LABORATORY FOR
COMPUTER SCIENCE

MASSACHUSETTS
INSTITUTE OF
TECHNOLOGY

MIT/LCS/TM-313

THEORY OF COMPUTATION GROUP RESEARCH SUMMARY JUNE 1985 - JULY 1986

Theory of Computation Group

August 1986

545 TECHNOLOGY SQUARE, CAMBRIDGE, MASSACHUSETTS 02139

MIT/LCS/TM-313

THEORY OF COMPUTATION GROUP RESEARCH SUMMARY JUNE 1985 - JULY 1986

THEORY OF COMPUTATION GROUP

AUGUST 1986

Theory of Computation Group
Research Summary
June, 1985 -- July, 1986

M.I.T. Laboratory for Computer Science

13 August 1986

KEYWORDS: algorithms, complexity, cryptology, distributed, logic of programs, pseudorandom, VLSI routing, semantics

Table of Contents

1. Personnel

Academic Staff

B. Awerbuch
L. Heath
N. Lynch
R. Rivest (Group Leader)

P. Elias
F.T. Leighton
A. Meyer
D. Shmoys

S. Goldwasser
C. Leiserson
S. Micali
M. Sipser

Associates

B. Chor

O. Goldreich

B. Trakhtenbrot

Graduate Students

W. Aiello
B. Bloom
D. Britsimas
T. Cormen
A. Goldberg
R. Hirschfeld
J. Kilian
R. Koch
S. Malitz
J. Park
M. Reinhold
J. Siskind
L. Yedwab

D. Barrington
R. Boppana
T. Bui
P. Feldman
S. Goldman
A. Ishii
S. Kipnis
B. Maggs
S. Mentzer
C. Phillips
P. Rogaway
R. Sloan

B. Berger
V. Breazu-Tannen
J. Buss
R. Greenberg
J. Hastad
B. Kaliski
P. Klein
F.M. Maley
M. Newman
S. Rao
A. Sherman
S.-M. Wu

Undergraduate Students

T. Heigham

T. Leung

J. Hinsdale

B. Rogoff

C. Kaklamanis

Support Staff

A. Benford

I. Radzihovsky

B. Hubbard

R. Spenser

L. Melcher

Visitors

P. Gacs
L. Levin
J. Reif

S. Homer
R. Kemmerer
A. Shamir

E. Lander
Y. Moses
M. Wand

2. Research Overview

The Theory group continues to be vital and prolific. Principal research areas are:

- algorithms: combinatorial, geometric, graph-theoretic, number theoretic
- cryptology
- computational complexity
- distributed computation: algorithms and semantics
- randomness in computation
- semantics and logic of programs
- VLSI design theory.

Group members were responsible for over one hundred publications during the past year, as well as dozens of public lectures around the world. The reader may review the *individual reports* and the *annotated bibliography and lecture list* below for further descriptions of the year's activities.

The following major research accomplishments merit highlighting:

1. Goldreich and Micali's proof that all problem's in NP have "zero-knowledge proofs". As a consequence, any cryptographic protocol that is correct with respect to a very weak adversary can be automatically transformed into an equivalent protocol correct in the most adversarial scenario [52],
2. Goldwasser and Kilian's efficient probabilistic algorithm for generating certified primes numbers [56],
3. Leighton's minimax grid-matching algorithm and its application to average case analysis of algorithms for bin packing, dynamic allocation and wafer-scale integration [93], [43],
4. Meyer *et al.*'s analysis of polymorphic types in programming and the negative consequences of the *'type' is a type* assumption [28], [107], [101].

3. Individual Reports

3.1 Faculty and Research Associates

3.1.1 Baruch Awerbuch

Awerbuch has been working on designing efficient and reliable distributed protocols. He studied issues related to complexity of different distributed network protocols such as synchronization and deadlock resolution, as well as some classic graph-theoretic problems, like Breadth-First-Search and Spanning Tree. He also worked on cryptographic protocols for implementing simultaneous broadcast networks.

Research on network synchronization has provided a simple and efficient methodology for developing simple and efficient protocols in an asynchronous network. This was achieved by developing a communication protocol, referred to as synchronizer, which transforms an arbitrary synchronous protocol in such a way that the resulting protocol runs correctly on an asynchronous network. The synchronizer is, in effect, a compiler of high-level synchronous algorithms on a low-level asynchronous machine. The proposed synchronizer is proved to be optimal as far as its time and communication requirements are concerned. It also yields improvements for existing problems, *e.g.*, distributed Breadth-First-Search and Maximum Flow.

The research on distributed Breadth-First-Search algorithms was conducted jointly with R. Gallager (MIT). This problem arises in relation with the efficient routing of messages towards a certain destination node. A number of algorithms have been discovered whose complexities are close to optimal. Another graph problem that was investigated was the problem of constructing (distributedly) a Spanning Tree in the network. An optimal algorithm for this problem was found.

Research on detection and resolution of deadlocks in a communication network was conducted jointly with with Micali. Deadlock occurs if the amount of work that the network attempts to perform requires more resources than the network actually has. As a result, different network transactions get stuck in the middle, since no transaction can get enough resources to proceed to its completion. Efficient solution of this problem would result in improved performances of communication networks, and would prevent wasting precious resources like communication bandwidth and memory space. The main contribution of this research is a new protocol for deadlock resolution, which deals with the most general form of deadlocks. Its complexities in communication, time, and memory requirements are optimal.

Cryptographic protocols in unreliable networks were investigated jointly with Chor, Goldwasser and Micali. The main problem being studied was implementing simultaneous networks, where the abilities of potential adversaries are limited. Many fundamental protocols (*e.g.*, coin flipping, contract signing, etc.) can be implemented quite easily in simultaneous networks. The protocol which was developed simulates simultaneous network by a non-simultaneous network, provided that the number of adversaries is not too big.

Awerbuch plans to continue research in distributed algorithms and communication protocols. He is mainly interested in efficient algorithms for practice-oriented problems. In particular, he plans to work on protocols for implementing atomic access to shared registers for asynchronous system of processes and on hierarchical routing schemes in networks. Another problem he plans to work on is the development of modular ways of describing distributed algorithms.

3.1.2 Benny Chor

Chor worked on various aspects of randomized agreement algorithms in unreliable distributed environment. He studied the implementation of simultaneous broadcast in Byzantine environment (joint work with Goldwasser, Micali, and Awerbuch). Together with Merritt (MIT) and Shmoys, he developed protocols for distributed consensus in constant expected time with improved fault tolerance in various synchronous and asynchronous failure-by-omission scenarios. Chor wrote a survey paper (jointly with C. Dwork (IBM-SJ)) on randomized algorithms for fault-tolerant distributed agreement.

Chor investigated VLSI implementations of exponentiations in finite fields of characteristic two, which is the computational bottleneck of a public-key cryptosystem that he designed in his thesis with Rivest. In addition he worked on improving schemes for extracting unbiased bits from weak sources of randomness. He also worked on better lower bounds for probabilistic communication complexity (continuing past work with Oded Goldreich).

3.1.3 Peter Elias

Two simple-minded data-compression algorithms have been analyzed which permit immediate encoding of each message without delay or knowledge of the statistical characteristics of the message source [45]. Receiver and transmitter need only agree in advance on an indexing of the possible messages by the positive integers and on a sequence of (variable-length) binary codewords which represent those integers, no one of which is the beginning of another. In the algorithm simplest to implement, the transmitter encodes each message into the codeword which represent the total number of messages sent since it last occurred. The data structure required is simply an array which stores in index order the counter reading at the most recent occurrence of each message: an implementation could run very fast. A somewhat better scheme, with much more complex data structure requirements, encodes each message into the codeword which represents the number of distinct messages sent since its last occurrence. It has also been analyzed by Bentley, *et al.* If a good set of codewords is chosen to represent the integers then either scheme can do almost as well on any finite sequence of text as would an off-line Huffman encoding which measured the frequencies of the letters in the sequence and designed a code to match, providing that the information per letter of text is not too small: for example if the text is taken to be a sequence of words rather than letters. Other adaptive on-line schemes can do better, but at added costs in complexity, both of concept and of necessary data structure.

3.1.4 Oded Goldreich

Goldreich has been mainly interested in two topics: zero-knowledge proofs and software protection. In addition he improved the Goldwasser-Micali-Rivest signature scheme, and has been working with Chor on developing better schemes for extracting unbiased bits from weak sources of randomness.

Together with Micali and A. Wigderson (UC-Berkeley), he has demonstrated the generality and wide applicability of zero-knowledge proofs, a fundamental notion introduced by Goldwasser, Micali and Rackoff (Toronto). Zero-knowledge proofs are probabilistic and interactive proofs that efficiently demonstrate membership in a language without conveying any additional knowledge. So far, zero-knowledge proofs has been known for some number theoretic languages in the intersection of NP and Co-NP. Under the assumption that encryption functions exist, it is shown that all languages in NP possess zero-knowledge proofs. This result is used to prove two fundamental theorems in the field of cryptographic protocols. These theorems consists of standard and efficient transformations that, given a protocol that is correct with respect to a very weak adversary, output a protocol correct in the most adversarial scenario. Using no assumptions, zero-knowledge proofs for both graph isomorphism and graph non-isomorphism, are presented.

Software protection is one of the most important issues concerning computer practice. The problem is to sell programs that can be executed by the buyer, yet cannot be duplicated and/or distributed by him to other users. Goldreich made the first steps towards a theoretic treatment of software protection, by distilling and formulating the key problem of learning about a program from its execution, and by presenting an efficient way of executing programs (*i.e.*, an interpreter) such that it is infeasible to learn anything about the program by monitoring its executions. Current cryptographic techniques can be applied to keep the contents of the memory unknown throughout the execution, but are not applicable to the problem of hiding the access pattern. Hiding the access pattern efficiently is the essence of new solution. It is shown how to implement (on-line) t fetch instructions to a memory of size m by making $t \cdot m^{\epsilon}$ actual accesses, for every fixed $\epsilon > 0$. A reasonable scheme that protects against duplication follows.

Goldreich served on the program committee of Crypto85 conference.

3.1.5 Shafi Goldwasser

Goldwasser has done research in two areas: computational number theory and the study of interactive proof systems.

Goldwasser and Kilian devised a new probabilistic primality test which outputs a "short" (deterministic polynomial time verifiable) proof of correctness of its assertions of primality and compositeness. Thus its assertions of primality are certain rather than being correct with high probability (if a fair coin is used) or dependent on unproven assertions as in the tests of Miller, Solovay-Strassen, and Rabin in that its assertions of primality are certain, rather than being correct with high probability. The test terminates in expected polynomial time on all but at most an exponentially vanishing fraction of the inputs of every length.

This result shows that:

- There exist an infinite set of primes which can be recognized in expected polynomial time.
- Large certified primes can be generated in expected polynomial time (all previous methods were heuristics).

Under a very plausible condition on the distribution of primes in "small" intervals, the proposed algorithm can be shown to run in expected polynomial time on every input. This condition is implied by Cramer's conjecture. The methods employed are from the theory of elliptic curves over finite fields.

Goldwasser and Sipser examined the relative power of two interactive proof systems generalizing NP. An interactive proof system is a method by which one party of unlimited resources, the "prover", can convince a party of limited resources, the "verifier," of the truth of a proposition. The verifier may toss coins, ask repeated questions of the prover, and run efficient tests upon the prover's responses before deciding whether to be convinced. This extends the familiar proof system implicit in the notion of NP in that there the verifier may not toss coins or speak, but only listen and verify. Interactive proof systems may not yield proof in the strict mathematical sense: the "proofs" are probabilistic with an exponentially small, though non-zero chance of error.

Two notions of interactive proof system have been defined. One, by Goldwasser, Micali, and Rackoff permits the verifier a coin that can be tossed in "private" *i.e.*, a secret source of randomness. The second, due to Babai requires that the outcome of the verifier's coin tosses be "public" and thus accessible to the prover.

Goldwasser and Sipser show that the two systems are equivalent in power with respect to language recognition. This result has implications on the ease of design of cryptographic protocols, as interactive proof systems provide a useful framework for studying the correctness of cryptographic protocols.

Goldwasser has co-organized an NSF sponsored workshop on the Mathematical Theory of Security, which was held in Endicott house in the past summer. She is currently on the program committee of Crypto 86.

3.1.6 Leonard S. Heath

Heath completed his PhD in computer science at the University of North Carolina in August, 1986 [67]. He has worked on the book embedding problem, which is related to the realization of arrays of VLSI processors fault-tolerantly. He has developed algorithms for embedding various classes of graphs in books [68], [69]. Recently, Heath and Istrail (Wesleyan University) have obtained an efficient algorithm for embedding a graph of genus g in a book of $O(g)$ pages [70]. This disproves an implicit conjecture of Bernhart and Kainen that the page number of genus g graphs is unbounded.

Heath will continue research into approaches for fault-tolerant computing. He hopes to develop approaches that are both theoretically sound and practical.

3.1.7 Tom Leighton

The highlight of the year is the joint work with Peter Shor (MSRI) and Ed Coffman (Bell Labs) on the minimax grid matching problem and its application to the average case analysis of algorithms for such problems as bin packing, dynamic allocation and wafer-scale integration. The work also turned out to have applications to some planar matching problems, possibly to testing pseudorandom number generators and to some problems in mathematical statistics. Leighton spoke about some of this material at last year's review, and is in the process of finishing off the related papers now.

Leighton is continuing work on channel routing with Berger, Donna Brown (U. Colorado), and Marty Brady (U. Illinois). They are in the process of improving the bounds and algorithms for channel routing in two or more layers. The results are not yet practical, but are definitely getting closer. The last "factor of two" from the channel widths produced by the algorithm has been eliminated, and the focus is now on the additive terms.

