configuration X', conditioned on the event $X' - e = X - e$, i.e., that X' and X agree except perhaps at e. The transition probabilities defined above are another example application of the heat-bath dynamics. Note that $\theta_{X,e}$ is easy to compute from the explicit expression

$$\theta_{X,e} = \begin{cases} p, & \text{if } \kappa(X+e) = \kappa(X-e); \\ p/(p+(1-p)q), & \text{otherwise.} \end{cases} \qquad (7.4)$$

The trial just described is easily extended to a (coupled) coupling, simply by using the same choice of random edge e and number r are used independently of X. Specifically, the probability distribution F is defined by the following trial.

(1) Select $e \in E$ and $r \in [0,1]$ u.a.r.

(2) Define the function $f : \Omega \to \Omega$ by

$$f(X) = \begin{cases} X + e, & \text{if } r \le \theta_{X,e}; \\ X - e, & \text{otherwise.} \end{cases}$$

The function f is a random sample from F.

This coupling is monotone with respect to the inclusion ordering on configurations (subsets), provided $q \ge 1$; i.e., for any two states $X, Y \in \Omega$ with $X \subseteq Y$, and any function f in the support of F, it is the case that $f(X) \subseteq f(Y)$. To see this, simply observe that for any such pair of states, $\theta_{X,e} \le \theta_{Y,e}$ for all $e \in E$.

For any integer $q \ge 3$, Gore and Jerrum [30] have shown that the mixing time of $\mathcal{MC}_{rc}(G,p,q)$ may be exponential in n, the number of vertices in the graph G. The important special case $q = 2$, equivalent to the celebrated ferromagnetic Ising model in statistical physics, is completely open: it may be the case that the mixing time of $\mathcal{MC}_{rc}(G,p,q)$ is bounded by $\mathrm{poly}(n, p^{-1})$ uniformly over G, but there is thus far evidence either way. Nevertheless, the point about coupling from the past is exactly this: we don't need a priori analytical bounds on the mixing time: we can just implement the coupling suggested above and proceed empirically.

Figure 7.2 illustrates the result of one such experiment. Here we see Propp-Wilson CFTP applied to the random cluster model on a 10×10 rectangular grid at $q = 2$ and $p = \sqrt{2}/(1+\sqrt{2})$. (The chosen values for p and q correspond to the Ising model at the critical temperature for the infinite 2-dimensional square lattice.) To save space, not all the doubling steps demanded by the procedure of Figure 7.1 are illustrated. Salient features to note are that $F^0_{-t}(\bot)$ (respectively, $F^0_{-t}(\top)$) is monotonically increasing (respectively decreasing) with t, and that $F^0_{-t}(\bot) \subseteq F^0_{-\infty} \subseteq F^0_{-t}(\top)$ for all $t \ge 0$. As t increases, we learn more about the identity of $F^0_{-\infty}$. Convergence in this case is surprisingly rapid when one considers that the expected number of steps before all 180 edges in the grid have been selected is about 1019 (cf. the "coupon collector" problem). Note that after 1024 steps the lower and upper bounds differ in just five edges, and that convergence proper occurs at most twice that many steps.

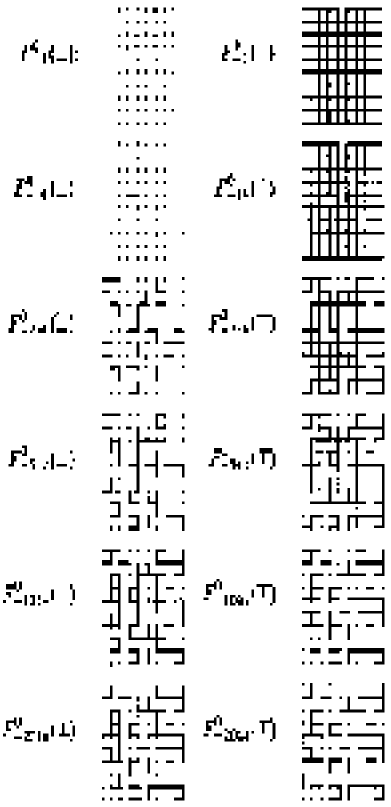


Fig. 7.2. A sample run of coupling from the past

7.3 A Non-Monotone Example: Random Forests

When $q < 1$, the coupling just described for the random cluster model ceases to be monotone; worse still, no monotone coupling exists. (The existence of a monotone coupling when $q \ge 1$ is connected to the "FKG inequality," which fails when $q < 1$.) Fortunately, Huber [40] has shown how to extend the Propp-Wilson framework to encompass many non-monotone situations. In the original Propp-Wilson proposal, the two extreme trajectories of a Markov chain 𝔐—starting from the extreme states 0 and 1—are known to bound all the others, so we can be certain that once those two extreme trajectories have converged then so have all the others. Kendall's idea is that the two bounding trajectories do not have to be bona fide simulations of 𝔐; it is enough that the upper one remains above all of the actual trajectories (in the specified partial order), while the lower one remains below.

In general, the situation is as follows. Recall that Ω is endowed with a partial order $\preceq$. An interval I of Ω is defined by two elements $l, u \in \Omega$ with $l \preceq u$, and consists of all points lying between l and u; thus $I = \{x \in \Omega : l \preceq x \preceq u\}$. Denote by $\mathbb{I} = \mathbb{I}(\Omega)$ the set of all intervals of Ω. The probability distribution $\mathcal{F}$ is extended to a distribution $\mathcal{F}'$ on pairs (f, g), where $f : \Omega \to \Omega$ and $g : \mathbb{I} \to \mathbb{I}$. As before, we stipulate that the component f satisfies (7.1), which roughly says that the coupling defined by $\mathcal{F}'$ has the correct marginals. The condition that replaces monotonicity is

$$x \in I \text{ entails } f(x) \in g(I), \text{ for all } I \in \mathbb{I} \text{ and } (f,g) \in \operatorname{supp} \mathcal{F}'. \qquad (7.5)$$

By analogy with (7.3), define

$$G_{t_0}^{t} = g_{t-1} \circ g_{t-2} \circ \cdots \circ g_{t_0}, \qquad (7.6)$$

where $(f_{t_0}, g_{t_0}), \ldots, (f_{t-1}, g_{t-1})$ are random samples from $\mathcal{F}'$. It follows from condition (7.5) that $G^0_{-T}([0,1]) = \{x_0\}$ implies that $F^0_{-T}(\cdot)$ is the constant function x_0, which in turn implies $F^0_{-\infty} = x_0$. So we have the following extension to Theorem 7.1:

Theorem 7.2. *Suppose that $(f_{-1}, g_{-1}), (f_{-2}, g_{-2}), \ldots$ is a sequence of independent samples from $\mathcal{F}'$. Let the stopping time T be defined as the smallest number t for which $G^0_{-t}([0,1]) = \{x_0\}$, for some $x_0 \in \Omega$, and assume that $\Pr(T < \infty) = 1$. Then x_0 (which is defined with probability 1) is distributed according to the stationary distribution of 𝔐.*

Note that the samples $f_{-1}, f_{-2}, \ldots$ are a conceptual convenience only, having no algorithmic significance. The algorithm for the Kendall variant of CFTP is a simple modification of the monotone one presented in Figure 7.1: simply replace the lines

```
lower ← $f_t$(lower);
upper ← $f_t$(upper);
```

by

```
(lower, upper) ← $g_t$(lower, upper);
```

As an illustrative example, let us consider how CFTP might be applied to the random cluster model with $0 \le q < 1$. The probability distribution $\mathcal{F}'$ is specified by the following trial:

(1) Select $e \in E$ and $\alpha \in [0,1]$ u.a.r.

(2) Define the function $f : \Omega \to \Omega$ by

$$f(X) = \begin{cases} X + e, & \text{if } \alpha < \theta_{X,e}; \\ X - e, & \text{otherwise,} \end{cases}$$

where $\theta_{X,e}$ is defined as in (7.4).

(3) Define the function $g : \mathbb{I} \to \mathbb{I}$ by

$$g(L,U) = \begin{cases} (L + e, U + e), & \text{if } \alpha < \theta_{U,e}; \\ (L - e, U + e), & \text{if } \theta_{U,e} \le \alpha < \theta_{L,e}; \\ (L - e, U - e), & \text{if } \alpha \ge \theta_{L,e}. \end{cases}$$

(4) The pair (f, g) is a random sample from $\mathcal{F}'$.

Informally, the function g updates its first or "lower" argument using the threshold $\theta_{U,e}$ appropriate to its second or "upper" argument, and vice versa. This definition ensures that g preserves intervals—that is to say, $L \subseteq U$ and $(L', U') = g(L, U)$ entail $L' \subseteq U'$—even though f itself is not monotone. Indeed, it is routine to verify that condition (7.5) holds with $\mathcal{F}'$ defined as above.

The picture to have in mind is that the iterates F_t of f define coupled sample paths of $\mathfrak{M}_{\mathrm{RC}}$ starting at all possible initial states. When $q \ge 1$ (the monotone case) these paths move in an orderly fashion, and their joint evolution is summarised by the lowest and uppermost sample paths $F_t(\emptyset)$ and $F_t(E)$. When $q < 1$ the sample paths are mutually crossing and recrossing each other; nevertheless, the iterates $G_t(\emptyset, E)$ continue to provide conservative lower and upper bounds on their joint evolution.

The set of forests (acyclic, spanning, not necessarily connected subgraphs) of a graph G endowed with the uniform distribution can be regarded as the set of configurations of the limit of the random cluster model as $p, q \to 0$ with $p/q \to 1$. Explicitly, the threshold $\theta_{X,e}$ in this limit is

$$\theta_{X,e} = \begin{cases} 0, & \text{if } c(X + e) = c(X - e); \\ \frac{1}{2}, & \text{otherwise.} \end{cases}$$

Plugging this threshold into the non-monotone coupling for $\mathfrak{M}_{\mathrm{RC}}(G; p, q)$ with $q < 1$, we obtain (in principle) an exact sampler for forests in a graph G. As

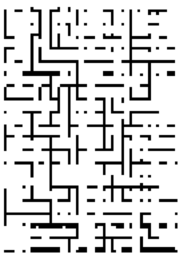


Fig. 7.3. Exact sampling of a random forest in a 20 × 20 square grid

an experiment, ten runs of this sampler were conducted with G being the 20 × 20 square grid. Figure 7.3 illustrates the end result of a typical run. All ten runs terminated within 2^{30} steps (about 12 minutes on a Sun Ultra-SPARC E450), with an average run time of about 7 minutes. This seems to be the limit of the method: the run time degrades rapidly beyond the 20 × 20 grid, and the 30 × 30 grid appears to be inaccessible. Nevertheless, it is perhaps surprising that the apparently very conservative lower and upper bounds provided by $(\hat{F}_L, \hat{F}_U)$ should converge in any realistic time bound. It certainly seems worth experimenting further with this approach. See Häggström and Nelander [31] for some more extensive experiments with non-monotone CFTP.

7.2 Further Applications

Exact sampling by CFTP and other methods is a thriving research topic, and only a small sample of the burgeoning literature will be mentioned here. Refer to Wilson's online bibliography [59] for a much wider selection. Sampling from Markov random fields was covered (in the monotone case) in Propp and Wilson's original article [54], and (more generally) by Häggström and Nelander [31]. A further twist was introduced by Kendall [45] in applying CFTP to a situation (area-interaction point processes) where there is no natural "top state."

In statistical physics, one is concerned with infinite Markov random fields, the Ising model on the infinite 2-dimensional square lattice being a prime example. In a remarkable development, van den Berg and Steif [6] point out that it is possible in some cases to sample exactly from infinite random fields, even though the configurations are unbounded in extent. The sense in which infinite configurations may be "sampled" is the following: given any positive integer N, the sampler produces, with probability 1, a configuration on the $[-N, N] \times [-N, N]$ grid which is a $(2N+1) \times (2N+1)$ "window" onto a perfectly sampled infinite configuration. The insight here is that, with probability 1, the spin (= state = colour) at a given lattice site (= vertex) at time 0 can be computed by coupling from a point in time only finitely many steps before and within a region of the lattice stretching only finitely far from the site in question. To get a picture of this, think of "light cones" of relativistic physics, which if bounded temporally must be bounded spatially too. See also Kendall [46].

CFTP à la Propp and Wilson requires a simultaneous coupling on all states Ω—encapsulated in the probability distribution $\mathcal{F}$—rather than the more familiar and less demanding pairwise coupling. Fill's version of exact sampling [28] requires only pairwise coupling, and deals with the (at least psychologically significant) problem of bias induced by "user impatience." Since the running time of the Propp-Wilson sampler is unbounded, there is a danger that an impatient user will abort a run, leading to a biased sample. Fill's proposal has the property that if the user decides to abort a run after some number of steps have elapsed, the samples obtained are not biased.

8. Key Open Problems

There are many unresolved questions in the area of rapid mixing and approximate counting. A few of the most pressing are collected together in this section.

8.1 Matroid Bases

Perhaps the major open problem in this area, and one that would be very rich in terms of consequences, is to determine useful bounds on the mixing time of the basis-exchange Markov chain for a general matroid. (A matroid is an algebraic structure that provides an abstract treatment of the concept of linear independence.) The states of this Markov chain are the bases (maximum independent sets) of a given matroid, and a transition is available from base B to base B' if the symmetric difference of B and B' consists of precisely two elements of the ground set. All transition probabilities are equal, so the chain is ergodic and reversible with uniform stationary distribution.

A concrete example is provided by the graphic matroid associated with an undirected graph G. In this case, the bases are spanning trees of G, and a transition from a given tree T is effected by adding a single edge (selected u.a.r.) to T, thus creating a cycle, and then breaking the cycle by deleting one of its edges (relevant here). The basis-exchange Markov chain is known to be rapidly mixing for graphic matroids, and, somewhat more generally, for matroids satisfying a certain "balance condition" (see Feder and Mihail [27]). A proof of rapid mixing in the general case would imply the existence of an FPRAS for a number of important problems in combinatorial enumeration, all of which are #P-complete, including counting connected spanning subgraphs of a graph (network reliability), forests of given size in a graph, and independent subsets of vectors in a set of n-vectors over GF(2).

8.2 Permanent of a 0,1 Matrix

Is there an FPRAS for the permanent of a general 0,1 matrix? Equivalently, is there an FPRAS for the number of perfect matchings in a bipartite graph? Note that this problem is not phrased as a question about the mixing time of a specific Markov chain, and certainly the chain $\mathfrak{M}_{\mathrm{match}}$ described in Section 5.1 is not directly applicable. To have a good chance of observing perfect matchings (or "dimer covers"), the parameter λ must be of order m_{n-1}/m_n; however, it is possible to construct graphs where this ratio is exponential in n. Nevertheless, the Markov chain Monte Carlo method seems to offer the best hope for a positive resolution of this question. Essentially, the issue is whether the Markov chain $\mathfrak{M}_{\mathrm{match}}$ can be suitably adapted to provide a general solution, or perhaps used as a "black box" following some ingenious preprocessing of the input matrix. (This latter idea has been used in a weaker way by Jerrum and Vazirani [38] to obtain a randomised approximation scheme for the general 0,1 permanent whose running time, while still not polynomial, is asymptotically significantly faster than that of more naive methods.)

8.3 Contingency Tables

Consider the following task: given $m+n$ positive integers $r_1,\dots,r_m$ and $c_1,\dots,c_n$, sample u.a.r. from the set of $m\times n$ non-negative integer matrices ("contingency tables") with row sums $r_1,\dots,r_m$ and column sums $c_1,\dots,c_n$. This problem arises in the interpretation of the results of certain kinds of statistical experiment; see, for example, Diaconis and Efron [19].

An elegant direct approach to sampling contingency tables has been proposed by Diaconis. Consider the Markov chain $\mathfrak{M}_{\mathrm{CT}}$, whose state space is the set of all matrices with specified row and column sums, and whose transition probabilities are defined as follows. Let the current state (matrix) be $A=(a_{ij})$. Select a pair of rows (i,i') with $i\neq i'$, and a pair of columns (j,j') with $j\neq j'$, both u.a.r. Form a new matrix A' from A by incrementing by one the array elements $a_{ij}, a_{i'j'}$, and decrementing by one the elements $a_{ij'}, a_{i'j}$. Note that A' has the same row- and column-sums as A. If A' is non-negative then we accept it as the next state; otherwise the chain remains at state A. It is easy to verify that $\mathfrak{M}_{\mathrm{CT}}$ is ergodic and reversible with uniform stationary distribution. Moreover, it appears to work well in practice as a uniform sampling procedure for contingency tables. However, its mixing time is not known to be bounded by any polynomial in the size of the input. (We assume that the row- and column-sums are expressed in unary notation when defining the input size; otherwise even the shortest path between two states may be exponentially long.) Dyer, Kannan and Mount [25] have a partial result.

To deal with tables with large entries, a natural idea is to use a kind of heat-bath dynamics. As before, select a pair of rows (i,i') with $i\neq i'$, and a pair of columns (j,j') with $j\neq j'$. Now choose the new matrix A' u.a.r. from those which agree with A except at the four entries a_{ij}, $a_{ij'}$, $a_{i'j}$ and $a_{i'j'}$ (and have the correct row and column sums). Again, little is known about the mixing time in general, but see Dyer and Greenhill [24] for a special case.

9. Details

Proof of Proposition 3.4. The techniques we employ are standard in the area [37]. Recall from Section 3 (refer to equation (3.2)) that we have expressed the number of q-colourings of G as a product

$$|\Omega(G)| = q^n \varrho_1 \cdots \varrho_m \tag{9.1}$$

where

$$\varrho_i = \frac{|\Omega(G_i)|}{|\Omega(G_{i-1})|}.$$

Suppose that the graphs G_i and G_{i-1} differ in the edge $\{u,v\}$, which is present in G_i but absent from G_{i-1}. Clearly, $\Omega(G_i)\subseteq\Omega(G_{i-1})$. Any colouring in $\Omega(G_{i-1})\setminus\Omega(G_i)$ assigns the same colour to u and v, and may be perturbed to a colouring in $\Omega(G_i)$ by recolouring vertex u with one of at least $q-\Delta\geq 1$ colours. (To resolve ambiguity, let u be the smaller of the two vertices.) On the other hand, each colouring in $\Omega(G_i)$ can be obtained in at most one way as the result of such a perturbation; hence $|\Omega(G_{i-1})\setminus\Omega(G_i)|\leq|\Omega(G_i)|$ and

$$\frac{1}{2}\leq\varrho_i\leq 1. \tag{9.2}$$

To avoid trivialities, assume $0 < \varepsilon \le 1$ and $m \ge 1$. Let $Z_i \in \{0,1\}$ denote the random variable which results from running the postulated almost uniform sampler on the graph G_{i-1} and returning one if the resulting q-colouring is also a q-colouring of G_i, and zero otherwise. Denote by $\mu_i = \mathbb{E}(Z_i)$ the expectation of Z_i. By setting $\delta = \varepsilon/6m$, we may ensure

$$\varrho_i - \frac{\varepsilon}{6m} \le \mu_i \le \varrho_i + \frac{\varepsilon}{6m}, \qquad (9.3)$$

or, using inequality (9.2),

$$\left(1 - \frac{\varepsilon}{4m}\right)\varrho_i \le \mu_i \le \left(1 + \frac{\varepsilon}{4m}\right)\varrho_i. \qquad (9.4)$$

So the mean of a sufficiently large (but still polynomial) number of independent copies of Z_i will provide a good estimate for ϱ_i. Note that, by inequalities (9.2) and (9.3), $\mu_i \ge \frac{1}{2}$.

So let $Z_i^{(1)}, \dots, Z_i^{(s)}$ be a sequence of $s = \lceil 74\varepsilon^{-2}m \rceil \le 75\varepsilon^{-2}m$ independent copies of the random variable Z_i obtained from independent trials using the postulated almost uniform sampler, and let $\bar{Z}_i = s^{-1}\sum_{j=1}^{s} Z_i^{(j)}$ be their mean. Since Z_i is a random variable taking values from $\{0,1\}$, it follows easily that $\mu_i^{-2}\operatorname{Var}(Z_i) = \mu_i^{-1} - 1 \le 1$, and hence $\mu_i^{-2}\operatorname{Var}(\bar{Z}_i) \le s^{-1}$. As our estimator for $|\Omega(G)|$, we use the random variable $Y = q^n \bar{Z}_1\bar{Z}_2 \dots \bar{Z}_m$. Note that $\mathbb{E}(Y) = q^n \mu_1\mu_2 \dots \mu_m$.

The performance of this estimator is characterised by its variance, which may be bounded as follows:

$$\begin{aligned}\frac{\operatorname{Var}(\bar{Z}_1\bar{Z}_2 \dots \bar{Z}_m)}{(\mu_1\mu_2 \dots \mu_m)^2} &= \prod_{i=1}^{m}\left(1 + \frac{\operatorname{Var}(\bar{Z}_i)}{\mu_i^2}\right) - 1 \\ &\le \left(1 + \frac{1}{s}\right)^m - 1 \\ &\le \exp\left(\frac{\varepsilon^2}{74}\right) - 1 \\ &\le \frac{\varepsilon^2}{36}\end{aligned}$$

since $e^{x/74} \le 1 + x/36$ provided $0 \le x \le 1$. Thus, by Chebyshev's inequality,

$$\left(1 - \frac{\varepsilon}{3}\right)\mu_1\mu_2 \dots \mu_m \le q^{-n}Y \le \left(1 + \frac{\varepsilon}{3}\right)\mu_1\mu_2 \dots \mu_m,$$

with probability at least $\frac{3}{4}$. But from inequality (9.4), we have

$$\left(1 - \frac{\varepsilon}{3}\right)\varrho_1\varrho_2 \dots \varrho_m \le \mu_1\mu_2 \dots \mu_m \le \left(1 + \frac{\varepsilon}{3}\right)\varrho_1\varrho_2 \dots \varrho_m$$

which, combined with the previous inequality and (9.1), implies that the estimator Y satisfies the requirements of a randomised approximation scheme for the number of colourings $|\Omega(G)|$.

To estimate each ratio ϱ_i we need $O(\varepsilon^{-2}m)$ samples from the almost uniform sampler, and there are m such ratios in all to estimate. The claimed time complexity for approximate counting follows. □

Proof of equation (4.8). Consider a facet $R(c) \cap R(c')$, where c and c' are adjacent states (colourings). Up to symmetry, such a facet is a $(qn-1)$-dimensional polytope defined by inequalities

$$1 \ge x_{0,0} = x_{0,1} \ge x_{0,2}, x_{0,3}, \dots, x_{0,q-1} \ge 0 \qquad (9.5)$$

$$1 \ge x_{1,0} \ge x_{1,1}, x_{1,2}, \dots, x_{1,q-1} \ge 0 \qquad (9.6)$$

$$\vdots$$

$$1 \ge x_{n-1,0} \ge x_{n-1,1}, x_{n-1,2}, \dots, x_{n-1,q-1} \ge 0. \qquad (9.7)$$

This particular facet corresponds to the boundary between the all-0 state and the adjacent state in which vertex 0 assumes colour 1; the facet clearly lies in the plane defined by $x_{00} = x_{01}$.

We wish to compute $\operatorname{vol}_{qn-1}(R(c) \cap R(c'))$, the area (i.e., $(qn-1)$-dimensional volume) of the facet $R(c) \cap R(c')$. Each line of the above display refers to a different set of q variables, so the required volume is the product of the volumes of the polytopes defined by each line. The polytope defined by (9.5) is of dimension $q-1$ and all the others, namely (9.6)–(9.7), are of dimension q. The q-dimensional volume of the polytope defined by any of (9.6)–(9.7) is simply

$$\int_0^1 x^{q-1}\,dx = \left[\frac{x^q}{q}\right]_0^1 = \frac{1}{q}. \qquad (9.8)$$

To calculate the volume of the polytope defined by (9.5), project it onto the plane $x_{00} = 0$ to obtain the polytope

$$1 \ge x_{01} \ge x_{02}, x_{03}, \dots, x_{0,q-1} \ge 0,$$

which, by comparison with (9.8), has $(q-1)$-dimensional volume $(q-1)^{-1}$. Projecting from the plane $x_{00} = x_{01}$ to the plane $x_{00} = 0$ contracts volume by a factor $\sqrt{2}$ (the scalar product of the normals to the two planes) so the actual volume before projection is $\sqrt{2}(q-1)^{-1}$.

Multiplying the n factors just computed together, we obtain

$$\operatorname{vol}_{qn-1}(R(c) \cap R(c')) = \frac{\sqrt{2}}{q^{n-1}(q-1)}$$

as claimed. □

Acknowledgements. Two colleagues merit special acknowledgement for their contribution to this survey. Some of the results presented here were products of an extended period of collaboration with Alistair Sinclair, and in describing these I have freely borrowed and adapted material from our joint articles. The experiment described in Section 7.3 was jointly undertaken with Vivek Gore, and is published here for the first time.

I also thank Bruce Reed and an anonymous referee for carefully reading and providing helpful comments on a draft of this chapter.

References

1. Aldous D. (1983): Random walks on finite groups and rapidly mixing Markov chains, Séminaire de Probabilités XVII 1981/82 (A. Dold and B. Eckmann, eds), Springer Lecture Notes in Mathematics 986, 243–297.
2. Aldous D. (1987): On the Markov chain simulation method for uniform combinatorial distributions and simulated annealing, Probability in the Engineering and Informational Sciences 1, 33–46.
3. Aldous D. (1990): The random walk construction of uniform spanning trees and uniform labelled trees, SIAM Journal of Discrete Mathematics 3, 450–465.
4. Alon N. (1986): Eigenvalues and expanders, Combinatorica 6, 83–96.
5. Applegate D. and Kannan R. (1991): Sampling and integration of near log-concave functions, Proceedings of the 23rd Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 156–163.
6. van den Berg J. and Steif J.E. (1994): On the existence and nonexistence of finitary codings for a class of random fields. Preprint.
7. Bollobás B. (1978): Extremal Graph Theory, Academic Press.
8. Broder A.Z. (1986): How hard is it to marry at random? (On the approximation of the permanent), Proceedings of the 18th Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 1986, 50–58. Erratum in Proceedings of the 20th Annual ACM Symposium on Theory of Computing, p. 551.
9. Brooks R.L. (1941): On colouring the nodes of a network, Proceedings of the Cambridge Philosophical Society 37, 194–197.
10. Bubley R. and Dyer M. (1997). Path coupling, Dobrushin uniqueness, and approximate counting, Report 97.04, School of Computer Studies, University of Leeds.
11. Bubley R. and Dyer M. (1997). Path coupling: a technique for proving rapid mixing in Markov chains, Proceedings of the 38th IEEE Symposium on Foundations of Computer Science, IEEE Computer Society Press, 223–231.
12. Bubley R. and Dyer M. (1998): Faster random generation of linear extensions, Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), ACM/SIAM, 350–354.
13. Bubley R., Dyer M.E. and Greenhill C. (1998): Beating the 2Δ bound for approximately counting colourings: a computer-assisted proof of rapid mixing, Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), ACM/SIAM, 355–363.
14. Bubley R., Dyer M. and Jerrum M. (1998): An elementary analysis of a procedure for sampling points in a convex body, Random Structures and Algorithms 12, 213–235.
15. Cooper C. and Frieze A.M. (1998): Mixing Properties of the Swendsen-Wang Process on Classes of Graphs, Preprint.
16. Cordovil R. and Moreira M.L. (1993): Bases-cobases graphs and polytopes of matroids, Combinatorica 13, 157–165.
17. Diaconis P. (1988): Group representations in probability and statistics, Institute of Mathematical Statistics, Hayward CA.
18. Diaconis P. and Efron B. (1985): Testing for independence in a two-way table: new interpretations of the chi-square statistic, Annals of Statistics 13, 845–913.
19. Diaconis P. and Stroock D. (1991): Geometric bounds for eigenvalues of Markov chains, Annals of Applied Probability 1, 36–61.
20. Dyer M. and Frieze A. (1991): Computing the volume of convex bodies: a case where randomness provably helps, Probabilistic Combinatorics and its Applications, Proceedings of AMS Symposia in Applied Mathematics 44, 123–170.
21. Dyer M. and Frieze A. (1994): Random walks, totally unimodular matrices, and a randomised dual simplex algorithm, Mathematical Programming 64, 1–16.
22. Dyer M., Frieze A. and Kannan R. (1991): A random polynomial-time algorithm for approximating the volume of convex bodies, Journal of the ACM 38, 1–17.
23. Dyer M. and Greenhill C. (1998): On Markov chains for independent sets. Preprint. (Available from [illegible])
24. Dyer M. and Greenhill C. (1998): A genuinely polynomial-time algorithm for sampling two-rowed contingency tables, 25th EATCS International Colloquium on Automata, Languages and Programming, Aalborg, Denmark, Springer-Verlag LNCS Series.
25. Dyer M., Kannan R. and Mount J. (1997): Sampling contingency tables, Random Structures and Algorithms 10, 487–506.
26. Edwards R.G. and Sokal A.D. (1988): Generalization of the Fortuin-Kasteleyn-Swendsen-Wang representation and Monte Carlo algorithm, Physical Review D 38, 2009–2012.
27. Feder T. and Mihail M. (1992): Balanced matroids, Proceedings of the 24th Annual ACM Symposium on Theory of Computing, ACM Press, 26–38.
28. Fill J.A. (1997): An interruptible algorithm for perfect sampling via Markov chains, Proceedings of the 29th Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 688–695.
29. Fortuin C.M. and Kasteleyn P.W. (1972): On the random cluster model I: introduction and relation to other models, Physica 57, 536–564.
30. Gore V. and Jerrum M. (1997): The Swendsen-Wang process does not always mix rapidly, Proceedings of the 29th ACM Symposium on Theory of Computation, ACM Press, 674–681.
31. Häggström O. and Nelander K. (1997): Exact sampling from anti-monotone systems. Preprint. To appear in Statistica Neerlandica.
32. Heilmann O.J. and Lieb E.H. (1972): Theory of monomer-dimer systems, Communications in Mathematical Physics 25, 190–232.
33. Jerrum M. (1995): A very simple algorithm for estimating the number of k-colourings of a low-degree graph, Random Structures and Algorithms 7, 157–165.
34. Jerrum M.R. and Sinclair A.J. (1989): Approximating the permanent, SIAM Journal on Computing 18, 1149–1178.
35. Jerrum M.R. and Sinclair A.J. (1989): Approximate counting, uniform generation and rapidly mixing Markov chains, Information and Computation 82, 93–133.
36. Jerrum M. and Sinclair A. (1993): Polynomial-time approximation algorithms for the Ising model, SIAM Journal on Computing 22, 1087–1116.

37. Jerrum M. and Sinclair A. (1996): The Markov chain Monte Carlo method: an approach to approximate counting and integration. In Approximation Algorithms for NP-hard Problems (Dorit Hochbaum, ed.), PWS, 482–520.

38. Jerrum M.R., Valiant L.G. and Vazirani V.V. (1986): Random generation of combinatorial structures from a uniform distribution, Theoretical Computer Science 43, 169–188.

39. Jerrum M. and Vazirani U.V. (1996): A mildly exponential approximation algorithm for the permanent, Algorithmica 16, 392–401.

40. Johnson V.E. (1996): Studying convergence of Markov chain Monte Carlo algorithms using coupled sample paths, Journal of the American Statistical Association 91, 154–166.

41. Kannan R. (1994): Markov chains and polynomial time algorithms, Proceedings of the 35th Annual IEEE Symposium on Foundations of Computer Science (FOCS), Computer Society Press, 656–671.

42. Kannan R., Lovász L. and Simonovits M. (1996): Random Walks and an $O^*(n^5)$ Volume Algorithm for Convex Bodies, Preprint, January.

43. Karp R.M. and Luby M. (1983): Monte-Carlo algorithms for enumeration and reliability problems, Proceedings of the 24th Annual IEEE Symposium on Foundations of Computer Science, Computer Society Press, 56–64.

44. Karzanov A. and Khachiyan L. (1990): On the conductance of order Markov chains, Technical Report DCS 268, Rutgers University.

45. Kendall W.S. (1996): Perfect simulation for the area-interaction point process, University of Warwick Department of Statistics Research Report 292. To appear in Probability Perspective (C.C. Heyde and L. Accardi, editors), World Scientific Press, Singapore.

46. Kendall W.S. (1997): Perfect simulation for spatial point processes, University of Warwick Department of Statistics Research Report 308, 1997. To appear in Proceedings of ISI 51st session, Istanbul, August, 1997.

47. Kenyon C., Randall D. and Sinclair A. (1993): Matchings in lattice graphs, Proceedings of the 25th Annual ACM Symposium on Theory of Computing (STOC), ACM Press, 738–746.

48. Knuth D.E. (1975): Estimating the efficiency of backtrack programs, Mathematics of Computation 29, 121–136.

49. Lindvall T. and Rogers L.C.G. (1986): Coupling of Multidimensional Diffusions by Reflection, Annals of Probability 14, 860–872.

50. Lovász L. and Simonovits M. (1993): Random walks in a convex body and an improved volume algorithm, Random Structures and Algorithms 4, 359–412.

51. Luby M. and Vigoda E. (1997): Approximately counting up to four, Proceedings of the 29th Annual ACM Symposium on Theory of Computation (STOC), ACM Press, 682–687.

52. Metropolis N., Rosenbluth A.W., Rosenbluth M.N., Teller A.H. and Teller E. (1953): Equation of state calculation by fast computing machines, Journal of Chemical Physics 21, 1087–1092.

53. Mihail M. and Winkler P. (1992): On the number of Eulerian orientations of a graph, Proceedings of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), ACM Press, 138–145.

54. Propp J.G. and Wilson D.B. (1996): Exact sampling with coupled Markov chains and applications to statistical mechanics, Random Structures and Algorithms 9, 223–252.

55. Rasmussen L.E. (1994): Approximating the permanent: a simple approach, Random Structures and Algorithms 5, 349–361.

56. Sinclair A.J. (1992): Improved bounds for mixing rates of Markov chains and multicommodity flow, Combinatorics, Probability and Computing 1, 351–370.

57. Sinclair A.J. (1993): Randomised algorithms for counting and generating combinatorial structures, Advances in Theoretical Computer Science, Birkhäuser, Boston.

58. Welsh D.J.A. (1997): Approximate Counting, in Surveys in Combinatorics, London Mathematical Society Lecture Note 241, Cambridge University Press, 287–323.

59. Wilson D. (1998): Annotated Bibliography of Perfectly Random Sampling with Markov Chains, http://dimacs.rutgers.edu/~dbwilson/exact.html/

Percolation and the Random Cluster Model: Combinatorial and Algorithmic Problems

Dominic Welsh*

University of Oxford

1. Introduction

In 1961, Harry Frisch, John Hammersley and I [13] carried out what were in those days massive Monte Carlo experiments attempting to determine the critical percolation probabilities of the various standard lattices. The calculations at that time were, as today, tedious indeed. The programmes were written in machine code on a computer which was the size of a large room with less power than a modern day calculator. Today, the situation has radically changed. Several of these critical probabilities which we were trying to estimate are now known exactly. However the problems posed then have been replaced by problems of just as much interest and seeming intractability and it is some of these that I shall address in these lectures.

The plan of this article is as follows. In the first section I shall review classical percolation theory and then discuss from a combinatorial point of view the Ising, Potts and random cluster models. In §5 I shall survey properties of the Tutte polynomial and in particular highlight its relationship with the previous three models. In §6 I shall return to the random cluster model. The remaining sections are concerned with the difficulties involved in obtaining good approximation schemes for the partition function of the Potts and random cluster models.

The graph terminology used is standard. The complexity theory and notation follows Garey and Johnson [18]. Further details of many of the concepts treated here can be found in [51].

2. Classical Percolation Theory

As its name suggests, percolation theory is concerned with flow in random media. Its origin is in 1957 in the work of Broadbent & Hammersley [6] was as a model for molecules penetrating a porous solid, electrons migrating over an atomic lattice, a solute diffusing through a solvent, or disease infecting a community. Here we shall attempt to introduce the main concepts of classical percolation theory and also to relate it with other topics such as the Ising model of ferromagnetism, the reliability problem in random networks, the Potts model of statistical physics and the random cluster model of Fortuin and Kasteleyn [11].

For illustrative purposes we shall be principally concerned with the two dimensional square lattice L. However the basic ideas apply to any regular lattices in arbitrary dimensions.

Suppose that there is a supply of fluid at the origin and that each edge of L allows fluid to pass along it with probability p, independently for each edge. Let $P_n(p)$ be the probability that at least n vertices of L get wet by the fluid. Thus

$$P_1(p) = 1$$
$$P_2(p) = 1 - (1-p)^4$$

and in theory $P_N(p)$ can be calculated for any integer N. However, the reader will rapidly find it prohibitively time consuming. Obviously

$$P_N(p) \geq P_{N+1}(p)$$

and hence we know that $P(p)$ exists where

$$P(p) = \lim_{N \to \infty} P_N(p) \qquad (2.1)$$

and it represents the probability that fluid spreads an infinite distance from the origin.

Broadbent and Hammersley [6] showed that (for a wide class of lattices) there exists a critical probability p_H such that

$$p < p_H \Rightarrow P(p) = 0 \qquad (2.2)$$
$$p > p_H \Rightarrow P(p) > 0,$$

and Monte Carlo simulations suggest that for all the well-known lattices the behaviour of $P(p)$ is roughly the same in the qualitative sense.

Historically, the subject of percolation had its roots in the physics literature, and in this area 'bond' is usually used to denote an 'edge' of a graph and likewise 'site' or 'atom' denotes a 'vertex'. We shall use these terms interchangeably.

In site percolation on L instead of each edge of L being randomly blocked with probability $1-p$ or open with probability p the vertices of L are blocked

* Supported in part by Esprit Working Group No. 7097 "RAND"

with probability $1-p$ or open with probability p. Again we are interested in the probability of fluid spreading locally or an infinite distance.

Exactly analogous results hold for atom percolation as for bond percolation though of course the numerical values of the critical probabilities and percolation probabilities $P(p)$ differ.

It can be argued that atom percolation is the more important, on the grounds that any bond percolation problem on a lattice L can be turned into an atom percolation problem on a related lattice $\hat{L}$, got by letting each edge of L be a vertex of $\hat{L}$ and joining two vertices of $\hat{L}$ if and only if the corresponding edges of L are incident.

For any regular lattice, if $P^A(p)$, $P^B(p)$ represent respectively the atom and bond percolation probabilities then it has been known from Hammersley [19] that

$$P^A(p) \le P^B(p) \qquad 0 < p < 1 \tag{2.3}$$

Very recently, stronger versions of this inequality have been announced by Grimmett and Stacey [18].

Another way of looking at percolation theory is to regard it as the study of the distribution of white and black clusters when the edges (or vertices) of a graph are painted white with probability p and black with probability $q = 1-p$. A white cluster is a maximal connected subset of white edges where isolated vertices are regarded as clusters. Two quantities of obvious physical interest are (a) the average number of white clusters (b) the average number of vertices in a white cluster.

The Critical Probability or Probabilities

As stated earlier, p_H, the critical probability, is defined to be the critical value below which there is zero probability that fluid from a source at the origin spreads to infinitely many points. At least two other 'critical probabilities' occur in the literature and there is still confusion about the relationship between them. The first, p_T, is defined to be the critical value of p above which the expected number of points wet by fluid from the origin becomes infinite. Now if there is a positive probability that infinitely many points are wet then a fortiori the average number of points wet is infinite. Thus for any lattice,

$$p_T \le p_H \tag{2.4}$$

Essam and Sykes [11] in a very ingenious paper, obtained some precise results about a quantity p_E which they call the critical probability but which is defined in terms of singularities of functions giving the mean number of clusters on the lattice. For example, for bond percolation on the square lattice L, they proved that

$$p_E(L) = \frac{1}{2} \tag{2.5}$$

and for the triangular lattice T and hexagonal lattice H they showed that

$$p_E(T) = 2\sin(\pi/18) = 1 - p_E(H) \tag{2.6}$$

It seems to be extremely difficult to relate p_E with either of the other two critical probabilities p_T and p_H, and physically it does not appear (from its definition at least) to be as natural an object as p_T or p_H. Exact rigorous bounds for p_T and p_H on general lattices seem difficult to obtain. However, for the bond percolation problem on the square lattice Kesten [27] showed that $p_T = p_H$ and that this common value was 1/2. Wierman [41] extended Kesten's argument and proved a similar result for the hexagonal and triangular lattices thus verifying the earlier result of Essam and Sykes.

For rigorous elegant accounts of the very considerable progress made on percolation problems see the monographs of Kesten [28] and Grimmett [15]. We close this section by stating two outstanding open problems.

Problem. Find good bounds or better still, exact values for the critical probabilities of (a) site percolation on the square lattice and (b) bond or site percolation on the 3-dimensional cubic lattice.

3. The Ising and Q-State Potts Models

We first consider two classical models of statistical physics, namely the *Ising model* and the *Q-state Potts model*.

In the *Q-state Potts model* Q is a positive integer and the sites of the underlying lattice or graph are assigned spins from the set $\{1, 2, \ldots, Q\}$. These spins then change according to the probabilistic rules to be specified later and the full spin configuration can be regarded as a Markov chain on a very large state space of size Q^n where n is the number of vertices of the underlying lattice or graph.

The limiting behaviour as time increases may vary quite considerably depending on the parameters of the model. Clear qualitative differences in behaviour constitute what is called a phase transition and deciding whether such phenomena occur and if so when, is a major area of study in statistical physics. The Ising model, which was introduced in 1925 is a mathematical model used to study such systems. It has a huge literature and is relatively well understood. The Potts model, introduced in 1952, contains the Ising model as a special case and is less well understood. This in turn is contained in the random cluster model which we describe in the next section and which is also a reasonably natural extension of the percolation model described earlier. However, in order to motivate the random cluster model we need first to describe the Ising and Potts models.

In the general Ising model on a graph or lattice G each vertex i of G is assigned a spin σ_i which is either $+1$ (called 'up') or -1 (called 'down'). An assignment of spins to all the vertices of G is called a *configuration* or *state* and is denoted by σ.

In addition each edge $e = (i,j)$ of G has an associated *interaction energy* J_{ij} which is constant but may vary from edge to edge. It measures the strength of the interaction between neighbouring pairs of vertices.

For each state $\sigma = (\sigma_1, \ldots, \sigma_n)$ define the *Hamiltonian* $H = H(\sigma)$ by

$$H(\sigma) = -\sum_{\langle ij \rangle} J_{ij}\sigma_i\sigma_j - \sum M\sigma_i \qquad (3.1)$$

where M is the external field.

The Hamiltonian $H(\sigma)$ measures the energy of the state σ.

In a ferromagnet the J_{ij} are positive; this means that a configuration of spins in which nearest neighbour pairs have parallel spins $(\sigma_i = \sigma_j)$ has a lower energy than a state in which spins are arbitrary.

The external field M has an effect of aligning spins with the direction of the field, thus again favouring states of low energy.

The *partition function* $Z = Z(G; \beta, J, M)$ is defined by

$$Z = \sum_{\sigma} e^{-\beta H(\sigma)} \qquad (3.2)$$

where the sum is over all possible spin configurations σ with $\sigma_i \in \{-1, 1\}$, and $\beta = 1/kT$ is a parameter determined by the temperature T (in absolute degrees) and where k is Boltzmann's constant. The importance of Z is that it is assumed that the probability of finding the system in a state or configuration σ is given by

$$\Pr(\sigma) = e^{-\beta H(\sigma)}/Z. \qquad (3.3)$$

Thus we see that

(i) High temperature ⇒ low value of β ⇒ probability distribution of states becomes more flat.

(ii) Low temperature ⇒ high β ⇒ greater probability to low energy states.

The quantity

$$U = -\frac{\partial}{\partial \beta}\log Z$$

is called the internal energy, and the *free energy* F is defined to be $\log Z$.

A major problem with the Ising model on a given lattice is to find a closed expression for

$$\lim_{n\to\infty} n^{-1}\log Z(G_n) \qquad (3.4)$$

where G_n is a sequence of graphs approaching in some reasonable sense the infinite lattice graph. There is no guarantee that the limit is well defined or even when well defined, will exist though there are important classes where this has been rigorously proved. On the assumption that it does, it is called the *free energy per lattice site*.

The *pair* or *two-point correlation* function is

$$\langle\sigma_i\sigma_j\rangle = \left[\sum_{\sigma}\sigma_i\sigma_j e^{-\beta H(\sigma)}\right]/Z.$$

This is a natural measure of disorder in the lattice and, as we shall see later, is closely related to percolatory behaviour in the random cluster model.

There is a very obvious generalisation of the Ising model in which each atom can be in Q different states $(Q \geq 2)$. In this model introduced by Potts [56] the energy between two interacting spins is taken to be zero if the spins are the same and equal to a constant if they are different. If we now denote the constant associated with an edge (ij) by K_{ij} then in state σ, provided we assume a zero external magnetic field, the Hamiltonian $H(\sigma)$ is defined by

$$H(\sigma) = \sum_{\langle ij \rangle} K_{ij}(1 - \delta(\sigma_i, \sigma_j))$$

where δ is the usual Kronecker delta function defined by

$$\delta(a,b) = \begin{cases} 1 & a = b \\ 0 & a \neq b \end{cases}$$

The partition function Z is again defined by

$$Z = \sum_{\sigma} e^{-\beta H(\sigma)} \qquad (3.3)$$

where the sum is over all possible spins σ.

Suppose now that we partition the edge set E into $E^{+} \cup E^{-}$ where E^{+} (E^{-}) respectively denotes the set of edges whose endpoints are the same (different) under a given state σ.

Then the contribution of σ to the Hamiltonian will be $2K(E^{-})$ where

$$K(E^{-}) = \sum_{e \in E^{-}} K_e.$$

If we now assume $K_e = K$ is constant, so that we can write $K = 2\beta J$, then

$$\begin{aligned} Z(G)_{\text{Potts}} &= \sum_{\sigma} e^{-\beta H(\sigma)} \\ &= \sum_{\sigma} e^{-K|E^{-}(\sigma)|} \end{aligned} \qquad (3.4)$$

An excellent, accessible review of the Potts model can be found in [42].

4. The Random Cluster Model

The general random cluster model on a finite graph G was introduced by Fortuin and Kasteleyn [11] and is a correlated bond percolation model on the edge set E of G defined by the probability distribution,

$$\mu(A) = Z^{-1}\left(\prod_{e \in A} p_e\right)\left(\prod_{e \notin A}(1 - p_e)\right) Q^{k(A)} \quad (A \subseteq E), \qquad (4.1)$$

where $k(A)$ is the number of connected components (including isolated vertices) of the subgraph $G : A = (V, A)$, p_e $(0 \le p_e \le 1)$ are parameters associated with each edge of G, $Q > 0$ is a parameter of the model, and Z is the normalising constant introduced so that

$$\sum_{A \subseteq E} \mu(A) = 1.$$

We will sometimes use $A(G)$ to denote the random configuration produced by μ, and P_μ to denote the associated probability distribution.

Thus in particular, $\mu(A) = P_\mu(A(G) = A)$. When $Q = 1$, μ is what Fortuin and Kasteleyn call a percolation model and when each of the p_e are made equal, say to p, then $\mu(A)$ is easily seen to be the probability that the set of open edges is A in bond percolation.

For an account of the many different interpretations of the random cluster model we refer to the original paper of Fortuin and Kasteleyn.

Here we shall be concentrating on the percolation problem when each of the p_e are equal, to say p, and, henceforth, this will be assumed.

Thus we will be concerned with a two parameter family of probability measures

$$\mu = \mu(p, Q) \quad \text{where} \quad 0 \le p \le 1 \quad \text{and} \quad Q > 0$$

defined on the edge set of the finite graph $G = (V, E)$ by

$$\mu(A) = p^{|A|} q^{|E \setminus A|} Q^{k(A)} / Z$$

where Z is the appropriate normalising constant, and $q = 1 - p$.

The reason for studying percolation in the random cluster model is its relation with phase transitions via the two-point correlation function. This was pointed out first by Fortuin and Kasteleyn and given further prominence recently by Edwards and Sokal [8] in connection with the Swendsen-Wang algorithm [31] for simulating the Potts model. We describe briefly the connection.

Let Q be a positive integer and consider the Q-state Potts model on G.

The probability of finding the system in the state σ is given by the probability

$$\Pr(\sigma) = e^{-\beta H(\sigma)} / Z.$$

The key result is the following.

Theorem 4.1. *For any pair of sites (vertices) i, j and positive integer Q, the probability that σ_i equals σ_j in the Q-state Potts model is given by*

$$\frac{1}{Q} + \frac{(Q - 1)}{Q} P_\mu(i \leftrightarrow j) \qquad (4.2)$$

where P_μ is the random cluster measure on G given by taking $p_e = 1 - e^{-K_e}$ for each edge $e = (i, j)$, and $(i \leftrightarrow j)$ is the event that under P_μ there is an open path from i to j.

The attractive interpretation of this is that the probability in (4.4) can be regarded as being made up of two component parts.

The first term $\tau_Q(i,j)$ is just the probability that under a proper random Q-colouring of the vertices of G, i and j have the same colour. The second term measures the probability of long range interaction. Thus we interpret the above as expressing an equivalence between long range spin correlations and long range percolatory behaviour.

Phase transition (in an infinite system) occurs at the onset of an infinite cluster in the random cluster model and corresponds to the spins on the vertices of the Potts model having a long range two-point correlation. Thus the random cluster model can be regarded as the extension of the Potts model to non integer Q.

5. The Tutte Polynomial

The Tutte polynomial is a polynomial in two variables x, y which can be defined for a graph, matrix or even more generally a matroid. For example each of the following is a special case of the general problem of evaluating the Tutte polynomial of a graph (or matroid) along particular curves of the (x, y) plane: (i) the chromatic and flow polynomials of a graph; (ii) the all terminal reliability probability of a network; (iii) the partition function of a Q-state Potts model; (iv) the Jones polynomial of an alternating knot; (v) the weight enumerator of a linear code over $GF(q)$.

Our study of the Tutte polynomial in what follows is motivated principally by its intimate relationship with the Ising, Potts and random cluster model.

First consider the following recursive definition of the function $T(G; x, y)$ of a graph G, and two independent variables x, y.

If G has no edges then $T(G; x, y) = 1$, otherwise for any $e \in E(G)$;

(5.1) $T(G; x, y) = T(G'_e; x, y) + T(G''_e; x, y)$, where G'_e denotes the deletion of the edge e from G and G''_e denotes the contraction of e in G,

(5.2) $T(G; x, y) = xT(G''_e; x, y)$ if e is an isthmus or coloop in a matroid,

(5.3) $T(G; x, y) = yT(G'_e; x, y)$ if e is a loop.

Remarkably, it is easy to show by induction that T is a 2-variable polynomial in x, y, which we call the *Tutte polynomial* of G.

In other words, T may be calculated recursively by choosing the edges in any order and repeatedly using (5.1–5.3) to evaluate T. The remarkable fact is that T is well defined in the sense that the resulting polynomial is independent of the order in which the edges are chosen.

