ESPRIT BR Project RAND-REC
(EC-US Exploratory Collaborative Activity –
EC-US030)

Annual Progress Report

July 1, 1993 – June 30, 1994

Contents

1 Overview of Research Activities

The research within the project RAND-REC has concentrated on the three major areas (see Section 2, Research Papers):

(1) Design Of Efficient Randomized and Approximative Algorithms ([2], [3], [6], [7], [8], [14], [15]),

(2) VC Dimension and Sample Sizes of Sigmoidal Neural Networks and Efficient Learnability ([4], [10], [12]),

(3) Derandomized Probabilistic Methods and Algorithms ([9]),

(4) Deterministic and Randomized PET (Priority Encoding Transmission) Systems ([1], [4]).

2 Research Papers (RAND-REC)

1. Albanese, A., Bloemer, J., Edmonds, J., Luby, M. and Sudan, M.:
   Priority Encoding Transmission,
   Proc. 35th IEEE FOCS (1994), pp. 604–612.

2. Alon, N., Frieze, A. and Welsh, D.J.A.:
   Polynomial Time Randomized Approximation Schemes for the Tutte Polynomial of Dense Graphs,
   Proc. 35th IEEE FOCS (1994), pp. 24-35.

3. Arora, S., Karger, D. and Karpinski, M.:
   Polynomial Time Approximation Schemes for Dense Instances of NP-Hard Problems,
   Research Report No. 85119-CS, Univ. Bonn, 1994; to appear in 1995 ACM STOC.

4. Bloemer, J., Karp, R., Karpinski, M., Luby, M. and Zuckerman, D.:
   An XOR-Based Erasure-Resilient Coding Scheme,
   Preprint, ICSI Berkeley, 1994.

5. Bshouty, N. Hancock, T.R., Hellerstein, L. and Karpinski, M.:
   An Algorithm to Learn Read–Once Threshold Formulas, and Transformations between Learning Models,
   Journal of Computational Complexity 4, pp. 37–61.

6. Freivalds, R. and Karpinski, M.:
   Lower Space Bounds for Randomized Computation,
   Research Report No. 85104-CS, Universität Bonn, 1994,
   Proc. ICALP '94, Lecture Notes in Computer Science Vol 820 (1994), pp. 580–592.

7. Frieze, A. and Jerrum, M.:
Improved approximation algorithms for MAX k-CUT and MAX BISECTION,
Report ECS-LFCS-94-292, Department of Computer Science, Universityof Edinburgh, June 1994; to appear in: *Proceedings of the 4th Integer Programming and Combinatorial Optimisation Conference (IPCO4)*, Köbenhaven, May 1994; Journal version submitted to *Algorithmica.*

8. Frieze, A., Jerrum, M., Molloy, M., Robinson, R. and Wormald, N.:
Approximately counting Hamilton cycles in random regular graphs;
submitted to: *Journal of Algorithms.*

9. Goldmann, M. and Karpinski, M.:
Simulating Threshold Circuits by Majority Circuits (Extended Version),
Technical Report TR-94-030, International Computer Science Institute, Berkeley, 1994, submitted to SIAM Journal on Computing.

10. Karpinski, M.:
Approximation Hardness of Some Counting Problems in Algebra,
Preprint, Universität Bonn, 1994.

11. Karpinski, M. and MacIntyre, A.:
Polynomial Bounds for VC Dimension of Sigmoidal Neural Networks,
Research Report No. 85116-CS, Universität Bonn, 1994; to appear in 1995 ACM STOC.

12. Karpinski, M. and Rytter, W.:
On a Sublinear Time Parallel Construction of Optimal Binary Search Trees,
Research Report No. 85102-CS, Universität Bonn, 1993; in Proc. MFCS '94, Lecture Notes in Computer Science, Springer-Verlag Vol. 841 (1994), pp. 453–461.

