RESEARCH REPORT

INSTITUT FÜR INFORMATIK DER UNIVERSITÄT BONN
Report No. 8590-CS

Bonn Workshop On Randomized Algorithms(RAND)

Bonn, March 29 - 31, 1993

HRSG.: MAREK KARPINSKI
HANS-JÜRGEN PRÖMEL

Institut für Informatik der Universität Bonn
*

PREFACE

The Workshop was organized by the ESPRIT BR Workshop Group on Randomized Algorithms (RAND), and the Department of Computer Science of the University of Bonn. It was concerned with the newest development in the design of efficient and pseudo-randomized algorithms, approximation algorithms, circuit design, probabilistic methods and the constructions of small sampling spaces as well as with the foundations of complexity theory of randomized computation.

We are thankful to Lida Han and Kirsten Steinhauer for the superior organisation of this meeting.

April 1993

Marek Karpinski
Hans-Jürgen Prömel

Bonn Workshop On Randomized Algorithms (RAND)

Bonn, March 29 – 31, 1993

PROGRAM

Hörsaal 1, Römerstr. 164, 5300 Bonn 1

Monday, March 29, 1993

09:00 – 09:10	Opening
Chair:	Marek Karpinski
09:15 – 10:00	Eli Shamir *Randomized Algorithms in Computational Learning Theory*
10:15 – 11:00	Angelika Steger *Testing Hereditary Properties Efficiently*
11:00 – 14:00	coffee and lunch break (RAND WG Technical Meeting)
Chair:	Hans-Jürgen Prömel
14:00 – 14:45	Joachim von zur Gathen *Probabilistic Methods in Finite Fields*
15:00 – 15:45	Seffi Naor *On Removing Randomness from a Parallel Algorithm for Minimum Cuts*
15:45 – 16:15	coffee break
16:15 – 17:00	Oscar Garrido *A Simple Randomized Parallel Algorithm for Maximal f-Matchings*

Tuesday, March 30, 1993

Chair:	Michael Luby
09:00 – 09:45	Christian Schindelhauer *Precise Average Case Complexity*
10:00 – 10:45	Ingo Althöfer *Derandomization: Upper and Lower Bounds*
10:45 – 11:15	coffee break
11:15 – 12:00	W. Fernandez de la Vega *The Threshold for the Almost Sure Satisfiability of a Random Set of 3-Clauses*
12:00 – 14:00	lunch break
Chair:	Miklos Santha
14:00 – 14:45	Michael Luby *A Parallel Approximation Algorithm for Positive Linear Programming*
15:00 – 15:45	Andrzej Lingas *Linear-Time Randomized Algorithms for Voronoi Diagrams of Simple Polygons*
15:45 – 16:15	coffee break
16:15 – 17:00	Thomas Lickteig *On Randomized Test Complexity*
20:00	Informal & Open Problem Session

Wednesday, March 31, 1993

Chair:	Joachim von zur Gathen
09:00 – 09:45	Miklos Santha *On the Interactive Complexity of Graph Enumeration Problems*
10:00 – 10:45	Rutger Verbeek *On Randomized Versus Deterministic Computation*
10:45 – 11:15	coffee break
11:15 – 12:00	Boban Veličković *Approximation of General Independent Distributions*

Information, Prediction and Query by Committee

Eli Shamir
Jerusalem

A highly desirable goal in approximate learning of concepts by queries is to drive the "prediction error" [exponentially] fast to 0. We show this is achieved if the "expected information-gain" by a query is bounded from 0. "Query by committee" randomized algorithms provide filters which from a random stream of inputs pick up the informative queries. The typical situation we discuss are "generalized perceptrons", i.e. concepts defined by thresholds of smooth functions.

Testing Hereditary Properties efficiently

Jens Gutstedt
Berlin

Angelika Steger
Bonn

The aim of this talk is to develop fast algorithms for hereditary properties that are, while not fast in the worst case, at least fast on average.

The key observation for such algorithms is that if the probability that a fixed obstruction H op property $\mathcal{E}$ is not contained in the input is low then most possible inputs don't have the property $\mathcal{E}$ and this can be verified by testing for the obstruction H. In this talk we will describe the three parts of such an approach, namely
(1) to show that a fixed obstruction H of a property $\mathcal{E}$ occurs with high probability,
(2) to develop an algorithm that is fast on average and test for a given obstruction H,
(3) to design an exact algorithm for $\mathcal{E}$ whose running time is sufficiently small compared to the probability that the obstruction H does not occur.

