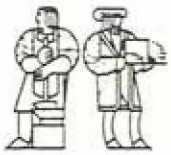
LABORATORY FOR
COMPUTER SCIENCE



MASSACHUSETTS
INSTITUTE OF
TECHNOLOGY

MIT/LCS/TM-404

THEORY OF COMPUTATION GROUP RESEARCH SUMMARY JUNE 1988 - JULY 1989

Theory of Computation Group

July 1989

545 TECHNOLOGY SQUARE, CAMBRIDGE, MASSACHUSETTS 02139

Theory of Computation Group Research Summary June 1988 - July 1989

M.I.T. Laboratory for Computer Science

July 31, 1989

Keywords: Algorithms, complexity, cryptology, distributed, logic of programs, pseudorandom, VLSI routing, semantics

June 1988 - June 1989

Academic Staff

B. Awerbuch	P. Elias	S. Goldwasser
F. Leighton	C. Leiserson	N. Lynch
A. Meyer	S. Micali	R. Rivest (Group Leader)
D. Shmoys	M. Sipser	É. Tardos

Visitors and Post-Docs

W. Aiello	S. Cosmadakis	S. Istrail
M. Kearns	L. Levin	N. Nisan
S. Safra	N. Shavit	A. Sherman

Graduate Students

R. Ashcroft	J. Aslam	M. Bellare	B. Berger
B. Bloom	A. Blum	T. Cormen	L. Cowen
C. Crépeau	C. Chan	A. Dhagat	R. Ehrenborg
B. Eisenberg	L. Fortnow	J. Fried	W. Goddard
S. Goldman	K. Graves	R. Greenberg	M. Grigni
C. Haibt	M. Hansen	R. Hirschfeld	A. Ishii
L. Jategaonkar	T. Jim	B. Kaliski	J. Kilian
S. Kipnis	P. Klein	R. Koch	D. Kravets
B. Maggs	S. Malitz	Y. Mansour	S. Mentzer
M. Newman	M. Papaefthymiou	J. Park	C. Phillips
S. Plotkin	S. Rao	J. Riecke	P. Rogaway
J. Rompel	A. Rudich	R. Schapire	E. Schwabe
L. Schulman	J. Siskind	R. Sloan	M. Soclof
C. Stein	M. Tuttle	P. Wang	J. Wein
S.-M. Wu	J. Yang		

Undergraduate Students

M. Ernst	J. Fernandez	R. Kaseguma
S. Trowbridge	P. Wang	D. Williamson

Support Staff

S. Bemus	C. Brownlie	D. Crowell
D. Grupp	B. Hubbard	S. Merritt

Contents

1 Introduction

The MIT Theory of Computation(TOC) group is one of the largest theoretical computer science research groups in the world. It includes faculty, students and visitors from both the Electrical Engineering & Computer Science department and the Applied Mathematics department.

The principal **research areas** investigated by members of the TOC Group are:

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

Group members were responsible for over one hundred and fifty publications and several dozen public lectures around the world during the past year. The *individual reports* by faculty and students in the next sections, and the *annotated reference and lecture lists* offer further descriptions of the year's activities.

The following **major research contributions** merit highlighting:

- Awerbuch, Mansour, and Shavit's polynomial solution to the basic network problem of "end to end communication".
- Awerbuch and Sipser's efficient implementation (constant time overhead) of the new notion of a "synchronizer for dynamic networks" implying that dynamic networks are as fast as static networks.
- Elias's geometric demonstration that reliable communication at a positive rate is possible over a channel which introduces a fraction $1/2 - \epsilon$ of errors, so long as the receiver is allowed to list $O(1/\epsilon^2)$ possible transmitted codewords rather than just one.
- Fortnow and Sipser's oracle collapsing the probabilistic polynomial time hierarchy.
- Koch proved a decade old conjecture about the expected throughput of the dilated butterfly switching network that has application to the optimal design of networks like that used in the BBN Butterfly Machine. (PhD Thesis)

- Leighton and Maggs developed highly efficient packet routing algorithms for a twin butterfly. The algorithms are the first fault tolerant routing algorithms for bounded degree switching networks, and appear to be superior to currently used algorithms even if there are no faults.

The following are special awards received:

- Lynch was chosen to deliver the keynote address at last summer's symposium on Principles of Distributed Computing.
- Meyer was chosen to present an Invited Lecture at the Third IEEE Symposium on Logic in Computer Science, July, 1988.
- Sipser was chosen as the principal lecturer in the American Mathematical Society conference on circuit complexity; to be held this August in Chicago.

2 Faculty Reports

Baruch Awerbuch

Awerbuch has been working on designing efficient and reliable distributed protocols, with emphasis on issues related to dynamic networks.

Awerbuch has put a great deal of effort into development of efficient compiler for dynamic network protocols. In [9] he used techniques of amortized analysis to improve the best known compiler for asynchronous protocols. Together with Sipser [16] he introduced a new concept of *Dynamic synchronizer* which allows us to apply static synchronous protocols in a dynamic asynchronous network. This protocol is very fast, requiring $O(1)$ time overhead, thus showing that dynamic asynchronous networks are as fast as static synchronous ones. Finally, working with Afek and Moriel (Tel-Aviv) [1], he has discovered a compiler whose overheads depend exclusively on the overheads of the original protocol.

Awerbuch also worked on many specific problems in dynamic networks. Together with Shavit and Mansour [15], Awerbuch has discovered the first polynomial solution to the end-to-end communication problem. This is one of the basic network problems; it was conjectured in [2] that it has no polynomial solution. Together with Goldberg (Stanford), Luby (ICSI,California) and Plotkin (Stanford) he has found a new technique [12] for removing randomness from distributed computing that has yielded fast deterministic algorithms for Maximal Independent Set, $\Delta + 1$ Coloring and Breadth First Search. Together with Kutten (IBM Yorktown) and Cidon (IBM Yorktown), he has discovered an efficient algorithm for maintaining a tree in a dynamic network. Together with Goldreich and Herzberg (Technion,Israel) [13] he has developed a quantitative framework for analyzing performance of broadcast protocols in dynamic networks.

Another area of Awerbuch's research has been distributed graph algorithms. Together with Bar-Noy (Stanford University, California), Linial (IBM Almaden Research Center, California), and Peleg (Stanford University, California) [11], he discovered new routing schemes that use only bounded space, have low communication overhead, can be constructed on-line, work for weighted graphs, and do not require changes in node identities. He discovered a new efficient BFS and Shortest Paths algorithm [10] which is efficient both in time and in communication. This algorithm has an interesting recursive structure. Together with Goldreich (The Technion, Israel), Peleg (Stanford University), and Vainish (The Technion, Israel) [14], Awerbuch studied performance of broadcast protocols in point-to-point networks.