Leighton is also continuing work on the development of algorithms for integrating two-dimensional arrays on wafers that contain faults. He is looking at both worst case and average case models, and is optimistic that recent results will prove to be practical. Leighton is working with a UROP student over the summer to implement the most recent algorithms on realistic problems.

In the coming year, Leighton hopes to return to earlier work on the problem of efficiently routing messages in a fixed connection network, an increasingly important problem in parallel computation and an old favorite. The problems in this area have mostly been solved "theoretically" but most of the solutions are inefficient for small values of N (say $N \leq 2^{20}$), which of course is where all the action is. It would be nice to find a theoretically sound and practically efficient scheme for routing messages in a high bandwidth, low diameter network such as the hypercube.

Leighton is also planning to work on the design and analysis of ethernet protocols which work well on average. Currently, most ethernet systems use a protocol called exponential backoff to decide when to retransmit a message that got garbled because of simultaneous transmissions. While this has worked reasonably well in practice so far, there is very strong evidence that the protocol will start to fail dramatically as the traffic on typical ethernets continues to increase. He has an excellent candidate for replacing exponential backoff, and are currently trying to prove that it will work, even for very high levels of message traffic. The experimental evidence is very encouraging and already Hastad (who will postdoc in the Math Dept. next year) and Leighton have proved some fairly positive lemmas concerning performance.

By far the biggest project (both recently and in the coming year) is Leighton's book on "An Introduction to the Theory of Networks, Parallel Computation, and VLSI Design." The book is based on the course that he has taught for the last three years on parallel computation and VLSI design, and is planned to be usable as a textbook and a reference text for people working in the field (probably a hopeless combination). So far, about 100 pages have been written.

3.1.8 Charles Leiserson

Leiserson has continued his work on volume-universal networks and now has several improved designs. One simple network contains only n small switches for a network with n processors and is a generalization of the Benes network. He and Greenberg have discovered good message-routing algorithms for this and other volume-universal networks. He has also developed with Maggs a PRAM-like abstraction of volume-universal networks. The new model, called a *distributed random-access machine* (DRAM) allows the communication requirements of an algorithm to be effectively measured.

Leiserson has also been working with Ishii on the semantics and timing analysis of clocked circuits, and with Kilian and Kipnis on understanding the power of multiple-pin interconnections. He was the program chairman for the recent Fourth MIT Conference on Advanced Research in VLSI, whose proceedings is now a book from MIT Press. He also won the Best Presentation Award for his paper on fat-trees at the 1985 International Conference on Parallel Processing.

3.1.9 Nancy Lynch

Lynch's research is on theoretical aspects of distributed computation. This involves design and analysis of distributed algorithms, proofs of lower bounds, and formal modelling of distributed algorithms and systems.

A major effort was a system modelling project, with collaborator Dr. Michael Merritt (AT&T Bell Labs), in the area of distributed database concurrency control and resiliency. This area is of critical importance to distributed computing, but the work was previously described in hundreds of unrelated research papers, with no common framework to aid in comprehension. Lynch and Merritt developed a satisfactory common framework, and used it to present and verify, an important exclusive-locking algorithm for concurrency control and resiliency. They are now extending the results to many other algorithms.

Other work involved specifying properties that can be guaranteed by (nonserializable) highly available replicated database systems. Another project, joint with student Mark Tuttle, involved establishing a basic model for concurrent computation which could be used in carrying out hierarchical correctness proofs for distributed algorithms. Work with students Brian Coan, Jennifer Lundelius and Alan Fekete involved determining the costs of solving various problems of reaching consensus in fault-prone distributed networks.

For further details and references, see the report of the Theory of Distributed Computation research group.

3.1.10 Albert R. Meyer

Meyer's research focuses on the semantics and logic of programming languages. He recently solved an open problem of some years' standing about axiomatic semantics of programs [99]. His joint paper with Reinhold [101] analyzes the differing criteria desired or expected of typechecking disciplines and establishes that the appealing 'type of all types' concept of polymorphic type discipline violates many of these criteria. Meyer also gave an invited lecture on these issues at the 1986 ACM Symposium on Principles of Programming Languages. For further information on specific research topics, see the references [108], [100], [28], [107], [27], and the personal reports of his students Breazu-Tannen and Reinhold (working on type theory), and Bloom (on concurrency).

Meyer supervised two weekly research seminars, one on semantics and logic of concurrent processes (jointly with Lynch) and another on types in programming (jointly with Trakhtenbrot). Both seminars were attended by faculty from several local universities as well as MIT. He was Program Chairman of the newly organized IEEE Symposium on Logic in Computer Science held June 16-18, 1986. He continues as Editor(-in-Chief) of the journal Information and Control, and is an Editor of a major Handbook of Theoretical Computer Science to be published in 1987 by North-Holland.

3.1.11 Silvio Micali

One focus of Micali's research has been on the use of randomness for proving theorems. Micali and coauthors have explored new avenues for efficiently proving theorems that would allow one to handle more theorems than in the traditional "NP" framework. So far, for example, it was not known whether a "prover", even having infinite power, could quickly convince a polynomial-time "verifier" that two given graphs are *not* isomorphic. These investigations resulted in a new, probabilistic and interactive way of proving theorems. For example, this way is powerful enough to allow an efficient and interactive verification of graph non-isomorphism. Micali intends to determine the power of the theorem-proving procedure and, more generally, to determine what theorems can be proved in any other efficient way.

Another focus is measuring and reducing the "amount of knowledge" conveyed by a communication. This is the basis of a rigorous and general study of the slippery field of cryptographic protocols. Of particular interest is a new "compiler-type" algorithm that seems fundamental in the the design of cryptographic protocols with more than two participants. This algorithm, given the specification of *any* protocol for totally honest parties (which is usually trivial to design), outputs an alternative and equivalent protocol that *remains correct* even if up to *half* of its participants deviate for their prescribed programs in an *arbitrary* (but polynomial-time bounded) way. Micali proposes to extend his "compiler" to handle the case of two-party protocols as well.

He also proposes to develop theory in the field of asynchronous distributed systems when all participants are reliable. Here the adversary to defeat is the arbitrary arrival time of messages. The only guarantee we have is that all messages will be eventually all delivered. However, messages sent first may arrive later and we do not have an upper bound the time that will take for a message to reach its recipient. Clearly, finding robust good solutions in this extreme model will, *a fortiori*, imply having found good solutions for more reasonable models.

3.1.12 Ronald L. Rivest

Goldman and Rivest have been investigating ways to improve the efficiency of computing the maximum-entropy probability distribution, given a set of constraints (in terms of expected values) for the unknown distribution. They are working on improvements to a recent approach suggested by Peter Cheeseman (NASA), which is similar in spirit to the fill-in-minimization problem encountered when solving sparse linear systems. They expect that the techniques being developed will yield very substantial improvements in the running time, in practice. (Some experimentation is needed to validate their ideas on realistic examples since the running times are exponentially sensitive to certain characteristics of the input.)

Rivest has invented three new algorithms for the control of a broadcast channel (of the slotted-ALOHA type). These algorithms are called Bayesian Broadcast, Pseudo-Bayesian Broadcast, and Recursive Pseudo- Bayesian Broadcast. The essential characteristic of these algorithms is that each station will use Bayes' Rule to estimate the probability, for each r, that exactly r stations have a packet to transmit in the next time slot. Here each station receives feedback as to whether each slot was empty, successful, or contained a collision. Experimental results (conducted by Michelle Lee), show that the Pseudo-Bayesian Broadcast algorithm has significantly less delay than previously published algorithms. It is also extremely simple to implement. (More recently, J. Tsitsiklis (MIT) has proven that it is stable with a packet arrival rate of up to e^{-1} packets/slot.)

Rivest has worked with Bloom on "Abstract AI" -- the study of computational mechanisms attempting to learn an environment into which they have been placed. They derived conditions on the environment which permit the environment to be learned effectively.

With Sherman and Kaliski, Rivest has investigated the security of the Data Encryption Standard (DES). Using a special-purpose board for computing DES quickly on an IBM PC, they ran experiments to determine whether DES suffers from being a pure cipher. (Such a weakness would allow a very efficient attack to be mounted against DES.) They found no such weakness.

3.1.13 David Shmoys

Shmoys has been studying several problems in fault-tolerant distributed computing. One of the fundamental problems in this area is "Byzantine Agreement", where each processor starts with a private input and at termination all processors that operate correctly agree on an output that depends non-trivially on the input (*e.g.*, if all correct processors have the same input bit *b*, then the output bit is *b*.) Shmoys, in joint work with Chor and M. Merritt (MIT), gave a family of protocols that terminate in constant expected time while tolerating a linear fraction of the processors failing. The results hold in a variety of computation models, both synchronous and asynchronous. However, this algorithm tolerates only a benign type of failure, where messages may not be delivered, but messages which arrive are correct. In separate work, Shmoys also provided a family of protocols that could handle arbitrarily malicious behavior by faulty processors, but for this approach to tolerate a linear fraction of faulty processors, the expected completion time grows to log log *n*.

Shmoys also continued his investigations in the design of provably good heuristic procedures for the solution of intractable optimization problems. The central notion in this work is that of a dual approximation algorithm. A traditional (or primal) approximation algorithm seeks a feasible but suboptimal solution for the problem where the degree of suboptimality possible is bounded. A dual approximation algorithm seeks a superoptimal but infeasible solution for the problem where the degree of infeasibility possible is bounded. In addition to their immediate application, Shmoys has shown that dual approximation algorithms are often useful in the design of primal approximation algorithms. This general strategy was applied in the case of several scheduling problems. For all of these problems the aim is to assign a given set of jobs to a set of parallel processors, so that all jobs are completed as quickly as possible. If the machines run at identical speeds, or if the processing requirement for each job varies uniformly on different machine, Shmoys proved that there is a polynomial approximation scheme, thereby showing that solutions arbitrarily close to the optimum can be obtained. In addition, Shmoys showed that if the processing times varied arbitrarily from machine to machine, then there was an algorithm that produced solutions within a logarithmic factor of the optimum.

3.1.14 Michael Sipser

Sipser has been continuing his work on questions in complexity theory. Recent results include:

- the equivalence of public-coin and private-coin interactive proof systems joint with Goldwasser [59] described in her section above,
- a connection between expander graphs and the relative power of time versus space [112]. This second result shows that explicit construction of certain kinds of expander graphs would enable their use as "randomness magnifiers" for simulating randomized computations.

Current emphasis is on developing new methods for proving the inherent computational complexity of a variety of problems in different settings. This Fall Sipser will conduct a seminar to survey the latest results and to discuss further directions.

3.1.15 B.A. Trakhtenbrot

Trakhtenbrot's research focus is on computability, sequentiality, and invariance of functionals on the one hand, and the language constructs which are able to express them on the other hand. The goal is to develop a unifying approach based on the theory of typed lambda-calculus and logical relations. Current plans are about the comparative study of different approaches to sequentiality:

- the semantical approach (as in earlier work with his students);
- the syntactical approach (as in the rewriting paradigm);
- interpretations used in the theory of concurrency.

3.2 Students and Visitors

3.2.1 William Aiello

Aiello has been working with Goldwasser and Hastad on open problems concerning Arthur-Merlin games and Interactive Proofs (IP). Recently, they constructed an oracle which separates the class of languages recognized by IP from the polynomial hierarchy. In particular, this implies that relative to our oracle more languages can be recognized by IP's (or Arthur-Merlin games) with a polynomial number of interactions than with a constant number. This in turn gives evidence that the unrelativized statement is also true. We also hope to show that Babai's methods for proving that the AM hierarchy collapses at the second level are optimal.

3.2.2 Dave Barrington

Barrington has been finishing up his Ph.D. research under Sipser in the Math department. He has been studying branching programs, a model of computation where the settings of Boolean input variables determine a path through a directed acyclic graph to a final node which accepts or rejects the input. He showed that branching programs of constant width and polynomial size are much more powerful than previously believed --- they can simulate all of the parallel complexity class NC^1 (Boolean circuits of fan-in two and depth $O(\log n)$.

Further work has shown that this unexpected power comes from the ability of branching programs of width at least five to represent non-solvable permutation groups. Recently, with Denis Th erien of McGill University, Barrington has used the connection between branching programs and finite monoids to give new characterizations of important subclasses of NC^1. He has also applied his main result to show that Boolean circuits of width four and polynomial size can recognize exactly NC^1.

Barrington has been working with complexity classes with very constrained resources, such as NC^1 or log-space Turing machines. These classes can serve as an important laboratory for developing insights about more general computations and the big questions --- what do non-determinism, randomness, parallelism, or non-uniformity add to computational power?

In Fall, '86, he will take a faculty position in the Computer and Information Sciences department of the University of Massachusetts, Amherst, but expects to be a frequent visitor to MIT.

3.2.3 Bonnie A. Berger

Berger and Leighton have worked on achieving better bounds and algorithms for channel routing. Channel routing plays a crucial role in the development of automated layout systems for integrated circuits. Jointly with Brady (U. Ill.) and Brown (U. Col.), they have developed algorithms for routing channels with several layers [17]. They not only have achieved substantial new results, but have provided a unified framework in which many previously known results can be obtained.

For the unit-vertical-overlap model, they have developed a 2-layer channel routing algorithm which uses at most $d+O(d^{1/2})$ tracks to route two-point net problems and $2d+O(d^{1/2})$ tracks to route multipoint nets. They have also shown a $d+\Omega(\log d)$ lower bound for routing two-point nets on 2 layers even when unrestricted vertical overlap is allowed; hence, their upper bound is nearly optimal even in a more general setting. Moreover, they have demonstrated the robustness of their algorithm by showing how it can be used to obtain the known bounds for the Manhattan and knock-knee models.

