6

Probabilistic Analysis

R.M. Karp
University of California, Berkeley

J.K. Lenstra
Centre for Mathematics and Computer Science, Amsterdam

C.J.H. McDiarmid
Wolfson College, Oxford

A.H.G. Rinnooy Kan
Erasmus University, Rotterdam

CONTENTS

The analysis of combinatorial algorithms is traditionally concerned with *worst case* time and space bounds. Such an analysis has to account for the isolated time consuming problem instance, and hence the results may be pessimistic and give a misleading picture of the *average case.* This point is supported by an abundance of empirical evidence. Thus the ultimate explanation of why algorithms behave as they do must be of a *probabilistic* nature.

A probabilistic analysis requires first of all the specification of a probability distribution over the set of all problem instances. For example, several models for generating random graphs have been extensively investigated, but for other combinatorial structures the choice of a reasonable probability model is less obvious.

A probabilistic analysis of combinatorial problems and algorithms is usually far from trivial. The main reasons for this are the discrete structure of problem instances and solutions, as well as the interdependence between the various steps of an algorithm. What happens at a node of a search tree, for example, depends highly on what has happened at its predecessor.

In recent years, progress has been made on various fronts. One of these is *probabilistic running time analysis.* An example of this approach is the collective effort to explain the success of the simplex method for linear programming. One of the great challenges here is to give rigorous proofs of the polynomial expected running time of various search algorithms, in order to confirm informal analyses or empirical evidence. Secondly, there is the area of *probabilistic error analysis,* where the error refers to the (absolute or relative) difference between an approximate solution value and the optimum. The empirical behavior of heuristics suggests that the worst case error is seldom met in practice, but analytical verification may be quite difficult. Much research of this type is actually based on *probabilistic value analysis,* the third and perhaps most surprising area. Many hard combinatorial optimization problems, especially those with a Euclidean structure, allow a simple probabilistic description of their optimal solution value in terms of the problem parameters.

This bibliography concentrates on these types of probabilistic analyses in combinatorial optimization. It excludes other approaches involving probability models, notably *randomized algorithms* (see Ch.7) and *stochastic optimization* problems in which the realization of the data is not known in advance (see, e.g., Ch.11, §11.2 on stochastic scheduling). We have also excluded topics that are insufficiently related to the area of combinatorial optimization, such as probabilistic models for sorting and for VLSI circuit design.

The organization of this bibliography is as follows. §1 lists a number of *surveys* on the probabilistic analysis of combinatorial algorithms. §2 presents the *basic tools* that are used in the area; the references selected here are intended only as a means of access to the literature on this subject. §3 and §4 review results on *unweighted* and *weighted graphs,* respectively. Most papers in §3 deal with the problems of finding matchings, stable sets, colorings, and Hamiltonian cycles in random graphs. The main subjects in §4 are the problems of finding optimal assignments and shortest traveling salesman tours. §5 is concerned with problems defined in *Euclidean space,* with emphasis on location and routing problems. §6 surveys the recent literature on the average case behavior of variants of the *simplex method* for linear programming. §7 collects results on other *non-graphical, non-Euclidean* problems, such as bin packing, scheduling and knapsack problems. §8 finally discusses *search techniques* for solving hard problems.

This research was partially supported by NSF grants MCS-8105217 and MCS-8311422.

1. Surveys

R.M. Karp (1976). The probabilistic analysis of some combinatorial search algorithms. J.F. Traub (ed.). *Algorithms and Complexity: New Directions and Recent Results,* Academic Press, New York, 1-19.

A general framework for the probabilistic analysis of combinatorial algorithms is introduced. Several algorithms are analyzed probabilistically, including a cellular dissection algorithm for the Euclidean traveling salesman problem, sequential algorithms for constructing cliques and colorings, an extension-rotation algorithm for the Hamiltonian cycle problem and a tree search algorithm for the approximate solution of set covering problems.

G. d'Atri, C. Puech (1978). Analyse probabilistique des problèmes combinatoires. *Mathématiques Appliquées, 1er Coll. AFCET-SMF, Tome II,* Palaiseau, 261-273.

A survey of probabilistic approaches to the analysis of combinatorial problems. The principal examples concern random graphs and random shortest path problems.

G. d'Atri (1980). Outline of a probabilistic framework for combinatorial optimization. F. Archetti, M. Cugiani (eds.). *Numerical Techniques for Stochastic Systems,* North-Holland, Amsterdam, 347-368.

The basic concepts of the probabilistic analysis of algorithms and of the concept of a randomized algorithm are explained through examples: a search problem, the knapsack problem, and primality testing.

L. Slominski (1982). Probabilistic analysis of combinatorial algorithms: a bibliography with selected annotations. *Computing 28,* 257-267.

An excellent survey with especially good coverage of early Soviet work not available in English. This survey is a useful complement to the present one, which covers the early Soviet literature rather sparsely.

D.S. Johnson (1984). The NP-completeness column: an ongoing guide; eleventh edition. *J. Algorithms 5,* 284-299.

This column first surveys results that show $\mathcal{NP}$-complete problems to be solvable in polynomial time on average. It then considers the concept of 'hardness on average' and discusses an important result of L.A. Levin (see §7.1).

R.M. Karp (to appear). The probabilistic analysis of combinatorial optimization algorithms. *Proc. Internat. Congress Math.,* Warsaw.

Several problems are discussed from the viewpoint of probabilistic analysis. These include multiprocessor scheduling, matchings and Hamiltonian cycles, cliques and colorings, the assignment problem and the traveling salesman problem.

2. Basic Tools

2.1. *Moment methods*

P. Erdös, J. Spencer (1974). *Probabilistic Methods in Combinatorics,* Academic Press, New York.
B. Bollobás (1979). *Graph Theory,* Graduate Texts in Mathematics 63, Springer, Berlin.

Existence results concerning random graphs or networks are often proved by using a *moment method.* Let the random variable X take values 0,1,2,..., and have mean $E[X] = \mu > 0$ and variance $E[(X-\mu)^2] = \sigma^2$. (Perhaps X counts the number of subgraphs of a random graph with a certain property.) Then we have

(a) the *Markov Inequality*: $Pr\{X>0\} \leqslant \mu$;
(b) from the *Chebyshev Inequality*: $Pr\{X = 0\} \leqslant \sigma^2/\mu^2$;
(c) from the *Cauchy-Schwarz Inequality*: $Pr\{X>0\} \geqslant \mu^2/E[X^2]$.

When we use (a) [(b) or (c)], then we say that we are using a first [second] moment method.

2.2. *Binomial properties*

W. Feller (1968). *An Introduction to Probability Theory and Its Applications, Volume 1, Third Edition,* Wiley, New York.
D.S. Mitrinovic (1970). *Analytic Inequalities,* Springer, Berlin.

Let $0 < p = 1-q < 1$ and let n be a positive integer. A key result in combinatorial probability is *Stirling's Formula* (as refined by H. Robbins):

$$n! = n^n e^{-n}\sqrt{2\pi n}\,e^{\alpha(n)}, \text{ where } \frac{1}{12n+1} < \alpha(n) < \frac{1}{12n}.$$

The following simple bounds for binomial coefficients are often useful. If $1 \leqslant k \leqslant n$ then

$$\left(\frac{n}{k}\right)^k \leqslant \binom{n}{k} < \left(\frac{ne}{k}\right)^k.$$

There exists a constant $c>0$ (independent of p) such that, if np is an integer, then

$$\binom{n}{np} p^{np} q^{nq} \geqslant cn^{-\frac{1}{2}}.$$

2.3. *Bounds on tails of distributions*

H. Chernoff (1952). A measure of asymptotic efficiency for tests of a hypothesis based on a sum of observations. *Ann. Math. Statist. 23,* 493-507.

Chernoff proves the following bounds on the tails of the distribution of the sum of independent observations: Let $X_1,...,X_n$ be independent identically distributed random variables with finite expectation μ and let $m(\alpha) = \inf_t \{e^{-\alpha t} E[e^{tX_1}]\}$. If $\alpha \leq [\geq] \mu$, then $Pr\{\Sigma_{k=1}^n X_k \leq [\geq] n\alpha\} \leq (m(\alpha))^n$. For the tails of a binomial distribution this implies that, if $0 \leq \beta \leq 1$, then

$$\sum_{k \leq (1-\beta)np} \binom{n}{k} p^k q^{n-k} \leq e^{-\beta^2 np/2},$$

$$\sum_{k \geq (1+\beta)np} \binom{n}{k} p^k q^{n-k} \leq e^{-\beta^2 np/3}.$$

These inequalities as well as the last inequality in §2.2 were used, for example, in [Angluin & Valiant 1979] (see §3.3, §3.7).

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

V. Chvátal (1979). The tail of the hypergeometric distribution. *Discrete Math. 25,* 285-287.

Related results and extensions.

A.W. Marshall, I. Olkin (1979). *Inequalities: Theory of Majorisation and Its Applications,* Academic Press, New York.

Chapter 17C gives extensions of the following intuitively obvious result that has on occasion been useful. Let $X_1, X_2,...$ be a sequence of 0,1-valued random variables such that for $t = 1,2,...$, given any history concerning $X_1,...,X_t$, the probability that $X_{t+1} = 1$ is at most p. Then $X_1 + ... + X_n$ is stochastically less than a binomial random variable with parameters n and p.

P.J. Boland, F. Proschan (1983). The reliability of k out of n systems. *Ann. Probab. 11,* 760-764.

A related result of Hoeffding set in the context of majorization.

2.4. *Conditioning*

M.L. Eaton (1982). A review of selected topics in multivariate probability inequalities. *Ann. Statist. 10,* 11-43.

R.L. Graham (1983). Applications of the FKG inequality and its relatives. A. Bachem, M. Grötschel, B. Korte (eds.). *Mathematical Programming: the State of the Art - Bonn 1982,* Springer, Berlin, 115-131.

Two recent reviews of the most useful results on 'benevolent' conditioning: *Harris' Lemma* and its extension, the *FKG Inequality.*

Y.L. Tong (1980). *Probability Inequalities in Multivariate Distributions,* Academic Press, New York.

A useful concept of 'positive dependence' is that of *association* - see for example [Eaton 1982] above and this reference.

K. Joag-Dev, F. Proschan (1983). Negative association of random variables, with applications. *Ann. Statist. 11,* 286-295.

A recent paper on one of the various concepts of 'negative dependence' that have been proposed.

C. McDiarmid (1981). General percolation and random graphs. *Adv. in Appl. Probab. 13,* 40-60.
C. McDiarmid (1983). General first-passage percolation. *Adv. in Appl. Probab. 15,* 149-161.

Certain 'general percolation' results are useful for handling random directed graphs and networks. For example, the random directed graph $D_{n,p}$ is more likely than the random graph $G_{n,p}$ to have a Hamiltonian cycle.

C. McDiarmid (to appear). On some conditioning results in the probabilistic analysis of algorithms. *Discrete Appl. Math.*

A simple combinatorial approach for handling conditioning problems that arise in the probabilistic analysis of graph algorithms. Arguments from [Angluin & Valiant 1979] (see §3.3, §3.7) and [Karp & Tarjan 1980] (see § 3.2) are substantially simplified.