Peter Elias

The paper on the zero-error capacity of a binary channel under jamming using list decoding, which was accepted for publication at the time of the last annual progress report, has since appeared [61]. Its appearance led to correspondence with Körner, who has been working on related topics with Marton and Simonyi. Their work arose from a paper on hashing by Fredman and Komlos [67]. They have published one paper [108] and submitted two more, which include new results relevant to zero-error capacity under list decoding.

The second paper mentioned in the last progress report, which has to do not with zero-error capacity but with error-correcting codes under list decoding, has appeared as a technical report and has been submitted for publication [62].

Current work explores iterative coding schemes. These schemes generate codes which differ from typical error-correcting block codes in that they are not guaranteed to correct *all* sets of less than k errors out of n for some integers k,n but only *most* such sets. Only codes with this property can be used to communicate at rates near channel capacity: as discussed in [61] and [62], the capacity of a channel subject to a jammer who can alter any k symbols out of n is significantly less than that of a channel in which bits are subject to statistically independent errors with probability k/n.

The first analysis of these codes appeared in [60]. It showed that they could be used to transmit without error at a positive rate, by using check symbols to correct each row of transmitted symbols, rows of check symbols to correct each column in a two-dimensional array, layers of check symbols to check preceding layers in a rectangular solid and so on. The fraction of the symbols used for checking is less than 1 in the limit if the sizes of successive dimensions increase, e.g. in a geometric series.

In [60] each order of check symbols is used only once and then discarded. That sufficed to show that communication at a positive rate is possible, but the proof gives a rate substantially below channel capacity. The rates of iterated codes come much closer to capacity when lower order check bits are used to make further corrections after each use of higher order check bits, and the process is continued until a stable state is reached. Since statistical independence disappears after such recycling, getting tight bounds on the amount of improvement is difficult. Both analysis and simulation are being used to explore this domain.

Shafi Goldwasser

Goldwasser's work focused on designing efficient digital signature schemes and on designing multi-party secure cryptographic protocols.

Don Beaver and Goldwasser[19] designed a protocol for n processors, a majority of which can be faulty, to compute any polynomial time function defined on the processors private inputs. The function is computed preserving privacy. Namely, no coalition of faulty processors can discover more about non-faulty processors inputs than implies by the function value. Moreover, the faulty processors can find out the function value "if and only if" the non-faulty processors find out the function value, in a strong probabilistic sense. This is the first solution in the case where the faults constitute more than a majority of the network processors.

Ben-Or, Kilian, Goldwasser and Wigderson[26] designed two extremely efficient user identification methods (using no modular multiplications and based on the difficulty of the NP-complete subset-sum problem). These schemes work in the two prover interactive proof model introduced by the same authors in '88. Namely, the prover (e.g Bank Card holder) is split into two agents, and the verifier (e.g the Bank teller machine) guarantees that the two agents can not transfer information to each other during the identification process.

Bellare and Goldwasser[20] introduced new paradigms for digital signatures and message authentications which are a complete departure from the digital signatures schemes based on Diffie-Hellman trapdoor function model or the recent digital signature scheme of Naor-Yung. The new scheme is based on the use of random functions and non-interactive zero-knowledge proofs.

Goldwasser has also been developing a monograph of lecture notes in cryptography, an outgrowth of her lectures in the MIT cryptography and cryptanalysis class. Goldwasser chaired the crypto88 conference held in Santa Barbara August 1988. She was a member of the STOC 1989 conference committee and together with Prof. Rivest wrote a survey article on cryptography for the handbook on computer science.

Tom Leighton

Together, Leighton and his students made solid progress on packet routing algorithms, fault tolerance in networks, and on graph embedding problems. At this point they are getting close to asymptotically optimal results that also appear to work well in reality. In fact, the highlight of the coming summer and fall will be to help design and lay out a multibutterfly network for Tom Knight's new machine. With a little luck, theory will be able to play an important role in the development of a state of the art machine. They are also working with Bill Dally and his students to see if theory can be helpful with the routing protocols on his new machine, and have been talking with Alan Baratz about the possibilities of implementing some of the new theory routing algorithms on the GF11 so that it can become a general purpose routing machine.

Another highlight of the coming year will be the new ACM Symposium on Parallel Algorithms and Architectures that Leighton has been helping to organize. The first meeting will be in Santa Fe in mid-June, and there should be a large contingent from MIT at the meeting. Papers to be presented range from theory to practice and the meeting should provide a good forum for interaction between people who think about parallel machines, those who build them, and those who use them. The 1990 meeting is in Crete, so now would be a good time to start thinking about submitting a paper!

Bruce Maggs, Satish Rao, Richard Koch, and Mark Newman are all getting their PhD's this year.

Leighton is continuing work on his book on parallel computation. He expects to have Volume I done by early next year. Anyway, he'll choose a publisher soon, and maybe that will help him stay home enough to get it done.

Charles E. Leiserson

Leiserson returned from a leave of absense at Thinking Machines Corporation January 1, 1989, where he worked on the design of a parallel computer. He was an invited speaker at the 25th Anniverary Symposium for Project MAC at MIT, and at the Decennial Caltech VLSI Conference. He served on the program committee for the IEEE Foundations of Computer

Science Conference. He also served on the first program committee for the ACM Symposium on Parallel Algorithms and Architectures.

Leiserson has spent much of his time in the past year working on a textbook entitled *Introduction to Algorithms*, coauthored with Cormen and Rivest. The textbook attempts to provide a rigorous, but elementary, introduction to the area of analysis of algorithms. It will be published jointly by MIT Press and McGraw-Hill later this year.

Two of Leiserson's Ph.D. student's completed their degrees in the past year. Serge Plotkin's thesis is entitled *Graph-Theoretic Techniques for Parallel, Distributed, and Sequential Computation.* Plotkin assumed a postdoctoral position at Stanford and will be an assistant professor there in the fall. Guy Blelloch's thesis is entitled *Scan Primitives and Parallel Vector Models.* Blelloch accepted an assistant professorship at Carnegie Mellon.

Three students completed their masters degrees under the supervision of Leiserson. Alexander Ishii's thesis is *A Digital Model for Level-Clocked Circuitry.* James Park's thesis is *Notes on Searching in Multidimensional Monotone Arrays.* Jeffrey Fried's thesis is *VLSI Processor Design for Communication Networks.*

Leiserson has also been supervising Cormen, Greenberg, Kipnis, Maggs, Phillips, and Papaefthymiou.