We will do that in a general setting, but we will also consider some special examples of combinatorial structures. Among these will be several calsses of graphs equipped with three different relations, namely the induced and weak subgraph relation, and the graph minor relation.

Probabilistic Methods in Finite Fields

Joachim von zur Gathen
Toronto & Zürich

A polynomial $f \in \mathsf{F}_q[x]$ over a finite field F_q is a *permutation polynomial* if and only if the associated mapping $\mathsf{F}_q \to \mathsf{F}_q$ is bijective. We present a probabilistic test for this property using essentially $O(n \log q)$ operations in F_q, where $n = \deg f$; this solves a problem posed by Lidl & Mullen.

Furthermore, we give approximation schemes for the size of the image of a polynomial or rational function, and the size of an algebraic curve; these results are joint work with MA and KARPINSKI & SHPARLINSKI, respectively.

On Removing Randomness from a Parallel Algorithm for Minimum Cuts

Michael Luby
Berkeley

Seffi (Joseph) Naor
Haifa

M. Naor
Haifa

The minimum cut problem is the following: partition the vertices of a graph into two disjoint sets so as to minimize the number of edges in the cut, i.e., edges adjacent to vertices that are in different sets. The graph may be weighted, in which case we want to minimize the weight of the edges in the cut. This problem has received much attention in the literature in the last 40 years. It is a fundamental problem in combinatorial optimization and has numerous applications, e.g., network design and reliability, sequencing and scheduling, location theory, partitioning problems, and heuristics for solving integer programming problems. The parallel complexity, however, remained unresolved. Recently, Karger for computing the minimum cut in a graph. This placed the problem in the complexity class RNC.

We show that a similar algorithm can be implemented using only $O(\log^2 n)$ random bits. We also show that this result holds for computing minimum weight k-cuts, where k is fixed. We view our algorithm as a step towards obtaining a deterministic algorithm for the problem. Alternatively, one can view random bits as a resource (such as time and space), to be used as sparingly as possible, and our result reduces the use of this resource over the algorithm suggested by Karger. Reducing the number of random bits needed in computation is a line that has been explored by many researchers in recent years.

A Simple Randomized Parallel Algorithm for Maximal f-Matchings

Oscar Garrido
Lund

Stefan Jarominek
Warsaw

Andrzej Lingas
Lund

Wojciech Rytter
Warsaw

We show how to extend the RNC-algorithm for maximal matchings due to Israeli-Itai to compute maximal (with respect to set of edges inclusion) f-matchings. Our algorithm works in $\mathcal{O}(\log^2 n)$ time on an arbitrary CRCW PRAM with a linear number of processors. The algorithm can be used also for multigraphs and then it preserves its complexity.

Precise Average Case Complexity

Rüdiger Reischuk
Darmstadt

Christian Schindelhauer
Darmstadt

A new definition is given for the average growth of a function $f : \Sigma^* \rightarrow \mathbb{N}$ with respect to a probability measure μ on Σ^*. This allows us to define meaningful average case distributional complexity classes for arbitrary time bounds (previously, one could only distinguish between polynomial and superpolynomial growth). It is shown that basically only the ranking of the inputs by decreasing probabilities are of importance.

To compare the average and worst case complexity of problems we study average case complexity classes defined by a time bound and a bound on the complexity of possible distributions. Here, the complexity is measured by the time to compute the rank functions of the distributions. We obtain tight and optimal separation results between these average case classes. Also the worst case classes can be embedded into this hierarchy. They are shown to be identical to average case classes with respect to distributions of exponential complexity.

These ideas are finally applied to study the average case complexity of problems in $\mathcal{N}P$. A reduction between distributional problems is defined for this new approach. We study the average case complexity class $\mathcal{A}\mathrm{v}\mathcal{P}$ consisting of those problems that can be solved by DTMs on the average in polynomial time for all distributions with efficiently computable rank function. Fast algorithms are known for some $\mathcal{N}P$–complete problems under very simple distributions. For languages in $\mathcal{N}P$ we consider the maximal allowable complexity of distributions such that the problem can still be solved efficiently by a DTM, at least on the average. As an example we can show that either the satisfiability problem remains hard, even for simple distributions, or $\mathcal{N}P$ is contained in $\mathcal{A}\mathrm{v}\mathcal{P}$, that means every problem in $\mathcal{N}P$ can be solved efficiently on the average for arbitrary not too complex distributions.

Derandomization: Upper and Lower Bounds

Ingo Althöfer
Bielefeld

A randomized strategy or a convex combination may be represented by a probability vector $p = (p_1, \ldots, p_m)$. p is called sparse if it has only few positive entries.