2.5. *Stochastic convergence*

R.J. Serfling (1980). *Approximation Theorems of Mathematical Statistics,* Wiley, New York.

A sequence of random variables $X_1,X_2,\ldots$ is said to converge to a random variable X

(a) *in probability* if $\lim_{n\to\infty} Pr\{|X_n - X|>\epsilon\} = 0$ for every $\epsilon>0$;
(b) *with probability* 1 or *almost surely* ('a.s.') if $Pr\{\lim_{n\to\infty} X_n = X\} = 1$;
(c) *completely* if $\Sigma_{n=1}^{\infty} Pr\{|X_n - X|>\epsilon\} < \infty$ for every $\epsilon>0$.

According to the *Borel-Cantelli Lemma,* (c) implies (b). Also, (b) implies (a), but the inverse implications do not hold. For comments on these concepts that are relevant in the present context, see [Karp & Steele 1985, §2.4] in §4.5.

3. Unweighted Graphs

3.1. *Random graphs*

In this section we follow the loose but common practice of saying that an event concerning random graphs happens *almost surely* or for *almost all* graphs if the probability that it happens tends to 1 as $n\to\infty$.

B. Bollobás (to appear). *Lectures on Random Graphs.*

Let $G_{n,p}$ $[D_{n,p}]$ denote the random graph [directed graph] with vertex set $\{1, \ldots, n\}$ in which the $n(n-1)/2$ $[n(n-1)]$ possible edges occur independently with probability p. The random graph $G_{n,N}$ [directed graph $D_{n,N}$] has the same vertex set but now the $\binom{n(n-1)/2}{N}$ possible graphs $[\binom{n(n-1)}{N}$ possible directed graphs] occur with the same probability.

[Erdös & Spencer 1974] and [Bollobás 1979] (see §2.1) give an introduction to the theory of random graphs. The current reference gives a full treatment. For relations between the models $G_{n,p}$ and $G_{n,N}$ see, for example, [Angluin & Valiant 1979] (§3.3, §3.7).

B. Bollobás (1981). Random graphs. H.N.V. Temperley (ed.). *Combinatorics,* London Mathematical Society Lecture Notes 52, 80-102.
M. Karónski (1982). A review of random graphs. *J. Graph Theory 6,* 349-389.
K. Weber (1982). Random graphs - a survey. *Rostock. Math. Kolloq. 21,* 83-98.
G. Grimmett (1983). Random graphs. L. Beineke, R. Wilson (eds.). *Selected Topics in Graph Theory 2,* Academic Press, London, 201-235.

Four recent surveys on random graphs. For a discussion of random directed graphs see [McDiarmid 1981] (§2.4).

3.2. *Connectivity*

We consider here some problems related to the connectivity of a graph or directed graph for which there exist algorithms that are quite fast in the worst case.

P.A. Bloniarz, M.J. Fischer, A.R. Meyer (1976). A note on the average time to compute transitive closures. S. Michaelson, R. Milner (eds.). *Automata, Languages and Programming,* Edinburgh University Press, Edinburgh, 425-434.

An algorithm for the transitive closure of a directed graph has average time $O(n^2 \log n)$. The analysis is for random directed graphs with probabilities depending only on the number of vertices and set of outdegrees (more general than $D_{n,N}$).

C.P. Schnorr (1978). An algorithm for transitive closure with linear expected time. *SIAM J. Comput. 7,* 127-133.

An algorithm for transitive closure is given with average time $O(n+m^*)$ where n is the number of vertices and m^* is the expected number of edges in the transitive closure. The analysis is for random directed graphs $D_{n,N}$.

R.M. Karp, R.E. Tarjan (1980). Linear expected time algorithms for connectivity problems. *J. Algorithms 1,* 374-393.

Algorithms that run in linear expected time are given for finding connected components, strong components and biconnected components. The analysis is for random graphs $G_{n,N}$ and random directed graphs $D_{n,N}$ (uniformly over N).

3.3. *Matching*

A *matching* in a graph is a set of edges with no endpoints in common. In a graph with n vertices, a *maximum* matching (a matching of maximum cardinality) can be found in time $O(n^{2.5})$; it is *perfect* if it contains $n/2$ edges. A *cover* of vertices by edges is a set of edges such that each vertex is an endpoint of some edge in the set. A minimum cover consists of a maximum matching together with any minimal set of edges covering the remaining vertices. For some early papers on matchings and coverings see [Slominski 1982] (§1).

D. Angluin, L.G. Valiant (1979). Fast probabilistic algorithms for Hamiltonian circuits and matchings. *J. Comput. System Sci. 19,* 155-193.

An $O(n \log n)$ time heuristic a.s. finds a perfect matching in $G_{n,N}$ when n is even and $N \geqslant cn \log n$, for a suitable constant c. Earlier related work is discussed.

W.F. de la Vega (1980). Sur la cardinalité maximum des couplages d'hypergraphes aléatoires uniformes. *Discrete Math. 40,* 315-318.

If $N/n \rightarrow \infty$ as $n \rightarrow \infty$, then a greedy heuristic yields a matching M_n in $G_{n,N}$ with $|M_n|/n \rightarrow ½$ in probability; whilst with fewer edges the proportion of isolated vertices does not tend to 0 in probability.

R.M. Karp, M. Sipser (1981). Maximum matchings in sparse random graphs. *Proc. 22nd Annual IEEE Symp. Foundations of Computer Science,* 364-375.

A linear time heuristic based on trimming away low-degree vertices is shown to give near maximum matchings in $G_{n,p}$ when $p = \lambda/(n-1)$.

E. Shamir, E. Upfal (1981). On factors in random graphs. *Israel J. Math. 39,* 296-302.

An existence result concerning f-factors in $G_{n,p}$ is proved by showing that subfactors can almost surely be augmented by using alternating paths.

E. Shamir, E. Upfal (1982). N-Processors graphs distributively achieve perfect matchings in $O(\log^2 N)$ beats. *Proc. Annual ACM Symp. Principles of Distributed Computing,* 238-241.

A parallel algorithm (one processor at each vertex, no shared memory) operates for $O(\log^2 n)$ beats and a.s. finds a perfect matching in $G_{n,p}$ when $np > c \log n$ (for a suitable constant c).

G. Tinhofer (1984). A probabilistic analysis of the greedy heuristic for the matching problem. *Ann. Oper. Res. 1.*

Theoretical and simulation results are given concerning variants of a greedy heuristic for maximum matchings.

A.M. Frieze (1984A). *On Large Matchings and Cycles in Sparse Random*

Graphs, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA.

The random graph $G_{n,c/n}$ a.s. contains a matching of cardinality $n(1-(1+\epsilon(c))e^{-c})/2$, where $\epsilon(c)\to 0$ as $c\to\infty$.

A.M. Frieze (1984B). *Maximum Matchings in a Class of Random Graphs,* Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA.

Let G_n^m denote the random graph with n vertices in which each vertex independently chooses m edges incident with it. G_n^1 a.s. does not contain a perfect matching, and G_n^m ($m \geqslant 2$, n even) a.s. does.

3.4. *Graph isomorphism*

It is not known if the problem of testing whether two graphs are isomorphic is $\mathfrak{N}\mathfrak{P}$-complete, although if the graphs have bounded degree the problem is known to be in $\mathfrak{P}$.

R.J. Lipton (1978). *The Beacon Set Approach to Graph Isomorphism,* Yale University.
L. Babai, L. Kučera (1979). Canonical labelling of graphs in linear average time. *Proc. 20th Annual IEEE Symp. Foundations of Computer Science,* 39-46.
R.M. Karp (1979). Probabilistic analysis of a canonical numbering algorithm for graphs. *Proc. Symp. Pure Mathematics 34,* AMS, Providence, RI, 365-378.
L. Babai, P. Erdös, S.M. Selkow (1980). Random graph isomorphism. *SIAM J. Comput. 9,* 628-635.

Each of these papers gives a fast *canonical labeling* algorithm that works for almost all graphs. Thus a.s. for $G_{n,1/2}$ any graph can be tested for isomorphism to this graph by a naive fast algorithm. The paper listed last above discusses the papers listed earlier. See also [Johnson 1984] (§1).

3.5. *Stable sets and coloring*

A set of vertices in a graph G is *stable* if no two are adjacent. The *stability number* $\alpha(G)$ is the maximum size of a stable set in G. A *coloring* of G is an assignment of colors to the vertices so that no two adjacent vertices receive the same color. The *chromatic number* $\chi(G)$ is the least number of colors in a coloring of G.

No polynomial time algorithms are known to approximate either $\alpha(G)$ or $\chi(G)$ to within any constant factor. If $\mathfrak{P}\neq\mathfrak{N}\mathfrak{P}$ then no such algorithm colors within a ratio less than 2.

In [Erdös & Spencer 1974] (see §2.1) bounds are stated for $\chi(G_{n,p})$ both in the *constant density* case (Ch. 11, Exercise 2) and in the *constant average degree* case (Ch. 16, Exercise 9).

C. McDiarmid (1984). Colouring random graphs. *Ann. Oper. Res.* 1.

The greedy stable set algorithm considers the vertices in a given order and adds them to the current stable set if possible. The greedy or simple sequential coloring algorithm does this repeatedly to form the different color sets. McDiarmid's survey (with 57 references) focuses mainly on approaches of this type. It includes a treatment of the constant average degree case.

G. Grimmett, C. McDiarmid (1975). On colouring random graphs. *Math. Proc. Cambridge Philos. Soc. 77,* 313-324.

In the constant density random graph $G_{n,p}$ simple greedy approaches a.s. yield a stable set of size at least $(\frac{1}{2}-\epsilon)\alpha(G_{n,p})$ and a coloring using at most $(2+\epsilon)\chi(G_{n,p})$ colors.

B. Bollobás, P. Erdös (1976). Cliques in random graphs. *Math. Proc. Cambridge Philos. Soc. 80,* 419-427.

D.W. Matula (1976). *The Largest Clique Size in a Random Graph,* Technical report CS7608, Department of Computer Science, Southern Methodist University, Dallas, TX.

Results from [Grimmett & McDiarmid 1975] (see above) are sharpened.

V. Chvátal (1977). Determining the stability number of a graph. *SIAM J. Comput. 6,* 643-662.

For almost all random graphs with (large) constant average degree all recursive proofs bounding the stability number are of exponential length, and hence any 'Tarjan-type' algorithm must be slow.

L. Kučera (1977). Expected behaviour of graph coloring algorithms. M. Karpinski (ed.) *Fundamentals of Computation Theory,* Lecture Notes in Computer Science 56, Springer, Berlin, 447-451.

The expected behavior of the simple greedy coloring algorithm and of a variant (involving considering vertex degrees) are discussed when they act on random graphs $G_{n,p}$ and on random k-partite graphs.

C. McDiarmid (1979). Determining the chromatic number of a graph. *SIAM J. Comput. 8,* 1-14.

For constant density random graphs $G_{n,p}$ a.s. all algorithms in a certain class of branch-and-bound algorithms for determining the chromatic number will take more than exponential time.