Nancy Lynch

Please see her entry under Theory of Distributed Systems.

Albert R. Meyer

Meyer's research has focused on semantics and logic of programming languages. During the past year, he worked on the following particular research topics.

Research Topics

- *Semantics of concurrency.* Meyer, with Bloom and Istrail (Wesleyan), question the foundations of Hoare's CSP and Milner's CCS theories of concurrency [34, 36, 35]. They propose a new notion of process equivalence and show it lies strictly between that of CSP and CCS. See the report of Bard Bloom for more complete discussion.

- *Semantics of Terminating Evaluation.* Research with Bloom, Riecke, and Cosmadakis (IBM Watson Lab.) on the general connection between operational and denotational semantics, focusing on repairing the mismatch between semantics in which expressions M and $\lambda x.M$ mean the same thing, even though evaluation of M diverges but evaluation of $\lambda x.M$ terminates immediately, cf. [127, 53, 37]. See the report of Jon Riecke.

- *Dataflow Semantics.* See the report of Arie Rudich.

- *Theory of Sequential Functions.* See the report of Trevor Jim.
- *Type-checking for records with inheritance.* See the report of Lalita Jategoankar.

Professional Activities

- Chairman, MIT Project MAC Twenty-fifth Anniversary Celebration, October, 1988.
- Conference Chairman, IEEE Symposium on Logic in Computer Science (LICS), Seattle, WA, May, 1989.
- Moderator for three Computer Science research email forums on (1) Types, (2) Concurrency, and (3) Logic.
- Member, Program Committee, Int'l. Symp. Logic at Botik, Pereslavl-Zalessky, USSR, July, 1989; "Kleene '90" Logic Symposium, Chaika, Bulgaria, June, 1990.
- Thesis Supervision:

  Ph.D. : Bard Bloom, expected September, '89.

  S.M. :

  1. Jon Riecke completed January, '89 [144].
  2. Lalita Jategoankar, expected September, '89 [96].
  3. Trevor Jim, expected January, '90.
  4. Arie Rudich, expected January, '90.

  B.S. :

  1. Michael Ernst, completed May, '89 [63].
  2. Arthur F. Lent, expected May, '90.
  3. Jeffrey Siegel, expected September, '89.

- Editorial Activity:

  Editor-in-Chief, *Information and Computation*; Managing Editor, *Annals of Pure and Applied Logic*; Editorial Board Member, *SIAM J. Computing, J. Computer and System Sciences, Theoretical Computer Science*, and *Advances in Applied Mathematics*; Advisory Editor, *Handbook of Logic in Computer Science* and *Handbook of Theoretical Computer Science*; Co-Editor, *Proc. Logic at Botik* [129]; MIT Press *Foundations of Computing Series* Co-Editor; MIT Press Editorial Board Member.

Silvio Micali

Micali's work focused on cryptography and zero-knowledge proofs. In particular, the following results were obtained:

1. Goldreich, Micali, and Wigderson had previously proven that all theorems in NP possess a zero-knowledge proof. Extending that work, [24] showed what can be efficiently verified can be proven in zero knowledge.

2. [131] constructed a very efficient "password" scheme. The person seeking identification is required to perform the equivalent of two multiplication modulo on an integer that is hard to factor. These special "passwords" are hard to compromise both by someone simply listening to the identification process and by the password verifier herself.

Ronald L. Rivest

Rivest's work focuses on the theoretical aspects of machine learning.

Rivest is continuing to work with Rob Schapire on problems related to the inference of finite automata. Their motivation has been the "artificial intelligence" problem faced by a robot placed in an unfamiliar environment with no *a priori* knowledge of its world. The goal of the robot is to learn the structure of its environment through systematic experimentation.

Schapire and Rivest [145] have been developed an interesting extension to Dana Angluin's finite automaton inference procedure [7]. The new algorithm can infer an automaton even when no "reset" is available (i.e., there is no means of bringing the automaton back to the start state), and can be used for inferring automata using either the global state-space representation or the diversity-based representation previously developed by Rivest and Schapire. The algorithm has been implemented and seems quite efficient in practice.

Together with Sally Goldman and Rob Schapire, Rivest has studied the problem of "learning a binary relation" [78]. In this problem, the entries of a matrix representing a binary relation are repeatedly probed. Before each probe, the "learner" must predict the value of the matrix entry about to be probed. The goal of the learner is to make as few prediction errors as possible. In order to model the natural "structure" that may be present in many binary relations, such structure being what gives the learner the leverage needed to make fewer than the maximum possible number of prediction errors, it is assumed that there are only a small number k of different row types. Algorithms are developed and analyzed that make a small number of errors in this case, and some interesting lower bounds (based on the existence of projective geometries) are proved.

Sally Goldman and Rivest [77] have also worked on the problem of efficiently implementing the "halving algorithm". The halving algorithm applies to situations (like the relation-learning problem of the last paragraph) where the learner must predict the classification of each instance before being told the true classification, and where the learner's goal is to minimize the number of prediction errors made. The halving algorithm (due to Barzdin and Freivalds [18], and refined by Littlestone [120]) predicts in according to the majority of the hypotheses consistent with all previous data; when a prediction error is made it therefore reduces by half the number of consistent hypotheses remaining. Based on a proposal by Manfred Warmuth, Goldman and Rivest have investigated the use of approximate counting scheme in order to implement approximations to the halving algorithm. This idea can be made to work out, and can be applied to problems such as learning a total order. (This

problem is then rather like the problem of sorting, where an adversary gets to pick which elements are to be compared next, and where you must predict the outcome before each comparison is made.)

Robert Sloan has finished up his Ph.D. under Rivest's supervision [155]; his thesis explores a number of fascinating issues and topics in machine learning theory, such as the effect of noisy data on learnability, techniques for learning a complicated concept reliably and usefully by learning it "gate by gate" (subconcept by subconcept), and methods for combining classical Bayesian inference with computational complexity considerations.

Linial, Mansour, and Rivest extended and presented their work showing that a finite Vapnik-Chervonenkis dimension is not a limitation for learning a concept class, if the size of the data sample used for learning can be adjusted dynamically as learning proceeds [117]. Intuitively, an algorithm can dynamically request more data when it discovers that the concept being learned is "complex".

Avrim Blum finished up his master's thesis [38] under Rivest's supervision, and the work he and Rivest have done on the complexity of training even very simple neural networks was presented at NIPS [40]. The basic result is that training a three-neuron neural network is NP-Complete.