Finally, they generalized the algorithm to unrestricted multilayer routing and used only $d/(L\text{-}1) + O(d^{1/2})$ tracks for two-point nets (within $O(d^{1/2})$ tracks of optimal) and $d/(L\text{-}2) + O(d^{1/2})$ tracks for multipoint net problems (within a factor of $(L\text{-}1)/(L\text{-}2)$ times optimal).

3.2.4 Bard Bloom

Meyer and Bloom are studying the denotational semantics of parallel and nondeterministic processes. Dana Scott's very successful models for the semantics of sequential, deterministic programs do not extend naturally to the more general domain. There are a number of proposals for a replacement; Meyer and Bloom are investigating several of these models.

Meyer and Bloom are also investigating the complexity of the operational semantics of Milner's SCCS. They have proved that the problem of deciding if an arbitrary program can take a single step is not recursively enumerable, under some barely reasonable assumptions about the set of actions. They are in the process of extending this result to completely reasonable assumptions.

3.2.5 Ravi B. Boppana

Boppana has worked on proving lower bounds for several restricted models of Boolean circuits. His Ph.D. thesis [23] on this topic was completed in May 1986.

The first circuit model studied is monotone circuits. A major development appeared in [A. A. Razborov, "Lower bounds for the monotone complexity of some Boolean functions," Dokl. Ak. Nauk. SSSR 281 (1985), pp. 798-801 (in Russian). English translation in Sov. Math. Dokl. 31 (1985), pp. 354-357] which showed that monotone circuits detecting cliques in graphs must have superpolynomial size. Alon and Boppana [2] show that the lower bound can be improved to exponential size.

The second model is that of Boolean formulas. [L.G. Valiant, "Short monotone formulae for the majority function," J. Algorithms 5, 363-366, 1984] shows that the majority function has short monotone formulas; his construction uses the general technique of combining independent copies of probabilistic formulas to amplify the performance of the formulas. Boppana [22] showed that the amount of amplification obtained by Valiant is actually best possible.

Lagarias and Boppana [26] studied the computational complexity of checking a relation versus evaluating a relation. Hirschfeld and Boppana [25] wrote an expository paper on pseudorandom number generators and their relationship to complexity theory. Hastad and Boppana [24] studied approximation properties of constant depth circuits.

Boppana plans to continue work in Boolean circuit complexity. In particular, he will investigate whether the recent lower bounds for monotone circuits can be extended to the nonmonotone case.

3.2.6 Val Breazu-Tannen

Breazu-Tannen has studied equational type disciplines (see [27]) and λ-calculus with least fixpoint operators. He served as organizer for the TOC " Types in programming" seminar where he also made several presentations.

His current interests include:

- Conservative extension theorems in typed λ-calculi; Polymorphism and second-order λ-calculus; Dependent types and the calculus of constructions; Relations between type disciplines.

- λ-calculus with equational type disciplines (*e.g.*, types satisfying domain equations); Type coercion.
- λ-calculus with least fixpoint operators.
- Type theory; Higher-order categorical logic (cartesian closed categories, topoi); Higher-order algebraic theories.
- Logical relations and representation independence theorems.
- Learning about *concrete* programming environments to which the ideas of the above fields of theoretical investigation apply.

Breazu-Tannen's current research focuses on conservative extension theorems about programming language data types. The history of programming language design has seen a progressive addition of powerful features such as higher-order types, recursively defined types, polymorphic types, dependent types and others, giving rise to more and more complex type disciplines. Such features are theoretically modeled by various *typed λ-calculi*. Adding a new feature corresponds to seeing the new calculus as an *extension* of an old one. If the old calculus is consistent, is the new one too? A stronger question is: are all old program phrases that were not provably equivalent in the old calculus still not provably equivalent in the extended calculus? (*i.e.*, is the extension conservative?) An affirmative answer provides formal evidence for the *orthogonality* of the new feature with respect to the the old ones, a well-established desideratum in programming language design. Both model-theoretic and proof-theoretic techniques have been used to establish conservative extension, *e.g.*, finitely typed λ-calculus is conservative over any first-order equational theory, but untyped λ-calculus is not.

3.2.7 Jonathan Buss

Buss' research has been in the area of relativized complexity theory. Previous work in this area has been largely concerned with polynomial-time-bounded machines. In the case of space-bounded machines, there are several possible models in the literature, none of which is entirely satisfactory. In particular, results connecting time and space do not relativize in the previous models. These definitional problems have obstructed progress in this area.

Principal work has been the formulation of a new model for alternation oracle machines. In the new model, the connections between time and space bounds provided by alternation (ALOGSPACE=P, AP=PSPACE) hold relative to any oracle. When the model is reduced to the case of deterministic machines, a slightly different model of relativized time and a new model of relativized space result. These deterministic models are better-behaved than the standard models. Several results on the simulation of time by space and the simulation of multihead Turing machines hold for all oracles only in the new model. This work is reported in his doctoral thesis [30].

Continuing work is concerned with the implication of the new model for log-space hierarchies and with relativized probabilistic machines.

3.2.8 Tom Cormen

Cormen has continued his work in VLSI designs of concentrator switches, building on earlier joint work with Leiserson in designing a fast hyperconcentrator switch. Addressing the problem of pin boundedness, Cormen has designed an (N,m,α)-partial concentrator switch based on a recent technique for sorting on a mesh. This switch uses the earlier hyperconcentrator switch as a subcircuit, requires at most $2N^{1/2} + 1/2 \log N$ data pins per chip, has $\alpha = 1\text{-}O(N^{3/4}/m)$, and can be packaged in three dimensions with volume $O(N^{3/2})$. The switch can route any set of $k \leq \alpha m$ bit-serial messages from input wires to output wires. A signal incurs at most $7/2 \log N + O(1)$ gate delays in passing through the switch.

3.2.9 Paul Feldman

Feldman has been working with Micali on cryptographic protocols even before transferring to MIT from Harvard this year. One result of this collaboration was an algorithm for reaching Byzantine agreement in constant expected time whenever fewer than a third of the processors are faulty [48]. Another result, an efficient scheme for verifiable secret sharing, is expected to be included as part of his doctoral thesis.

Feldman obtained some results about connecting networks [47] working with Joel Friedman and Nick Pippinger at IBM-San Jose.

Feldman has been interested in the field of interactive proof systems, and has shown that if any prover can convince a polynomial time verifier about membership in a certain language, then there exists a polynomial space prover that can do the same.

3.2.10 Andrew V. Goldberg

Goldberg has been working on network flow problems. He coauthored a paper [51] that introduces a new method for solving a classical max-flow problem. Currently he is looking at other network flow problems, including 0-1 flows, multicommodity flows, load balancing, etc. His other interest include parallel algorithms, parallel architectures, and complexity theory.

3.2.11 Sally A. Goldman

Goldman has been working with Rivest on theoretical aspects of artificial intelligence. Specifically, they are studying the problem of calculating the maximum entropy distribution satisfying a given set of constraints. These constraints may be given by an expert or discovered by a learning program. They have compared several methods from the literature as well as beginning work on developing a new method. Their method simplifies maximum entropy computations by adding extra constraints.

3.2.12 Ron Greenberg

Greenberg has continued his investigation of the fat-tree architecture for general-purpose parallel supercomputing. The randomized routing algorithm for fat-trees which he developed with Leiserson demonstrates that fat-trees are universal routing networks [60]. That is, in a VLSI model equating hardware cost with physical volume, any routing network can be efficiently simulated by a fat-tree of comparable hardware cost.

3.2.13 Johan Hastad

The two major topics of Hastad's research have been integer lattices and lower bonds on circuit complexity.

The work on lattices consists of two different types of work. First applying old techniques to new problems and secondly to find new techniques. In [62] the LLL algorithm is applied to prove that encoding linearly related messages with RSA with low exponent is insecure. In the area of new techniques the joint paper [65] proposes a polynomial time algorithm which finds integer relations among real numbers. The algorithm works in the model where real arithmetic can be done at unit cost.

The work on lower bounds for circuits has been concentrated on small depth circuits. Hastad has established almost optimal lower bounds for small depth circuits computing functions like parity and majority [63]. He has also proved that for any constant k there are functions which have linear size circuits of depth k but which require exponential size circuits for depth k-1. This work is included in his recently completed Ph.D. thesis [64].

These lower bounds have implications for relativized complexity. With Aiello and Goldwasser [1], Hastad has used similar methods to prove that there is an oracle A and a language L(A) with the following property: L(A) is not in the polynomial time hierarchy relative to A, but given the oracle A it is possible to effectively prove membership in L(A) using a randomized interactive proof.

Future plans include trying to prove lower bounds for circuits with less severe restrictions on the depth. For instance to prove that some explicit function is not in NC^1. He also would like to investigate further the notion of interactive proof.

3.2.14 Rafael Hirschfeld

The topics of Hirschfeld's research are pseudorandom number generators and complexity theory. He has investigated necessary and sufficient conditions for the existence of pseudorandom generators that cannot feasibly be distinguished from true random sources, as well as the implications of the existence of such generators for the relationship between deterministic and probabilistic computation. Boppana and Hirschfeld [25] have written an expository paper on this and related work.

3.2.15 Alexander T. Ishii

Ishii, in joint work with Leiserson, has been studying the analysis of clocked multi-phase VLSI systems. Progress to date includes the development of a new semantic model for the functionality of VLSI systems. Promising aspects of the model include strong correspondence to widely held intuitive notions of functionality, and freedom from *a priori* assumptions about the relationship between different clock phases. In addition, study of the model has resulted in the identification of a number of clocking phenomena, which raise questions about the ability of standard simulation and analysis techniques to verify an optimally clocked circuit. It is hoped that continued study will result in the formulation of an algorithm to compute provably optimal clock phases for any specific VLSI system.

3.2.16 Burt Kaliski

Kaliski completed experimental research on the Data Encryption Standard (DES) and studied applications of elliptic curves to cryptography.

The DES research, with Rivest and Sherman, involved the design, implementation and testing of special-purpose hardware for an IBM PC to perform the 2^{32} or more DES encryptions per day necessary for a variety of statistical tests. In summer 1985, the board was used to perform eight experiments, most of which confirmed expected properties of DES (*i.e.,* that it behaves like a set of randomly-chosen permutations). The experiments also uncovered fixed points for the "weak keys" in DES.

Some of the most interesting recent developments in computational number theory are related to elliptic curves. H.W. Lenstra's algorithm for integer factorization, and two other related algorithms (by Ren e Schoof and Victor Miller) all were published in the last two years, following a century of theoretical development. The elliptic curve study, with Rivest and Micali, and helpful insight from Goldreich, sought to apply the *elliptic logarithm problem* to cryptography, just as several other hard problems (factoring, discrete logarithm) have been applied. The main result is the construction of a *pseudo-random bit generator* using elliptic curves. Two related results of independent interest were also developed: a method of determining the order of an element in an arbitrary abelian group, with negligible error; and an oracle proof method for the simultaneous security of multiple bits of a discrete logarithm in an arbitrary abelian group.

Future work will include the documentation of the special-purpose DES hardware, and extensions of the results on elliptic curves. The latter may involve generalization to more complicated algebraic structures, an attempt at a sub-exponential time algorithm for computing elliptic logarithms, or the use of elliptic curves in "certifying" elements that generate a cyclic group.

3.2.17 Joe Kilian

Kilian's chief work has been a joint paper with Goldwasser [56] in which they apply the theory of elliptic curves to solve some open problems relating to primality testing.

He is also working on a paper with Kipnis and Leiserson applying elementary group theory to some problems concerning the power of multi-point nets.

3.2.18 Philip Klein

Klein has been studying parallel algorithms for language-theoretic and graph-theoretic problems. He has revised his joint paper with Reif [84] on a parallel algorithm for recognition of any fixed deterministic context-free language. Previously a parallel algorithm existed for general context-free language recognition, but this algorithm was slower and used many more processors.

For his Masters' thesis, Klein has developed, in collaboration with Reif, a new parallel algorithm for finding a planar embedding of a graph [83].

Klein will continue to work towards developing new efficient parallel algorithms.

3.2.19 Shlomo Kipnis

During this first year at MIT, Kipnis worked with Leiserson on theoretical VLSI problems. They tried to investigate multiple-pin nets, where each net may connect more than two processors (chips). Attention was given mainly to networks that realize some group of permutations among the chips. Together with Kilian some theoretical results from group theory were applied that characterize some networks.

Currently he is gathering the results on this topic into a paper with Leiserson and Kilian. He intends to continue research in this direction and related VLSI problems.

3.2.20 Bruce Maggs

Leiserson and Maggs have shown that many graph problems can be solved in parallel, not only with polylogarithmic performance, but with efficient communication at each step of the computation. The communication requirements of an algorithm are measured in a parallel random-access machine model called the distributed random-access machine, which can be viewed as an abstraction of volume- and area-universal networks such as fat-trees. In this model, communication cost is measured in terms of the congestion across cuts of the machine. The graph algorithms are based on a generalization of the prefix computation on lists to trees. These treefix computations, which can be performed in a communication-efficient fashion using a variant of the tree contraction technique of Miller and Reif, simplify many parallel graph algorithms in the literature.

3.2.21 Miller Maley

Maley continued his work on the mathematical foundations of wire routing. Using ideas from algebraic topology, he began developing a theory of planar wiring under homotopy constraints. This research aims to extend the scope of proposed algorithms for single-layer routing and compaction of VLSI layouts. The new theory facilitates the construction and rigorous justification of such algorithms.

3.2.22 Seth Malitz

Malitz has established some surprising facts connecting measure- and graph-theoretic properties of infinite graphs whose edges are represented by points on the real unit square [98]. Future plans are to work with Goldwasser and Sipser on interactive proof systems and with Bjorner (MIT Math.) on continuous analogues of finite lattices.