C. McDiarmid (1979). Colouring random graphs badly. R.J. Wilson (ed.). *Graph Theory and Combinatorics,* Pitman Research Notes in Mathematics 34, Pitman, London, 76-86.

For constant density random graphs $G_{n,p}$ the greedy or simple sequential coloring algorithm a.s. performs essentially as badly as possible. Results on the achromatic number are tightened up later in [McDiarmid 1982] (see below).

A.D. Korsunov (1980). The chromatic number of n-vertex graphs. *Metody Diskret. Analiz. 35,* 14-44, 104 (in Russian).

The conjecture that $\chi(G_{n,½})\,(\log_2 n)/n \rightarrow ½$ in probability as $n \rightarrow \infty$ is given as a theorem. The proof uses the second moment method.

C. McDiarmid (1982). Achromatic numbers of random graphs. *Math. Proc. Cambridge Philos. Soc. 92,* 21-28.

The *achromatic number* $\psi(G)$ of a graph G is the largest number of colors in a coloring of G such that no two colors may be identified. For constant density random graphs $G_{n,p}$, it is shown that a.s. $n/(k+1) \leqslant \psi(G_{n,p}) \leqslant n/(k-1)$, where $k = \log n / \log (1/(1-p))$. The lower bound is obtained by an analysis of a silly variant of the greedy coloring algorithm.

B. Pittel (1982). On the probable behaviour of some algorithms for finding the stability number of a graph. *Math. Proc. Cambridge Philos. Soc. 92,* 511-526.

This paper uses martingale arguments to analyze the greedy stable set algorithm, and investigates Chvátal's 'f-driven' algorithms for determining the stability number (without subtleties of the monotone rule).

A. Johri, D.W. Matula (1982). *Probabilistic Bounds and Heuristic Algorithms for Coloring Large Random Graphs,* Technical report 82-CSE-6, Southern Methodist University, Dallas, TX.

Nonasymptotic theoretical work together with simulation results indicate that with high probability the random graph $G_{1000,½}$ has chromatic number in the range 85 ± 12.

T. Kawaguchi, H. Nakano, Y. Nakanishi (1982). *Probabilistic Analysis of a Heuristic Graph Colouring Algorithm,* Unpublished manuscript.

Asymptotic results are given for the greedy coloring algorithm acting on $G_{n,p}$ when $np = cn^{\delta}$ for some constants c and $½ < \delta \leqslant 1$. Stronger results appear in [McDiarmid 1983], [De la Vega 1982A] and [Shamir & Upfal 1984] (see below). Some nonasymptotic results are also given.

W.F. de la Vega (1982A). *On the Chromatic Number of Sparse Random Graphs,* Laboratoire de Recherche en Informatique, Université de Paris-Sud.

E. Shamir, E. Upfal (1984). Sequential and distributed graph coloring algorithms with performance analyses in random graph spaces. *J. Algorithms.*

Both these papers consider random graphs $G_{n,p}$ with decreasing density and increasing average degree, more specifically $p \rightarrow 0$ and $np \rightarrow \infty$ as $n \rightarrow \infty$. They show that a variant of the greedy coloring algorithm (involving a different end phase) is a.s. optimal to within a factor $2+\epsilon$, and thus settle a conjecture of Erdös and Spencer.

W.F. de la Vega (1982B). *Crowded Graphs Can Be Colored Within a Factor $1+\epsilon$ in Polynomial Time,* Laboratoire de Recherche en Informatique,

Université de Paris-Sud.

Consider very dense random graphs for which $\alpha(G_{n,p})$ is a.s. a constant $r+1$. A greedy heuristic for picking disjoint stable sets of size r is a.s. optimal to within a factor $1+\epsilon$.

C. McDiarmid (1983). On the chromatic forcing number of a random graph. *Discrete Appl. Math. 5,* 123-132.

If we wish to compute lower bounds for the chromatic number $\chi(G)$ of a graph G we may be interested in the *chromatic forcing number* $f_\chi(G)$ which is defined to be the least number of vertices in a subgraph H of G with $\chi(H) = \chi(G)$. For random graphs $G_{n,p}$ with say $(\log n)^{-1} < p(n) < 1-(\log n)^{-1}$ we have $f_\chi \geqslant (\frac{1}{2}-\epsilon)n$ a.s.

D.W. Matula (1983). *Improved Bounds on the Chromatic Number of a Graph,* Abstract, Department of Computer Science, Southern Methodist University, Dallas, TX.

A certain nonpolynomial time coloring algorithm is a.s. optimal to within a factor $(3/2+\epsilon)$, thus beating the greedy coloring algorithm.

H.S. Wilf (1984). Backtrack: an $O(1)$ expected time algorithm for the graph coloring problem. *Inform. Process. Lett. 18,* 119-121.
E.A. Bender, H.S. Wilf (1984). A theoretical analysis of backtracking in the graph coloring problem. *J. Algorithms.*

A simple backtracking procedure will test if a graph can be colored with k colors. For fixed k, the average time taken for certain random graphs $G_{n,p}$ is shown to be small. See also [Johnson 1984] (§1).

We conclude this subsection with results for miscellaneous problems related to finding stable sets and colorings in graphs.

G. Cornuéjols, G.L. Nemhauser, L.A. Wolsey (1978). *Worst-Case and Probabilistic Analysis of Algorithms for a Location Problem,* Technical report 375, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY.

The problem considered here is to choose a set S of k vertices in a given n-vertex graph to maximize the number of edges incident with S. For almost all graphs with n vertices and all positive integers $k \leqslant n^\alpha$ where $\alpha < 1/6$, the k vertices of largest degree generate an optimal solution.

D. Hochbaum (1982). *Easy Solutions for the k-Center Problem or the Dominating Set Problem on Random Graphs,* School of Business Administration, University of California, Berkeley.

A set of vertices in a graph is *dominating* if each vertex not in the set is adjacent to some vertex in the set. Thus maximal stable sets are dominating. This paper considers the average behavior of a problem related to dominating sets.

J. Schmidt-Pruzan, E. Shamir, E. Upfal (1984). Random hypergraph coloring algorithms and the weak chromatic number. *J. Combin. Theory Ser. B.*
J. Schmidt-Pruzan (1983). *Probabilistic Analysis of Strong Hypergraph Coloring Algorithms and the Strong Chromatic Number,* Department of Applied Mathematics, Weizmann Institute of Science, Rehovot.

Algorithms are proposed for various hypergraph coloring problems. For certain random hypergraphs they are a.s. optimal to within a small constant factor.

R.M. MacGregor (1978). *On Partitioning a Graph: a Theoretical and Empirical Study,* Memorandum UCB/ERL M78/14, Electronics Research Laboratory, University of California, Berkeley.
T. Bui, S. Chaudhuri, T. Leighton, M. Sipser (1984). Graph bisection algorithms with good average case behavior. *Proc. 25th Annual IEEE Symp. Foundations of Computer Science.*

The *graph k-partition* problem involves the determination of a mimimum set of edges whose removal disconnects the graph into k equal-sized subgraphs. MacGregor investigates the performance of iterative improvement schemes and also provides probabilistic lower and upper bounds on the size of a minimum 2-partition. Bui *et al.* propose a polynomial-time algorithm and show that it finds a minimum 2-partition with high probability. They use a special sample space with the property that, with high probability, very few edges need to be deleted to partition the graph.

J.H. Reif, P.G. Spirakis (1980). Random matroids. *Proc. 12th Annual ACM Symp. Theory of Computing,* 385-397.

An analysis is made of the probability that a greedy algorithm will find a maximum independent set in a random independence system.

3.6. *Long paths*

It is easily seen that, given a constant $c > 0$, the problem of determining if a graph with n vertices has a simple path of length at least cn is $\mathcal{NP}$-complete.

W.F. de la Vega (1979). Long paths in random graphs. *Studia Sci. Math. Hungar. 14,* 335-340.

A simple heuristic a.s. yields a path of length at least $(1-1.39/c)n$ in $G_{n,cn}$, and a similar results holds for random directed graphs.

M. Ajtai, J. Komlós, E. Szemerédi (1981). The longest path in a random graph. *Combinatorica 1,* 1-12.
B. Bollobás (1982). Long paths in sparse random graphs. *Combinatorica 2,* 223-228.

A.M. Frieze (1984B). *On Large Matchings and Cycles in Sparse Random Graphs,* Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA.

These papers concern the existence of long paths rather than the analysis of algorithms. Frieze proves that $G_{n,c/n}$ a.s. contains a cycle of length $n(1-(1+\epsilon(c))ce^{-c})$, where $\epsilon(c)\rightarrow 0$ as $c\rightarrow\infty$.

3.7. *Hamiltonian cycles*

A Hamiltonian cycle in a graph or a directed graph is a closed path that passes through each vertex exactly once. It is a classical result that the problem of determining if a graph has a Hamiltonian cycle is $\mathcal{NP}$-complete.

V. Chvátal (1985). Hamiltonian cycles. E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys (eds.). *The Traveling Salesman Problem,* Wiley, Chichester, Ch.11.

Section 3 of this chapter is an excellent survey of the research on the existence of Hamiltonian cycles in random graphs. It includes Karp's version and analysis of the *extension-rotation* algorithm. Other surveys are included in [Angluin & Valiant 1979] below and [Slominski 1982] (§1). See also [Johnson 1984] (§1).

L. Pósa (1976). Hamiltonian circuits in random graphs. *Discrete Math. 14,* 359-364.

There is a.s. a Hamiltonian cycle in the random graph $G_{n,N}$ with a number of edges $N \geqslant cn \log n$ for a suitable constant c. The proof is not based on an algorithm, but the extension-rotation idea can lead to good algorithms.

A.D. Koršunov (1976). Solution of a problem of Erdös and Rényi on Hamiltonian cycles in non-oriented graphs. *Soviet Math. Dokl. 17,* 760-764.

Consider the random graph $G_{n,N}$ with $N \geqslant n(\log n + \log\log n + \omega(n))/2$, where $\omega(n)\rightarrow\infty$ as $n\rightarrow\infty$. It is indicated that an algorithm a.s. finds a Hamiltonian cycle.

D. Angluin, L.G. Valiant (1979). Fast probabilistic algorithms for Hamiltonian circuits and matchings. *J. Comput. System Sci. 19,* 155-193.

Fast heuristics ($O(n \log^2 n)$ time) are shown to find Hamiltonian cycles a.s. in random graphs $G_{n,N}$ or similar random directed graphs when $N \geqslant cn \log n$ for a suitable constant c.

E. Shamir (1983). How many random edges make a graph Hamiltonian? *Combinatorica 3,* 123-131.

For the random graph $G_{n,p}$ with $p = p(n) = n^{-1}(\log n + c \log\log n)$, $c > 3$, there is an extension-rotation procedure which a.s. finds a Hamiltonian path within $O(n^2)$ steps.

The following five papers are concerned more with existence than with algorithms. They all use the extension-rotation idea.

J. Komlós, E. Szemerédi (1983). Limit distribution for the existence of hamiltonian cycles in random graphs. *Discrete Math. 43,* 55-63.