13. Karpinski, M. and Werther, T.:
VC Dimension and Uniform Learnability of Sparse Polynomials and Rational Functions,
SIAM Journal of Computing **22** (1993), pp. 1276–1283.

14. Karpinski, M. and Shparlinski, I.:
Efficient Approximation Algorithms for Sparse Polynomials over Finite Fields,
Technical Report TR-94-029, International Computer Science Institute, Berkeley, 1994; to appear in Theoretical Computer Science.

15. Karpinski, M. and Zelikovsky, A.:
1.757 and 1.267 - Approximation Algorithms for the Network and Rectilinear Steiner Tree Problems,
Research Report No. 85115-CS, Universität Bonn, 1994.

16. Santha, M. and Tan, S.:
Verifying the Determinant in Parallel,
Proc. 5th International Symposium on Algorithms and Computation, INCS, Springer-Verlag, 1994.

17. Welsh, D.J.A:
Complexity: Knots, Colourings and Counting,
London Mathematical Society Lecture Note Series 186, Cambridge, University Press (1993), pp. 1–163.

18. Welsh, D.J.A.:
The Random Cluster Process,
Discrete Mathematics 136 (1994), pp. 373–390.

3 Reports on visits

Report on a visit to Carnegie Mellon University supported by RAND–REC (M. Jerrum)

I visited the Mathematics Department of Carnegie Mellon University from 1st to 31st June 1994 for collaborative research with Alan Frieze and Ravi Kannan. The main topics we considered were (a) extensions of a recently introduced technique of Goemans and Williamson for combinatorial optimisation, and (b) learning product distributions and classes of convex bodies in high dimensional Euclidean space. Topic (a) has been thoroughly worked out and is the subject of a report [1], while (b) is more speculative. I shall describe (a) here, and leave (b) until such time as the picture becomes clearer.

Goemans and Williams [2] have recently achieved a significant advance in the theory of approximation algorithms. Previous work on approximation algorithms for problems in combinatorial optimisation was largely dependent on comparing a heuristic solution value to that of an Linear Programming (LP) relaxation, either implicitly or explicitly. The main novelty of [2] is that it uses a Semi-Definite Program (SDP) as a relaxation. To be more precise let us consider the problem MAX-CUT studied (among others) in [2]: we are given a vertex set V, with $|V| = n$ and non-negative weights $w_{i,j}$ for $1 \leq i, j \leq n$, where $w_{i,j} = w_{j,i}$ and $w_{i,i} = 0$ for all i, j. If $S \subseteq V$ and $\bar{S} = V \setminus S$ then the *weight* of the *cut* $(S : \bar{S})$ is $w(S : \bar{S}) = \sum_{i \in S, j \in \bar{S}} w_{i,j}$. The aim is to find the cut of maximum weight.

Introducing integer variables $y_j \in \{-1, 1\}$ for $j \in V$, we can formulate the MAX CUT problem as

$$\begin{array}{lll} \text{IP:} & \text{maximise} & \frac{1}{2}\sum_{i<j} w_{i,j}(1 - y_i y_j) \\ & \text{subject to} & y_j \in \{-1, 1\}, \quad \forall j \in V \end{array} \tag{1}$$

The key insight of Goemans and Williamson is that instead of converting this to an integer linear program and then considering the LP relaxation, it is possible to relax IP directly to a semidefinite program (a special class of convex program, and hence solvable in polynomial time). The basic strategy is to interpret the values ± 1 taken by the variables y_1 as the two points of the 1-dimensional unit sphere centred at the origin, and the product $y_i y_j$ as scalar product of vectors. The relaxation is obtained by allowing y_i to range over the n-dimensional unit sphere. It turns out that the relaxation is a semidefinite program, which is a special kind of a convex program and hence solvable in polynomial time.

The idea of Goemans and Williamson is to solve the SDP and then use a simple (randomised rounding) heuristic to obtain a remarkably good solution to MAX-CUT. The heuristic used is to choose a random hyperplane and partition the vectors y_i according to which side of the hyperplane they lie. It can be shown that the expected weight of the resulting partition is within a factor 0·878 of the optimum; previously, no algorithm with worst-case performance ratio better than the trivial $\frac{1}{2}$ was known.