Example. If G is the complete graph K_4 then

$$T(G; x, y) = x^3 + 3x^2 + 2x + 4xy + 2y + 3y^2 + y^3.$$

Alternatively, and this is often the easiest way to prove properties of T, we can show that T has the following expansion.

If $A \subseteq E(G)$, the rank of A, $r(A)$ is defined by

$$r(A) = |V(G)| - k(A), \tag{5.4}$$

where $k(A)$ is the number of connected components of the graph $G : A$ having vertex set $V = V(G)$ and edge set A.

It is now straightforward to prove:

(5.5) The Tutte polynomial $T(G; x, y)$ can be expressed in the form

$$T(G; x, y) = \sum_{A \subseteq E} (x-1)^{r(E)-r(A)}(y-1)^{|A|-r(A)}.$$

It is easy and useful to extend these ideas to matroids.

A *matroid* M is just a generalization of a matrix and can be simply defined as a pair (E, r) where E is a finite set and r is a submodular *rank function* mapping $2^E \to \mathbb{Z}$ and satisfying the conditions

$$0 \le r(A) \le |A| \quad A \subseteq E \tag{5.6}$$

$$A \subseteq B \Rightarrow r(A) \le r(B), \tag{5.7}$$

$$r(A \cup B) + r(A \cap B) \le r(A) + r(B) \quad A, B \subseteq E. \tag{5.8}$$

The edge set of any graph G with its associated rank function as defined by (5.4) is a matroid, but this is just a very small subclass of matroids known as graphic matroids.

A much larger class is obtained by taking any matrix B with entries in a field F and letting E be its set of columns and for $X \subseteq E$ defining the rank $r(X)$ to be the maximum size of a linearly independent set in X. Any abstract matroid which can be represented in this way is called representable over F.

A basic fact which we shall need is the following.

(5.9) A matroid M is representable over every field iff it has a representation over the reals by a matrix B which is totally unimodular, that is the value of every subdeterminant is 0, 1 or −1. Such a matroid is called regular. Every graphic matroid is regular.

Given $M = (E, r)$ the dual matroid is $M^* = (E, r^*)$ where r^* is defined by

$$r^*(E \setminus A) = |E| - r(E) - |A| + r(A).$$

We now just extend the definition of the Tutte polynomial from graphs to matroids by,

$$T(M; x, y) = \sum_{A \subseteq E(M)} (x-1)^{r(E)-r(A)} (y-1)^{|A|-r(A)}. \qquad (5.10)$$

Much of the theory developed for graphs goes through in this more general setting.

We close this section with what I call the "recipe theorem" from [31]. Its crude interpretation is that whenever a function f on some class of matroids can be shown to satisfy an equation of the form $f(M) = af(M'_e) + bf(M''_e)$ for some $e \in E(M)$, then f is essentially an evaluation of the Tutte polynomial.

Here M'_e is the restriction of $M = (E, r)$ to the set $E \setminus \{e\}$ with r unchanged. The contraction M''_e can be defined by $M''_e = (M^*)'_e$ and is the exact analogue of contraction in graphs. For matrices it corresponds to projection from the column vector e. A minor of M is any matroid N obtainable from M by a sequence of contractions and deletions.

The recipe theorem can now be stated as follows.

Theorem 5.1. *Let $\mathcal{C}$ be a class of matroids which is closed under direct sums and the taking of minors and suppose that f is well defined on $\mathcal{C}$ and satisfies*

$$f(M) = af(M'_e) + bf(M''_e) \qquad e \in E(M) \qquad (5.11)$$

$$f(M_1 \oplus M_2) = f(M_1)f(M_2) \qquad (5.12)$$

where $M_1 \oplus M_2$ denotes the direct sum then f is given by

$$f(M) = a^{|E|-r(E)} b^{r(E)} T\left(M; \frac{x_0}{b}, \frac{y_0}{a}\right)$$

where x_0 and y_0 are the values f takes on coloops and loops respectively.

Any invariant f which satisfies (5.11)-(5.12) is called a *Tutte-Grothendieck (TG)-invariant*.

Thus, what we are saying is that any TG-invariant has an interpretation as an evaluation of the Tutte polynomial.

Example: The Ising model

It is not difficult to show that in the absence of an external magnetic field, and with $J_e = J$ for all edges e, then whenever e is not a loop or coloop of G

$$Z(G) = e^{\beta J} Z(G'_e) + 2\sinh(\beta J) Z(G''_e).$$

Also consider the graphs C consisting of a single edge and L consisting of a single loop. Then

$$Z(C) = 2e^{\beta J} + 2e^{-\beta J} = 4\cosh(\beta J)$$
$$Z(L) = 2e^{\beta J}.$$

Thus applying the recipe theorem we get the result:

$$Z(G) = 2e^{-\beta J(|E|-|V|)} (\sinh \beta J)^{|V|-1} T(G; \coth \beta J, e^{2\beta J}).$$

Example. The Potts model

Let $b_i(\lambda)$ be the number of λ-colourings of the vertex set V of a graph G, in which there are i monochromatic or bad edges, that is they have endpoints of the same colour.

Consider the generating function

$$B(G; \lambda, s) = \sum_{i=0}^{m} s^i b_i(\lambda).$$

Clearly $B(G; \lambda, 0)$ is the chromatic polynomial of G and like $P_G(\lambda)$ we see that the following relationships hold.

(5.13) If G is connected then provided e is not a loop or coloop,

$$B(G;\lambda,s) = B(G'_e;\lambda,s) + (s-1)B(G''_e;\lambda,s)$$

(5.14) $B(G;\lambda,s) = sB(G'_e)$ if e is a loop

(5.15) $B(G;\lambda,s) = (\lambda+s-1)B(G''_e)$ if e is a coloop

Combining these, we get by using the recipe theorem

(5.16) $B(G;\lambda,s) = \lambda(s-1)^{r(E)}T(G;\frac{s+\lambda-1}{s-1},s)$

Consider now the relation with the Potts model. From (3.3) we can write

$$\begin{aligned} Z_{Potts}(G) &= \sum_{\sigma} e^{-K|F(\sigma)|} \\ &= e^{-K|E|}\sum_{\sigma} e^{K|E\setminus F(\sigma)|} \\ &= e^{-K|E|}\sum_{Q\text{-colourings}} b_j(G)e^{Kj} \\ &= e^{-K|E|}B(G;Q,e^K) \end{aligned}$$

Then using the relationship (5.16) we get

$$Z_{Potts}(G) = Q(e^K-1)^{r(E)}e^{-K|E|}T\left(G;\frac{e^K+Q-1}{e^K-1},e^K\right) \qquad (5.17)$$

It is not difficult (with hindsight) to verify that $T(G;x,y)$ can be recovered from the monochrome polynomial and therefore from the Potts partition function by using the formula

$$T(G;x,y) = \frac{1}{(y-1)^{r(E)}(x-1)}B(G;(x-1)(y-1),y). \qquad (5.18)$$

The relation of the random cluster model with T is that it is not hard to check that

$$Z(G;p,Q) = q^{r^*(E)}p^{r(E)}T\left(G;1+\frac{Qq}{p},\frac{1}{q}\right) \qquad (5.19)$$

where r^* is the dual rank and $q = 1-p$.

It follows that for any given $Q > 0$, determining the partition function Z reduces to determining T along the hyperbola H_Q given by $(x-1)(y-1) = Q$. Moreover, since in its physical interpretation, p is a probability, the reparametrization means that Z is evaluated only along the positive branch of this hyperbola. In other words, Z is the specialization of T to the quadrant $x > 1$, $y > 1$.

The antiferromagnetic Ising and Potts models are contained in T along the negative branches of the hyperbolae H_Q, but do not have representations in the random cluster model. For more on this model and its relation with T see [37, Chapter 4].

We now collect together some of the other naturally occurring interpretations of the Tutte polynomial.

(5.20) The chromatic polynomial $P(G;\lambda)$ is given by

$$P(G;\lambda) = (-1)^{r(E)}\lambda^{k(G)}T(G;1-\lambda,0)$$

where $k(G)$ is the number of connected components.

(5.21) The flow polynomial $F(G;\lambda)$ is given by

$$F(G;\lambda) = (-1)^{|E|-r(E)}T(G;0,1-\lambda)$$

(5.22) The (all terminal) reliability $R(G;p)$ is given by

$$R(G;p) = q^{|E|-r(E)}p^{r(E)}T(G;1,1/q)$$

where $q = 1-p$.

In each of the above cases, the interesting quantity (on the left hand side) is given (up to an easily determined term) by an evaluation of the Tutte polynomial. We shall use the phrase "specialises to" to indicate this. Thus for example, along $y = 0$, T specialises to the chromatic polynomial.

It turns out that the hyperbolae H_α defined by

$$H_\alpha = \{(x,y) : (x-1)(y-1) = \alpha\}$$

seem to have a special role in the theory. We note several important specialisations below.

(5.23) Along H_1, $T(G;x,y) = x^{|E|}(x-1)^{r(E)-|E|}$.

(5.24) Along H_2, when G is a graph T specialises to the partition function of the Ising model.

(5.25) Along H_q for general positive integer q, T specialises to the partition function of the Potts model.

(5.26) Along H_q when q is a prime power, for a matroid M of vectors over $GF(q)$, T specialises to the weight enumerator of the linear code over $GF(q)$, determined by M.

(5.27) Along H_q for any positive, not necessarily integer, q, T specialises to the partition function of the random cluster model discussed in §4.

(5.28) Along the hyperbola $xy = 1$ when G is planar, T specialises to the Jones polynomial of the alternating link or knot associated with G. This connection was first discovered by Thistlethwaite [35].

Some more recent applications are obtained in Welsh [40] which gives new interpretations as the expected value of classical counting functions.

Given an arbitrary graph G and $p \in [0,1]$ we denote by G_p the random subgraph of G obtained by deleting each edge of G independently with probability $1-p$.

(5.29) For any connected graph G and $0 < p \leq 1$, the random subgraph G_p has chromatic polynomial whose expectation is given by

$$EP(G_p;\lambda) = (-p)^{|V|-1}\lambda T(G; 1-\lambda p^{-1}, 1-p).$$

For the flow polynomial there is a similar, but more complicated relation, namely

(5.30) For any graph G the flow polynomial $F(G_p;\lambda)$ has expectation given by

(a) if $p \in (0,\frac{1}{2}) \cup (\frac{1}{2},1)$ then

$$EF(G_p;\lambda) = p^{r}(q-p)^{r^*}T(G;qp^{-1},1+\frac{\lambda p}{q-p})$$

where $q = 1-p$

(b) if $p = \frac{1}{2}$, then

$$EF(G_{\frac{1}{2}};\lambda) = \lambda^{|E|-|V|+k(G)}2^{-|E|}.$$

A very recent new specialisation of T concerns a version of chip firing games in [4] and gives a specific relationship between evaluations of T along the line $x = 1$ and the generating function of critical configurations in the chip firing game, we refer to [30] for details.

Other more specialised interpretations can be found in the survey by Brylawski and Oxley [8] and Welsh [37].

6. The Random Cluster Model Again

In order to be able to calculate or even simulate the state probabilities in the random cluster model it seems to be necessary to know (or be able to approximate) the partition function Z. In the case of ordinary percolation $Q = 1$ and $Z = 1$, but in general, determining Z is equivalent to determining the Tutte polynomial, as it follows from (5.19) that the following holds.

(6.1) For any finite graph G and subset A of $E(G)$, the random cluster measure μ is given by

$$\mu(A) = \frac{\left(\frac{p}{q}\right)^{|A|}Q^{k(A)}}{\left(\frac{p}{q}\right)^{r(E)}T(G;1+\frac{Qq}{p},\frac{1}{q})},$$

where T is the Tutte polynomial of G, where $q = 1-p$, and where r is given by $r(A) = |V(G)| - k(A)$.

A first consequence of this is that, as we see later, determining the measure μ is an intractable problem for most Q and most graphs.

An obvious quantity of interest is the probability that a particular set is open, that is, that every edge in the set is open. We call this the distribution function, denoted by λ and note that it is given by

$$\lambda(A) = \sum_{X \supseteq A}\mu(X).$$

The sort of questions we need to be able to answer are: how does λ vary with p and Q and how difficult is it to calculate λ?

Two very useful inequalities in working with the random cluster model are the FKG inequality of Fortuin, Kasteleyn and Ginibre [12] and an extension of this due to Holley [23], both of which we present below in Theorem 6.1 and 6.2.

The FKG inequality can be stated as follows.

Let E be a finite set and $\Omega_E = \{0,1\}^E$. Write $\mathcal{F}_E$ for the set of all subsets of Ω_E and call a probability measure μ on $(\Omega_E, \mathcal{F}_E)$ positive if $\mu(A) > 0$ for all $A \in \Omega_E$.

Theorem 6.1. *Let μ be a positive probability measure on $(\Omega_E, \mathcal{F}_E)$ such that*

$$\mu(A \cup B)\mu(A \cap B) \ge \mu(A)\mu(B)$$

for all $A, B \in \Omega_E$. Then for all increasing random functions $f, g : \Omega_E \to \mathbb{R}$,

$$\langle fg \rangle_\mu \ge \langle f \rangle_\mu \langle g \rangle_\mu,$$

where we use $\langle f \rangle$ to denote expectation with respect to the measure μ. That is

$$\langle f \rangle_\mu = \sum_{A \subseteq E} f(A)\mu(A).$$

Holley's inequality is the following.

Theorem 6.2. (Holley's inequality) *Let μ_1 and μ_2 be positive probability measures on $(\Omega_E, \mathcal{F}_E)$ such that*

$$\mu_2(A \cup B)\mu_1(A \cap B) \ge \mu_1(A)\mu_2(B)$$

for all $A, B \in \mathcal{F}_E$. Then for all increasing functions $f : \Omega_E \to \mathbb{R}$,

$$\langle f \rangle_{\mu_2} \ge \langle f \rangle_{\mu_1}.$$

Using this we almost immediately get

Proposition 6.3. *Provided $1 \le Q_1 \le Q_2$, for any fixed p, $0 \le p \le 1$ and any nondecreasing function $f : 2^E \to \mathbb{R}$,*

$$\langle f \rangle_{\mu_1} \ge \langle f \rangle_{\mu_2}$$

where μ_1 and μ_2 are the random cluster measures induced by p and Q_1, Q_2 respectively.

A special case of this gives

Corollary 6.4. *For fixed p, the distribution function λ is a monotone nonincreasing function of Q, for $Q \ge 1$.*

A fundamental question which seems difficult is the following.

(6.2) **Problem.** *How does λ vary with Q when $0 < Q < 1$?*

We now look at more combinatorial questions and consider a random cluster model $\mu = \mu(p, Q)$ on E the edge set E of a planar graph G. We follow the treatment given in [30] see also [16]. Let G^* be the dual plane graph with edge set also E identified in the natural and obvious way.

Now define the dual measure $\tilde{\mu}$ of $\mu = \mu(p, Q)$ to be the random cluster measure $\mu(\tilde{p}, \tilde{Q})$ where

$$\tilde{p} = \frac{qQ}{p + qQ}, \quad \tilde{Q} = Q.$$

Thus

$$\mu(A) = \left(\frac{qQ}{p}\right)^{|A|} Q^{-r(A)} \Big/ \left(\sum_{A \subseteq E} \left(\frac{qQ}{p}\right)^{|A|} Q^{-r(A)}\right).$$

Proposition 6.5. *For any plane graph G and random cluster measure μ*

$$P_\mu(\omega(G) = A) = P_{\tilde{\mu}}(\omega(G^*) = E \backslash A).$$

Corollary 6.6. *If G, G^* are dual planar graphs, $\tilde{\mu}$ on G^* produces white configurations with exactly the same probability distribution as μ produces black configurations on G.*

We now turn to the specific case of the square lattice. We adopt the terminology of ordinary ($Q = 1$) percolation as much as possible.

Let Λ_n denote the box on the square lattice having corners $(\pm n, \pm n)$. Let p, Q be fixed and let $\mu_n = \mu_n(p, Q)$ be the sequence of random cluster measures indexed by Λ_n as n runs through the positive integers.

The events in which we have a particular interest are of type $\{0 \leftrightarrow \partial_n\}$ denoting the event that there is an open path from 0 to ∂_n, the boundary of the box Λ_n.

(6.3) For $Q \ge 1$ and $m \ge n$,

$$\mu_{m+1}\{0 \to \partial_n\} \ge \mu_m\{0 \to \partial_n\}.$$

This is just a special case of the following.

Proposition 8.3. *Let G be a finite graph and let H be a subgraph of G on the same vertex set. If μ_G and μ_H denote the random cluster measures induced by G, H respectively for any fixed p and $Q \geq 1$, then for any monotone nondecreasing f on the edge set of G, if the value of f is determined by the state of the edges of H, then*

$$\mu_H f \leq \mu_G f.$$

Since the quantities in (8.1) are probabilities and thus bounded, we can therefore define

$$\theta_n(p, Q) = \lim_{m \to \infty} \mu_m(0 \leftrightarrow \partial S_n).$$

Now for $m \geq n$, it is trivial that

$$\mu_m(0 \leftrightarrow \partial S_n) \leq \mu_m(0 \leftrightarrow \partial S_{n-1}).$$

Consequently

$$\theta_n(p, Q) \leq \theta_{n-1}(p, Q)$$

and we define

$$\theta(p, Q) = \lim_{n \to \infty} \theta_n(p, Q)$$

to be the *percolation probability* of the model.

Note that when $Q = 1$, $\theta(p, Q)$ is essentially the same quantity as $P(p)$ defined in (2.1). Accordingly, for $Q \geq 1$, we can define the *critical probability* $p_H(Q)$ by

$$p_H(Q) = \inf\{p : \theta(p, Q) > 0\}.$$

It is easy to see that:

(8.4) For $Q \geq 1$, both critical probabilities $p_H(Q)$ and $p_T(Q)$ are monotone nondecreasing in Q. In this $p_T(Q)$ is defined analogously to p_T in §2.

In [25] it is shown that the following is true.

(8.5) For $Q \geq 1$, the critical probabilities $p_H(Q)$ and $p_T(Q)$ satisfy

$$p_T(Q) \leq \frac{\sqrt{Q}}{1 + \sqrt{Q}} \leq p_H(Q).$$

In the same paper I also conjectured that the following Q-extension of Kesten's Theorem is true.

Conjecture 8.6. *For $Q \geq 1$, the critical probability $p_c(Q)$ equals $\sqrt{Q}/(1 + \sqrt{Q})$.*

I originally made this conjecture following on from a seminar on the random cluster model by G.R. Grimmett, in Oxford in the summer of 1986. The motivation was the duality formula above and since this duality was widely known to physicists working on the Potts model I suspect that many physicists believe Conjecture 8.6 to be a proved theorem, at least for integer Q. As far as I am aware the first explicit consideration of the problem in connection with the random cluster model is in [33], see for example [17]. At the same time I readily acknowledge that, for reasons given below, this may have been a folklore conjecture (theorem) in the world of Potts modellers where Q is integral.

There is also a word of warning. Consider the model with finite graphs, the combinatorial approach described above is fine. However moving to the infinite does pose serious problems of rigour. Grimmett [17] gives a very detailed and nice account of the "Gibbs technology" and in particular discusses the existence of, perhaps uncountably infinite, set of distinct critical probabilities $p_c(Q)$.

Despite this worrying aspect of the advanced theory, a rigorous definition of $p_c(Q)$ can be given for $Q \geq 1$ and $d \geq 2$ and is according to [17] pp 276 "widely believed" to equal $\sqrt{Q}/(1 + \sqrt{Q})$ for $Q \geq 1$ and $d = 2$.

When $Q = 1$ the conjecture is certainly true by Kesten's theorem that the critical probability of the square lattice is $\frac{1}{2}$. It is also true when $Q = 2$ because using the relation $p = 1 - e^{-K}$, when $Q = 2$, this corresponds to a critical value of $\sinh^{-1} 1 = 0.88137$ for the critical exponent K, agreeing with the Onsager solution to the Ising model.

For integer $Q \geq 3$ the critical value of $p_c(Q)$ given by the conjecture agrees with the critical points of the Potts model located by singularity based arguments see for example [30]. However it does not appear easy to make these arguments rigorous in this context, and the situation seems not dissimilar from that in ordinary percolation when it took 16 years before Kesten [27] and Wierman [47] were able to give rigorous justifications of the exact values obtained by Essam and Sykes [40].

A remarkable paper by Laanait et al. [29] shows that Conjecture 8.6 is true for sufficiently large Q, certainly $Q \geq 26$ suffices, see [17] pp 276. This survey also gives an excellent account of the probabilistic background.

7. Approximation Schemes

The main result of [33] is the following:

Theorem 7.1. *The problem of evaluating the Tutte polynomial of a graph at a point (a,b) is #P-hard except when (a,b) is on the special hyperbola*

$$H_1 \equiv (x-1)(y-1) = 1$$

or when (a,b) is one of the special points $(1,1)$, $(-1,-1)$, $(0,-1)$, $(-1,0)$, $(i,-i)$, $(-i,i)$, (j,j^2), and (j^2,j), where $j = e^{2\pi i/3}$. In each of these exceptional cases the evaluation can be done in polynomial time.

Since for any graph G, $Z(p,Q)$ in the random cluster model is essentially $T(G; 1+\frac{qQ}{p}, \frac{1}{q})$ it follows that we have

Corollary 7.2. *When $Q \neq 1$, determining $Z(p,Q)$ for a general graph is #P-hard for all $p \in (0,1)$.*

As far as planar graphs are concerned, there is a significant difference. The techniques developed using the Pfaffian to solve the Ising problem for the plane square lattice by Kasteleyn [25] can be extended to give a polynomial time algorithm for the evaluation of $T(G;x,y)$ for any planar graph along the special hyperbola. However, this seems to be the limiting point for we have the following extension of Theorem 7.1 due to Vertigan and Welsh [36].

Theorem 7.3. *The evaluation of the Tutte polynomial of bipartite planar graphs at a point (a,b) is #P-hard except when*

$$(a,b) \in H_1 \cup H_2 \cup \{(1,1),(-1,-1),(j,j^2),(j^2,j)\},$$

in which cases it is computable in polynomial time.

Corollary 7.4. *Even for the class of bipartite planar graphs, evaluating $Z(p,Q)$ for general p,Q is #P-hard unless $Q = 1$ or 2.*

We are thus led to approximate or Monte Carlo methods. For positive numbers a and $r \geq 1$ we say that a third quantity $\hat{a}$ approximates a within ratio r or is an r-approximation to a, if

$$r^{-1}a \leq \hat{a} \leq ra.$$

In other words the ratio $\hat{a}/a$ lies in $[r^{-1}, r]$.

We now consider a randomised approach to counting problems and make the following definition.

An *ϵ-δ-approximation scheme* for a counting problem f is a Monte Carlo algorithm which on every input $\langle x, \epsilon, \delta \rangle$, $\epsilon > 0$, $\delta > 0$, outputs a number $\hat{Y}$ such that

$$\Pr\{(1-\epsilon)f(x) \leq \hat{Y} \leq (1+\epsilon)f(x)\} \geq 1-\delta.$$

Now let f be a function from input strings to the natural numbers. A *randomised approximation scheme* for f is a probabilistic algorithm that takes as an input a string x and a rational number ϵ, $0 < \epsilon < 1$, and produces as output a random variable Y, such that Y approximates $f(x)$ within ratio $1+\epsilon$ with probability $\geq 3/4$.

In other words,

$$\Pr\left(\frac{1}{1+\epsilon} \leq \frac{Y}{f(x)} \leq 1+\epsilon\right) \geq \frac{3}{4}.$$

A *fully polynomial randomised approximation scheme* FPRAS for a function $f : \Sigma^* \to N$ is a randomised approximation scheme which runs in time which is a polynomial function of n and ϵ^{-1}.

Suppose now we have such an approximation scheme and suppose further that it works in polynomial time. Then we can boost the success probability up to $1-\delta$ for any desired $\delta > 0$, by using the following trick of Jerrum, Valiant and Vazirani [24]. This consists of running the algorithm $O(\log \delta^{-1})$ times and taking the median of the results.

The existence of an FPRAS for a counting problem is a very strong result, it is the analogue of an *RP* algorithm for a decision problem and corresponds to the notion of tractability. However we should also note:

Proposition 7.5. *If $f : \Sigma^* \to N$ is such that deciding if f is nonzero is NP-hard then there cannot exist an FPRAS for f unless NP is equal to random polynomial time RP.*

Since this is thought to be unlikely, it makes sense only to look for an FPRAS when counting objects for which the decision problem is not *NP*-hard.

In an important paper Jerrum and Sinclair [23] have proved:

(7.1) There exists an FPRAS for the partition function of the ferromagnetic Ising model.

However it seems to be difficult to extend the argument to prove a similar result for the Q-state Potts model with $Q > 2$ and this remains one of the outstanding open problems in this area.

A second result of Jerrum and Sinclair is the following:

(7.2) There is no FPRAS for estimating the antiferromagnetic Ising partition function unless $NP = RP$.

In the context of the Tutte plane representation this can be restated as follows.

(7.3) Unless $NP = RP$, there is no FPRAS for estimating T along the curve

$$\{(x,y) : (x-1)(y-1) = 2,\ 0 < y < 1\}.$$

The following extension of this result is proved in [39]. It implies similar results about the antiferromagnetic version of the Q-state Potts model.

(7.4) On the assumption that $NP \neq RP$, the following statements are true.

(a) Even in the planar case, there is no fully polynomial randomised approximation scheme for T along the negative branch of the hyperbola H_2.

(b) For $Q = 2, 4, 5, \ldots$, there is no fully polynomial randomised approximation scheme for T along the curves

$$H_Q \cap \{y < 0\}.$$

The reader will also note that all the 'negative results' are about evaluations of T in the region outside the quadrant $x \geq 1$, $y \geq 1$. In [39] I conjecture that the following is true.

Conjecture 7.6. *There exists an FPRAS for estimating T at all points of the quadrant $x \geq 1$, $y \geq 1$. This implies and is almost equivalent to the statement that there is an FPRAS for $Z(p,Q)$ in the random cluster model for all $p, Q > 0$.*

Some evidence in support of this is the following.

If we let $\mathcal{G}_\alpha$ be the collection of graphs $G = (V,E)$ such that each vertex has at least $\alpha|V|$ neighbours then we call a class $\mathcal{C}$ of graphs dense if $\mathcal{C} \subseteq \mathcal{G}_\alpha$ for some fixed $\alpha > 0$.

Annan [2] showed that:

(7.5) There exists an FPRAS for counting forests in any class of dense graphs.

Now the number of forests is just the evaluation of Z at a point on $Q = 0$ and a more general version of this is the following result, also by Annan.

(7.6) For any class of dense graphs, there is an FPRAS for evaluating $T(G; x, 1)$ for positive integer x.

The natural question suggested is about the matroidal dual - namely, does there exist an FPRAS for evaluating T at $(1,x)$? This is the reliability question and in particular the point $(1,2)$ enumerates the number of connected subgraphs. It is impossible to combine duality with denseness so Annan's methods don't seem to work.

What can be proved is the following. The main result of Alon, Frieze and Welsh [1] can be stated as

Theorem 7.7. *There exists a fully polynomial randomised scheme for evaluating $Z(p,Q)$ for all $p \geq 0$, $Q \geq 0$ for any dense class of graphs.*

Even more recently Karger [18] has proved the existence of a similar scheme for the case of graphs with no small edge cut set. This can be stated as follows.

For $\alpha > 0$ define the class $\mathcal{G}^\alpha$ by $G \in \mathcal{G}^\alpha$ iff its edge-connectivity is at least $\alpha \log |V(G)|$. A class of graphs is well connected if it is contained in $\mathcal{G}^\alpha$ for some fixed α.

Theorem 7.8. *For any fixed (x,y), $y > 1$, there exists α, depending on (x,y) such that for any class $\mathcal{C} \subseteq \mathcal{G}^\alpha$, there is an FPRAS for evaluating $T(G;x,y)$.*

Notice that though the properties of being well connected and dense are very similar neither property implies the other.

■

Thus Conjecture 7.6 has been proved for classes of dense and well connected graphs. There is also no "natural impediment" to it being true for all graphs. However for the d-dimensional hypercubical lattice it is known that there exists $Q(d)$ such that the random cluster model has a first-order discontinuity for $Q > Q(d)$. Indeed it is believed that

$$Q(d) = \begin{cases} 4 & d = 2 \\ 2 & d \geq 6 \end{cases}$$

It is not unreasonable to associate a first-order discontinuity with an inability to approximate. There is no proof of such a general statement but there are persuasive arguments to suggest that such discontinuities would prevent an approximation scheme based on sampling by the Markov chain method. Hence a major open question must be whether or not there exists an FPRAS for the ferromagnetic random cluster model for hypercubical lattices. These are neither dense nor well connected so the above results do not apply.

8. A Geometric Approach

Two simple but key questions in terms of the work that has been done in this area are the following.

(8.1) **Problem.** *Does there exist an FPRAS for estimating either the number of forests or the number of acyclic orientations of a general graph?*

A new approach to approximation of these points is proposed by Bartels, Mount and Welsh [3]. This is based on the interpretation of T as the Ehrhart polynomial of a unimodular zonotope $Z(A)$. Counting the number of forests is the problem of counting lattice points contained in the zonotope $Z(A)$. Counting the number of acyclic orientations is the problem of counting the vertices of this zonotope. The latter is a much more difficult problem and goes some way to explaining the total lack of success with it.

We now sketch this approach.

Let $\mathbb{Z}^n$ denote the n-dimensional integer lattice in $\mathbb{R}^n$ and let P be an n-dimensional lattice polytope in $\mathbb{R}^n$, that is a convex polytope whose vertices have integer coordinates. Consider the function $i(P;t)$ which when t is a positive integer counts the number of lattice points which lie inside the dilated polytope tP. Ehrhart [9] initiated the systematic study of this function by proving that it was always a polynomial in t, and that in fact

$$i(P;t) = \chi(P) + c_1 t + \cdots + c_{n-1} t^{n-1} + \mathrm{vol}(P) t^n,$$

where

$$c_0 = \chi(P) \text{ is the Euler characteristic}$$

of P and $\mathrm{vol}(P)$ is the volume of P.

Until recently the other coefficients of $i(P,t)$ remained a mystery, even for simplices, see for example [7].

However, in the special case that P is a unimodular zonotope there is a nice interpretation of these coefficients. First recall that if A is an $r \times n$ matrix, written in the form $A = [a_1, \ldots, a_n]$, then it defines a zonotope $Z(A)$ which consists of those points p of $\mathbb{R}^r$ which can be expressed in the form

$$p = \sum_{i=1}^{n} \lambda_i a_i, \quad 0 \leq \lambda_i \leq 1.$$

In other words, $Z(A)$ is the Minkowski sum of the line segments $[0, a_i]$, $1 \leq i \leq n$.

It is a convex polytope which, when A is a totally unimodular matrix, has all integer vertices and in this case it is then known as a *unimodular zonotope*. For these polytopes a result from Stanley [34] shows that

$$i(Z(A);t) = \sum_{k=0}^{r} i_k t^k$$

where i_k is the number of subsets of columns of the matrix A which are linearly independent and have cardinality k.

In other words, the Ehrhart polynomial $i(Z(A);t)$ is the generating function of the number of independent sets in the matroid $M(A)$. But we also know that for any matroid M, the evaluation of $T(M;x,y)$ along the line $y = 1$ also gives this generating function. Hence, combining these observations we have the result

Theorem 8.1. *If M is a regular matroid and A is any totally unimodular representation of M then the Ehrhart polynomial of the zonotope $Z(A)$ is given by*

$$i(Z(A);\lambda) = \lambda^r T(M; 1 + \frac{1}{\lambda}, 1)$$

where r is the rank of M.

The approximation scheme proposed by Bartels, Mount and Welsh [3] works as follows. For any graph G the *win polytope* W_G is the convex polytope defined by

$$\sum_{e \ni v} x_e \le d(v) \quad \forall v \in V, \quad x_e \ge 0$$

where $d(v)$ is the number of edges incident with v.

It has the property that its bounding hypersurface is combinatorially equivalent to $Z(A)$ where A is any totally unimodular representation of the graphic matroid determined by G. Now carry out simple random walk X_t in a slightly dilated version of W_G (call it W'_G). Associate with each lattice point a box of equal volume ensuring that the boxes are disjoint but otherwise as large as possible. Now let t be large enough, say $t = T$, so that the stopping point X_T is almost uniform in W'_G, and map X_T to the lattice point associated with the box containing it. Accept the output as an almost uniform point of W_G if it lies inside it. Repeat N times, where N is large enough to ensure we have a good estimate of the number of lattice points inside W_G. Ideally this process would work successfully enough to enable us also to get a good estimate of the number of lattice points in the bounding face and hence $Z(A)$.

Curiously and somewhat depressingly, in order for the method to work in polynomial time we need exactly the same density condition on the underlying graph as did Annan [2]. For, alongside the remarks at the end of §7 this suggests that it might be more profitable to look for a mathematical reason why good approximation schemes should not exist for $Z(p, Q)$ for general p and Q.

Acknowledgement. I am grateful for very helpful comments from Geoffrey Grimmett and one of the referees.

References

1. Alon N., Frieze A.M. and Welsh D.J.A. (1995): Polynomial time randomised approximation schemes for Tutte-Grothendieck invariants: the dense case, Random Structures and Algorithms, 6, 459–478.
2. Annan J.D. (1994): A randomised approximation algorithm for counting the number of forests in dense graphs, Combinatorics, Probability and Computing, 3, 273–283.
3. Bartels E., Mount J. and Welsh D.J.A. (1997): The win polytope of a graph, Annals of Combinatorics 1, 1–15.
4. Björner A., Lovász L. and Shor P. (1991): Chip-firing games on graphs, European Journal of Combinatorics 12, 283–291.
5. Broadbent S.R. and Hammersley J.M. (1957): Percolation processes I. Crystals and mazes, Proceedings of the Cambridge Philosophical Society 53, 629–641.
6. Brylawski T.H. and Oxley J.G. (1992): The Tutte polynomial and its applications, Matroid Applications (ed. N. White), Cambridge Univ. Press 123–225.
7. Diaz R. and Robins S. (1996): The Ehrhart polynomial of a lattice n-simplex, Electronic Research Announcements of the American Mathematical Society, 2 (1), 1–6.
8. Edwards R.G. and Sokal A.D. (1988): Generalization of the Fortuin-Kasteleyn-Swendsen-Wang representation and Monte Carlo algorithm, Phys. Rev. D 38, 2009–2012.
9. Ehrhart E. (1967): Sur un problème de géométrie diophantienne linéaire I, II, Journal für die Reine und Angewandte Mathematik 226, 1–29 and 227, 25–49. Correction 231 (1968), 220.
10. Essam J.W. and Sykes M.F. (1964): Exact critical percolation probabilities for site and bond problems in two dimensions, J. Math. Phys. 5, 1117–1127.
11. Fortuin C.M. and Kasteleyn P.W. (1972): On the random cluster model. I. Introduction and relation to other models, Physica 57, 536–564.
12. Fortuin C.M., Kasteleyn P.W. and Ginibre J. (1971): Correlation inequalities on some partially ordered sets, Comm. Math. Phys. 22, 89–103.
13. Frisch H.L., Hammersley J.M. and Welsh D.J.A. (1962): Monte-Carlo estimates of percolation probabilities for various lattices, Physical Review 126, 949–951.
14. Garey M.R. and Johnson D.S. (1979): Computers and Intractability — A guide to the theory of NP-completeness, W.H. Freeman, San Francisco.
15. Grimmett G.R. (1989): Percolation, Springer Verlag, Berlin.
16. Grimmett G.R. (1995): The stochastic random cluster process and the uniqueness of random cluster measures, Annals of Probability 23, 1461–1510.
17. Grimmett G.R. (1997): Percolation and disordered systems, Ecole d'Eté de Probabilités de Saint Flour XXVI – 1996 (P. Bernard ed.) Lecture Notes in Mathematics 1665, Springer-Verlag, Berlin, pp. 153–300.
18. Grimmett G.R. and Stacey A.M. (1998): Critical probabilities for site and bond percolation models, preprint.
19. Hammersley J.M. (1961): Comparison of atom and bond percolation, Journal of Mathematical Physics 2, 728–733.
20. Hintermann A., Kunz H. and Wu F.Y. (1978): Exact results for the Potts model in two dimensions, J. Statist. Phys. 19, 623–632.
21. Holley R. (1974): Remarks on the FKG inequalities, Comm. Math. Phys. 36, 227–231.
22. Jaeger F., Vertigan D.L. and Welsh D.J.A. (1990): On the computational complexity of the Jones and Tutte polynomials, Math. Proc. Camb. Phil. Soc. 108, 35–53.
23. Jerrum M.R. and Sinclair A. (1990): Polynomial-time approximation algorithms for the Ising model, Proc. 17th ICALP, EATCS, 462–475.
24. Jerrum M.R., Valiant L.G. and Vazirani V.V. (1986): Random generation of combinatorial structures from a uniform distribution, Theoretical Computer Science 43, 169–188.
25. Karger D.R. (1995): A randomized fully polynomial time approximation scheme for the all terminal network reliability problem, in Proceedings of the 36th annual IEEE Symposium on Foundations of Computer Science, pp. 328–337.
26. Kasteleyn P.W. (1961): The statistics of dimers on a lattice, Physica 27, 1209–1225.
27. Kesten H. (1980): The critical probability of bond percolation on the square lattice equals 1/2, Comm. Math. Phys. 74, 41–59.
28. Kesten H. (1982): Percolation Theory for Mathematicians, Birkhäuser, Boston (1982).
29. Laanait L., Messager A., Miracle-Solé S., Ruiz J. and Shlosman S. (1991): Interfaces in Potts model I: Pirogov-Sinai theory of the Fortuin-Kasteleyn representation, Comm. Math. Phys. 140, 81–92.

30. Merino-López C. (1997) Chip-firing and the Tutte polynomial, Annals of Combinatorics 1, 253–259.
31. Oxley J.G. and Welsh D.J.A. (1979) The Tutte polynomial and percolation, Graph Theory and Related Topics (eds. J.A. Bondy and U.S.R. Murty), Academic Press, London, 329–339.
32. Potts R.B. (1952) Some generalised order-disorder transformations, Proceedings Cambridge Philosophical Society 48, 106–109.
33. Stanley R.P. (1980) Decompositions of rational convex polytopes, Annals of Discrete Mathematics 6 (1980), 333–342.
34. Swendsen R.H. and Wang J.-S. (1987) Nonuniversal critical dynamics in Monte Carlo simulations, Phys. Rev. Lett. 58, 86–88.
35. Thistlethwaite M.B. (1987) A spanning tree expansion of the Jones polynomial, Topology 26, 297–309.
36. Vertigan D.L. and Welsh D.J.A. (1992) The computational complexity of the Tutte plane: the bipartite case, Probability, Combinatorics and Computer Science 1, 181–187.
37. Welsh D.J.A. (1993a) Complexity: Knots, Colourings and Counting, London Mathematical Society Lecture Note Series 186, Cambridge University Press.
38. Welsh D.J.A. (1993b) Randomised approximation in the Tutte plane, Combinatorics, Probability and Computing 2, 137–143.
39. Welsh D.J.A. (1993c) Percolation in the random cluster model and Q-state Potts model, J. Phys. A (Math. and General) 26, 2471–2483.
40. Welsh D.J.A. (1996) Counting, colourings and flows in random graphs, Bolyai Society Mathematical Studies 2, pp 491–505.
41. Wierman J.C. (1981) Bond percolation on honeycomb and triangular lattices, Adv. Appl. Probab. 13, 298–313.
42. Wu F.Y. (1982) The Potts model, Rev. Modern Phys. 54, 235–268.

Concentration

Colin McDiarmid

Department of Statistics, University of Oxford

Summary. Upper bounds on probabilities of large deviations for sums of bounded independent random variables may be extended to handle functions which depend in a limited way on a number of independent random variables. This 'method of bounded differences' has over the last dozen or so years had a great impact in probabilistic methods in discrete mathematics and in the mathematics of operational research and theoretical computer science. Recently Talagrand introduced an exciting new method for bounding probabilities of large deviations, which often proves superior to the bounded differences approach. In this chapter we introduce and survey these two approaches and some of their applications.

1. Introduction

What do we mean by concentration here and why should we be concerned with it?

Suppose that a random variable X has expected value $\mathbf{E}(X) = \mu$ and variance $\mathbf{E}((X-\mu)^2) = \sigma^2$. Then Chebyshev's inequality states that

$$\mathbf{Pr}(|X-\mu| \geq t) \leq \sigma^2/t^2$$

for any $t > 0$. Thus for $t >> \sigma$ the probability of deviating by more than t from μ is small. However, we shall often want or need the probability of large deviations to be very small, that is, we want to know that X is strongly concentrated around μ. The archetypical concentration result is Chernoff's bound on the tails of the binomial distribution [14], in other words on the tails of the sums of independent identically distributed binary (that is, $\{0,1\}$-valued) random variables.

Theorem 1.1. *Let $X_1, X_2, \ldots, X_n$ be independent binary random variables, with $\mathbf{Pr}(X_k = 1) = p$ and $\mathbf{Pr}(X_k = 0) = 1-p$ for each k, and let $S_n = \sum X_k$. Then for any $t \geq 0$*

$$\mathbf{Pr}(|S_n - np| \geq nt) \leq 2e^{-2nt^2}.$$

Typically we shall be interested in a random variable like S_n, and not in the corresponding 'bounded differences' X_k that make it up. The variance of S_n

here is $np(1-p) = n/4$ when $p = 1/2$, and then Chebyshev's inequality yields only that $\Pr(|S_n - np| \geq \epsilon n) \leq 1/(4\epsilon^2 n)$, which will often not be a small enough bound. In some cases we shall want good bounds for their own interest, and sometimes as tools within some larger endeavour.

As an example of the former case, consider quicksort. Quicksort is one of the most important sorting algorithms, and its value rests entirely on its good typical behaviour. It is well known that it has good average time complexity. Further, the variance of the time taken is not too large, and so large deviations from the average are not very likely – see for example [36, 56]. However, one would hope that large positive deviations are very unlikely, and the bounds that can be obtained from the variance and Chebyshev's inequality are weak. It turns out [49] that the method of bounded differences shows that indeed large deviations are extremely unlikely (and the method yields essentially best possible results). We shall meet several further examples below, including the study of isoperimetric inequalities.

There are also many cases when we need to prove concentration results as a step towards something else. One example concerns the behaviour of the chromatic number of a random graph – see Section 3.1 below. Concentration inequalities have become essential tools in the probabilistic analysis of algorithms [16, 53, 63] and the study of randomized algorithms [51], and in probabilistic methods in discrete mathematics (in particular when we wish to use the Lovász Local Lemma) [4]. Some have reached standard undergraduate text books in probability – see for example [29] section 12.2, or [37] section 6.3.

We shall introduce the two main approaches for proving concentration results, namely the bounded differences or martingale method and the recent method of Talagrand, and give several applications of each. We shall also mention briefly how some such results can be proved using ideas from information theory.

The natural starting point is to consider sums of independent random variables, starting with the classical Chernoff bound introduced above. We do this in Section 2, where we give full proofs in a form which is intended to be widely accessible, and to generalise for the next section.

Section 3 is devoted to the martingale method. We shall not use any results about martingales beyond understanding the definition, and indeed the first two subsections do not even mention the word martingale. We first present the 'independent bounded differences inequality'. This is a special case of various more powerful inequalities which we develop later, but it is easy to grasp and has proved to be very useful. We give applications to bin packing, colouring random graphs and isoperimetric inequalities involving Hamming distance.

After that we present closely related extensions of the independent bounded differences inequality, namely Theorems 3.7, 3.8 and 3.9, and illustrate these extensions by describing an early application concerning permutations and a recent application to finding matchings in hypergraphs. These extensions include some results that have been presented very recently, though they can be traced back to earlier work.

In these first two subsections of Section 3 which we have just discussed, the applications are proved but not the concentration inequalities, as it is most natural to prove the concentration results in the framework of martingales. The third subsection introduces martingales onto the scene. Following that, the next subsection starts by paralleling the earlier treatment of sums of independent random variables but now considering martingale difference sequences: we find that we can readily rework the earlier proofs. Then we give a pair of more general results, Theorems 3.14 and 3.15, which include (nearly) all the previous results, and prove them in the following subsection. Thus Theorems 3.14 and 3.15 could be regarded as the most important of all the results discussed so far, but often a more focussed special case such as Theorem 3.1 or 3.9, is sufficient for an application, and is then the best tool to use. We end the section on the martingale method with a brief discussion on 'centering' sequences.

The final part, Section 4, introduces Talagrand's inequality (or rather what seems to be the most useful of his many inequalities). We give applications to increasing subsequences and common subsequences, to travelling salesman tours and Steiner trees, and to minimum spanning trees. While presenting these applications we deduce from Talagrand's inequality two useful 'packaged' results, Theorems 4.3 and 4.5, which in fact handle all the applications in this chapter. These 'packaged' results, which are tailored to our applications, are in fact rather easy deductions from Talagrand's inequality which itself is proved afterwards. Finally, we discuss briefly how results from information theory may be used to derive concentration results.

We shall stick throughout to 'bounded discrete models', typically $1, \ldots, n$. Thus there are two major related topics that we shall not discuss: for unbounded martingale results in continuous time see for example [39], and for an introduction to the asymptotic theory of large deviations see for example [20, 18, 28]. Both these topics are harder work than the discrete case we consider, and seem to be of much less use in discrete mathematics and theoretical computer science.

2. Inequalities for Sums of Bounded Independent Random Variables

We restate from above the 1952 Chernoff [16] bound on the tails of the binomial distribution.

Theorem 2.1. *Let $0 < p < 1$, let $X_1, X_2, \ldots, X_n$ be independent binary random variables, with $\Pr(X_k = 1) = p$ and $\Pr(X_k = 0) = 1 - p$ for each k, and let $S_n = \sum X_k$. Then for any $t \geq 0$,*

$$\Pr(|S_n - np| \geq nt) \leq 2e^{-2nt^2}.$$

The sum above is over k running from 1 to n. Throughout the chapter, when we write an unadorned sum $\sum$ or product $\prod$ the index k runs from 1 to n. The above result will be proved below by bounding the moment generating function $M(h) = \mathbf{E}(e^{hS_n})$ and using Markov's inequality, following the method introduced by Bernstein. Indeed, all the results of this section and the next section use this method. (See [80] for a variant of this method which yields similar results but requiring only limited independence, and see also [84].)

Recall that Markov's inequality states that for a non-negative random variable X, $\Pr(X \geq t) \leq \mathbf{E}(X)/t$ for each $t > 0$. To prove this, we use the indicator function 1_A for an event A, and note that, since $X \geq t1_{\{X \geq t\}}$, we have

$$\mathbf{E}(X) \geq t\,\mathbf{E}(1_{\{X \geq t\}}) = t \Pr(X \geq t).$$

Proof of Theorem 2.1.

Let $m = n(p + t)$. Let $h > 0$. Then

$$\Pr(S_n \geq m) = \Pr(e^{hS_n} \geq e^{hm}) \leq e^{-hm}\mathbf{E}(e^{hS_n}), \qquad (2.1)$$

by Markov's (or Bernstein's) inequality. By the independence of the random variables X_k,

$$\mathbf{E}(e^{hS_n}) = \mathbf{E}\left(\prod e^{hX_k}\right) = \prod \mathbf{E}(e^{hX_k}) = (1 - p + pe^h)^n.$$

Hence, for any $h > 0$,

$$\Pr(S_n \geq m) \leq e^{-hm}(1 - p + pe^h)^n.$$

If $0 < t < 1 - p$ then we may set $e^h = \frac{(p+t)(1-p)}{p(1-p-t)}$ to minimise the above bound, and we obtain

$$\Pr(S_n - np \geq nt) \leq e^{-2nt^2} \qquad (2.2)$$

This implies by a continuity argument that the inequality holds also for $t = 1 - p$. But the inequality is trivial for $t = 0$ or $t > 1 - p$ and thus it holds for all $t \geq 0$.

Now let $Y_k = 1 - X_k$ for each k. Then by the above result (2.2),

$$\Pr(S_n - np \leq -nt) = \Pr\left(\sum Y_k - n(1-p) \geq nt\right) \leq e^{-2nt^2}$$

for any $t \geq 0$. □

Hoeffding [39] presents extensions of the above theorem which can be based on the following lemma.

Lemma 2.2. *Let the random variables $X_1, X_2, \ldots, X_n$ be independent, with $0 \leq X_k \leq 1$ for each k. Let $S_n = \sum X_k$, let $\mu = \mathbf{E}(S_n)$, let $p = \mu/n$ and let $q = 1 - p$. Then for any $0 \leq t < q$,*

$$\Pr(S_n - \mu \geq nt) \leq \left(\left(\frac{p}{p+t}\right)^{p+t}\left(\frac{q}{q-t}\right)^{q-t}\right)^n.$$

Proof. We follow the lines of the proof of Theorem 2.1. Let $p_k = \mathbf{E}(X_k)$ for each k. Let $m = \mu + nt$, and let $h > 0$. Note that, by the convexity of the function e^{hx} for $0 \leq x \leq 1$, we have $e^{hx} \leq 1 - x + xe^h$, and so $\mathbf{E}(e^{hX_k}) \leq 1 - p_k + p_ke^h$. Thus since S_k is the sum of the independent random variables S_{k-1} and X_k,

$$\begin{aligned}\mathbf{E}(e^{hS_k}) &= \mathbf{E}(e^{hS_{k-1}})\mathbf{E}(e^{hX_k})\\ &\leq \mathbf{E}(e^{hS_{k-1}})(1 - p_k + p_ke^h)\\ &\leq \prod(1 - p_k + p_ke^h),\end{aligned}$$

on iterating. Hence

$$\mathbf{E}(e^{hS_n}) \leq (1 - p + pe^h)^n,$$

by the arithmetic mean – geometric mean inequality. But by Markov's inequality

$$\Pr(S_n \geq m) \leq e^{-hm}\mathbf{E}(e^{hS_n}) \leq e^{-hm}(1 - p + pe^h)^n.$$

Thus, for any $h \geq 0$,

$$\Pr(S_n - \mu \geq nt) \leq \left(e^{-h(p+t)}(1 - p + pe^h)\right)^n. \qquad (2.3)$$

The desired inequality now follows on setting $e^h = \frac{(p+t)(1-p)}{p(1-p-t)}$, as in the proof of Theorem 2.1. □

Our interest is in large deviations and the above bound is good in this case (though inequalities based on the normal approximation and De Moivre-Laplace are naturally better for small deviations - see for example [8]). From the above result we may deduce weaker but more useful bounds, which generalise the Chernoff bounds in Theorem 2.1 and improve on them when p is small.