We present the following

Approximation Lemma:

Let $A = (a_{ij})$ be an $m \times n$-matrix over the real numbers with $0 \leq a_{ij} \leq 1$ for $1 \leq i \leq m$, $1 \leq j \leq n$. Let $p = (p_1, \ldots, p_m)$ be a probabiblity vector, i.e., $0 \leq p_i$ for i and $\sum_{i=1}^{n} p_i = 1$, and $\epsilon > 0$ any positive constant. Then there exists another probability vector $q = (q_1, \ldots, q_m)$ with at most $k = \lceil \frac{\log 2n}{2\epsilon^2} \rceil$ many positive coordinates q_i such that

$$\left| \sum_{i=1}^{n} p_i a_{ij} - \sum_{i=1}^{m} q_i a_{ij} \right| \leq \epsilon \quad \text{for all} \quad j = 1, \ldots, n.$$

More precisely, the probability vector q can be chosen such that $q_i = \frac{k_i}{k}$ with natural numbers k_i for all $i = 1, \ldots, m$.

The bound k is asymptotically optimal up to the multiplicative constant $4 \log 2 \approx 2.77$.

The Approximation Lemma is applied to matrix games, certain linear programs, and computer chess.

On the Threshold for the Almost Sure Satisfiability of a Random Set of 3-clauses

W. Fernandez de la Vega
Paris

A. El Mafthoui
Paris

Let S be a set of m clauses each containing 3 literals chosen at random in a set $\{p_1, \neg p_1, ..., p_n, \neg p_n\}$ of n propositional variables and their negations. Let c denote the biggest number such that if m and n tend to infinity with $\frac{m}{n} > c$, then the probability that the set S is satisfiable tends to 1 as n tends to infinity. Frieze and Suen have shown recently that c exceeds 3 and it is known that $c \leq \log_{8/7} 2 = 5.19....$ We will present some methods and results concerning better upper bounds for c.

A Parallel Approximation Algorithm for Positive Linear Programming

Michael Luby
Berkeley

Noam Nisan
Jerusalem

We introduce a fast parallel approximation algorithm for the positive linear programming optimization problem, i.e., the special case of the linear programming optimization problem where the input constraint matrix and constraint vector constist entirely of positive entries. The algorithm is elementary, and has a simple parallel implementation that runs in polylog time using a linear number of processors.

Linear-Time Randomized Algorithms for Voronoi Diagrams of Simple Polygons

Andrzej Lingas
Lund

Rolf Klein
Hagen

We present linear-time generalizations of Chew's randomized algorithm for the Voronoi diagram of a convex polygon to include the convex hull of a special polygon in 3D, the Voronoi diagram of a monotone polygon and the bounded Voronoi diagram of a simple polygon.

On Randomized Test Complexity

Peter Bürgisser
Bonn

Marek Karpinski
Bonn

Thomas Lickteig
Bonn

We investigate the impact of randomization on the complexity of deciding membership in a (semi-)algebraic subset $X \subset \mathbf{R}^m$. Examples are exhibited where allowing for a certain error probability ϵ in the answer of the algorithms the complexity of decision problems decreases. A randomized $(\Omega^k, \{=, \leq\})$-decision tree ($k \subseteq \mathbf{R}$ a subfield) over m will be defined as a pair (T, μ) where μ a probability measure on some $\mathbf{R}^n$ and T is a $(\Omega^k, \{=, \leq\})$-decision tree over $m + n$. We prove a general lower bound on the average decision complexity for testing membership in an irreducible algebraic subset $X \subset \mathbf{R}^m$ and apply it to k-generic complete intersection of polynomials of the same degree, extending results of Lickteig, Bürgisser and Lickteig and Bürgisser, Lickteig and Shub. We also give applications to nongeneric cases, such as graphs of elementary symmetric functions, $\mathrm{SL}(m, \mathbf{R})$, and determinant varieties, extending results of Lickteig

On the Interactive Complexity of Graph Enumeration Problems

J. Diaz
Paris

M. de Rougemont
Paris

Miklos Santha
Paris

We consider three $\#P$–complete enumeration problems on graphs : $s - t$ PATHS, $s - t$ CONNECTEDNESS and $s - t$ RELIABILITY, and give IP protocols for them. If $IP(f(n))$ is the class of languages whose interactive complexity is $O(f(n))$, that is the set of languages which can be accepted by an interactive proof system with $O(f(n))$ number of rounds, then our protocols imply that the interactive complexity of these problems is significantly smaller than what one could get by using generic reductions via Cook's Theorem. Indeed, we show that $s - t$ PATH $\in IP(n)$, $s - t$ CONNECTEDNESS $\in IP(n^2)$, and $s - t$ RELIABILITY $\in IP(n^2)$.