3.2.23 Yoram Moses

See the report of the Theory of Distributed Computation group.

3.2.24 Cynthia Phillips

Phillips completed her Master's Thesis on space-efficient algorithms for computational geometry [109] supervised by Leiserson. She is currently investigating several possible topics for her Ph.D. research including parallel data structures, theory of VLSI and parallel computation, and pseudorandom permuter circuits.

3.2.25 Satish Rao

Rao is continuing work on a Master's thesis (begun at Bell Laboratories) on finding optimal separators for planar graphs. He currently is working with some success on restricted versions of the problem, but hopes to eventually tackle the more general problem.

3.2.26 Mark B. Reinhold

Dependent function types, which originated in the type theory of intuitionistic mathematics, have recently appeared in programming languages such as CLU, Pebble, and Russell. (A function has a dependent type when the type of its result depends upon the value of its argument.)

In [101], Reinhold and Meyer investigate the consequences of assuming that there exists a *type of all types* in a λ-calculus with dependent function types. The type of all types is the type of every type, including itself. When a language with dependent function types is enriched with the type-of-all-types assumption, enormous expressive power is gained at very little apparent cost. By reconstructing and analyzing a paradox of Girard, they show that this combination leads to several serious problems. The most significant of these are that for such a language (1) typechecking is undecidable, and (2) classical reasoning about programs is not sound.

The technical properties if λ-calculus with dependent types are complex, and Reinhold is currently developing the basic theory of conversion and type-checking for such languages in his Master's thesis.

3.2.27 Alan Sherman

Sherman, with Kaliski and Rivest, has been investigating the relationship between algebraic and security properties of cryptosystems [80], [79], [111]. Using special-purpose hardware, they carried out a series of cycling experiments on the Data Encryption Standard (DES) to see if DES has certain extreme algebraic weaknesses. Their experiments show, with overwhelming confidence, that the set of DES transformations does not form a group under functional composition. Their experiments also detected fixed points for the so-called "weak-key" transformations, thereby revealing a previously unpublished additional weakness of the weak keys.

3.2.28 Jeffrey Mark Siskind

Siskind's research interests have centered around three different topics, namely silicon compilation from first principles, generalized phrase structure grammars, and logic programming and constraint propagation.

Silicon Compilation: All present silicon compilers, including MacPitts, generate layouts by using a set of predefined module generators. There are two fundamental problems with module generators. First, they are capable of producing only fixed classes of architectures. Second, they are technology dependent and require a significant effort to retarget to new technologies. The new approach, like MacPitts, separates the silicon compilation task into two phases, a translation from behavior to structure, and then a translation from structure to layout. Unlike MacPitts however, the intermediate structural format is a simple flattened network of transistors, rather than specifications for specialized module generators. Experimentation is underway with two different approaches towards the translation of this transistor network to efficient layout. The first uses a divide and conquer min-cut placement algorithm followed by a combination of Lee routing and slice and expand routing. The second more sophisticated approach is based on planar graphs. This work is being pursued under the supervision of Leiserson.

GPSG: Generalized Phrase Structure Grammars (GPSG) are a linguistic formalism developed by Gerald Gazdar for describing natural languages. They consist of small base set of context free grammar rules which are expanded by application of rule schemas, metarule transformations, and feature agreement to form a much larger set of context free rules which can be used to parse sentences. Several approaches are being explored towards the goal of testing the accuracy and adequacy of this formalism for both describing and parsing English. A GPSG grammar of about 100 rules has been constructed which can handle a fairly large subset of English. A system which expands this GPSG representation into a CFG representation has been constructed, and produces around 6000 CFG rules for the above GPSG rule set. The CFG rules are fed into an implementation of the Earley parsing algorithm and run against a small corpus of test sentences. This approach has proven to be very unwieldy and has prompted the construction of two alternative systems. The first is a modification of the Earley parser to handle feature agreement directly by using a unification algorithm internally avoiding the need to expand the rule set. This alone is not sufficient as capability to handle derived rules without expansion must be provided. The second approach is to use Prolog based definite clause grammars which imply a depth first backtracking approach rather than the breadth first approach of Earley's algorithm. This work is being pursued under the supervision of Robert Berwick (MIT) and William Woods (Applied Expert Systems, Cambridge, MA).

Logic Programming: Two different paths of research in logic programming have been pursued until now.

One path, centered around Prolog, provides unification and chronological backtracking as its main tenants. The second path, that of constraint propagation, focuses on dependency directed backtracking, but does not provide the necessary data and program abstraction capabilities to form a useful programming language. This research centers on a new language, Conlog, which merges the concept of unification (for providing data and program abstraction) with constraint propagation and dependency directed backtracking. As the new language is purely functional, simple syntactic transformations allow Conlog to express the lambda calculus as well, with the added benefit that many functions thus described are reversible. There are two new features which the above framework supports. The first is an extension of the dependency directed backtracking component to support generalized search of the solution space for minima, maxima, etc., in addition to just finding solutions. The second is a technique which is called intensional mode vs. extensional mode predicates, which allow a way of delaying and trading off computation. As a final benefit, viewing logic programming in the light of constraint propagation allows an implementation of Prolog for fine grain massively parallel computer architectures.

3.2.29 Bob Sloan

Sloan's major research activity has been to analyze several definitions of security for probabilistic public-key cryptosystems. He has been working with Micali, and have proved that some apparently distinct definitions are equivalent. His master's thesis on this topic is in preparation.

3.2.30 Mitch Wand

Wand has recently obtained results on the completeness of type inference systems, extending the line of work published in the last three POPL conferences. These results concern the completeness of type inference systems, which infer type declarations from programs without them. Wand was able to formalize the proof of completeness of the basic system in a way which allowed easy extensions to more complex type systems than had previously been considered. From this work, the following papers are in progress:

- A Simple Algorithm and Proof for Type Inference
- Deriving Typechecking Rules for Macros
- A Complete Type Inference System for Simple Objects with Inheritance

3.2.31 Su-Ming Wu

For his Master's thesis, Wu will work with Micali on algorithms for finding maximum matchings in general graphs.

4. Publications (Annotated)

1. Aiello, W., Goldwasser, S., Hastad, J. On the power of interaction. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, 00. Submitted.

> There exists an oracle which separates the class of languages recognized by interactive proofs from the polynomial hierarchy. In particular, this implies that more languages can be recognized by interactive proofs with a polynomial number of interactions than with a constant number of interactions.

2. Alon, N., and Boppana, R.B. The monotone circuit complexity of Boolean functions. *Combinatorica* 0 (1986), 00. To appear.

> Recently, Razborov obtained superpolynomial lower bounds for monotone circuits that detect cliques in graphs. Here his arguments are modified to obtain exponential lower bounds for monotone circuits that detect cliques.

3. Awerbuch, B. A new distributed depth-first search algorithm. *Information Processing Letters* **20** (1985), 147-150.

> A new distributed Depth-First Search (DFS) algorithm for an asynchronous communication network whose communication and time complexities are $O(|E|)$ and $O(|V|)$, respectively. The output of the algorithm is the DFS tree kept in a distributed fashion. The previous best algorithm due to Cheung required $O(|E|)$ both in communication and time complexities.

4. Awerbuch, B. Complexity of network synchronization. *J. ACM* **32** (1985), 804-823.

> The problem of simulation of a synchronous network by an asynchronous one. We propose a new simulation technique, referred to as "Synchronizer" which is a simple methodology for designing efficient distributed algorithms in asynchronous networks. Our Synchronizer exhibits a trade-off between its communication and time complexities, which is proved to be within a constant factor of the lower bound.

5. Awerbuch, B. Reducing complexities of distributed maximum flow and breadth-first search algorithms by means of network synchronization. *Networks* **15** (1985), 425-437.

> New simple distributed Maximum Flow and Breadth-First Search algorithms for an asynchronous communication network. Our algorithms improve the best known algorithms both in the communication and time complexities. The basic idea is first to "synchronize" the network and then to apply synchronous algorithms which use efficiently the parallelism of the model.

6. Awerbuch, B. Communication-time trade-offs in network synchronization. In *4th ACM Symp. Principles of Distributed Computing*, Strong, R.H., Ed., 1985, 272-276.

> Communication-time trade-off is intrinsic to the nature of asynchronous networks. We exemplify this phenomenon by proving a lower bound on the complexity of a very fundamental problem of network Synchronization. Namely, we show that any solution of this problem exhibits a certain trade-off between its worst-case communication and time complexities. This lower bound matches the known upper bounds within a constant factor.

7. Awerbuch, B. Optimal distributed spanning tree algorithm. 1986, Unpublished.

> A new distributed spanning tree algorithm for an asynchronous communication network. This is one of the most fundamental and most studied problems in the field of distributed network algorithms. The lower bounds on communication and time complexities of any algorithm for this problem are $\Omega(E + V \log V)$ and $\Omega(V)$, respectively. Our new algorithm achieves these bounds. The complexities of the best previous algorithms are $O(E + V \log V)$ in communication and $O(V\,\alpha(V))$ in time, where α is the inverse of the Ackerman's function.

8. Awerbuch, B., Gallager, R. Distributed breadth-first-search algorithms. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 250-256.

> Two new distributed BFS algorithms. The first has complexity $O((E+V^{3/2})\log V)$ in communication and $O(V^{3/2}\log V)$ in time. The second uses the technique of the first recursively and achieves $O(E^{1+\epsilon})$ messages and $O(V^{1+\epsilon})$ time, for any $\epsilon > 0$.

9. Awerbuch, B., Gallager, R. Communication complexity of distributed shortest path algorithms. Tech. Rep. LIDS-P-1473, MIT, 1985.

> A new distributed shortest path algorithm for finding shortest paths from a given root to all other nodes in an asynchronous communication network is presented. We assume unit edge weights, so that the shortest paths problem is essentially equivalent to constructing a Breadth First Search tree in the network. The communication complexity of our algorithm is $O(V^{1.6}+E)$ where V is the number of nodes and E the number of edges. For dense networks with $E = O(V^{1.6})$, this order of complexity is optimum.

10. Awerbuch, B., Micali, S. Dynamic deadlock resolution protocols. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, . To appear.

> The problem of deadlock resolution naturally arises in concurrent systems, *e.g.*, communication networks, distributed operating systems and distributed databases. The complexity of the problem stems from its inherent dynamic nature, since deadlock resolution protocol must operate 'on-line' without any knowledge of the future and using only local information. We are concerned with efficient deadlock resolution protocols that prevent wasting precious network resources, such as bandwidth, time, space, etc. The main contribution of the paper is a reduction of the most general dynamic problem to a conceptually simpler static off-line problem, in which all requests are generated simultaneously and are known *a priori*. The complexity of our reduction is $O(E + V \log V)$ in communication and $O(V)$ in time. Since the static deadlock resolution requires at least $\Omega(E + V \log V)$ in communication and $\Omega(V)$ in time, our reduction essentially shows that the resolution of dynamic deadlocks is not any harder than resolution of the static ones. We show here a simple and optimal algorithm for static off-line case, which, together with the above reduction, yields an optimal dynamic deadlock resolution protocol: $\Theta(E + V \log V)$ messages and $\Theta(V)$ time.

11. Barrington, D.A. Width 3 permutation branching programs. Tech. Rep. TM-293, MIT LCS, December, 1985.

> A restricted class of width 3 branching programs where each column of nodes depends on a single variable, and the 0-edges and the 1-edges out of each column form a permutation. In this model, parity and the mod-3 function are easy to calculate, but the *and*-function is hard. Any function of n inputs can be calculated in length $O(2^n)$, and that the *and*-function in particular requires length $O(2^n)$ if the branching program has one accept node and one reject node.

12. Barrington, D.A. Bounded width polynomial size branching programs recognize exactly those languages in NC^1. In *18th ACM Symp. Theory of Computing*, 1986, 1-5. Invited for a special issue of JCSS, to appear 1987.

> Any language recognized by an NC^1 circuit (fan-in two, depth $O(\log n)$ can be recognized by a particular type of branching program of width five and polynomial size. As any polynomial size constant width branching program can be simulated by an NC^1 circuit, the class of languages recognized by such programs is exactly non-uniform NC^1. This refutes a conjecture of Borodin *et al.* that *majority* could not be calculated by such programs. Similar results hold for the uniform setting of Ruzzo and Cook. The method of proof extends to the complexity of the word problem for a fixed permutation group. Journal version will include the newer result that width 4, polynomial-size Boolean circuits recognize exactly NC^1.

13. Barrington, D.A. Bounded Width Branching Programs. Ph.D. Th., MIT Dept. of Math., 1986. Supervised by Sipser, to appear as MIT/LCS TM.

The branching program model of computation and in particular the classes of languages which can be recognized when the width of the programs is bounded by a constant. Contains the results of the 1986 STOC paper and MIT report TM-293 and additional material.

14. Ben-Or, M., Goldreich, O., Micali, S., Rivest, R.L. A fair protocol for signing contracts. In *12th Int'l. Coll. on Automata, Languages and Programming, Lect. Notes in Comp. Sci.* **194**, Brauer, W., Ed., Springer-Verlag, 1985, 43-52.

We argue for a probabilistic approach to the problem of exchanging signatures to a contract. A contract signing protocol is considered to be fair if, at any stage during its execution, the following hold: the conditional probability that party A obtains B's signature to the contract given that B has obtained A's signature to the contract, is approximately 1. We present a two-party contract signing protocol that is fair under the assumption that one-way functions exist.

15. Bentley, J.L., Leighton, F.T., Lepley, M., Stanat, D., Steele, M. A randomized data structure for ordered sets. In *Advances in Computing Research*, Micali, S., Ed., 1986, 00. Submitted.