Theorem 2.3. *Let the random variables $X_1, X_2, \ldots, X_n$ be independent, with $0 \le X_k \le 1$ for each k. Let $S_n = \sum X_k$, let $\mu = E(S_n)$, let $p = \mu/n$ and let $q = 1 - p$.*

(a) For any $t \ge 0$,

$$\Pr(|S_n - \mu| \ge nt) \le 2e^{-2nt^2}.$$

(b) For any $\epsilon > 0$,

$$\Pr(S_n \ge (1+\epsilon)\mu) \le e^{-((1+\epsilon)\ln(1+\epsilon)-\epsilon)\mu} \le e^{-\frac{\epsilon^2\mu}{2(1+\epsilon/3)}}.$$

(c) For any $\epsilon > 0$,

$$\Pr(S_n \le (1-\epsilon)\mu) \le e^{-\frac{\epsilon^2\mu}{2}}.$$

Part (a) is due to Hoeffding [38], who also discusses relationships between these and other similar inequalities. Results similar to parts (b) and (c) appear in [4] (in the binomial case). For similar results in the binomial case based on Stirling's approximation see [9] Chapter 1. In order to prove Theorem 2.3 we need one technical lemma.

Lemma 2.4. *For all $x \ge 0$,*

$$(1+x)\ln(1+x) - x \ge 3x^2/(6+2x).$$

Proof. Let

$$f_1(x) = (6 + 8x + 2x^2)\ln(1+x) - 6x - 5x^2.$$

We want to show that $f_1(x) \ge 0$ for all $x \ge 0$. Now $f_1(0) = 0$ and $f_1'(x) = 4f_2(x)$ where $f_2(x) = (2+x)\ln(1+x) - 2x$. It suffices to show that $f_2(x) \ge 0$ for all $x \ge 0$. Now $f_2(0) = 0$ and $f_2'(x) = (1+x)^{-1} + \ln(1+x) - 1$. Now $f_2'(0) = 0$ so it suffices to show that $f_2''(x) \ge 0$ for all $x \ge 0$. But $f_2''(x) = x(1+x)^{-2} \ge 0$ and so we are done. □

Proof of Theorem 2.3.

(a) Consider n fixed, let $q = 1 - p$ and for $0 \le t < q$ let

$$f(t) = \ln\left(\left(\frac{p}{p+t}\right)^{p+t}\left(\frac{q}{q-t}\right)^{q-t}\right).$$

Then

$$f'(t) = \ln\left(\frac{p(q-t)}{(p+t)q}\right),$$

and

$$f''(t) = -((p+t)(q-t))^{-1} \le -4.$$

Now $f(0) = f'(0) = 0$ and so it follows by Taylor's theorem that for $0 \le t < q$, $f(t) = (t^2/2)f''(s)$ for some s with $0 < s < t$. Hence $f(t) \le -2t^2$. Hence by Lemma 2.2,

$$\Pr(S_n - \mu \ge nt) \le e^{-2nt^2}. \tag{2.4}$$

By applying this result to $n - S_n$ we obtain

$$\Pr(S_n - \mu \le -nt) \le e^{-2nt^2} \tag{2.5}$$

(b) To prove part (b) it is simpler to use the inequality (2.2) in the proof of Lemma 2.2 rather than the lemma itself. If we set $t = \epsilon p$ and $e^h = (1+\epsilon)$ there, and use the inequality $1 + x \le e^x$, we obtain

$$\Pr(S_n \ge (1+\epsilon)\mu) \le \left((1+\epsilon)^{-(1+\epsilon)p}(1+\epsilon p)\right)^n \le \left((1+\epsilon)^{-(1+\epsilon)}e^{\epsilon}\right)^{\mu}$$

and this gives the first inequality in (b) (see also Appendix A of [5]). The second inequality in (b) follows from Lemma 2.4.

(c) Let the function f be as in (a) above, and let $h(x) = f(-xp)$ for $0 \le x < 1$. Then $h'(x) = -pf'(-xp)$ and

$$h''(x) = p^2 f''(-xp) = -\frac{p}{(1-x)(q+xp)} \le -p.$$

Thus we may use Taylor's theorem as above to see that $h(x) \le -px^2/2$, and then Lemma 2.2 completes the proof. □

The first inequality in part (b) yields useful results for very large deviations. In particular,

$$\Pr(S_n \ge 2\mu) \le e^{-\mu/3}. \tag{2.6}$$

Also

$$\Pr(S_n \ge k\mu) \le e^{-(k\ln k - k + 1)\mu} \le e^{-k\ln(k/e)\mu}$$

and so, if $k \ge e^2$ then

$$\Pr(S_n \ge k\mu) \le e^{-k\mu}. \tag{2.7}$$

The second inequality in part (b) yields immediately that

$$\Pr(S_n \ge (1+\epsilon)\mu) \le e^{-\frac{\epsilon^2\mu}{3}} \tag{2.8}$$

for $0 < \epsilon \le 1$, which is often a sufficiently precise inequality in applications - see for example [4]. Hoeffding also gave the following extension of part (a) above to the case when the ranges of the summands may differ.

Theorem 2.5. *Let the random variables $X_1,\ldots,X_n$ be independent, with $a_k \le X_k \le b_k$ for each k, for suitable constants a_k, b_k. Let $S_n = \sum X_k$ and let $\mu = \mathbf{E}(S_n)$. Then for any $t \ge 0$,*

$$\Pr(|S_n - \mu| \ge t) \le 2e^{-2t^2/\sum(b_k-a_k)^2}.$$

To prove this result we need a lemma from [29].

Lemma 2.6. *Let the random variable X satisfy $\mathbf{E}(X) = 0$ and $a \le X \le b$, where a and b are constants. Then for any $h > 0$*

$$\mathbf{E}(e^{hX}) \le e^{\frac{1}{8}h^2(b-a)^2}.$$

Proof. Since e^{hx} is a convex function of x, for $a \le x \le b$

$$e^{hx} \le \frac{x-a}{b-a}e^{hb} + \frac{b-x}{b-a}e^{ha},$$

and so

$$\begin{aligned}\mathbf{E}(e^{hX}) &\le \frac{b}{b-a}e^{ha} - \frac{a}{b-a}e^{hb}\\ &= (1-p)e^{-pu} + pe^{(1-p)u}\\ &= e^{-pu}(1-p+pe^{u}) = e^{f(u)}\end{aligned}$$

where $p = -a/(b-a)$, $u = (b-a)h$ and $f(u) = -pu + \ln(1-p+pe^u)$. But

$$f'(u) = -p + \frac{pe^u}{(1-p)+pe^u} = -p + \frac{p}{p+(1-p)e^{-u}},$$

and so

$$f''(u) = \frac{p(1-p)e^{-u}}{(p+(1-p)e^{-u})^2} \le \frac{1}{4}$$

(since the geometric mean is at most the arithmetic mean). Also $f(0) = f'(0) = 0$, and hence by Taylor's theorem

$$f(u) \le \frac{1}{8}u^2 = \frac{1}{8}h^2(b-a)^2,$$

which gives the desired inequality. □

Proof of Theorem 2.5. By Lemma 2.6, for $h > 0$

$$\begin{aligned}\mathbf{E}(e^{h(S_n-\mu)}) &= \mathbf{E}\left(\prod e^{h(X_k-\mathbf{E}(X_k))}\right)\\ &= \prod \mathbf{E}\left(e^{h(X_k-\mathbf{E}(X_k))}\right)\\ &\le e^{\frac{1}{8}h^2\sum(b_k-a_k)^2}.\end{aligned}$$

Hence by Markov's inequality,

$$\begin{aligned}\Pr(S_n - \mu \ge t) &\le e^{-ht}\mathbf{E}(e^{h(S_n-\mu)})\\ &\le e^{-ht+\frac{1}{8}h^2\sum(b_k-a_k)^2}.\end{aligned}$$

Now set $h = 4t/\sum(b_k-a_k)^2$ to obtain

$$\Pr(S_n - \mu \ge t) \le e^{-2t^2/\sum(b_k-a_k)^2}.$$

Finally, replace X_k by $-X_k$ to obtain

$$\Pr(S_n - \mu \le -t) \le e^{-2t^2/\sum(b_k-a_k)^2},$$

and thus complete the proof. □

Much work has also been done on tail bounds for the sum S_n when, as well as knowing bounds on the ranges of the summands X_k, we know bounds on their variances $\operatorname{var}(X_k)$ – see for example [7, 29]. The following result builds on work of Bernstein (see [7] and [29] equation (2.13)). We shall develop more general results along these lines later. The reader may notice the similarity to part (b) of Theorem 2.3.

Theorem 2.7. *Let the random variables $X_1,\ldots,X_n$ be independent, with $X_k - \mathbf{E}(X_k) \le b$ for each k. Let $S_n = \sum X_k$, and let S_n have expected value μ and variance V (the sum of the variances of the X_k). Then for any $t \ge 0$,*

$$\begin{aligned}&\Pr(S_n - \mu \ge t)\\ &\le e^{-(V/b^2)((1+\varepsilon)\ln(1+\varepsilon)-\varepsilon)} \quad \text{where } \varepsilon = bt/V \qquad (2.9)\\ &\le e^{-\frac{t^2}{2V(1+\varepsilon/3)}}. \qquad (2.10)\end{aligned}$$

In typical applications of the inequality (2.10), the 'error' term $bt/3V$ will be negligible. Suppose for example that the random variables X_k have the same bounded distribution, with positive variance σ^2, and so $V = n\sigma^2$. Then for $t = o(n)$, the bound in (2.10) is $e^{-(1+o(1))\frac{t^2}{2V}}$ (which is the natural 'target', since by the Central Limit Theorem $S_n - \mu$ is asymptotically normal with mean 0 and variance V).

In the proof of Theorem 2.5 above we used Lemma 2.6 to give a bound on the moment generating function of a bounded random variable X with expected value 0. In order to prove Theorem 2.7, we now need a related result, see [69].

Lemma 2.8. Let

$$g(x) = \frac{1}{2} + \frac{x}{6} + \frac{x^2}{24} + \cdots = (e^x - 1 - x)/x^2$$

for $x \neq 0$. Then the function g is increasing, and if the random variable X satisfies $\mathbf{E}(X) = 0$ and $X \leq b$, then

$$\mathbf{E}(e^X) \leq e^{g(b)\mathrm{var}(X)}.$$

Proof. To show that g is increasing, note that for $x \neq 0$,

$$g'(x) = x^{-3}((x-2)e^x + 2 + x),$$

and so it suffices to show that $h(x) = (x-2)e^x + 2 + x$ satisfies $h(x) \geq 0$ for all x. Now $h(0) = 0$ and $h'(x) = (x-1)e^x + 1$. Then $h'(0) = 0$ and $h''(x) = xe^x$ so $h(x) < 0$ for $x < 0$ and $h(x) > 0$ for $x > 0$, and thus indeed $g'(x) \geq 0$ for all x as required.

For the second part of the lemma, note that

$$e^x = 1 + x + x^2 g(x) \leq 1 + x + x^2 g(b)$$

for $x \leq b$. Hence, if $\mathbf{E}(X) = 0$ and $X \leq b$, then

$$\mathbf{E}(e^X) \leq 1 + g(b)\mathrm{var}(X) \leq e^{g(b)\mathrm{var}(X)},$$

as required. □

Proof of Theorem 2.7. The proof follows the lines of the proof of Theorem 2.5 above. By Lemma 2.8, for any h

$$\mathbf{E}(e^{h(S_n - \mu)}) = \prod_k \mathbf{E}\left(e^{h(X_k - \mathbf{E}(X_k))}\right) \leq e^{h^2 g(hb)V}.$$

Hence by Markov's inequality, for any $h > 0$

$$\mathbf{Pr}(S_n - \mu \geq t) \leq e^{-ht}\mathbf{E}(e^{h(S_n - \mu)}) \leq e^{-ht + h^2 g(hb)V}. \tag{2.12}$$

To minimise this bound we set $h = \frac{1}{b}\ln(1 + \frac{bt}{V})$, and then we obtain (2.10), and finally Lemma 2.4 yields (2.11).

Inequalities for maxima

All the theorems above on sums of independent random variables can be strengthened to refer to maxima. Since we have no natural applications in the present context for these strengthenings, we restrict ourselves to a comment here and then say a little more at the end of subsection 3.5.

Each of the theorems is based on the elementary Bernstein inequality

$$\mathbf{Pr}(Z \geq t) \leq e^{-ht}\mathbf{E}(e^{hZ}) \quad \text{for each } h \geq 0.$$

Consider for example the Chernoff Theorem, Theorem 2.3, where $S_n = \sum_k X_k$ and $\mu_n = \mathbf{E}(S_n)$: to prove this result we may apply the above inequality with $Z = S_n - \mu_n$ where $\mu_n = \mathbf{E}(S_n) = np$, that is we use the inequality

$$\mathbf{Pr}(S_n - \mu_n \geq t) \leq e^{-ht}\mathbf{E}(e^{h(S_n - \mu_n)}) \quad \text{for each } h \geq 0.$$

However, a stronger inequality holds. Let $S_k = \sum_{i=1}^{k} X_i$ and $\mu_k = \mathbf{E}(S_k)$: then

$$\mathbf{Pr}(\max_k (S_k - \mu_k) \geq t) \leq e^{-ht}\mathbf{E}(e^{h(S_n - \mu_n)}) \quad \text{for each } h \geq 0.$$

Here the maximum is over $k = 1, \ldots, n$. Thus the same proof as before shows that, for any $t \geq 0$,

$$\mathbf{Pr}(\max_k |S_k - kp| \geq nt) \leq 2e^{-2nt^2}.$$

However, in typical applications of concentration inequalities in discrete mathematics or theoretical computer science, we do not start with the X_k and then wish to investigate the sums $S_1, S_2, \ldots$: we start with a random quantity Z of interest and then define further random variables X_k such that $Z = \sum_k X_k$ in order to investigate Z, so that we are not interested for example in S_{n-1}.

Not only may the theorems above on sums of independent random variables be strengthened to refer to maxima, but also the same is true for many of the more general results in the next section, as they are also based on the Bernstein inequality – see the comment at the end of subsection 3.5.

3. Martingale Methods

We shall make some introductory comments about martingales in subsection 3.1 below. No knowledge of martingales will be required in the first two subsections after that. Indeed, they will not be mentioned, though we shall see later that the inequalities presented in these subsections are most naturally understood in the context of martingales, and indeed they can be called clean martingale results.

3.1 The Independent Bounded Differences Inequality

In this subsection, we introduce and give several applications for the 'independent bounded differences inequality', Theorem 3.1 below, from [45]. This result is a special case of Theorem 3.7 below (see the proof of Theorem 3.8) but it has proved very useful and is immediately accessible and so we discuss it first. (We should insist below that the function f be appropriately measurable: we ignore such details here and throughout the chapter.)

Theorem 3.1. *Let $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ be a family of independent random variables with X_k taking values in a set A_k for each k. Suppose that the real-valued function f defined on $\prod A_k$ satisfies*

$$|f(\mathbf{x}) - f(\mathbf{x}')| \leq c_k \tag{3.1}$$

whenever the vectors $\mathbf{x}$ and $\mathbf{x}'$ differ only in the kth co-ordinate. Let μ be the expected value of the random variable $f(\mathbf{X})$. Then for any $t \geq 0$,

$$\mathbf{Pr}(f(\mathbf{X}) - \mu \geq t) \leq e^{-2t^2/\sum c_k^2}. \tag{3.2}$$

The inequality (3.2) is 'one-sided'. If we apply it to $-f$ we obtain

$$\mathbf{Pr}(f(\mathbf{X}) - \mu \leq -t) \leq e^{-2t^2/\sum c_k^2}, \tag{3.3}$$

and so we have deduced the 'two-sided' inequality

$$\mathbf{Pr}(|f(\mathbf{X}) - \mu| \geq t) \leq 2e^{-2t^2/\sum c_k^2}. \tag{3.4}$$

A similar comment holds for most of the one-sided results we present.

If we let each set $A_k = [0,1]$ and let $f(\mathbf{x}) = \sum x_k$ we obtain Theorem 2.1 above, and if each set A_k is a bounded set of numbers we obtain Theorem 2.5. We consider a variety of applications below. We do not prove Theorem 3.1 at this point, as the proof is most naturally set in the framework of martingales and we shall shortly develop more general results – see in particular Theorem 3.7 below.

3.1.1 Bin Packing. Our first application is quick and easy. Given an n-vector $\mathbf{x} = (x_1, \ldots, x_n)$ where $0 \leq x_k \leq 1$ for each k, let $B(\mathbf{x})$ be the least number of unit size bins needed to store items with these sizes. We assume that the items have independent random sizes. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables each taking values in $[0,1]$. Then the bounded differences condition (3.1) holds with each $c_k = 1$, and so (as noted in [61, 54]) it follows from Theorem 3.1 that

$$\mathbf{Pr}(|B(\mathbf{X}) - \mu| \geq t) \leq 2e^{-2t^2/n} \tag{3.5}$$

where μ is the expected value of $B(\mathbf{X})$. Thus if $\omega(n) \to \infty$ as $n \to \infty$, then the probability that $B(\mathbf{X})$ deviates from its mean by more than $\omega(n)\sqrt{n}$ tends to 0 as $n \to \infty$. We may say that $B(\mathbf{X})$ is concentrated within width $O(\sqrt{n})$. For a similar result on random knapsacks see [46]. (For sharper concentration results on bin packing that use also the variance of the random variables X_k see [59, 45].)

3.1.2 Random Graphs. In Theorem 3.1 we may take A_k as a set of edges in a graph, as in the results below – see for example [13, 19]. Recall that the random graph $G_{n,p}$ has vertices $1, \ldots, n$ and the possible edges appear independently with probability p.

Lemma 3.2. *Let $(A_1, \ldots, A_m)$ be a partition of the edge set of the complete graph K_n into m blocks, and suppose that the graph function f satisfies $|f(G) - f(G')| \leq 1$ whenever the symmetric difference $E(G) \triangle E(G')$ of the edge sets is contained in a single block A_k. Then the random variable $Y = f(G_{n,p})$ satisfies*

$$\mathbf{Pr}(Y - \mathbf{E}(Y) \geq t) \leq e^{-2t^2/m} \quad \textit{for } t \geq 0.$$

This result follows directly from Theorem 3.1 with each $c_k = 1$. The next two results are immediate consequences of Lemma 3.2: for the former let A_k be the set of edges $\{j,k\}$ where $j < k$, and for the latter let the blocks A_k be singletons. We may think of 'exposing' the random graph step-by-step: at step k we expose which edges in the set A_k are present.

Lemma 3.3. *Suppose that the graph function f satisfies $|f(G) - f(G')| \leq 1$ whenever G' can be obtained from G by changing edges incident with a single vertex. Then the corresponding random variable $Y = f(G_{n,p})$ satisfies*

$$\mathbf{Pr}(Y - \mathbf{E}(Y) \geq t) \leq e^{-2t^2/n} \quad \textit{for } t \geq 0.$$

When we consider the chromatic number $\chi(G)$ and let $Y = \chi(G_{n,p})$ (and use the two-sided version of the last lemma), we find that

$$\mathbf{Pr}(|Y - \mathbf{E}(Y)| \geq t) \leq 2e^{-2t^2/n}, \tag{3.6}$$

which is (a slight sharpening of) the early result of Shamir and Spencer [60] which was important in introducing martingale methods into this area.

Lemma 3.4. *Suppose that the graph function f satisfies $|f(G) - f(G')| \le 1$ whenever G and G' differ in only one edge. Then the corresponding random variable $Y = f(G_{n,p})$ satisfies*

$$\Pr(Y - E(Y) \ge t) \le e^{-4t^2/n^2} \quad \textit{for } t \ge 0.$$

Perhaps the most exciting application of the bounded differences method uses this lemma. It is the proof by Bollobás [17] of what was a longstanding conjecture about the chromatic number $\chi(G_{n,p})$ of random graphs. Consider a constant edge probability p with $0 < p < 1$ and let $q = 1 - p$. Then for any $\epsilon > 0$,

$$\Pr\left((1-\epsilon)\frac{n}{2\log_b n} \le \chi(G_{n,p}) \le (1+\epsilon)\frac{n}{2\log_b n}\right) \to 1 \text{ as } n \to \infty.$$

(For a more precise result see [86].)

The lower bound part of the proof is easy: the interest is in establishing the upper bound for $\chi(G_{n,p})$. The key step in the proof is to show that the probability $p(n)$ that $G_{n,p}$ fails to contain a stable (independent) set with $s(n) = \lceil(2-\epsilon)\log_b n\rceil$ vertices is very small, say

$$p(n) = O(e^{-n^{4/3}}). \tag{3.7}$$

To see how this will yield the upper bound on $\chi(G_{n,p})$, let $\tilde{n} = \lceil n/\log^2 n\rceil$ and call a set W of at least $\tilde{n}$ vertices in $G_{n,p}$ bad if it contains no stable set of size at least $s(\tilde{n})$. The probability that there is a bad set is at most $2^n p(\tilde{n}) = o(1)$. But if there is no bad set W, then we can repeatedly colour a stable set of size at least $s(\tilde{n})$ and delete it, until there remain fewer than $\tilde{n}$ vertices, which may each get a new colour. The total number of colours used by this procedure is then at most

$$n/s(\tilde{n}) + \tilde{n} = \left(\frac{1}{2-\epsilon} + o(1)\right) n/\log_b n.$$

Thus we wish to see that (3.7) is true. The clever idea is to consider not just big stable sets but packings of such sets. Given a graph G on n vertices define $f(G)$ to be the maximum number of stable sets of size $s(n)$ which pairwise contain at most one common vertex. If graphs G and G' differ in only one edge then $f(G)$ and $f(G')$ differ by at most 1. Let $X_n = f(G_{n,p})$. It is not hard to check that $\mu_n = E(X_n)$ is large, say at least $n^{5/3}$ for n sufficiently large. Hence by (the other one-sided version of) Lemma 3.4, the probability $p(n)$ that $G_{n,p}$ has no stable set of size $s(n)$ equals

$$\Pr(X_n = 0) = \Pr(X_n - \mu_n \le -\mu_n) \le e^{-4\mu_n^2/n^2} \le e^{-4n^{4/3}}$$

for n sufficiently large.

3.1.3 Hamming Distances and Isoperimetric Inequalities. Next let us consider an application of the independent bounded differences inequality Theorem 3.1 involving Hamming distance in product spaces, and corresponding isoperimetric inequalities. This application will link in with our discussion later on Talagrand's inequality and on the use of ideas from information theory to prove concentration results.

Let $\Omega_1, \ldots, \Omega_n$ be probability spaces, and let Ω denote the product space $\prod \Omega_k$. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables with X_k taking values in Ω_k. Recall that for points $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ in Ω, the *Hamming distance* $d_H(\mathbf{x}, \mathbf{y})$ is the number of indices i such that $x_i \ne y_i$. We can use the independent bounded differences inequality to show that for any subset A of Ω such that $\Pr(\mathbf{X} \in A)$ is not too small, the probability that a random point $\mathbf{X}$ is 'close' to A is near 1. Recall that the Hamming distance from a point $\mathbf{x}$ to a set A is defined by setting $d_H(\mathbf{x}, A)$ to be $\inf\{d_H(\mathbf{x}, \mathbf{y}) : \mathbf{y} \in A\}$.

Theorem 3.5. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables and let A be a subset of the product space. Then, for any $t \ge 0$,*

$$\Pr(\mathbf{X} \in A)\Pr(d_H(\mathbf{X}, A) \ge t) \le e^{-t^2/2n}. \tag{3.8}$$

Let us rephrase this result before we prove it. Define the *t-fattening* of a subset A of Ω to be the set of points $\mathbf{x} \in \Omega$ such that $d_H(\mathbf{x}, A) < t$, and let the *measure* $\nu(A)$ be $\Pr(\mathbf{X} \in A)$. Then (3.8) says that

$$\nu(A)(1 - \nu(A_t)) \le e^{-t^2/2n}.$$

Thus if $\nu(A) \ge \frac{1}{2}$ then $\nu(A_t) \ge 1 - 2e^{-t^2/2n}$. In particular, when each random variable X_k is uniformly distributed on the set $\Omega_k = \{0,1\}$ we obtain an isoperimetric inequality for the n-cube – see for example [37, 45, 38].

Proof of Theorem 3.5. Let $p = \Pr(\mathbf{X} \in A)$ and let $\mu = E(d_H(\mathbf{X}, A))$. We may assume that $p > 0$. By the independent bounded differences inequality for $t \ge 0$

$$\Pr(d_H(\mathbf{X}, A) - \mu \ge t) \le e^{-2t^2/n}, \tag{3.9}$$

and

$$\Pr(d_H(\mathbf{X}, A) - \mu \le -t) \le e^{-2t^2/n}. \tag{3.10}$$

Now $d_H(\mathbf{x}, A) = 0$ if and only if $\mathbf{x} \in A$, so if we take $t = \mu$ in the inequality (3.10) above, we obtain

$$p = \Pr(\mathbf{X} \in A) = \Pr(d_H(\mathbf{X}, A) - \mu \le -\mu) \le e^{-2\mu^2/n}.$$

and so

$$\mu \le (\tfrac{1}{2}n\ln(1/p))^{\frac{1}{2}} = t_0, \text{ say.}$$

Now use the bound in the inequality (3.9) above to find

$$\Pr(d_H(\mathbf{X},A) \ge t + t_0) \le e^{-2t^2/n}.$$

Thus for $t \ge t_0$ we have

$$\Pr(d_H(\mathbf{X},A) \ge t) \le e^{-2(t-t_0)^2/n}. \qquad (3.11)$$

Now $(t-t_0)^2 \ge t^2/4$ for $t \ge 2t_0$, so if we take $t \ge 2t_0$ in the inequality (3.11) we obtain

$$\Pr(d_H(\mathbf{X},A) \ge t) \le e^{-t^2/2n}.$$

But for $0 \le t \le 2t_0$, the right hand side above is at least $e^{-2t_0^2/n} = p = \Pr(A)$. Thus

$$\min\{\Pr(\mathbf{X} \in A), \Pr(d_H(\mathbf{X},A) \ge t)\} \le e^{-t^2/2n}$$

for any $t \ge 0$. ∎

We may generalise the above discussion. Let $\alpha = (\alpha_1, \ldots, \alpha_n) \ge 0$ be an n-vector of non-negative real numbers. Recall that the (L_2) norm is given by

$$\|\alpha\| = (\sum_i \alpha_i^2)^{\frac{1}{2}}$$

and we call α a unit vector if it has norm $\|\alpha\| = 1$. For points $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ in Ω, the α-*Hamming distance* $d_\alpha(\mathbf{x},\mathbf{y})$ is the sum of the values α_i over those indices i such that $x_i \ne y_i$. Thus when α is the all 1's vector, it has norm $\sqrt{n}$ and α-Hamming distance is just the same as Hamming distance. Also, for a subset A of Ω, we define

$$d_\alpha(\mathbf{x},A) = \inf\{d_\alpha(\mathbf{x},\mathbf{y}) : \mathbf{y} \in A\}.$$

Exactly the same proof as for Theorem 3.5 yields the following extension of it.

Theorem 3.6. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables, let α be a non-negative unit n-vector, and let A be a subset of the product space. Then for any $t \ge 0$,*

$$\Pr(\mathbf{X} \in A)\Pr(d_\alpha(\mathbf{X},A) \ge t) \le e^{-t^2/2}.$$

Similar results appear in [56, 58, 60]. The central result of Section 4, namely Talagrand's inequality Theorem 4.1, looks rather similar to Theorem 3.6 but is in fact far more powerful, since it refers not just to one unit vector α but simultaneously to all such vectors.

The above result will give us a result like Theorem 3.1 but now around a median rather than the mean. Let us see how to do this. Consider a function f defined on $\prod A_k$ as there, and let $\mathbf{c}$ be the vector $(c_1, \ldots, c_n)$. Then the bounded differences condition (3.1), that $|f(\mathbf{x}) - f(\mathbf{x}')| \le c_k$ whenever the vectors $\mathbf{x}$ and $\mathbf{x}'$ differ only in the kth coordinate, is equivalent to the condition that $|f(\mathbf{x}) - f(\mathbf{x}')| \le d_{\mathbf{c}}(\mathbf{x},\mathbf{x}')$. Now assume that the condition (3.1) holds. Let

$$A_a = \{\mathbf{y} \in \prod A_k : f(\mathbf{y}) \le a\}.$$

Consider an $\mathbf{x} \in \prod A_k$. For each $\mathbf{y} \in A_a$,

$$f(\mathbf{x}) \le f(\mathbf{y}) + d_{\mathbf{c}}(\mathbf{x},\mathbf{y}) \le a + d_{\mathbf{c}}(\mathbf{x},\mathbf{y}),$$

and so, minimising over such $\mathbf{y}$,

$$f(\mathbf{x}) \le a + d_{\mathbf{c}}(\mathbf{x},A_a).$$

Let $c = \|\mathbf{c}\|$, and let α be the unit vector $\mathbf{c}/c$ along $\mathbf{c}$. If $f(\mathbf{x}) \ge a + t$ then

$$d_\alpha(\mathbf{x},A_a) = d_{\mathbf{c}}(\mathbf{x},A_a)/c \ge (f(\mathbf{x}) - a)/c \ge t/c.$$

Hence by Theorem 3.6, for any $t \ge 0$,

$$\begin{aligned}\Pr(f(\mathbf{X}) \le a)\Pr(f(\mathbf{X}) \ge a + t) &\le \Pr(\mathbf{X} \in A_a)\Pr(d_\alpha(\mathbf{X},A_a) \ge t/c) \\ &\le e^{-t^2/2c^2}.\end{aligned}$$

Now let m be a median of $f(\mathbf{X})$, that is $\Pr(f(\mathbf{X}) \le m) \ge \frac{1}{2}$ and $\Pr(f(\mathbf{X}) \ge m) \ge \frac{1}{2}$. Taking $a = m$ above gives

$$\Pr(f(\mathbf{X}) \ge m + t) \le 2e^{-t^2/2c^2}, \qquad (3.12)$$

and taking $a = m - t$ we have

$$\Pr(f(\mathbf{X}) \le m - t) \le 2e^{-t^2/2c^2}. \qquad (3.13)$$

The above two inequalities are like the conclusion of Theorem 3.1, at least if we are not too bothered about constants. They refer to concentration about the median m rather than the mean $\mu = \mathbf{E}[f(\mathbf{X})]$, but this makes little difference since the concentration inequalities themselves imply that $|\mu - m|$

is small. Indeed, the inequalities (3.29) and (3.31) together with Lemma 4.6 in subsection 4.3 below show that

$$|\mu - \nu| \le \sqrt{2\pi}\,c. \qquad (3.34)$$

So it is not important whether we refer to median or mean, and Theorem 3.6 and Theorem 4.1 are quite similar.

3.2 Extensions

In this subsection we refine the independent bounded differences inequality, Theorem 3.1, and the Bernstein inequality, Theorem 2.7, to obtain more widely applicable results, namely Theorems 3.7, 3.8 and 3.9, but at the cost of some added complication. We shall deduce these theorems quickly as immediate consequences of martingale theorems (though they do not themselves mention martingales). Theorems akin to these have recently proved useful when the random variables X_k correspond to answering questions such as whether two given vertices are adjacent in a random graph, and the question asked at time k may depend on the answers to previous questions – see for example [32, 5, 26]. We shall give part of an argument from [5] concerning hypergraph matchings at the end of this subsection.

Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of random variables with X_k taking values in a set A_k, and let f be a real-valued function defined on $\prod A_k$. Typically the random variables X_k will be independent but we shall not assume this here. We define quantities which measure the variability of the random variable $f(\mathbf{X})$ when the random variables $X_1, \ldots, X_{k-1}$ are fixed. These quantities correspond to deviation, range and variance. It is convenient to note first an easy bound on variance. If the random variable X satisfies $\mathbf{E}(X) = 0$ and $a \le X \le b$ then

$$\mathrm{var}(X) = \mathbf{E}(X^2) = \mathbf{E}(X(X-a)) \le \mathbf{E}(b(X-a)) = |ab| \le (b-a)^2/4. \quad (3.35)$$

Let $x_i \in A_i$ for each $i = 1, \ldots, k-1$, and let B denote the event that $X_i = x_i$ for each $i = 1, \ldots, k-1$. Let the random variable Y be distributed like X_k conditional on the event B (so if $k = 1$ then Y is distributed like X_1 with no conditioning, and if the random variables X_i are independent then for each k the random variable Y is distributed like X_k). For $x \in A_k$ let

$$g(x) = \mathbf{E}(f(\mathbf{X}) \mid B, X_k = x) - \mathbf{E}(f(\mathbf{X}) \mid B).$$

If the random variables X_i are independent then we may write $g(x)$ as

$$\mathbf{E}(f(x_1, \ldots, x_{k-1}, x, X_{k+1}, \ldots, X_n)) - \mathbf{E}(f(x_1, \ldots, x_{k-1}, X_k, X_{k+1}, \ldots, X_n)).$$

The function $g(x)$ measures how much the expected value of $f(\mathbf{X})$ changes if it is revealed that X_k takes the value x. Observe that $\mathbf{E}(g(Y)) = 0$.

Let $\mathrm{dev}^+(x_1, \ldots, x_{k-1})$ be $\sup\{g(x) : x \in A_k\}$, the positive deviation of $g(Y)$, and similarly let $\mathrm{dev}(x_1, \ldots, x_{k-1})$ be $\sup\{|g(x)| : x \in A_k\}$, the deviation of $g(Y)$. (If we denote $\mathbf{E}(f(\mathbf{X}))$ by μ then for each $\mathbf{x} = (x_1, \ldots, x_n) \in \prod A_k$ we have

$$|f(\mathbf{x}) - \mu| \le \sum \mathrm{dev}(x_1, \ldots, x_{k-1}). \qquad (3.36)$$

This inequality may be combined [illegible] with other inequalities like Theorem 3.1 – see [3, 18].) Let $\mathrm{ran}(x_1, \ldots, x_{k-1})$ denote $\sup\{|g(x) - g(y)| : x, y \in A_k\}$, the range of $g(Y)$. Also, denote the variance of $g(Y)$ by $\mathrm{var}(x_1, \ldots, x_{k-1})$.

For $\mathbf{x} \in \prod A_k$, let the sum of squared ranges be

$$R^2(\mathbf{x}) = \sum_{k=1}^{n} \mathrm{ran}(x_1, \ldots, x_{k-1})^2,$$

and let the *maximum sum of squared ranges* $\hat{r}^2$ be the supremum of the values $R^2(\mathbf{x})$ over all $\mathbf{x} \in \prod A_k$. Similarly let the *sum of variances* be

$$V(\mathbf{x}) = \sum_{k=1}^{n} \mathrm{var}(x_1, \ldots, x_{k-1}),$$

and let the *maximum sum of variances* $\hat{v}$ be the supremum of the values $V(\mathbf{x})$ over all $\mathbf{x} \in \prod A_k$. Observe that $V(\mathbf{x}) \le R^2(\mathbf{x})/4$ for each $\mathbf{x}$ by (3.35), and so $\hat{v} \le \hat{r}^2/4$. It is also of interest to note that

$$\mathrm{var}(f(\mathbf{X})) = \mathbf{E}(V(\mathbf{X})) \le \hat{v},$$

as is shown just before Theorem 3.14 below. Finally here, let maxdev^+ be the maximum of all the positive deviation values $\mathrm{dev}^+(x_1, \ldots, x_{k-1})$, over all choices of k and the x_i, and similarly let maxdev be the maximum of all the deviation values $\mathrm{dev}(x_1, \ldots, x_{k-1})$.

Example. Define the function $f : \{0,1\}^3 \to \{0,1\}$ by letting $f(\mathbf{x})$ be 0 on $(0,0,0)$, $(0,1,0)$, $(1,0,1)$ and be 1 otherwise. Let $\mathbf{X} = (X_1, X_2, X_3)$ be a family of independent random variables with $\Pr(X_k = 0) = \Pr(X_k = 1) = \frac{1}{2}$ for each k. Thus $\mathbf{E}(f(\mathbf{X})) = 5/8$, and $\mathrm{var}(f(\mathbf{X})) = 5/8 - (5/8)^2 = 15/64$.

At the 'root', $g(0) = \mathbf{E}(f(0, X_2, X_3)) - \mathbf{E}(f(\mathbf{X})) = 1/2 - 5/8 = -1/8$ and similarly $g(1) = 3/4 - 5/8 = 1/8$. Thus $\mathrm{ran}() = 1/4$, $\mathrm{dev}^+() = \mathrm{dev}() = 1/8$ and $\mathrm{var}() = 1/64$.

What happens if $X_1 = 1$? We have $\mathbf{E}(f(\mathbf{X}) \mid X_1 = 1) = \mathbf{E}(f(1, X_2, X_3)) = 5/4$, and so $g(0) = \mathbf{E}(f(1, 0, X_3)) - 5/4 = -1/4$ and $g(1) = \mathbf{E}(f(1, 1, X_3)) - 5/4 = 1/4$. Thus $\mathrm{ran}(1) = 1/2$, $\mathrm{dev}^+(1) = \mathrm{dev}(1) = 1/4$, and $\mathrm{var}(1) = 1/16$. Similarly $\mathrm{ran}(1, 0) = 1$ and $\mathrm{var}(1, 0) = 1/4$.

Now let $\mathbf{x} = (1, 0, 1)$ (or $(1, 0, 0)$). The corresponding sum of squared ranges $R^2(\mathbf{x})$ is $\mathrm{ran}()^2 + \mathrm{ran}(1)^2 + \mathrm{ran}(1, 0)^2 = 1/16 + 1/4 + 1 = 21/16$, which in fact equals $\hat{r}^2$. The corresponding sum of variances V (or v) is $\mathrm{var}() + \mathrm{var}(1) + \mathrm{var}(1, 0) = 15/64 + 1/64 + 1/4 = 1/2$, which in fact equals $\hat{v}$.

We are now ready to state the first of our more general results, which extends the independent bounded differences inequality, Theorem 3.1.

Theorem 3.7. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of random variables with X_k taking values in a set A_k, and let f be a bounded real-valued function defined on $\prod A_k$. Let μ denote the mean of $f(\mathbf{X})$, and let $\hat{r}^2$ denote the maximum sum of squared ranges. Then for any $t \geq 0$,*

$$\Pr(f(\mathbf{X}) - \mu \geq t) \leq e^{-2t^2/\hat{r}^2}.$$

More generally, let B be any 'bad' subset of $\prod A_k$, such that $R^2(\mathbf{x}) \leq r^2$ for each $\mathbf{x} \notin B$. Then

$$\Pr(f(\mathbf{X}) - \mu \geq t) \leq e^{-2t^2/r^2} + \Pr(\mathbf{X} \in B).$$

The first inequality above of course yields

$$\Pr(f(\mathbf{X}) - \mu \leq -t) \leq e^{-2t^2/\hat{r}^2}$$

by considering $-f$ (as in the comment after Theorem 3.1), and thus

$$\Pr(|f(\mathbf{X}) - \mu| \geq t) \leq 2e^{-2t^2/\hat{r}^2}. \qquad (3.17)$$

If for each $k = 1, \ldots, n$ we let r_k be the supremum over all choices of the x_i of the values $\mathrm{ran}(x_1, \ldots, x_{k-1})$ then of course $\hat{r}^2$ is at most $\sum r_k^2$. This bound for $\hat{r}^2$ yields Corollary 6.10 of [44]. Further, it yields also the independent bounded differences inequality, Theorem 3.1. For suppose that f satisfies the bounded differences condition (3.1) in that theorem. Let $1 \leq k \leq n$ and let $x_i \in A_i$ for $i = 1, \ldots, k-1$. We shall see that $\mathrm{ran}(x_1, \ldots, x_{k-1}) \leq c_k$, so $\hat{r}^2 \leq \sum r_k^2 \leq \sum c_k^2$, and then Theorem 3.1 follows. To see this, for each $x \in A_k$ let Z_x be the random variable $f(x_1, \ldots, x_{k-1}, x, X_{k+1}, \ldots, X_n)$. Then $|Z_x - Z_y| \leq c_k$. Hence, in the notation introduced before the statement of the last theorem, for any $x, y \in A_k$

$$|g(x) - g(y)| = |\mathbf{E}(Z_x) - \mathbf{E}(Z_y)| \leq \mathbf{E}(|Z_x - Z_y|) \leq c_k.$$

Thus $\mathrm{ran}(x_1, \ldots, x_{k-1}) \leq c_k$, as required.

Observe that the above argument will in fact yield a slightly stronger form of Theorem 3.1. Denote $\sum c_k^2$ by c^2. The theorem will still hold if we weaken the assumption on f to the condition that for each $\mathbf{x}$ there exist c_k (possibly depending on $\mathbf{x}$) such that $\sum c_k^2 \leq c^2$, and $|f(\mathbf{x}) - f(\mathbf{x}')| \leq c_k$ whenever the vectors $\mathbf{x}$ and $\mathbf{x}'$ differ only in the kth co-ordinate. The inequality of Talagrand that we shall meet later has a similar flavour.

Let us give one application of the above result, Theorem 3.7, before we go on to give extensions of the Bernstein theorem, Theorem 2.7. This application is from Maurey [54], and was, together with [?], one of the first uses of a martingale inequality outside probability theory.

Permutation graphs

Let S_n denote the set of all $n!$ permutations or linear orders on $\{1, \ldots, n\}$. The permutation graph G_n has vertex set S_n, and two vertices σ and τ are adjacent when $\sigma\tau^{-1}$ is a transposition, that is when τ can be obtained from σ by swapping the order of two elements. We are interested in isoperimetric inequalities for this graph. Given a set $A \subseteq S_n$ and $t > 0$, the t-fattening A_t of A consists of the vertices in G_n at graph distance less than t from some vertex in A. Thus, we want lower bounds on $|A_t|$ in terms of $|A|$, or upper bounds on $1 - |A_t|/n!$. We shall show that

$$(|A|/n!)(1 - |A_t|/n!) \leq e^{-t^2/8n}. \qquad (3.18)$$

Think of a linear order in S_n as an n-tuple $\mathbf{x} = (x_1, \ldots, x_n)$ where the x_i are distinct. Let $a_1, \ldots, a_k$ be distinct and let B be the set of linear orders $\mathbf{x} \in S_n$ such that $x_1 = a_1, \ldots, x_k = a_k$. For a distinct from the a_i let B_a be the set of $\mathbf{x} \in B$ with $x_{k+1} = a$. Let f be any function on S_n satisfying the Lipschitz or unit change condition $|f(\mathbf{x}) - f(\mathbf{y})| \leq 1$ if $\mathbf{x}$ and $\mathbf{y}$ are adjacent in G_n.

Now let $\mathbf{X}$ be uniformly distributed over S_n. In the notation introduced before the last theorem above, consider

$$g(a) = \mathbf{E}(f(\mathbf{X}) \mid \mathbf{X} \in B_a) - \mathbf{E}(f(\mathbf{X}) \mid \mathbf{X} \in B).$$

For any relevant distinct a and b, there is a bijection ϕ between B_a and B_b such that $\mathbf{x}$ and $\phi(\mathbf{x})$ are adjacent in G_n. (We simply swap the positions of a and b.) Thus $\mathbf{E}(f(\mathbf{X}) \mid \mathbf{X} \in B_b) = \mathbf{E}(f(\phi(\mathbf{X})) \mid \mathbf{X} \in B_a)$. It follows that

$$\begin{aligned} |g(a) - g(b)| &= |\mathbf{E}(f(\mathbf{X}) - f(\phi(\mathbf{X})) \mid \mathbf{X} \in B_a)| \\ &\leq \mathbf{E}(|f(\mathbf{X}) - f(\phi(\mathbf{X}))| \mid \mathbf{X} \in B_a) \leq 1. \end{aligned}$$

Hence by Theorem 3.7

$$\Pr(f(X) - E(f(X)) \geq t) \leq e^{-2t^2/n}.$$

Now let us specialise to the case when $f(x)$ is the graph distance between x and the set A. We may proceed exactly as in the proof of Theorem 3.5 above (after the first two inequalities) to show (3.16) as required. For related results and extensions see for example [60, 30, 45, 87, 88].

The next result extends the Bernstein theorem, Theorem 2.7.

Theorem 3.8. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of random variables with X_k taking values in a set A_k, and let f be a real-valued function defined on $\prod A_k$. Let μ denote the mean of $f(\mathbf{X})$. Let $b = \text{maxdev}^+$ and let $\hat{v}$ be the maximum sum of variances, both of which we assume to be finite. Then for any $t \geq 0$,*

$$\Pr(f(\mathbf{X}) - \mu \geq t) \leq e^{-\frac{t^2}{2\hat{v}(1+(bt/3\hat{v}))}}.$$

More generally, let B be any 'bad' subset of $\prod A_k$ such that $V(\mathbf{x}) \leq v$ for each $\mathbf{x} \notin B$. Then

$$\Pr(f(\mathbf{X}) - \mu \geq t) \leq e^{-\frac{t^2}{2v(1+(bt/3v))}} + \Pr(\mathbf{X} \in B).$$

As with Theorem 2.7 above, in typical applications of this result the 'error term' $bt/3v$ is negligible. Also, the 'bad set' B if present at all is such that $\Pr(\mathbf{X} \in B)$ is negligible. If we use the bounds $V(\mathbf{x}) \leq \hat{r}^2(\mathbf{x})/4$ for each $\mathbf{x}$ and $\hat{v} \leq \hat{r}^2/4$, we can easily obtain the bound in Theorem 3.7 for small t. If for each $k = 1, \ldots, n$ we let v_k be the maximum of the values $\text{var}_k(x_1, \ldots, x_{k-1})$ over all choices of the x_i, then $\hat{v}$ is at most $\sum v_k$. If we use this bound for $\hat{v}$ together with the discussion below, we obtain a result related to inequalities used by Kim [52] in his marvellous $R(3,t)$ paper. However, the present more general result is needed for certain applications – see for example [32, 3, 26] and the example below.

Observe that if a random variable X has mean 0 and takes only two values, with probabilities p and $1-p$, then the two values are $-pr$ and $(1-p)r$ where r is the range of X, and $\text{var}(X) = p(1-p)r^2 \leq pr^2$ – see also (3.15) above. Thus if p is small so is $\text{var}(X)$ and we can get tight bounds on deviations. Let us state one corollary of Theorem 3.8, which is a tightening of the martingale inequality in [2].

Theorem 3.9. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of random variables with X_k taking values in a set A_k, and let f be a bounded real-valued function defined on $\prod A_k$. Let μ denote the mean of $f(\mathbf{X})$, let b denote the maximum deviation maxdev, and let $\hat{r}^2$ denote the maximum sum of squared ranges. Suppose that for any given values taken by $X_1, \ldots, X_{k-1}$, the random variable X_k takes at most two values, and if it can take two values then the smaller of the probabilities is at most p, where $p \leq \frac{1}{2}$. Then for any $t \geq 0$,*

$$\Pr(f(\mathbf{X}) - \mu \geq t) \leq e^{-\frac{t^2}{2p\hat{r}^2(1+(bt/3p\hat{r}^2))}}.$$

As with Theorems 2.7 and 3.8 above, we hope to be able to ignore the 'error term' $bt/3p\hat{r}^2$. The important term in the bound is $e^{-\frac{t^2}{2p\hat{r}^2}}$, which is significantly better (smaller) than the corresponding term $e^{-\frac{2t^2}{\hat{r}^2}}$ from Theorem 3.7 when $p = o(1)$. In the next subsection we describe an application where this difference is critical.

3.2.1 An Application to Hypergraph Matchings. A *matching* in H is a set of pairwise disjoint edges. Let $k \geq 3$ be a fixed integer, and consider a k-uniform d-regular simple hypergraph H on n vertices. (Thus each edge contains exactly k vertices, each vertex is contained in exactly d edges, and each pair of distinct edges meet in at most one vertex.) It is shown in [2] that such a hypergraph H contains a matching covering all but a vanishing proportion of the vertices as $n \to \infty$. (Earlier results showed that the proportion of vertices that could not be covered tended to zero, but perhaps slowly.)

The idea of the proof is to find such a matching by repeatedly taking random 'bites' (like large 'Rödl nibbles' – see for example [3]). We take each bite as follows. Form a set X of edges by choosing the edges independently with probability ϵ/d. Call an edge 'isolated' if it meets no other edge in X. Let M consist of the isolated edges in X – these will form part of the final matching. Now delete from H all the vertices in the edges in M and all the edges meeting these vertices, forming a hypergraph H^* on the vertex set V^* and take the next bite from H^*. We must show that H^* is approximately regular of appropriately smaller degree. (Many details here have been omitted, in particular a neat degree stabilisation technique, but they do not affect the ideas that we wish to illustrate.) A key part of the proof is to check that each vertex degree in H^* is close to its expected value with high probability, and that is what we now proceed to do. (We need the probability of a significant deviation to be very small since the next step in the proof is to use the Lovász Local Lemma: when using a 'Rödl nibble' often a second moment bound suffices – see for example [3].)

For each vertex $v \in V$ let Z_v be the number of edges $E \in H$ containing v such that $E \setminus \{v\} \subseteq V^*$. Observe that if $v \in V^*$ then Z_v equals the degree of v in H^*. (By defining Z_v in this way we need not worry about whether or not the vertex v is in V^*.) It turns out that it suffices to consider a fixed vertex $v \in V$, and show that for $t = o(d^{\frac{1}{2}})$ we have

$$\Pr(|Z_v - \mathbf{E}(Z_v)| > \lambda d^{\frac{1}{2}}) \le e^{-c\lambda^2}.$$

(See Claim 2 in [8].) Let us see how we can obtain this result from Theorem 3.8. Recall that Theorem 3.9 gives a bound of roughly $e^{-\frac{t^2}{2V}}$ as long as the deviation t is not too large.

For each edge $E \in H$ let the random variable $X_E = 1$ if E appears in the random set X and let $X_E = 0$ if not. Thus $\Pr(X_E = 1) = p = 1/d$ and we shall be in business as long as the maximum sum of squared ranges $\hat{r}^2 = \max_x R^2(x)$ is $O(d^2)$. (In order to use Theorem 3.7 we could tolerate only $\hat{r}^2 = O(d)$, which is not the case.)

Call an edge in H primary if it contains the vertex v, secondary if it is not primary but meets a primary edge, and tertiary if it is not primary or secondary but meets a secondary edge. Let $\mathcal{E}_1$, $\mathcal{E}_2$ and $\mathcal{E}_3$ denote the sets of primary, secondary and tertiary edges respectively, and note that $|\mathcal{E}_1| = d$, $|\mathcal{E}_2| \le (k-1)d^2$ and $|\mathcal{E}_3| \le (k-1)^2 d^3$. Let $\mathcal{E}$ be the union of the sets $\mathcal{E}_i$.

The random variable Z_v is determined by the values of the random variables X_E for $E \in \mathcal{E}$. Let Ω be the set of binary vectors x indexed by $\mathcal{E}$. For each $x \in \Omega$ let $f(x)$ be the corresponding value of the degree Z_v. Let $x, y \in \Omega$ differ only in coordinate F, where $F \in \mathcal{E}$. If $F \in \mathcal{E}_1$ then $|f(x) - f(y)| \le 1$. If $F \in \mathcal{E}_2$ then $|f(x) - f(y)| \le k$. So far the contribution to the term $R^2(x)$ is at most

$$|\mathcal{E}_1| + |\mathcal{E}_2| k^2 \le k^3 d^2 = O(d^2),$$

which as we saw above is small enough. Similarly, if $F \in \mathcal{E}_3$ then $|f(x) - f(y)| \le d$. However, we cannot tolerate a contribution to $R^2(x)$ of order d^3, so we must do better.