This is an exciting new idea, and Alan Frieze and I thought it was important to see in what directions it can be generalised. First we considered MAX k-CUT where the aim is to partition V into k subsets. Note that MAX k-CUT has an important interpretation as the search for the ground state in the anti-ferromagnetic k-state Potts model. To attack this problem we needed to be able to handle variables which can take on one of k values, as opposed to just two. Our approach was conceptually simple: modify the integer program IP by constraining the variables y_i to take one of k vector values, corresponding to the vertices of a $(k-1)$-dimensional equilateral simplex. As before, the relaxation is obtained by allowing the variables y_i to range over the entire unit sphere in n-dimensions rather than just the k points determined by the simplex. Our heuristic is (roughly) to take k random unit vectors and associate each vector y_i to the nearest random vector: in this way we obtain a natural partition of the y_i and hence of the vertex set V.

The resulting algorithm, though simple to describe, proved difficult to analyse. However we were able to show that its performance ratio is always better than $1 - k^{-1}$, which is achieved by random partitioning, and was the best previously known. For example, when $k = 3$, the performance ratio is better than 0·800217, and when $k = 10$ better than 0·926642. The perfomance ratio for $k = 2$ is the same as that achieved by Goemans and Williamson, as our heuristic is a generalisation of theirs.

Next we considered the problem MAX BISECTION: partition V into two subsets of equal size (assuming that n is even) so as to maximise the weight of the cut. A random bisection produces an expected performance ratio of $\frac{1}{2}$. We modify the SDP by including a linear constraint that (infomally) guarantees that the average angle between the vectors (variables) y_{ij} is large, i.e., that the vectors are "spread out." Then we show that a modified version of Goemans and Williamson's heuristic produces a cut whose weight is at least 0·65 times the weight of an optimal cut.

Mark Jerrum
June 3, 1994

References

[1] Alan Frieze, Mark Jerrum, *Improved approximation algorithms for MAX k–CUT and MAX BISECTION*, Report ECS–LFCS–94–???, Department of Computer Science, University of Edinburgh, June 1994.

[2] M. X. Goemans and D.P. Williamson, *.878-Approximation algorithms for MAX–CUT and MAX 2SAT*, in *Proceedings of the 26^{th} Annual ACM Symposium on Theory of Computing*, pp 422–431, 1994.

Report on Visit to the University of Edinburgh under EU–US Exploratory Collaborative Activity RAND–REC ECUS030 (D. Randall)

The principal purpose of my visit to Edinburgh was to continue work with Dr. Alistair Sinclair on designing algorithms for combinatorial problems arising in statistical mechanics. We explored directions for extending ideas developed in our recent paper *Testable Algorithms for Self–Avoiding Walks*, which appeared in the ACM–SIAM Symposium on Discrete Algorithms, January 1994. In this paper we present provably correct algorithms for uniformly generating and counting self–avoiding walks of a given length in a d–dimensional Cartesian lattice. The algorithms rely on a single, widely believed conjecture which the algorithm checks during its execution. Thus, we either have reliable outputs, or the program will alert us that the conjecture is false. Either of these outcomes would be instructive to physicists. This type of a testable algorithm is well suited to other problems in statistical mechanics, for which there are many well–established conjectures but very few provably correct algorithms.

Our main research project during this time attempts to generalize these techniques to generating and counting trees of a given length in a d–dimensional lattice (this is joint work with Claire Kenyon, who was visiting Edinburgh from Lyon, France). A lattice tree is any acyclic connected subset of edges in the lattice (containing the origin). Lattice trees are studied as models for branched polymers in dilute solution. Also, it is conjectured that the critical exponents for the number of lattice trees are related to those of lattice animals. The growth rate of the number of lattice trees of size n is expected to have a similar form to the number of self–avoiding walks, which suggests that similar methods can be used. We are currently writing up our progress on this problem. Both of these projects will be part of the doctoral thesis I am now completing under the supervision of Dr. Sinclair.