The random graph $G_{n,N}$ with N about $\frac{1}{2}n \log n + \frac{1}{2}n \log\log n + cn$ is Hamiltonian with probability tending to $\exp(\exp(-2c))$ as $n \to \infty$.

T.I. Fenner, A.M. Frieze (1983). On the existence of Hamiltonian cycles in a class of random graphs. *Discrete Math. 45,* 301-305.

The random graph G_n^m (see [Frieze 1984B], §3.3) is a.s. Hamiltonian for $m \geqslant 23$.

T.I. Fenner, A.M. Frieze (1982). *Hamiltonian Cycles in Random Regular Graphs,* Queen Mary College, University of London.

B. Bollobás (1983). Almost all regular graphs are Hamiltonian. *European J. Combin. 4,* 97-106.

In both these papers it is shown that for fixed k sufficiently large, the proportion of k-regular graphs on n vertices which are Hamiltonian tends to 1 as $n \to \infty$.

A.M. Frieze (1982). *Limit Distribution for the Existence of Hamiltonian Cycles in Random Bipartite Graphs,* Department of Computer Science and Statistics, Queen Mary College, University of London.

Let P_n be the probability that there is a Hamiltonian cycle in the random bipartite graph with $2n$ vertices in which the n^2 possible edges occur independently with probability $n^{-1}(\log n + \log\log n + c_n)$. Then $P_n \to 0$ if $c_n \to -\infty$, $P_n \to 1$ if $c_n \to \infty$ and $P_n \to \exp(-2e^{-c})$ if $c_n \to c$.

R.W. Robinson, N.C. Wormald (to appear). Almost all bipartite cubic graphs are Hamiltonian. *Proc. Silver Jubilee Conf. Combinatorics, Waterloo,* Academic Press, New York.

The proportion of bipartite labeled cubic graphs which are Hamiltonian tends to 1 as $n \to \infty$. The proof uses the second moment method.

3.8. *Bandwidth*

The *bandwidth* of a graph is the minimum over all labelings of the vertices with distinct integers of the maximum difference of the labels of adjacent vertices. The problem of determining the bandwidth is $\mathcal{NP}$-hard.

P.Z. Chinn, J. Chvátalovà, A.K. Dewdney, N.E. Gibbs (1982). The bandwidth problem for graphs and matrices - a survey. *J. Graph Theory 6,* 223-254.

There is a brief mention of the average-case complexity of bandwidth algorithms.

J. Turner (1983). Probabilistic analysis of bandwidth minimization algorithms. *Proc. 15th Annual ACM Symp. Theory of Computing,* 467-476.

This paper provides a probabilistic explanation of the effectiveness of *level algorithms* for bandwidth minimization on certain classes of graphs.

4. Weighted Graphs

4.1. *Shortest paths*

The shortest paths problem is the problem of finding minimum weight paths between specified source vertices and destination vertices in a graph or digraph with weighted edges. In probabilistic analyses the edge weights are often taken to be independent identically distributed random variables.

P.M. Spira (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 J. Comput. 2,* 28-32.

A new algorithm is presented for the all pairs shortest path problem in a digraph with nonnegative edge weights. If the weights are drawn independently from a nonnegative continuous distribution, then the expected execution time is as stated in the title. This is one of the earliest papers to conduct a sound probabilistic analysis of an interesting combinatorial algorithm.

Y. Perl (1977). *Average Analysis of Simple Path Algorithms,* Technical report UIUCDCS-R-77-905, Department of Computer Science, University of Illinois at Urbana-Champaign.

In a random graph with n vertices and m edges, the expected number of edges inspected in searching for a path from a source vertex to a destination vertex using breadth-first or depth-first search is $O(n)$. In a random graph with edge weights drawn independently from a common nonnegative distribution, Prim's minimum spanning tree algorithm and Dijkstra's shortest path algorithm both run in expected time $O(n \log n \log(m/n))$ and Kruskal's minimum spanning tree algorithm runs in expected time $O(n \log n \log m)$.

P.A. Bloniarz, R.M. Meyer, M.J. Fischer (1979). *Some Observations on Spira's Shortest Path Algorithm,* Technical report 79-6, Computer Science Department, State University of New York, Albany.

Considers Spira's algorithm (see above) for the case that edges may have equal weights.

P. Bloniarz (1983). A shortest-path algorithm with expected time $O(n^2 \log n \log^* n)$. *SIAM J. Comput. 12,* 588-600.

Spira's algorithm is refined and the expected execution time is improved.

A.M. Frieze, G.R. Grimmett (1983). *The Shortest-Path Problem for Graphs with Random Arc Lengths,* School of Mathematics, University of Bristol.

For 'endpoint-independent' distributions, a modification of Spira's algorithm runs in $O(n(m+n \log n))$ expected time, where m is the expected number of edges with finite weight. When edge weights are independent with certain distributions, a further modification runs in $O(n^2 \log n)$ expected time.

M. Luby, P. Ragde (1983). *Bidirectional Search is $O(\sqrt{n})$ Faster Than Dijkstra's Shortest Path Algorithm*, Computer Science Division, University of California, Berkeley.

On a complete n-vertex digraph with independent exponentially distributed edge weights a variant of bidirectional search finds the shortest path from a given source to a given destination in expected time $O(n^{1/2} \log n)$. The algorithm also has a preprocessing phase requiring $O(n^2)$ expected time.

4.2. *Spanning trees*

We here consider the problem of finding a spanning tree of minimum weight in a graph with weighted edges. In probabilistic analyses the edge weights are usually independent identically distributed random variables.

Y. Perl (1977). *Average Analysis of Simple Path Algorithms*, Technical report UIUCDCS-R-77-905, Department of Computer Science, University of Illinois at Urbana-Champaign.

See §4.1.

R.M. Karp, R.E. Tarjan (1980). Linear expected time algorithms for connectivity problems. *J. Algorithms 1*, 374-393.

See §3.2 for a review of the first part of this paper. For random graphs with m edges and edge weights drawn independently from a common distribution, an algorithm is given which finds a minimum spanning forest in expected time $O(m)$.

A.M. Frieze (1982). *On the Value of a Random Minimum Spanning Tree Problem*, Technical report, Department of Computer Science and Statistics, Queen Mary College, University of London.

In a complete graph in which the weights of the edges are drawn independently from the uniform distribution on [0,1], the expected cost of the minimum spanning tree is asymptotic to $\Sigma_{k=1}^{\infty} 1/k^3 = 1.202...$

4.3. *Linear assignment*

An instance of the assignment problem is specified by an $n \times n$ real matrix (d_{ij}). An assignment is a permutation σ of $\{1,2,\ldots,n\}$. The cost of assignment σ is $\Sigma_{i=1}^{n} d_{i\sigma(i)}$. An optimal assignment is one of minimum cost.

The assignment problem can be viewed as the problem of finding a minimum weight perfect matching in a bipartite graph with n vertices in each part. The assignment problem is often used as a relaxation of the traveling salesman problem, in which σ is required to be a cyclic permutation. An optimal assignment can be found in $O(n^3)$ time using network flow techniques.

A.A. Borovkov (1962). Toward a probabilistic formulation of two problems from economy. *Math. Dokl. Akad. Nauk SSSR 146,* 983-986.

The d_{ij} are assumed to be drawn independently from a common distribution satisfying certain technical conditions. A greedy algorithm of complexity $O(n^2)$ has the property that, for every $\epsilon>0$, $Pr\{\text{GREEDY}\geqslant(1+\epsilon)\text{OPT}\}\rightarrow 0$ as $n\rightarrow\infty$. Here GREEDY is the cost of the solution produced by the greedy algorithm and OPT is the cost of an optimal assignment. This is one of the earliest publications concerned with the probabilistic analysis of approximation algorithms.

W.E. Donath (1969). Algorithm and average-value bounds for assignment problems. *IBM J. Res. Develop. 13,* 380-386.

An argument is offered suggesting that, when the d_{ij} are drawn independently from the uniform distribution over [0,1], the expected cost of an optimal assignment is less than 2.37. Certain conditioning effects are neglected. Computational results are presented indicating that the expected cost of an optimal assignment is close to 1.6 when n is large. Other probability distributions of the d_{ij} are also considered.

A.J. Lazarus (1979). *The Assignment Problem with Uniform* (0,1) *Cost Matrix,* B.A. thesis, Department of Mathematics, Princeton University, Princeton, NJ.

Let the d_{ij} be independent and uniformly distributed over [0,1]. Let Y_n be the expected cost of an optimal assignment and $Y = \lim_{n\rightarrow\infty} Y_n$. It is shown that $Y_2 = 23/30$ and $Y \geqslant 1+(1/e)$. Empirical evidence that $Y<\infty$ is given.

D.W. Walkup (1979). On the expected value of a random assignment problem. *SIAM J. Comput. 8,* 440-442.

If the d_{ij} are drawn independently from the uniform distribution over [0,1] then, with high probability, the cost of the optimal assignment is less than 3.

R.M. Karp (1980). An algorithm to solve the $m\times n$ assignment problem in expected time $O(mn \log n)$. *Networks 10,* 143-152.

An algorithm for the construction of an optimal $m\times n$ assignment is presented. If the elements of each row of (d_{ij}) are independent identically distributed random variables then the expected execution time of the algorithm is $O(mn \log n)$. The best guaranteed time bound known for an assignment algorithm is $O(m^2n)$.

R. Loulou (1982). *Average Behavior of Heuristic and Optimal Solutions to the Maximization Assignment Problem,* Faculty of Management, McGill University, Montreal.

Let the d_{ij} be independent and exponentially distributed with rate λ. Let Z be the expected cost of a maximum-cost assignment. Then $1+1/n-1/\log n \leqslant \lambda Z/(n \log n) \leqslant 1+1/n+1/\log n$.

R.M. Karp (1984). *An Upper Bound on the Expected Cost of an Optimal Assignment,* Computer Science Division, University of California, Berkeley.

When the d_{ij} are drawn independently from the uniform distribution over [0,1], the expected cost of the optimal assignment is less than 2.

J.B.G. Frenk, A.H.G. Rinnooy Kan (1984). *Order Statistics and the Linear Assignment Problem,* Econometric Institute, Erasmus University, Rotterdam.

For a wide range of distribution functions F, the expected cost of the optimal assignment is asymptotic to $nF^{-1}(1/n)$.

4.4. *Network flow*

G.R. Grimmett, D.J.A. Welsh (1982). Flow in networks with random capacities. *Stochastics 7,* 205-229.
G.R. Grimmett, H.-C.S. Suen (1982). The maximal flow through a directed graph with random capacities. *Stochastics 8,* 153-159.
G.R. Grimmett, H. Kesten (1982). *First Passage Percolation, Network Flows and Electrical Resistances,* School of Mathematics, University of Bristol.

These papers prove existential results concerning the maximum value of a flow through certain randomly capacitated networks.