Let $x \in \Omega$. Call an edge $F \in \mathcal{E}_2$ important if $x_F = 1$ and F meets no other edges $F' \in \mathcal{E}_2$ with $x_{F'} = 1$. There are at most $(k-1)d$ important edges, and so at most $k^2 d^2$ tertiary edges that meet an important edge. Further, if $y \in \Omega$ differs from x only in coordinate F for some tertiary edge F which meets no important edge then $f(x) = f(y)$. Thus we can bound $R^2(x)$ by $k^3 d^2 + (k^2 d^2) k^4 \le 2k^6 d^2$, and so the maximum sum of squared ranges $\hat{r}^2 \le 2k^6 d^2$. Since each $\Pr(X_F = 1) = 1/d$ we may now use Theorem 3.8 to show that

$$\begin{aligned}\Pr(|Z_v - \mathbf{E}(Z_v)| > td^{\frac{1}{2}}) &\le 2\exp\left(-\frac{t^2 d}{2(2k^6 d)(1 + ((td^{\frac{1}{2}})/(6k^6 d)))}\right)\\ &= 2\exp\left(-\frac{t^2}{4k^6(1 + t/(6k^6 d^{\frac{1}{2}}))}\right),\end{aligned}$$

and this bound is at most $e^{-\lambda^2}$ for $t = 3k^3 \lambda$.

3.3 Martingales

We give here a brief introduction to the theory of martingales, focussing on the case when the underlying probability space is finite. For much fuller introductions see for example [49] or [73].

The starting point is a probability space $(\Omega, \mathcal{F}, \Pr)$. Thus Ω is the non-empty set of all 'elementary outcomes', $\mathcal{F}$ is the set of 'events', and $\Pr$ is the probability measure. The collection $\mathcal{F}$ of events must be suitably closed under unions, intersections and complements, and is assumed to be a σ-field. A σ-field on Ω is a collection $\mathcal{G}$ of subsets of Ω which contains the empty set, and is closed under complementation (if $A \in \mathcal{G}$ then $\Omega \setminus A \in \mathcal{G}$) and under countable unions (if $A_1, A_2, \ldots \in \mathcal{G}$ then their union is in $\mathcal{G}$). It follows that such a collection $\mathcal{G}$ is also closed under countable intersections. In many applications the underlying set Ω is finite, and the σ-field $\mathcal{F}$ of events is the collection of all subsets of Ω. Let us assume in the meantime that Ω is finite, though what we say is either true in general or at least tells the right story.

Corresponding to any σ-field $\mathcal{G}$ on Ω there is a partition of Ω into non-empty sets, the blocks of the partition, such that the σ-field $\mathcal{G}$ is the collection of all sets which are unions of blocks. Corresponding to the σ-field of all subsets of Ω is the partition of Ω into singleton blocks. Suppose that we have a σ-field $\mathcal{G}$ contained in $\mathcal{F}$. Any function on Ω which is constant on the blocks of $\mathcal{G}$ is called $\mathcal{G}$-measurable. A random variable is an $\mathcal{F}$-measurable real-valued function X defined on Ω: so that in the case when $\mathcal{F}$ consists of all subsets of Ω any real-valued function defined on Ω is a random variable.

The expectation of X conditional on $\mathcal{G}$, $\mathbf{E}(X \mid \mathcal{G})$, is the $\mathcal{G}$-measurable function whose (constant) value on each block of $\mathcal{G}$ is the average value of X on the block. This is a very important notion. We may see that $\mathbf{E}(X \mid \mathcal{F}) = X$ (that is, $\mathbf{E}(X \mid \mathcal{F})(\omega) = X(\omega)$ for each $\omega \in \Omega$), and if $\mathcal{G}$ is the trivial σ-field $\{\emptyset, \Omega\}$ corresponding to the trivial partition of Ω into one block, then $\mathbf{E}(X \mid \mathcal{G})$ is the constant function with constant value $\mathbf{E}(X)$. Key properties of conditional expectations that we shall need are that if $\mathcal{G}_1 \subseteq \mathcal{G}_2$ then

$$\mathbf{E}(\mathbf{E}(X \mid \mathcal{G}_2) \mid \mathcal{G}_1) = \mathbf{E}(X \mid \mathcal{G}_1) \qquad (3.29)$$

and so in particular

$$\mathbf{E}(\mathbf{E}(X \mid \mathcal{G})) = \mathbf{E}(X), \qquad (3.30)$$

and

$$\mathbf{E}(XY \mid \mathcal{G}) = Y\mathbf{E}(X \mid \mathcal{G}) \text{ if } Y \text{ is } \mathcal{G}\text{-measurable}. \qquad (3.31)$$

The maximum of X in $\mathcal{G}$, $\sup(X \mid \mathcal{G})$, is the $\mathcal{G}$-measurable random variable which takes the value at ω equal to the maximum value of X over the block containing ω. Clearly

$$\mathbf{E}(X \mid \mathcal{G}) \le \sup(X \mid \mathcal{G}), \tag{3.22}$$

and if $\mathcal{G}_1 \subseteq \mathcal{G}_2$ then

$$\sup(X \mid \mathcal{G}_2) \le \sup(X \mid \mathcal{G}_1). \tag{3.23}$$

Note that each of the above results holds for each $\omega \in \Omega$. It is time for an example!

Example Let $\Omega = \{0,1\}^n$, let $\mathcal{F}$ be the collection of all subsets of Ω, let $0 < p < 1$ and for each $\omega = (\omega_1, \ldots, \omega_n)$ let $\mathbf{Pr}(\{\omega\}) = p^j(1-p)^{n-j}$ where $j = \sum \omega_i$. This defines our probability space. For each $k = 1, \ldots, n$ define $X_k(\omega) = \omega_k$ for each $\omega \in \Omega$. Then $X_1, \ldots, X_n$ are independent random variables with $\mathbf{Pr}(X_k = 1) = 1 - \mathbf{Pr}(X_k = 0) = p$ for each k. Also, let $S_k = X_1 + \cdots + X_k$. Let $\mathcal{F}_k$ be the σ-field corresponding to the partition of Ω into the 2^k blocks $\{\omega \in \Omega : \omega_1 = x_1, \ldots, \omega_k = x_k\}$ for each $(x_1, \ldots, x_k) \in \{0,1\}^k$. Then the random variable $\mathbf{E}(S_n \mid \mathcal{F}_k)$ satisfies (for each $\omega \in \Omega$)

$$\mathbf{E}(S_n \mid \mathcal{F}_k) = S_k + (n-k)p = \omega_1 + \cdots + \omega_k + (n-k)p,$$

and $\mathbf{E}(S_n \mid \mathcal{F}_n) = S_n$, $\mathbf{E}(S_n \mid \mathcal{F}_0) = \mathbf{E}(S_n) = np$ and $\mathbf{E}(\mathbf{E}(S_n \mid \mathcal{F}_k)) = \mathbf{E}(S_k) + (n-k)p = np$. Also for example

$$\mathbf{E}(S_k S_n \mid \mathcal{F}_k) = S_k \mathbf{E}(S_n \mid \mathcal{F}_k) = S_k^2 + (n-k)pS_k.$$

Further

$$\sup(S_n \mid \mathcal{F}_k) = S_k + (n-k) \le S_{k-1} + (n-k+1) = \sup(S_n \mid \mathcal{F}_{k-1}).$$

Another important idea is that of a filter. A nested sequence $\{\emptyset, \Omega\} = \mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots$ of σ-fields contained in $\mathcal{F}$ is called a *filter*. This corresponds (in the finite case) to a sequence of increasingly refined partitions of Ω, starting with the trivial partition into one block. We may think of the filter as corresponding to acquiring information as time goes on: at time k we know which block of the partition corresponding to $\mathcal{F}_k$ contains our random elementary outcome ω. Given a filter, a sequence $X_0, X_1, X_2, \ldots$ of random variables is called a *martingale* if $\mathbf{E}(X_{k+1} \mid \mathcal{F}_k) = X_k$ for each $k = 0, 1, \ldots$. This implies that X_k is $\mathcal{F}_k$-measurable (so at time k we know the value of X_k). It also implies that $\mathbf{E}(X_k) = \mathbf{E}(X_0)$ for each k. A sequence $Y_1, Y_2, \ldots$ of random variables is called a *martingale difference sequence* if Y_k is $\mathcal{F}_k$-measurable and $\mathbf{E}(Y_k \mid \mathcal{F}_{k-1}) = 0$ for each positive integer k.

From a martingale $X_0, X_1, X_2, \ldots$ we obtain a martingale difference sequence by setting $Y_k = X_k - X_{k-1}$, and conversely from X_0 and a martingale difference sequence we obtain a martingale $X_0, X_1, X_2, \ldots$ by setting $X_k = X_0 + \sum_{i=1}^k Y_i$. Thus we may focus on either form.

We shall be interested here only in finite filters $\{\emptyset, \Omega\} = \mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_n$ where $\mathcal{F}_n \subseteq \mathcal{F}$. Let X be a random variable and define $X_k = \mathbf{E}(X \mid \mathcal{F}_k)$ for $k = 0, 1, \ldots, n$. Then $X_0, X_1, \ldots, X_n$ is a martingale, with $X_0 = \mathbf{E}(X)$ and $X_n = X$ if X is $\mathcal{F}_n$-measurable. This is called Doob's martingale process, and for finite filters all corresponding martingales may be obtained in this way. If $Y_1, \ldots, Y_n$ is the corresponding martingale difference sequence then we have $X - \mathbf{E}(X) = \sum Y_k$.

Example (continued) There is a natural filter here, namely

$$\{\emptyset, \Omega\} = \mathcal{F}_0 \subset \mathcal{F}_1 \subset \cdots \subset \mathcal{F}_n = \mathcal{F}$$

which corresponds to learning the values of the coordinates of ω one by one. The σ-field $\mathcal{F}_k$ is the σ-field generated by the random variables $X_1, \ldots, X_k$, that is, the smallest σ-field $\mathcal{G}$ such that each of $X_1, \ldots, X_k$ is $\mathcal{G}$-measurable. For each $k = 1, \ldots, n$ let T_k be the random variable $S_k - kp = (X_1 - p) + \cdots + (X_k - p)$. Then $\mathbf{E}(T_k \mid \mathcal{F}_{k-1}) = T_{k-1}$, and so the random variables T_k form a martingale, with corresponding martingale difference sequence $X_k - p$.

When the underlying set Ω is infinite we need to be a little more careful. In particular, the results discussed above hold with probability 1 (also called 'almost surely') rather than for every $\omega \in \Omega$, and we need to assume that various expectations are finite. However, the brief introduction above should give the right ideas.

The most basic inequality for a bounded martingale difference sequence is the following lemma of Hoeffding (1963) [39] and Azuma (1967) [6], which we shall refer to as 'the Hoeffding-Azuma inequality'.

Theorem 3.10. *Let $c_1, \ldots, c_n$ be constants, and let $Y_1, \ldots, Y_n$ be a martingale difference sequence with $|Y_k| \le c_k$ for each k. Then for any $t \ge 0$,*

$$\mathbf{Pr}(|\sum Y_k| \ge t) \le 2e^{-t^2/2\sum c_k^2}.$$

Suppose that $X_1, \ldots, X_n$ are independent, with $\mathbf{Pr}(X_k = 1) = p$ and $\mathbf{Pr}(X_k = 0) = 1 - p$. Set $Y_k = X_k - p$ and $c_k = \max(p, 1-p)$. We may then apply the above lemma to obtain the Chernoff bound in Theorem 2.1 except that the bound is weaker if $p \neq \frac{1}{2}$. All our applications will be based on less symmetrical forms of the above result, and will thus avoid gratuitous factors less than 1 in the exponent in the bounds. In particular, Theorem 3.10 is a special case of Theorem 3.13 below.

3.4 Martingale Results

The results in this subsection extend all the earlier results. In particular, the next result extends Lemma 2.2 on independent random variables.

Lemma 3.11. *Let $Y_1, Y_2, \ldots, Y_n$ be a martingale difference sequence with $-a_k \le Y_k \le 1 - a_k$ for each k, for suitable constants a_k; and let $a = \frac{1}{n}\sum a_k$ and let $\bar{a} = 1 - a$. Then for any $0 \le t < \bar{a}$,*

$$\Pr(\sum Y_k \ge nt) \le \left(\left(\frac{a}{a+t}\right)^{a+t}\left(\frac{\bar{a}}{\bar{a}-t}\right)^{\bar{a}-t}\right)^n. \qquad (3.24)$$

Proof. Since $S_n = S_{n-1} + Y_n$ and S_{n-1} is $\mathcal{F}_{n-1}$-measurable (and hence so is $e^{hS_{n-1}}$), we may use (3.20) and (3.21) to show that for any h

$$\mathbf{E}(e^{hS_n}) = \mathbf{E}(e^{hS_{n-1}}e^{hY_n}) = \mathbf{E}(e^{hS_{n-1}}\mathbf{E}(e^{hY_n} \mid \mathcal{F}_{n-1})).$$

Thus as in the proof of Lemma 2.2, for any $h > 0$,

$$\begin{aligned}\mathbf{E}(e^{hS_n}) &= \mathbf{E}(e^{hS_{n-1}}\mathbf{E}(e^{hY_n} \mid \mathcal{F}_{n-1}))\\ &\le \mathbf{E}(e^{hS_{n-1}})\left((1-a_n)e^{-ha_n} + a_n e^{h(1-a_n)}\right)\\ &\le \prod_k\left((1-a_k)e^{-ha_k} + a_k e^{h(1-a_k)}\right)\end{aligned}$$

on iterating, and we may complete the proof exactly as for Lemma 2.2. □

We may deduce many useful inequalities from this lemma, just as we obtained Theorem 2.3 from Lemma 2.2.

Theorem 3.12. *Let $Y_1, Y_2, \ldots, Y_n$ be a martingale difference sequence with $-a_k \le Y_k \le 1 - a_k$ for each k, for suitable constants a_k; and let $a = \frac{1}{n}\sum a_k$.*

(a) For any $t \ge 0$,

$$\Pr(|\sum Y_k| \ge t) \le 2e^{-2t^2/n}.$$

(b) For any $\varepsilon > 0$,

$$\Pr(\sum Y_k \ge \varepsilon an) \le e^{-((1+\varepsilon)\ln(1+\varepsilon)-\varepsilon)an} \le e^{-\frac{\varepsilon^2 an}{2(1+\varepsilon/3)}}.$$

(c) For any $\varepsilon > 0$,

$$\Pr(\sum Y_k \le -\varepsilon an) \le e^{-\frac{1}{2}\varepsilon^2 an}.$$

To deduce Theorem 2.3 from Theorem 3.12, let $a_k = \mathbf{E}(X_k)$ and $Y_k = X_k - a_k$, so that $-a_k \le Y_k \le 1 - a_k$; then $\mu = \sum a_k = na$, $p = a$ and $\sum Y_k = S_n - \mu$. The next result extends both the independent bounded differences inequality, Theorem 3.1, and the Hoeffding-Azuma inequality, Theorem 3.10.

Theorem 3.13. *Let $Y_1, \ldots, Y_n$ be a martingale difference sequence with $a_k \le Y_k \le b_k$ for each k, for suitable constants a_k, b_k. Then for any $t \ge 0$,*

$$\Pr(|\sum Y_k| \ge t) \le 2e^{-2t^2/\sum(b_k-a_k)^2}. \qquad (3.25)$$

The next pair of results, Theorems 3.14 and 3.15, are the most powerful of the martingale results we present, and include all the previous theorems (except for the first inequality in part (b) of Theorem 2.3 and of Theorem 3.12). In particular, Theorem 3.13 will follow immediately from Theorem 3.14. In order to state the two results we need some more definitions and notation. We postpone their proofs to the next subsection.

Let X be a bounded random variable and let $\mathcal{G}$ be a σ-field contained in the σ-field $\mathcal{F}$ of all events. The *conditional range of X in $\mathcal{G}$*, $\mathrm{ran}(X \mid \mathcal{G})$ is the $\mathcal{G}$-measurable function $\sup(X \mid \mathcal{G}) + \sup(-X \mid \mathcal{G})$. The *conditional variance of X in $\mathcal{G}$*, $\mathrm{var}(X \mid \mathcal{G})$ is $\mathbf{E}((X - Y)^2 \mid \mathcal{G})$, where $Y = \mathbf{E}(X \mid \mathcal{G})$. In the example in the last subsection, the conditional range of S_n in $\mathcal{F}_k$, $\mathrm{ran}(S_n \mid \mathcal{F}_k)$ is the constant function $n - k$, and the conditional variance $\mathrm{var}(S_n \mid \mathcal{F}_k)$ is the constant function $(n-k)p(1-p)$.

Now let $\{\emptyset, \Omega\} = \mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_n$ be a filter in $\mathcal{F}$. Let the bounded random variable X be $\mathcal{F}_n$-measurable, and let $X_0, X_1, \ldots, X_n$ be the martingale obtained by setting $X_k = \mathbf{E}(X \mid \mathcal{F}_k)$. Further let $Y_1, \ldots, Y_n$ be the corresponding martingale difference sequence obtained by setting $Y_k = X_k - X_{k-1}$. For $1 \le k \le n$, we define four $\mathcal{F}_{k-1}$-measurable functions ran_k, dev_k^+, dev_k and var_k as follows. We let ran_k denote $\mathrm{ran}(Y_k \mid \mathcal{F}_{k-1})$ ($= \mathrm{ran}(X_k \mid \mathcal{F}_{k-1})$), let dev_k^+ denote $\sup(Y_k \mid \mathcal{F}_{k-1})$, let dev_k denote $\sup(|Y_k| \mid \mathcal{F}_{k-1})$, and finally we let var_k denote $\mathrm{var}(Y_k \mid \mathcal{F}_{k-1})$ ($= \mathrm{var}(X_k \mid \mathcal{F}_{k-1})$). Note that $\mathrm{dev}_k^+ \le \mathrm{dev}_k \le \mathrm{ran}_k \le 2\mathrm{dev}_k$, and $\mathrm{var}_k \le (1/4)\mathrm{ran}_k^2$ by (3.13).

Finally we define two random variables R^2 and V and four constants $\hat{r}^2$, $\hat{v}$, b and maxdev^+ as follows. Let the *sum of squared conditional ranges* R^2 be the random variable $\sum \mathrm{ran}_k^2$, and let the *maximum sum of squared conditional ranges* $\hat{r}^2$ be the (essential) supremum of the random variable R^2. Let the *sum of conditional variances* V be the random variable $\sum \mathrm{var}_k$, and let the *maximum sum of conditional variances* $\hat{v}$ be the supremum of the random variable V. Finally let the *maximum conditional positive deviation* maxdev^+ be the supremum over all k of the random variables dev_k^+, and let the *maximum conditional deviation* b be the supremum over all k of the random variables dev_k.

The random variable V is also called the 'predictable quadratic variation' of the martingale (X_k) – see for example [61], or the 'increasing sequence' associated with (X_k), see for example [23]. Note that

$$\begin{aligned}\mathbf{E}(V) &= \mathbf{E}\left(\sum_{k=1}^{n}\mathbf{E}((X_k - X_{k-1})^2 \mid \mathcal{F}_{k-1})\right)\\ &= \mathbf{E}\left(\sum_{k=1}^{n}(\mathbf{E}(X_k^2 \mid \mathcal{F}_{k-1}) - X_{k-1}^2)\right)\\ &= \sum_{k=1}^{n}(\mathbf{E}(X_k^2) - \mathbf{E}(X_{k-1}^2))\\ &= \mathbf{E}(X_n^2) - \mathbf{E}(X_0^2) = \mathrm{var}(X).\end{aligned}$$

Theorem 3.14. *Let X be a bounded random variable with $\mathbf{E}(X) = \mu$, and let $\{\emptyset, \Omega\} = \mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_n$ be a filter in $\mathcal{F}$. Then for any $t \geq 0$,*

$$\mathbf{Pr}(X - \mu \geq t) \leq e^{-2t^2/\hat{r}^2}, \tag{3.26}$$

where $\hat{r}^2$ is the maximum sum of squared conditional ranges. More generally, for any $t \geq 0$ and any value r^2,

$$\mathbf{Pr}((X - \mu \geq t) \wedge (R^2 \leq r^2)) \leq e^{-2t^2/r^2}, \tag{3.27}$$

where the random variable R^2 is the sum of squared conditional ranges.

The earlier result Theorem 3.7 is essentially this result when the σ-field $\mathcal{F}_k$ in the filter is the σ-field generated by $X_1, \ldots, X_k$. Suppose that for each $k = 1, \ldots, n$, we let $\hat{r}_k$ be the supremum of the values $\mathrm{ran}(x_1, \ldots, x_{k-1})$ over all choices of the x_i. (This corresponds to our earlier use of the notation $\hat{r}_k$ immediately after Theorem 3.7.) Then $\hat{r}^2$ is at most $\sum \hat{r}_k^2$. If we use this bound for $\hat{r}^2$ in Theorem 3.14 above we obtain Theorem 6.7 of [45], which extends Theorem 3.10 above. The next result extends the earlier results that use bounds on the variance, namely Theorem 2.7 and Theorem 3.8 (and thus Theorem 3.9), and is close to Theorem 4.1 in [87] – see also [10, 2, 36].

Theorem 3.15. *Let X be a random variable with $\mathbf{E}(X) = \mu$, and let $\{\emptyset, \Omega\} = \mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_n$ be a filter in $\mathcal{F}$. Let $b = \max\mathrm{dev}^+$, the maximum conditional positive deviation (and assume that b is finite). Then for any $t \geq 0$*

$$\mathbf{Pr}(X - \mu \geq t) \leq e^{-\frac{t^2}{2\hat{v}(1+(bt/3\hat{v}))}}, \tag{3.28}$$

where $\hat{v}$ is the maximum sum of conditional variances (which is assumed to be finite). More generally, for any $t \geq 0$ and any value $v \geq 0$,

$$\mathbf{Pr}((X - \mu \geq t) \wedge (V \leq v)) \leq e^{-\frac{t^2}{2v(1+(bt/3v))}}, \tag{3.29}$$

where the random variable V is the sum of conditional variances.

As with the earlier results of this form, we think of the term $(bt/3v)$ as a negligible error term. To complete the proofs of all the results given above it suffices to prove the last two results. We do this in the next subsection.

3.5 Remaining Proofs for Martingale Results

The following lemma is partly based on Lemma 3.4 of Kahn [55]. The lemma itself (in a special case) is used (rather than one of the theorems derived from it) in the proofs in [48] concerning the concentration of the number of comparisons used by quicksort. We shall always take $\mathcal{F}_0$ as the trivial σ-field $\{\emptyset, \Omega\}$ when we use the lemma, but we allow any $\mathcal{F}_0$ to give a neater induction.

Lemma 3.16. *Let $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots \subseteq \mathcal{F}_n$ be a filter in $\mathcal{F}$, and let $Y_1, \ldots, Y_n$ be a corresponding martingale difference sequence, where each Y_k is bounded above. Let the random variable Z be the indicator of some event. Then for any h,*

$$\mathbf{E}(Ze^{h\sum Y_k} \mid \mathcal{F}_0) \leq \sup\left(Z\prod \mathbf{E}(e^{hY_k} \mid \mathcal{F}_{k-1}) \mid \mathcal{F}_0\right).$$

Proof. We use induction on n. The case $n = 0$ is trivial since it says that $\mathbf{E}(Z \mid \mathcal{F}_0) \leq \sup(Z \mid \mathcal{F}_0)$ as in (3.23). Now let $n \geq 1$ and suppose that the result holds for $n - 1$. Let

$$A = Ze^{h\sum_{k=2}^{n} Y_k}$$

and

$$B = Z\prod_{k=2}^{n}\mathbf{E}(e^{hY_k} \mid \mathcal{F}_{k-1}).$$

Then by the induction hypothesis, $\mathbf{E}(A \mid \mathcal{F}_1) \leq \sup(B \mid \mathcal{F}_1)$, and $\sup(B \mid \mathcal{F}_1) \leq \sup(B \mid \mathcal{F}_0)$ as in (3.22). Hence

$$\begin{aligned}\mathbf{E}(Ze^{h\sum_{k=1}^{n} Y_k} \mid \mathcal{F}_0) &= \mathbf{E}(e^{hY_1}\mathbf{E}(A \mid \mathcal{F}_1) \mid \mathcal{F}_0)\\ &\leq \mathbf{E}(e^{hY_1}\sup(B \mid \mathcal{F}_1) \mid \mathcal{F}_0)\\ &\leq \sup(B \mid \mathcal{F}_0)\mathbf{E}(e^{hY_1} \mid \mathcal{F}_0) \quad \text{as in (3.23)}\\ &= \sup\left(Z\prod_{k=1}^{n}\mathbf{E}(e^{hY_k} \mid \mathcal{F}_{k-1}) \mid \mathcal{F}_0\right),\end{aligned}$$

which completes the induction step. □

Proof of Theorem 3.14. Let $Y_1, \dots, Y_n$ be the corresponding martingale difference sequence. Let the random variable Z be the indicator of the event that $R^2 \le r^2$, so that $0 \le ZR^2 \le r^2$. For any h, by Lemma 2.6,

$$\mathbf{E}(e^{hY_k} \mid \mathcal{F}_{k-1}) \le e^{h^2 \mathrm{ran}_k^2/8}.$$

Hence by Lemma 3.16,

$$\begin{aligned}\mathbf{E}(Ze^{h(X-\mu)}) &\le \sup\left(Z\prod_k e^{h^2\mathrm{ran}_k^2/8}\right)\\ &= \sup(Ze^{h^2R^2/8})\\ &\le e^{h^2\sup(ZR^2)/8}\\ &\le e^{h^2r^2/8}.\end{aligned}$$

Thus for any $h > 0$, by Markov's inequality,

$$\begin{aligned}\Pr((X-\mu \ge t) \cap (R^2 \le r^2)) &= \Pr(Ze^{h(X-\mu)} \ge e^{ht})\\ &\le e^{-ht}\mathbf{E}(Ze^{h(X-\mu)})\\ &\le e^{-ht+h^2r^2/8}\\ &= e^{-2t^2/r^2}\end{aligned}$$

when $h = 4t/r^2$. □

Proof of Theorem 3.15. Let $Y_1, \dots, Y_n$ be the corresponding martingale difference sequence. Note that $Y_k \le b$ for each k. Let the random variable Z be the indicator of the event that $V \le v$, so that $0 \le ZV \le v$. Now as in the proof of Theorem 2.7 we use Lemma 2.8, and use those definitions there. We find that, for any $h > 0$,

$$\mathbf{E}(e^{hY_k} \mid \mathcal{F}_{k-1}) \le e^{h^2 g(hb)\,\mathrm{var}(Y_k \mid \mathcal{F}_{k-1})} \le e^{h^2 g(hb)\,\mathrm{var}_k}.$$

Hence by Lemma 3.16

$$\begin{aligned}\mathbf{E}(Ze^{h(X-\mu)}) &\le \sup\left(Z\prod_k e^{h^2 g(hb)\,\mathrm{var}_k}\right)\\ &= \sup\left(Ze^{h^2 g(hb)V}\right)\\ &\le e^{h^2 g(hb)\sup(ZV)}\\ &\le e^{h^2 g(hb)v}.\end{aligned}$$

But now as in the proof of the last theorem

$$\begin{aligned}\Pr((X-\mu \ge t) \cap (V \le v)) &\le e^{-ht}\mathbf{E}(Ze^{h(X-\mu)})\\ &\le e^{-ht+h^2g(hb)v},\end{aligned}$$

and we may complete the proof as for Theorem 2.7. □

Inequalities for maxima

We now amplify the comment at the end of Section 2 on maxima. Let $Y_1, \dots, Y_n$ be a martingale difference sequence and let $S_k = Y_1 + \cdots + Y_k$ as usual. Let $h > 0$ and let $T_k = e^{hS_k}$. Then $T_1, \dots, T_n$ form a submartingale (as long as the T_k are integrable), so we may apply Doob's maximal inequality for submartingales – see for example [21] section 12.6 or [?] section 14.6. We find that for any $t \ge 0$

$$\Pr(\max_k S_k \ge t) = \Pr(\max_k T_k \ge e^{ht}) \le e^{-ht}\mathbf{E}(T_n) = e^{-ht}\mathbf{E}(e^{hS_n}).$$

Thus all the martingale results based directly on the Bernstein inequality may be strengthened immediately to refer to maxima, just like those in Section 2 as noted in [30]. (See also [34, 65, 66].)

This comment applies to Lemma 3.11 and Theorems 3.12 and 3.13 (and thus also to Theorem 3.10), and to the inequalities (3.26) and (3.29). In particular, for example, in Theorem 3.13 the inequality (3.25) may be strengthened to read that for any $t \ge 0$,

$$\Pr(\max|\sum_{i=1}^{k} Y_i| \ge t) \le 2e^{-2t^2/\sum(b_k-a_k)^2}, \qquad (3.30)$$

where the maximum is over $k = 1, \dots, n$.

3.6 Centering Sequences

Given a sequence $X_1, X_2, \dots$ of random variables the corresponding difference sequence is $Y_1, Y_2, \dots$ where $Y_k = X_k - X_{k-1}$ (and where we set $X_0 = 0$). Let $\mu_k(x) = \mathbf{E}(Y_k \mid X_{k-1} = x)$. We call the distribution of the sequence centering if for each $k = 2, 3, \dots$, $\mu_k(x)$ is a non-increasing function of x – see [47]. Observe that a martingale is trivially centering since $\mu_k(x) = 0$.

The basic inequalities discussed above for a martingale difference sequence may be extended to centering sequences with bounded differences. The most fundamental example for the martingale inequalities involves the binomial distribution, as in Theorem 2.1. Now we can include the hypergeometric distribution naturally in the same inequalities – see also [29, 15].

Let $(x_1, \dots, x_n) \in \{0,1\}^n$ with $\sum x_i = r$. Let $(Z_1, \dots, Z_n)$ be a random linear order of the set $\{1, \dots, n\}$, where all $n!$ such orders are equally likely. Let $Y_k = x_{Z_k}$ and $X_k = \sum_{i=1}^{k} Y_i$. Then X_k has the hypergeometric distribution corresponding to counting the red elements in a random sample picked without replacement from the set $\{1, \dots, n\}$ with r elements painted red. We are interested in the concentration of X_k. Note that $E(X_k) = rk/n$. But the sequence $X_0, X_1, \ldots, X_n$ is centering, since

$$u_k(x) = E(X_k \mid X_{k-1} = x) = \frac{r - x}{n - k + 1},$$

which is a decreasing function of x. From the centering version in [47] of Theorem 2.3(c) above, it follows for example that, if μ denotes $E(X_k)$, then for any $\epsilon > 0$

$$\Pr(X_k \le (1-\epsilon)\mu) \le e^{-\frac{1}{2}\epsilon^2\mu}.$$

If we try to apply here the inequalities for martingales with bounded differences in the natural way (that is, with $\mathcal{F}_k$ as the σ-field generated by revealing the first k elements picked), we obtain an unwanted factor < 1 in the exponent in the bound. Centering sequences also arise naturally in occupancy or 'balls in boxes' problems – see [31, 47].

4. Talagrand's Inequality

4.1 The Inequality

Let $\Omega_1, \ldots, \Omega_n$ be probability spaces, and let Ω denote the product space. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables with X_k taking values in Ω_k. We saw earlier that for any subset A of Ω such that $\Pr(\mathbf{X} \in A)$ is not too small, with high probability a random point $\mathbf{X}$ is close to A when we consider Hamming distance or generalised Hamming distance. It turns out to be very fruitful to consider a related notion of distance.

Let $\alpha = (\alpha_1, \ldots, \alpha_n) \ge 0$ be an n-vector of non-negative real numbers. Recall that for points $\mathbf{x} = (x_1, \ldots, x_n)$ and $\mathbf{y} = (y_1, \ldots, y_n)$ in Ω, the α-Hamming distance $d_\alpha(\mathbf{x}, \mathbf{y})$ is the sum of the values α_i over those indices i such that $x_i \ne y_i$; and for a subset A of Ω, $d_\alpha(\mathbf{x}, A) = \inf\{d_\alpha(\mathbf{x}, \mathbf{y}) : \mathbf{y} \in A\}$. Talagrand's convex distance $d_T(\mathbf{x}, A)$ is defined to be $\sup\{d_\alpha(\mathbf{x}, A)\}$ where the supremum is over all choices of non-negative unit n-vector α (that is, with $\|\alpha\| = 1$).

By considering the n-vector α with each coordinate $1/\sqrt{n}$, we see that $d_T(\mathbf{x}, A) \ge d_\alpha(\mathbf{x}, A) = (1/\sqrt{n}) d_H(\mathbf{x}, A)$, so upper bounds on $d_T(\mathbf{x}, A)$ give upper bounds on $d_H(\mathbf{x}, A)$, but we shall see that they will tell us much more. The reason for the name 'convex distance' will emerge later. Talagrand [63] in fact considers also other notions of distance (see also [70]), but we shall focus only on the convex distance. We call the following fundamental result 'Talagrand's inequality'.

Theorem 4.1. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables and let A be a subset of the product space. Then for any $t \ge 0$,*

$$\Pr(\mathbf{X} \in A) \Pr(d_T(\mathbf{X}, A) \ge t) \le e^{-t^2/4}. \qquad (4.1)$$

If we consider a single non-negative unit vector α, then $d_T \ge d_\alpha$ and the above result yields a form of Theorem 3.6, but it is in fact far more powerful since it refers simultaneously to all possible generalised Hamming distances, as will be evident from our applications below. We shall see that this power is most evident when we consider the concentration of a function $f(\mathbf{X})$ where an inequality $f(\mathbf{x}) \ge b$ typically can be verified by examining only a few of the co-ordinate values x_i, and for different vectors $\mathbf{x}$ we may examine different co-ordinates. In some applications we profit greatly from the flexibility of choosing an appropriate unit vector α for each $\mathbf{x}$, rather than having to consider any Hamming distance. Note that we must assume that the random variables X_k are independent, in contrast to the situation with the martingale results (but see the recent paper of Marton [42], which gives an extension of Talagrand's inequality in which a limited dependence is allowed). Theorems 4.3 and 4.5 below are useful specialisations of Talagrand's inequality, on which we base all the applications here. We shall prove Theorem 4.1 later, but before that let us consider some applications.

4.2 Some Applications

4.2.1 Subsequences and Configuration Functions. Given a sequence $\mathbf{x} = (x_1, \ldots, x_n)$ of real numbers, we let $\mathrm{inc}(\mathbf{x})$ denote the length of a longest increasing subsequence. Thus $\mathrm{inc}(\mathbf{x})$ is the maximum value of $|K|$ over all subsets K of $\{1, \ldots, n\}$ such that the corresponding subsequence $(x_k : k \in K)$ is increasing, that is $x_i < x_j$ whenever $i, j \in K$ with $i < j$.

Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables each taking real values. We are interested in the concentration of the random variable $\mathrm{inc}(\mathbf{X})$. Let μ be the mean of $\mathrm{inc}(\mathbf{X})$. It follows directly from the independent bounded differences inequality, Theorem 3.1, that for any $t \ge 0$,

$$\Pr(|\mathrm{inc}(\mathbf{X}) - \mu| \ge t) \le 2e^{-2t^2/n}. \qquad (4.2)$$

This shows that for large n, with high probability $\mathrm{inc}(\mathbf{X})$ is confined within an interval of length $O(\sqrt{n})$. Using Talagrand's inequality we can deduce a much improved result. Let m be a median for $\mathrm{inc}(\mathbf{X})$.

Theorem 4.2. *For any $t \ge 0$,*

$$\Pr(\mathrm{inc}(\mathbf{X}) \ge m + t) \le 2e^{-t^2/4(m+t)} \qquad (4.3)$$

and

$$\Pr(\mathrm{inc}(\mathbf{X}) \le m - t) \le 2e^{-t^2/4m}. \qquad (4.4)$$

Such inequality and moreover the bounded differences method will give nearly as good results – see [15]. It is known (see for example [5]) that when the random variables X_i all have the same continuous distribution, the median $m \sim 2\sqrt{n}$ as $n \to \infty$. Thus the above results show that with high probability $\mathrm{inc}(\mathbf{X})$ is confined within an interval of length $O(n^{1/4})$, which is the best bound known. (In particular, the mean μ and the median m must be within $O(n^{1/4})$ of each other – see Lemma 4.6 below.)

It turns out that the approach based on Talagrand's inequality to the longest increasing subsequence problem will handle a general class of problems. Observe that the function $f(\mathbf{x}) = \mathrm{inc}(\mathbf{x})$ has the following property. For each $\mathbf{x} \in \Omega$ there is a subset $K = K(\mathbf{x})$ of the index set $\{1, \ldots, n\}$ such that $f(\mathbf{x}) = |K|$, and for each $\mathbf{y} \in \Omega$ we have

$$f(\mathbf{y}) \ge |\{i \in K : y_i = x_i\}| = f(\mathbf{x}) - |\{i \in K : y_i \ne x_i\}|.$$

Thus for each $\mathbf{x} \in \Omega$ there is a non-negative unit n-vector α (namely the incidence vector of the set $K(\mathbf{x})$ scaled by dividing by $\sqrt{f(\mathbf{x})}$) such that for each $\mathbf{y} \in \Omega$ we have

$$f(\mathbf{y}) \ge f(\mathbf{x}) - \sqrt{f(\mathbf{x})}\, d_\alpha(\mathbf{x}, \mathbf{y}).$$

This is the key property. We call a function f defined on a set Ω of n-vectors a *c-configuration function* if it has the following property: for each $\mathbf{x} \in \Omega$ there is a non-negative unit n-vector α such that for each $\mathbf{y} \in \Omega$ we have

$$f(\mathbf{y}) \ge f(\mathbf{x}) - \sqrt{c f(\mathbf{x})}\, d_\alpha(\mathbf{x}, \mathbf{y}).$$

Thus $\mathrm{inc}(\mathbf{x})$ gives a 1-configuration function, and so the next result extends the last one. (We shall give a related example below concerning common subsequences. Also we shall discuss concentration around the mean rather than the median in the next subsection – see Lemma 4.6.)

Theorem 4.5. *Let f be a c-configuration function, and let m be a median for $f(\mathbf{X})$. Then for any $t \ge 0$*

$$\Pr(f(\mathbf{X}) \ge m + t) \le 2e^{-t^2/4c(m+t)} \qquad (4.5)$$

and

$$\Pr(f(\mathbf{X}) \le m - t) \le 2e^{-t^2/4cm}. \qquad (4.6)$$

Proof. Let $\mathbf{x} \in \Omega$, and let α be a non-negative unit n-vector such that for any $\mathbf{y} \in \Omega$,

$$f(\mathbf{x}) \le f(\mathbf{y}) + \sqrt{c f(\mathbf{x})}\, d_\alpha(\mathbf{x}, \mathbf{y}).$$

Let $A_a = \{\mathbf{y} \in \Omega : f(\mathbf{y}) \le a\}$. Then by the above

$$f(\mathbf{x}) \le a + \sqrt{c f(\mathbf{x})}\, d_\alpha(\mathbf{x}, \mathbf{y})$$

for each $\mathbf{y} \in A_a$, and so by minimising over such $\mathbf{y}$ we have

$$f(\mathbf{x}) \le a + \sqrt{c f(\mathbf{x})}\, d_\alpha(\mathbf{x}, A_a) \le a + \sqrt{c f(\mathbf{x})}\, d_T(\mathbf{x}, A_a).$$

Thus if $f(\mathbf{x}) \ge a + t$ then

$$d_T(\mathbf{x}, A_a) \ge \frac{f(\mathbf{x}) - a}{\sqrt{c}\sqrt{f(\mathbf{x})}} \ge \frac{t}{\sqrt{c(a+t)}},$$

since the function $g(u) = (u - a)/\sqrt{u}$ is increasing for $u \ge a$. Thus for any $t \ge 0$,

$$\Pr(f(\mathbf{X}) \ge a + t) \le \Pr\left(d_T(\mathbf{X}, A_a) \ge \frac{t}{\sqrt{c(a+t)}}\right).$$

Hence by Talagrand's inequality, for any $t \ge 0$

$$\begin{aligned} \Pr(f(\mathbf{X}) \le a)\Pr(f(\mathbf{X}) \ge a + t) &\le \Pr(\mathbf{X} \in A_a)\Pr\left(d_T(\mathbf{X}, A_a) \ge \frac{t}{\sqrt{c(a+t)}}\right) \\ &\le e^{-\frac{t^2}{4c(a+t)}}. \end{aligned}$$

Now we may complete the proof by appropriate choices of a in this last inequality. If we let $a = m$ then since $\Pr(f(\mathbf{X}) \le m) \ge \frac{1}{2}$, we obtain (4.5); and if we let $a = m - t$ then since $\Pr(f(\mathbf{X}) \ge m) \ge \frac{1}{2}$, we obtain (4.6). □

Now let us consider a related problem concerning common subsequences of two sequences. Given two sequences $\mathbf{x} = (x_1, \ldots, x_{n_1})$ and $\mathbf{y} = (y_1, \ldots, y_{n_2})$, let $\mathrm{com}(\mathbf{x}, \mathbf{y})$ denote the maximum length of a common subsequence of $\mathbf{x}$ and $\mathbf{y}$. Let $\mathbf{X} = (X_1, \ldots, X_{n_1})$ and $\mathbf{Y} = (Y_1, \ldots, Y_{n_2})$ be independent families of independent random variables. We are interested in the concentration of the random variable $\mathrm{com}(\mathbf{X}, \mathbf{Y})$. Let μ be the mean of $\mathrm{com}(\mathbf{X}, \mathbf{Y})$.

As for the longest increasing subsequence problem, it follows directly from the independent bounded differences inequality, Theorem 3.1, that for any $t > 0$,

$$\Pr(|\mathrm{com}(\mathbf{X}, \mathbf{Y}) - \mu| \ge t) \le 2e^{-2t^2/(n_1+n_2)}. \qquad (4.7)$$

This shows that, when say $n_1 = n_2 = n$ and n is large, with high probability $\mathrm{com}(\mathbf{X}, \mathbf{Y})$ is confined within an interval of length $O(n^{1/2})$. Using the above results for configuration functions we may obtain a similar result. For, if we regard $\mathrm{lcs}(\mathbf{x},\mathbf{y})$ as a function of the n_1+n_2 variables in the natural way then it is a 2-configuration function. So if we let m be a median for $\mathrm{lcs}(\mathbf{X},\mathbf{Y})$, we obtain

Theorem 4.4. *For any $t \geq 0$,*

$$\Pr(\mathrm{lcs}(\mathbf{X},\mathbf{Y}) \geq m+t) \leq 2e^{-t^2/4(m+t)} \tag{4.8}$$

and

$$\Pr(\mathrm{lcs}(\mathbf{X},\mathbf{Y}) \leq m-t) \leq 2e^{-t^2/4m}. \tag{4.9}$$

Consider the case when $n_1 = n_2 = n$ and n is large, and when the random variables X_i all have the same (fixed) discrete distribution F. It is easy to see (using superadditivity) that there is a constant $\delta_F > 0$ (depending on the distribution F) such that

$$\mathbf{E}[\mathrm{lcs}((X_1,\ldots,X_n),(Y_1,\ldots,Y_n))]/n \to \delta_F,$$

and the corresponding result holds for the median. But if say F is the uniform distribution on the set $\{1,\ldots,N\}$ where N is large, then the constant δ_F will be very small, and then the theorem above improves on (4.7).

4.3.2 Two Geometric Applications. We now consider applications to the lengths of travelling salesman tours and Steiner trees in the unit square. We shall use the following general result, which is derived from Talagrand's inequality Theorem 4.1, and which is similar to Theorem 4.3.

Theorem 4.5. *Let $\mathbf{X} = (X_1,\ldots,X_n)$ be a family of independent random variables with X_k taking values in a set Ω_k, and let $\Omega = \prod \Omega_k$. Let the real-valued function f on Ω satisfy the condition that, for each $\mathbf{x} \in \Omega$ there exists a non-negative unit α-vector α such that*

$$f(\mathbf{x}) \leq f(\mathbf{y}) + c\,d_\alpha(\mathbf{x},\mathbf{y}) \quad \text{for each } \mathbf{y} \in \Omega. \tag{4.10}$$

Then

$$\Pr(|f(\mathbf{X}) - m| \geq t) \leq 4e^{-t^2/4c^2},$$

where m is a median of $f(\mathbf{X})$. The same conclusion holds if the condition (4.10) is replaced by

$$f(\mathbf{y}) \leq f(\mathbf{x}) + c\,d_\alpha(\mathbf{x},\mathbf{y}) \quad \text{for each } \mathbf{y} \in \Omega. \tag{4.11}$$

Part of the power of this result comes from the asymmetry, that we do not require that both conditions (4.10) and (4.11) hold. Observe that if both hold then we have a bound on $|f(\mathbf{x}) - f(\mathbf{y})|$, and thus on the sum of squared ranges $\hat{r}^2$ when the random variables X_k are independent.

Proof. For each real number a let $A_a = \{\mathbf{y} \in \Omega : f(\mathbf{y}) \leq a\}$. Consider any point $\mathbf{x} \in \Omega$. There is a non-negative unit α-vector α such that for each $\mathbf{y} \in \Omega$

$$f(\mathbf{x}) \leq f(\mathbf{y}) + c\,d_\alpha(\mathbf{x},\mathbf{y}),$$

and so

$$f(\mathbf{x}) \leq a + c\,d_\alpha(\mathbf{x},\mathbf{y})$$

for each $\mathbf{y} \in A_a$. By minimising over such $\mathbf{y}$ we see that

$$f(\mathbf{x}) \leq a + c\,d_\alpha(\mathbf{x},A_a) \leq a + c\,d_T(\mathbf{x},A_a).$$

Thus if $f(\mathbf{x}) \geq a+t$ then $d_T(\mathbf{x},A_a) \geq t/c$. Hence

$$\begin{aligned}\Pr(f(\mathbf{X}) \leq a)\Pr(f(\mathbf{X}) \geq a+t) &\leq \Pr(\mathbf{X} \in A_a)\Pr(d_T(\mathbf{X},A_a) \geq t/c)\\ &\leq e^{-t^2/4c^2}\end{aligned}$$

by Talagrand's inequality, Theorem 4.1. If we let $a = m$ we obtain

$$\Pr(f(\mathbf{X}) \geq m+t) \leq 2e^{-t^2/4c^2}$$

and similarly if we let $a = m-t$ we obtain

$$\Pr(f(\mathbf{X}) \leq m-t) \leq 2e^{-t^2/4c^2}$$

which completes the proof for the case when condition (4.10) holds.

Suppose now that condition (4.11) holds (but not necessarily condition (4.10)). Let $g(\mathbf{x}) = -f(\mathbf{x})$. Then g satisfies condition (4.10), and $(-m)$ is a median of $g(\mathbf{X})$, and so by the above

$$\Pr(|f(\mathbf{X}) - m| \geq t) = \Pr(|g(\mathbf{X}) - (-m)| \geq t) \leq 4e^{-t^2/4c^2},$$

as required. □

Before we consider the geometric applications let us check that, indeed as mentioned earlier, it does not much matter that Theorems 4.3 and 4.5 concern concentration around the median m rather than the mean μ, since the concentration inequalities themselves imply that $|\mu - m|$ is small.

Lemma 4.6. *Let the random variable Y have mean μ and median m, and let $a, b > 0$.*

(a) If $\Pr(Y - m \geq t) \leq ae^{-t^2/b}$ for any $t > 0$, then $\mu - m \leq (\sqrt{\pi}/2)a\sqrt{b}$, and so if also $\Pr(Y - m \leq -t) \leq ae^{-t^2/b}$ for any $t > 0$, then $|\mu - m| \leq (\sqrt{\pi}/2)a\sqrt{b}$.

(b) If $\Pr(Y - m \geq t) \leq ae^{-t^2/b(m+t)}$ for any $t > 0$, then $\mu - m \leq \sqrt{\pi}a\sqrt{bm} + 2abe^{-m/b}$ (which is $O(\sqrt{m})$ as $m \to \infty$, assuming that a and b are constants).

Proof. We have

$$\mu - m = \mathbf{E}(Y - m) \le \mathbf{E}((Y - m)^+) = \int_0^\infty \Pr(Y - m > t)\,dt. \qquad (4.12)$$

In case (a)

$$\int_0^\infty \Pr(Y - m > t)\,dt \le a \int_0^\infty e^{-t^2/b}\,dt = (\sqrt{\pi}/2)a\sqrt{b},$$

and so the first part of (a) follows from (4.12). For the second part, note that $(-m)$ is a median for $(-Y)$ and $\Pr((-Y) - (-m) \ge t) = \Pr(Y - m \le -t)$. So if $\Pr(Y - m \le -t) \le ae^{-t^2/b}$ for any $t > 0$ then by what we have just proved

$$m - \mu = \mathbf{E}(-Y - (-m)) \le (\sqrt{\pi}/2)a\sqrt{b}.$$

In case (b), we again use (4.12). Now we have

$$\begin{aligned}\int_0^\infty \Pr(Y - m > t)\,dt &\le \int_0^\infty ae^{-t^2/b(m+t)}\,dt \\ &\le a\int_0^m e^{-t^2/2bm}\,dt + a\int_m^\infty e^{-t/2b}\,dt \\ &\le \sqrt{\pi/2}\,a\sqrt{bm} + 2abe^{-m/2b}.\end{aligned}$$

□

We shall consider a family $\mathbf{X} = (X_1, \ldots, X_n)$ of independent random variables where each X_i takes values in the unit square $[0,1]^2$. Thus here $\Omega = ([0,1]^2)^n$.

Travelling salesman tours

Given a point $\mathbf{x} \in \Omega$, let $tsp(\mathbf{x})$ be the minimum length of a travelling salesman tour through these points. Much effort has been devoted to investigating the random variable $tsp(\mathbf{X})$, and to investigating its concentration in particular – see for example [66]. Talagrand's inequality effortlessly yields results which previously took great ingenuity.

We need to know one deterministic result, namely that there is a constant c such that the following holds. For every n and every $\mathbf{x} \in \Omega$, there is a tour $T(\mathbf{x})$ through the points in $\mathbf{x}$ such that the sum of the squares of the lengths of the edges in this tour is at most c. This may be proved for example by considering space-filling curves – see [53, 66]. We shall use $T(\mathbf{x})$ to define an appropriate vector α, where the coordinate α_i corresponds to the 'awkwardness' of the point x_i.