Under Rivest's supervision, Nancy Perugini has experimentally examined the effect of training set data size on the efficacy of the "back-propagation" training algorithm for neural nets [138]. The results were not crisp, but some interesting pathologies were uncovered.

Together with Tom Cormen and Charles Leiserson, Rivest has worked on a introductory text and algorithms [52]. This text should be suitable for both introductory undergraduate and introductory graduate students; it should be out later this year.

David B. Shmoys

Shmoys has studied a wide range of questions in the design and analysis of efficient algorithms. He has continued his work in the design and analysis of approximation algorithms, as well as in the design of parallel algorithms for graph problems.

One of the most important algorithms used in the solution of the traveling salesman problem is a procedure due to Held and Karp [90] that produces an extremely tight lower bound on the value of the optimal solution. With Williamson [152], Shmoys considered this procedure as an approximation algorithm for the value of the optimal TSP solution. First, they showed that the algorithm has an important monotonicity property, in the sense that the bound delivered for a subset of the input is no more than for the entire input. This property makes it possible to prove that the procedure delivers a value at least 2/3 of the optimal value. Unlike Christofides' algorithm, which is the best known approximation algorithm for the problem (and guarantees identical performance), this bound is not known to be tight.

One major area of Shmoys' research is in the area of the theory of scheduling. Together with Lawler (UC/Berkeley), Lenstra (CWI) and Rinnooy Kan (Erasmus) [110], he wrote a survey article of the field. This survey was written as part of the preparation for a book on this subject by these authors.

With Hall (Sloan School/MIT) [87, 86], Shmoys has been considering a variety of approximation algorithms for scheduling problems. In particular, he has been studying the effect of precedence constraints and related timing constraints on the possibility of obtaining good approximate solutions. Hall and Shmoys [87] consider the problem of scheduling n jobs on a single machine, where each job j has a specified release date r_j before which is cannot be processed, a time p_j that specifies the amount of (continuous) processing required, and a deadline d_j. (For technical reasons, the deadlines are non-positive.) If the lateness of a job is the difference between the time that a job completes processing and its deadline, the aim is to find a schedule that minimizes the total lateness. For the variant of the problem without precedence constraints, a polynomial approximation scheme is obtained. For the problem with precedence constraints, they give an algorithm that delivers a solution that finishes within a factor of $4/3$ the optimal time (improving on the previous best algorithm that only came within a factor of 2). This represents an interesting breakthrough of a "factor of 2" barrier that is prevalent in approximation algorithms for precedence constrained scheduling problems. Also with Hall [86], Shmoys considers the natural generalization of the previous work to the case when there are parallel identical machines to do the processing. For this problem without precedence constraints, a polynomial approximation scheme was obtained. With precedence constraints, an algorithm that delivers a solution at most a factor of 2 more the optimal was obtained.

In the area of parallel graph algorithms, together with Goldberg (Stanford) Plotkin (Stanford) and Tardos [74], Shmoys considers the question of parallel algorithms for bipartite matching. By using techniques developed for general-purpose sequential algorithms for linear programming, so-called interior-point methods, they obtain an algorithm that requires only $O^*(\sqrt{m})$ steps on a polynomial number of processors, where m denotes the number of edges in the graph, and O^* indicates that lower order polylogarithmic factors have been ignored.

Michael Sipser

Sipser is continuing his work on lower bounds in complexity theory and the structure of complexity classes.

One of the important achievements of the past year was the construction of an oracle collapsing the probabilistic time hierarchy done jointly with Lance Fortnow [65]. Time hierarchies for deterministic and nondeterministic computation are among the earliest results proved in complexity theory. They show that if one is allowed a little more time then one can solve a larger class of problems. Oddly, this has never been established for probabilistic computation. It is possible that any problem solvable in probabilistic polynomial time can also be solved in probabilistic linear time, surprising though this would be. Our result shows why this problem has remained open. The existence of our oracle indicates that the techniques of recursive function theory which solved the previous cases are insufficient to solve this case. Fortnow is receiving his Ph.D. this year under Sipser's guidance.

Sipser also considered some problems in the theory of distributed computing. Together with Awerbuch he gave a method which facilitates the design of network protocols [16]. Using

this method one can first design a protocol to run on a static, synchronous network and then automatically convert it to run on a dynamic, asynchronous network. The former network model is a simpler one on which to conceive designs, whereas the latter model is more realistic.

Together with Ravi Boppana, Sipser has prepared a definitive survey on lower bounds on the circuit complexity of boolean functions [43]. This will appear in the forthcoming Handbook of Theoretical Computer Science. Sipser has been selected to be the principal speaker at an American Mathematical Society conference on circuit complexity. He will prepare a monograph of these lectures to be included in the AMS CBMS series.

Éva Tardos

Tardos has mainly been working on combinatorial optimization problems. Together with Goldberg and Plotkin from Stanford and Shmoys from MIT [74] developed an $O^*(\sqrt{m})$ time algorithm, where n and m denotes the number of nodes and edges of the input graph and an algorithm is said to run in $O^*(f(n))$ time if it runs in $O(f(n)\log^k(n))$ time for some constant k. In this paper interior-point methods for linear programming, developed in the context of sequential computation, are used to obtain a parallel algorithm for the bipartite matching problem. The results extend to the weighted bipartite matching problem and to the zero-one minimum-cost flow problem, yielding $O^*(\sqrt{m}\log C)$ algorithms, where it is assumed that the weights are integers in the range $[-C \ldots C]$ and $C > 1$. These results improve previous bounds on these problems and introduce interior-point methods to the context of parallel algorithm design.

In a joint paper with Plotkin from Stanford [142] Tardos gave an improved dual network simplex algorithm. A simplified version of Orlin's[135] strongly polynomial minimum-cost flow algorithm is developed, and it is shown how to convert it to a dual network simplex. The pivoting strategy leads to an $O(m^2 \log n)$ bound on the number of pivots, which is better by a factor of m compared to the previously best pivoting strategy due to Orlin [134]. Here n and m denotes the number of nodes and arcs in the input network.

In a joint paper with Frank from Budapest and Nishizeki, Saito and Suzuki from Tokyo Tardos has developed simple efficient algorithms for the routing problems around a rectangle. These algorithms find a routing with two or three layers for two-terminal nets specified on the sides of a rectangle. The minimum area routing problem is also solved. All algorithms run in linear time. The minimum area routing problem has previously been considered by LaPaugh and Gonzalez and Lee. The algorithms they developed run time $O(n^3)$ and $O(n)$ respectively. The simple linear time algorithm is based on a theorem of Okamura and Seymour and on a data structure developed by Suzuki, Ishiguro and Nishizeki.