4.5. *Asymmetric traveling salesman*

An instance of the asymmetric traveling salesman problem is specified by an $n \times n$ matrix (d_{ij}). The object is to find a cyclic permutation σ to minimize $\Sigma_{i=1}^{n} d_{i\sigma(i)}$. If d_{ij} is interpreted as the distance from city i to city j then the problem amounts to finding a closed tour of minimum total distance which passes through each city exactly once. The problem is $\mathcal{NP}$-hard, and it is known that a polynomial-time approximation algorithm with good worst case performance does not exist unless $\mathcal{P} = \mathcal{NP}$. Probabilistic analyses of the problem usually assume that the d_{ij} are drawn independently from a common distribution. Another variant, considered in §5.5, is the Euclidean traveling salesman problem, in which the cities are points in the plane and distance is Euclidean distance.

M. Bellmore, J.C. Malone (1971). Pathology of traveling-salesman subtour-elimination algorithms. *Oper. Res. 19,* 278-307, 1766.

A branch-and-bound algorithm based on subtour elimination is claimed to solve random asymmetric traveling salesman problems in a polynomial-bounded expected number of steps. The argument neglects certain conditioning effects.

E.H. Guimady, V.A. Pereplitsa (1974). An asymptotical approach to solving the traveling salesman problem. *Upravljaemye Sistemy*, 35-45.

The nearest neighbor algorithm is analyzed on the asssumption that the d_{ij} are drawn independently from a uniform distribution.

J.K. Lenstra, A.H.G. Rinnooy Kan (1978). On the expected performance of branch-and-bound algorithms. *Oper. Res. 26*, 347-349.

A flaw is pointed out in the alleged proof of [Bellmore & Malone 1971] that a branch-and-bound algorithm for the asymmetric traveling salesman problem runs in polynomial expected time.

T. Leipala (1978). On the solutions of stochastic traveling salesman problems. *European J. Oper. Res. 2*, 291-297.

Stochastic upper and lower bounds on the length of the optimal tour are derived under various probabilistic assumptions. The upper bounds come from an analysis of the nearest neighbor algorithm.

R.S. Garfinkel, K.C. Gilbert (1978). The bottleneck traveling salesman problem: algorithms and probabilistic analysis. *J. Assoc. Comput. Mach. 25*, 435-448.

The bottleneck traveling salesman problem asks for a tour in which the weight of the heaviest arc is minimized. The distribution of the cost of such a tour is studied under the assumption that the edge weights are drawn independently from a common distribution. The results are closely related to recent work on the existence of Hamiltonian cycles in random digraphs (see §3.7).

R.M. Karp (1979). A patching algorithm for the nonsymmetric traveling-salesman problem. *SIAM J. Comput 8*, 561-573.

A patching algorithm is given which solves the asymmetric traveling salesman problem by solving the assignment problem and then patching the cycles of the optimal assignment together to form a tour. If the d_{ij} are drawn independently from the uniform distribution over [0,1], then the expected ratio between the cost of the tour produced and the cost of the optimal tour tends to 1. This result is refined in [Karp & Steele 1985] (see below).

J.-C. Panayiotopoulos (1982). Probabilistic analysis of solving the assignment problem for the traveling salesman problem. *European J. Oper. Res. 9*, 77-82.

The traveling salesman problem is solved by generating the assignments in order of increasing costs, until the first cyclic permutation is found. The analysis of this method ignores certain conditioning effects.

R.M. Karp, J.M. Steele (1985). Probabilistic analysis of heuristics. E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys (eds.). *The Traveling Salesman Problem*, Wiley, Chichester, Ch. 6.

This chapter surveys and extends the existing results on the probabilistic

analysis of the Euclidean and the asymmetric traveling salesman problems. For the first problem, see §5.5. For the second, a variant of the patching algorithm from [Karp 1979] is analyzed. It is shown that, if the d_{ij} are drawn independently from the uniform distribution over [0,1], then $E[(\text{PATCH}-\text{OPT})/\text{OPT}] = O(n^{-1/2})$. Here OPT is the cost of an optimal tour and PATCH is the cost of the tour produced by the patching algorithm.

4.6. *Quadratic assignment*

The quadratic assignment problem generalizes many $\mathcal{NP}$-hard combinatorial optimization problems, including the traveling salesman problem. An instance is specified by two $n \times n$ matrices (a_{ij}) and (b_{ij}). In the standard quadratic assignment problem the objective is to find a permutation σ that minimizes $\Sigma_{i,j} a_{ij} b_{\sigma(i)\sigma(j)}$. In the *bottleneck* quadratic assignment problem the objective is to minimize over all σ $\max_{i,j}\{a_{ij} b_{\sigma(i)\sigma(j)}\}$. The problem is called *planar* if the a_{ij} are distances between points in the plane.

R.E. Burkard, U. Fincke (1982A). On random quadratic bottleneck assignment problems. *Math. Programming 23,* 227-232.

R.E. Burkard, U. Fincke (1982B). Probabilistic asymptotic properties of quadratic assignment problems. *Z. Oper. Res. 27,* 73-81.

The ratio between the maximum and the minimum cost of an assignment a.s. tends to 1 as $n \to \infty$. This holds for the standard as well as the bottleneck problem, both when the a_{ij} are arbitrary and when they are planar distances (for any L_q-norm).

J.B.G. Frenk, M. van Houweninge, A.H.G. Rinnooy Kan (1982). *Asymptotic Properties of Assignment Problems,* Econometric Institute, Erasmus University, Rotterdam.

The results from [Burkard & Fincke 1982B] are generalized. E.g., an explicit expression for the asymptotic optimal solution value is obtained.

4.7. *Miscellaneous*

B.W. Weide (1980). Random graphs and graph optimization problems. *SIAM J. Comput. 9,* 552-557.

A method is presented to relate results regarding the probability of existence of certain subgraphs in random graphs $G_{n,p}$ to the probabilistic behavior of the optimal solution value to graph optimization problems, where the edge weights are independently chosen from an arbitrary distribution. The method is illustrated on the bottleneck and the standard traveling salesman problem.

G.S. Lueker (1981). Optimization problems on graphs with independent random edge weights. *SIAM J. Comput. 10,* 338-351.

Optimization problems are considered in which the input is a complete graph on n vertices together with a numerical weight for each edge. The edge weights are assumed to be drawn independently from the normal distribution with mean 0 and variance 1. The expected value of the maximum cost of a traveling salesman tour and the expected value of the cost of a maximum spanning tree are both asymptotic to $n\sqrt{(2 \log n)}$. If k is fixed then the expected maximum cost of a k-clique is asymptotic to $k\sqrt{((k-1)\log n)}$. For the traveling salesman problem, a simple greedy algorithm achieves the same expected asymptotic behavior as the optimal solution. A greedy algorithm for the maximum weight clique problem is also considered.

R.E. Burkard, U. Fincke (1982). *Probabilistic Asymptotic Properties of Some Combinatorial Optimization Problems*, Bericht 82-3, Institut für Mathematik, Technische Universität Graz.

Continuing their work reviewed in §4.6, the authors consider problems of the form $\min_{S \in T}\{\Sigma_{e \in S} c(e)\}$ and $\min_{S \in T}\{\max_{e \in S}\{c(e)\}\}$, where T is a family of subsets of an m-element set and each element e in the universe has a cost $c(e)$. The costs are assumed to be drawn independently from the uniform distribution over [0,1]. Conditions are given under which, for every $\epsilon > 0$, the probability tends to 1 as $m \rightarrow \infty$ that the ratio between the cost of the worst solution and the cost of the best solution is less than $1+\epsilon$. This result applies to quadratic assignment problems as well as to certain network flow, linear programming and matching problems.

5. Euclidean Problems

This section is concerned with combinatorial optimization problems whose specification includes a set of points in Euclidean space. Probabilistic analyses of optimal solution values and approximation algorithms for such problems often start from the assumption that the points are independent and uniformly distributed over a fixed 2-dimensional region, e.g. a circle or a square. Many results can be extended to other distributions and to higher dimensions.

In addition to the Euclidean problems dealt with below, results have been obtained for Euclidean *quadratic assignment* problems. These have been reviewed in §4.6.

5.1. *Closest points*

J.L. Bentley, M.I. Shamos (1978). Divide and conquer for linear expected time. *Inform. Process. Lett. 7*, 87-91.

A divide-and-conquer scheme is proposed to find the convex hull of n points in the plane in $O(n \log n)$ worst case and $O(n)$ expected time.

J.L. Bentley, B.W. Weide, A.C. Yao (1980). Optimal expected-time algorithms for closest-point problems. *ACM Trans. Math. Software 6*, 563-580.

A basic approach to solve closest point problems is the *cell technique,* which partitions the region containing the n points into cells with a constant expected number of points per cell. Spiral search of neighboring cells solves the nearest neighbor problem for arbitrary dimension in $O(1)$ expected time. In the 2-dimensional case, the cell technique allows construction of the Voronoi diagram, and hence of the minimum spanning tree, in $O(n)$ expected time. In higher dimensions, the minimum spanning tree problem requires $O(n \log\log n)$ time, and it is an open question if $O(n)$ expected time can be achieved.

P. Lehert (1981). Clustering by connected components in $O(n)$ expected time. *RAIRO Inform. 15,* 207-218.

Connected components of a graph, defined by a threshold distance, are found through the cell technique in linear expected time, provided that the L_∞-metric is used.

J.M. Steele (1982). Optimal triangulation of random samples in the plane. *Ann. Probab. 10,* 548-553.

The length of a minimal triangulation of n points in the Euclidean plane is a.s. asymptotic to $\alpha\sqrt{n}$ for some constant α. This settles a conjecture of G. Turán.

5.2. *Shortest paths*

R. Sedgewick, J.S. Vitter (1984). Shortest paths in Euclidean graphs. *Proc. 25th Annual IEEE Symp. Foundations of Computer Science.*

For a variety of Euclidean random graph models with n vertices and m edges, the authors' algorithm finds the shortest path between a specified pair of vertices in $O(n)$ expected time. Classical algorithms require $O(n^2)$ time for dense graphs and $O(n \log^2 n)$ time for sparse graphs on average.

5.3. *Matching*

C.H. Papadimitriou (1978). The probabilistic analysis of matching heuristics. *Proc. 15th Annual Allerton Conf. Communication, Control and Computing,* 368-378.

Steele's asymptotic result for subadditive Euclidean functionals (see [Steele 1981B], §5.5) implies that the optimal value of a Euclidean matching is a.s. asymptotic to $\beta\sqrt{n}$, for some constant β. It is shown that $0.25 \leqslant \beta \leqslant 0.40106$ and conjectured that $\beta \simeq 0.35$.

5.4. *Location*

Two basic location problems are the *K-median* and the *K-center* problem, in which K locations from among n given points are to be chosen so as to minimize the sum and the maximum respectively of the distances of each point to the nearest location. Asymptotic analyses for these problems necessarily involve an assumption about the growth rate of K as a function of n.

M.L. Fisher, D.S. Hochbaum (1980). Probabilistic analysis of the planar K-median problem. *Math. Oper. Res. 5*, 27-34.