Given $\mathbf{x} \in \Omega$, we let β_i be the sum of the lengths of the two edges incident to the point x_i in the tour $T(\mathbf{x})$. Thus $\sum \beta_i^2 \le 4c$ (using the fact that $(a+b)^2 \le 2a^2 + 2b^2$). We shall see that for any $\mathbf{y} \in \Omega$

$$tsp(\mathbf{x}) \le tsp(\mathbf{y}) + d_\beta(\mathbf{x}, \mathbf{y}) \le tsp(\mathbf{y}) + (2\sqrt{c})d_\alpha(\mathbf{x}, \mathbf{y}), \qquad (4.13)$$

where α is the unit vector $\beta/\|\beta\|$. Thus the function $tsp(\mathbf{x})$ satisfies the condition (4.10) in Theorem 4.5 (with the value 'c' there being $2\sqrt{c}$). Hence for any $t > 0$,

$$\Pr(|tsp(\mathbf{X}) - m| \ge t) \le 4e^{-t^2/16c}, \qquad (4.14)$$

where m is a median for $tsp(\mathbf{X})$. A result of this form was first proved by Rhee and Talagrand [56], by a much more involved argument based on the martingale approach.

It remains then to prove (4.13). Let x, y denote the sets of points corresponding to $\mathbf{x}$, $\mathbf{y}$ respectively. If $x \cap y = \emptyset$ then $d_\beta(\mathbf{x}, \mathbf{y})$ is twice the length of the tour $T(\mathbf{x})$, and so certainly the inequality (4.13) holds. Suppose then that $x \cap y \neq \emptyset$. We pick a multiset F of edges between the points of x as follows. For each segment in the tour $T(\mathbf{x})$ of the form $a, x_i, \ldots, x_j, b$ where $a, b \in x \cap y$ and $x_i, \ldots, x_j \in x \setminus y$ (note that $a = b$ if $|x \cap y| = 1$), we put into F each of the edges $x_k x_{k+1}$ doubled for $k = i, \ldots, j-1$, and the doubles of the edges ax_i and $x_j b$ also. Thus corresponding to each such segment we obtain a cycle containing exactly one point in y, and with the sum of the lengths of the edges at most the sum of the coordinates of β corresponding to the points x_k. These cycles between them cover all the points in $x \setminus y$, and the sum of the lengths of all the edges in F is at most $d_\beta(\mathbf{x}, \mathbf{y})$.

Now let $T^*(\mathbf{y})$ be an optimal tour for $\mathbf{y}$. Consider the (multi)graph G with vertex set $x \cup y$ and with edge set consisting of the edges in $T^*(\mathbf{y})$ together with the edges in F. The graph G is connected and each vertex degree is even, and so G has an Eulerian tour. This tour can be shortcut to give a travelling salesman tour, which by the triangle inequality has length no more than the sum of the lengths of the edges in G, and this sum is at most $tsp(\mathbf{y}) + d_\beta(\mathbf{x}, \mathbf{y})$. This completes the proof of (4.13), as required.

Steiner trees

A *Steiner tree* for a set x of points in the unit square is a tree with vertex set some set of points in the plane containing x. Given $\mathbf{x} \in \Omega$, we let $st(\mathbf{x})$ denote the minimal length of a Steiner tree for the corresponding set x. We may use the tour $T(\mathbf{x})$ exactly as above to define a corresponding vector β.

Now let $\mathbf{y} \in \Omega$, and let $S^*(\mathbf{y})$ be an optimal Steiner tree for the corresponding set of points y. Consider the set E of edges consisting of the edges in $S^*(\mathbf{y})$ together with those edges in $T(\mathbf{x})$ with at least one end not in y. The total length of these edges is at most $s(y) + d_\alpha(x, y)$, and we have already seen that $\sum_i \alpha_i^2 \le 4r$. The key observation is that the graph G on $x \cup y$ with edge set E is connected. For, since $T(x)$ is connected each point in x is in the same component as some point in x_0, and since $S^*(y)$ is connected each point in y is in the same component. It follows that $s(x)$ is at most the sum of the lengths of the edges in G, and thus $s(x) \le s(y) + d_\alpha(x, y)$. Hence by Theorem 4.5, for $t \ge 0$

$$\Pr(|s(X) - m| \ge t) \le 4e^{-t^2/16r}, \tag{4.16}$$

where m is a median for $s(X)$.

4.2.9 Random Minimum Spanning Trees. Consider the complete graph K_n with random independent edge lengths X_i, each uniformly distributed on $(0,1)$. Let L_n be the corresponding random length of a minimum spanning tree. It is known [28] that the expected value of L_n tends to $\zeta(3)$ as $n \to \infty$, where

$$\zeta(3) = \sum_{i=1}^{\infty} i^{-3} \sim 1.202.$$

It is shown in [29] that L_n is quite concentrated around $\zeta(3)$ using the method of bounded differences, and this result is improved in [8] using Talagrand's method. (Also, it is shown in [33] that $\sqrt{n}(L_n - \zeta(3))$ is asymptotically normally distributed.)

Both the bounded differences method and Talagrand's method can in fact be used to prove that L_n is very highly concentrated around the value $\zeta(3)$ – see [46] but the latter method is far easier and will be described below. (In fact the bounded differences approach seems to yield a slightly stronger result.) Both approaches depend on the fact that long edges are not important. For $0 \le b \le 1$ let $L_n^{(b)}$ be the minimum length of a spanning tree when the edge lengths X_i are replaced by $\min(X_i, b)$. For simplicity we consider here the case of a fixed deviation $t > 0$. We need the following lemma.

Lemma 4.7 [46] *For any $t > 0$ there exist constants $c_1 > 0$ and n_0 such that if we let $b = c_1/n$ then*

$$\Pr(L_n - L_n^{(b)} \ge t) \le e^{-n}.$$

We shall prove the following concentration result for the minimum spanning tree length L_n.

Theorem 4.8 *For any $t > 0$ there exists $\delta > 0$ such that*

$$\Pr(|L_n - \zeta(3)| > t) \le e^{-\delta n} \quad \textit{for all } n.$$

It is easy to see that the bound above is of the right order. For example, for each $n \ge 5$ the probability that $L_n \ge 2$ is at least the probability that each edge incident with the first four vertices has length at least 1/2, and this probability is at least $(1/16)^n$.

Proof. Let $N = \binom{n}{2}$ and let $Y = (Y_1, \ldots, Y_N)$ be a family of independent random variables with each Y_i uniformly distributed on $(0,1)$, corresponding to the edge lengths in the graph K_n. We may write the random variable L_n as $mst(Y)$.

Let $0 < b \le 1$ and let $\Omega = [0,b]^N$. For each $i = 1, \ldots, N$ let $X_i = \min(Y_i, b)$. Then $X = (X_1, \ldots, X_N)$ is a family of independent random variables each taking values in $[0,b]$, and $L_n^{(b)} = mst(X)$.

Now consider the random variable $mst(X)$. Let $\Omega = [0,b]^N$ and let $x \in \Omega$. Denote the set of edges in a corresponding minimum spanning tree by $T = T(x)$. Let $\beta = \beta(x)$ be the N-vector with $\beta_i = b$ for $i \in T$ and $\beta_i = 0$ otherwise, and let $\alpha = \alpha(x)$ be the unit vector $\beta/(b\sqrt{n-1})$. Then for any $y \in \Omega$,

$$\begin{aligned} mst(y) &\le \sum_{i \in T} y_i \\ &\le \sum_{i \in T} x_i + \sum_{i \in T} (y_i - x_i)^+ \\ &\le mst(x) + d_\beta(x, y) \\ &\le mst(x) + b\sqrt{n}\, d_\alpha(x, y). \end{aligned}$$

Thus the function $mst(x)$ satisfies condition (4.11) in Theorem 4.5 with $c = b\sqrt{n}$, and so for any $t \ge 0$

$$\Pr(|mst(X) - m| \ge t) \le 4e^{-t^2/4b^2 n},$$

where m is a median for $mst(X)$. We may use Lemma 4.7, together with this last inequality with $b = c_1/n$, to obtain

$$\begin{aligned} \Pr(|mst(Y) - m| \ge 2t) &\le \Pr(mst(Y) - mst(X) \ge t) + \Pr(|mst(X) - m| \ge t) \\ &\le e^{-n} + 4e^{-t^2 n/4c_1^2}. \end{aligned}$$

It follows that for any $t > 0$ there exists $\delta_1 = \delta_1(t) > 0$ such that

$$\Pr(|L_n - m| \ge t) \le e^{-\delta_1 n}.$$

It remains to tidy up by replacing the median by $\zeta(3)$ (in the spirit of Lemma 4.6). By the above

∎

$$\mathbf{E}(L_n) - m \le \mathbf{E}(|L_n - m|) \le \frac{t}{4} + n\Pr(|L_n - m| \ge t/4) \le t/3$$

for n sufficiently large. Also we saw earlier that for n sufficiently large, $|\mathbf{E}(L_n) - \gamma(2)| \le t/3$ and so $|m - \gamma(2)| \le 2t/3$ for n sufficiently large. Hence for n sufficiently large

$$\Pr(|L_n - \gamma(2)| \ge t) \le \Pr(|L_n - m| \ge t/3) \le e^{-\beta n}$$

where $\beta = \beta(t/3)$, and the theorem follows. □

4.3 Proof of Talagrand's Inequality

In this subsection we shall prove an extended form of Theorem 4.1.

Theorem 4.9. *Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of independent random variables where X_k takes values in a set Ω_k, and let A be a subset of the product space $\Omega = \prod \Omega_k$. Then*

$$\Pr(\mathbf{X} \in A)\,\mathbf{E}\left(e^{d_T(\mathbf{X},A)^2/4}\right) \le 1, \tag{4.16}$$

and so, for any $t \ge 0$

$$\Pr(\mathbf{X} \in A)\Pr(d_T(\mathbf{X}, A) \ge t) \le e^{-t^2/4}. \tag{4.17}$$

The latter inequality (4.17) (which is Theorem 4.1) follows immediately from the former (4.16) by Markov's inequality. The scheme of the proof of (4.16) is as follows. We first develop an equivalent definition of Talagrand's distance d_T. Then after two technical lemmas we start the main proof by induction on n. We prove a claim bounding the distance $d_T(x, A)$ in dimension $n+1$ in terms of distances involving only the first n co-ordinates. This claim involves a parameter λ. The induction hypothesis yields bounds for the distances in dimension n. We then optimise over λ and average over the last coordinate. The whole proof is neither long nor hard, but it is one of those proofs by induction in which it is not easy to get a good feel about why the result really is true. For a brief discussion of an alternative approach based on ideas from information theory see the next (final) subsection.

In order to prove (4.17) we first develop the alternative characterisation of Talagrand's convex distance $d_T(x, A)$. For a point x and a set A in Ω, let $U = U(x, A)$ be the set of all binary vectors u such that starting from x we may reach a vector $y \in A$ by changing only coordinates x_i such that $u_i = 1$ (and not necessarily changing all of them). Thus $0 \in U$ if and only if $x \in A$. Further let $V = V(x, A)$ be the convex hull of the set U. The following lemma explains the term 'convex distance'.

Lemma 4.10

$$d_T(x, A) = \min\{\|v\| : v \in V\}. \tag{4.18}$$

Proof. If $x \in A$ then both sides above equal 0. So we may assume that $x \notin A$, and then both sides are positive. Denote the right hand side above by ρ. Let $\alpha = (\alpha_1, \ldots, \alpha_n) \ge 0$ be a unit vector. We write $\alpha \cdot u$ to denote the inner product $\sum \alpha_i u_i$. Then

$$d_\alpha(x, A) = \min_{y \in A} d_\alpha(x, y) = \min_{u \in U} \alpha \cdot u = \min_{v \in V} \alpha \cdot v, \tag{4.19}$$

since the minimum of a linear functional over the convex hull V of the finite set U must be achieved at a point of U. But by the Cauchy–Schwarz inequality,

$$\alpha \cdot v \le \|\alpha\|\,\|v\| = \|v\|.$$

Thus $d_\alpha(x, A) \le \rho$, and since this holds for every choice of α we deduce that $d_T(x, A) \le \rho$.

For the converse result, note that the minimum in (4.18) is achieved, that is there is a point $\bar{v} \in V$ with norm equal to ρ, since V is compact. Let α be the unit vector $\bar{v}/\rho$. Consider any point $v \in V$. Since V is convex, the point $\bar{v} + \lambda(v - \bar{v})$ is in V for each $0 \le \lambda \le 1$, and so

$$(\bar{v} + \lambda(v - \bar{v})) \cdot (\bar{v} + \lambda(v - \bar{v})) \ge \bar{v} \cdot \bar{v}.$$

This yields

$$2\lambda\,\bar{v} \cdot (v - \bar{v}) + \lambda^2 (v - \bar{v}) \cdot (v - \bar{v}) \ge 0,$$

and by considering small λ we see that $\bar{v} \cdot (v - \bar{v}) \ge 0$. Thus $\alpha \cdot v \ge \alpha \cdot \bar{v} = \rho$ for all $v \in V$. Hence by (4.19),

$$d_T(x, A) \ge d_\alpha(x, A) = \min_{v \in V} \alpha \cdot v = \rho,$$

and we are done. □

We need two further lemmas before we start the main proof of Talagrand's inequality. The first is from [81, 95].

Lemma 4.11. *For all $0 \le r \le 1$,*

$$\inf_{0 \le \lambda \le 1} r^{-\lambda} e^{(1-\lambda)^2/4} \le 2 - r.$$

Proof. For the case $0 \le r \le e^{-\frac{1}{2}}$ we may consider $\lambda = 0$ and check that $e^{\frac{1}{4}} \le 2 - e^{-\frac{1}{2}}$. So suppose that $e^{-\frac{1}{2}} < r \le 1$. Let $\lambda = 1 + 2\ln r$, so $0 \le \lambda \le 1$. We need to show that $f(r) \ge 0$, where $f(r)$ is the logarithm of the ratio of the right side of the inequality to the left side. Now

$$f(r) = \ln(2-r) + \lambda \ln r - (1-\lambda)^2/4 = \ln(2-r) + \ln r + (\ln r)^2.$$

Since $f(1) = 0$, it suffices to show that $g(r) = rf'(r) \le 0$. Note that

$$g(r) = r\left(-\frac{1}{2-r} + \frac{1}{r} + \frac{2\ln r}{r}\right) = -\frac{r}{2-r} + 1 + 2\ln r.$$

Since $g(1) = 0$, it suffices now to show that $g'(r) \ge 0$. But $g'(r) = 2\left(\frac{1}{r} - \frac{1}{(2-r)^2}\right)$, and $\frac{1}{r} \ge 1 \ge \frac{1}{(2-r)^2}$, thus indeed $g'(r) \ge 0$, which completes the proof.

The last preliminary result we need is a form of Hölder's inequality (see for example [21] page 80), which we state and prove here for completeness, in a form useful to us.

Lemma 4.19 *For any (appropriately integrable) functions f and g, and any $0 \le t \le 1$,*

$$\mathbf{E}\left(e^{tf(X)+(1-t)g(X)}\right) \le \left(\mathbf{E}(e^{f(X)})\right)^t \left(\mathbf{E}(e^{g(X)})\right)^{1-t}.$$

Proof. Let $a, b > 0$ and for $0 \le t \le 1$ let $h(t) = a^t b^{1-t}$. Then $h''(t) = h(t)(\ln(a/b))^2 \ge 0$, so h is convex, and thus $a^t b^{1-t} \le ta + (1-t)b$. Now let $F = \mathbf{E}(e^{f(X)})$ and $G = \mathbf{E}(e^{g(X)})$. Then

$$(e^{f(X)}/F)^t (e^{g(X)}/G)^{1-t} \le (t/F)e^{f(X)} + ((1-t)/G)e^{g(X)}.$$

Taking expected values,

$$\begin{aligned}\mathbf{E}\left(e^{tf(X)+(1-t)g(X)}\right)/(F^t G^{1-t}) &= \mathbf{E}\left((e^{f(X)}/F)^t (e^{g(X)}/G)^{1-t}\right)\\ &\le (t/F)\mathbf{E}(e^{f(X)}) + ((1-t)/G)\mathbf{E}(e^{g(X)})\\ &= t + (1-t) = 1,\end{aligned}$$

which yields the required inequality. □

We may now start the main proof of the inequality (4.16). Let us write $p_1(A)$ for $\Pr(X \in A)$. We use induction on n. Consider first the case $n = 1$. Now $d_T(x, A)$ equals 0 if $x \in A$ and otherwise equals 1. So

$$\mathbf{E}\left(e^{\frac{1}{4}d_T(X,A)^2}\right) = p_1(A) + e^{\frac{1}{4}}(1 - p_1(A)).$$

But for $0 \le p \le 1$,

$$p(p + e^{\frac{1}{4}}(1-p)) \le p(p + 2(1-p)) = p(2-p) \le 1,$$

which completes the proof of the case $n = 1$.

Now let $n \ge 1$, suppose that the inequality (4.16) holds for n, and consider the case $n+1$. Denote $\prod_{i=1}^n \Omega_i$ by $\Omega^{(n)}$. Write $\prod_{i=1}^{n+1} \Omega_i$ as $\Omega^{(n+1)} = \Omega^{(n)} \times \Omega_{n+1}$, with typical element written as $z = (x, \omega)$, where $x \in \Omega^{(n)}$ and $\omega \in \Omega_{n+1}$. Let $A \subseteq \Omega^{(n+1)}$. For $\omega \in \Omega_{n+1}$ the ω-section A_ω of A is defined by

$$A_\omega = \{x \in \Omega^{(n)} : (x, \omega) \in A\}.$$

The projection of A is the set B defined by

$$B = \cup_\omega A_\omega = \{x \in \Omega^{(n)} : (x, \omega) \in A \text{ for some } \omega \in \Omega_{n+1}\}.$$

We next prove an inequality relating $d_T(z, A)$ to corresponding distances between x and the section and projection of A. The inequality involves a parameter λ which we shall later choose appropriately.

Claim *Let $z = (x, \omega) \in \Omega^{(n)} \times \Omega_{n+1}$ and let $0 \le \lambda \le 1$. Then*

$$d_T(z, A)^2 \le \lambda d_T(x, A_\omega)^2 + (1-\lambda) d_T(x, B)^2 + (1-\lambda)^2 \qquad (4.20)$$

Proof of Claim. By Lemma 4.10 above, there is a vector $v_1 \in V(x, A_\omega)$ with norm equal to $d_T(x, A_\omega)$ and a vector $v_2 \in V(x, B)$ with norm equal to $d_T(x, B)$. Now if $s \in U(x, A_\omega)$ then $(s, 0) \in U(z, A)$, and so if $v \in V(x, A_\omega)$ then $(v, 0) \in V(z, A)$. Similarly, if $s \in U(x, B)$ then $(s, 1) \in U(z, A)$, and so if $v \in V(x, B)$ then $(v, 1) \in V(z, A)$. Hence both $(v_1, 0)$ and $(v_2, 1)$ are in the convex set $V(z, A)$, and so if we set

$$v_\lambda = \lambda(v_1, 0) + (1-\lambda)(v_2, 1) = (\lambda v_1 + (1-\lambda)v_2, 1-\lambda),$$

then $v_\lambda \in V(z, A)$. By Lemma 4.10 again, $d_T(z, A)$ is at most the norm of v_λ. Now the function $f(x) = x^2$ is convex, and so

$$(\lambda a + (1-\lambda)b)^2 \le \lambda a^2 + (1-\lambda)b^2.$$

Hence

$$\begin{aligned}\|v_\lambda\|^2 &= \|\lambda v_1 + (1-\lambda)v_2\|^2 + (1-\lambda)^2\\ &\le \lambda\|v_1\|^2 + (1-\lambda)\|v_2\|^2 + (1-\lambda)^2\\ &= \lambda d_T(x, A_\omega)^2 + (1-\lambda) d_T(x, B)^2 + (1-\lambda)^2.\end{aligned}$$

This completes the proof of the claim.

We are now ready to tackle the induction step. For each fixed ω, let $E(\omega)$ denote

$$E\left(e^{\frac14 d((X,\omega),A)^2}\right) = E\left(e^{\frac14 d((X,X_{n+1}),A)^2} \mid X_{n+1}=\omega\right).$$

We shall first give an upper bound for $E(\omega)$ and then average over ω. Fix ω, and note that the claim gives

$$e^{\frac14 d((X,\omega),A)^2} \le e^{\frac14(1-\lambda)^2} e^{\frac{\lambda}{4} d(X,A_\omega)^2} e^{\frac{1-\lambda}{4} d(X,B)^2}.$$

Hence by Lemma 4.13 (Hölder's inequality) we obtain

$$E(\omega) \le e^{\frac14(1-\lambda)^2} E\left(e^{\frac14 d(X,A_\omega)^2}\right)^{\lambda} E\left(e^{\frac14 d(X,B)^2}\right)^{1-\lambda}.$$

By the induction hypothesis applied to the two expectations above, we find that

$$\begin{aligned} E(\omega) &\le e^{\frac14(1-\lambda)^2}(\mu_n(A_\omega))^{-\lambda}(\mu_n(B))^{-(1-\lambda)} \\ &= e^{\frac14(1-\lambda)^2}(\mu_n(B))^{-1}\left(\frac{\mu_n(A_\omega)}{\mu_n(B)}\right)^{-\lambda}. \end{aligned}$$

Thus for all $0 \le \lambda \le 1$,

$$E(\omega) \le (\mu_n(B))^{-1} r^{-\lambda} e^{\frac14(1-\lambda)^2},$$

where $r = \mu_n(A_\omega)/\mu_n(B)$, and so $0 \le r \le 1$. By Lemma 4.14 we find

$$E(\omega) \le (\mu_n(B))^{-1}(2 - \mu_n(A_\omega)/\mu_n(B)).$$

Now $\mu_n(A_\omega) = \Pr((X,X_{n+1}) \in A \mid X_{n+1} = \omega)$. We can average over the values ω taken by X_{n+1} to obtain

$$\begin{aligned} \mu_{n+1}(A)\, E\left(e^{\frac14 d((X,X_{n+1}),A)^2}\right) &\le \mu_{n+1}(A)(\mu_n(B))^{-1}(2 - \mu_{n+1}(A)/\mu_n(B)) \\ &= x(2-x) \le 1, \end{aligned}$$

where $x = \mu_{n+1}(A)/\mu_n(B)$. We have now completed the proof of the induction step and thus of the theorem. □

4.4 Ideas from Information Theory

There is a third main approach to proving general concentration results which uses ideas from information theory. Indeed, the first general concentration result seems to have been proved and used in this context, by Ahlswede, Gács and Körner [1] in 1976. Their concentration result, the 'blowing-up lemma', was sharpened by Csiszár and Körner [17], and then in 1986 Marton [40] gave a simple and elegant proof. This result resembles Theorem 3.5 above, though with a worse constant in the exponent. The optimal constant was obtained in 1996 by Marton [41], using the same elegant information-theoretic approach. Dembo [18] showed that the method is strong enough to recover all of the inequalities of Talagrand in [56] (including Theorem 4.9 above), where it is assumed that the random variables involved are independent. The method is extended in [42] to handle certain cases of weak dependence. For other recent work see [43, 57].

It is not clear if these ideas will lead to further new applications in discrete mathematics and theoretical computer science. However, they are very elegant and powerful, and so we try here to give a flavour of the method. We shall show how they give a very different proof of Theorem 3.5, following [40, 41].

Let $\Omega_1, \ldots, \Omega_n$ be finite sets, and let Ω denote their product $\prod_k \Omega_k$. Let $\mathbf{p} = (p_\omega : \omega \in \Omega)$ and $\mathbf{q} = (q_\omega : \omega \in \Omega)$ specify probability distributions on Ω. Let $\mathbf{X} = (X_1, \ldots, X_n)$ be a family of random variables with X_k taking values in Ω_k, and let $\mathbf{Y} = (Y_1, \ldots, Y_n)$ be another such family. We shall be interested in joint distributions for $\mathbf{X}$ and $\mathbf{Y}$ which have marginals $\mathbf{p}$ and $\mathbf{q}$, that is, such that

$$\Pr(\mathbf{X}=\omega) = \sum_{\omega' \in \Omega} \Pr((\mathbf{X},\mathbf{Y}) = (\omega,\omega')) = p_\omega$$

for each $\omega \in \Omega$, and similarly for $\mathbf{Y}$ and $\mathbf{q}$. We shall define a notion of distance between the distributions $\mathbf{p}$ and $\mathbf{q}$ based on the expected Hamming distance between random points $\mathbf{X}$ and $\mathbf{Y}$. Observe that the expected Hamming distance between $\mathbf{X}$ and $\mathbf{Y}$ is given by

$$E(d_H(\mathbf{X},\mathbf{Y})) = \sum_i \Pr(X_i \ne Y_i).$$

We define $d_H(\mathbf{p},\mathbf{q})$ to be the minimum value of $E(d_H(\mathbf{X},\mathbf{Y}))$ over all choices of joint distribution for $\mathbf{X}$ and $\mathbf{Y}$ with marginals $\mathbf{p}$ and $\mathbf{q}$. It turns out that we may obtain concentration results by giving an upper bound on $d_H(\mathbf{p},\mathbf{q})$ when the distribution $\mathbf{q}$ is a product distribution (that is, corresponds to independent random variables).

For the key lemma, we need one last definition. The informational divergence of p with respect to q is

$$D(p\|q) = \sum_{x\in\Omega} p_x \ln(p_x/q_x).$$

Lemma 4.33 *If q is a product distribution, then*

$$d_H(p,q)^2 \le (n/2)D(p\|q).$$

Using this information theoretic lemma we shall prove the following elegant symmetrical inequality, closely related to Theorem 3.5. Recall that the Hamming distance $d_H(A,B)$ between two subsets A and B of Ω is the minimum value of $d_H(x,y)$ over all choices of $x \in A$ and $y \in B$.

Theorem 4.34. *Let q be a product distribution. Then*

$$d_H(A,B) \le \left(\frac{n}{2}\ln\frac{1}{q(A)}\right)^{\frac{1}{2}} + \left(\frac{n}{2}\ln\frac{1}{q(B)}\right)^{\frac{1}{2}}.$$

Proof. Let p denote the distribution with $p_x = q_x/q(A)$ for $x \in A$ and $p_x = 0$ otherwise, and define the distribution r similarly corresponding to B. Then

$$\begin{aligned} D(p\|q) &= \sum_{x\in\Omega} p_x \ln(p_x/q_x) \\ &= \sum_{x\in\Omega} p_x \ln(1/q(A)) \\ &\le \ln(1/q(A)). \end{aligned}$$

Similarly, $D(r\|q) \le \ln(1/q(B))$. Next we use the observation that, since $d_H(p,r)$ is the expected Hamming distance between certain random points in A and in B, it must be at least the minimum value $d_H(A,B)$. Hence by the triangle inequality and the above lemma

$$\begin{aligned} d_H(A,B) &\le d_H(p,r) \\ &\le d_H(p,q) + d_H(q,r) \\ &\le \left(\frac{n}{2}\ln\frac{1}{q(A)}\right)^{\frac{1}{2}} + \left(\frac{n}{2}\ln\frac{1}{q(B)}\right)^{\frac{1}{2}}, \end{aligned}$$

as required. □

Finally let us see that Theorem 3.5 follows directly from the last result. Let $t > 0$ and let $B = \Omega \setminus A_t$, the complement of the t-fattening of A – see the sentence immediately after Theorem 3.6. We shall take $q(A)$ to be $\Pr(X \in A)$ in the notation there. Since $d_H(A,B) > t$, by Theorem 4.34 above we have

$$\left(\frac{n}{2}\ln\frac{1}{q(B)}\right)^{\frac{1}{2}} \ge t - t_0,$$

where

$$t_0 = \left(\frac{n}{2}\ln\frac{1}{q(A)}\right)^{\frac{1}{2}},$$

and so

$$\Pr(d_H(X,A) \le t) = q(A_t) \ge 1 - e^{-2(t-t_0)^2/n}.$$

But this is exactly the inequality (3.1) in the proof of Theorem 3.5 and so the theorem follows.

Acknowledgement. I am pleased to acknowledge very helpful comments from the referee.

References

1. Ahlswede R., Gács P. and Körner J. (1976) Bounds on conditional probabilities with applications in multi-user communication. Z. Wahrscheinlichkeitstheorie verw. Geb. 34 157 – 173. (Correction (1977) 39, 353 – 354.)
2. Alon N., Kim J.H. and Spencer J. (1997) Nearly perfect matchings in regular simple hypergraphs, Israel J. Math. 100, 171 – 187.
3. Alon N. and Spencer J. (1992) The Probabilistic Method. John Wiley & Sons.
4. Angluin D. and Valiant L. (1979) Fast probabilistic algorithms for Hamiltonian circuits and matchings, J. Computer and System Sciences 18, 155 – 193.
5. Avram F. and Bertsimas D. (1993) The minimum spanning tree constant in geometrical probability and under the independent model: a unified approach. Annals of Applied Probability 3, 113 – 130.
6. Azuma K. (1967) Weighted sums of certain dependent random variables. Tôhoku Math. J. 19, 357 – 367.
7. Bennett G. (1962) Probability inequalities for the sum of independent random variables. J. Amer. Statist. Assoc. 57 33 – 45.
8. Beveridge A., Frieze A. and McDiarmid C. (1998) Random minimum length spanning trees in regular graphs, Combinatorica, to appear.
9. Bollobás B. (1985) Random Graphs. Academic Press.
10. Bollobás B. (1987) Martingales, isoperimetric inequalities and random graphs, Colloq. Math. Soc. János Bolyai 52, 113 – 139.
11. Bollobás B. (1988) The chromatic number of random graphs. Combinatorica 8, 49 – 55.

12. Bollobás B. (1988). Sharp concentration of measure phenomena in the theory of random graphs, in Random Graphs '87 (M. Karoński, J. Jaworski and A. Ruciński editors), John Wiley and Sons, 1 – 15.
13. Bollobás B. and Brightwell G. (1992). The height of a random partial order: concentration of measure. Ann. Appl. Probab. 2, 1009 – 1018.
14. Chernoff H. (1952). A measure of the asymptotic efficiency for tests of a hypothesis based on the sum of observations, Ann. Math. Statist. 23, 493 – 509.
15. Chvátal V. (1979). The tail of the hypergeometric distribution, Discrete Mathematics 25, 285-287.
16. Coffman E.G. and Lueker G.S. (1991). Probabilistic Analysis of Packing and Partitioning Algorithms, Wiley, New York.
17. Csiszár I. and Körner J. (1981). Information Theory: Coding Theorems for Discrete Memoryless Systems, Academic Press, New York.
18. Dembo A. (1997). Information inequalities and concentration of measure, Ann. Probab. 25, 927 – 939.
19. Dembo A. and Zeitouni O. (1993). Large Deviation Techniques, Jones and Bartlett.
20. Durrett R. (1996). Probability: Theory and Examples, Second edition, Duxbury Press.
21. Freedman D.A. (1975). On tail probabilities for martingales, Ann. Probab. 3, 100 – 118.
22. Feller W. (1968). An Introduction to Probability Theory and its Applications, Volume I, Third Edition, John Wiley & Sons, New York.
23. Frieze A.M. (1985). On the value of a random minimum spanning tree problem, Discrete Applied Mathematics 10, 47 – 56.
24. Frieze A.M. and McDiarmid C.J.H. (1989). On random minimum length spanning trees, Combinatorica 9, 363 – 374.
25. Frieze A.M. and McDiarmid C.J.H. (1997). Algorithmic theory of random graphs, Random Structures and Algorithms 10, 5 – 42.
26. Grable D.A. (1998). A large deviation inequality for functions of independent, multi-way choices, Combinatorics, Probability and Computing 7, 57 – 63.
27. Grable D.A. and Panconesi A. (1997). Nearly optimal distributed edge colouring in $O(\log\log n)$ rounds, Random Structures and Algorithms 10, 385 – 405.
28. Grimmett G.R. and Stirzaker D.R. (1992). Probability and Random Processes, Second edition, Oxford University Press.
29. Hoeffding W.J. (1963). Probability inequalities for sums of bounded random variables, J. Amer. Statist. Assoc. 58, 13-30.
30. Janson S. (1995). The minimal spanning tree in a complete graph and a functional limit theorem for trees in a random graph, Random Structures and Algorithms 7, 337 – 355.
31. Johnson W. and Schechtman G. (1991). Remarks on Talagrand's deviation inequality for Rademacher's functions, Lecture Notes in Mathematics 1470, Springer-Verlag, 72 – 77.
32. Kahn J. (1996). Asymptotically good list colorings, J. Combinatorial Theory A 73, 1 – 59.
33. Kamath A., Motwani R., Palem K. and Spirakis P. (1995). Tail bounds for occupancy and the satisfiability threshold conjecture, Random Structures and Algorithms 7, 59 – 80.
34. Kim J.H. (1995). On Brooks' theorem for sparse graphs, Combinatorics, Probability and Computing 4, 97 – 132.
35. Kim J.H. (1995). The Ramsey number $R(3,t)$ has order of magnitude $t^2/\log t$, Random Structures and Algorithms 7, 173 – 207.
36. Knuth D.E. (1973). The Art of Computer Programming Volume 3: Sorting and Searching, Addison-Wesley.
37. Leader I. (1991). Discrete isoperimetric inequalities, Proc. Symposia Appl. Math. 44, 57 – 80.
38. Ledoux M. and Talagrand M. (1991). Probability in Banach Spaces, Springer-Verlag.
39. Liptser R. Sh. and Shiryaev A.N. (1989). Theory of Martingales, Kluwer, Dordrecht.
40. Marton K. (1986). A simple proof of the blowing-up lemma, IEEE Transactions on Information Theory 32, 445 – 446.
41. Marton K. (1996). Bounding $\bar{d}$-distance by informational divergence: a method to prove measure concentration, Ann. Probab. 24, 857 – 866.
42. Marton K. (1996). A measure concentration inequality for contracting Markov chains, Geometric and Functional Analysis 6, 556 – 571. (Erratum (1997) 7, 609 – 613.)
43. Marton K. and Shields P.C. (1994). The positive divergence and blowing-up properties, Israel J. Math. 86, 331 – 348.
44. Maurey B. (1979). Construction de suites symétriques, Compt. Rend. Acad. Sci. Paris 288, 679 – 681.
45. McDiarmid C. (1989). On the method of bounded differences, in Surveys in Combinatorics, ed J. Siemons, London Mathematical Society Lecture Note Series 141, Cambridge University Press.
46. McDiarmid C. (1990). On the chromatic number of random graphs, Random Structures and Algorithms 1, 435 – 442.
47. McDiarmid C. (1997). Centering sequences with bounded differences, Combinatorics, Probability and Computing 6, 79 – 86.
48. McDiarmid C. (1998). Concentration for minimum spanning tree lengths, manuscript.
49. McDiarmid C. and Hayward R. (1996). Large deviations for quicksort, J. Algorithms 21, 476 – 507.
50. Milman V. and Schechtman G. (1986). Asymptotic theory of finite dimensional normed spaces, Lecture Notes in Math. 1200, Springer-Verlag.
51. Motwani R. and Raghavan P. (1995). Randomized Algorithms, Cambridge University Press.
52. Penrose M. (1998). Random minimal spanning tree and percolation on the n-cube, Random Structures and Algorithms 12, 63 – 82.
53. Platzman L.K. and Bartholdi J.J. (1989). Spacefilling curves and the planar travelling salesman problem, J. Assoc. Comput. Mach. 36, 719 – 737.
54. Rhee W.T. and Talagrand M. (1987). Martingale inequalities and NP-complete problems, Math. Oper. Res. 12, 177 – 181.
55. Rhee W.T. and Talagrand M. (1989). Martingale inequalities, interpolation and NP-complete problems, Math. Oper. Res. 14, 189 – 202.
56. Rhee W.T. and Talagrand M. (1989). A sharp deviation for the stochastic traveling salesman problem, Ann. Probab. 17, 1 – 8.
57. Ross S.M. (1996). Stochastic Processes, Second edition, Wiley.
58. Schmidt J., Siegel A. and Srinivasan A. (1995). Chernoff-Hoeffding bounds for applications with limited independence, SIAM J. Discrete Math. 8, 223 – 250.
59. Sedgewick R. and Flajolet P. (1996). Analysis of Algorithms, Addison-Wesley.
60. Shamir E. and Spencer J. (1987). Sharp concentration of the chromatic number on random graphs $G_{n,p}$, Combinatorica 7, 121 – 129.
61. Shiryaev A.N. (1996). Probability, Second edition, Graduate Texts in Mathematics 95, Springer.

62. Steele J.M. (1995): Variations on the long increasing subsequence theme of Erdős and Szekeres. In Discrete Probability and Algorithms, D. Aldous, P. Diaconis and J.M. Steele, eds., Volumes in Mathematics and its Applications 72, Springer-Verlag, New York 111 – 131.
63. Steele J.M. (1997): Probability Theory and Combinatorial Optimization, SIAM CBMS 69.
64. Steiger W.L. (1967): Some Kolmogoroff-type inequalities for bounded random variables, Biometrika 54, 641 – 647.
65. Steiger W.L. (1969): A best possible Kolmogoroff-type inequality for martingales and a characteristic property, Ann. Math. Statist. 40, 764 – 769.
66. Steiger W.L. (1970): Bernstein's inequality for martingales, Z. Wahrscheinlichkeitstheorie verw. Geb. 16, 104 – 106.
67. Talagrand M. (1991): A new isoperimetric inequality for product measure and the tails of sums of independent random variables, Geometric and Functional Analysis 1, 211 – 223.
68. Talagrand M. (1995): Concentration of measure and isoperimetric inequalities in product spaces, Publ. Math. Institut des Hautes Études Scientifiques 81, 73 – 205.
69. Talagrand M. (1996): A new look at independence, Annals of Probability 24, 1 – 34.
70. Talagrand M. (1996): New concentration inequalities in product spaces, Invent. Math. 126, 505 – 563.
71. Talagrand M. (1996): Transportation cost for Gaussian and other product measures, Geometric and Functional Analysis 6, 587 – 600.
72. Williams D. (1991): Probability with Martingales, Cambridge University Press.

Branching Processes and Their Applications in the Analysis of Tree Structures and Tree Algorithms

Luc Devroye

School of Computer Science, McGill University, Montreal, Canada

Summary. We give a partial overview of some results from the rich theory of branching processes and illustrate their use in the probabilistic analysis of algorithms and data structures. The branching processes we discuss include the Galton-Watson process, the branching random walk, the Crump-Mode-Jagers process, and conditional branching processes. The applications include the analysis of the height of random binary search trees, random m-ary search trees, quadtrees, union-find trees, uniform random recursive trees and plane oriented recursive trees. All these trees have heights that grow logarithmically in the size of the tree. A different behavior is observed for the combinatorial models of trees, where one considers the uniform distribution over all trees in a certain family of trees. In many cases, such trees are distributed like trees in a Galton-Watson process conditioned on the tree size. This fact allows us to review Cayley trees (random labeled free trees), random binary trees, random unary-binary trees, random oriented plane trees, and indeed many other species of uniform trees. We also review a combinatorial optimization problem first suggested by Karp and Pearl. The analysis there is particularly beautiful and shows the flexibility of even the simplest branching process.

1 Branching Processes

1.1 Branching Processes

Around 1873, Galton and Watson came up with a model for explaining the disappearance of certain family names in England (see the historical survey by Kendall 1966). Their model, now known as the Galton-Watson process, is extremely simple: in a population, we begin with one node (individual, or root). The root has Z_1 children, where Z_1 has a fixed distribution (the reproduction distribution). It is convenient to let Z denote a prototype random variable with this distribution, and to set

$$p_i = \mathbf{P}\{Z = i\}, \ i \geq 0 .$$

Each child in turn reproduces independently according to the same distribution, and so forth. This leads to a random tree, the Galton-Watson tree, and a random process, the Galton-Watson process. Let Z_n denote the number of particles in the nth generation, with $Z_0 = 1$. Only one of two possible situations can occur: either the population survives forever ($Z_n > 0$ for all n) or it becomes extinct after a finite time. To analyze the Galton–Watson process it is convenient to use the RGF (the reproduction generating function), or simply generating function

$$f(s) = \sum_{k=0}^{\infty} p_k s^k = E(s^{Z_1}), \quad s \in [0,1].$$

This is a function of s that contains exactly the same information as the vector $(p_0, p_1, \dots)$. It is strictly convex (if $p_0 \neq 1$) and increases from p_0 at $s = 0$ to 1 at $s = 1$. Different RGFs define different Galton–Watson branching processes. Intuitively, it should be clear that a population explodes if the expected number of children per particle is greater than one and that it is bound to extinct if it is less than one. An important parameter thus is the expected number of children (or Malthusian parameter)

$$m = E(X) = E(Z_1) = \sum_{k=0}^{\infty} kp_k = f'(1).$$

We will prove that this intuition is partly correct. In fact, whether a population explodes or becomes extinct depends solely on the value of m, and not on the individual probabilities of the RGF. Consider the RGF for Z_n, the size of the nth generation

$$f_n(s) \overset{\text{def}}{=} E(s^{Z_n}), \quad 0 \le s \le 1.$$

With this notation, we clearly have $f_1(s) \equiv f(s)$, and $f_0(s) = s$. Conditional expectations help us in relating f_n to f. To this end, let Z_{n-1} be the number of particles in generation $n-1$. These have offspring of sizes $X_n(1), \dots, X_n(Z_{n-1})$, and these form an independently identically distributed (i.i.d.) sequence distributed as Z_1 (i.e., all the $X_n(j)$ have the same distribution as Z_1 and the choices of the $X_n(j)$ are made independently). Therefore

$$\begin{aligned} f_n(s) &= E(E(s^{Z_n} | Z_{n-1})) \\ &= E(E(s^{X_n(1)+\cdots+X_n(Z_{n-1})} | Z_{n-1})) \\ &= E\left(\prod_{j=1}^{Z_{n-1}} E(s^{X_n(j)} | Z_{n-1})\right) \quad \text{(by independence)} \\ &= E\left(\prod_{j=1}^{Z_{n-1}} E(s^{Z_1})\right) \quad \text{(identical distributions)} \\ &= E((f(s))^{Z_{n-1}}) \\ &= f_{n-1}(f(s)) \\ &= \cdots \\ &= \overbrace{f(f(\cdots f(s)))}^{n \text{ times}} .\end{aligned}$$

When $m \le 1$, the graph of $f(s)$ lies above s and $f(s) = s$ only at $s = 1$. It is not difficult to see that $f_n(s) \to 1$ for any s. In particular $f_n(0) = \Pr(Z_n = 0) \to 1$. When $m > 1$, there is a unique solution q of $f(s) = s$ that is less than one. See the figure above.

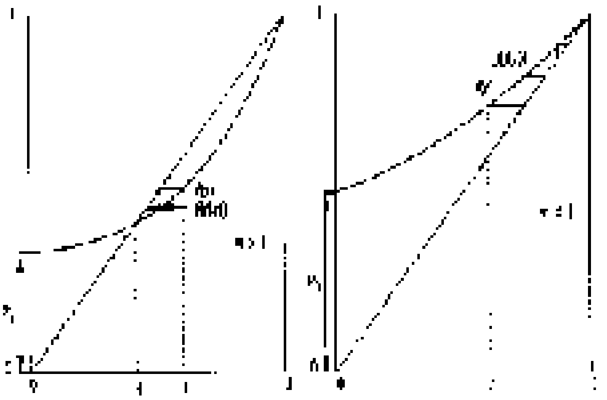


Fig. 1.1. The two possible behaviors

It is easy to see that for any $s \in [0,1)$, $f_n(s) \to q$. In particular, $\Pr(Z_n = 0) \to q$.

We now show that q is the probability that the process becomes extinct. The point I am making here is subtle, but important, as the event 'extinction' relates to the entire history of the process, not a particular n. Note the following:

$$\begin{aligned} \Pr(\text{extinction}) &= \Pr(Z_n = 0 \text{ for some } n) \\ &= \Pr(\cup_{n=1}^{\infty} \{Z_n = 0\}) \\ &= \lim_{n\to\infty} \Pr(\cup_{k=1}^{n} \{Z_k = 0\}) \\ &= \lim_{n\to\infty} \Pr(Z_n = 0) \\ &= q .\end{aligned}$$

Therefore, q is the extinction probability. We have thus shown the fundamental property of Galton–Watson processes.

Theorem 1.1. *In a Galton–Watson process, if $m > 1$, then*

$$q = \Pr(Z_n = 0 \text{ for some } n) = \lim_{n\to\infty} \Pr(Z_n = 0) < 1 .$$

When $m \le 1$, the process becomes extinct with probability one, unless we have the degenerate case $p_1 = 1$, in which case every generation contains one particle.

Processes are called supercritical, critical and subcritical when $m > 1$, $m = 1$ and $m < 1$ respectively. We also introduce the hypercritical processes which have $m = \infty$, and the exploding processes (which may be of any of the four types above) which have $\mathbf{E}(Z_1 \log Z_1) = \infty$. These last two terms are nonstandard, but will be convenient to work with. It is worth noting that in all cases,

$$\mathbf{E}(Z_n) = (\mathbf{E}(Z_1))^n = m^n$$

(by induction and conditioning, as $\mathbf{E}(Z_n | Z_{n-1}) = mZ_{n-1}$). In the critical case, the expected size of the population remains constant, while the population becomes extinct with probability one!

1.2 Some Limit Results

Theorem 1.2. *Assume that $p_1 < 1$. For a Galton-Watson branching process, $\mathbf{Pr}\{\lim_{n\to\infty} Z_n \in \{0, \infty\}\} = 1$.*

Proof. Clearly,

$$\mathbf{Pr}\{\lim_{n\to\infty} Z_n \notin \{0, \infty\}\} \le \sum_{k=1}^{\infty} \mathbf{Pr}\{Z_n = k \text{ infinitely often}\}$$

and this is zero if every term is zero. Thus, it suffices to show that for every finite k,

$$\mathbf{Pr}\{Z_n = k \text{ infinitely often}\} = 0 .$$

We say that the population is in state k if $Z_n = k$. Let r_k be the probability that the population returns to state k given that we are in state k now, so that $1 - r_k$ is the probability that we never visit it again ($Z_j \neq k$ for all $j > n$). If $p_0 = 0$, then

$$r_k \le \mathbf{Pr}\{Z_1 = k | Z_0 = k\} = p_1^k < 1 .$$

If $p_0 > 0$, then

$$r_k \le \mathbf{Pr}\{Z_1 > 0 | Z_0 = k\} = 1 - p_0^k < 1 .$$

Therefore, $r_k < 1$.

If N is the number of visits to state k, then

$$\mathbf{Pr}\{N \ge n\} \le r_k^{n-1}$$

because we need to have at least $n - 1$ transitions from state k to state k for the process driven by the transition probability r_k. Note that

$$\mathbf{E}(N) = \sum_{n=1}^{\infty} \mathbf{Pr}\{N \ge n\} \le \sum_{n=0}^{\infty} r_k^n = \frac{1}{1 - r_k} .$$

Take M arbitrary. Finally,

$$\begin{aligned} \mathbf{Pr}\{Z_n = k \text{ infinitely often}\} &\le \mathbf{Pr}\{N \ge M\} \\ &\le \frac{\mathbf{E}(N)}{M} \\ &\le \frac{1}{M(1 - r_k)} , \end{aligned}$$

which is as small as desired by our choice of M. We conclude that

$$\mathbf{Pr}\{Z_n = k \text{ infinitely often}\} = 0 .$$

□

Theorem 1.2, which is valid for any $m \in [0, \infty]$, shows that it is impossible to have oscillating populations, that is, populations in which the size drops below some finite level infinitely often when $m > 1$: in fact, with probability one, the limit of Z_n is zero or infinity. The remainder of this section is more advanced and rather technical. It can be skipped without harm (except for the definition of convergence in distribution and the statement of Lotka's lemma, which can be referred to when and if required).

We can improve on Theorem 1.2 by noting that Z_n behaves roughly speaking as m^n (recall that $\mathbf{E}(Z_n) = m^n$), and its behavior is best captured in Doob's limit law.

Theorem 1.3. (Doob's limit law) *Let m be finite. The random variables $W_n = Z_n/m^n$ form a martingale sequence with $\mathbf{E}(W_n) \equiv 1$, and $W_n \to W$ almost surely as $n \to \infty$, where W is a nonnegative random variable.*

For readers not familiar with martingales, we refer to the chapter on concentration inequalities by McDiarmid in the present volume.

We use the symbol $\xrightarrow{\mathcal{L}}$ for convergence in distribution. For random variables $(X_n)_n$ and X, and a distribution function F, we say that $X_n \xrightarrow{\mathcal{L}} X$ or $X_n \xrightarrow{\mathcal{L}} F$ when for all $x \in \mathbb{R}$ at which $F(x) = \mathbf{Pr}\{X \le x\}$ is continuous, $\mathbf{Pr}\{X_n \le x\} \to F(x)$.

While we don't know the limit distribution of W_n in general, we know a few things: in the case $m \le 1$, $p_1 < 1$, we have $\mathbf{Pr}\{W = 0\} = 1$, an uninteresting case. If $m > 1$ and $\sigma^2 = \mathrm{var}(Z_1) < \infty$, then $\mathbf{Pr}\{W = 0\} = q$, $\mathbf{E}(W) = 1$, $\mathrm{var}(W) = \sigma^2/(m^2 - m)$, and $\mathbf{E}(W_n - W)^2 \to 0$. In fact, the second moment condition on Z_1 is too strict, as the following result shows.

Theorem 1.4. (Kesten-Stigum theorem, 1966) *For a supercritical Galton-Watson process, the following properties are equivalent:*

A. $\lim_{n\to\infty} \mathbf{E}(|W_n - W|) = 0$;

B. $\mathbf{E}(Z \log(1+Z)) < \infty$;

C. $\mathbf{E}(W) = 1$;

D. $\mathbf{Pr}(W = 0) = q$.

When $m > 1$, then the above results imply

$$\frac{\log Z_n}{n} \to \log m$$

almost surely on non-extinction. Note that in general, by Fatou's lemma (which in a special form states that for positive sequences of functions f_n with $\liminf_{n\to\infty} f_n = f$ that $\liminf_{n\to\infty} \int f_n \geq \int f$) we have (as expected values are just integrals)

$$\mathbf{E}(W) \leq \liminf_{n\to\infty} \mathbf{E}(W_n) = 1$$

but we cannot conclude that $\mathbf{E}(W) = 1$. Indeed, when $m \leq 1$ and $p_1 < 1$, $W = 0$ almost surely, and when $m > 1$, there exist distributions for Z for which $W = 0$ almost surely as well. In the critical case, $Z_n \to 0$ almost surely, so finer results are needed.

We can recast the extinction problem by studying the branching process conditional on survival at time n ($Z_n > 0$). Some results for the critical case are provided in the following theorem.

Theorem 1.5. [Kesten, Ney and Spitzer, 1966] *Assume that $m = 1$ and $\sigma^2 = \mathrm{var}(Z) \leq \infty$. Let E be an exponentially distributed random variable (that is, a random variable with density e^{-x} on $[0,\infty)$). Then*

$$\lim_{n\to\infty} n\mathbf{Pr}(Z_n > 0) = \frac{2}{\sigma^2} .$$

Furthermore, if $\sigma^2 < \infty$, $Z_n/n \xrightarrow{\mathcal{L}} \sigma^2 E/2$, where Z_n is distributed as Z_n given $Z_n > 0$. If $\sigma^2 = \infty$, then $Z_n/n \to \infty$ in probability, and $\lim_{n\to\infty} n\mathbf{Pr}(Z_n > 0) = 0$.