The techniques that we use can be summarized as follows: Each lattice tree can be uniquely described by a depth–first traversal of the edges, starting at the origin. This traversal defines a walk in the lattice, where each edge in the tree is followed exactely once in each direction. Consider the space of all walks of length at most $2n$ which follow any edge at most once in each direction, and such thet the union of all edges forms a tree. Our goal is to develop a Markov chain on this state space with the following two properties: First, the stationary distribution should be reasonably well concentrated on walks of length $2n$ which end at the origin (corresponding bijectively to lattice trees of length n), and uniform on this set. Second, we require that the Markov chain converge quickly to this stationary distribution so that we can sample in polynomial time. Our Markov chain walks on the set of partial walks via an appropriately parametrized backtracking scheme which allows any walk to be extended or shortened by one edge in any step.

Dana Randall, UC Berkeley
Automn 1993

Report on Visit to the University of California, Berkeley, supported by RAND–REC (M. Santha)

I have visited the International Computer Science Institute and the Computer Science Division of the University of California between April 29 and May 13. The main purpose of my visit was to talk with people.

I spent many time with Umesh Vazirani at the UC Berkeley. He is mostly working in quantum complexity where recently important progress has been made which might have serious consequences for our view about cryptography and complexity in general. The mathematical model of a quantum computer was formulated by Deutsch in 1985. He also gave the description of a universal quantum Turing machine which can be conceived as a first step for the the physical realization of such a computer. Nonetheless this result was not satisfying from a complexity theoretical point of view since this universal machine had an exponential overhead with respect the running time of the simulated machine.

In STOC 1993 Bernstein and Vazirani ("Quantum Complexity Theory") have proved the existence of a universal quantum Turing machine whose overhead is only polynomial. They also gave a certain evidence that quantum Turing machines might be more powerful than classical probabilistic Turing machines. In particular they have shown that with respect some oracle there is a language which can be accepted in polynomial time by a quantum TM but can not be accepted by a bounded error probabilistic TM in time $n^{o(\log n)}$. Therefore with respect to this oracle quantum polynomial time is not included into the class BPP.

In a very recent development Shor ("Algorithms for Quantum Computation") has shown that factorization and the computation of the discrete logarithm function can be done in random polynomial time on a quantum computer with one-sided error. These number theoretical problems have been extensively studied and no polynomial time probabilistic algorithm is known for them. Since the apparent computational difficulty of these problems plays a crucial role in numerous branches of modern, complexity based cryptography, the existence of a fast (quantum) algorithm for them poses a threat for the security of the concerned cryptosystems. The threat is real even if clearly no actual quantum computer is in view in the short future.

Also in a recent paper, Bennett, Bernstein, Brassard and Vazirani ("What is Feasible on a Quantum Computer") show some possible limitation of quantum computation. They prove that with probability 1, NP with respect to a random oracle is not included into bounded-error quantum polynomial time with the same oracle. They also show that on the universal machine any bounded-error quantum polynomial time algorithm can be simulated with only logarithmical bit-precision in function of the running time of the simulated machine.

In ICSI I mostly worked with Jeff Edmonds on a nice combinatorial problem of his. The problem came up from a paper of Edmonds and Impagliazzo ("Towards Time-Space Lower Bounds on Branching Programs") about oblivious branching programs. It can be the most easily described as a game. Let $f(x, y)$ be some Boolean function, where x and y are Boolean strings of the same (constant) length. The input to the game is $2n$ strings $x_1, \ldots, x_n$ and $y_1, \ldots y_n$. The game is played by several collaborating players who share a blackboard. Initially the blackboard is empty. Each player takes part in exactly one round of the game. In that round the player looks at the blackboard, and in function of it contents he choses to see for $i = 1, \ldots, n$ exactly one of the strings x_i, y_i. Afterwards he writes a single bit on the blackboard and he quits the game. The game is over, when a player can decide if for all i, $f(x_i, y_i) = 1$. The complexity $c(f)$ of the game is the smallest number of players such that with some common strategy such a decision can be made.