A *partitioning* heuristic, much in the spirit of Karp's traveling salesman heuristic (see [Karp 1977], §5.5), splits the square into congruent subsquares and solves the weighted K-median problem on these subsquares, with the weight of each subsquare equal to the number of points that it contains. The heuristic is asymptotically optimal in probability, and the asymptotic optimal solution value lies a.s. in $[\gamma' n / \sqrt{K}, \gamma'' n / \sqrt{K}]$, for constants γ', γ''. (Cf. Ch.9, §5.2.)

C.H. Papadimitriou (1981). Worst-case and probabilistic analysis of a geometric location problem. *SIAM J. Comput. 10*, 542-557.

For the case that $K = o(n/\log n)$, a *honeycomb* heuristic for the K-median problem, which divides the region into regular hexagons, is asymptotically optimal in probability, and the optimal solution value is a.s. asymptotic to $\gamma n / \sqrt{K}$, with $\gamma = 2^{1/4}/3^{3/4}$. The proof is based on the observation that, as $n \to \infty$, the discrete problem approaches the continuous problem in which demand is not concentrated at points but spread uniformly over the region. For the case that $K = \Theta(n)$, a *partitioning* heuristic is asymptotically optimal in probability, under the assumption that the points are generated by a Poisson process. (Cf. Ch.9, §5.2.)

D.S. Hochbaum, J.M. Steele (1981). Steinhaus' geometric location problem for random samples in the plane. *Adv. in Appl. Probab. 14*, 56-67.

For the case that $K = \Theta(n)$, Steele's asymptotic result for subadditive Euclidean functionals (see [Steele 1981B], §5.5) is extended to the K-median problem, so that the optimal solution value is a.s. asymptotic to $\delta\sqrt{n}$, for some constant δ. We note that for $K = \Theta(n)$ the K-center problem is still open and that for neither model an a.s. asymptotically optimal heuristic is known.

E. Zemel (1984). Probabilistic analysis of geometric location problems. *Ann. Oper. Res. 1*.

For the case that $K = o(n/\log n)$, the *honeycomb* heuristic is a.s. asymptotically optimal for both the K-median and the K-center problem; the relation to the continuous version of these problems again provides the clue to the analysis. The optimal solution value for the K-center problem is a.s. asymptotic to $\epsilon / \sqrt{K}$, with $\epsilon \simeq 0.377$.

We conclude this subsection with a paper on a related problem.

C.H. Papadimitriou (1978). The complexity of the capacitated tree problem. *Networks 8,* 217-230.

The capacitated tree problem is to link customers to a depot by means of a spanning tree such that deletion of the depot yields components of bounded cardinality. The proof that the proposed partitioning heuristic is a.s. asymptotically optimal neglects the dependence of the region over which the asymptotic analysis is carried out on the actual sample.

5.5. *Routing*

The seminal work in the probabilistic analysis of Euclidean problems has been carried out in the context of routing problems and, more specifically, of the Euclidean traveling salesman problem.

J. Beardwood, J.H. Halton, J.M. Hammersley (1959). The shortest path through many points. *Proc. Cambridge Philos. Soc. 55,* 299-327.

The length of the optimal traveling salesman tour through n cities is a.s. asymptotic to $\zeta\sqrt{n}$, where ζ depends on the size and shape of the region and on the probability distribution of the cities over it.

R.M. Karp (1977). Probabilistic analysis of partitioning algorithms for the traveling-salesman problem in the plane. *Math. Oper. Res. 2,* 209-224.

Karp's *partitioning* heuristic constructs a tour by dividing the region into subregions, each containing a small number of points, and then linking optimal tours through each subregion together. The absolute error of the heuristic grows more slowly than $\sqrt{n}$. Hence, the above result from Beardwood *et al.* implies that the relative error tends to 0 a.s. Depending on whether or not the partitioning scheme takes the actual location of the cities into account, the running time of the method is in expectation or deterministically polynomial.

J.M. Steele (1981A). Complete convergence of short paths and Karp's algorithm for the TSP. *Math. Oper. Res. 6,* 374-378.

Steele establishes complete convergence for the result of Beardwood *c.s.*, something that Karp in his above paper had tacitly assumed.

J.M. Steele (1981B). Subadditive Euclidean functionals and nonlinear growth in geometric probability. *Ann. Probab. 9,* 365-376.

A generalization of the BHH result is established for a class of subadditive functions defined on random sets of independently distributed points. Examples of such functions are the lengths of an optimal traveling salesman tour, of a rectilinear Steiner tree, and of a minimum matching.

J.H. Halton, R. Terada (1982). A fast algorithm for the Euclidean traveling salesman problem, optimal with probability one. *SIAM J. Comput. 11,* 28-46.

A partitioning method, similar to Karp's approach, yields asymptotically optimal tours a.s., whereas its running time is almost linear in probability.

R.M. Karp, J.M. Steele (1985). Probabilistic analysis of heuristics. E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys (eds.). *The Traveling Salesman Problem,* Wiley, Chichester, Ch. 6.

This chapter reviews probabilistic analyses for traveling salesman problems. See also §4.5.

J.M. Steele (to appear). Probabilistic algorithm for the directed traveling salesman problem. *Math. Oper. Res.*

In a model for random Euclidean asymmetric traveling salesman problems, the expected optimal tour length is shown to be asymptotic to $\eta\sqrt{n}$. A partitioning heuristic is presented whose relative error tends to 0 in expectation.

A. Marchetti-Spaccamela, A.H.G. Rinnooy Kan, L. Stougie (to appear). Hierarchical routing problems. *Networks.*

The two-stage planning problem under consideration asks for the determination of a number of vehicles minimizing the sum of the acquisition costs and the length of the longest tour through the customers assigned to any vehicle. At the time of acquiring the vehicles, all that is known about the customers is that they are uniformly distributed over a circular region. The asymptotic optimal solution value and an asymptotically optimal heuristic are derived. See Ch.11, §12.2 for related material.

M. Haimovich, A.H.G. Rinnooy Kan (to appear). Bounds and heuristics for capacitated routing problems. *Math. Oper. Res.*

A probabilistic value analysis for a capacitated Euclidean multisalesmen problem is presented, together with several heuristics whose relative error tends to 0 a.s.

6. Linear Programming

The linear programming problem requires the minimization of a linear function subject to linear constraints. It can be written in the form $\min_x \{c^T x : Ax \geqslant b, x \geqslant 0\}$, where $c, x \in \mathbb{R}^d$, $A \in \mathbb{R}^{m \times d}$, and $b \in \mathbb{R}^m$.

The outstanding practical experience obtained with the *simplex method* for linear programming is in sharp contrast to its exponential worst case behavior. Only recently, a sequence of papers has appeared that provides a beginning of an analytical explanation of the method's efficiency, by demonstrating that the expected number of pivots for certain simplex variants is bounded by a polynomial function of d and m.

K.H. Borgwardt (1982). Some distribution-independent results about the asymptotic order of the average number of pivot steps of the simplex method.

Math. Oper. Res. 7, 441-462.
K.H. Borgwardt (1982). The average number of pivot steps required by the simplex-method is polynomial. *Z. Oper. Res. 26,* 157-177.

Borgwardt investigates linear programs of the form $\max\{c^T x : Ax \leqslant e\}$, where $c, x \in \mathbb{R}^d$ and $A \in \mathbb{R}^{n \times d}$ $(n \geqslant d)$. Let A_i^T denote the ith row of A. The behavior of a parametric simplex variant (the *shadow vertex* algorithm) is analyzed in a probabilistic model in which the vectors $c, A_1, \ldots, A_n$ are independently drawn from the same spherically symmetric distribution. All problems generated in this way are feasible. In the second paper, the expected number of pivots is shown to be $O(d^4 n)$.

These papers received the 1982 Lanchester prize of the Operations Research Society of America.

S. Smale (1983). On the average speed of the simplex method of linear programming. *Math. Programming 27,* 241-262.
S. Smale (1983). The problem of the average speed of the simplex method. A. Bachem, M. Grötschel, B. Korte (eds.). *Mathematical Programming: the State of the Art - Bonn 1982,* Springer, Berlin, 530-539.

Smale analyzes the behavior of Dantzig's *self-dual* simplex method, when viewed as a special case of Lemke's algorithm for the linear complementarity problem, in a probabilistic model in which the data are independently drawn from a spherically symmetric distribution. This model does not allow any kind of degeneracy. The expected number of pivots is shown to be $O(c_d(\log m)^{d(d+1)})$, where c_d depends superexponentially on d.

C. Blair (1983). *Random Linear Programs with Many Variables and Few Constraints,* Working paper 946, College of Business Administration, University of Illinois, Urbana-Champaign.

A much simplified proof for a slightly weaker version of Smale's results.

M. Haimovich (1983). *The Simplex Method is Very Good! - On the Expected Number of Pivot Steps and Related Properties of Random Linear Programs,* Columbia University, New York.
I. Adler (1983). *The Expected Number of Pivots Needed to Solve Parametric Linear Programs and the Efficiency of the Self-Dual Simplex Method,* Department of Industrial Engineering and Operations Research, University of California, Berkeley.

Adler and Haimovich independently investigate the length of a path generated by some parametric simplex variants, in a probabilistic model that requires only almost sure nondegeneracy and 'sign invariance', i.e., invariance of the distribution under changing the sign of any subset of rows or columns. This model enables a simpler probabilistic analysis in terms of elegant combinatorial counting arguments. The expected length of a path from worst to best solution is shown to be $\min\{m, d\} + 1$; but the analysis presupposes that a feasible initial vertex is given.

I. Adler, R.M. Karp, R. Shamir (1983). *A Family of Simplex Variants Solving an $m \times d$ Linear Program in Expected Number of Pivots Depending on d Only*, Report UCB CSD 83/157, Computer Science Division, University of California, Berkeley.

Adler, Karp & Shamir analyze a family of algorithms that proceed according to a *constraint-by-constraint* principle (and that include a whole family of known simplex variants) in a general probabilistic model that does not require sign invariance. The expected number of pivots is shown to be bounded by an exponential function of d, independent of m.

I. Adler, R.M. Karp, R. Shamir (1983). *A Simplex Variant Solving an $m \times d$ Linear Program in $O(\min(m^2, d^2))$ Expected Number of Pivot Steps*, Report UCB CSD 83/158, Computer Science Division, University of California, Berkeley.

I. Adler, N. Megiddo (1983). *A Simplex Algorithm Whose Average Number of Steps is Bounded Between Two Quadratic Functions of the Smaller Dimension.*

M.J. Todd (1983). *Polynomial Expected Behavior of a Pivoting Algorithm for Linear Complementarity and Linear Programming Problems*, Technical report 595, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY.

Adler, Karp & Shamir analyze a parametric version of the *constraint-by-constraint* method and Adler & Megiddo and Todd analyze the *self-dual* algorithm in a probabilistic model that is closely related to that used by Adler and Haimovich. As Megiddo pointed out, the lexicographic versions that are subjected to these independent analyses execute the same sequence of pivots. The expected number of pivots is shown to be $O(\min\{m^2, d^2\})$. Under stronger probabilistic assumptions, Adler & Megiddo also obtain a quadratic *lower* bound.