Tardos has also written two surveys this year. A general survey on complexity theory for The Handbook of Combinatorics [151] jointly with Shmoys from MIT, and a survey on the recent development in the theory of network flows [75] jointly with Goldberg from Stanford and Tarjan from Princeton.

3 Student, Research Associate and Visitor Reports

Javed A. Aslam

Aslam has been working with Rivest on algorithms for machine learning. Specifically, he has been studying the radial mapping problem where a device must infer the shape of its surroundings by rotating in place and taking distance measurements. Relevant cases studied have included those where angular positioning error and distance measurement error are present in varying degrees. Aslam has recently begun work on the inference of Markov chains, and he plans to continue this work with Rivest over the summer.

Mihir Bellare

Basic cryptographic primitives such as zero knowledge proofs and oblivious transfer have classically relied on *interaction* between the parties involved. A part of Bellare's work has focused on a new *public key* model in which such interaction can be removed.

Bellare and Micali [21] proposed a method via which a collection of users may first establish public keys and then be able to accomplish oblivious transfer *without* interaction. Using earlier work of [130] this yields non-interactive methods for zero knowledge proofs.

Bellare and Goldwasser [20] demonstrated the wide applicability of such non-interactive zero knowledge proofs by using them to get simple and efficient schemes for digital signatures and message authentication. A feature of this work was an implementation of non-interactive zero knowledge proofs which could be checked by any user in the system rather than by a single recipient.

In further work related to the role of interaction in zero knowledge proofs, Bellare, Micali and Ostrovsky [22] showed that the languages of graph isomorphism and quadratic residuosity have *constant round* perfect zero knowledge interactive proofs. They also provided a general mechanism to collapse rounds in a statistical zero knowledge proof while preserving the statistical zero knowledge, given some standard cryptographic assumption.

Bonnie Berger

Bonnie Berger has been working on removing randomness from parallel and sequential algorithms. This involves coming up with a randomized algorithm for a problem, if one does not exist, and devising or using known techniques to remove this randomness.

Berger began this work at Bell Labs last summer when, with Peter Shor, she devised a randomized sequential algorithm for the acyclic subgraph problem (the dual of the feedback arc set problem) and used known, highly sequential techniques to convert it to a deterministic one, thereby achieving tight bounds determisitically for the problem [31]. This work also included an RNC algorithm for the problem which, by applying techniques explored in her subsequent work, Berger is attempting to convert to a deterministic one.

Berger's subsequent work has centered around removing randomness from parallel algorithms.

Berger and Rompel [29, 27] developed a general framework for removing randomness from randomized NC algorithms whose analysis uses only polylogarithmic independence. Previously no techniques were known to determinize those RNC algorithms depending on more than constant independence. One application of their techniques is an NC algorithm for the set discrepancy problem, which can be used to obtain many other NC algorithms, including a better NC edge coloring algorithm. As another application of their techniques, they provided an NC algorithm for the hypergraph coloring problem. This work has been chosen for the FOCS '89 Machtey Award.

Berger, Rompel, and Peter Shor [30] gave NC approximation algorithms for the unweighted and weighted set cover problems. Their algorithms use a linear number of processors and give a cover that has at most $\log n$ times the optimal size/weight, thus matching the performance of the best sequential algorithms. Previously, there were no known parallel algorithms for the general set cover problem. Berger, Rompel and Shor devised a randomized algorithm, depending on only pairwise independence, and then converted it to a deterministic one. The hard part here was coming up with the randomized algorithm. Furthermore, they applied their set cover algorithm to learning theory, giving an NC algorithm to learn the concept class obtained by taking the closure under finite union or finite intersection of any concept class of finite VC-dimension which has an NC hypothesis finder. In addition, they gave a linear-processor NC algorithm for a variant of the set cover problem first proposed by Chazelle and Friedman, and used it to obtain NC algorithms for several problems in computational geometry.

Bard Bloom

Bloom, working with Albert Meyer (M.I.T.) and Sorin Istrail (Wesleyan) is 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. One central question in semantics is, "when shall we consider two programs equivalent?" Two proposed notions are *trace congruence* (used in Hoare's language CSP and variants) and *bisimulation* (used in Milner's SCCS). Bloom, Meyer, and Istrail have found an extension of SCCS in which the two notions coincide. The new operation is somewhat peculiar in nature; they have shown that no finite set of operators defined in a clean way can cause the two to coincide. Similarly, bisimulation cannot be understood as equivalence with respect to any set of reasonable experiments. It can be understood in a probabilistic setting; however, the translation from the usual setting to the probabilistic one is not effective.

This work has lead to a notion of "ready simulation" which seems to have the same sorts of formal properties as bisimulation (various alternate definitions, complete axiomatizations and polynomial time decision procedures for finite processes, and so forth), but can also be understood as congruence with respect to a fairly reasonable language.

A classic paper in denotational semantics (Gordon Plotkin's *LCF Considered as a Programming Language*) gives two kinds of semantics for a simple but extremely powerful language based on typed lambda calculus. One semantics is *operational*, describing how a particular interpreter computes; the other kind is *denotational*, assigning meaning to the programs in moderately familiar mathematical terms, using several varieties of Scott domains. The paper shows that the two semantics coincide in a weak sense (*computational adequacy*; two integer terms evaluate to the same constant iff they have the same denotational meaning), but not in a stronger sense (*full abstraction*: two routines behave identically in all contexts iff they have the same denotational meaning). The programming language can be extended by the addition of a "parallel conditional" such that the extended language is fully abstract for one of the denotational models. The classic paper shows that this extension is not fully abstract for the other languages.

However, one of the other denotational models (Scott domains built from complete lattices rather than cpo's) is mathematically appealing, and it is somewhat surprising that the classic paper did not find a fully abstract extension of LCF using this model. However, this is not the author's oversight. Bloom has shown that there is *no* fully abstract extension of LCF with a reasonable evaluator for which this model is fully abstract, where "reasonable" means that an arithmetic expression can evaluate to at most one value. If the evaluator is not required to be reasonable in this sense, there is a simple extension of LCF after the spirit of the classic paper which is fully abstract for the lattice model. If the evaluator is allowed to have a technically peculiar property, it can be made fully abstract for virtually any model of the typed lambda calculus.

Bloom and Riecke have been investigating similar questions for the so-called "lifted Scott domains." Ordinary functional languages exhibit some behavior on higher-order terms: if a program evaluates to a function, it stops and prints "function" — even if the function will always diverge when applied to any argument. In ordinary Scott domains, there is no semantic difference between the function which always diverges given any argument, and a divergent computation of functional type. Lifted domains repair this deficiency. Bloom and Riecke have achieved a close correspondance between operational and denotational semantics for this setting, and are investigating axiom systems.