A simple randomized data structure for representing ordered sets, and gives a precise combinatorial analysis of the time required to perform various operations. In addition to a practical data structure, this work provides new and nontrivial probabilistic lower bounds and an instance of a practical problem whose randomized complexity is provably less than its deterministic complexity.

16. Berger, B. New Upper Bounds for Two-Layer Channel Routing. Master's Th., MIT Dept. of Electrical Eng. and Comp. Sci., May, 1986. VLSI Memo No. 86-312. Supervised by Leighton.

An algorithm for routing channels with two layers. We consider a generalized version of the two-layer knock-knee model in which wires are allowed to overlap for one vertical unit. By using this model, we achieve substantial new results for two-layer channel routing which are near optimal. Moreover, the algorithm presented in this report can be extended to reproduce many previously known results.

17. Berger, B., Brady, M., Brown, D. , Leighton, F.T. An almost optimal algorithm for multilayer channel routing. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, 00. Submitted.

A less restrictive (but still realistic) model of 2-layer channel routing, for which improved and nearly tight bounds on channel width are proved. The results are extended to multiple layer models for which much improved, and nearly tight bounds are also proved. Provides a unified framework in which many previously known results can be obtained.

18. Berman, F., Johnson, D.S., Leighton, F.T., Shor, P., Snyder, L. Generalized planar matching. *J. Algorithms* **0** (1986), 00. Submitted.

Maximum planar H-matching is NP-complete for any connected planar graph H with three or more nodes. We also show that perfect planar H-matching is NP-complete for any connected outerplanar graph H with three or more nodes, and is solvable in polynomial time for any triangulated planar graph H with four or more nodes. The results substantially generalize previous results in the literature and can be applied to determine the complexity of several applied problems, including the optimal tile salvage problem from VLSI design and the classic children's game of Dots and Boxes.

19. Bhatt, S., Chung, F.R.K., Leighton, F.T., Rosenberg, A. Optimal simulations of tree machines. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, 00. Submitted.

Embeddings of arbitrary binary trees into the hypercube so that the nodes of the tree are mapped one-to-one and onto the nodes of the hypercube and so that adjacent tree nodes are mapped to nodes which are only constant distance apart in the hypercube. Hence, we show that the hypercube is universal for binary trees. We also construct a bounded degree graph which contains all binary trees as spanning trees, thus establishing the existence of bounded degree tree-universal graphs.

20. Bloom, B. Multi-writer atomic memory. 1986, Unpublished.

An optimal construction of a two-writer atomic memory cell from two one-writer cells in a completely asynchronous model of computation. The construction permits any number of readers.

21. Bloom, B. and Rivest, R. Learning automata. 1986, Unpublished.

A class of worlds in which it is possible for an automaton to learn.

22. Boppana, R.B. Amplification of probabilistic Boolean formulas. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 20-29.

The amplification of probabilistic Boolean formulas refers to combining independent copies of such formulas to reduce the error probability. Valiant used the amplification method to produce monotone Boolean formulas of polynomial size for the majority function. We show that the amount of amplification that Valiant obtained is optimal.

23. Boppana, R.B. Lower Bounds for Monotone Circuits and Formulas. Ph.D. Th., MIT Dept. of Electrical Engin. and Comp. Sci., 1986. Supervised by Sipser.

Study of the computational complexity of monotone Boolean circuits and monotone Boolean formulas. Monotone means that AND gates and OR gates are allowed, but NOT gates are not allowed. Detecting cliques in graphs is shown to require monotone circuits of exponential size. The amount of amplification of probabilistic Boolean formulas obtained recently by Valiant is shown to be best possible.

24. Boppana, R. B., and Hastad, J. Approximation properties of constant depth circuits. 1986, In preparation.

Two approximation properties of constant depth circuits are studied: how well a constant depth circuit can approximate the parity function, and algorithms for approximating the number of satisfying assignments of a constant depth circuit. The first result is that a constant depth circuit of not too large size can agree with the parity function on only slightly more than one-half of the inputs. The second result is that the number of satisfying assignments of a constant depth circuit can be approximated by a deterministic algorithm using little space.

25. Boppana, R.B., and Hirschfeld, R. Pseudorandom generators and complexity classes. In *Advances in Computing Research*, Micali, S., Ed., 1986, 00. Submitted.

A notion of computational randomness and its implications for complexity theory. The probabilistic complexity class BPP is shown to lie within subexponential time if there exist pseudorandom bit generators whose outputs cannot be distinguished from random strings by polynomial-time statistical tests. Some of the results included here have been published previously without proofs; this paper collects the known results, proves them, and establishes some new related results.

26. Boppana, R.B., and Lagarias, J.C. One-way functions and circuit complexity. In *ACM Structure in Complexity Theory Conf.*, 1986, 00. To appear.

A function f can be checked if some polynomial size circuit with inputs x and y can determine if $f(x)=y$. A function f can be evaluated if a polynomial size circuit with input x can compute $f(x)$. Can all functions (in a certain class) which can be checked also be evaluated? Relations between this question and the existence of one-way functions are established.

27. Breazu-Tannen, V. and Meyer, A.R. Lambda calculus with constrained types (Extended abstract). In *Logics of Programs, Lect. Notes in Comp. Sci.* **193**, Parikh, Rohit, Ed., Springer-Verlag, 1985, 23-40.

> Motivated by domain equations, we consider types satisfying arbitrary equational constraints thus generalizing a range of situations with the finitely typed case at one extreme and the type-free case at the other. The abstract model theory of the $\beta\eta$ type-free case is generalized. We investigate the relation between λ-calculus with constrained types and cartesian closed categories (ccc's) at proof-theoretic and model-theoretic levels. We find an adjoint equivalence between the category of typed λ-algebras and that of ccc's such that the subcategories of typed λ-models and concrete ccc's correspond to each other. All results are parameterized by an arbitrary set of higher-order constants and an arbitrary set of higher-order equations.

28. Bruce, K., Meyer, A.R., and Mitchell, J.C. The semantics of second-order polymorphic lambda calculus. *Information and Control* (1986), 00. To appear. An earlier version authored by Bruce and Meyer appeared in Springer Lecture Notes in Computer Science, vol. 173.

> The second-order polymorphic lambda-calculus formally models dependent polymorphic types in programming languages. Types are values -- though not quite 'first-class' values -- in such languages. In particular, a function which takes types as arguments may be applied to its own type. This kind of indirect self-application is inconsistent with a naive interpretation of functions and types, so the mathematical meaning of polymorphism needs explanation. This paper explains what a mathematical model of polymorphism should be, exhibits some models, and proves that the proposed notion of model implies a completeness theorem for the standard inference rules for polymorphic equalities.

29. Bui, T., Chaudhuri, S. Leighton, F.T., Sipser, M. Graph bisection algorithms with good average case behavior. *Combinatorica* **0** (1986), 00. To appear.

> Algorithms for graph bisection which almost always find the optimal bisection of the graph, along with a proof that the proposed bisection is, in fact, optimal. The algorithm works for large classes of random graphs with smaller than expected bisection, and can be demonstrated to work much better than standard approaches on these graphs.

30. Buss, J. Relativized Alternation and Space-Bounded Computation. Ph.D. Th., MIT Dept. of Math., 1986. To appear, September, '86. Supervised by Sipser.

31. Chor, B. Two Issues in Public-Key Cryptography. MIT Press, 1986. ACM Distinguished Computer Science Ph.D. dissertation.

32. Chor, B., Friedmann, J., Goldreich, O., Hastad, J., Rudich, S., and Smolensky, R. The bit extraction problem, or t-resilient functions. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 396-407.

> Extracting unbiased independent bits from a string containing some random bits and some predetermined bits. Specifically, let n, m and t be arbitrary integers, and let f be a function from n-bit strings to m-bit strings. An adversary, knowing the function f, sets t of the n input bits, while the rest (n-t input bits) are chosen at random. The adversary tries to prevent the outcome of f from being uniformly distributed. The question addressed is for what values of n, m and t does the adversary necessarily fail in biasing the outcome of f. The problem has applications to distributed computing.

33. Chor, B., Goldwasser, S., Micali, S., Awerbuch, B. Verifiable secret sharing and simultaneous broadcast. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 383-395.

> Two new primitives: verifiable secret sharing in the area of cryptographic protocols and simultaneous broadcast in the field of fault tolerant computing. Many problems, such as distributed coin flipping and "uninfluenced" voting, can be reduced to the solution of our primitive problems.

34. Chor, B., Leiserson, C., Rivest, R., Shearer, J. An application of number theory to the organization of raster-graphics memory. *J. ACM* **33** (1986), 86-104.

35. Chor, B., Dwork, C. Randomized algorithms for distributed agreement. 1986, Survey paper, submitted for journal publication.

36. Chor, B., Goldreich, O. Unbiased bits from sources of weak randomness and probabilistic communication complexity. In *26^{th} IEEE Symp. Found. of Comp. Sci.*, 1985, 429-442.

> A new model for weak random physical sources is presented. The model strictly generalizes a previous model of Santha and Vazirani. The sources considered output strings according to probability distributions in which no single string is too probable. The new model provides allows elegant and fruitful treatment of problems studied previously, such as:
>
> - Extracting almost perfect bits from weak sources of randomness.
> - Probabilistic Communication Complexity.
> - Robustness of BPP with respect to sources of weak randomness.

37. Chor, B., Goldreich, O. An improved parallel algorithm for integer GCD. *Algorithmica* **0** (1986), 00. Submitted.

> A new parallel algorithm for computing the greatest common divisor of two n bit integers. The run time of the algorithm in terms of bit operations is $O(n \log n)$, using $n^{1+\epsilon}$ processors.

38. Chor, B., Goldreich, O. On the power of two-point based sampling. *J. Complexity* **0** (1986), 00. Submitted.

> A new sampling technique and some of its properties. The technique consists of picking two elements at random and deterministically generating from them a long sequence of pairwise independent elements.

39. Chor, B., Goldreich, O., Goldwasser, S. The bit security of modular squaring given partial factorization of the modulos. In *Advances in Cryptology: Proceedings of Crypto85*, Williams, H.C., Ed., Springer Verlag, 1986, 448-457.

> The difficulty of guessing a bit of the square root of a quadratic residue modulo N, remains as intractable as factoring, even if all but two large prime factors of N are also known.

40. Chor, B., Merritt, M., Shmoys, D.B. Simple constant-time consensus protocols in realistic failure models. In *4^{th} ACM Symp. Principles of Distributed Computing*, Strong, R.H., Ed., 1985, 152-162.

> Protocols are given for achieving consensus in a potentially faulty network. The algorithms all tolerate a linear fraction of faulty processors and terminate within expected constant time. Several varieties of synchronous and asynchronous omission models are considered.

41. Chung, F.R.K., Leighton, F.T., Rosenberg, A. Embedding graphs in books: a layout problem with applications to VLSI design. *SIAM J. Algebraic and Discrete Methods* **0** (1986), 00. To appear.

> Techniques and bounds for embedding graphs in books, a combinatorial problem that arises in a variety of applications, including VLSI design.

42. Coffman, E.G., Kadota, T., Leighton, F.T., Shepp, L. Stochastic analysis of storage fragmentation. In *Int'l. Sem. Teletraffic Analysis and Computer Performance Evaluation*, North Holland, 1986, 00. To appear.

> A summary of most of the known research on the design and analysis of algorithms for dynamic storage allocation which work well in a stochastic sense. Dramatic progress has recently been made in this area.

43. Coffman, E., Leighton, F.T. A provably efficient algorithm for dynamic storage allocation. In *18th ACM Symp. Theory of Computing*, 1986, 77-90.

Description and analysis of a fast and surprisingly efficient algorithm for on-line dynamic storage allocation. Given any distribution of file sizes and a Poisson arrival/departure process for the files, the algorithm will waste only $O(N^{1/2}\log^{3/4}N)$ space with high probability where N is the expected used space. The algorithm is close to Best Fit in structure but superior in performance.

44. Cormen, T.H., Leiserson, C.E. A hyperconcentrator switch for routing bit-serial messages. In *15th IEEE Conf. Parallel Processing*, Penn. State Univ., 1986, 00. To appear.

In highly parallel message routing networks, it is sometimes desirable to concentrate relatively few messages on many wires onto fewer wires. We have designed a hyperconcentrator switch for this purpose which is capable of concentrating bit-serial messages quickly. The switch has a highly regular layout using ratioed nMOS, and the same architecture works for domino CMOS as well. Our circuit takes advantage of the relatively fast performance of large fan-in NOR gates in these technologies. A signal incurs exactly $2 \lg n$ gate delays through the switch.

45. Elias, Peter. Interval and recency-rank source coding: two on-line adaptive variable-length schemes. *IEEE Trans. on Information Theory* (1986). To appear, also available as MIT/LCS/TM-301.

46. Even, S., Goldreich, O., Shamir, A. On the security of ping-pong protocols when implemented using RSA. In *Advances in Cryptology: Proceedings of Crypto85*, Williams, H.C., Ed., Springer Verlag, 1986, 58-72.

The obvious RSA properties, such as "multiplicativity", do not endanger the security of ping-pong protocols. Namely, if a ping-pong protocol is secure in general, then its implementation using an "ideal RSA" is also secure.

47. Feldman, P., Friedman, J., Pippinger, N. Non-blocking networks. In *18th ACM Symp. Theory of Computing*, 1986, 247-253.

48. Feldman, P., Micali, S. Byzantine agreement in constant expected time (and trusting no one). In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 267-276.

A novel cryptographic algorithm for Byzantine agreement in a network with $t{=}O(n)$ faulty processors and in the *most adversarial* setting. The algorithm requires, once and for all, $O(t)$ rounds of preprocessing. Afterwards it reaches *each* individual Byzantine agreement in *constant* expected time. The solution *does not* make use of any trusted party.