Under the stronger condition $\mathbf{E}(Z^3) < \infty$, the theorem above is referred to as the Kolmogorov-Yaglom theorem after Kolmogorov (1938) and Yaglom (1947). The conditional random variable Z_n is also useful to understand subcritical branching processes. The main results in this respect are again provided by Yaglom (1947) and Heathcote, Seneta and Vere-Jones (1967) (see also Asmussen and Hering, 1983 and Lyons, 1997).

Theorem 1.6. [Yaglom-Heathcote-Seneta-Vere-Jones theorem] *If $m < 1$, then $Z_n \xrightarrow{\mathcal{L}} Y$, where $\mathbf{Pr}(Y < \infty) = 1$. Furthermore, $\mathbf{Pr}(Z_n > 0)/m^n$ is nonincreasing (for any m). Finally, the following properties are equivalent:*

A. $\lim_{n\to\infty} \mathbf{Pr}(Z_n > 0)/m^n > 0$;

B. $\sup_n \mathbf{E}(Z_n) = \sup_n \mathbf{E}(Z_n | Z_n > 0) < \infty$;

C. $\mathbf{E}(Z \log(Z+1)) < \infty$.

Proof. We will not give a complete proof here. However, it is worthwhile to note Lyons' proof of the equivalence of A and B. We know that for any m,

$$\mathbf{Pr}(Z_n > 0) = \frac{\mathbf{E}(Z_n)}{\mathbf{E}(Z_n | Z_n > 0)} = \frac{m^n}{\mathbf{E}(Z_n)}$$

Thus $\mathbf{Pr}(Z_n > 0)/m^n \downarrow$ iff $\mathbf{E}(Z_n) \uparrow$. Thus A is equivalent to B if we can prove that $\mathbf{E}(Z_n) \uparrow$. Let Y_n be the size of the n-th generation in the subtree rooted at the leftmost child of the root with a descendant in the n-th generation, and let I_n be the index of this child (counted from left to right). Then as $Z_n \geq Y_n$, for any $k \geq 1$,

$$\begin{aligned}\mathbf{Pr}(Z_n \geq k | Z_n > 0) &\geq \mathbf{Pr}(Y_n \geq k | Z_n > 0) \\ &= \sum_j \mathbf{Pr}(Y_n \geq k, I_n = j | Z_n > 0) \\ &= \sum_j \mathbf{Pr}(Y_n \geq k | I_n = j, Z_n > 0)\mathbf{Pr}(I_n = j | Z_n > 0) \\ &= \sum_j \mathbf{Pr}(Z_{n-1} \geq k | Z_{n-1} > 0)\mathbf{Pr}(I_n = j | Z_n > 0) \\ &= \mathbf{Pr}(Z_{n-1} \geq k | Z_{n-1} > 0) .\end{aligned}$$

∎

1.5 Bibliographic Remarks

For an account of the theory of branching processes, see Athreya and Ney (1972), Grimmett and Stirzaker (1992), Harris (1963), Jagers (1975), or Asmussen and Hering (1983). Kendall (1966) gives an enjoyable historical overview. Neveu (1986) provides a rigorous background for studying random trees in general and Galton-Watson trees in particular. A unified proof of the Kesten-Stigum, Kolmogorov-Yaglom and Heathcote-Seneta-Vere-Jones theorems based on Galton-Watson processes with immigration and/or trees with distinguished paths may be found in Lyons, Pemantle and Peres (1995, 1996). In these papers, size-biased trees are introduced that scale probabilities of events in the n-th generation by Z_n/m^n, which turns out be equivalent to looking at $\lim_{n\to\infty} \mathbf{Pr}(\cdot | Z_n > 0)$. The idea of size biasing is also due to Hawkes (1981) and Joffe and Waugh (1982).

For critical processes, Weiner (1984) showed that there exist positive constants $c \le C$ such that $\text{Bmax}_{i \le n} Z_i \in [c \log n, C \log n]$ and $\text{argmax}_{i \le n} Z_i \in [cn, Cn]$.

For a supercritical process, Heyde (1970) shows that if Z has a finite variance σ^2, and $Z_n/m^n \to W$ almost surely, then $(W - Z_n/m^n)m^{n/2}$ converges in distribution to a random variable Y. Thus, Z_n/m^n is rather concentrated around W. Conditional on $W > 0$,

$$\frac{m^{n}(W - W_n)\sqrt{m^2 - m}}{\sqrt{Z_n}\sigma} \xrightarrow{\mathcal{L}} \mathcal{N},$$

where $\mathcal{N}$ denotes the normal distribution (Heyde, 1971). A Berry-Esseen type inequality to quantify this convergence is given by Heyde and Brown (1971). Again on the set where $W > 0$, we have almost surely,

$$\limsup_{n\to\infty} \frac{m^n W - Z_n}{\sqrt{2\sigma^2(m^2 - m)^{-1} Z_n \log n}} = 1$$

and a similar statement for the liminf obtained with 1 replaced by -1 on the right-hand side.

The tail behavior of W was investigated by Athreya (1994), who shows faster than exponential drop-offs. For finite m, quasi-exponential tail inequalities for $\Pr(Z_n > cE(Z_n))$ and $\Pr(Z_n < E(Z_n)/c)$ for large c were derived by Karp and Zhang (1995). See also Biggins and Bingham (1993) about the description of W.

Darling (1970) examines the behavior when Z has very large tails, so that, in fact, $\log(Z_n + 1)/b^n$ tends to a limit law for some $b > 1$. Thus, Z_n increases at a doubly exponentially quickly. This sort of transformation is necessary, because as shown by Seneta (1969), if $m = \infty$, then no constants c_n can exist such that Z_n/c_n converges in distribution to a non-degenerate random variable.

2. Search Trees

2.1 Height of the Random Binary Search Tree

A binary search tree for distinct real numbers $x_1, \ldots, x_n$ is a binary tree in which x_1 is the root, whose left subtree is a binary search tree for $\{x_2, \ldots, x_n\} \cap (-\infty, x_1)$ and whose right subtree is a binary search tree for $\{x_2, \ldots, x_n\} \cap (x_1, \infty)$ (thus the structure of the search tree depends heavily on the order in which the reals are presented). If the left subtree has k points (nodes), then the rank of the root in the total ordering of the x_i's is $k + 1$. We can grow the tree incrementally: if x_{n+1} is to be added (inserted), we start at the root and recursively find the subtree to which x_{n+1} must belong by comparing x_{n+1} to the current root and choosing the left or right subtree as appropriate. Eventually, we locate an empty subtree, which is then formally replaced by a one-node subtree having x_{n+1} as its root. The insertion time is equal to the distance in the tree (path length) between the root (x_1) and the inserted node (x_{n+1}); this distance is referred to as the depth of x_{n+1}. The *height* of a binary search tree is the maximal depth of a node, and it measures the worst-case insertion time, an important quantity if we are to maintain a binary search tree when new data arrive.

By a random binary search tree, we mean a binary search tree on a set of random variables $\{x_1, \ldots, x_n\}$ which is obtained by taking a permutation of $\{1, \ldots, n\}$ with each permutation equally probable. It is easy to see that the structure of the tree we obtain will be the same if we pick the x_i independently at random from the same distribution F provided the probability that we choose the same number twice in n trials under F is zero, e.g. if the x_i are uniformly chosen elements of $[0,1]$. The depth D_n of the last node to be inserted satisfies $E(D_n) \sim 2\log n$ (Lynch, 1965; Knuth, 1973). Further $(D_n - 2\log n)/\sqrt{2\log n} \xrightarrow{\mathcal{L}} \mathcal{N}(0,1)$ (Mahmoud and Pittel, 1984; Devroye, 1988). For the height H_n, the maximal path distance between any node and the root, Robson (1979) showed that for all $\epsilon > 0$,

$$\lim_{n\to\infty} \Pr(H_n \ge (\gamma + \epsilon)\log n) = 0,$$

where $\gamma = 4.31107\ldots$ is the unique solution greater than 2 of the equation $c\log(2e/c) = 1$. To actually show that $H_n/\log n \to \gamma$ in probability (we recall that $X_n \to c$ in probability means that for any positive ϵ, $\lim_{n\to\infty}\Pr(|X_n - c| > \epsilon) = 0$), branching processes were the first successful methodology (Devroye, 1986, 1987). Drmota (1997) was the first to prove this result by generating function analysis. The theorem below will be considerably generalized further on in the chapter.

Theorem 2.1. [Devroye, 1986, 1987] *In a random binary search tree on n nodes, $H_n/\log n \to \gamma = 4.31107\ldots$ in probability.*

Proof. We briefly show here that the height can be studied with the aid of Galton-Watson branching processes. To make the connection, we introduce a new representation of a binary search tree. Call the (random) binary search tree T. Augment the tree T by associating with each node the size of the subtree rooted at that node, and call the augmented tree T'. The root of T' has value n. Since the rank of the root element of T is equally likely to

$i = 1, \ldots, n$, the number N of nodes in the left subtree of the root of T is uniformly distributed on $\{0, 1, \ldots, n-1\}$. A moment's thought shows we can choose U by setting $N = \lfloor nU \rfloor$, where U is uniformly distributed on $[0,1]$. Also, the size of the right subtree of the root of T is $n - 1 - N$, which is distributed as $\lfloor n(1-U) \rfloor$. All subsequent splits can be represented similarly by introducing independent uniform [0,1] random variables. This is a splitting argument: we have described a new fictitious collection of random variables $U_1, U_2, \ldots$, and we can derive all the values of nodes in T' from it. This in turn determines (the shape of) T. More precisely, the rule is simple: Make T' an infinite binary tree, give the root the value n. Also, associate with each node an independent copy of U. If a node has value V and its assigned copy of U is U' (say), then the values of the two children of the node are $\lfloor VU' \rfloor$ and $\lfloor V(1-U') \rfloor$, respectively. Thus, the value of any node at distance k from the root of T' is distributed as

$$\lfloor \cdots \lfloor \lfloor nU_1 \rfloor U_2 \rfloor \cdots U_k \rfloor ,$$

where $U_1, \ldots, U_k$ are i.i.d. uniform [0,1]. We have just described a second way of generating a random tree with exactly the same distribution as a random binary search tree. This second method of generating the tree is much more amenable to analysis.

The above representation has a myriad of applications. One of them involves the study of the height. Let H_n be the height of T when $|T| = n$. Then $H_n \ge k$ if and only if one of the 2^k values V_i of nodes at distance k from the root of T' is at least equal to one, which we write as

$$[H_n \ge k] = \left[\max_{1 \le i \le 2^k} V_i \ge 1\right].$$

This is a beautiful duality indeed. Some care must be exercised when manipulating it though, as the V_i's are very dependent—just consider the values V_i and V_j for nodes that are near one another in the tree. To steer around this we will derive separate upper and lower bounds for H_n.

In doing so, we need to be able to analyze the distribution of the V_i, which boils down to analyzing the distribution of the product of k uniform [0,1] random variables for various k. To do so we pass to the logarithm. It turns out the logarithms we are interested in studying are drawn from a very well studied class of distributions, the Gamma distributions. To be precise, a uniform random variable is distributed as e^{-E} where E is exponentially distributed (it has density e^{-x} on $\mathbb{R}^+$) and a gamma k random variable G_k is distributed as the sum of k independent exponentials (see Grimmett and Stirzaker 1992). Thus the product of k uniforms is e^{-G_k}.

The upper bound. By the dual relationship shown above, we see that

$$\begin{aligned}
\Pr\{H_n \ge k\} &= \Pr\left\{\cup_{i=1}^{2^k} [V_i \ge 1]\right\} \\
&\le 2^k \Pr\{V_1 \ge 1\} \\
&\quad \text{(by the union bound (Bonferroni's inequality)} \\
&\quad \text{and symmetry)} \\
&\le 2^k \Pr\left\{n \textstyle\prod_{i=1}^{k} U_i \ge 1\right\} \\
&\quad (U_1, \ldots, U_k \text{ are i.i.d. uniform } [0,1]) \\
&\quad (\text{omit the } \lfloor\cdot\rfloor \text{ in the definition of } V_1) \\
&= 2^k \Pr\{n e^{-G_k} \ge 1\} \\
&\quad (G_k \text{ is a gamma } (k) \text{ random variable}) \\
&= 2^k \Pr\{G_k \le \log n\}.
\end{aligned}$$

The point now is to find the smallest k such that the upper bound tends to zero. Recall that a G_k random variable has mean k. Thus, if $k = \log n$, the upper bound is $\Theta(2^k)$, which is obviously useless. In fact, k will have to be much larger than $\log n$ in order that the effect of the 2^k term be canceled. Let us try the next best thing: $k \sim c \log n$ for some $c > 1$. The whole enterprise now focuses on the probability in the left tail of the gamma distribution. We provide the details as they explain the choice of c. Let G_k be a gamma (k) random variable. We have

$$1 \le \frac{\Pr\{G_k \le y\}}{\frac{y^k e^{-y}}{k!}} \le \frac{1}{1 - \frac{y}{k+1}},$$

where the lower bound is valid for all $y > 0$ and the upper bound is applicable when $0 < y < k+1$. In particular

$$\Pr\{G_k \le \log n\} \le \frac{(\log n)^k}{n k!} \times \frac{1}{1 - \frac{\log n}{k+1}}$$

valid for $\log n < k+1$. Thus, we have, setting $k = \lceil c \log n \rceil$, and using $k! \ge (k/e)^k$ (which follows from Stirling's formula),

$$\begin{aligned}
\Pr\{H_n \ge k\} &\le \frac{(2\log n)^k}{n k!} \times \frac{1+o(1)}{1 - \frac{1}{c}} \\
&\le n^{-1} (2e \log n / k)^k \times \frac{1+o(1)}{1 - \frac{1}{c}} \\
&\le \left(\frac{1}{e}\left(\frac{2e}{c}\right)^c\right)^{\log n} \times \frac{1+o(1)}{1 - \frac{1}{c}} \\
&\to 0
\end{aligned}$$

if $(1/e)(2e/c)^c < 1$. Let $\gamma = 4.31107\ldots$ be the only solution greater than one of

$$\left(\frac{1}{e}\right)\left(\frac{2e}{c}\right)^c = 1 .$$

We conclude that $\lim_{n\to\infty}\Pr\{H_n > c\log n\} = 0$ for all $c > \gamma$. A more careful use of Stirling's inequality shows that $\lim_{n\to\infty}\Pr\{H_n > \gamma\log n\} = 0$.

The lower bound. We know now that H_n is very likely less than $\gamma\log n$. Take $\epsilon > 0$. To show that it is more than $(\gamma - \epsilon)\log n$ with high probability, all we have to do is exhibit a path in the augmented tree with the property that at distance k from the root, the augmented value is at least one. Now, you will say, this is a piece of cake. Why don't we just follow the path dictated by the longest split, that is, when we are at a node with uniform split value U, we go left if $U > 1/2$ and right otherwise? It turns out that if we do so, the augmented value drops below 1 (at k near $c\log n$) with $c = 3.25$ only. So, this is not a good way to prove the existence of a node far from the root. Instead, we will use branching processes to show that the height is greater than $c\log n$ with probability tending to one, when $c < \gamma$. Thus, we need to track down nodes with large values in the augmented tree. For now, we define $V = nU_1U_2\cdots U_k$ for a node at distance k from the root, where the U_i's are the uniform $[0,1]$ random variables describing the splits on the path to the root. The purpose is to construct a surviving Galton-Watson process. The root of T becomes the proto-familias of the branching process. Consider all descendants in T L levels away, and declare these nodes Galton-Watson children if the product of uniform splitting random variables encountered on the path from the root to the possible child is $> d^L$ for a given constant d. The number of Galton-Watson children per node is bounded between 0 and 2^L. Clearly, all nodes in the Galton-Watson process reproduce independently according to identical reproduction distributions. If T were infinite, the corresponding Galton-Watson process would survive with probability $1 - q > 0$ if the expected number of Galton-Watson children per node were greater than one. But this expected number is

$$
\begin{aligned}
2^L\Pr\{U_1\cdots U_L > d^L\} &= 2^L\Pr\{G_L < L\log(1/d)\} \\
&\quad (G_L \text{ is a gamma } (L) \text{ random variable}) \\
&\ge \frac{(2dL\log(1/d))^L}{L!} \\
&\quad (\text{by an inequality for the tail of the gamma distribution}) \\
&\sim \frac{(2ed\log(1/d))^L}{\sqrt{2\pi L}} \\
&\quad (\text{by Stirling's approximation, as } L \to \infty) \\
&> 1
\end{aligned}
$$

for L large enough when $2ed\log(1/d) > 1$. We choose $d = e^{-1/c}$, recall that $e^{-1/c}(2e/c) > 1$ and obtain $2e^{1-1/c}/c > 1$.

So, with probability $1 - q > 0$, there exists a node at distance kL from the root with value $V \ge nd^{kL} = ne^{-kL/c}$. If we take uniform rounding into account to get the real augmented value of that node, it takes only a minute to verify by induction that it is at least equal to $V - kL$ as we can use one unit at most in every truncation. In conclusion,

$$\Pr\{H_n \ge kL\} \ge 1 - q$$

if $ne^{-kL/c} - kL \ge 1$. Take for example $kL = c'\log n - \theta L$ for $c' < c$, where $\theta \in [0,1)$ is possibly dependent upon n. Then the last condition is verified as

$$ne^{-kL/c} - kL \ge n^{1-c'/c} - c'\log n > 1$$

for all n large enough, and c' is arbitrarily close to c, which in turn is arbitrarily close to γ. We have thus shown that $\liminf_{n\to\infty}\Pr\{H_n > (\gamma - \epsilon)\log n\} \ge 1 - q$ for all $\epsilon > 0$ and some $q < 1$. But we are not finished yet! Indeed, what if $1 - q = 0.00001$? Clearly, we want the latter probability to be $1 - o(1)$. So, we take M such that M is integer-valued. The 2^M nodes at distance M from the root of T are roots of subtrees each of height kL (i.e., of height k in the Galton-Watson tree). Each of the subtrees leads to an independent run of a Galton-Watson process. If M is large enough, the probability that at least one of these processes survives is close to one. Let $a \in (0,1/2)$ be another constant, and let A be the event that the $2^M - 1$ uniform $[0,1]$ random variables associated with the top M levels of nodes take values in $[a, 1-a]$. We see that

$$\Pr\{A^c\} \le 2a \times (2^M - 1) < a2^{M+1},$$

and this is as small as desired by our choice of a. If A is true, then the augmented values V associated with the nodes at distance M from the root are all at least na^M. Let B be the event that one of the 2^M Galton-Watson processes defined with the aid of the parameters c and L, and rooted at one of the given 2^M nodes survives. From the previous discussion, using independence,

$$\Pr\{B^c\} = q^{2^M},$$

which is as close to zero as desired by choice of M. If A and B happen simultaneously, then there exists a node at distance $M + kL$ from the root whose augmented value is at least equal to

$$na^M e^{-kL/c} - (M + kL).$$

Take for example $kL = c'\log n - \theta L$ as above. Then the augmented value is at least equal to

$$a^M n^{1-c'/c} - c'\log n - M.$$

This is greater than one for n large enough. Therefore

$$\liminf_{n\to\infty}\Pr\{H_n \ge c'\log n - L + M\} \ge \Pr\{A \cap B\} \ge 1 - \Pr\{A^c\} - \Pr\{B^c\}.$$

The latter bound is as close to one as desired by the choice of a and M. Also, c' is arbitrarily close to γ. Hence, for all $\epsilon > 0$

$$\lim_{n\to\infty}\Pr\{H_n > (\gamma - \epsilon)\log n\} = 1.$$

This concludes the proof of the result that $H_n/\log n \to \gamma$ in probability. □

2.8 Quadtrees

We round off this section by showing the universality of the above methodology with the aid of quadtrees. The point quadtree in R^d (Finkel and Bentley, 1974; see Samet (1990) for a survey) generalizes the binary search tree. Each data point is a node in a tree having 2^d subtrees corresponding to the quadrants formed by considering this data point as the new origin. Insertion into point quadtrees is as for binary search trees.

We assume that a random quadtree is constructed on the basis of an i.i.d. sequence with a given distribution in the plane. If this distribution is uniform in the unit square, we call it a uniform random quadtree. In the latter case, the root is easily seen to induce a split into 4 portions of sizes approximately equal to n times the product of two independent uniform [0, 1] random variables.

The height H_n of a random quadtree has a distribution which depends upon the distribution of the data points. For this reason, we look only at uniform random quadtrees. It is easy to show that

$$\Pr(H_n \ge k) \le 2^{dk}\Pr\left(n\prod_{i=1}^{dk} U_i \ge 1\right),$$

where the U_i's are i.i.d. uniform [0, 1] random variables. We deduce that $\Pr(H_n > (c/d)\log n) \to 0$ whenever $c > \gamma$. Furthermore,

$$\Pr(H_n \ge k) \ge \Pr\left(\max_{1\le j\le 2^{dk}} nV_j > 1 + k\right)$$

where V_j is a product of independent products of two uniform [0, 1] random variables along the j-th path of length k down the quadtree (Devroye, 1987). We deduce that $\Pr(H_n < (c/d)\log n) \to 0$ whenever $c < \gamma$ by mimicking the proof of Theorem 2.1. We conclude that $H_n/\log n \to \gamma/d$ in probability. This result still requires appropriate generalization to non-uniform distributions.

2.9 Bibliographic Remarks

The use of branching processes in the study of binary search trees was introduced in Devroye (1986, 1987). A nice account of this approach can be found in Mahmoud (1992). One can also prove that $\mathbf{E}(H_n^k)/\log^k n \le \gamma^k + o(1)$ for all $n > 0$ and find a positive number δ such that

$$\lim_{n\to\infty} \Pr(H_n > \gamma\log n - \delta\log\log n) = 0.$$

By mimicking the proof of the latter fact, show that $F_n/\log n \to 0.3711\ldots$ in probability, where F_n is the fill-up level, i.e. the maximal depth at which the binary search tree truncated to that depth is complete—thus, level F_n has 2^{F_n} nodes. The constant $0.3711\ldots$ is the only solution < 1 of $(2e/c)^c(1/e) = 1$. See Devroye (1986, 1987).

3. Heuristic Search

3.1 Introduction.

In this section we present two other beautiful applications of the theory of branching processes. Both involve heuristics for finding the optimal path in a tree with random costs. The tree model studied here was first proposed and analyzed by Karp and Pearl (1983), who decided to look at the simplest possible nontrivial model so as to make the greatest didactical impact.

Consider an infinite complete binary tree in which we associate with every edge e an $0-1$ random variable X_e which is 1 with probability p and 0 with probability $1-p$. The value of a node is the sum of the values of the edges on the path from the root to that node. The object is to find the best node at distance n from the root, that is, the node of minimal value. Interestingly, for $p < 1/2$, we can discover one of the optima in $O(n)$ expected time. This is largely due to the fact that there are many more zeroes than ones in the tree, allowing us to use simple yet fast search algorithms (see section 3.2). In section 3.3, we deal with the much more difficult case $p > 1/2$. Rather than trying to reach the optimum, Karp and Pearl propose looking for a near-optimum that would be reachable in $O(n)$ expected time. The heuristic proposed by them employs bounded lookahead and backtrack search.

3.2 Depth First Search

The infinite subtree rooted at a node u is called T_u. All the nodes in this subtree that can be reached via 0-valued edges form a subtree called Z_u. The heuristic we consider here simply performs a series of depth first searches of trees Z. We can also think of 1-valued edges as blocked pipes, and 0-valued edges as open pipes. When we pour water in the root, it trickles down and reaches all the 0-valued nodes we. If we reach level n in this manner, we stop. Otherwise, we open one blocked pipe and start all over from there. During the depth first search of a given Z_u, the nodes v with the property that edge (v, w) is blocked and $w \notin Z_u$ are collected in a set B_u. Since the method consists of always going for the easiest bait, we will call it depth first search. Note that the above procedure first visits all nodes with value 0, then all nodes with value 1, and so forth. This guarantees that an optimum will be returned. The question we have to answer is how long the algorithm runs on the average.

In order to analyze this algorithm, we offer the following crucial result of Karp and Pearl (1983).

Theorem 3.1. The family tree traversal theorem. *Consider a Galton-Watson branching process with reproduction probabilities $p_0, \ldots, p_M$, where M is a deterministic bound on the number of children of a node. Consider the (possibly infinite) family tree T thus generated. Let D_n be the number of nodes encountered in the depth first search of T, stopped as soon as level n is reached. Then $\mathbf{E}(D_n) = O(n)$.*

Proof. We consider three cases. In case 1, we assume that m, the mean number of children per node, is ≤ 1. Let $Z_0, Z_1, \ldots$ denote the generation sizes in T. We bound D_n by the total size of T. We recall that

$$\mathbf{E}(Z_i) = m^i \le 1 .$$

Therefore,

$$\mathbf{E}(D_n) \le \sum_{i=0}^{n} \mathbf{E}(Z_i) = \sum_{i=0}^{n} m^i \le n + 1 .$$

In case 2, we assume that $m > 1$, yet T is finite. This corresponds to a process that becomes extinct. We introduce the notation $\mathbf{E}^*$ for the conditional expectation given that T is finite. We also introduce q, the probability of eventual extinction, and $f(s)$, the size (reproduction generating function). Once again, we bound

$$D_n \le \sum_{i=0}^{\infty} Z_i .$$

Note first that for $k \ge 0$

$$\Pr(Z_1 = k | T \text{ finite}) = \frac{\Pr(Z_1 = k)\Pr(T \text{ finite} | Z_1 = k)}{\Pr(T \text{ finite})} = \frac{p_k q^k}{q} = p_k q^{k-1} .$$

Note that

$$\mathbf{E}^*(Z_1) = \sum_{k=0}^{\infty} k p_k q^{k-1} = f'(q) .$$

Thus, the derivative of f at q tells us the expected number of children of the root of an extinct tree; note that this is less than one. But this formula should be universally valid for all generation sizes. Therefore,

$$\begin{aligned}\mathbf{E}^*(Z_i) &= \left(\overbrace{f(f(\cdots f(q)\cdots))}^{i \text{ times}}\right)' \\ &= f'\left(\overbrace{f(f(\cdots f(q))\cdots)}^{i-1 \text{ times}}\right) \times f'\left(\overbrace{f(f(\cdots f(q))\cdots)}^{i-2 \text{ times}}\right) \times \cdots \times f'(q) \\ &= (f'(q))^i .\end{aligned}$$

Thus,

$$\mathbf{E}^*(D_n) \le \sum_{i=0}^{\infty} (f'(q))^i = \frac{1}{1 - f'(q)} .$$

This concludes the proof of case 2. (Note that for supercritical Galton-Watson processes, the branching process given T finite is an unconditional branching process with GF $f(qs)/q$.) Finally, in case 3, we assume that $m > 1$ and that T is infinite. Nodes in the search are designated as mortal or immortal according to whether their subtrees are finite or not. Note that the search at a given node at worst visits all the nodes in the subtrees with mortal nodes as roots. The expected size of each such subtree is not more than $1/(1 - f'(q))$, by case 2. When the search visits the first immortal child, it will never return to visit another child, as an infinite tree is bound to have at least one node at level n. As each node has no more than M mortal children, we have the following recurrence:

$$\mathbf{E}(D_n | T \text{ infinite}) \le 1 + \mathbf{E}(D_{n-1} | T \text{ infinite}) + \frac{M}{1 - f'(q)} .$$

This recurrence leads trivially to

$$\mathbf{E}(D_n | T \text{ infinite}) \le n + (n-1)\frac{M}{1 - f'(q)} .$$

Cases 2 and 3 may be combined easily as

$$\begin{aligned}\mathbf{E}(D_n) &= \Pr(T \text{ finite})\,\mathbf{E}(D_n | T \text{ finite}) \\ &\quad + \Pr(T \text{ infinite})\,\mathbf{E}(D_n | T \text{ infinite}) \\ &\le \max(\mathbf{E}(D_n | T \text{ finite}), \mathbf{E}(D_n | T \text{ infinite})) .\end{aligned}$$

This concludes the proof of the family tree traversal theorem. □

Next, we claim that the expected running time of iterated depth first search is $O(n)$ when $p < 1/2$. A depth first search that is one iteration of this process: at a node, all the nodes in its subtree reachable via 0 valued edges are visited. We call this collection of nodes the expansion tree of the node. A node with an infinite expansion tree is called immortal. The other ones are mortal. Consider the branching process defined by zero edges only. The progeny distribution has $p_0 = (1-p)^2$ (two zero edges), $p_1 = 2p(1-p)$ and $p_2 = p^2$. The expected number of children per node is

$$m = 2(1-p)^2 + 2p(1-p) = 2(1-p) > 1 .$$

Thus, the extinction probability for this branching process is $q < 1$; q is also the probability that a given node is mortal.

The running time is conveniently decomposed as follows: any trial started at any node takes expected time bounded by cn (Theorem 3.1). Thus, the total expected time before halting is not more than the expected number of trials times cn. The total number of trials in turn is not more than the total number of trials started at mortal nodes plus one. Therefore,

$$E(\text{total time}) \le \frac{cn}{1-q},$$

since the probability of having an immortal node is $1-q$, and a search started at an immortal node surely reaches level n. This concludes the proof of the linear expected time claim.

Remark 3.1. The case $p = 1/2$. When $p = 1/2$, the given iterated depth-first-search procedure takes quadratic expected time.

We conclude this section with another analysis: what is the value C_n of the minimal node at distance n from the root? Clearly, C_n is a random variable sandwiched between 0 and n. When n grows, C_n increases as well (for a given tree). As a monotone sequence has a (possibly infinite) limit, we may call our limit C. Interestingly, when $p < 1/2$, C is finite with probability one! This means that we can find an infinite path in almost every tree with only a finite number of nonzero edges. We have the following:

A. For every k, $\Pr(C_n > k) \le \Pr(C > k)$. (Obvious, since $C_n \uparrow C$.)

B. $\lim_{n\to\infty} \Pr(C_n > k) = \Pr(C > k)$. (Thus, C really matters, as it describes the situation for all n large enough.)

C. For $p < 1/2$,

$$\Pr(C > k) \le (2p)^{2^{k+1}}, \quad k = 0, 1, 2, \dots$$

Proof. Consider a branching process in which we keep only the 0-valued edges in the complete binary tree. As the number of children per node is binomially distributed with parameters 2 and $1-p$, the expected number of children is $2(1-p) > 1$. Let q be the extinction probability. Then

$$\Pr(C > k) \le q^{2^k}$$

since $[C > k]$ implies that each of the 2^k subtrees rooted at the nodes at depth k must fail to have an infinite path of zero-cost branches (that is, each of the 2^k branching processes started at these nodes must become extinct). Since the generating function of this branching process is $f(s) = (p + (1-p)s)^2$, it is easy to see that $q \le (2p)^2$. To prove this, we need only show that $f((2p)^2) < (2p)^2$, or that

$$p + (1-p)(2p)^2 < 2p ,$$

or that $4p(1-p) < 1$. But the last inequality is obviously true. □

3.3 Bounded Lookahead and Backtrack

In the case of a majority of 1-valued edges ($p > 1/2$), depth-first search yields exponential expected time. In fact, it seems impossible to construct any kind of polynomial expected time algorithm for locating the optimal value. We can do the next best thing, that is, we can try to find an almost optimal solution. To set the stage, we first define C_n, the optimal value of a solution found by an algorithm, and C_n^*, the value of the true optimum in the random tree. Clearly, $C_n^* \le C_n$. For a given algorithm, two issues have to be dealt with:

A. What is the expected time $E(T)$ taken by the algorithm?

B. How close is C_n to C_n^* (in some probabilistic sense)?

The bounded-lookahead-and-backtrack (or BLAB) algorithm proposed by Karp and Pearl (1983) introduces three design parameters d, α and L, where $d \ge 1$ is an integer, $\alpha \in (0, 1)$ is a real number, and $L > 1$ is an integer. If u is a node in our tree and v is a descendant of u such that the path distance from u to v is L, then we say that v is an (α, L)-child of u if the sum of the edge values on the linking path is $\le \alpha L$. To make things more readable, we will simply say that v is a good child of u.

We now construct a fake branching process as follows: start with a given node and make it the root of the branching process. Declare all the good ones to be its offspring. So, the process jumps L levels at a time. (This is illustrated in the first figure of this section.) Repeat this definition for all the nodes thus obtained. The Malthusian parameter for this process is the expected number of good sons per node, or

$$m \stackrel{\text{def}}{=} 2^L \Pr(\text{Bin}(L, p) \le \alpha L) .$$

The fake branching process is supposed to help us locate near-optimal nodes at level n. If it is to work for us, we surely would like the process to survive for every, thus leading to the condition $m > 1$. From the properties of the binomial distribution, we recall that if $\alpha < p$ is fixed, then, as $L \to \infty$,

$$m \sim 2^L \frac{\Theta(1)}{\sqrt{L}} \{\phi(\alpha,p)\}^L = 2^L \frac{\Theta(1)}{\sqrt{L}} \left(\left(\frac{p}{\alpha}\right)^{\alpha} \left(\frac{1-p}{1-\alpha}\right)^{1-\alpha} \right)^L ,$$

where the function $\phi(\alpha,p)$ increases monotonically from $1-p$ at $\alpha = 0$ to 1 at $\alpha = p$. Thus, it takes the value 1/2 somewhere in the interval (0,p), at a place we will call α^*. We have the freedom to choose α and L. So, we first pick $\alpha \in (\alpha^*,1)$. Then we choose L so large that $m > 1$. This fixes the branching process. We let the probability of extinction be q. The new algorithm proceeds as follows: we select d in some way (as yet unspecified here), such that $n - d$ is a multiple of L. Repeat for each of the 2^d nodes at level d until successful the following process: traverse the "good node" branching process in a depth-first-search manner until a node is found at level n or until the subtree is exhausted without ever reaching level n. If a node at level n is reached, then its value is guaranteed to be no more than $d + \alpha(n - d)$. But the probability of a given depth-first-search succeeding is at least $1 - q$. Thus, the overall procedure returns a failure with probability less than q^{2^d}. In that case, if a node has to be returned, we might as well return the leftmost node in the tree, with value $\leq n$. Putting this together, we see that

$$\begin{aligned} E(C_n) &\leq n\Pr(\text{search fails}) + d + \alpha(n-d) \\ &\leq nq^{2^d} + d + \alpha(n-d) . \end{aligned}$$

For fixed $\epsilon > 0$, this is less than $\alpha^*(1+\epsilon)n$ by making d large, e.g., $\alpha \leq \alpha^*(1+\epsilon/2)$ will do, L is as above, and d large, but fixed. We also see that

$$\lim_{n\to\infty} \Pr(C_n > \alpha^*(1+\epsilon)n) = 0$$

for all $\epsilon > 0$ if we choose α and L as above and $d \to \infty$ while $d/n \to 0$ (example: $d \sim \log n$).

The second thing we need to prove is that $E(C^*_n) \geq \alpha^* n$ or something close to that. Note the following:

$$\begin{aligned} \Pr(C^*_n < \alpha^* n) &\leq \Pr(\text{at least one } (\alpha^* n)\text{-good node at level } n) \\ &\leq 2^n \Pr(\text{Bin}(n,p) \leq \alpha^* n) \\ &= 2^n \frac{\Theta(1)}{\sqrt{n}} \{\phi(\alpha^*,p)\}^n \\ &= \frac{\Theta(1)}{\sqrt{n}} . \end{aligned}$$

Thus $\Pr(C^*_n \geq \alpha^* n) \to 1$. Also,

$$\begin{aligned} E(C^*_n) &\geq E(C^*_n I_{C^*_n \geq \alpha^* n}) \\ &\geq \alpha^* n \Pr(C^*_n \geq \alpha^* n) \\ &\geq \alpha^* n(1 - \Theta(1)/\sqrt{n}) \\ &\geq \alpha^* n - \Theta(\sqrt{n}) . \end{aligned}$$

For given $\epsilon > 0$, we can design an algorithm that guarantees the following:

$$\limsup_{n\to\infty} \frac{E(C_n)}{E(C^*_n)} < 1 + \epsilon .$$

Or, if one wants it,

$$\lim_{n\to\infty} \Pr\left(\frac{C_n}{C^*_n} > 1 + \epsilon \right) = 0 .$$

(The last event implies either $C_n > \alpha^*(1+\epsilon/2)n$ or $C^*_n < \alpha^* n$, and the probabilities of both of these events tend to zero with n.)

We conclude this section with a proof of the linear expected time complexity: $E(T) = O(n)$. When finding a good son of a node in the branching process, an effort not exceeding 2^L is spent. Then, by the family tree universal lemma, each depth first search takes time not exceeding cn, where c is a constant depending upon the branching process parameters. The expected number of depth-first-searches until a node is encountered that is the root of a surviving branching process is not more than $1/(1-q)$. Thus, the total expected time does not exceed

$$\frac{cn}{1-q} = O(n) .$$

Recently, McDiarmid and Provan (1991) pointed out that bounded lookahead without backtrack is also feasible. Assume that we find the optimal path from the root to a node at depth L. Make this node the new starting point and repeat. L is a large integer constant. For $p > 1/2$, and $\epsilon > 0$, one can show that there exists an L such that this algorithm runs in linear expected time, and that the best value found by the algorithm (C_n) satisfies the inequality

$$C_n \leq (1+\epsilon)C^*_n$$

with probability tending to one.

3.4 Bibliographic Remarks

The problem dealt with here was proposed and analyzed by Karp and Pearl (1983). An alternate short proof of Theorem 3.1 is given by McDiarmid (1990), where additional information about the problem may be found as well. The analysis of the optimal value C_n in the case $p < 1/2$ is due to McDiarmid and Provan (1991). Consider now depth first search in a complete binary tree in which the probability of a "one" edge is p and $2(1-p) > 1$. The following inequality is due to McDiarmid and Provan (1991): if C_n is the optimal value of a node at distance n from the root, then

$$\mathbf{P}\{C_n > k\} \le \left(\frac{b_2}{b-1}\right)^{b^{k-1}}, \quad k \ge 0 .$$

Karp and Zhang (1995) analyze models for AND/OR trees, where internal nodes at even (odd) distances from the root are AND (OR) nodes and each node has a boolean value 0 or 1. The value of a node is the outcome of the logical operator of the node on its children's values. The evaluation problem is to determine the root's value by examining the leaf values (which are randomly and independently assigned) while keeping computation to a minimum. This is Pearl's minimax tree model (1980). Karp and Zhang propose and analyze various algorithms using tail bounds on generation sizes in Galton-Watson processes. For minimax trees, Devroye and Kamoun (1996) analyze the value of the root in a random minimax tree, in which the leaf values in the n-th generation are those of a branching random walk, and intermediate level values are obtained by alternating the operations minimum and maximum.

4. Branching Random Walks

4.1 Definition

In a branching random walk, we superpose a random walk on each path from the root down in a Galton-Watson tree. More specifically, we associate with each individual u in a Galton-Watson tree a value V_u, the value of the root being zero. If u has N offspring (where N follows the model of the Galton-Watson process), then the values of the offspring relative to the value V_u of the parent u jointly have a given distribution. In the simplest model, for every child v of u, we have $V_v = V_u + X_v$, and all displacements X_v are independent (this will be called the independent branching random walk). However, in general, if the children have displacements $X_{u1}, \ldots, X_{uN}$, then the joint distribution of $(N, X_{u1}, \ldots, X_{uN})$ is quite arbitrary. What is important is that each parent produces children (and their values) in the same manner.

The analysis of branching random walks is greatly facilitated by the following function

$$m(\theta) = \mathbf{E}\left(\sum_{i=1}^{N} e^{-\theta V_{u_i}}\right)$$

where $u_1, \ldots, u_N$ are the children of the root. We assume throughout that $m(\theta) < \infty$ for some θ. This function may be considered as the Laplace-Stieltjes transform of $F(t) = \mathbf{E}\{Z_1(t)\}$, the expected number of individuals in the first generation with value less than or equal to t. In general, we introduce the notation $Z_n(t)$, the number of individuals in the n-th generation, with value $\le t$. Note that $Z_n = Z_n(\infty)$ so that this definition generalizes that of the previous section. Let Z^n be the point process with atoms V_u for all u in the n-th generation. Then, following Kingman (1975), introduce

$$W_n(\theta) = \frac{1}{m(\theta)^n} \sum_{u \text{ in generation } n} e^{-\theta V_u} .$$

This is a martingale for $\mathcal{F}_n$, the σ-field generated by all events in the first n generations. There is an almost sure limit, $W(\theta)$ (as $W_n(\theta) \ge 0$), and by Fatou's lemma, $\mathbf{E}(W(\theta)) \le 1$. The study of W_n and W reveals that there may be several modes of behavior, and this was studied by Biggins (1977) in more detail. In this section, we do not wish any distractions due to extinction of the underlying Galton-Watson process and assume therefore that N, the number of children per parent, is a fixed positive integer: $N = b$. For more general theorems, we refer to the cited papers.

In subsection 4.2, for $N = b$, we survey the main results on the first birth in the n-th generation, or $B_n = \min\{V_u : u \text{ in } n\text{-th generation}\}$, and on $Z_n(t)$, the distribution of values in the n-th generation. A straightforward application in the study of the height of trees then concludes this section.

4.2 Main Properties

Let X be a random variable equal to the value V_u of a randomly picked child of the root. Since $N = b$, the earlier definition of $m(\theta)$ specializes to

$$m(\theta) \stackrel{\text{def}}{=} b\mathbf{E}\left(e^{-\theta X}\right) .$$

Thus, if $X \ge 0$ is nondegenerate, we define the μ-function by

$$\mu(a) = \inf_{\theta \ge 0} \left\{e^{\theta a} m(\theta)\right\} = \inf_{\theta \ge 0} b\mathbf{E}\left(e^{\theta(a-X)}\right) .$$

Theorem 4.1. (Biggins, 1977). *If $\mu(a) < 1$, then with probability one, $Z_n(na) = 0$ for all but finitely many n. If $a \in \operatorname{int}\{a : \mu(a) > 1\}$, then*

$$\lim_{n\to\infty} (Z_n(na))^{1/n} = \mu(a)$$

almost surely.

This theorem shows that $\mu(a)^n$ is about equal to the number of individuals in the n-th generation with value $\le na$. Its simple proof is not given here, but it follows the lines of the proof of Theorem 2.1. In fact, Theorem 4.1 is nothing but a refined large deviation theorem: along any path from the root, the values form a standard random walk.

As a corollary of the above result, we have:

Theorem 4.3. [Kingman, 1975; Hammersley, 1974; Biggins, 1977] *Assume $m(\theta) < \infty$ for some $\theta > 0$. Let $B_n = \min Z_u$ (u is in the n-th generation). Then,*

$$\lim_{n\to\infty} \frac{B_n}{n} = \gamma \stackrel{\text{def}}{=} \inf\{a : \mu(a) > 1\}$$

almost surely, and γ is finite.

Interestingly, B_n grows linearly with n, while the n-th generation size itself grows exponentially with n. As the μ-function has an impact on both results, it is useful to have its properties at hand.

Lemma 4.3. *Let $X \ge 0$ be a nondegenerate random variable. Then its μ-function satisfies the following properties:*

(i) μ is an increasing function on $[0,\infty)$.

(ii) μ is continuous on $\operatorname{int}\{a : \mu(a) > 0\}$.

(iii) $\log\mu$ is concave on $\operatorname{int}\{a : \mu(a) > 0\}$.

(iv) $\sup_{a\ge 0}\mu(a) \le c$.

(v) If $\mathbf{E}(X) < \infty$, then $\mu(a) = b$ for $a > \mathbf{E}(X)$.

(vi) $\lim_{a\uparrow\infty}\mu(a) = b$.

(vii) If $X \ge c > 0$, then $\mu(a) = 0$ for $a < c$.

(viii) Let $s = \sup\{t : \mathbf{Pr}(X < t) = 0\}$, and define $p = \mathbf{Pr}(X = s)$. Then μ is continuous on (s,∞), $\mu(s) = bp$, and $\mu(a) = 0$ for $a < s$.

(ix) If $bp < 1$, and $\gamma = \inf\{a : \mu(a) \ge 1\}$, then $\mu(\gamma) = 1$.

If all displacements with respect to a parent are identical, then we speak of a Bellman-Harris branching random walk. McDiarmid (1995) calls this a common branching random walk. Of course, all theorems above also apply to this situation. It is of interest to pin down the asymptotic behavior of B_n beyond Theorem 4.3. Consider for example an infinite b-ary tree on which we superimpose a branching random walk, with all displacements Bernoulli $(1/b)$, that is, they are 1 with probability $1/b$ and 0 otherwise. The case $b = 2$ is easiest to picture, as all displacements are independent equiprobable bits. Joffe, LeCam and Neveu (1973) showed that $B_n/n \to 0$ almost surely, and this also follows from Theorem 4.3, which was published later. Bramson (1978) went one further and showed that there exists a random variable W such that

$$\lim_{n\to\infty} B_n - \frac{\log\log n - \log W + o(1)}{\log 2} = 0$$

almost surely, where the $o(1)$ term is stochastic. In the binary case, each individual in the n-th generation has a binomial $(n, 1/2)$ distribution. If these 2^n binomials had been independent, we would have had $\liminf_{n\to\infty} B_n = 0$ almost surely and $\limsup_{n\to\infty} B_n = 1$ almost surely. This follows from the fact that $\mathbf{Pr}(B_n = 0) \to 1 - 1/e$ as $n \to \infty$ and $\mathbf{Pr}(B_n \ge 2) \le e^{-(n+1)}$. Thus Bramson's result exposes a crucial property of branching random walks. Dekking and Host (1991) consider the general branching random walk with nonnegative integer-valued displacements. Thus, $B_n \uparrow$. Let $N(i)$ be the number of children of the root with displacement i. Let $N = \sum_{i=0}^{\infty} N(i)$ be the number of offspring of the root. Again, we assume $N = b$ with probability one, although the results of Dekking and Host treat the general case. Some of their results can be summarized as follows:

Theorem 4.4. [Dekking and Host, 1991] *If γ denotes the constant of Theorem 4.3, then $\gamma = 0$ if and only if $\mathbf{E}(N(0)) \ge 1$.*

Assume that $\mathbf{Pr}(N(0) = 1) < 1$. Then $\mathbf{Pr}(B_n \to \infty) \in \{0, 1\}$, and the latter case happens if and only if $\mathbf{E}(N(0)) \le 1$. Also,

A. If $\mathbf{E}(N(0)) > 1$, then there exists a proper random variable W such that $B_n \to W$ almost surely.

B. If $\mathbf{E}(N(0)) = 1$, $\mathbf{E}(N^2) < \infty$, and $g = \min\{i > 0 : \mathbf{E}(N(i)) > 0\}$, then $B_n \log 2 / \log\log n \to g$ almost surely.

If $\mu = \mathbf{E}(N(1)) > 1$ and $r = (1/2)\operatorname{var}(N(0))$, then for integer $k \ge 1$,

$$\mathbf{Pr}(B_n \le k) \sim \frac{\mu^k}{r(\mu n)^{1/2^k}} \text{ as } n \to \infty.$$

McDiarmid (1995) extends the results of Dekking and Host in some cases. Consider only nonnegative displacements, and recall that the branch factor is b. Then, if b_n is the median of B_n, McDiarmid establishes the existence of positive constants c, d such that for all a,

$$\mathbf{Pr}\{|B_n - b_n| \ge a\} \le c e^{-da}$$

for all $x \in [0, \alpha]$. This implies that, almost surely, for all n large enough, $B_n - b_n = O(\log n)$. Clearly, by Theorem 4.2, b_n should be near γn. The following result describes the closeness of B_n to γn. We give only the version for the case that the underlying Galton-Watson tree is the complete infinite b-ary tree.

Theorem 4.5. [McDiarmid, 1995] *Consider a common branching random walk in which every individual has b children, and all displacements are on $[\alpha, \infty)$, where α is the leftmost point of the support of the displacement random variable Z_1, and $b\Pr\{Z_1 = \alpha\} < 1$. Let $\gamma > 0$ be the (necessarily unique) solution of $\mu(\gamma) = 1$, and let m be finite in a neighborhood of γ. Then there are positive constants c, d, d' such that*

$$\Pr\{B_n \le \gamma n - c\log n - x\} \le e^{-dx}, \quad x \ge 0,$$

and

$$\Pr\{B_n \ge \gamma n + c'\log n - x\} \le e^{-d'x}, \quad 0 \le x \le n.$$

McDiarmid's proof does not imply $c = c'$, but it strengthens earlier results, such as a result by Biggins (1977), who showed that under the stated conditions, $B_n - \gamma n \to \infty$ almost surely. Interestingly, his argument is based on the second moment method and the idea of leading sequences. A sequence $(x_1, \ldots, x_k)$ is leading if for all $j = 1, \ldots, k-1$,

$$\sum_{i=1}^{j} x_i \ge \frac{j}{k}\sum_{i=1}^{k} x_i .$$

If $(X_1, \ldots, X_k)$ are exchangeable random variables, then indeed,

$$\Pr\{(X_1, \ldots, X_k) \text{ is leading}\} \ge 1/k.$$

Given an individual u in the n-th generation, we denote by $Y_1, \ldots, Y_n$ the displacements encountered on the path from the root to u. We call u leading if this displacement sequence is leading, that is, if $W_j \ge (j/n)W_n$, where $W_1, \ldots, W_n$ are the values of the ancestors of u in generations 1 through n. Clearly, $Z_n(t) \ge Z_n^*(t)$, where $Z_n^*(t)$ is the number of leading individuals in the n-th generation with value $\le t$. It should be clear that $Z_n^*(t)$ is not small when $Z_n(t)$ is large, and one gains a lot by considering $Z_n^*(t)$, or by considering the minimum value B_n^* among leading individuals, instead of just B_n. A careful application of the second moment method ($\Pr\{X > 0\} \ge (\mathbf{E}X)^2/\mathbf{E}\{X^2\}$ for any random variable X with finite mean $\mathbf{E}X > 0$) then yields Theorem 4.5.