Avrim Blum

Avrim Blum has been working in two main areas this past year and has also finished his Master's thesis [38] under Ron Rivest's supervision.

He has continued his work with Rivest on problems in computational learning theory—in particular, computational complexity issues in the training of neural networks. One result of this work is a proof that training a very simple neural network with only three computational nodes is NP-complete. This work was presented at the NIPS and COLT conferences [40].

Blum has also been working on approximate graph coloring. The 3-Coloring problem is one of the most well-known NP-complete problems, but there is an enormous gap between the results achieved by the best approximation algorithms for this problem and the best lower

bounds known. Blum devised a new approximation algorithm [39] that reduced this gap somewhat and introduced different techniques for attacking this problem.

Thomas H. Cormen

Cormen continued his work on the textbook *Introduction to Algorithms* with Professors Charles E. Leiserson and Ronald L. Rivest. He plans to start working on parallel computing research over the summer.

Lenore Cowen

Cowen continues work with Goldwasser on two areas: key exchange protocols and information theoretic properties of private functions.

Claude Crépeau

Crepeau's current research interest is mainly the study of two-party cryptographic protocols. His earlier study of disclosure protocols [48, 49] has evolved in a series of results [54, 57, 56, 100, 55] essentially stating that very complex two-party protocols known as *fair oblivious circuit evaluation* (see [55] for definition) can be achieved from very simple devices. Such a device can be a simple noisy channel, for instance. Another such possible device follows the lines of Bennett and Brassard and rely on the correctness of quantum physics. This work was accomplished in part while Crepeau was visiting Aarhus University (Denmark) in the summer.

Crepeau is currently completing his Ph.D. thesis, that will cover some recent material selected from the above papers. He is expected to defend his thesis during the summer.

Zero-knowledge protocols is another of Crepeau's favorite research topics. While visiting IBM Almaden Research Center last summer, he contributed two papers on this subject [50, 47]. These two papers are follow up to [44], in the fact that they are concerned with a model where the prover involved in the protocol is computationally bounded.

Aditi Dhagat

During Fall, 1988, Dhagat was a teaching assistant for the graduate course in Theory of Computation taught by Mike Sipser. During the year she has worked with Sipser in complexity theory and cryptography, trying to construct a pseudorandom number generator secure against monotone circuits without any unproven assmptions. In the process, they have looked at monotone statistical tests and shown that there exist exponential size monotone statistical tests which break the security of the Nisan-Wigderson generator based on parity. They have also shown that if there exist monotone functions which are hard to approximate for polynomial size monotone circuits, then there exist pseudorandom number generators secure against polynomial size monotone circuits.

Dhagat plans to continue to work on this question during the summer of 1989.

Michael Ernst

Ernst became a graduate student at MIT in January, 1989. He worked under Meyer's supervision to prove a monotone model adequate for recursive program schemes. In order to prove adequacy, most proofs in the literature directly use a stronger continuous model which simplifies the proof and which implies the weaker result; the typical approach is via Tait's method of computability [141, 158]. The introduction of continuity is poorly motivated from an expository and pedagogical viewpoint; we would hope to be able to show the result directly [126].

Ernst and Meyer [63] found that this was not possible; although they were able to produce a clear exposition of the concept, at one crucial point continuity was required. While the result holds for the monotone model without mention of continuity, a weaker assumption of monotonicity in the proof leads to a failure of the result.

Ernst spent much of 1989 finishing up his undergraduate requirements; he plans to get started on his SM thesis during the upcoming year.

Lance J. Fortnow

Fortnow working with Sipser examined the relationship between probabilistic polynomial time and probabilistic linear time. They showed [65] the existence of an oracle under which the two classes are identical. This result means the techniques of separating the deterministic and non-deterministic time hierarchies will not work for probabilistic computation. They also show many other results relating to probabilistic computation and linear time.

During the spring of 1989, Fortnow spent the semester writing his thesis [64] and looking for a job.

Jeff Fried

Fried has continued research on the architecture, design, and analysis of communication networks for use in parallel computers and telecommunications. He completed a master's thesis [70], supervised by Leiserson, which includes two switch designs for such networks [73, 69]. Follow-up work in this area has included an improved circuit design for one of the VLSI functions used in these designs [71], and a study of some of the modularity tradeoffs found in sparse circuit-switched interconnection networks [72].

Fried is currently working on a number of problems related to the architecture and control algorithms needed for high performance communication networks. This work includes a study of the impact of synchrony on the performance of distributed algorithms, and design studies of a VLSI packet router for use in broadband networks [68].

Sally A. Goldman

Goldman has been working with Rivest on studying learning algorithms for concepts that have polynomial sized instance spaces [77, 78]. They have focused on polynomial prediction algorithms in which the learner predicts a value for each entry in the instance space and then receives feedback as to whether the prediction was correct. They consider the worst case mistake bounds under several models for the selection of the instances. Often good mistake bounds are obtained by the halving algorithm. They discuss an approximate halving algorithm and show how a fully polynomial randomized approximation schemes can be used to implement (with high probability) the approximate halving algorithm. They demonstrate these technqiues on the problem of learning a total order on a set of n elements.

Goldman has also been working with Rivest and Schapire on the particular problem of learning a binary relation between n objects of one kind and m of another [78]. This can be viewed as the problem of learning an $n \times m$ binary matrix. Here the instance space contains the elements of the matrix and is thus of polynomial size. They present numerous upper and lower bounds on the number of mistakes that prediction algorithms can make under different models for the selection of the instances.

Goldman has also done some research in the field of computational geometry. In particular, she has developed an algorithm to compute the greedy triangulation of an arbitrary point set that takes $O(n^2 \lg n)$ time and $O(n)$ space [76]. In January, Goldman participated in the robot building project lead by Schapire.

Ronald I. Greenberg

Greenberg has worked on three main topics during the past year: networks for general-purpose parallel computation, multi-layer channel routing, and bounds on the area for VLSI implementations of finite-state machines.

Recent work on networks for general-purpose parallel computation is reported in [83]. This paper provides several extensions and generalizations of earlier work on the problem of designing "universal" networks which can simulate any other network of comparable physical size with only polylogarithmic overhead in simulation time.