On Randomized Versus Deterministic Computation

Marek Karpinski
Bonn

Rutger Verbeek
Hagen

In contrast to deterministic or nondeterministic computation, it is a fundamental open problem in randomized computation how to separate different randomized time classes (at this point we do not even know how to separate linear randomized time from $O(n^{\log n})$ randomized time) or

how to compare them relative to corresponding deterministic time classes. In another words we are far from understanding the power of *random coin tosses* in the computation, and the possible ways of simulating them deterministically.

In this paper we study the relative power of linear and polynomial randomized time compared with exponential deterministic time. Surprisingly, we are able to construct an oracle A such that exponential time (with or without the oracle A) is simulated by linear time Las Vegas algorithms using the oracle A. We are also able to prove, for the first time, that in some situations the randomized reductions are exponentially more powerful than deterministic ones (cf. [Adleman, Manders, 1977]).

Furthermore, a set B is constructed such that Monte Carlo polynomial time (BPP) under the oracle B is exponentially more powerful than deterministic time with nondeterministic oracles. This strengthens considerably a result of Stockmeyer about the polynomial time hierarchy that for some decidable oracle B, $\mathrm{BPP}^B \not\subseteq \Delta_2\mathrm{P}^B$. Under our oracle BPP^B is exponentially more powerful than $\Delta_2\mathrm{P}^B$, and B does not add any power to Δ_2EXPTIME.

Approximation of general independent distributions

Guy Even
Haifa

Oded Goldreich
Haifa

Micheal Luby
Berkeley

Noam Nisan
Jerusalem

Boban Veličković
Berkeley

In this talk we discuss the problem of efficiently constructing small sample space probability distributions on n Boolean variables which approximate a given independent but not necessarily uniform distribution on n Boolean variables. This problem is frequently encountered in practice, for instance, in the network reliability problem.

We establish an intimate connection of this question with the problem of construction small discrepancy sets in I^n, the n-dimensional unit cube. This problem has been studied extensively in numerical analysis and essentially optimal constructions have been given in case the dimension is constant.

We present several examples of efficiently constructible small discrepancy sets in I^n of size $\exp(O(\log(n/\epsilon)^2))$, where ϵ is the error parameter.

Informal and Open Problem Session

On extending Ben-Or's lower bound to randomized decision trees

Peter Bürgisser
Bonn

We prove a generalization of Ben-Or's lower bound holding for randomized algebraic decision trees with one-sided error. As a consequence we obtain a m^2 lower bound for the m-dimensional knapsack problem in this model. We note that the distinction between one-sided and two-sided error randomized decision trees is crucial here.

Average Search Time in Random Majority Trees

Ingo Althöfer
Bielefeld

We consider a 3-regular rooted tree T with node values $v(x) \in \{0,1\}$. (T,v) is called a majority tree if

$$v(x) = majority(v(y_1), v(y_2), v(y_3))$$

for every non-terminal node x and its 3 direct successors y_1, y_2, y_3.

At the beginning owe know the tree T, but not its v-values. In every elementary unit of time we are allowed to ask one leaf for its v-value. The answer will be 0 or 1, each with probability $\frac{1}{2}$, and independently of all answers before.

Our task is to fix the root value as soon as possible.

For trees with uniform depth t the following bounds are known for the average search time $a(t)$:

$$\left(\frac{9}{4}\right)^t \leq a(t) \leq (2.470\ldots)^t,$$

where the left inequality holds for all $t \geq 0$, and the right inequality holds if t is a multiple of 3.

Find better bounds!

Central Elements, Order Ideals and Searching/Sorting in Posets

Devdatt Dubhashi
Saarbrücken

Let $\mathbf{P} := (P, \leq)$ be a poset, and $f : \mathbf{P} \to \mathbf{R}$, an order-preserving map.

Search Problem: Given $\alpha \in \mathbf{R}$, determine if there exists an $a \in P, f(a) = \alpha$, (employing comparisons of the form $f(x)?\alpha$, $? \in \{\leq, \geq\}, x \in P$)

Sorting Problem: Assume P is known, but $\leq$ is not. Determine the poset $\mathbf{P} = (P, \leq)$ (employing comparisons of the form $f(x) \leq f(y)$ for $x, y \in P$).

How many comparisons are needed for these problems (clearly generalizations of the usual searching and sorting problems)?