4.3. Application to Analysis of Height of Trees

One may use Theorem 4.2 in the study of the height of a large class of random trees. These trees can be modeled indirectly by the size tree, a tree in which we associate with each node u the size of its subtree S_u. For the root, we have $S_r = n$, and for each leaf, $S_u = 1$. Often, these size trees are close to a split tree, in a manner to be made precise. A split tree T is easily described: with root value $V_r = 1$, it is an infinite b-ary tree, and the values of the children $u_1, \ldots, u_b$ are $V_u X_{u1}, \ldots, V_u X_{ub}$. Furthermore, $\sum_{i=1}^{b} X_{ui} = 1$ and $X_{ui} \ge 0$ for all i. In other words, considering the value as mass of a subtree, the mass at the root is partitioned into smaller masses that again add up to one. This process continues forever, each particle splitting in the same manner. The distribution of values in the split tree is governed by the joint distribution of the b child values of the root. If we consider $Y_u = -\log V_u$, then the above model describes a branching random walk. Let m() and μ() be defined as for that random walk, that is, if X is the value of a randomly picked child of the root (so $0 \le X \le 1$), then

$$m(\theta) = b\mathbf{E}\left(e^{-\theta(-\log X)}\right) = b\mathbf{E}(X^{\theta}) .$$

Define

$$\mu(c) = \inf_{\theta \le 0}\left(e^{\theta c} m(\theta)\right) = \inf_{\theta \le 0} b\mathbf{E}\left(X^{\theta} e^{\theta c}\right) .$$

Finally, let $N_n(t)$ be the number of n-th generation individuals with value exceeding t in the split tree. The following is a corollary of Theorem 4.2:

Theorem 4.6. *If $\mu(c) < 1$, then with probability one, $N_n(e^{-cn}) = 0$ for all but finitely many n. If $c \in \text{int}\{c : \mu(c) > 1\}$, then $\lim_{n\to\infty}(N_n(e^{-cn}))^{1/n} = \mu(c)$ almost surely. Furthermore, if H_n is the maximal value of any individual in the n-th generation of the split tree, then*

$$\lim_{n\to\infty} \frac{-\log H_n}{n} = \gamma \stackrel{\text{def}}{=} \inf\{c : \mu(c) > 1\}$$

almost surely.

The above results may be applied in the study of Kolmogorov's rock (see Athreya and Ney, 1972), which is subjected to many rounds of breaking, and each break results in two rocks with uniform size. If the initial rock has mass one, then Theorem 4.6 describes the maximal rock size among 2^n shattered rocks in the n-th generation. The random variables that govern the splitting are $(U, 1-U)$, where U is uniformly distributed on [0, 1]. In this case, we have

$$m(\theta) = 2\mathbf{E}(U^{\theta}) = \frac{2}{\theta+1} .$$

Also,

$$\mu(c) = \inf_{\lambda>0}\left\{\frac{2e^{\lambda c}}{\lambda+1}\right\} = 2ce^{1-c} .$$

From this, we determine γ as the solution of $2ce^{1-c} = 1$ and obtain $\gamma = 0.2319\ldots$. As a consequence, the size M_n of the largest rock is almost surely $e^{-n(\gamma+o(1))}$. For comparison, if we were to break the rocks evenly, then $M_n = 2^{-n} = e^{-n\log 2}$, almost the third power of the maximal rock in the random model!

However, the way these split trees are used is different. A search tree with n nodes has n nodes in the root, so we define our split tree in such a way that each node has n times the value of the corresponding node in the original split tree. These (generally non-integer) roughly represent the sizes of the subtrees. Nodes with value (after multiplication with n) less than 1 correspond to nothing and will be cut. In this manner the size tree is finite. For example, in a random binary search tree, the sizes of the left and right subtrees of the root are distributed as $\lfloor nU\rfloor$ and $\lfloor n(1-U)\rfloor$ respectively, where U is uniform $[0,1]$. These sizes are jointly smaller than $(nU, n(1-U))$ and thus by proceeding, we can say that the values in the size tree are jointly (over the whole tree) smaller than the values in a split tree with multiplicative factor n and with root child values $(U, 1-U)$. Furthermore, the sizes of the left and right subtrees are jointly larger than $(nU-1, n(1-U)) - 1)$. If we repeat this sort of bounding for k generations, then it is easy to see that all values in the size tree in generation k are jointly larger than the values in the split tree just defined, minus k. The connection between size trees and split trees is thus established. In particular, what interests us most is that if H_n is the height of the binary search tree with n nodes, then

$$\begin{aligned}\Pr\{H_n > k\} &= \Pr\{\text{maximum value in generation } k \text{ of size tree} \geq 1\}\\ &\leq \Pr\{nM_k \geq 1\}\end{aligned}$$

where M_k is the maximal value of a kth generation node in the original split tree (not the multiplicative factor). Similarly,

$$\begin{aligned}\Pr\{H_n < k\} &= \Pr\{\text{maximum value in generation } k \text{ of size tree} < 1\}\\ &\leq \Pr\{nM_k - k < 1\}\end{aligned}$$

As $M_k = e^{-\gamma k(1+o(1))}$ almost surely as $k\to\infty$, where γ is precisely as in the example of Kolmogorov's rock, it is easy to conclude from these inequalities the following (essentially Theorem 2.1): for $\epsilon > 0$,

$$\lim_{n\to\infty}\Pr\left\{\frac{H_n}{\log n} > \frac{1}{\gamma}+\epsilon\right\} = 0$$

and

$$\lim_{n\to\infty}\Pr\left\{\frac{H_n}{\log n} < \frac{1}{\gamma}-\epsilon\right\} = 0 .$$

Thus, $H_n/\log n \to 1/\gamma = 4.31107\ldots$ in probability, where γ is defined in Theorem 2.1. For the random binary search tree, we thus have a second proof of Theorem 2.1.

The technique above consists in controlling the sizes of the subtrees of a random tree by an embedding argument, and to relate these sizes to those of a split tree by suitable inequalities. This has been done in the literature for a number of random trees, and rather than dwelling on the details we will review the known results. The remainder of this section is rather specialized and may be skipped upon first reading.

EXAMPLE 1: THE RANDOM b-ARY SEARCH TREE. Let n i.i.d. random variables with a common density be used to construct a random b-ary search tree, where each physical node holds up to $b-1$ elements. As soon as a node is full, new nodes searching it on the path down from the root are sent down to one of the b child trees by a comparison of values of the $b-1$ (sorted) elements in the node. Here the tree size is measured in number of elements, not number of nodes. The first $b-1$ elements occupy the root. Without loss of generality, they are i.i.d. uniform $[0,1]$. Thus, as the other elements are independent, we see that the subtree sizes $(N_1,\ldots,N_b)$ are distributed as a multinomial random vector with sum $n-b+1$ and probabilities given by $S_1,\ldots,S_b$, the spacings determined on $[0,1]$ by a uniform sample of size $b-1$. Now, the relationship between the size tree and the split tree is only slightly more intricate, but the split tree clearly should have multiplicative factor n and split random vectors $(S_1,\ldots,S_b)$ (see Devroye, 1990 for the details). In particular, the S_i's are beta$(1, b-1)$ distributed (Pyke, 1965), and we can thus easily compute

$$m(\theta) = \mathbf{E}\left(\sum_i S_i^\theta\right) = b\mathbf{E}(S_1^\theta) = b\int_0^1 x^\theta(b-1)(1-x)^{b-2}\,dx = \frac{\Gamma(\theta+1)\Gamma(b+1)}{\Gamma(\theta+b)} .$$

Unfortunately, the expression for γ is in general not simple. We have $H_n/\log n \to \zeta$ in probability, where

$$\zeta = \inf\left\{c > 0 : c\sum_{j=2}^{b}\log j + t - c\sum_{i=1}^{b-1}\log(i+t) < 0\right\}$$

and $t > 0$ is the unique solution of

$$\frac{1}{c} = \sum_{i=1}^{b-1}\frac{1}{i+t}$$

(Devroye, 1990). Particular values of ζ include $\zeta = 4.31107\ldots$ ($b = 2$), $\zeta = 2.4699\ldots$ ($b = 3$), $\zeta = 0.8679\ldots$ ($b = 5$) and $\zeta = 0.3615\ldots$ ($b = 100$). The depth of the last node, D_n, is in probability asymptotic to $\log n/\sum_{i=2}^{b}(1/i)$ (Mahmoud and Pittel, 1984). Devroye (1987) showed that if $\mu = 1/\sum_{i=2}^{b} 1/i$ and $\sigma^2 = \sum_{i=2}^{b} 1/i^2$, then

$$\frac{D_n - \mu\log n}{\sqrt{\sigma^2\mu^3\log n}} \xrightarrow{\mathcal{L}} N(0,1)$$

where N denotes a normal random variable. As an example, if $b = 3$,

$$\frac{D_n - (6/5)\log n}{\sqrt{(78/125)\log n}} \xrightarrow{\mathcal{L}} N(0,1).$$

EXAMPLE 3: THE RANDOM QUADTREE. The point quadtree in R^d (Finkel and Bentley, 1974; see Samet (1990b) for a survey) generalizes the binary search tree. Defined in the previous chapter, we only consider uniform data in $[0,1]^d$. Note that if the root is $X = (X_1, \ldots, X_d)$, then the probabilities (volumes) of the 2^d quadrants are given by the identically distributed (but dependent) random variables

$$\prod_{i=1}^{d} X_i^{b_i}(1 - X_i)^{1-b_i}$$

where $b_1, \ldots, b_d$ is a vector of d bits identifying one of the 2^d quadrants. Devroye (1987) establishes probability inequalities between the values in the size tree and the values in the split tree, which imply for first order results that it suffices to study the split tree. Then we note that

$$m(\theta) = 2^d\mathbf{E}\left(\prod_{i=1}^{d} X_i^{\theta}\right) = 2^d\prod_{i=1}^{d}\mathbf{E}(X_i^{\theta}) = \left(\frac{2}{\theta+1}\right)^d,$$

thus generalizing the binary search tree (obtained when $d = 1$). Thus,

$$\rho(\mu) = \inf_{\theta<0}\left\{\frac{2e^{\theta\mu}}{\theta+1}\right\}^d = \left(2e\mu e^{-\mu}\right)^d.$$

Therefore, by simple inspection, $\rho(c/d) = 1$, where c is the parameter for the binary search tree. As a result, the height H_n of a random quadtree is in probability asymptotic to $(c/d)\log n$, where $c = 4.31107\ldots$ is the constant in the height of the random binary search tree (Devroye, 1987). Let D_n be the depth of the last node. It is also known that

$$\frac{D_n}{\log n} \to \frac{2}{d} \text{ in probability},$$

a result first noted by Devroye and Laforest, 1990. See also Flajolet, Gonnet, Puech and Robson (1991). Furthermore

$$\frac{D_n - (2/d)\log n}{\sqrt{(2/d^2)\log n}} \xrightarrow{\mathcal{L}} N(0,1),$$

valid for any $d \ge 1$. This result was obtained via complex analysis by Flajolet and Lafforgue (1994) and by random central limit theorems by Devroye (1987). EXAMPLE 4: THE RANDOM MEDIAN-OF-$(2k+1)$ BINARY SEARCH TREE. Bell (1965) and Walker and Wood (1976) introduced the following method for constructing a binary search tree. Take $2k+1$ points at random from the set of n points on which a total order is defined, where k is integer. The median of these points serves as the root of a binary tree. The remaining points are thrown back into the collection of points and we recur to the subtrees. Following Poblete and Munro (1985), we may look at this tree by considering internal nodes and external nodes, where internal nodes hold one data point and external nodes are bags of capacity $2k$. Insertion proceeds as usual. As soon as an external node overflows (i.e., when it would grow to size $2k+1$), the bag is split about the median, leaving two new external nodes (bags) of size k each, and an internal node holding the median. After the insertion process is completed, we may wish to expand the bags into balanced trees. Using the branching process method of proof (Devroye, 1986b, 1987, 1993; see also Mahmoud, 1992) the almost sure limit of $H_n/\log n$ for all k may be obtained (Devroye, 1993). For another possible proof method see Pittel (1992). The depth D_n of the last node when the fringe heuristic is used has been studied by the theory of Markov processes or urn models in a series of papers, notably by Poblete and Munro (1985), Aldous, Flannery and Palacios (1988). See also Gonnet and Baeza-Yates (1991, p. 109). Poblete and Munro (1985) showed that

$$\frac{D_n}{\log n} \to \frac{1}{\sum_{i=k+2}^{2k+2}\frac{1}{i}} \stackrel{\text{def}}{=} \lambda(k),$$

in probability. It should be clear by now that the height of this tree may be studied as a split tree with split vector distributed as $(B, 1-B)$, where B is beta $(k+1, k+1)$. That is, B is distributed as the median of $2k+1$ i.i.d. uniform $[0,1]$ random variables. This representation is obtained by associating with each point in the data an independent uniform $[0,1]$ random variable. Equivalently, if the U_i are independent uniform $[0,1]$ random variables, then B is distributed as

$$\prod_{i=k+1}^{2k+1} U_i^{1/i}.$$

Note that in this case

$$m(\theta) = 2\mathbf{E}(B^{\theta}) = \frac{\Gamma(2k+2-\theta)\Gamma(k+1)}{\Gamma(2k+2)\Gamma(k-\theta+1)}.$$

The computation of ρ is a little bit more tedious, but the result can be obtained indirectly:

Theorem 4.7. [Devroye, 1993] *A random binary search tree constructed with the aid of the fringe heuristic with parameter k has the following property: $\frac{H_n}{\log n} \to c(k)$ in probability where $c(k)$ is the unique solution greater than $\lambda(k)$ of the equation*

$$\psi(c) - c\sum_{i=k+1}^{2k+1} \log\left(1+\frac{\psi(c)}{i}\right) + c\log 2 = 0\,,$$

where $\psi(c)$ is defined by the equation

$$\frac{1}{c} = \sum_{i=k+1}^{2k+1} \frac{1}{\psi+i}\,.$$

In particular, $c(0) = 4.31107\ldots$ (the ordinary binary search tree), $c(1) = 3.92370\ldots$, $c(2) = 3.65558\ldots$, $c(10) = 3.30268\ldots$ and $c(100) = 3.02895\ldots$

With

$$\sigma^2 = \sum_{i=k+2}^{2k+2} \frac{1}{i^2}\,,$$

Devroye (1997) obtained a central limit theorem for D_n for all k:

$$\frac{D_n - \lambda\log n}{\sqrt{\sigma^2\lambda^3\log n}} \xrightarrow{\mathcal{L}} N(0,1)\,.$$

As an example, for $k = 1$, we obtain

$$\frac{D_n - (12/7)\log n}{\sqrt{(300/343)\log n}} \xrightarrow{\mathcal{L}} N(0,1)\,.$$

Example 4. RANDOM SIMPLEX TREES. Triangulating polygons and objects in the plane is an important problem in computational geometry. Arkin, Held, Mitchell and Skiena (1994) obtained a simple fast $O(n\log n)$ expected time algorithm for triangulating any collection of n planar points in general position. We look more specifically at their triangulation and its d-dimensional extension to simplices, and ask what the tree generated by this partitioning looks like if the points are uniformly distributed in the unit simplex. Given are n vectors $X_1, \ldots, X_n$ taking values in a fixed simplex S of $\mathbb{R}^d$. It is assumed that this is an i.i.d. sequence with a uniform distribution on S for the purposes of analysis. X_1 is associated with the root of a $d+1$-ary tree. It splits S into $d+1$ new simplices by connecting X_1 with the $d+1$ vertices of S. Associate with each of these simplices the subset of $X_2, \ldots, X_n$ consisting of those points that fall in the simplex. Each nonempty subset is sent to a child of the root, and the splitting is applied recursively to each child. As every split takes linear time in the number of points processed, it is clear that the expected time is proportional to $nE\{D_n\}$, where D_n is the expected depth of a random node in the tree. The partition consists of $dn+1$ simplices, each associated with an external node of the tree. There are precisely n nodes in the tree and each node contains one point. If $|S|$ denotes the size of a simplex S, then the following crucial property is valid.

Lemma 4.8. [Devroye, 1997] *If simplex S is split into $d+1$ simplices $S_1, \ldots, S_{d+1}$ by a point X distributed uniformly in S, then $(|S_1|, \ldots, |S_{d+1}|)$ is jointly distributed as $(|S|V_1, \ldots, |S|V_{d+1})$, where $V_1, \ldots, V_{d+1}$ are the spacings of $[0,1]$ induced by d i.i.d. uniform $[0,1]$ random variables.*

It is immediate that the random simplex tree is a split tree with split vector distributed as the spacings defined by d i.i.d. uniform $[0,1]$ random variables on $[0,1]$ and branch factor $d+1$. Therefore, H_n (and also D_n) behave precisely as for the random $d+1$-ary tree discussed earlier. Thus, if $\sigma^2 = \sum_{i=2}^{d+1} 1/i^2$,

$$\frac{D_n}{\log n} \to \lambda \stackrel{\text{def}}{=} \frac{1}{\sum_{i=2}^{d+1}\frac{1}{i}} \quad \text{in probability}$$

and

$$\frac{D_n - \lambda\log n}{\sqrt{\sigma^2\lambda^3\log n}} \xrightarrow{\mathcal{L}} N(0,1)\,.$$

As an example, if $d = 2$, then and

$$\frac{D_n - (6/5)\log n}{\sqrt{(78/125)\log n}} \xrightarrow{\mathcal{L}} N(0,1)\,.$$

We also know that $H_n/\log n \to c(d)$ in probability for a function c of d that may be computed via the recipe described in the example on d-ary search trees.

4.4 Refinements for Binary Search Trees

The results of the previous section permit fundamentally only first order asymptotic analysis of H_n. For the study of the depth of the last node D_n or the depth of a typical node, branching processes are really not necessary although they could be used. Devroye (1998) derives a general central limit theorem for D_n, illustrated in the previous examples, based on a split tree model as in the previous section. By allowing n balls to drop according to a certain process down an infinite b-ary tree in which nodes may hold zero, one or more balls, the model is rich enough to encompass both search trees and tries or digital search trees. Recall that $\gamma = 4.31107\ldots$ the unique solution greater than 2 of $c\log(2e/c) = 1$. Theorem 3.1 implies that the height H_n of the random binary search tree satisfies $H_n/\log n \to \gamma$ in probability. In fact, convergence is in the almost sure sense as well, a fact first noted by Pittel (1984). Using elementary inequalities and essentially the bounds found in this survey, Devroye (1987) showed that $H_n - \gamma\log n = O(\sqrt{\log n \log\log n})$ in probability. Robson (1979) reported that H_n was much more concentrated than that, and conjectured even that $\mathrm{Var}(H_n) = O(1)$. There have been three attempts to crack this conjecture.

Michael Drmota (1997) uses generating functions to prove that $\mathbf{E}(H_n) \sim \gamma\log n$, and his proof is the first one based on this approach. This method may have two benefits: first of all, it may provide detailed behavior on the exact behavior of $\mathbf{E}(H_n)$ (the lower order terms may be useful therefore), and the method may perhaps one day be extended to treat $\mathrm{Var}(H_n)$ in a similar manner.

Devroye and Reed (1995) provided the first analysis of the height that did not require any results from the theory of branching processes. Instead, they mark certain paths to leaves in the split tree that corresponds to the binary search tree, and apply the second moment method to compute bounds on probabilities. Interestingly, the marked leaves are sufficiently spread out to make this method work. This method was later generalized, via the notion of leaving sequences, to compute branching random walks, by McDiarmid (1995) (see Theorem 4.5). They were able to show that

$$\lim_{n\to\infty} \mathbf{P}\left\{ |H_n - \gamma\log n| > \frac{15\gamma}{\log 2}\log\log n \right\} = 0 .$$

(Note that $15\gamma/\log 2 = 93.2933\ldots$.) Using a surprisingly elementary recursive argument, Robson (1997) showed that for any $\epsilon > 0$, infinitely often, we have

$$\mathbf{E}(|H_n - \mathbf{E}(H_n)|) \le \frac{8\gamma}{\log 2} + 1 + \epsilon .$$

In fact, if

$$\sup_n (\mathbf{E}(H_{2n}) - \mathbf{E}(H_n)) < \infty ,$$

then his method allows one to conclude that

$$\sup_n \mathbf{E}(|H_n - \mathbf{E}(H_n)|) < \infty .$$

If we knew $\mathbf{E}(H_n)$ down to $O(1)$ terms, we would be done, at least for first moment deviations.

Finally, we just learned from Jean Jabbour (1998) at the University of Versailles that he has a proof of Theorem 3.1 based solely on martingales. This may be yet another path along which to proceed.

4.6 Bibliographic Remarks

For general background information see, for example, Asmussen and Hering (1983), Athreya and Ney (1972), and Harris (1963). Lemma 4.8 takes elements from Kingman (1975), Biggins (1977), and Devroye and Zamora (1997). The minimal displacement B_n was compared by Durrett (1979) with that of the independent tree model in which all nth generation individuals have independent values of their common distribution. Bramson (1978) also worked out the finer behavior of B_n when the displacements are gaussian, or in general when particles describe Brownian motion and split at random times. Biggins (1990) derives a central limit theorem for $Z_n(.)$ when $\mathbf{E}(Y\log Y) < \infty$, where Y is the number of offspring. Lemma 4.9 is implicit in many older references, such as Rubinstein (1982), Smith (1984) or Devroye (1986a).

5. Crump-Mode-Jagers Process

5.1 Introduction

The Crump-Mode-Jagers (or CMJ) branching (Crump and Mode 1968) starts with a single ancestor born at time $t = 0$. $Z_1(t)$, the number of children born to the ancestor before time t is an arbitrary counting process. The children of the ancestor, from their births, behave independently of one another and of their parent, producing children at random according to random processes with the same joint distribution as $Z_1(.)$. Their children produce children in the same way, and so on. We speak of a Poisson CMJ branching process if the between-birth intervals are exponentially distributed with parameters $\lambda_0, \lambda_1, \ldots$ respectively. Thus, births occur at intervals distributed as $E_0/\lambda_0, E_1/\lambda_1, \ldots$, where the E_i's are independent and exponentially distributed random variables. Note that if $\lambda_k = 0$, for some k, then the number of offspring of an individual can never exceed k.

If we link each individual with its parent, then we obtain a tree, and the notion of a generation becomes meaningful again. Several random variables are of interest here:

A. τ_n, the time at which the tree has exactly n nodes.

B. S_k, the time of the first birth in the k-th generation.

C. H_n, the height of the tree at time τ_n.

D. Z_k, the number of individuals in generation k.

E. $Z(t)$, the number of individuals at time t.

F. $H(t)$, the height of the tree at time t.

The reason CMJ processes are important to us is because of the following connection with random trees that can be grown in an incremental manner. The random trees are grown one edge at a time, starting from the root. If the degrees of the current nodes are denoted by D_i, then node i is selected with probability proportional to λ_{D_i}. This node becomes the parent of a new node. Observe that the order of the births in the Poisson CMJ process follows exactly that of the incremental random trees just described. Also, both are probabilistically equivalent if we are only interested in studying depths and heights of nodes. The last remark is rooted in the observation that if we have a number of birth processes with rates λ_i, then process i gives the next birth with probability proportional to λ_i. The model described above and the continuous time embedding ideas are due to Pittel (1984).

EXAMPLES.

A. The uniform random recursive tree (URRT) has $\lambda_i = 1$ for all i. It is grown by choosing a parent with equal probability from among all possible parents.

B. The random m-ary pyramid with $m \geq 2$ has $\lambda_i = 1$ for $i < m$ and $\lambda_i = 0$ for $i \geq m$. Here we choose a parent uniformly at random from among those parents with less than m children. See Mahmoud (1994).

C. In the random binary search tree, we have $\lambda_0 = 2$, $\lambda_1 = 1$ and $\lambda_2 = 0$. To see quickly why this incremental tree model corresponds to the standard random binary search tree, consider a random binary search tree constructed on the basis of an i.i.d. sequence of uniform $[0,1]$ random variables $U_1, U_2, \ldots$. Given that the tree has $n-1$ nodes, the n-th node has a rank that is uniformly distributed on $\{1, 2, \ldots, n\}$. That is, it falls in one of the n intervals on $[0,1]$ defined by the first $n-1$ uniform random variables. But each such interval corresponds uniquely to a potential new node (these are called external nodes), and there are two external nodes for a node with no children, and one for a node with one child.

D. The linear recursive tree has $\lambda_i = 1 + bi$ for some positive constant b. To visualize this, consider $b = 1$. To grow a tree, we pick a parent with probability proportional to one plus the number of children. For $b = 1$, this is called a plane-oriented recursive tree by Mahmoud (1993) and Mahmoud, Smythe and Szymański (1993) (see also Szymański 1987, and Bergeron, Flajolet and Salvy 1992). The last name is selected because of the following planar visualization: draw the tree in the plane, and place a new edge uniformly at random in any possible child of any possible node. In this manner, a plane-oriented tree is defined.

There are three recent papers that provide an analysis of the height of these random trees using Crump-Mode processes: Pittel (1994) for the URRT and linear recursive tree, Mahmoud (1994) for random pyramids, and Biggins and Grey (1996) in the more general setting followed in this chapter. The height H_n can be analyzed using the Biggins-Hammersley-Kingman theorem (Theorem 4.2). We conclude by working out the details for the various tree models mentioned above.

5.2 The Main Result

The relationship between the CMJ process and the branching random walk is clear, if we let the displacements in the branching random walk be the inter-birth times. As the branch factor may be unbounded (as for the URRT case), we need to follow a general set-up. For simplicity, to ensure survival, we assume throughout that $Z_1(\infty) > 1$. For a general branching walk process, we define the Laplace transform of the mean reproduction measure,

$$m(\theta) = E\left(\sum_i e^{-\theta X_i}\right)$$

where the X_i's are the realizations of $Z_1(\cdot)$, and the sum ranges over all children of the root.

Example. For a Poisson CMJ process, we have $Y_1 = E_0/\lambda_0$, $Y_2 = Y_1 + E_1/\lambda_1$, and so forth, so that

$$\begin{aligned} m(\theta) &= \sum_{i=0}^{\infty} E\left(e^{-\theta(E_0/\lambda_0 + \cdots + E_i/\lambda_i)}\right) \\ &= \sum_{i=0}^{\infty} \prod_{j=0}^{i} E\left(e^{-\theta E_j/\lambda_j}\right) \\ &= \sum_{i=0}^{\infty} \prod_{j=0}^{i} \frac{1}{1+\frac{\theta}{\lambda_j}} . \end{aligned}$$

Assuming that $m(\theta) < \infty$ for some $\theta > 0$, we note that as $\theta \to \infty$, $m(\theta) \to 0$. Observe that a sufficient condition for this is that $\lambda_i = O(1)$ as $i \to \infty$ (in the Bellman-Harris case). Define

$$\mu(a) = \inf\left\{e^{a\theta} m(\theta) : \theta \geq 0\right\},$$

and observe that $\log \mu(a)$ is concave (the infimum of a family of lines is concave) and $\mu(a)$ is continuous on the interior of $\{a : \mu(a) > 0\}$.

Define $Z_k(t)$, the number of individuals in generation k with birth time at most t. Biggins (1977) uses classical large deviation results by Bahadur and Rao (1960) and Chernoff (1952) to prove the following.

Theorem 5.1. [Biggins, 1977; Hammersley, 1974; Kingman, 1975] *If $m(\theta) < \infty$ for some $\theta > 0$, then $(E(Z_n(na)))^{1/n} \to \mu(a)$ as $n \to \infty$. Furthermore, if $\mu(a) < 1$, then with probability one, $Z_n(a)(na) = 0$ for all but finitely many n. If $a \in \operatorname{int}\{a : \mu(a) > 1\}$, then $\lim_{n\to\infty}(Z_n(na))^{1/n} = \mu(a)$ almost surely. Finally,*

$$\lim_{n\to\infty} \frac{B_n}{n} = \gamma \overset{\text{def}}{=} \sup\{a : \mu(a) < 1\}$$

almost surely, and γ is finite.

We must relate B_n to H_n. Observe that at the moment t_n the family tree is of size n and of height H_n and that $B(H_n)$ and $B(H_n + 1)$ are the first moments when the height becomes equal to H_n and $H_n + 1$ respectively. Therefore,

$$B(H_n) \leq t_n \leq B(H_n + 1).$$

Since $t_n \to \infty$ almost surely, we have $H_n \to \infty$ almost surely as well. Thus, $B(H_n)/H_n \to \gamma$ almost surely, and $t_n/H_n \to \gamma$ almost surely. Therefore it suffices to study t_n. This can be done on a case by case basis, as is routinely done in the literature. However, there is a universal theorem:

Theorem 5.2. [Nerman, 1981; Biggins, 1995] *If $m(\theta) < \infty$ for some $\theta > 0$, and $Z(t)$ denotes the number of births up to time t, and*

$$\alpha \overset{\text{def}}{=} \inf\{\theta : m(\theta) < 1\},$$

which is positive and finite, as $m(0) \geq 1$ and $m(\theta) \to 0$ as $\theta \to \infty$, then

$$\frac{\log Z(t)}{t} \to \alpha$$

almost surely as $t \to \infty$. Equivalently,

$$\frac{t_n}{\log n} \to \frac{1}{\alpha}$$

almost surely as $n \to \infty$.

From this, we have

Theorem 5.3. [Biggins and Grey, 1996] *Under the conditions of Theorem 5.2*

$$\frac{H_n}{\log n} \to \frac{1}{\alpha\gamma}$$

almost surely as $n \to \infty$.

5.3 Application to Various Tree Models

In a few special cases, we have very refined information about t_n. This occurs principally when we can describe the spacings between consecutive births quite accurately. Consider first a branching process with one child per node and the inter-birth times are exponential of unit parameter, then t_n is the sum of n independent standard exponential random variables so that $t_n/n \to 1$ almost surely. Also, $H_n = n - 1$, $m(\theta) = 1/(1+\theta)$ and

$$\mu(a) = \inf\left\{\frac{e^{a\theta}}{1+\theta} : \theta \geq 0\right\}.$$

The minimum occurs at $\theta = \max(1/a - 1, 0)$, so that

$$\mu(a) = \begin{cases} ae^{1-a} & (0 < a < 1) \\ 1 & (a \geq 1). \end{cases}$$

Since $\mu(1) = 1$, we have $\gamma = 1$. This was just a (simple) roundabout way of checking what we already knew, that $H_n/n \to 1$ almost surely (as $H_n = n - 1$).

In the second example, let Y_1, Y_2, the children of the root, be born at independent standard exponential times. In this case

$$m(\theta) = \frac{2}{1+\theta}.$$

Clearly

$$\mu(a) = \inf\left\{\frac{2e^{a\theta}}{1+\theta} : \theta > 0\right\}.$$

The minimum occurs at $\theta = \max(1/a - 1, 0)$, so that

$$\mu(a) = \begin{cases} 2ae^{1-a} & (0 < a < 1), \\ 2 & (a \geq 1). \end{cases}$$

Thus γ is the solution less than one of $2ae^{1-a} = 1$. Regarding t_n, note that we have inter-birth times that are distributed as $E_1/2, E_2/3, \ldots, E_n/n$, where the E_i's are independent exponential random variables. From this, it is easy to show that

$$\frac{L_n}{\log n} \to 1$$

almost surely. Therefore, $H_n/\log n \to 1/\gamma$ almost surely. This may be cast in the Poisson case model, as the first birth to the ancestor occurs at a time distributed as $E_1/2$, and the second at a time distributed as $E_1/2 + E_2$, where the E_i's are exponential random variables. Thus $\lambda_0 = 2$, $\lambda_1 = 1$, and $\lambda_i = 0$ for $i \geq 2$. This of course yields the same results.

In a third example, let the root have children whose times of birth are distributed like a Poisson point process of unit rate. Then,

$$m(\theta) = \sum_{j=1}^{\infty} \left(\frac{1}{1+\theta}\right)^j = \frac{1}{\theta}.$$

Therefore,

$$\mu(a) = \inf\left\{\frac{e^{\theta a}}{\theta} : \theta > 0\right\}.$$

The minimum occurs at $\theta = 1/a$, so that

$$\mu(a) = ea.$$

Thus $\gamma = 1/e$. The study of L_n is equally simple, as L_n is distributed as $E_1/1 + E_2/2 + \cdots + E_{n-1}/(n-1)$. To see this, note that if k elements are alive, the time until the next birth is distributed as E_k/k, as the minimum of k independent exponential random variables. Thus, as before, $L_n/\log n \to 1$ almost surely. It is easily seen that $H_n/\log n \to 1/\gamma = e$ almost surely. This result for the uniform random recursive tree was first obtained in Devroye (1987).

Our fourth example involves the plane-oriented recursive tree. In that tree, if a node u has degree $d(u)$, then its probability of making a child is proportional to $1 + d(u)$. This is like saying that the children of the root are born with inter-birth times distributed like E_1, $E_2/2$, $E_3/3$, and so forth. A simple computation shows that

$$m(\theta) = \sum_{i=1}^{\infty} \prod_{j=1}^{i} \left(\frac{j}{j+\theta}\right).$$

The computation of γ is a bit more complicated (see Pittel (1994) or Mahmoud (1994)). However, the inter-birth times are easy to deal with. Indeed, the sum of the intensities of the birth process is $\sum_u (1 + d(u)) = 2|t| - 1$ where $|t|$ denotes the number of nodes. Therefore, the inter-birth times for the tree are distributed like $E_1/1, E_2/3, \ldots$. Hence, it is not hard to show that $L_n/\log n \to 1/2$ almost surely, so that $H_n/\log n \to 1/(2\gamma)$ almost surely.

In the random m-ary pyramid, we have $m(\theta) = (1 - (1+\theta)^{-m})/\theta$. One can easily see that for $m = 2$, $\alpha = (\sqrt{5} - 1)/2$ (Theorem 5.2), but γ requires numerical computation. See Mahmoud (1994).

Finally, for the linear recursive trees, Pittel (1994) and Biggins and Grey (1996) show that $m(\theta) = \frac{1}{\theta - \beta}$ for $\theta > \beta$, so $\alpha = 1 + \beta$, $\mu(a) = ae^{1+\beta a}$, and γ is the unique root of $ae^{1+\beta a} = 1$. Thus $H_n/\log n \to 1/(\gamma(\beta + 1))$ almost surely as $n \to \infty$.

In a Bellman-Harris set-up the whole litter is born simultaneously at time T. If there are b children per parent, then we have $m(\theta) = b\mathbf{E}(e^{-\theta T})$. When T is exponential and $b = 2$, this is the celebrated Yule process. Clearly, $m(\theta) = 2/(1 + \theta)$, exactly as for the binary search tree discussed earlier. Thus, the height behaves in a manner similar to that of the binary search tree, even though the two processes are very different indeed. When T is not necessarily exponential, and the litter size follows a general distribution, we obtain the Bellman-Harris branching process, which is the subject of the next section.

§ 4. The Bellman-Harris Branching Process

In 1952, Bellman and Harris described a generalization of the Galton-Watson branching process by embedding it in continuous time. The (so-called age-dependent branching) process is described by two parameters, a discrete distribution $\{p_i, i \geq 0\}$ for the number of children, as in a standard Galton-Watson process, and a distribution of a strictly positive random variable T, the time between birth and reproduction. With each edge in the Galton-Watson tree, we associate an independent copy of T. The process is started with a single root at time 0. The elements are still grouped in generations. The root element produces a litter of size determined by $\{p_i\}$ after a time T_1 distributed as T. Each individual in the litter reproduces in the same manner and independently.

This model can also be used for describing the growth of the random binary search tree. We take the point of view that we let the random binary search tree grow by at each iteration picking an external node uniformly and at random. This node becomes an internal node, gets removed from the pool of external nodes, and produces two new external nodes, its potential children. At any moment, there are n internal nodes if and only if there are $n + 1$ external nodes. If T is standard exponential, then given that there are k external nodes at time t, by the memoryless property of the exponential distribution, we in fact pick as our next node any external node with equal probability. Thus, the order in which the nodes are chosen is identical to used for growing the random binary search tree. In notation of the previous section, the tree obtained at the time when there are exactly $n+1$ external nodes is a random binary search tree on n internal nodes. Recall that the process in which T is exponential and the number of offspring is always two is the Yule process, or binary fission (Athreya and Ney, 1972, p. 109). For different distributions of T, we obtain different kinds of random binary trees. We will not explore the Yule process construction of random binary search trees any further, except for the mention of the following theorem below, valid when T is standard exponential.

Theorem 5.4. *Assume that $\{p_i\}$ has finite second moment and that T is standard exponential. Let $Z(t)$ be the number of particles alive at time t in a Bellman-Harris process. Then $Z(t)e^{-t}$ tends almost surely to a random variable W,*

$$\frac{Z(t)-e^tW}{\sqrt{Z(t)}} \xrightarrow{\mathcal{L}} N(0,\sigma^2)$$

where $\sigma^2 = \mathrm{var}(W)$. Finally, conditional on W, $U(t) \overset{\mathrm{def}}{=} Z(\log(1+t/W))$ is a unit rate Poisson process. That is, for any $0 < t_1 < \cdots < t_k < \infty$ and integers $n_i \ge 0$, $1 \le i \le k$, and Borel set $B \subseteq [0,\infty)$,

$$\begin{aligned}\Pr\{U(t_1)-U(t_0) = n_1, \ldots, U(t_k)-U(t_{k-1}) = n_k, W \in B\}\\ = \Pr\{W \in B\}\textstyle\prod_{i=1}^k \Pr\{P(t_i-t_{i-1}) = n_i\}\end{aligned}$$

where $P(u)$ is a Poisson (u) random variable. Furthermore, $U(0) = Z(0) = 1$. For the Yule process, the random variable W has the standard exponential distribution.

The Poisson representation in the theorem above is due to Kendall (1966). If T is standard exponential, as in the Yule process, $Z(0) = 1$ and $Z(t)$ increases by one each time a particle gets replaced (as one dies and two are born). Two interesting properties of the exponential distribution are the following: if $E_1, E_2, \ldots$ are i.i.d. exponential random variables, then

A. For any n, $\min(E_1, \ldots, E_n) \overset{\mathcal{L}}{=} \frac{E_1}{n}$.

B. (The memoryless property.) For any $t > 0$, $E_1 - t$ given $E_1 > t$ is distributed as E_1.

Thus, the intervals between times of birth in a Yule process are distributed like $E_1, E_2/2, E_3/3, \ldots$. Using these two properties repeatedly we have

$$\begin{aligned}\Pr\{Z(t) > k\} &= \Pr\{E_1 + E_2/2 + E_3/3 + \cdots + E_k/k \le t\}\\ &= \Pr\{\max(E_1, E_2, \ldots, E_k) \le t\}\\ &= (1-e^{-t})^k\end{aligned}$$

so that everything is known about the distribution of $Z(t)$. For example,

$$\mathbf{E}(Z(t)) = \sum_{k\ge 0}\Pr\{Z(t) > k\} = e^t.$$

In fact, at any t, $Z(t)$ has the geometric distribution with parameter e^{-t}.

6. Conditional Branching Processes

6.1 Introduction

Of particular interest is the conditional Galton-Watson process, or conditional branching process, or simply cbp, in which we condition on $N = n$, where $N = \sum_{i=0}^\infty Z_i$ is the total size of the population, Z_i is the size of the population in generation i, and $Z_0 = 1$. These processes were studied by Kennedy (1975) and Kolchin (1978, 1986), who made key connections between them and so-called simply generated random trees, introduced by Meir and Moon (1978). These trees are uniformly picked in given collections such as, for example, all binary trees on n nodes.

Several examples will be given in the next section. In the other sections we review some results for the distribution, size and height of the trees in this model.

Consider a multiset of trees, that is, a set in which repetitions are allowed. Let the weight $\Omega(t)$ of a tree t be the number of occurrences of t. Let $|t|$ denote the size of t, i.e., the number of nodes contained in t. Then

$$a_n = \sum_{|t|=n}\Omega(t)$$

is the number of trees in this multiset with n nodes. The *generating function* for $\{a_n\}$ is denoted by

$$y(z) = \sum_{n\ge 0} a_n z^n.$$

We define a *uniform tree* T_n of size n by

$$\Pr\{T_n = t\} = c\,\Omega(t)\mathbb{1}_{[|t|=n]} = \frac{\Omega(t)}{a_n},$$

where $c > 0$ is a normalization constant. Thus, each of the a_n occurrences of elements in the multiset of trees of size n has the same probability. Therefore it is appropriate to speak of a uniform model if we can somehow distinguish patterns with $T(t)$ copies of t thrown into the multiset. This is illustrated in the next section.

A particularly interesting multiset of trees is the *simply generated family of trees* (Meir and Moon, 1978), which requires a descriptor

$$\varphi(y) = \sum_{i=0}^{\infty} c_i y^i ,$$

where $c_0 > 0$, and the c_i's are nonnegative integers (usually, but not necessarily, uniformly bounded in i). The notation φ, y and c_i is by now standard, so we will adopt it as well. Consider ordered trees, that is, trees in which the order of the children matters. For each ordered tree t, let $D_i(t)$ be the number of nodes in t with i children (successors). Then define

$$\Omega(t) \stackrel{\text{def}}{=} \prod_{i\ge 0} c_i^{D_i(t)}$$

The family of trees is *aperiodic* if $\gcd\{i > 0 : c_i > 0\} = 1$, and *periodic* otherwise. We define a random simply generated tree T_n of size n by

$$\Pr(T_n = t) = c\,\Omega(t)1_{|t|=n}$$

where c is a normalization constant. We note here that because we have ordered trees,

$$y(z) = z\varphi(y(z)) .$$

A proof is given in Theorem 5.4.

Next, we define a Galton-Watson branching process with parameter $\theta > 0$ with offspring distribution

$$p_i = \frac{c_i\theta^i}{\varphi(\theta)} ,\quad i \ge 0 .$$

Here we assume that $\varphi(\theta) < \infty$. It is easy to verify that $(p_0, p_1, \ldots)$ is indeed a probability vector. Furthermore, the expected number of offspring, an increasing function of θ, is

$$\sum_{i\ge 1} i p_i = \sum_{i\ge 0} \frac{i c_i \theta^i}{\varphi(\theta)} = \frac{\theta\varphi'(\theta)}{\varphi(\theta)} .$$

Let τ be the smallest positive root of $\varphi(\tau) = \tau\varphi'(\tau)$. Then for $\theta = \tau$, the branching process is critical, while for $0 < \theta < \tau$, it is subcritical. We now define CGW with parameter θ as the above Galton-Watson process conditioned on the total population size n, and let T_n' denote a realization of CGW.

The crucial properties of the two random trees defined above are captured in Theorem 8.1, which states that the conditioned Galton-Watson tree T_n' has the same distribution as the random simply generated tree.

Theorem 8.1. (Kennedy, 1975) *The distribution of T_n' is independent of $\theta \in (0, \tau]$. Furthermore, $T_n \stackrel{\mathcal{L}}{=} T_n'$, where $\stackrel{\mathcal{L}}{=}$ denotes equality in distribution.*

Proof. The first statement follows from the second one. Let t be an arbitrary fixed ordered tree with $|t| = n$. Let T^* be a family tree produced by the (unconditional) Galton-Watson process. Then

$$\begin{aligned}\Pr(T^* = t) &= \prod_{i\ge 0} (\Pr\{Z_1 = i\})^{D_i(t)} \\ &= \prod_{i\ge 0} \left(\frac{c_i\theta^i}{\varphi(\theta)}\right)^{D_i(t)} \\ &= \prod_{i\ge 0} c_i^{D_i(t)} \times (\varphi(\theta))^{-\sum_i D_i(t)} \times \theta^{\sum_i iD_i(t)} \\ &= \Omega(t) \times (\varphi(\theta))^{-n} \times \theta^{n-1} \\ &= \Omega(t) \times (\varphi(\theta))^{-n} \times \theta^{n-1} .\end{aligned}$$

Also

$$\begin{aligned}\Pr(|T^*| = n) &= \sum_{t:|t|=n} \Pr(T^* = t) \\ &= \sum_{t:|t|=n} \Omega(t)(\varphi(\theta))^{-n} \times \theta^{n-1} \\ &= a_n(\varphi(\theta))^{-n} \times \theta^{n-1} ,\end{aligned}$$

where a_n is the number of trees in the multiset of size n. Therefore, with $|t| = n$,

$$\Pr(T^* = t \mid |T^*| = n) = \frac{\Pr(T^* = t)}{\Pr(|T^*| = n)} = \frac{\Omega(t)}{a_n} .$$

But this is proportional to $\Omega(t)$, so that T_n' is indeed distributed as T^* conditioned on $|T^*| = n$, that is, as T_n. □

Trees are used in symbolic computations to represent formulas, with internal nodes representing operators or functions, and leaves operands. These are also called expression trees in the literature on parsing and the evaluation of expressions in higher level languages. In the analysis of such objects, it is natural to assume that all objects are equally likely. For example, in ordinary trigonometric expressions on three operands, x, y and z, there are internal nodes with two children (+ and −), internal nodes with one child (sin, cos, tan, cot), and leaves with zero children (x, y and z). The nodes are thus labeled, with a different number of labels according to the type of node. In the formalism of the previous section, we have $c_0 = 3$, $c_1 = 4$ and $c_2 = 2$. As $y(z) = z\varphi(y(z))$, we may get exact or asymptotically accurate expressions by analytic methods: see Vitter and Flajolet (1990) for a survey of such methods based on Lagrange inversion and singularity analysis. For expected values of various additive parameters, this is indeed a natural route to follow.

6.2 Examples of Trees in the Uniform Random Tree Model

(1,1). Several choices of descriptors lead to various types of trees. Consider first the choice (1,1). The weight of a tree is one for every tree consisting of just leaves and one-child nodes. Thus, the multiset will contain one of each of these trees, which in fact are just linked chains. The GWP has probability vector

$$\left(\frac{1}{1+\theta},\frac{\theta}{1+\theta}\right).$$

But clearly, conditioned on the size of the tree being n, we see that it does not matter which θ we picked. The tree has height exactly $n-1$. One can easily verify that the same result would have been obtained if we had selected the descriptor (a,b) for any $a,b>0$. Therefore, interesting trees only occur when $c_i>0$ for some $i>1$.

(1,0,1). The next simplest choice is (1,0,1), giving us a pleasant set of trees: trees with only leaves and two-child nodes. Such trees must have an odd cardinality. If $|t|=2k+1$, there are necessarily $k+1$ leaves and k two-child nodes. The weight of each tree of size $n=2k+1$ is thus identical, and equal to 1 (as all nonzero c_i's are one). Hence, each tree in the multiset is different, and all possible trees of the type described above are present. The family is the family of full binary trees. Again, all such trees occur equally often in the multiset.

(1,0,m). If we take $(1,0,m)$, then the weight of each tree of size $n=2k+1$ is m^k, and within this class, all trees occur equally often in the multiset. Therefore, there is no difference between random simply generated trees for $(1,0,m)$ for any $m>0$.

(1,2,1). The next member on the ladder of complexity is (1,2,1). Here we have trees with nodes having up to two children, and the weight of a tree with n nodes of which there are ℓ leaves is given by $2^{n-(2\ell-1)}$ as the number of nodes with two children is $\ell-1$. Interestingly, not all trees with n nodes have equal representation. We can however force equal weight on them by additional ways of distinguishing between trees. For example, for each node with one child, we may make the child a left child or a right child of its parent. For a tree with $n-(2\ell-1)$ such nodes, there are $2^{n-(2\ell-1)}$ possible combinations of left/right distinctions. Let us associate exactly one of these combinations to each of the $2^{n-(2\ell-1)}$ trees with n nodes and ℓ leaves in our multiset. Then, each tree in the multiset is distinct, and is in fact an ordinary binary tree. And all binary trees on n nodes are included in the multiset. An equivalent multiset (for our purposes) would have been obtained with the choice $(1,2m,m^2)$ for any $m>0$. We will also refer to these trees as Catalan trees.

(1,m,1). If we pick $(1,m,1)$, then it is necessary to create a designation for each single child, and we could associate a label between 1 and m with each such lone child. This creates a bijection between all such "labeled" trees with up to two children per node and the trees in the multiset. With $m=1$, labeling is superfluous, and one obtains the so-called unary-binary trees, which are the ordered trees with up to two children per node.

$(1,m,m^2)$. If we pick $(1,m,m^2)$, then we color each child in one of m colors, and note that with all possible colorings, all trees in the multiset occur only once, and that there is a bijection. The family is that of trees with up to two children per node and all nodes except the root are colored in one of m colors. In the GWP, we may set $\theta=1/m$ to obtain the reproduction distribution $(1/3,1/3,1/3)$. Thus, the shape properties of all these trees are identical, regardless of the choice of m.

Binomial. Position trees of branch factor b are those in which each node has up to b children and each child is given a position, and only one child can occupy each position. With $b=2$, this yields the binary trees. For general b, it is not hard to see that the descriptor must be binomial of the form $\left(1,\binom{b}{1},\binom{b}{2},\ldots,\binom{b}{b-1},\binom{b}{b}\right)$. Ternary trees are obtained by using the descriptor $(1,3,3,1)$, for example.

$(1,1,1,\ldots)$ or geometric. All ordered trees without restrictions on the number of children are obtained by the infinite descriptor $(1,1,1,\ldots)$. These are also called unlabeled rooted ordered trees or unlabeled planted plane trees, or unlabeled rooted plane trees, or just rooted plane trees. For the GWP, we must take $\theta<1$, so that $\psi(\theta)=1/(1-\theta)$, and the basic reproduction distribution is given by $(1/(1-\theta),\theta/(1-\theta),\ldots,\theta^i/(1-\theta),\ldots)$. This is a geometrically decreasing probability vector. From Theorem 5.1, we note that any $\theta\in(0,1)$ yields the same random tree in the conditional branching process model. We might thus as well take $\theta=1/2$. It takes just a moment to verify that all unlabeled rooted plane trees with non-root nodes colored in one of m colors are obtained from $(1,m,m^2,m^3,\ldots)$. For the GWP, we require therefore $\theta<1/m$. But then the GWP is exactly as in the case $m=1$ (geometric), and thus this choice of descriptor is equivalent to $(1,1,1,\ldots)$ if we want to study shape properties of the trees, unrelated to color choices.

$(1,0,0,\ldots,1)$. If the only nonzero coefficients are the 0-th and the t-th, with $t>1$, we obtain the so-called t-ary trees of Flajolet and Odlyzko [FO82].

$(1,1,2,3,4,5,\ldots)$. A node with i children gets a label between 1 and i which may indicate which of its children (in the ordered tree) is "best". We will call these trees favorite son trees.

If we remove structure in the order, by removing the order of the children altogether, or by replacing the total order by a circular order or a partial ordering, we in fact allow c_i's to take values less than one. This will not be pursued here. See, however, the section on Cayley trees where a connection is made with Poisson-distributed GWP's.

2.3 Catalan Trees and Dyck Paths

There are especially pretty derivations of the equivalence between a GWP and a uniform random Catalan tree. We first consider a nonnegative random walk in which all steps are +1 or −1, we start at $X_0 = 0$, and come to $X_{2n} = 0$. If we replace +1 and −1 by a and b respectively, then the sequence of $2n$ symbols thus obtained is a Dyck word. The walk is also called a Dyck path. If a_n is the number of different Dyck paths of length $2n$, by conditioning on the place $2p$ of the first return to the origin, we have

$$a_n = \sum_{p=0}^{n-1} a_p a_{n-1-p}$$

and $a_0 = 1, a_1 = 1$. It is well known that

$$a_n = \frac{1}{n+1}\binom{2n}{n},$$

the n-th Catalan number. There is a bijection between a Dyck path of length $2n$ and a binary tree on n nodes. Draw the binary tree in the standard manner. Write an a to the left of every node, and a b underneath each node. Then start at the root and walk around the tree by following edges just like a boat would follow the shoreline, and note the sequence of a's and b's. The order of visit is called preorder. The sequence forms a Dyck word as the number of a's at any point must exceed the number of b's. This bijection is useful for many purposes but for the study of parameters as the height of the random binary tree, some extra work is needed. We just note that the rooted binary trees were correctly counted as far back as Cayley (1858).