49. Gilmore, P.C., Lawler, E.L., Shmoys, D.B. Well-solvable cases of the traveling salesman problem. In *The Traveling Salesman Problem*, Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B., Eds., J. Wiley and Sons, 1985, 87-143. Discrete Mathematics.

Many special cases of the traveling salesman problem that can be solved in polynomial time are given. In addition, a "theory" of well-solvable special cases is presented that unifies several of the previously known results.

50. Goldberg, A.V. A new max-flow algorithm. Tech. Rep. MIT/LCS/TM-291, MIT Lab. for Comp. Sci., November, 1985.

A max-flow algorithm that abandons Dinic's level network method used by all other efficient algorithms for the problem. A new method is introduced that results in a better parallel and distributed algorithms.

51. Goldberg, A.V., and Tarjan, R.E. A new approach to the maximum flow problem. In *18th ACM Symp. Theory of Computing*, 1986, 136-146.

A generic algorithm for the max-flow problem and specific implementations of the generic algorithm that give the best known sequential, parallel, and distributed max-flow algorithms.

52. Goldreich, O., Micali, S., Wigderson, A. Proofs that yield nothing but their validity and a methodology for cryptographic protocol design. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, 00. To appear.

Demonstration of the generality and applicability of zero-knowledge proofs, a fundamental notion introduced by Goldwasser, Micali and Rackoff. Zero-knowledge proofs are probabilistic, interactive protocols that efficiently demonstrate membership of words in a language without conveying any additional knowledge. Zero-knowledge proofs were known for some number theoretic languages in NP∩Co-NP. Zero-knowledge proofs for both graph isomorphism and graph non-isomorphism (which`is not known to be in NP) are presented. Under the assumption that encryption functions exist, it is shown that *all* languages in NP possess zero-knowledge proofs. This result is used to efficiently transform cryptographic protocols that are correct with respect to a very weak adversary into protocols correct in the most adversarial scenario.

53. Goldreich, O., Shrira, L. The effect of link failures on computations in asynchronous rings. In *5th ACM Symp. Principles of Distributed Computing*, J. Halpern, Ed., 1986, 00. To appear.

The message complexity of distributed computations on rings of asynchronous processors. In such computations, each processor has an initial local value and the task is to compute some predetermined function of all local values. Our work deviates from the traditional approach to complexity of ring computations in that we consider the effect of link failures. We show that the complexity of any non-trivial function is $\Theta(n \log n)$ messages when n, the number of processors, is *a priori* known; and is $\Theta(n^2)$ when n is not known. Interestingly, these tight bounds do not depend on whether the identity of a leader is *a priori* known before the computation starts. These results stand in sharp contrast to the situation in an asynchronous ring with no link failures.

54. Goldreich, O., Goldwasser, S., Micali, S. The cryptographic applications of random functions (extended abstract). In *Advances in Cryptology: Proc. Crypto 84, Lect. Notes in Comp. Sci.* **196**, Blackley, G.R. and Chaum, D., Ed., Springer-Verlag, 1985, 276-288.

55. Goldreich, O., Goldwasser, S., Micali, S. How to construct random functions. *J. ACM* **0** (1986), 00. To appear, October, 86.

56. Goldwasser, S., Kilian, J. Almost all primes can be quickly certified. In *18th ACM Symp. Theory of Computation*, 1986, 316-329.

How to generate short certificates of primality? Pratt showed that all primes have short certificates, but his proof does not yield a computationally useful algorithm. We exhibit an algorithm which will generate short certificates of primality in expected polynomial time for all but a vanishingly small fraction of the prime numbers. Furthermore, under a strongly believed hypothesis about the distribution of prime numbers, our algorithm will quickly generate short certificates for all primes.

Our approach differs radically from previous work done in probabilistic primality testing. Previous probabilistic tests may be thought of as compositeness provers, and cannot assert primality with 100% confidence. Using both classical and recent results on the theory of elliptic curves, we have developed a "Las Vegas" algorithm which uses randomness, but nevertheless is always correct.

Our work has numerous theoretical consequences. One implication is that there is an infinite set of primes that can be recognized in expected polynomial time. Another is that generating certified primes can be done in expected polynomial time. These results have generated excitement among researchers in the field and even in the press, and have already led to implemented algorithms which have yielded new large primes.

57. Goldwasser, S., Micali, S., Rackoff, C. The knowledge complexity of interactive proof systems. *J. ACM* **0** (1986), 00. Submitted.

58. Goldwasser, S., Micali, S., Rivest, R. A signature scheme which is secure against chosen cyphertext attack. *J. ACM* **0** (1986), 00. Submitted.

59. Goldwasser, S., Sipser, M. Interactive proof systems: public vs. private coins. In *18th ACM Symp. Theory of Computation*, 1986, 59-68.

60. Greenberg, R.I. and Leiserson, C.E. Randomized routing on fat-trees. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 241-249. Also available as MIT/LCS/TM-307.

Fat-trees are a class of routing networks for hardware-efficient parallel computation. We present a randomized algorithm for routing messages on a fat-tree. The quality of the algorithm is measured in terms of the load factor of a set of messages to be routed, which is a lower bound on the time required to deliver the messages. We show that if a set of messages has load factor $\lambda = \Omega(\lg n \lg\lg n)$ on a fat-tree with n processors, the number of delivery cycles (routing attempts) that the algorithm requires is $O(\lambda)$ with probability $1-O(1/n)$. The best previous bound was $O(\lambda \lg n)$ for the off-line problem where switch settings can be determined in advance. In a VLSI-like model where hardware cost is equated with physical volume, we use the routing algorithm to demonstrate that fat-trees are universal routing networks in the sense that any routing network can be efficiently simulated by a fat-tree of comparable hardware cost.

61. Halpern, J., Loui, M., Meyer, A. and Weise, D. On time versus space III. *Math. Systems Theory* **19** (1986), 13-28.

An ingenious general program transformation to save computation space (at the cost of repeatedly recomputing subresults) which is applicable to *any* program whose space requirements grow proportionally to its time requirements. (This is old result of theoretical interest which has taken awhile to write and publish because of its subtlety.)

62. Hastad, J. On using RSA with low exponent in a public key network. *SIAM J. Comput.* **0** (1986), 00. To appear.

Encrypting linearly related messages by RSA with a low exponent is insecure. The main tool is an algorithm solving simultaneous modular equations of low degree.

63. Hastad, J. Almost optimal lower bounds for small depth circuits. In *18th ACM Symp. Theory of Computing*, 1986, 6-20.

Almost optimal lower bounds for the size of small depth circuits computing functions like parity and majority. The main tool is a lemma which says that by assigning random values to a random subset of the variables it is possible with high probability to change a depth two circuit which is an AND of ORs to an OR of ANDs without increasing the size. For any constant k there is a function which has a linear size circuit of depth k but which has exponential size circuits when the depth is restricted to k-1.

64. Hastad, J. Computational Limitations for Small Depth Circuits. Ph.D. Th., MIT Dept. of Math., 1986. Supervised by Sipser.

Exponential lower bounds for small depth circuits computing certain functions. The relations to relativized complexity. The problem of inverting NC^0 permutations is shown to be P-hard.

65. Hastad, J., Just, B., Lagarias, J.C., and Schnorr, C.P. On finding integer relations among real numbers. *SIAM J. Comput.* **0** (1986), 00. Submitted.

A polynomial time algorithm which finds an integer relation among a set of real numbers. The model of computation is that real arithmetic can be done at unit cost.

66. Hastad, J., Leighton, F.T. Division in O(log N) depth using O($N^{1+\epsilon}$) processors. *Under revision for publication* (1985).

The main result decreases the number of processors needed for the Beame-Cook-Hoover division circuit from N^5 to slightly more than linear. The paper also extends the BCH method to find simple roots.

67. Heath, L.S. Algorithms for embedding graphs in books. Tech. Rep. TR 85-028, Univ. North Carolina, Chapel Hill, August, 1985.

A graph is embedded in a book by ordering its vertices on the spine (a line) and assigning each of its edges to a page so that, in any page, no two edges cross. Three algorithms for embedding graphs in books are presented. The first embeds any planar graph in a book of seven pages. The second embeds any trivalent planar graph in a book of two pages. The third exhibits a tradeoff for the class of outerplanar graphs between the number of pages in an embedding and the "width" of the pages.

68. Heath, L.S. Embedding outerplanar graphs in small books. *SIAM J. Algebraic and Discrete Methods* **0** (1985), 00. To appear.

An O(n log n) time algorithm for embedding an n-vertex outerplanar graph in a two-page book with O(d log n) pagewidth. This result exhibits a significant tradeoff between pagenumber and pagewidth for outerplanar graphs.

69. Heath, L.S. Embedding trivalent planar graphs in two pages. *SIAM J. Comput.* (1985). Submitted.

The relationship between the valence of a planar graph and the number of pages sufficient for embedding the graph in a book. We present a linear time algorithm for embedding a trivalent planar graph in a two-page book.

70. Heath, L.S., and Istrail, S. Surface-embeddable graphs can be embedded in a bounded number of pages. In *27^{th} IEEE Symp. Found. of Comp. Sci.*, 1986, 00. Submitted.

Any graph embedded in a surface (orientable or nonorientable) of positive genus g can be embedded in a book of O(g) pages. This compares to the lower bound we derive of O($g^{1/2}$) pages. Our result is constructive by an efficient algorithm. An important aspect of our constructions is a new decomposition of a graph embedded in a surface into planar and nonplanar parts.

71. Hirschfeld, R. Pseudorandom Generators and Complexity Classes. Master's Th., MIT Dept. of Electrical Eng. and Comp. Sci., 1986. Supervised by Micali.

See reference above to paper with same title by Boppana and Hirschfeld.

72. Hochbaum, D.S., Nishizeki, T., Shmoys, D.B. A better than 'best possible' algorithm to edge color multigraphs. *J. of Algorithms* **7** (1986), 79-104.

An algorithm that edge colors any multigraph with at most (9*OPT+6)/8 colors is presented, where OPT is the minimum number of colors needed to color the multigraph. The core of the algorithm is a procedure which colors a certain wide class of multigraphs optimally within polynomial time.

73. Hochbaum, D.S., Shmoys, D.B. An O(V^2) algorithm for the planar 3-cut problem. *SIAM J. Algebraic and Discrete Methods* **6** (1985), 707-712.

A simple algorithm based on breadth-first search for the problem of finding a minimum cardinality set of edges whose deletion disconnects a planar graph into three connected components.

74. Hochbaum, D.S., Shmoys, D.B. Using dual approximation algorithms for scheduling problems: theoretical and practical results. In *26th IEEE Symp. Found. of Comp. Sci.*, 1985, 79-89.

A polynomial approximation scheme is given for the problems of minimizing the completion time of a set of tasks to be scheduled on a set of identical parallel machines without precedence constraints or preemption. In other words, a family of polynomial-time algorithms is given where for each value $\epsilon > 0$, there is an algorithm in this family that produces a schedule taking time no more than $(1+\epsilon)$ times the optimum schedule. In addition, special cases are considered, giving extremely efficient algorithms that produced schedules guaranteed to finish within 20% and 16 2/3% more than the optimal schedule.

75. Hochbaum, D.S., Shmoys, D.B. A packing problem you can almost solve by sitting on your suitcase. *SIAM J. of Algebraic and Discrete Methods* **7** (1986), 247-257.

A new type of approximation algorithm is introduced: instead of searching for suboptimal, feasible solutions where the degree of suboptimality is bounded, a dual approximation algorithm seeks superoptimal, infeasible solutions where the degree of infeasibility is bounded. Using this notion, we show a trade-off between the 'degree of NP-completeness' of a certain packing problem and the performance of the dual approximation algorithm.

76. Hochbaum, D.S., Shmoys, D.B. A unified approach to approximation algorithms for bottleneck problems. *J. ACM* **33** (1986), 00. To appear.

A general technique based on the concept of a power of graph is presented that provides approximation algorithms for a wide range of problems from routing, communication network design, and location theory. Many of the algorithms presented here are the best possible in the sense that they deliver solutions that are guaranteed to be within a factor of k of the optimum, and if there existed a polynomial-time algorithm with a superior performance guarantee, then P would equal NP.

77. Kaklamanis, Christos. A Special Case of First-Order Strictness Analysis. Bachelor's Th., MIT Dept. of Electrical Engin. and Comp. Sci., 1986. Supervised by Meyer.

Meyer showed that strictness analysis of first-order declarations was of exponential complexity. Here a coarser analysis, essentially one in which *all* functions, even conditionals, are treated as strict in all arguments, is shown to be possible in a linear time for declarations of first-order functions.

78. Kaliski, B.S. Wyner's analog encryption scheme: results of a simulation. In *Advances in Cryptology: Proc. Crypto 84, Lect. Notes in Comp. Sci.* **196**, Blakley, G.R., and Chaum, D., Eds., Springer-Verlag, 1985, 83-94.

Results of a simulation of an analog encryption scheme. The scheme, introduced in 1979 by Aaron Wyner of Bell Telephone Laboratories, provides secure, accurate scrambling of speech waveforms while conforming to the bandlimitedness of a telephone channel. The simulation confirms the scheme's theoretical properties, based on numerical measures and on listening to encrypted and decrypted waveforms.

79. Kaliski, B.S., Rivest, R.L., and Sherman, A.T. Is DES a pure cipher? (Results of more cycling experiments on DES). In *Proc. of Crypto 85*, Springer-Verlag, 1985, 00. To appear.

Eight cycling experiments on the Data Encryption Standard (DES) were performed to see if DES has certain algebraic weaknesses. Except for a "weak key" experiment, our results were consistent with the hypothesis that DES acts like a set of randomly-chosen permutations. The weak key experiment produced a short cycle, the consequence of hitting a fixed point for each weak key.