Edmonds and Impagliazzo have shown that an $\Omega(n^c)$ lower bound for $c(f)$ with some $0 < c < 1$ would imply an $\Omega(n^{1+c'})$ lower bound on the space-time tradeoff for oblivious branching programs for f, with $0 < c' < c$. This would be significantly higher than the currently known best lower bounds. Edmonds and Impagliazzio have shown that if x and y are just 1-bit strings and by definition $g(x, y) = 1$ when $x = y$, then $c(g) = O(n^{1/2})$. They conjectured that actually $c(g) = \Theta(n^{1/2})$. This conjecture was disproved by Pudlak and Sgall ("An Upper Bound for a Communication Game Related to Space-Time Trade-offs") who have shown that $c(g) = O(n^{2/5}(\log n)^{3/5})$.

The question of a polynomial lower bound on $c(f)$ for any f is wide open. I was only able to show (which is a simple remark) that for any f, $c(f) = O(n^{1/2})$. In case of the particular g above, even an $\Omega((\log n)^2)$ lower bound seems to be very hard. For this function a better upper bound would also be interesting.

I also spoke extensively with Manuel Blum, Mike Luby and Alistair Sinclair. I gave a talk in ICSI about "Verifying the determinant in parallel".

Miklos Santha

A Report on the Workshop "Probability and Algorithms"supported by RAND-REC (D.J.A. Welsh)

The workshop formed part of a year long programme on *Emerging Applications of Probability* organized by the auspices of the American Mathematical Society and SIAM. It was preceeded by a tutorial for young workers (typically post–docs) on *Probability and Optimization.* This was a set of lectures aimed at providing perspective and background relevant to workshop 1, *Probability and Algorithms*, and workshop 2, *Finite Markov Chain Renaissance.* The background required was merely a good graduate course in probability, though some modest exposure to the theory of algorithms and linear programming provided motivation.

Turning now to the workshop itself: the topics addressed included derandomization, analysis of rapid factorization techniques, uses of probability in problems of Euclidian combinatorial optimizations, finite Vapnik Chevonenkis classes in problems of computational geometry, the "bounded difference method", and random coloring algorithms.

The main thrust of the tutorial was a set of lectures by Steele on subadditive processes and functionals, Aldous on random walks and their applications, Spencer on Janson's inequality an Azuma's Inequality, and Diaconis who gave a survey of the current state of the art in the world of rapidly mixing Markov Chains.

The main talks of the workshop are described below.

Dominic J.A. Welsh, Institute for Mathematics and its Applications
Minneapolis, September 1993

Abstracts of Talks

- **Richard M. Karp (UC Berkeley):**
  Selection in the presence of noise: The design of playoff systems

  In every sport, playoffs and tournaments are used to select the best among a set of competing players or teams. In this talk we consider the design of such systems. We assume that there are n players, and that one of them, denoted Player 1, is the best in the following sense: whenever Player 1 plays a game, he wins with a probability that depends on his opponent, but is always greater than 1/2. The identities of the players are initially unknown, and our goal is to determine Player 1 with high confidence in a minimum expected number of rounds where, in each round, each player who has not yet been eliminated participates in exactly one game. A secondary goal is to minimize the expected number of games played. We consider three models, which differ in their assumptions about what

happens in games that do not involve Player 1. In the adversary model the outcomes are determined by a malicious adversary. In the strong transitivity and discriminating models more restrictive assumptions are made. We also consider two versions of each model: one in which Player 1's win probabilities are known to the algorithm, and the other in which they are unknown. For each of the six models an upper bound and a lower level bound are derived on the expected number of rounds required. In the course of deriving these bounds we provide insight into the advantages and disadvantages of some commonly used systems for conducting elimination tournaments.
(Joint work with Micah Adler, Peter Gemmell, Mor Harchol and Claire Kenyon)