R. Shamir (1984). *The Efficiency of the Simplex Method: a Survey*, Department of Industrial Engineering and Operations Research, University of California, Berkeley.

§6 of this survey presents a more detailed review and a useful assessment of the above material.

7. Packing and Covering

7.1. *Satisfiability and tiling*

The satisfiability problem is the problem of deciding whether there exists an assignment of truth values to variables that makes a given Boolean formula true and was proved to be $\mathcal{NP}$-complete by S.A. Cook in 1971. The tiling problem was proved to be $\mathcal{NP}$-complete by L.A. Levin, the second discoverer of $\mathcal{NP}$-completeness theory, in 1973. For more information on the material reviewed in this subsection and, in particular, for an exposition of Levin's results on the 'random tiling' problem reported in [Levin 1984], see [Johnson

1984] (§1).

J.M. Plotkin, J.W. Rosenthal (1978). On the expected number of branches in analytic tableaux analysis in propositional calculus. *Notices Amer. Math. Soc. 25*, A-437.

Let E_n denote the expected number of branches generated when the method is applied to a random formula in which AND, OR and NOT are the only connectives, negation is applied only to atomic formulas and there are n occurrences of AND and OR. Then $c(1.08)^n \leqslant E_n \leqslant d(1.125)^n$, where c and d are constants.

A. Goldberg, P. Purdom, C. Brown (1982). Average time analysis of simplified Davis-Putnam procedures. *Inform. Process. Lett. 15*, 72-75. Corrigendum. *Inform. Process. Lett. 16*, 213.

A random conjunctive normal form formula consists of n independently generated random clauses. In each clause, each of the m variables occurs positively with probability p, negatively with probability p, and is absent with probability $1-2p$. The expected time for a backtracking algorithm due to Davis & Putnam to determine whether a random formula is satisfiable is polynomial in n and m (but exponential in $1/p$).

J. Franco, M. Paull (1983). Probabilistic analysis of the Davis Putnam procedure for solving the satisfiability problem. *Discrete Appl. Math. 5*, 77-87.

For random formulas with n clauses, m variables and constant clause lengths, the expected running time of the Davis-Putnam procedure is exponential. Note that the above result of Goldberg *et al.* applies to a model in which the expected clause length is proportional to m.

L.A. Levin (1984). Problems, complete in 'average' instance. *Proc. 16th Annual ACM Symp. Theory of Computing*, 465.

Levin develops a theory of $\mathcal{NP}$-completeness for the average rather than the worst case. He defines a class RANDOM $\mathcal{NP}$ of pairs (X,μ), where X may be any problem in $\mathcal{NP}$ and μ is a probability measure whose distribution function is computable in polynomial time, and a new notion of polynomial transformation within this class, which transforms distributions as well as instances. He proves that RANDOM TILING, i.e., the tiling problem with a very natural distribution of its instances, is complete in RANDOM $\mathcal{NP}$.

7.2. *Bin packing*

E.G. Coffman, Jr., M.R. Garey, D.S. Johnson (1983). *Approximation Algorithms for Bin-Packing - An Updated Survey,* Bell Laboratories, Murray Hill, NJ.

A thorough survey with excellent coverage of recent deterministic and probabilistic results.

(a) 1-*dimensional bin packing*

This is the problem of packing n items into a minimum number of bins of capacity 1. Associated with each item is a positive real number less than 1 called its *size*. The sum of the sizes of the items packed into a single bin may not exceed the bin capacity. Associated with any packing is the wasted space, defined as the number of bins used minus the sum of the sizes of all the items.

In probabilistic analyses of this problem it is usually assumed that the item sizes are drawn independently from a common distribution. Often, the uniform distribution over [0,1], or, less commonly, over [a,b], is postulated. A probability distribution is said to allow *perfect packing* if, with probability tending to 1 as $n \to \infty$, the waste is bounded above by some function which is $o(n)$. The two algorithms most commonly studied are *next fit*, an on-line algorithm which packs the items in their given order and starts a new bin whenever the next item cannot fit in the present bin, and *first fit decreasing*, which considers the items in decreasing order of size and places each item in the first bin that can accept it. In addition to the probabilistic studies reported here there is a very extensive literature on the worst case performance of bin packing algorithms (see [Coffman, Garey & Johnson 1983] above).

S.D. Shapiro (1977). Performance of heuristic bin packing algorithms with segments of random length. *Inform. and Control 35,* 146-158.

An approximate analysis of the next fit algorithm is given, on the assumption that the item sizes are drawn independently from an exponential distribution.

E.G. Coffman, Jr., M. Hofri, K. So, A.C. Yao (1980). A stochastic model of bin packing. *Inform. and Control 44,* 105-115.

On the assumption that the item sizes are uniformly distributed over [0,1], an upper bound is derived on the expected number of bins required by the next fit algorithm.

G.N. Frederickson (1980). Probabilistic analysis for simple one- and two-dimensional bin packing algorithms. *Inform. Process. Lett. 11,* 156-161.

For a simple 1-dimensional bin-packing algorithm it is shown that the expected waste is $O(n^{2/3})$, on the assumption that the item sizes are uniformly distributed over [0,1]. Since the given algorithm always uses at least as many bins as the first fit decreasing or best fit heuristics, the expected waste for these heuristics is also $O(n^{2/3})$. Several authors later improved this bound to $O(n^{1/2})$. Implications for 2-dimensional strip packing are also explored.

W. Knödel (1981). A bin packing algorithm with complexity $O(n \log n)$ and performance 1 in the stochastic limit. J. Gruska, M. Chytil (eds.). *Mathematical Foundations of Computer Science 1981,* Lecture Notes in Computer Science 118, Springer, Berlin, 369-378.

A simple packing algorithm has expected waste $O(n^{1/2})$ when the item sizes are drawn independently from a decreasing probability distribution over [0,1].

R. Loulou (1982). *Probabilistic Behavior of Optimal Bin Packing Solutions,* Faculty of Management, McGill University, Montreal.

Every convex distribution function on [0,1] with bounded second derivative on [0,½] allows perfect packing.

M. Hofri (1982). *Bin-Packing: an Analysis of the Next-Fit Algorithm,* Technical report 242, Department of Computer Science, Technion, Haifa.

The next fit algorithm is analyzed on the assumption that the item sizes are uniformly distributed over [0,1]. Detailed information about the distribution of the number of bins is derived.

N. Karmarkar (1982). Probabilistic analysis of some bin-packing algorithms. *Proc. 23rd Annual IEEE Symp. Foundations of Computer Science,* 107-111.

On the assumption that the item sizes are drawn from the uniform distribution over $[0,a]$, a closed form expression is derived for the expected number of bins required by the next fit algorithm. It is also proved that the uniform distribution over $[0,a]$ permits perfect packing.

G.S. Lueker (1983). *An Average-Case Analysis of Bin Packing with Uniformly Distributed Item Sizes,* Report 181, Department of Information and Computer Science, University of California, Irvine.

When the item sizes are uniformly distributed over [0,1], a simple algorithm which never places more than two items in a bin achieves expected waste $O(n^{1/2})$.

G.S. Lueker (1983). Bin packing with items uniformly distributed over intervals $[a,b]$. *Proc. 24th Annual IEEE Symp. Foundations of Computer Science,* 289-297.

For a large class of choices of the interval $[a,b]$ it is determined whether the uniform distribution over $[a,b]$ allows perfect packing. The analysis makes interesting use of a linear programming technique.

J.L. Bentley, D.S. Johnson, F.T. Leighton, C.C. McGeoch, L.A. McGeoch (1984). Some unexpected expected behavior results for bin packing. *Proc 16th Annual Symp. Theory of Computing,* 279-288.

For the uniform distribution over [0,1], the first fit algorithm has expected waste $O(n^{0.8})$. For the uniform distribution over $[0,a]$, $a \leq ½$, the first fit decreasing algorithm has expected waste $O(1)$. These results were entirely unexpected and are of extraordinary interest.

P.W. Shor (1984). The average-case analysis of some on-line algorithms for bin packing. *Proc. 25th Annual IEEE Symp. Foundations of Computer Science.*

For the uniform distribution over [0,1], tighter upper and lower bounds on the performance of on-line algorithms are derived. The analysis is based on the relation between the bin packing problem and a 2-dimensional matching problem.

(b) *d-dimensional bin packing*

This is the problem of packing d-dimensional items into d-dimensional bins. The variants of the problem include rectangle packing, strip packing and vector packing. In *rectangle packing* the items are d-dimensional rectangles with sides parallel to the coordinate axes and each bin is a unit hypercube. In 2-dimensional *strip packing* the items are again rectangles with sides parallel to the coordinate axes but there is a single bin of fixed width and infinite height. The object is to minimize the vertical extent of the area used for packing. In *vector packing*, each bin satisfies the following constraint: for $i = 1,2,\ldots,d$, the sum of the ith coordinates of the items in the bin is less than or equal to 1.

M. Hofri (1980). Two-dimensional packing: expected performance of simple level algorithms. *Inform. and Control 45*, 1-17.

The problem of packing rectangles into a semi-infinite strip is discussed. The vertical and horizontal dimensions of the rectangles are assumed to be independent random variables drawn from the uniform distribution over [0,1]. The expected efficiencies of the next fit, rotatable next fit and next fit decreasing algorithms are derived.

R.M. Karp, M. Luby, A. Marchetti-Spaccamela (1984). A probabilistic analysis of multidimensional bin packing problems. *Proc. 16th Annual ACM Symp. Theory of Computing*, 289-298.

A simple algorithm for rectangle packing is analyzed on the assumption that the dimensions of the items are independent and uniformly distributed over [0,1]. In the case $d = 2$, the expected waste is $\Omega(\sqrt{(n \log n)})$ and $O(\sqrt{n} \log n)$. For $d \geq 3$, the expected waste is $\Theta(n^{(d-1)/d})$. Probabilistic analyses of algorithms for strip packing and vector packing are also given.

7.3. *Multiprocessor scheduling*

This is the problem of scheduling n tasks with known execution times on m identical parallel processors to minimize the makespan, defined as the maximum, over all processors, of the sum of the execution times of the tasks assigned to that processor. Probabilistic analyses usually assume that the execution times are independent identically distributed random variables.

The problem can be viewed as a variant of the 1-dimensional bin packing problem in which the number of bins is fixed and the capacity of the largest bin is to be minimized.

An important class of on-line multiprocessor scheduling algorithms is formed by the *list scheduling* algorithms, in which the tasks are arranged in a linear list, and, whenever a processor completes a task, the first unscheduled task on the list is assigned to that processor. The LPT algorithm, in which the list is arranged in decreasing order of execution times, is especially popular. Another class of algorithms based on a *differencing operation* has recently been found to have excellent properties (see [Karmarkar & Karp] below).

In addition to the probabilistic work surveyed here, there is an extensive literature on the worst case analysis of multiprocessor scheduling algorithms; see [Coffman, Garey & Johnson 1983] in §7.2 and Ch.11, §4.2.