80. Kaliski, B.S., Rivest, R.L., and Sherman, A.T. Is the Data Encryption Standard a group? In *Advances in Cryptology: Eurocrypt 85*, Pichler, F., Ed., Springer-Verlag, 1986, 81-95.

Two statistical tests for determining if an indexed set of permutations acting on a finite message space forms a group under functional composition. Each test yields a known-plaintext attack against any finite, deterministic cryptosystem that generates a small group. Using a combination of software and special-purpose hardware, we applied a "cycling test" to the Data Encryption Standard (DES). Our experiments show, with a high degree of confidence, that DES is not a group.

81. Karloff, H.J., Shmoys, D.B. Efficient parallel algorithms for edge coloring problems. *J. of Algorithms* **7** (1986), 00. To appear.

Efficient parallel algorithms to edge color simple graphs of maximum degree d with $d+1$ colors are presented; for graphs of maximum degree bounded by a poly-log function of the input size, these are NC algorithms, requiring poly-log time using polynomially many processors. In addition, a RNC algorithm is given to edge color any simple simple graph with $d+o(d)$ colors.

82. Karp, R.M., Leighton, F.T., Rivest, R., Thompson, C., Vazirani, U., Vazirani, V. Global wire routing in two-dimensional arrays. *Algorithmica* **0** (1986), 00. Submitted.

Results on channel width when routing gate arrays, including NP-completeness, approximation algorithms, and tight worst case analysis. The results extend to a wide class of integer programming problems, and have recently been improved in a very nice paper by Raghavan and Thompson.

83. Klein, P., Reif, J. An Efficient Parallel Algorithm for Planarity. Master's Th., MIT Dept. of Electrical Eng. and Comp. Sci., 1986. In preparation.

A new parallel algorithm for finding a planar embedding of a graph (or reporting that none exists). The new algorithm is almost as efficient as theoretically possible. In contrast, the previous best algorithm for planarity used many more processors to achieve the same time bound. The most significant aspect of the new parallel algorithm is its use of a sophisticated data structure for representing sets of embeddings, the PQ-tree of Booth and Lueker. This data structure was developed by Booth and Lueker and used in a sequential algorithm for planarity; however, no parallel algorithms for the data structure were known previously. This is joint work with Reif.

84. Klein, P., Reif, J. Parallel time O(log n) acceptance of deterministic CFL's. *SIAM J. Comput.* **0** (1985), 00. Submitted. Also available as TR-05-84, Center for Research in Computing Technology, Harvard University.

85. Lagarias, J.C., and Hastad, J. Simultaneous diophantine approximation of rationals by rationals. *J. Number Theory* **0** (1986), 00. To appear.

Study of the number of good diophantine approximations to a vector of rational numbers. The results are crucial to the the analysis of Shamir's attack on the basic Merkle-Hellman knapsack crypto system.

86. Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D., (Eds.). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. J. Wiley and Sons, 1985. Discrete Mathematics.

A graduate textbook in combinatorial optimization. All of the aspects of this field, such as computational complexity, polyhedral combinatorics, and probabilistic, worst-case and empirical performance of heuristic procedures, are discussed from the point of view of the TSP.

87. Leighton, F.T. An Introduction to the Theory of Networks, Parallel Computation and VLSI Design. In progress, 1986.

A textbook based on the course taught 1983-1986 on theory of parallel computation and VLSI design.

88. Leighton, F.T., Leiserson, C.E. Wafer-scale integration of systolic arrays. *IEEE Trans. on Computers* **C-34** (1985), 448-461.

Several new and nearly optimal algorithms for integrating one and two dimensional arrays of processors on a wafer containing faults. The algorithms are shown to work with very high probability when cell failures are independent, minimizing maximum wire length and average channel width.

89. Leighton, F.T., Leiserson, C.E. Algorithms for integrating wafer-scale systolic arrays. In *VLSI Circuit and Architecture Design*, Swartzlander, E., Ed., Marcel-Dekker, 1986, 00. To appear.

Summary of most of the known techniques for integrating one and two-dimensional arrays around faults on a wafer. The techniques are analyzed in terms of maximum wire length and average channel width necessary for a typical wafer with randomly located faults.

90. Leighton, F.T., Leiserson, C.E. A survey of algorithms for integrating wafer-scale systolic arrays. In *IFIP Workshop on Wafer-Scale Integration at Grenoble*, Saucier, G., Truihle, J., Eds., North Holland, 1986, 00. To appear.

Essentially the same as the paper in the VLSI Systems book. The paper summarizes most of the known techniques for integrating one and two-dimensional arrays around faults on a wafer. The techniques are analyzed in terms of maximum wire length and average channel width necessary for a typical wafer with randomly located faults.

91. Leighton, F.T., Rivest, R. Estimating a probability using finite memory. *IEEE Trans. on Information Theory* **0** (1986), 00. To appear.

Tight bounds on the problem of estimating the mean of a Bernoulli random variable with a finite state automaton.

92. Leighton, F.T., Rosenberg, A. Three-dimensional circuit layouts. *SIAM J. Comput.* **0** (1986), 00. To appear.

The paper describes nearly optimal techniques for converting two-dimensional layouts into more efficient multilayer layouts. Both the one-active-layer model and many-active-layer model are considered.

93. Leighton, F.T., Shor, P. Tight bounds for minimax grid matching, with applications to the average case analysis of algorithms. In *18^{th} ACM Symp. Theory of Computing*, 1986, 91-103.

Solution to the minimax grid matching problem, a fundamental combinatorial problem associated with the average case analysis of algorithms. The results have application to a variety of problems, including bin packing, dynamic allocation, wafer-scale integration of systolic arrays, testing pseudorandom number generators, planar discrepancy and matching problems, and mathematical statistics.

94. Leiserson, C.E., and Maggs, B.M. Communication-efficient parallel graph algorithms. In *1986 Int'l. Conf. Parallel Processing*, 1986, 00. To appear.

95. Levin, L.A. Average case complete problems. *SIAM J. Comput.* **15** (1986), 285-6.

Many interesting combinatorial problems were found to be NP-complete. Since there is little hope to solve them fast in the worst case, researchers look for algorithms which are fast just "on average". This matter is sensitive to the choice of a particular NP-complete problem and a probability distribution of its instances. Some of these tasks were easy and some not. But one needs a way to distinguish the "difficult on average" problems. Such negative results could not

3. Many programming languages allow procedures which can take themselves as arguments. This kind of type-violation yields a contradiction in three lines, e.g., the function P declared as follows:

function $P(f)$ ::= **if** $f(f)=0$ **then** 1 **else** 0 **fi**

By definition, $P(f)$ is not equal to $f(f)$ for all functions f. So let f be P. Then $P(P)$ is not equal to $P(P)$! Why isn't this contradiction threatening to Computer Science?

101. Meyer, A.R., and Reinhold, M.B. 'Type' is not a type: preliminary report. In *13th ACM Symp. Principles of Programming Languages*, 1986, 287-295.

A function has a *dependent type* when the type of its result depends upon the value of its argument. Dependent types originated in the type theory of intuitionistic mathematics and have reappeared independently in programming languages such as CLU, Pebble, and Russell. Some of these languages make the assumption that there exists a *type of all types* which is its own type as well as the type of all other types. Girard proved in 1972 that this approach is inconsistent from the perspective of intuitionistic logic. We apply Girard's techniques to establish that the type-of-all-types assumption creates serious pathologies from a programming perspective: a system using this assumption is not normalizing, term equality is undecidable, and the resulting theory fails to be a conservative extension of the theory of the underlying base types, so that classical reasoning about programs is not sound.

102. Meyer, A.R. and M. Wand. Continuation semantics in typed lambda-calculi (Summary). In *Logics of Programs, Lect. Notes in Comp. Sci.* **193**, Parikh, Rohit, Ed., Springer-Verlag, 1985, 219-224.

103. Micali, S. Knowledge and efficient computation. In *1st ACM Symp. Theoretical Aspects of Reasoning About Knowledge*, Halpern, J., Ed., 1986, 353-362.

104. Micali, S., Fischer, M., Rackoff, C., Wittenberg, D. An oblivious transfer protocol. 1986. In preparation.

105. Micali, S., Galil, Z., Gabow, H. An $O(E\ V \log V)$ algorithm for finding a maximal weighted matching in general graphs. *SIAM J. Comput.* **15** (1986), 120-130.

106. Micali, S., Rackoff, C., Sloan, B. The notion of security for probabilistic cryptosystems. *SIAM J. Comput.* **0** (1986), 00. Submitted.

Goldwasser and Micali's definitions of polynomial security and semantic security, and Yao's information theoretic definition of security are all shown to be equivalent.

107. Mitchell, J.C. and A.R. Meyer. Second-order logical relations (Extended Abstract). In *Logics of Programs, Lect. Notes in Comp. Sci.* **193**, Parikh, Rohit, Ed., Springer-Verlag, 1985, 225-236.

An effort to formalize the notion of "representation independence" of abstract data types in programming led J. Reynolds in 1974 to discover *logical relations* and prove a version of what can now be recognized as Statman's "Fundamental Theorem of Logical Relations". Logical relations are to arbitrary systems of finite types what homomorphisms (structure-preserving maps) are to ordinary (first-order) algebras. That is, they are a basic concept in the semantics of finite types, and their theory on finite types has been developed extensively by G. Plotkin and R. Statman. Reynolds' efforts to generalize the Fundamental Theorem to polymorphic types were stymied by the semantical problems of polymorphism. Using the notion of polymorphic model developed by Bruce, Mitchell, and Meyer, this paper demonstrates that the desired generalization of logical relations and proof of representation independence for polymorphic types is relatively straightforward.

108. Parikh, R., Chandra, A., J.Y. Halpern, and A.R. Meyer. Equations between regular terms and an application to process logic. *SIAM J. Comput.* **14** (1985), 935-94.

A syntactically simple problem of satisfiability of equations between regular expressions with only save "positive" efforts but may also be used in areas (like cryptography) where hardness of some problems is a frequent assumption. It is shown in the paper that the Tiling problem with uniform distribution of instances has no polynomial "on average" algorithm, unless every NP-problem with every simple probability distribution has it. It is interesting to try to prove similar statements for other NP-problems which resisted so far "average case" attacks.

96. Levin, L.A. One way functions and pseudorandom generators. *Combinatorica* (1986), 00. To appear.

Pseudorandom generators transform in polynomial time a short random "seed" into a long "pseudorandom" string. This string cannot be random in the classical sense of [Kolmogorov 65], but testing that requires an unrealistic amount of time (say, exhaustive search for the seed). Such pseudorandom generators were first discovered in [Blum, Micali 82] assuming that the function $a^x \; mod \; b$ is one-way, i.e., easy to compute, but hard to invert on a noticeable fraction of instances. In [Yao 82] this assumption was generalized to the existence of any one-way permutation. The permutation requirement is sufficient but still very strong. It is unlikely to be proven necessary, unless something crucial, like P=NP, is discovered. The paper, among other observations, proposes a weaker assumption about one-way functions, which is not only sufficient, but also necessary for the existence of pseudorandom generators.

97. Maggs, Bruce. Communication-Efficient Parallel Graph Algorithms. Master's Th., MIT Dept. of Electrical Eng. and Comp. Sci., 1986. Supervised by Leiserson.

98. Malitz, S.M. Measures of graphs on the reals. 1986, In preparation.

Let G be a (continuum-sized) undirected graph with vertices in the unit interval [0,1]. To each bijection of [0,1] onto itself (i.e. each relabeling of the vertices of G), there corresponds a subset of the unit square which is essentially the adjacency matrix of G under the given labeling. We establish some surprising relationships between the structure of G and measure-theoretic properties of the corresponding family of adjacency matrices.

99. Meyer, A.R. Floyd-Hoare logic determines semantics. In *IEEE Symp. Logic in Computer Science*, 1986, 44-48.

The first-order partial correctness assertions provable in Floyd-Hoare logic about an uninterpreted **while**-program scheme determine the scheme up to equivalence. This settles an open problem of Meyer and Halpern. The simple proof of this fact carries over to other partial correctness axiomatizations given in the literature for wider classes of ALGOL-like program schemes.

100. Meyer, A.R. Thirteen puzzles in programming logic. In *Proc. DDC Workshop on Formal Software Development: Combining Specification Methods, Lect. Notes in Comp. Sci.* **0**, Bjorner, D., Ed., 1986, 00. Nyborg, Denmark, (May, 1984). To appear.

Programming languages attach new computational meanings to familiar expressions. The computational meaning may have unexpected properties, and reasoning about it can be tricky. In this tutorial paper we describe a few puzzles which illustrate why reasoning about program behavior raises some logical challenges. Here are some samples:

1. Exhibit a declaration of a procedure E which takes no arguments and returns an integer value such that the conditional expression

$$\textbf{if } E{=}E \textbf{ then } 0 \textbf{ else } 1 \textbf{ fi}$$

evaluates to 1 in most programming languages.

2. Exhibit a simple context into which either of

$$(1+2) \quad \text{or} \quad (2+1)$$

can be substituted so that in essentially all programming languages the resulting substitutions yield different results.

free variables is shown to be undecidable. The result implies that an attractive simplification of the syntax of a system of modal logic used for reasoning about concurrent processes would destroy the positive decidability properties of the logic.

109. Phillips, C.A. Space-Efficient Algorithms for Computational Geometry. Master's Th., MIT Dept. of Electrical Engin. and Comp. Sci., 1985. Supervised by Leiserson. VLSI memo 85-270.

An algorithm for determining the connectivity of a set of N rectangles in the plane. Based upon a technique called scanning, this algorithm runs in $O(N \lg N)$ time and requires $O(W)$ primary memory space where W is the maximum number of rectangles to cross a vertical cut. Because we use a machine model that explicitly incorporates secondary memory, the new connected components algorithm avoids unexpected disk thrashing. We also introduce interval trees: a simple, sparse, data structure for manipulating sets of line segments.