- **Carl Pommerance (University of Georgia):**
  The role of randomness in primality testing

  It has been known since the 1970's that there are simple probabilistic algorithms that find proofs of compositeness for composite inputs in expected polynomial time. More recently, Adleman and Huang gave a difficult probabilistic algorithm that finds proofs of primality for prime inputs in expected polynomial time. What of simpler primality tests? One of the most basic, known as the $n-1$ test, can find a proof that n is prime in expected polynomial time, if a large completely factored divisor of $n-1$ is given as part of the input. It was known that this test could be made deterministic on assumption of the Generalized Riemann Hypothesis (GRH). In this talk I shall describe some recent joint results with S. Konyagin which show how the $n-1$ test may be made deterministic without the assumption of the GRH. A corollary of our results is that the number of primes uo to x which can be proved prime in deterministic polynomial time exceeds $x^{1-\epsilon}$.

- **J. Spencer (Courant Institute/IMA):**
  From Erdos to algorithms

  The lecture was intended to illustrate the gap existing between the purely existence proofs based on clever use of the probabilistic method and the algorithmic search for a solution. This was colorfully illustrated by three solutions to the problem of the Tenure Game, together with other examples of derandomization technique.

- **Noga Alon (Tel Aviv University/Institute for Advanced Study, Princeton):**
  Expanders, nuts and bolts

  We describe various applications of pseudo random graphs to the derandomization of certain randomized sorting algorithms focusing on recent joint work with Blum, Fiat, Kannan, Naor and Ostrovsky concerning a new sorting problem.

- **Andrew Odlyzko (AT&T Bell Laboratories):**
  Search for the maximum of a random walk

  Let $X_1, X_2, \ldots$ be independent and identically distributed with Prob(X_j $= 1$) = Prob($X_j = -1$) = $^1/_2$, and let $S_k = X_1 + X_2 + \ldots + X_k$. Thus S_k is the position of a symmetric random walk on the line after k steps. It is shown that any algorithm that determines $\max\{S_0, \ldots, S_n\}$ with certainty must examine at least $c_1 n^{1/2}$ of the S_k on average for a certain constant $c_1 > 0$, if all random walks with n steps are considered equally likely, there is also an algorithm that on average examinesonly $c_s, n^{1/2}$ of the S_k to determine their maximum for another constant c_2.

- **Eli Shamir (Hebrew University/IMA):**
  The bounded difference method in learning algorithms and thresholds

  Our framework is Approximate Learning of concept classes by random queries, or related Rangespaces problems in Computational Geometry – usually for families of a finite VC dimension.

  1. Filtering random streams: A good filter should block (ultimately most) examples, allowing through examples which as queries have conditional expected "information gain" above a certain positive B. Submartingale tail estimates are handy in proving that the "prediction error" then becomes exponentially small in the number of filtered queries.
  2. Thresholds for ϵ nets: We study conditions under which the confidence level in getting ϵ net by a random sample exhibits a sharp threshold in the sample size.

- **M. Steele (University of Pennsylvania/IMA):**
  Subsequence optimizations – Random and pseudo–random

  A problem which typified the sort considered was that of finding the length of the longest increasing subsequence in a random sequence of given length. Techniques developed for working on this problem extend to problems more relevant to combinatorial Optimization such as Travelling Salesman in the unit hypercube. Basic techniques illustrated included the theory of subadditive stochastic processes and subadditive functionals.