On the topic of multi-layer channel routing, Greenberg has been seeking improvements upon algorithms recently developed with Alex Ishii and Alberto Sangiovanni-Vincentelli (U. C. Berkeley) for the program MULCH [84]. The basic approach of MULCH is to divide a multi-layer problem into essentially independent subproblems of one, two, or three layers. A main step in MULCH is to greedily partition the nets once a set of layer groups has been determined. As each net is considered, it is assigned to the group where the resulting subproblem seems to be the one requiring the least channel width. For testing the required channel width of single-layer partitions, Greenberg and Miller Maley (Princeton U.) have devised algorithms which are more efficient than naive approaches involving complete routing of the layer. Greenberg is also developing "incremental" algorithms to quickly determine the effect on certain subproblem characteristics when a new net is added, by taking advantage of knowledge derived from earlier computations on the subproblem.

Finally, Greenberg and Mike Foster (Columbia U. and NSF) have derived lower bounds on the area required for VLSI layout of finite-state machines [66]. These lower bounds show that naive layout approaches are optimal in the worst case.

Michelangelo Grigni

Grigni is a third year graduate student supervised by Dr. Sipser. His thesis research considers the construction of fast robust broadcasting networks, continuing work begun with David Peleg [85] of the Weizmann Institute. Current work with Dimitris Bertsimas of the Sloan School extends a their recent result [33] on the suboptimality of the space-filling curve heuristic for the Eulcidean TSP problem. Other work with Bertsimas includes a survey of various #NP complete problems. Grigni continues searching for new attacks on the matrix multiplication exponent problem.

Carolyn M. Haibt

Haibt spent most of the year on coursework, but also continued work with Tardos. They are currently working on algorithms for the generalized network flow problem. This is a generalization of the maximum flow problem, where each edge has an associated gain factor, and flow is multiplied by this factor when it passes through an edge.

Mark D. Hansen

Hansen has been studying graph embeddings with applications to parallel processing problems. In [88] he examines the problem of finding optimal geometric embeddings in the plane and higher dimensional spaces. Given an undirected graph G with n vertices, and a set P of n points in R^d, the *geometric embedding problem* consists of finding a bijection from the vertices of G to the points in the plane which minimizes the sum total of edge lengths of the embedded graph. In general this problem is NP-complete as it contains the Euclidean Traveling Salesman Problem as a special case. Hansen gives approximation algorithms for embedding many of the important graphs studied in the theory of parallel computation. He presents fast algorithms for embedding d-dimensional grids in the plane which are within a factor of $O(\log n)$ times optimal cost for $d > 2$ and $O(\log^2 n)$ for $d = 2$. He also shows that any embedding of a hypercube, butterfly, or shuffle exchange graph must be within an $O(\log n)$ factor of optimal cost. When the points of P are randomly distributed, or arranged in a grid, he is able to use the results of Leighton and Rao [112] to give a polynomial time algorithm which can embed arbitrary weighted graphs in these points with cost within an $O(\log^2 n)$ factor of optimal.

Hansen shows how the algorithms developed in [88] for geometric embeddings can be used to give solutions which are within an $O(\log^2 N)$ factor of optimal to problems of performance optimization for array-based parallel processors in the following areas: communication load

balancing, dynamic allocation of jobs to processors, reconfiguring around faults, and simulating other architectures. He also indicates some applications to wafer scale integration problems and the dynamic configuration of distributed computing networks.

Working with Leighton, Hansen was able to apply some of the techniques developed in [88] to give an $N^{O(\sqrt{N})}$ time algorithm for solving the Euclidean Traveling Salesman Problem. The previous best running time for this algorithm was $O(\log N 2^N)$. A year earlier Warren Smith [156] independently gave an algorithm with the same running time, using different techniques involving the Lipton-Tarjan planar separator theorem. [119] Hansen and Leighton are currently investigating the possibility of developing practical heuristics for solving Euclidean TSP using the ideas in these two algorithms.

Alexander T. Ishii

Alexander Ishii has completed his masters thesis[94], which describes his models for VLSI timing analysis. The model maps continuous data-domains, such as voltage, into discrete, or *digital*, data domains, while retaining a continuous notion of time. The majority of the thesis concentrates on developing lemmas and theorems that can serve as a set of "axioms" when analyzing algorithms based on the model. Key axioms include the fact that circuits in our model generate only well defined digital signals, and the fact that components in our model support and accurately handle the "undefined" values that electrical signals must take on when they make a transition between valid logic levels. In order to facilitate proofs for circuit properties, the class of *computational predicates* is defined. A circuit property can be proved by simply casting the property as a computational predicate.

Ishii has also been working with Ronald Greenberg and Alberto Sangiovanni-Vincentelli of Berkeley on a multi-layer channel router for VLSI circuits, called MULCH [84]. While based on the CHAMELEON system developed at Berkeley, MULCH incorporates the additional feature that nets may be routed entirely on a single interconnect layer (CHAMELEON requires the vertical and horizontal sections of a net be routed on different interconnect layers). When used on sample problems, MULCH shows significant improvements over CHAMELEON in area, total wire length, and via count.

Ishii has continued work, begun with Bruce Maggs, on a new VLSI design for a high-speed multi-port register file. Design goals include short cycle-time and single-cycle register window context changes. This research began as an advanced VLSI class project, under the supervision of Thomas Knight of the MIT Artificial Intelligence Laboratory.

Lalita A. Jategaonkar

Jategaonkar has been working jointly with Albert Meyer on further developing research begun last year at Bell Laboratories with John C. Mitchell. In [95], Jategaonkar and Mitchell develop an extension of the programming language ML in which a restricted object-oriented style can be achieved. In keeping with the framework of ML, a type derivation system and a type inference algorithm is presented. It is proved that the algorithm is sound and complete

with respect to the type derivation system, and that it infers a most general typing of every typable expression in the language. This research will comprise Jategaonkar's forthcoming Master's thesis.

In order to show that the type derivation system is "reasonable" in a precise, technical sense, Jategaonkar and Meyer have been developing an interpreter for this language. They aim to show that the interpreter satisfies certain desirable properties, and that the interpreter and the type derivation are well-matched in the sense that no typable expression in the language reduces to a type error. Jategaonkar is also interested in further extending ML to support subtyping of abstract types and recursive types. Another direction of research she is interested in pursuing is to develop a semantics for these extensions of ML.

Trevor Jim

Jim entered the department in September, 1988. His previous work with Appel [8] on a novel code generator for the language ML was presented at POPL '89 in January.

Under the direction of Meyer, he has been studying the work of Berry and Curien [58, 32] on models of PCF [141] based on stable functions and sequential algorithms. These models were developed as alternatives to the standard model, which contains troublesome "non-sequential" elements. Jim is trying to find extensions of PCF for which the alternate models are fully abstract.