110. Rivest, Ronald L. Network control by Bayesian broadcast. Tech. Rep. TM-287, MIT Lab. for Comp. Sci., September, 1985.

111. Sherman, A.T. Cryptology and VLSI (a two-part dissertation): I. Detecting and exploiting algebraic weaknesses in cryptosystems II. Algorithms for placing modules on a VLSI chip. Ph.D. Th., MIT Dept. of Electrical Engin. and Comp. Sci., 1986. Supervised by Rivest. To be completed in August.

Two independent parts. Part I explores relationships between algebraic and security properties of cryptosystems, focusing on finite, deterministic cryptosystems whose encryption transformations form a group under functional composition. Part II explores the problem of placing modules on a custom VLSI chip, focusing on the placement algorithms used in the MIT PI (Placement and Interconnect) System.

112. Sipser, M. Expanders, randomness, and time versus space. In *Conf. in Structure in Complexity Theory*, 1986, 325-329.

113. Trakhtenbrot, B.A. Selected Developments in Soviet Mathematical Cybernetics. Delphic Associates, Falls Church, VA, 1986. Monograph Series on Soviet Union.

A survey of thirty years' Soviet research in the areas of automata, combinatorial complexity, algorithmic complexity.

114. Trakhtenbrot, B.A. Logical relations in program semantics. In *Conf. Mathematical Logic and its Applications*, Plenum Press, 1986, 00. Druzhba, Bulgaria, Sept. 1986. Invited paper, to appear.

How logical relations may be used to characterize invariance properties of functionals, which provide the meaning (in the style of denotational semantics) of programming constructs.
In particular the following situations are considered:

- Schematological abstraction for languages which use a first order signature.
- The invariance of constructs in Algol-like languages with respect to memory locations.

115. Vitanyi, P., Awerbuch, B. Atomic register access by asynchronous hardware. In *27th IEEE Symp. Found. of Comp. Sci.*, 1986, 00. To appear.

The problem addressed is rooted in hardware design of concurrent registers access by asynchronous components and also in asynchronous interprocess communication. We want to construct multivalued registers which can be read and written asynchronously by many processors in a consistent fashion. Such a register is called an atomic register. We are not interested in a solution that requires one process to wait for another, *e.g.*, mutual exclusion, synchronization, execution rounds, and so on, because such a method slows the system to the speed of the slowest processor. It was known previously how to construct, starting from the most primitive asynchronous (hardware) components, an atomic register which can be read by one processor and written by one other processor. Using these atomic 1-reader, 1-writer registers, we construct atomic multireader, multiwriter registers.

5. Lectures (Annotated)

1. Barrington, D.A. Bounded width polynomial size branching programs recognize exactly those languages in NC^1. 1986, 18th ACM Symp. Theory of Computing; MIT Lab for Computer Science; IBM Research Center, San Jose; Mathematical Sciences Research Institute, Berkeley; U. Chicago; Wesleyan U.; U. Mass., Amherst; Amherst College; U. Toronto; U. Montreal.

2. Berger, B., Leighton, T. New bounds and algorithms for channel routing. 1985, Fall 1985 MIT VLSI Research Review.

Results of Master's Thesis.

3. Bloom, B. Lectures on denotational semantics of countable nondeterminism. 1986, a series of lectures, MIT LCS Theory of Computation Concurrency Seminar.

Papers of Pneuli, Milner, Apt and Plotkin, and de Bakker and Zucker, and related work by the lecturer.

4. Bloom, Bard, and Ronald L. Rivest. Abstract AI. 1985, Workshop in Machine Learning, Skytop, Pennsylvania.

5. Boppana, R.B. Lower bounds for monotone circuits. 1985, MIT Lab. for Comp. Sci.; Yale.

6. Boppana, R.B. Amplification of probabilistic Boolean formulas. 1985, 26th FOCS; Bell Communications Research.

7. Breazu-Tannen, V. Conservative extension situations in typed lambda calculi. 1986, Bell Labs; IBM Yorktown Heights; Brown U.; Cornell U.

Programming languages are modeled by various typed lambda calculi. Adding a new language feature corresponds to extending the calculus with new phrases and conversion rules. Are all old program phrases that were not provably equivalent in the old calculus still not provably equivalent in the extended calculus? (i.e. is the extension *conservative*?) We review several examples in which conservative extension holds and fails.

8. Chor, B., Goldwasser, S., Micali, S., Awerbuch, B. Verifiable secret sharing and simultaneous broadcast. 1985, Workshop on Security, MIT Endicott House.

9. Chor, B., Goldreich, O. Unbiased bits from sources of weak randomness and probabilistic communication complexity. 1985, MIT Lab. for Comp. Sci. (1985); MSRI, UC-Berkeley; IBM San-Jose Research Center; Marseille, workshop on Algorithms, Randomness and Complexity.

10. Chor, B., Goldreich, O. Efficient pseudo-random bits generators based on factoring/RSA. 1986, Marseille, workshop on Algorithms, Randomness and Complexity.

11. Chor, B., Goldreich, O. Unbiased bits from sources of weak randomness and probabilistic communication complexity. 1985-86, Harvard U.; U. Toronto; Tel-Aviv U.; IBM, Yorktown; IEEE Symp. Found. of Comp. Sci.

12. Cormen, T.H. and Leiserson, C.E. A hyperconcentrator switch for routing bit-serial messages. 1985, Darpa VLSI contractor's review, U. Utah.

13. Goldreich, O., Micali, S., Wigderson, A. Proofs that yield nothing but their validity, or all languages in NP have zero-knowledge proofs. 1986, MIT Lab for Comp. Sci.; Yale.

14. Goldreich, O., Micali, S., Wigderson, A. Methodological theorems for cryptographic protocol design. 1986, U. Toronto.

15. Goldwasser. Encryption and signatures in public key cryptography. 1985, Series of talks given at a course on cryptography held in Amsterdam.

16. Goldwasser, S., Kilian, J. A provably correct and probably fast primality test. 1985, NYU; Eighth Columbia Theory Day.

17. Goldwasser, S., Kilian, J. Almost all primes can be quickly certified. 1985, U. of Toronto; Marseille Workshop on Algorithmic Randomness and Complexity.

18. Goldwasser, S., Sipser, M. Interactive proof systems: public vs. private coins. 1986, Yale; Brown; ACM STOC.

19. Greenberg, R.I. Randomized routing on fat-trees. 1985, MIT; 1985 IEEE Symp. Found. of Comp. Sci.

20. Hastad J. Almost optimal lower bounds for small depth circuits. 1985, MIT; IBM, Yorktown Heights; CIRM, Marseille; MSRI, Berkeley; IBM, San Jose; UCLA; U. Toronto; Bell Communication Research.

21. Hastad, J. The bit-extraction problem or *t*-resilient functions. 1985, 26th IEEE Symp. Found. of Comp. Sci.

22. Hastad, J. On finding integer relations among real numbers. 1985, Conf. in Computational Number theory at Arcata.

23. Hastad, J. On using RSA with low exponent in a public key network. 1985, Crypto 1985, Santa Barbara, CA

24. Heath, L.S. Algorithms for embedding graphs in books. 1986, Minisymposium on Book Embeddings at the 3rd SIAM Conference on Discrete Mathematics.

> Two efficient algorithms for embedding graphs in books. The first algorithm embeds any trivalent planar graph in a two-page book. The second algorithm embeds a *d*-valent *n*-vertex outerplanar graph in a two-page book with $O(d \log n)$ pagewidth.

25. Kaliski, B.S. A pseudo-random bit generator based on elliptic logarithms. 1985, MIT Lab. for Comp. Sci.

> Recent research in cryptography has led to the construction of several pseudo-random bit generators -- programs producing bits as hard to predict as solving a hard problem. We present a new pseudo-random bit generator based on elliptic curves, and a new oracle proof method for simultaneous security of bits of a discrete logarithm in an arbitrary abelian group.

26. Kaliski, B.S. Lenstra's factoring algorithm using elliptic curves. 1985, Tufts.

> One of the most recent and interesting results in computational number theory is H.W. Lenstra's algorithm for factoring integers using the group structure of elliptic curves. We present relevant background in the study of elliptic curves and give a general outline of Lenstra's algorithm, concluding with an analysis of its running time and some applications.

27. Kaliski, Burton, Ronald L. Rivest, and Alan Sherman. Is DES a pure cipher? (Results of more cycling experiments on DES). 1985, CRYPTO 1985, Santa Barbara, CA.

28. Levin, L.A. Homogeneous measures and polynomial time invariants. 1986, French Math. Soc., Workshop on Algorithms, Randomness and Complexity, Marseille.

> The usual probability distributions share a remarkable feature. They are concentrated on strings which do not differ noticeably in any fundamental properties, except Kolmogorov complexity. The formalization of this statement distinguishes a promising class of "homogeneous" probability measures. In particular, it suggests a generalization of the

"average case NP-completeness" results and explains why they are so measure independent. It also demonstrates a sharp difference between pseudo-random strings and the objects known before.

Some characteristics of strings, like length, change dramatically when the string undergoes trivial transformations (like padding, doubling, etc.). More fundamental characteristics f, called p-invariants, have $f(t(x))$-$f(x)$ bounded by a constant for any transformation t computable and invertible in polynomial time. Obvious examples are $l(x)$=log log $|x|$ and Kolmogorov complexity K(x). The function K is not computable and ignores such important issues as running time. However, only exponentially small fraction of strings may have other p-invariants, not determined by K, l. This holds for a broad class of probability distributions called homogeneous.

> *Theorem.* All measures with distribution functions computable in polynomial time are homogeneous.

In view of this theorem, it is quite a challenge to find an efficient way to construct objects which have invariants not determined by K, l. It is interesting that the pseudorandom generators achieve this goal, thus bringing new dimensions to the computational properties of strings.

29. Malitz, S.M. Measures of graphs on the reals. 1986, MIT; 304th AMS Conference, Johns Hopkins U.

30. Meyer, A.R. The complexity of flow-analysis: application of a fundamental theorem of denotational semantics. 1985, Dept. of Computer Science, U. Pisa, Italy.

Call-by-value strategy specifies that evaluation of the following functional expression would not terminate.

```
letrec
     f(x,y,z) ::= if x≤0 then y else f(x-1,z,y) fi;
         g(z) ::= g(z)+1
in f(2*2,1,g(0)) end
```

The source of the trouble is the divergent argument $g(0)$. In contrast, call-by-need strategy postpones evaluation of $g(0)$ until it is needed in evaluating the body of f -- which it isn't -- and ultimately terminates with the value 1. A function is *operationally strict* in its k^{th} argument if its *call-by-need* application to some arguments fails to terminate whenever evaluation of the actual k^{th} argument fails to terminate. It is OK to evaluate operationally strict arguments at "apply time" according to call-by-value strategy, even when call-by-need semantics is specified. The f above is *not* operationally strict in its third or second arguments, but is in its first.

Call-by-need yields a mathematically more attractive semantics, but call-by-value is generally more efficient. This motivates the question of analyzing declarations to determine which arguments are strict. We discuss the possibility of carrying out an abstract "strictness flow-analysis" of functional programs, pointing out undecidability and complexity results. The investigation provides a case study of how denotational semantics yields an algorithmic solution to an operationally specified program optimization problem.

In the finitely typed case without any interpreted functions (including conditional), the problem is decidable but of iterated exponential complexity. Strictness analysis for first-order declarations (like f above) turns out to be complete in deterministic exponential time.

31. Meyer, A.R. Floyd-Hoare logic determines semantics. 1986, Presentation of paper at IEEE Symp. on Logic in Computer Science, Cambridge, MA.

32. Meyer, A.R. Axiomatic semantics of programs by Floyd-Hoare logic. 1986, Yale.

33. Meyer, A.R. Types in programming, an overview. 1986, Invited Tutorial Lecture, 13th ACM Symp. Principles of Programming Languages.

34. Meyer, A.R. Logical puzzles in programming. 1986, Wesleyan U., Conn.

35. Rivest, Ronald L. The RSA public-key cryptosystem. 1985, Cryptography Workshop in Amsterdam (organized by David Chaum).

36. Sherman, A.T. Algorithms for placing modules on a custom VLSI chip. 1986, MIT VLSI Research Review.

> Algorithms used in the PI (Placement and Interconnect) System for placing modules on a custom VLSI chip. These algorithms are based on a top-down recursive min-cut approach that is sensitive to graph-theoretic and geometric concerns.

37. Shmoys, D.B. Efficient parallel algorithms for edge coloring problems. 1985, Mathematical Programming Society Symposium, Boston; Dept. Mathematics, MIT.

38. Shmoys, D.B. Using dual approximation algorithms for scheduling problems. 1985, IEEE Symp. Found. of Comp. Sci.; Princeton.

39. Shmoys, D.B. A packing problem you can almost solve by sitting on your suitcase. 1986, ORSA/TIMS meeting, Los Angeles.

40. Shmoys, D.B. Simple constant-time consensus protocols. 1986, ACM Conf. Principles of Distributed Computing; Univ. Washington, Seattle; Univ. Chicago.

41. Trakhtenbrot, B.A. A characterization theorem for program schemes. 1986, Columbia.

42. Trakhtenbrot, B.A. Survey on type theory. 1986, U. Tennessee.

43. Trakhtenbrot, B.A. On semantics of storage allocation. 1986, U. Tennessee.

44. Trakhtenbrot, B.A. Logical relations over FLAT and sequentiality of functionals. 1986, series of lectures at MIT Lab. for Comp. Sci. Theory of Computation Seminar on Types in Programming.

8 August 1986, Cambridge, MA