- **John N. Tsitsiklis (MIT):**
  On the average communication complexity of distributed algorithms

  We study the communication complexity of asynchronous algorithms, in which message receptions can trigger further computation and transmission of new messages. Such algorithms can generate excessively many messages in the worst case. Nevertheless, we show that, under certain probabilistic assumptions, and under a very general model of distributed computation, the expected number of generated messages is only $\mathcal{O}(nT)$, where n is the number of processors and T is the running time. We conclude that (under our model) any asynchronous algorithm will also have good communication complexity, on the average.

- **M. Talagrand (Ohio State University):**
  Isoperimetric methods and optimization problems

  The lecture by Michel Talagrand aroused a considerable degree of interest particularly among the probabilists, in that his work gives a series of inequalities which considerably strengthen the very useful inequality of Azuma. They are fairly technical to explain but I do have a "partial" preprint and "translation" of one of the simplest forms of the inequality. I will be happy to send copies to anyone who asks.

- **Dominic J.A. Welsh (University of Oxford):**
  Randomized approximaton schemes for Tutte–Gröthendieck invariants

  Consider the following very simple counting problems associated with a graph G.

  (i) What is the number of connected subgraphs of G?
  (ii) How many subgraphs of G are forests?
  (iii) How many acyclic orientations has G?

  Each of these is a special case of the general problem of evaluating the Tutte polynomial of a graph (or matroid) at a particular point of the (x, y)–plane – in other words is a Tutte–Gröthendieck invariant (see [2]). Other invariants include:

  (iv) the chromatic and flow polynomials of a graph;
  (v) the partition function of a Q–state Potts model;
  (vi) the Jones polynomial of an alternating link;
  (vii) the weight enumerator of a linear code over $GF(q)$.

  It has been shown in [1] that apart from a few special points and 2 special hyperbolae, the exact evaluation of any such invariant is $\#P$–hard even for the very restricted class of planar bipartite graphs. However the question of which points have a fully polynomial randomized approximation scheme is wide open. I shall discuss this problem and give a survey of what is currently known.

- **Moni Naor (Weizmann Institute):**
  Derandomization, witnesses for Boolean matrix multiplication and construction of perfect hash functions

  Small sample spaces with almost independent random variables are applied to design efficient sequential deterministic algorithms for two problems. The first algorithm, motivated by the attempt to design efficient

algorithms for the All Pairs Shortest Path problem using fast matrix multiplication, solves the problem of computing *witnesses* for the Boolean product of two matrices. That is, if A and B are two n by n matrices, and $C = AB$ is their Boolean product, the algorithm finds for every entry $C_{ij} = 1$ a witness, an index k so that $A_{ik} = B_{kj} = 1$. Its running time exceeds that of computing the product of two n by n matrices with small integer entries by a polylogarithmic factor. The second algorithm is a nearly linear time deterministic procedure for constructing a perfect hash function for a given n–subset of $\{1, \ldots, m\}$.
(Joint work with Noga Alon)

- **Dimitris Berstimas (MIT):**
  Linear programming relaxations, approximation algorithms and randomization

  In recent years progress has been made in our finer understanding of *NP*. In terms of the degree of approximability, *NP*–hard problems can be divided into four broad categories: Problems that can be approximated in polynomial time

  1. within any constant (for example the Knapsack Problem),
  2. up to a finite constant factor (for example the Travelling Salesman problem),
  3. up to a logarithmic factor (for example the Set Covering Problem), and
  4. within a sublinear but superlogarithmic factor (for example the Coloring Problem).

  On the negative side, however, there is a lack of unification of the methods used to design the approximation algorithm. We show that the use of randomized rounding of linear programming relaxations of discrete optimization problems, but with nonlinear rounding functions leads to a unified way of approximating *NP*–hard problems matching the best known performance guarantees. We illustrate our methods using several examples: the Set Covering problem, facility location problems, *MAXSAT*, network connectivity problems and the minimum approximation algorithms through the use of randomization.

- **Alan Frieze (Carnegie Mellon University):**
  Greedy algorithms from a probabilistic point of view

  I will review some results on the average case of Greedy algorithms and on a randomized Greedy algorithm for matchings.

□