Another bijection may be considered, but now with rooted ordered trees with $n+1$ nodes (and thus n edges), by placing next to each edge an a to the left and a b to the right, and forming a Dyck word by the walk of the former bijection. This walk will be referred to as a Harris walk. The correspondence with a GWP can be seen as follows. Let $X_1, X_2, \ldots$ be i.i.d. random variables taking the values −1 and +1 with equal probability. Let $S_n = \sum_{i=1}^{n} X_i$ be the partial sums. Consider only $X_1 = 1$. Define ρ as the time of the first return to zero: $\rho = \inf\{n : S_n = 0\}$. Let $\rho_1, \ldots, \rho_N$ be the times less than ρ with $S_n = 1$. We set $\rho_0 = 1$, and note that $\rho_N = \rho - 1$. Define $t_1 = \rho_1 - \rho_0$, $t_2 = \rho_2 - \rho_1$, and so forth. Note that

$$\Pr\{N = k\} = \frac{1}{2^{k+1}},$$

where $\Pr\{\cdot\}$ denotes always conditional probability given $X_1 = 1$. This is best seen by noting that at each passage at one, the random walk has exactly 50% probability of returning to the origin. Thus, N is indeed geometrically distributed of parameter 1/2. Furthermore, given $N = k \ge 1$, the excursions above one of lengths $t_1, \ldots, t_N$ are independent and have the same distribution as the original positive excursion $S_1, \ldots, S_\rho$. This is just a manifestation of the strong Markov property applied to the ordinary random walk. We now construct the corresponding ordered tree explicitly: take a root, and give it N children, and associate with the children the positive excursions of lengths $t_1, \ldots, t_N$ respectively. Constructed in this manner, we note that the corresponding tree is nothing but a critical Galton-Watson tree with reproduction distribution $\Pr\{Y = k\} = 1/2^{k+1}$, $k \ge 0$. The bijection is formidable as it not only yields the desired connection, but it also is rather direct: for example, the maximum of an excursion corresponds to the height of the Galton-Watson tree, and the length of an excursion is twice the size of the Galton-Watson tree.

One may use the well-known bijection between rooted ordered trees on $n+1$ nodes and binary trees on n nodes. First copy all $n+1$ nodes from the ordered tree to the binary tree; then associate each parent-oldest child edge in the ordered tree with a parent-left child edge in the binary tree, and associate with each node-next sibling relationship in the ordered tree a parent-right child edge in the binary tree. Finally, remove the root and its left edge from the binary tree. This yields yet another (but slightly more indirect) bijection between Dyck paths and binary trees. The GWP connection follows easily: if N is the number of children of the root in the ordered tree, then the binary tree's root (before removal) has a left child if $N > 0$. A node in the ordered tree regarded as a child in a family has a number Y of younger siblings that is again geometric (1/2) by the memoryless property of the geometric distribution. Thus, it has a right child in the binary tree if $Y > 0$. To make a Galton-Watson process, place in the ordered tree a pair $(U, V) = (I_{N>0}, I_{Y>0})$, and observe that all these pairs in the tree are independent, and that U and V are also independent. Thus, the binary tree with a random number of nodes and after removal of the root is indeed a Galton-Watson tree with reproduction distribution $(p_0, p_1, p_2) = (1/4, 1/2, 1/4)$.

We should also mention that for symmetric random walks with zero mean having continuous distributions, Le Gall (1989) has proposed a beautiful tree construction that leads once again to a binary Galton-Watson tree with $(p_0, p_1, p_2) = (1/4, 1/2, 1/4)$.

6.4 Cayley Trees

The uniform random labeled tree C_n is the tree picked uniformly from the n^{n-2} trees on vertices $\{1,2,\ldots,n\}$. The uniform random rooted labeled tree (or rooted Cayley tree) R_n is the tree picked uniformly from the n^{n-1} trees on vertices $\{1,2,\ldots,n\}$ in which one vertex is declared to be the root. Cayley (1889) studied C_n and Riordan (1960) counted various related species of trees, including R_n. Rényi and Szekeres (1967) showed that the expected height H_n of R_n is $\sim \sqrt{2\pi n}$. They also showed that the limit distribution of $H_n/\sqrt{n}$ is the theta distribution (see further on). Rényi (1959) showed that the number of leaves is asymptotic to n/e, while Meir and Moon (1970) showed that the expected distance between two nodes taken at random is asymptotic to $\sqrt{\pi n/2}$.

Kolchin (1986), just like Meir and Moon (1978) and Moon (1970), studies C_n and R_n via generating functions, establishing a tight relationship with CGWP's. More probabilistic approaches may be found in Grimmett (1980) and Aldous (1988, 1991). The purpose of this section is to point out the key results in the latter papers.

Consider a Poisson (1) Galton-Watson tree P. Make P a labeled tree by randomly labeling the vertices $1,\ldots,|P|$. If t is a specific rooted labeled tree (having k vertices), then

$$\mathbf{Pr}(P = t) = \frac{e^{-k}}{k!}\,.$$

To see this, order all the sets of siblings in t by increasing labels, and let $N_1,\ldots,N_k$ be the number of children of all nodes listed in preorder. Then

$$\mathbf{Pr}(P = t) = \prod_{i=1}^{k} \frac{1}{eN_i!}\,\frac{\prod_{i=1}^{k} N_i!}{k!}$$

where the first factor accounts for matching the geometrical layout of the tree (it uses the independence of the number of offspring as well as the Poisson property) and the second factor is the probability of getting the random labels just right. Therefore, conditional on $|P| = n$, we see that P is uniform on labeled trees of size n, and is thus distributed as R_n. This property allows us to study the CGWP with Poisson (1) offspring. The calculation above establishes the connection and may be made into a construction of R_n. The theorems about CGWP's then provide information on random Cayley trees.

There is a second construction due to Aldous (1988). It requires i.i.d. random variables $U_2,\ldots,U_n$ uniformly distributed on $\{1,\ldots,n\}$. First we make 1 the root. Then with i varying from 2 to n, we add edge $(i,\min(i-1,U_i))$. Then we remove the labels to obtain a random rooted (nonuniform) unlabeled tree. It can be made into a tree distributed as R_n by randomly assigning labels.

Grimmett (1980) proposes yet another related process, and Aldous (1991) builds on it to derive a tool for studying local properties of such trees. For each $k = 0,1,2,\ldots$, we create independent Poisson (1) Galton-Watson trees, regarded as trees with roots r_k and other vertices unlabeled. Then we connect $r_0, r_1, r_2, \ldots$ as a path, make r_0 the root, and delete the labels. For fixed k, the restriction to copies of P is close in total variation distance to a random rooted unlabeled tree with a distinguished path of length $k-1$ attached to it. This connection will not be explored here.

Finally, we mention the Prüfer codes that are so useful in the generation and counting of all labeled trees (rooted or unrooted). The properties that may be deduced based on these codes are not directly linked to branching processes, and will thus not be studied here.

6.5 Fringe Subtrees

Following Aldous (1991), for a finite rooted ordered tree T, we call T^* the subtree rooted at a randomly and uniformly picked vertex from T. Aldous observed that in many (random or nonrandom) tree models, T^* tends in distribution to a certain random tree as $|T| \to \infty$. This has of course immediate consequences for the parameters of T^*. For example, we have the following (see Aldous, 1991):

Theorem 6.5. *Let ξ be an offspring distribution of a Galton-Watson process with $\mathbf{E}(\xi) = 1$, $\mathbf{Pr}(\xi = 1) < 1$, $\mathbf{E}\{\xi^2\} < \infty$ and ξ non lattice. Let T be the Galton-Watson tree (note: $|T| < \infty$ almost surely), and let T_n be T conditional on $|T| = n$. Let T_n^* be a tree rooted at a random vertex of T_n. Then for all trees t*

$$\lim_{n\to\infty} \mathbf{Pr}(T_n^* = t) = \mathbf{Pr}(T = t)\,.$$

Discussion. In this remarkable result, we see that the limit distribution of a fringe tree of the CGWP is the unconditional Galton-Watson tree! As a result, we may immediately deduce properties of local parameters from this. For example, the degree of a random vertex in a CGWP tends in distribution to the degree of the root of T, that is, ξ. Also, $|T_n^*| \xrightarrow{\mathcal{L}} |T|$. Note also that the number of vertices in a CGWP within distance k of a uniform random vertex tends in distribution to the number of vertices within distance k of the root of T^{∞}, that is, $Z_0 + Z_1 + \cdots + Z_k$, where $Z_0, Z_1, \ldots$ are the population sizes in the tree T^{∞}.

6.6. Size of a Galton-Watson Tree

Let T be a Galton-Watson tree that is either critical or subcritical. We know that if ξ is the offspring distribution and $\mathbf{P}\{\xi = 1\} < 1$, then $|T| < \infty$ almost surely. In fact, it is remarkable that the distribution of $|T|$ can be solely deduced from the distribution of ξ by a simple device discovered by Dwass (1969) and rediscovered by Kolchin (Kolchin, 1977, 1978, 1986; see 1986 p. 104).

Theorem 6.3. *For $n \geq 1$,*

$$\mathbf{P}\{|T| = n\} = \frac{\mathbf{P}\{\xi_1 + \cdots + \xi_n = n-1\}}{n}$$

where $\xi_1, \xi_2, \ldots$ are i.i.d. and distributed as ξ. Let $T_1, T_2, \ldots$ be independent and distributed as T. Then, for $n \geq m > 0, n \geq 1$,

$$\mathbf{P}\{|T_1| + \cdots + |T_m| = n\} = \frac{m\mathbf{P}\{\xi_1 + \cdots + \xi_n = n-m\}}{n}.$$

Proof. It suffices to prove the more general statement. Clearly, if Z_1 is the number of offspring of the root of T_1, assuming $m > 1$, we have

$$\begin{aligned}\mathbf{P}\{|T_1| + \cdots + |T_m| = n\} &= \sum_{j=0}^{\infty} p_j \mathbf{P}\{|T_1| + \cdots + |T_m| = n \mid Z_1 = j\}\\ &= \sum_{j=0}^{n-m} p_j \mathbf{P}\{|T_2| + \cdots + |T_{m+j}| = n-1\},\end{aligned}$$

where $p_j = \mathbf{P}\{\xi = j\}$ and $Z_1 = \xi$ is the number of children of the root. We easily verify the lemma for $n = 1$ and $m = 1$, as then $\mathbf{P}\{|T| = 1\} = \mathbf{P}\{\xi_1 = 0\}$. The remainder is by induction on n (for all $0 \leq m \leq n$), and we have

$$\begin{aligned}\mathbf{P}\{|T_1| + \cdots + |T_m| = n\} &= \sum_{j=0}^{n-m} p_j \mathbf{P}\{|T_2| + \cdots + |T_{m+j}| = n-1\}\\ &= \sum_{j=0}^{n-m} p_j \frac{m+j-1}{n-1} \mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_{n-1} = n-m-j\}\\ &\quad \text{(by the induction hypothesis)}\\ &= \frac{m-1}{n-1}\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_n = n-m\}\\ &\quad + \frac{1}{n-1}\sum_{j=0}^{n-m} jp_j \mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_{n-1} = n-m-j\}\\ &= \left(\frac{m-1}{n-1} + \frac{n-m}{n(n-1)}\right)\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_n = n-m\}\\ &\quad \text{(see below)}\\ &= \frac{m}{n}\mathbf{P}\{\xi_1 + \xi_2 + \cdots + \xi_n = n-m\}.\end{aligned}$$

We are done if we can explain the last step. But clearly,

$$\begin{aligned}\frac{n-m}{n} &= \mathbf{E}\{\xi_n \mid \xi_1 + \cdots + \xi_n = n-m\}\\ &= \frac{\sum_{j=0}^{n-m} j\mathbf{P}\{\xi_n = j\}\mathbf{P}\{\xi_1 + \cdots + \xi_{n-1} = n-m-j\}}{\mathbf{P}\{\xi_1 + \cdots + \xi_n = n-m\}}.\end{aligned}$$

This concludes the proof of Theorem 6.3. □

Theorem 6.3 makes a crucial connection with sums of independent random variables, and for this all is known. For example, following Kolchin (1986, p. 105), we note that if ξ has mean one (as in a critical branching process), variance σ^2 and maximal span d, when $n - 1$ tends to infinity over multiples of d,

$$\mathbf{P}\{|T| = n\} \sim \frac{d}{\sqrt{2\pi n^3}\,\sigma}.$$

It is easily seen that $\mathbf{E}\{|T|\} = \infty$, a result that also follows by noting that $|T| = \sum_{i=0}^{\infty} Z_i$ and $\mathbf{E}\{Z_i\} = 1$ for all i.

Finally, the size of a Galton-Watson tree may also be determined by analytic methods. Let $g(s)$ be the generating function of $|T|$. Then we have

Theorem 6.4. *The generating function $g(s) = \mathbf{E}\{s^{|T|}\}$ of $|T|$ satisfies*

$$g(s) = sf(g(s))$$

where f is the generating function of ξ in the Galton-Watson process.

Proof.

$$\begin{aligned}g(s) &= \mathbf{E}\{s^{|T|}\}\\ &= s\mathbf{E}\{s^{|T_1| + \cdots + |T_\xi|}\}\\ &= s\mathbf{E}\left\{\left(\mathbf{E}\{s^{|T|}\}\right)^{\xi}\right\}\\ &= s\mathbf{E}\{(g(s))^{\xi}\}\\ &= sf(g(s)).\end{aligned}$$

□

The asymptotic form of p_n, the n-th coefficient of $g(s)$, and thus $p_n = \mathbf{P}\{|T| = n\}$, may be obtained by singularity analysis (Meir and Moon, 1978; Pólya, 1937). For exact formulas, one may apply Lagrangian inversion and note that

$$p_n = \frac{1}{n} \times \text{coefficient of } u^{n-1} \text{ in } (f(u))^n.$$

See Vitter and Flajolet (1990) for more on this method, and for additional references.

5.7 Height of a Galton-Watson Tree

Let H_n be the height of a Galton-Watson tree T conditional on $|T| = n$. By equivalence, we will refer to these trees by the names used in the combinatorial literature, based on the equiprobable equivalent trees thus obtained.

It is known that $\mathbf{E}(H_n) \sim \sqrt{\pi n}$ for the planted plane trees (De Bruijn, Knuth and Rice, 1972), $\mathbf{E}(H_n) \sim \sqrt{2\pi n}$ for the rooted labeled trees (Cayley trees) (Rényi and Szekeres, 1967), $\mathbf{E}(H_n) \sim \sqrt{3\pi n}$ for the equiprobable unary-binary trees (Flajolet and Odlyzko, 1982) and $\mathbf{E}(H_n) \sim \sqrt{4\pi n}$ for the equiprobable binary trees (Flajolet and Odlyzko, 1982). For the last model, the expected depth of a random node is asymptotic to $\sqrt{\pi n}$ (Vitter and Flajolet, 1990). Rényi and Szekeres (1967) also computed the limit law for $H_n/\sqrt{n}$:

$$\lim_{n\to\infty} \mathbf{P}\left(\frac{H_n}{\sqrt{2n}} \le x\right) = H(x)$$

where

$$H(x) = \begin{cases} \frac{4\pi^{5/2}}{x^3}\sum_{j=1}^{\infty} j^2 e^{-\pi^2 j^2/x^2} \\ \sum_{j=-\infty}^{\infty}(1-2j^2x^2)e^{-j^2x^2} . \end{cases}$$

We will call H the theta distribution function. The theta distribution has first moment $\sqrt{\pi}$, variance $\pi(\pi-3)/3$ and general s-th moment $2\Gamma(1+s/2)s(s-1)\zeta(s)$. Interestingly, the theta distribution describes the limit for all simply generated random trees. This result, due to Flajolet and Odlyzko (1982), who used analysis of singularities of generating functions in their proofs, may be formulated as follows. Let $c_0, c_1, \ldots$ define the simply generated family of ordered trees, and let

$$y(x) = x\phi(y(x)) ,$$

where $y(x) = \sum_{n\ge1} y_n x^n$ and y_n is the total number of trees of size n, and $\phi(y) = \sum_{i\ge0} c_i y^i$.

Theorem 5.5 (Flajolet and Odlyzko, 1982). *For simple families of trees corresponding to the equation $y = x\phi(y)$ and for $n = 1 \bmod d$ with $d = \gcd(i : c_i \ne 0)$, if we set*

$$\sigma = \frac{2\phi'(\tau)^2}{\phi(\tau)\phi''(\tau)}$$

with τ the smallest positive root of the equation $\phi(\tau) - \tau\phi'(\tau) = 0$, we have

$$\frac{H_n}{\sqrt{\sigma n}} \stackrel{\mathcal{L}}{\to} H(\cdot) .$$

Furthermore, all the moments of $H_n/\sqrt{\sigma n}$ tend to those of H. In particular,

$$\lim_{n\to\infty} \frac{\mathbf{E}(H_n)}{\sqrt{n}} = \sqrt{\pi\sigma} .$$

The above result also applies to Cayley trees even though their generating functions do not satisfy the required equality. However, if $y(x) = \sum_{n\ge1} y_n x^n/n!$, then $y(x) = x\phi(y(x))$ with $\phi(y) = e^y$, which corresponds to the choice $c_i = 1/i!$. Combinatorialists know that $y(x) = xe^{y(x)}$ has a formal solution

$$y = \sum_{n=1}^{\infty} \frac{n^{n-2}}{(n-1)!} x^n$$

when $|x| \le 1/e$ (Riordan, 1968). From this, we also obtain the number of unrooted trees on n nodes.

By the remarks of the previous section, we note that indeed the limit law given above is applicable to random Cayley trees. In this case, we have

$$\sigma = \frac{2\phi'(\tau)^2}{\phi(\tau)\phi''(\tau)} = 2$$

for any value of τ. Hence, $\mathbf{E}(H_n) \sim \sqrt{2\pi n}$, a result due to Rényi and Szekeres (1967).

5.8 Components in Random Graphs

We conclude with Karp's (1990) construction of a branching process for studying the components of random graphs. We place this material here, as it relates to sizes of extinct branching processes. Random graphs were introduced by Erdős and Rényi in 1960: we have an edge probability p, possibly depending upon n, and call $G_{n,p}$ the graph on n labeled vertices obtained by independently adding each of the $\binom{n}{2}$ possible edges with probability p. Palmer (1985) gives a great account of the growth of $G_{n,p}$ as p increases. At least in the study of the behavior of $G_{n,p}$ for $p \le 1/n$, thus for sparse graphs, branching processes come in handy. So we set $p = c/n$, $c \le 1$. Around $p = 1/n$, $G_{n,p}$ undergoes a dramatic metamorphosis, as one giant component emerges which has size $\Theta(n)$ when $c > 1$. Karp's method is reconsidered in Alon, Spencer and Erdős (1992), where it is used to analyze the giant component in some detail (the case $c > 1$). We will fix $c < 1$ for simplicity.

Consider a fixed vertex u. We declare all other vertices alive, dead, or neutral. Originally, at discrete time $t = 0$, only u is alive and all other nodes are neutral. Let Y_t be the number of live nodes at time t. We set $Y_0 = 1$. Each time unit, we take a live vertex w and check all pairs (w, w') with w' neutral for membership in G. If (w, w') is indeed an edge, then we make w' live; after all such w' are examined, w dies, and we declare Y_t the new number of live vertices. When there are no live vertices ($Y_t = 0$), the process terminates.

and we equate $C(u)$, the component of u, as the collection of dead vertices. Clearly, we have

$$Y_t = Y_{t-1} + Z_t - 1 .$$

Each vertex w has independent probability p of becoming live and as pair $\{v,w\}$ is ever examined twice, so that the conditional probability of the membership of edge $\{v,w\}$ is always p. As $t-1$ vertices are dead and Y_{t-1} live, it is easy to see that

$$Z_t \overset{\mathcal{L}}{=} B(n-(t-1)-Y_{t-1}, p)$$

where $B(\cdot,\cdot)$ denotes the binomial distribution. Let T be the smallest t for which $Y_t = 0$, the time of extinction. Also, $T = |C(u)|$. We continue this definition recursively, and note that for all t,

$$Y_t \overset{\mathcal{L}}{=} B(n-1, 1-(1-p)^t) - t + 1 .$$

Proof. Define $N_t = n - t - Y_t$, the number of neutral vertices at time t. We will show that $N_t \overset{\mathcal{L}}{=} B(n-1,(1-p)^t)$. Clearly, $N_0 = n-1$. We argue by induction, and note that

$$\begin{aligned} N_t &= n - t - Y_t \\ &= n - t - B(n-(t-1)-Y_{t-1}, p) - Y_{t-1} + 1 \\ &= N_{t-1} - B(N_{t-1}, p) \\ &= B(N_{t-1}, 1-p) . \end{aligned}$$

□

The property above is valid for all p. For $p = c/n$, when t and Y_{t-1} are small, the binomial law is close to a Poisson law with mean c. So Z_t is close to $B(n, c/n)$, which is close to $P(c)$, a Poisson random variable with mean c. Thus, roughly speaking, the component grows as if it were a branching process with offspring distribution $P(c)$. For fixed c, let $Y_0^*, Y_1^*, \ldots, T^*, Z_1^*, Z_2^*, \ldots$ refer to the $P(c)$ branching process, and let the unstarred random variables refer to the random graph process. More precisely, the branching process starts with one live individual, so that $Y_0^* = 1$, and at each time unit, one live individual is selected at random. It produces a $P(c)$ number of children, and then dies, so that

$$Y_t^* = Y_{t-1}^* + Z_t^* - 1$$

where $Z_1^*, Z_2^*, \ldots$ are i.i.d. $P(c)$ random variables. Let T^* be the least t for which $Y_t^* = 0$. If no such t exists, we say that $T^* = \infty$. From Theorem 1.1, if $\mathbf{E}\{P(c)\} = c < 1$ with probability one, the process dies out, so that $T^* < \infty$ almost surely.

Let H, H^* denote the histories of the processes up to time t, that is, $H = (Z_1, \ldots, Z_t)$ and $H^* = (Z_1^*, \ldots, Z_t^*)$. Then

$$\Pr(H^* = (z_1, \ldots, z_t)) = \prod_{i=1}^{t} \Pr(P(c) = z_i)$$

and

$$\Pr(H = (z_1, \ldots, z_t)) = \prod_{i=1}^{t} \Pr(Z_i = z_i) ,$$

where Z_i is binomial $B(n-1-z_1-\cdots-z_{i-1}, c/n)$. If t, c and z_i are fixed, we have

$$\Pr(B(m, c/n) = z) \to \frac{c^z e^{-c}}{z!}$$

as $n \to \infty$. This may be used to show that

$$\lim_{n\to\infty} \Pr(H = (z_1, \ldots, z_t)) = \Pr(H^* = (z_1, \ldots, z_t)) .$$

Thus, for any fixed t, $\lim_{n\to\infty} \Pr(T = t) = \Pr(T^* = t)$. This may be used in several ways. First of all, T^* is the total size of a $P(c)$ Galton-Watson process. Therefore, as $n \to \infty$,

$$|C(u)| \xrightarrow{\mathcal{L}} T^* .$$

From Theorem 6.1, the generating function for $P(c)$ is $f(s) = e^{c(s-1)}$, while the generating function $g(s)$ for T^* is the solution of $y = sf(y)$, i.e., of

$$y = se^{c(y-1)} .$$

This describes the asymptotic distribution of the size of $C(u)$ in its entirety.

Secondly, if we consider $C_n = \max_u |C(u)|$ over the vertices u of $G_{n,c/n}$, then we can easily prove the known result (see Palmer, 1985) that $\Pr(C_n > \beta \log n) = o(1)$ for some $\beta > 0$. To see this, observe that for any t and for $h > 0$, by Chernoff's bounding method,

$$\begin{aligned} \Pr(T > t) &\le \Pr(Y_t > 0) = \Pr(B(n-1, 1-(1-p)^t) \ge t) \\ &\le \Pr(B(n, ct/n) \ge t) \le \mathbf{E}\{e^{h(B(n,ct/n)-t)}\} \\ &= e^{-ht}(1+(e^h-1)ct/n)^n \le e^{-ht+(e^h-1)ct} \\ &= e^{-t(\log(1/c)-1+c)} \quad (\text{take } h = \log(1/c)) \\ &\overset{\text{def}}{=} e^{-\alpha t} . \end{aligned}$$

Thus

$$\Pr(C_n > \beta \log n) \le n e^{-\alpha\beta\log n} = n^{1-\alpha\beta} \to 0$$

if we pick $\beta > 1/\alpha = 1/(\log(1/c) - 1 + c)$.

We leave it as an interesting exercise to show that the $P(c)$ branching process of this section, with $c > 1$, conditioned on extinction, has the same distribution as the (unconditional) $P(d)$ branching process, where $d = cq$

and q is the extinction probability of the $Po(c)$ branching process, that is, $q = e^{c(q-1)}$. (Note that $cqe^{-cq} = ce^{-c}$.) This fact is used in Alon, Spencer and Erdős (1992) to show for example that the structure of $G_{n,c/n}$ with the giant component removed is asymptotically that of $G_{m,cq/m}$ (without any removals), where m, the number of vertices not in the giant component, satisfies $m \sim nq$.

5.9 Bibliographic Remarks

Meir and Moon (1978) studied the expected depth $E(D_n)$ from root to nodes in simply generated random trees, and showed that $E(D_n)/\sqrt{n} \to c$, where c is again a constant only depending upon the species of tree. The work of Flajolet and Odlyzko (1982) is continued by Gutjahr (1993), who derives asymptotics for expected values of various other tree parameters such as the number of nodes at level k and the total path length. Even tree models with trees of given size and height are considered there. The branching process approach was used by Kennedy (1975) (see also Kolchin, 1986) to obtain the limit law for $Z_{\lfloor\theta\sqrt{n}\rfloor}/\sqrt{n}$ conditional on $N = n$ as $n \to \infty$, where Z_k is the size of the k-th generation. Thus, the bulk of the points is indeed at distance $\Theta(\sqrt{n})$ from the root. Finally, one might study the height of random binary trees, where each edge has an independent length drawn from a fixed distribution on the positive halfline. Height is then defined as the maximal sum of edge lengths of any path to the root. For the exponential distribution, Gupta, Mesa and Waymire (1990) showed that this height satisfies the same limit law as the standard height modulo a constant multiplicative factor. Their proof uses convergence of all moments.

References

1. Abramowitz M. (1970) and I. A. Stegun, Handbook of Mathematical Tables, Dover Publications, New York, NY.
2. Aho A. V. (1983) J. E. Hopcroft, and J. D. Ullman, Data Structures and Algorithms, Addison-Wesley, Reading, MA.
3. Aldous D. (1990): The random walk construction of uniform spanning trees and uniform labelled trees. SIAM Journal of Discrete Mathematics, 3, 450-465.
4. Aldous D. (1991): The continuum random tree II: an overview, in: Proceedings Durham Symposium on Stochastic Analysis, ed. M. T. Barlow and N. H. Bingham, 23-70, Cambridge University Press, Cambridge, UK.
5. Aldous D. (1991): The continuum random tree I, Annals of Probability, 19, 1-28.
6. Aldous D. (1993): The continuum random tree III, Annals of Probability, 21, 248-289.
7. Aldous D. (1994): Probability distributions on cladograms. Technical Report, Institute of Mathematics and Applications, University of Minnesota.
8. Aldous D., Flannery B. and Palacios J.L. (1988): Two applications of urn processes: the fringe analysis of search trees and the simulation of quasi-stationary distributions of Markov chains. Probability in the Engineering and Informational Sciences, 2, 293-307.
9. Alon N. (1992) J. H. Spencer, and P. Erdős, The Probabilistic Method, Wiley, New York.
10. Arkin E., Held M., Mitchell J. and Skiena S. (1994): Hamiltonian triangulations for fast rendering. In: Algorithms-ESA'94, ed. J. van Leeuwen, 855, 36-47, Lecture Notes in Computer Science, Springer-Verlag.
11. Asmussen S. and Hering H. (1983): Branching processes, Birkhäuser Verlag, Basel.
12. Athreya K.B. and Ney P.E. (1972): Branching Processes, Springer Verlag, Berlin.
13. Avis D. and ElGindy H. (1987): Triangulating point sets in space, Discrete Computational Geometry, 2, 99-111.
14. Bahadur R.R. and Rao R.R. (1960): On deviations of the sample mean, Annals of Mathematical Statistics, 31, 1015-1027.
15. Bell C.J. (1965): An investigation into the principles of the classification and analysis of data on an automatic digital computer. Doctoral Dissertation, Leeds University.
16. Bellman R. and Harris T.E. (1952): On age-dependent binary branching processes. Annals of Mathematics, 55, 280-295.
17. Bergeron F., Flajolet P., and Salvy B. (1992): Varieties of increasing trees, in CAAP 92, ed. J.-C. Raoult, 581, 24-48, Lecture Notes in Computer Science, Springer-Verlag.
18. Biggins J.D. (1976): The first and last-birth problems for a multitype age-dependent branching process. Advances in Applied Probability, 8, 446-459.
19. Biggins J.D. (1976): Asymptotic properties of the branching random walk, D.Phil. Thesis, University of Oxford.
20. Biggins J.D. (1977): Martingale convergence in the branching random walk, Journal of Applied Probability, 14, 25-37.
21. Biggins J.D. (1977): Chernoff's theorem in the branching random walk, Journal of Applied Probability, 14, 630-636.
22. Biggins J.D. (1978): The asymptotic shape of the branching random walk, Advances in Applied Probability, 10, 62-84.
23. Biggins J.D. (1979): Growth rates in the branching random walk, Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 48, 17-34.
24. Biggins J.D. (1990): The central limit theorem for the supercritical branching random walk, and related results, Stochastic Processes and their Applications, 34, 255-274.
25. Biggins J.D. (1995): The growth and spread of the general branching random walk, Annals of Applied Probability, 5, 1008-1024.
26. Biggins J.D. (1996): How fast does a general branching random walk spread? In: Classical and Modern Branching Processes, 84, 19-39, IMA Volumes in Mathematics and its Applications, Springer-Verlag, New York.
27. Biggins J.D. and Bingham N.H. (1993): Large deviations in the supercritical branching process, Advances in Applied Probability, 25, 757-772.
28. Biggins J.D. and Grey D.R. (1996): A note on the growth of random trees, Technical Report, School of Mathematics and Statistics, University of Sheffield, Sheffield, U.K.

29. Bingham N. (1988): On the limit of a supercritical branching process. Journal of Applied Probability, 25 A, 215-228.
30. Bramson M. (1983): Convergence of solutions of the Kolmogorov equation to travelling waves. Mem. American Mathematical Society, 285, 1-190.
31. Bramson M.D. (1978): Maximal displacement of branching Brownian motion. Communications on Pure and Applied Mathematics, 31, 531-581.
32. Bramson M.D. (1978): Minimal displacement of branching random walk. Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 45, 89-108.
33. Bramson M., Durrett R. and Swindle G. (1989): Statistical mechanics of crabgrass. Annals of Probability, 17, 444-481.
34. Brown G.G. and Purdom P.W. (1981): An average time analysis of backtracking. SIAM Journal of Computing, 10, 583-593.
35. Bruijn N.G. de, Knuth D.E. and Rice S.O. (1972): The average height of planted plane trees, in: Graph Theory and Computing, ed. R.C. Read, 15-22, Academic Press, New York.
36. Cayley A. (1857): On the analytical forms called trees. Philosophical Magazine, 28, 374-378.
37. Cayley A. (1889): A theorem on trees. Quarterly Journal of Pure and Applied Mathematics, 23, 376-378.
38. Chernoff H. (1952): A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. Annals of Mathematical Statistics, 23, 493-507.
39. Coffman E.G. and Eve J. (1970): File structures using hashing functions. Communications of the ACM, 13, 427-436.
40. Crump K.S. and Mode C.J. (1968): A general age-dependent branching process. Journal of Mathematical Analysis and its Applications, 24, 494-508.
41. Darling D.A. (1970): The Galton-Watson process with infinite mean. Journal of Applied Probability, 7, 455-456.
42. Dekking F.M. and Host B. (1991): Limit distributions for minimal displacement of branching random walks. Probability Theory and Related Fields, 90, 403-426.
43. Derrida B. and Spohn H. (1988): Polymers on disordered trees, spin glasses, and traveling waves. Journal of Statistical Physics, 51, 817-840.
44. Devroye L. (1986): A note on the height of binary search trees. Journal of the ACM, 33, 489-498.
45. Devroye L. (1986b): Non-Uniform Random Variate Generation, Springer-Verlag, New York.
46. Devroye L. (1987): Branching processes in the analysis of the heights of trees, Acta Informatica, 24, 277-298.
47. Devroye L. (1988): Applications of the theory of records in the study of random trees. Acta Informatica, 26, 123-130.
48. Devroye L. (1990): On the height of random m-ary search trees. Random Structures and Algorithms, 1, 191-203.
49. Devroye L. (1993): On the expected height of fringe-balanced trees. Acta Informatica, 30, 459-466.
50. Devroye L. (1997): Universal limit laws for depths in random trees. SIAM Journal on Computing, to appear.
51. Devroye L. and Kamoun O. (1996): Random minimax game trees, in: Random Discrete Structures, ed. D. Aldous and R. Pemantle, 55-80, John Wiley, New York.
52. Devroye L. and Laforest L. (1990): An analysis of random d-dimensional quadtrees. SIAM Journal on Computing, 19, 821-832.

53. Devroye L. and Reed B. (1995): On the variance of the height of random binary search trees. SIAM Journal on Computing, 24, 1157-1162.
54. Devroye L. and Zamora-Cura C. (1997): On the complexity of branch-and-bound search for random trees. Technical Report, McGill University.
55. Dharmadhikari S.W. and Jogdeo K. (1969): Bounds on moments of certain random variables. Annals of Mathematical Statistics, 40, 1506-1508.
56. Drmota M. (1997): An analytic approach to the height of the binary search tree, Technical Report, University of Vienna.
57. Durrett R. (1979): Maxima of branching random walks versus independent random walks. Stochastic Processes and Their Applications, 9, 117-135.
58. Durrett R. (1991): Probability: Theory and Examples, Wadsworth and Brooks, Pacific Grove, CA.
59. Dwass M. (1969): The total progeny in a branching process, Journal of Applied Probability, 6, 682-686.
60. Erdős P. and Rényi A. (1960): On the evolution of random graphs. Magyar Tud. Akad. Mat. Kutató Int. Közl., 5, 17-61.
61. Feller W. (1971): An Introduction to Probability Theory and its Applications, Volume 2, John Wiley, New York.
62. Finkel R.A. and Bentley J.L. (1974): Quad trees: a data structure for retrieval on composite keys, Acta Informatica, 4, 1-9.
63. Flajolet P., Gonnet G., Puech C. and Robson J.M. (1990): The analysis of multidimensional searching in quad-trees, in: Proceedings of the Second Annual ACM-SIAM Symposium on Discrete Algorithms, 100-109, ACM, New York, and SIAM, Philadelphia.
64. Flajolet P. and Lafforgue T. (1994): Search costs in quadtrees and singularity perturbation analysis, Discrete and Computational Geometry, 12, 151-175.
65. Flajolet P. and Odlyzko A. (1982): The average height of binary trees and other simple trees, Journal of Computer and System Sciences, 25, 171-213.
66. Flajolet P. and Odlyzko A. (1990): Singularity analysis of generating functions. SIAM Journal on Discrete Mathematics, 3, 216-240.
67. Flajolet P. and Sedgewick R. (1986): Digital search trees revisited, SIAM Journal on Computing, 15, 748-767.
68. Fredkin E.H. (1960): Trie memory, Communications of the ACM, 3, 490-500.
69. Fuk D.K. and Nagaev S.V. (1971): Probability inequalities for sums of independent random variables. Theory of Probability and its Applications, 16, 643-660.
70. Le Gall J.F. (1989): Marches aléatoires, mouvement Brownien et processus de branchement, in: Séminaire de Probabilités XXIII, ed. J. Azéma, P.A. Meyer and M. Yor, 1989, 258-274, Lecture Notes in Mathematics, Springer-Verlag, Berlin.
71. Gonnet G.H. and Baeza-Yates R. (1991): Handbook of Algorithms and Data Structures, Addison-Wesley, Workingham, England.
72. Gradshteyn I.S. and Ryzhik I.M. (1980): Table of Integrals, Series and Products, Academic Press, New York.
73. Grimmett G.R. (1980): Random labelled trees and their branching networks. Journal of the Australian Mathematical Society series A, 30, 229-237.
74. Grimmett G.R. and Stirzaker D.R. (1992): Probability and Random Processes, Oxford University Press.
75. Gupta V.K., Mesa O.J. and Waymire E. (1990): Tree-dependent extreme values: the exponential case, Journal of Applied Probability, 27, 124-133.
76. Gutjahr W. (1992): The variance of level numbers in certain families of trees, Random Structures and Algorithms, 3, 361-374.

77. Gutjahr W. (1993): Expectation transfer between branching processes and random trees, Random Structures and Algorithms, 4, 447–467.
78. Gutjahr W. and Pflug G.C. (1992): The asymptotic contour process of a binary tree is a Brownian excursion, Stochastic Processes and their Applications, 41, 69–89.
79. Gutjahr W. and Pflug G.C. (1992): The limiting common distribution of two leaf heights in a random binary tree, Theoretical Informatics and Applications, 26, 1–18.
80. Gutjahr W. and Pflug G.C. (1992): Average execution times of series parallel networks, Technical Report, University of Vienna.
81. Gutjahr W. and Pflug G.C. (1992): The asymptotic distribution of leaf heights in binary trees, Graphs and Combinatorics, 8, 243–251.
82. Hammersley J.M. (1974): Postulates for subadditive processes, Annals of Probability, 2, 652–680.
83. Harris T.E. (1963): The Theory of Branching Processes, Springer Verlag, Berlin.
84. Hawkes J. (1981): Trees generated by a simple branching process, Journal of the London Mathematical Society, 24, 373–384.
85. Heathcote C.R., Seneta E. and Vere-Jones D. (1967): A refinement of two theorems in the theory of branching processes, Theory of Probability and its Applications, 12, 297–301.
86. Heyde C.C. (1970): A rate of convergence result for the super-critical Galton-Watson process, Journal of Applied Probability, 7, 451–454.
87. Heyde C.C. (1971): Some central limit analogues for super-critical Galton-Watson processes, Journal of Applied Probability, 8, 52–59.
88. Heyde C.C. (1971): Some almost sure convergence theorems for branching processes, Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 20, 189–192.
89. Heyde C.C. and Brown B.M. (1971): An invariance principle and some convergence rate results for branching processes, Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 20, 271–278.
90. Jabbour J. (1996): Personal communication.
91. Jacquet P. and Régnier M. (1986): Trie partitioning process: limiting distributions, in: Lecture Notes in Computer Science, 214, 196–210.
92. Jagers P. (1975): Branching Processes with Biological Applications, John Wiley, New York.
93. Jagers P. and Nerman O. (1984): The growth and composition of branching populations, Advances in Applied Probability, 16, 221–259.
94. Janson S. (1983): Limit theorems for certain branching random walks on compact groups and homogeneous spaces, Annals of Probability, 11, 909–930.
95. Joffe A. and Waugh W.A.O'N. (1982): Exact distributions of kin numbers in a Galton-Watson process, Journal of Applied Probability, 19, 767–775.
96. Karp R.M. (1990): The transitive closure of a random digraph, Random Structures and Algorithms, 1, 73–93.
97. Karp R.M. and Pearl J. (1983): Searching for an optimal path in a tree with random costs, Artificial Intelligence, 21, 99–117.
98. Karp R.M. and Zhang Y. (1995): Bounded branching process and AND/OR tree evaluation, Random Structures and Algorithms, 7, 97–116.
99. Kemp R. (1984): Fundamentals of the Average Case Analysis of Particular Algorithms, B.G. Teubner, Stuttgart.
100. Kendall D.G. (1966): Branching processes since 1873, Journal of the London Mathematical Society, 41, 385–406.
101. Kennedy D.P. (1975): The Galton-Watson process conditioned on the total progeny, Journal of Applied Probability, 12, 800–806.
102. Kesten H., Ney P. and Spitzer F. (1966): The Galton-Watson process with mean one and finite variance, Theory of Probability and its Applications, 11, 513–540.
103. Kesten H. and Stigum B.P. (1966): A limit theorem for multidimensional Galton-Watson processes, Annals of Mathematical Statistics, 37, 1211–1223.
104. Kingman J.F.C. (1973): Subadditive ergodic theory, Annals of Probability, 1, 883–909.
105. Kingman J.F.C. (1975): The first-birth problem for an age-dependent branching process, Annals of Probability, 3, 790–801.
106. Knuth D.E. (1973): The Art of Computer Programming, Vol. 3: Sorting and Searching, Addison-Wesley, Reading, Mass.
107. Kolchin V.F. (1978): Moment of degeneration of a branching process and height of a random tree, Mathematical Notes of the Academy of Sciences of the USSR, 6, 954–961.
108. Kolmogorov A.N. (1938): Zur Lösung einer biologischen Aufgabe, 2, 1–6, Izvestiya nauchno-issledovatelskogo instituta matematiki i mechaniki pri Tomskom Gosudarstvennom Universitete.
109. Korf R.E. (1992): Search, search and beyond, in: Encyclopedia of Artificial Intelligence (2nd edition), ed. S.C. Shapiro, 1460–1467, Wiley-Interscience.
110. Le Gall J.F. (1990): Brownian excursions, trees and measure-valued branching processes, Technical Report, Université Pierre et Marie Curie, Paris.
111. Louchard G. (1987): Exact and asymptotic distributions in digital and binary search trees, Theoretical Informatics and Applications, 21, 479–495.
112. Lynch W.C. (1965): More combinatorial properties of certain trees, Computer Journal, 7, 299–302.
113. Lyons R. (1997): Probability and Trees, in press.
114. Lyons R., Pemantle R. and Peres Y. (1995): When does a branching process grow like its mean? Conceptual proofs of L log L criteria, Technical Report, Indiana University.
115. Lyons R., Pemantle R. and Peres Y. (1995): Conceptual proofs of L log L criteria for mean behavior of branching processes, Annals of Probability, 23, 1125–1138.
116. Mahmoud H. (1992): Distances in plane-oriented recursive trees, Journal of Computers and Applied Mathematics, 41, 237–245.
117. Mahmoud H.M. (1986): On the average internal path length of m-ary search trees, Acta Informatica, 23, 111–117.
118. Mahmoud H.M. (1992): Evolution of Random Search Trees, John Wiley, New York.
119. Mahmoud H.M. (1994): A strong law for the height of random binary pyramids, Annals of Applied Probability, 4, 923–932.
120. Mahmoud H.M. and Pittel B. (1988): On the joint distribution of the insertion path length and the number of comparisons in search trees, Discrete Applied Mathematics, 20, 243–251.
121. Mahmoud H. and Pittel B. (1984): On the most probable shape of a search tree grown from a random permutation, SIAM Journal on Algebraic and Discrete Methods, 5, 69–81.
122. Mahmoud H., Smythe R.T. and Szymański J. (1993): On the structure of random plane-oriented recursive trees and their branches, Random Structures and Algorithms, 4, 151–176.
123. Marcinkiewicz J. and Zygmund A. (1937): Sur les fonctions indépendantes, Fundamenta Mathematicae, 29, 60–90.

124. McDiarmid C.J.H. (1990): Probabilistic analysis of tree search, in: Disorder in Physical Systems, G.R. Grimmett and D.J.A. Welsh (editors), 249–260, Oxford Science Publications.

125. McDiarmid C.J.H. (1995): Minimal positions in a branching random walk, Annals of Applied Probability, 5, 128–139.

126. McDiarmid C.J.H. and Provan G.M.A. (1991): An expected-cost analysis of backtracking and non-backtracking algorithms, in: IJCAI-91: Proceedings of the Twelfth International Conference on Artificial Intelligence, 172–177, Morgan Kaufmann Publishing, San Mateo, CA.

127. Meir A. and Moon J.W. (1970): The distance between points in random trees, Journal of Combinatorial Theory, 8, 99–103.

128. Meir A. and Moon J.W. (1978): On the altitude of nodes in random trees, Canadian Journal of Mathematics, 30, 997–1015.

129. Moon J.W. (1970): Counting labelled trees, Canadian Mathematical Congress.

130. Moon J.W. (1973): Random walks on random trees, Journal of the Australian Mathematical Society, 15, 42–53.

131. Nagaev S.V. and Pinelis I.F. (1977): Some inequalities for sums of independent random variables, Theory of Probability and its Applications, 22, 248–256.

132. Nerman O. (1981): On the convergence of supercritical general (C-M-J) branching processes, Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete, 57, 365–395.

133. Neveu J. (1986): Arbres et processus de Galton-Watson, Annales de l'Institut Henri Poincaré, 22, 199–207.

134. Neveu J. and Pitman J.W. (1989): The branching process in a Brownian excursion, in: Séminaire de Probabilités XXIII, ed. J. Azéma, P.A. Meyer and M. Yor, 1372, 248–257, Lecture Notes in Mathematics, Springer-Verlag, Berlin.

135. Nievergelt J. and Hinrichs K.H. (1993): Algorithms and Data Structures with Applications to Graphics and Geometry, Prentice-Hall, Englewood Cliffs, NJ.

136. Nievergelt J., Hinterberger H. and Sevcik K.C. (1984): The grid file: an adaptable, symmetric multikey file structure, ACM Transactions on Database Systems, 9, 38–71.

137. Okamoto M. (1958): Some inequalities relating to the partial sum of binomial probabilities, Annals of Mathematical Statistics, 10, 29–35.

138. Pakes A.G. (1971): Some limit theorems for the total progeny of a branching process, Advances in Applied Probability, 3, 176–192.

139. Palmer E.M. (1985): Graphical Evolution, John Wiley, New York.

140. Pearl J. (1984): Heuristics: Intelligent Search Strategies for Computer Problem Solving, Addison-Wesley, Reading, MA.

141. Petrov V.V. (1975): Sums of Independent Random Variables, Springer-Verlag, Berlin.

142. Pittel B. (1984): On growing random binary trees, Journal of Mathematical Analysis and its Applications, 103, 461–480.

143. Pittel B. (1985): Asymptotical growth of a class of random trees, Annals of Probability, 13, 414–427.

144. Pittel B. (1986): Paths in a random digital tree: limiting distributions, Advances in Applied Probability, 18, 139–155.

145. Pittel B. (1994): Note on the heights of random recursive trees and random m-ary search trees, Random Structures and Algorithms, 5, 337–347.

146. Poblete P.V. and Munro J.I. (1985): The analysis of a fringe heuristic for binary search trees, Journal of Algorithms, 6, 336–350.

147. Prusinkiewicz P. and Lindenmayer A. (1990): The Algorithmic Beauty of Plants, Springer-Verlag, New York.

148. Purdom P.W. (1983): Search rearrangement backtracking and polynomial average time, Artificial Intelligence, 21, 117–133.

149. Pyke R. (1965): Spacings, Journal of the Royal Statistical Society Series B, 27, 395–445.

150. Reingold E.M., Nievergelt J. and Deo N. (1977): Combinatorial Algorithms: Theory and Practice, Prentice-Hall, Englewood Cliffs, NJ.

151. Rényi A. (1959): Some remarks on the theory of trees, MTA Mat. Kut. Int. Kozl., 4, 73–85.

152. Rényi A. and Szekeres G. (1967): On the height of trees, Journal of the Australian Mathematical Society, 7, 497–507.

153. Riordan J. (1960): The enumeration of trees by height and diameter, IBM Journal of Research and Development, 4, 473–478.

154. Robson J.M. (1979): The height of binary search trees, The Australian Computer Journal, 11, 151–153.

155. Robson J.M. (1982): The asymptotic behaviour of the height of binary search trees, Australian Computer Science Communications, p. 88.

156. Robson J.M. (1997): Bounds on the variance of binary search tree heights, Technical Report, Université Bordeaux I.

157. Rubinstein R.Y. (1982): Generating random vectors uniformly distributed inside and on the surface of different regions, European Journal of Operations Research, 10, 205–209.

158. Samet H. (1990): Applications of Spatial Data Structures, Addison-Wesley, Reading, MA.

159. Samet H. (1990): The Design and Analysis of Spatial Data Structures, Addison-Wesley, Reading, MA.

160. Sedgewick R. (1983): Mathematical analysis of combinatorial algorithms, in: Probability Theory and Computer Science, ed. G. Louchard and G. Latouche, 123–205, Academic Press, London.

161. Seneta E. (1969): Functional equations and the Galton-Watson process, Advances in Applied Probability, 1, 1–42.

162. Sibuya M. (1979): Generalized hypergeometric, digamma and trigamma distributions, Annals of the Institute of Statistical Mathematics, 31, 373–390.

163. Smith D.R. (1984): Random trees and the analysis of branch and bound procedures, Journal of the ACM, 31, 163–188.

164. Smith R.L. (1984): Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions, Operations Research, 32, 1296–1308.

165. Stepanov V.E. (1969): On the distribution of the number of vertices in strata of a random tree, Theory of Probability and its Applications, 14, 65–78.

166. Stone H.S. and Sipala P. (1986): The average complexity of depth-first search with backtracking and cutoff, IBM Journal of Research and Development, 30, 242–258.

167. Szpankowski W. (1988): Some results on V-ary asymmetric tries, Journal of Algorithms, 9, 224–244.

168. Szymański J. (1988): On the nonuniform random recursive tree, Annals of Discrete Mathematics, 33, 297–306.

169. Timofeev E.A. (1984): Random minimal trees, Theory of Probability and its Applications, 29, 134–141.

170. Timofeev E.A. (1988): On finding the expected length of a random minimal tree, Theory of Probability and its Applications, 33, 361–365.

171. Vitter J.S. (1990): Trees everywhere, in: CAAP 90, ed. A. Arnold, 431, 18–41, Lecture Notes in Computer Science, Springer-Verlag, Berlin.

[172] Vitter J.S. and Flajolet P. (1990): Average-case analysis of algorithms and data structures, in: Handbook of Theoretical Computer Science, Volume A: Algorithms and Complexity, ed. J. van Leeuwen, 431–524, MIT Press, Amsterdam.
[173] Wah B.W. and Yu C.F. (1985): Stochastic modeling of branch-and-bound algorithms with best-first search, IEEE Transactions of Software Engineering, SE-11, 922–934.
[174] Walker A. and Wood D. (1976): Locally balanced binary trees, Computer Journal, 19, 322–325.
[175] Weiner H. (1984): Moments on the maximum in a critical branching process, Journal of Applied Probability, 21, 920–923.
[176] Yaglom A.M. (1947): Certain limit theorems of the theory of branching processes, Dokl. Acad. Nauk SSSR, 56, 795–798.
[177] Zhang W. and Korf R.E. (1992): An average-case analysis of branch-and-bound with applications, in: Proceedings of the 10th National Conference on AI—AAAI-92, 1–6, San Jose, CA.

Author Index

Subject Index