Let $N(\mathbf{P})$ denote the number of order-ideals in $\mathbf{P}$. Then, there is the following *information-theoretic* lower bound on the search problem: $\lceil \log N(P) \rceil$. One can in fact achieve this performance, if one can find so called central elements in $\mathbf{P}$.

Definition An element $x \in P$ is a *δ-central element* $(0 \leq \delta \leq 1)$ if

$$\delta \leq \frac{N(x)}{N(P)} \leq 1 - \delta$$

(where $N(x)$ denotes the number of ideals containing x).

Briefly, to solve the search problem, one always compares α to a central element and then passes to the appropriate sub-lattice. To solve the sorting problem, one embeds this strategy into an insertion sort, to achieve an upper bound of $n \log N(P)$, where $n = |P|$. This is optimal [Mehl].

Question: How do we find central elements in lattices?

Linial and Saks [LS 85] prove the remarkable:

Theorem For any finite $\mathbf{P} = (P, \leq)$, there is an element $x \in P$ s.t.

$$\delta_0 \leq N(x) \leq 1 - \delta_0$$

where $\delta_0 = (3 - \log_2 5)/4 \approx 0.17$.

However, the proof is non-constructive, arising out of a probabilistic argument. For some special classes of posets (interval orders, series parallel posets, trees, 2-dimensional posets) constructive results are known.

There is a close relationship between the problem of finding central elements, and counting ideals in posets. Of course, if one can count ideals (even approximately), one can find central elements. Conversely, it is known that the existence of a polynomial time algorithm to find central elements implies that the number of ideals can be approximated (to within a factor of $(1 + \epsilon)$, for any ϵ). However, finding $N(\mathbf{P})$ exactly is $\#P$-complete for general posets.

Open Question: Give randomized algorithms for finding central elements in posets (or at least for special classes of posets).

Open Question: Give randomized approximation schemes for counting ideals in posets. Can the theory of rapidly-mixing Markov chains be applied?

Open Question: Does a randomized algorithm for counting ideals (approximately) yield a randomized algorithm for central elements (and vice versa)?

[LS 85] N. Linial and M.Saks: "Every Poset has a Central Element", J. Comb. Theory A, 40, pp.195–210, 1985.

[Mehl] K.Mehlhorn: "Private Communication".

On a Computing of Cellular Algebra of a Graph

Ilia N. Ponomarenko
St. Petersburg

Matrix Algebra $\mathcal{A}$ over $\mathbb{C}$ is called *cellular algebra* if it satisfies the following conditions:
(1) $\mathcal{A}$ contains the unit matrix and the matrix all of whos entries are ones;
(2) if $A \in \mathcal{A}$ then $A^T \in \mathcal{A}$;
(3) $A \circ B \in \mathcal{A}$ for $A, B \in \mathcal{A}$ where $A \circ B$ denotes the Hadamard (componentwise) product of A and B.

It follows from (3) that a cellular algebra contains a linear basis consisting of $(0, 1)$ matrices. Such basis is called *standard* one.

Let Γ be a finite graph and A be the adjacency matrix of Γ. Let $\mathcal{A}(\Gamma)$ be a smallest (by inclusion) cellular algebra wicht contains A. Then we say $\mathcal{A}(\Gamma)$ is a algebra of graph Γ.

It was shown in [Weis] that there is a close relationship between the automorphism group of a graph and properties of the algebra. In particular, there was described an approach to the Graph Isomorphism Problem using cellular algebras. This approach was developed in papers of the author where a polynomial-time reduction of Graph Isomorphism Problem for classes of directed path and circular arc graphs to the problem of computing the stadnard basis of the algebra of a graph was described. We will use the notation CCA for the problem of computing the standard basis of the algebra of a graph.

A polynomial-time procedure for CCA was proposed in [Weis]. The time analysis of the procedure (see [Pon]) gives a bound $O(n^{\omega+2} \log n)$ for an n-vertex graph, where $O(n^{\omega})$ is the complexity of $n \times n$ matrix multiplication. We hape that the above bound can be substantially improved. So, our problems are:

Problem 1: Is it possible to solve CCA in time $O(n^{\omega+1})$ for an n-vertex graph?

Problem 2: What is the NC-complexity of CCA?

[Pon] I. N. Ponomarenko, "On computation complexity problems concerning relations algebras" (appears in Zapiski POMI in 1993).

[Weis] B. Ju. Weisfeiler, editor. "On Construction and Identification of Graphs". Springer Lecture Notes, 558, 1976.