For the probabilistic analysis of a *single-machine scheduling* algorithm, see [Gazmuri 1981] in Ch.11, §11.1. For the probabilistic analysis of *hierarchical scheduling* systems, in which a processor acquisition phase precedes the actual scheduling phase, see Ch.11, §12.2.

J.L. Bruno, P.J. Downey (1982). *Probabilistic Bounds on the Performance of List Scheduling,* Technical report TR 82-19, Computer Science Department, University of Arizona, Tucson.

If the execution times are drawn independently from a uniform distribution then, with probability $\geqslant 1-\epsilon$, the ratio of the makespan of an LPT schedule to the optimal makespan is less than $1+f(m,n,\epsilon)$, where $f(m,n,\epsilon) \simeq 1+(2(m-1)/n)$. The analysis enables concrete numerical results on the distribution of the relative error to be obtained for small values of n.

E.G. Coffman, Jr., E.N. Gilbert (1983). *On the Expected Relative Performance of List Scheduling,* Technical memorandum, Bell Laboratories, Murray Hill, NJ.

The analysis in the above paper is refined and extended to exponential distributions.

J.B.G. Frenk, A.H.G. Rinnooy Kan (1983). *The Asymptotic Optimality of the LPT Heuristic,* Econometric Institute, Erasmus University, Rotterdam.

Let OPT denote the cost of an optimal solution, and let LPT be the cost of the solution obtained by the longest processing time first heuristic. If the execution time distribution has a finite second moment and positive density at zero, then LPT-OPT converges a.s. and in expectation to 0 as $n \rightarrow \infty$.

R. Loulou (1984). Tight bounds and probabilistic analysis of two heuristics for parallel processor scheduling. *Math. Oper. Res. 9,* 142-150.

Let the execution times of the tasks be independent and identically distributed with finite mean. Let RLP be the cost of the solution obtained by applying list scheduling to a randomly ordered list of the tasks. Let m be the number of processors. When $m \geqslant 2$ the random variable RLP-OPT is stochastically bounded by a finite random variable for any value of n. When $m = 2$, a certain upper bound on this random variable converges in distribution as $n \rightarrow \infty$. When $m \geqslant 2$, LPT-OPT is stochastically smaller than a fixed random variable which does not depend on n.

N. Karmarkar, R.M. Karp (to appear). The differencing method of set partitioning. *Math. Oper. Res.*

A simple differencing algorithm is presented. If the item sizes are drawn independently from a smooth distribution with bounded support then, with probability tending to 1, the difference between the completion times of the last and first machines to complete is bounded above by a quantity of the form $n^{-c \log n}$.

N. Karmarkar, R.M. Karp, G.S. Lueker, A. Odlyzko (1984). *The Probability Distribution of the Optimal Value of a Partitioning Problem*, Bell Laboratories, Murray Hill, NJ.

The 2-processor scheduling problem is considered under the assumption that the execution times are drawn independently from the uniform distribution over [0,1]. Let D denote the minimum, over all partitions of the tasks into two sets, of the absolute difference between the sums of the execution times of the tasks in the two sets. Then, with probability tending to 1 as $n \rightarrow \infty$, D is bounded between two quantities of the form $cn/2^n$. If n is even and each set is required to contain exactly $n/2$ elements, then, with probability tending to 1, the minimum absolute difference is bounded above by $cn^2/2^n$.

7.4. *Knapsack and subset sum*

The knapsack problem is the zero-one integer programming problem with a single linear constraint. The problem can be described as follows. A set of n items is given. Each item has a specified weight and a specified value. The objective is to select a set of items of maximal total value, such that the sum of the weights of the selected items does not exceed a given bound called the capacity of the knapsack. The knapsack problem is $\mathcal{NP}$-hard, but good worst case approximation algorithms whose execution time is quadratic in the length of the input are known.

The subset sum problem is an ($\mathcal{NP}$-complete) special case of the knapsack problem. The objective is to select a set of items whose total weight is equal to the capacity of the knapsack.

V. Chvátal (1980). Hard knapsack problems. *Oper. Res. 28*, 1402-1411.

A general class of recursive algorithms for the knapsack problem is introduced. These algorithms use the full power of branch-and-bound and dynamic programming as well as rudimentary divisibility arguments. If an n-item instance is generated by drawing the weights independently from the uniform distribution over $\{1,2, \ldots, 10^{n/2}\}$, setting each value equal to the corresponding weight and setting the knapsack capacity equal to half the sum of the weights, then, with probability tending to 1, every recursive algorithm requires at least $2^{n/2}$ steps to solve the instance.

G. d'Atri (1979). *Analyse Probabilistique du Problème du Sac-à-Dos,* Thèse, Université de Paris VI.

The coefficients of the n-variable knapsack problem are assumed to be drawn independently from the uniform distribution over $\{1,2,\ldots,n\}$ and the capacity of the knapsack is drawn from the uniform distribution over $\{1,2,\ldots,cn\}$. A linear time algorithm consisting of a greedy phase followed by an adjustment phase obtains an optimal solution with probability tending to 1.

V. Lifschitz (1980). The efficiency of an algorithm of integer programming: a probabilistic analysis. *Proc. Amer. Math. Soc. 79,* 72-76.

If the coefficients in an n-variable knapsack problem are drawn independently from a common continuous distribution, then a simple enumerative method based on a dominance relation between solutions finds an optimal solution in expected time E_n, where $\ln E_n \simeq 2\sqrt{n}$.

A.M. Frieze, M.R.B. Clarke (1981). *Approximation Algorithms for the m-Dimensional 0-1 Knapsack Problem: Worst-Case and Probabilistic Analyses,* Queen Mary College, University of London.

The integer programming problem $\max\{\Sigma_j c_j x_j : \Sigma_j a_{ij} x_j \leqslant 1,\ i = 1,\ldots,m;\ x_j \in \{0,1\},\ j = 1,\ldots,n\}$ is considered, with m held fixed and n variable. The distribution of the optimal value OPT is considered, and it is shown that a simple rounding algorithm gives a solution of value ROUND, where OPT-ROUND $\leqslant \epsilon$.OPT with probability tending to 1, for all ϵ of the form $n^{-\alpha}$, $\alpha \leqslant 1/(m+1)$.

G.S. Lueker (1982). On the average difference between the solutions to linear and integer knapsack problems. R.L. Disney, T.J. Ott (eds.). *Applied Probability - Computer Science: the Interface, Volume I,* Birkhäuser, Basel, 489-504.

The expected difference between the values of the integer and linear versions of the 0-1 knapsack problem is $O(\log^2 n)$ and $\Omega(1/n)$ when the coefficients are drawn independently from the uniform distribution over $[0,1]$.

A.V. Goldberg, A. Marchetti-Spaccamela (1984). On finding the exact solution of a zero-one knapsack problem. *Proc. 16th Annual ACM Symp. Theory of Computing,* 359-368.

When the (weight, value) pairs are generated by a Poisson process with n as the expected number of items, then, for every $\epsilon>0$, there is a polynomial time algorithm that solves the knapsack problem to optimality with probability at least $1-\epsilon$. The algorithm depends on the parameters of the Poisson process, and its running time is exponential in $1/\epsilon$.

J.C. Lagarias, A.M. Odlyzko (1983). Solving low-density subset sum problems. *Proc. 24th Annual IEEE Symp. Foundations of Computer Science,* 1-10.

The weights are drawn independently from a uniform distribution over

$\{1, \ldots, 2^{n/d}\}$, and the capacity is the total weight of a randomly chosen subset of items. A polynomial time algorithm is developed that, for 'almost all' problems with $d < 1/n$, finds the subset of items whose total weight equals the capacity. The algorithm uses the lattice basis reduction algorithm due to A.K. Lenstra, H.W. Lenstra, Jr., and L. Lovász [*Math. Ann. 261* (1982), 515-534].

A.M. Frieze (1984). *On the Lagarias-Odlyzko Algorithm for the Subset Sum Problem*, Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA.

The above result of Lagarias & Odlyzko is proved in a simpler way and extended to $d < 2/n$.

7.5. *Set covering*

The set covering problem is of the form $\min\{e_n x\colon\ Ax \geqslant e_m;\ x_j \in \{0,1\},\ j = 1, \ldots, n\}$, where A is an $m \times n$ matrix of zeros and ones and e_h denotes the vector of h ones. A random (m,n,p) problem is one in which the elements of A are independent and each is equal to 1 with probability p.

J.F. Gimpel (1967). A stochastic approach to the solution of large set covering problems. *Proc. 8th Annual IEEE Symp. Switching and Automata Theory*, 76-82.

If $m = \lfloor an \rfloor$ and p is held constant as $n \to \infty$ then, with probability 1, the ratio between the cost of the optimal solution and the cost of the solution produced by a simple greedy algorithm tends to 1. This is one of the earliest results on the probabilistic analysis of combinatorial algorithms. The result is not true in general if p depends on n (see [Karp 1976] in §1).

C. Vercellis (1984). A probabilistic analysis of the set covering problem. *Ann. Oper. Res. 1*.

Here, p is constant and $m \to \infty$. If n grows faster than $\log m$ but remains polynomially bounded in M, then the set covering problem is a.s. feasible and the ratio between the optimal solution value and $\log m / \log(1/(1-p))$ tends to 1 a.s. A probabilistic analysis of the asymptotic behavior of two simple heuristics is performed.

8. Branch-and-Bound and Local Search

R.M. Karp, J. Pearl (1983). Searching for an optimal path in a tree with random costs. *Artificial Intelligence 21*, 99-116.

Let $F(T)$ be the minimum total weight of a root-leaf path in a tree T. In the case where T is a uniform binary tree of height n and the edge weights are independent Bernoulli random variables with mean p, the distribution of $F(T)$ is studied and polynomial time search strategies are shown to give good near-optimal paths with high probability. The model is proposed as an abstraction of branch-and-bound search.

H. Nakano, Y. Nakanishi (1983). An analysis of local neighborhood search method for combinatorial optimization problems. *Proc. Internat. IEEE Symp. Circuits and Systems*, 1055-1058.

The neighborhood search method of finding locally optimal solutions of combinatorial optimization problems is modeled by a Markov chain. In the case of the λ-opt method for the traveling salesman problem, the predictions of the model are in good agreement with experiment.

D.R. Smith (1984). Random trees and the analysis of branch and bound procedures. *J. Assoc. Comput. Mach. 31*, 163-188.

A model of random branch-and-bound search trees is investigated. The expected time and space complexity of the best bound first and depth first strategies are presented and compared. The results are applied to the traveling salesman problem with random inputs.

C.A. Tovey (1983). On the number of iterations of local improvement algorithms. *Oper. Res. Lett. 2*, 231-238.
C.A. Tovey (to appear). Low order polynomial bounds on the expected performance of local improvement algorithms. *Math. Programming Stud.*
C.A. Tovey (to appear). Hill climbing with multiple local optima. *SIAM J. Algebraic Discrete Methods.*

Tovey's model of local improvement algorithms in combinatorial optimization confirms empirical observations. The model predicts exponential worst case and low order polynomial average performance for problems with a single optimum, such as linear programming and linear complementarity problems. For hard problems with multiple local optima, average speed is linearly bounded but accuracy is poor.