Joe Kilian

Kilian has spent most of his time working on his thesis, "Randomness in Algorithms and Protocols" [99], which he has recently completed. He has also done some work in efficient zero-knowledge interactive proofs, bounded interaction zero-knowledge proofs, noninteractive zero-knowledge proofs, multi-prover zero-knowledge proofs, space-bounded secure protocols, communication lower bounds for secret sharing, and IP v.s. AM.

A troubling issue in theoretical cryptography is the chasm between what is efficient in theory and what is efficient in practice. One area in which this gap is particularly large is in zero-knowledge proofs for NP predicates. Suppose one wishes to prove in zero-knowledge that some circuit, $C(x_1, \ldots, x_n)$, is satisfiable. The previously most efficient solutions to this problem ([44], [93]) required the prover and the verifier to send $O(k|C|)$ bits back and forth per iteration of the protocol. Here, $|C|$ denotes the number of gates in the circuit C, and k denotes the security parameter. Using pseudorandom generators, Kilian [100] has exhibited a protocol in which the prover and the verifier communicate only $O(|C|+k^2)$ bits per iteration of the protocol. In real life circumstances, $|C|$ is likely to be very large, in which case, this protocol should behave better in practice as well as in theory.

Zero-knowledge proofs typically require a great deal of interaction between the prover and the verifier. It is of both theoretical and practical interest to see how much interaction is truly needed, which has led to the notions of *bounded interactive protocols* and *noninteractive protocols with a common random string*. In bounded interaction protocols, the prover and

the verifier interact for time polynomial in the security parameter. After the interaction phase, the prover proves theorems to the verifier by sending him a letter in the mail. In a noninteractive protocol with a common random string, the prover and verifier do not interact at all, but are both presented with a uniformly distributed string of length polynomial in the security parameter.

Prior to Kilian's work, there existed three proposed protocols for these scenarios, due to Blum–Feldman–Micali [41], De Santis–Persiano–Micali [59], and Micali–Ostrovsky [136].

Kilian has developed a very simple and efficient protocol for bounded interaction zero-knowledge proofs, and a provably secure protocol for noninteractive zero-knowledge with a common random string. Both of these protocols' security bis ased on reasonable cryptographic assumptions. His protocol for bounded interaction zero-knowledge proofs is more communication efficient than the best previously known interactive zero-knowledge protocols. In both of these protocols, the prover can prove polynomially many polynomial-sized theorems.

In [25], Kilian, along with Ben-Or, Goldwasser, and Wigderson, developed a multiprover generalization of interactive proof systems. They showd that, informally, anything 2 provers could prove, they could prove in statistical zero-knowledge. Recently, Kilian has strengthened this result, showing that anything 2 provers could prove, they could prove in perfect zero-knowledge.

With Noam Nisan, Kilian has applied knowledge complexity notions from cryptography to space-bounded automata [102]. They have developed protocols in this scenario for a number of cryptographic protocols: secret key exchange, bit-commital, secure circuit evaluation, and zero-knowledge proofs. In the space-bounded scenario, the security of these protocols may be proven without any assumptions whatsoever. Furthermore, these protocols are robust against adversaries who have asymptotically more space that used by the good players.

Nisan and Kilian have also investigated upper and lower bounds for secret sharing. They consider schemes in which a bit b is shared among n players, such that,

1. A majority of the n players can reconstruct b.
2. A nonmajority of the players can't reconstruct any information about b.

They show a lower bound of $\Omega(n \log n)$ on the total number of bits that must be distributed amongst the n players. They also consider a weakened form of secret sharing, in which $2n/3$ players can reconstruct b, and $n/3$ players learn nothing. They use coding theory to prove the existence of secret sharing schemes that are more efficient than the lower bounds proven for the more stringent conditions.

A classic theorem of Goldwasser and Sipser [82] states that IP=AM. In other words, public coins are as powerful as private coins for interactive proof systems. Kilian has found a very simple proof of this fact, using random selection techniques from [81]. This proof will be included in a paper on random selection, with Oded Goldreich, Johan Håstad, and Yishay Mansour.

Shlomo Kipnis

Shlomo Kipnis has been investigating parallel architectures and interconnection networks. He is trying to further explore the power of bussed interconnection schemes in routing permutations and realizing various communication patterns. Bussed interconnection schemes and their relation to difference covers was explored by Joe Kilian, Shlomo Kipnis, and Charles Leiserson in [101]. In addition, he is investigating various arbitration schemes for bussed based architectures.

Recently, he has also studied the problem of range queries in computational geometry. Range queries is a fundamental problem in computational geometry with applications to computer graphics and database retrieval systems. He compiled a survey report on three different methods for range queries in computational geometry [103].

Richard R. Koch

Koch's Ph.D. thesis [106] is a probabilistic analysis of routing on a parallel architecture. Koch analyzes the bandwith of the butterfly network. In a dilated butterfly network nodes are connected by parallel edges instead of just one edge as in the usual butterfly network. He proves a previous conjecture that the expected bandwidth of an N node dilated butterfly network is $\Theta(N(\log N)^{-\frac{1}{q}})$, where q is the number of parallel edges. He also explores some implications of his results for design tradeoffs. He also develops interesting techniques for finding asymptotics for nonlinear systems of recurrences. Many of the results appeared previously in [105].

In [107] Koch, Leighton, Maggs, Rao and Rosenberg study the problem of emulating T_G steps of an N_G-node guest network on an N_H-node host network. Although many isolated emulation results have been proved for specific networks in the past, and measures such as dilation and congestion were known to be important, the field has lacked a model within which general results and meaningful lower bounds can be proved. They attempt to provide such a model, along with corresponding general techniques and specific results in this paper.

Dina Kravets

Kravets spent most of the year working with Alok Aggarwal and James Park on problems in computational geometry. In January, she finished her Master's thesis [109] which included the following results:

1. An algorithm to find all the farthest neighbors of every vertex on a convex n-gon in $\Theta(n)$ time.
2. An $O(n^2)$ algorithm to sort the distances of all the vertices of a convex n-gon with respect to each vertex of the convex n-gon.
3. An $O(kn \log k)$ time algorithm to find k *farthest vertices* for every vertex of a convex n-gon.

4. A worst-case optimal algorithm to sort a set of numbers given lower bounds on the ranks.

The first of these algorithms appeared in the *Information Processing Letters* [4]. Park and Kravets are planning to improve the third result and submit it to the ACM-SIAM Symposium on Discrete Algorithms.

Kravets is also looking at some problems in parallel computation and VLSI with Leighton.

Leonid A. Levin

The topic of Leonid Levin's research in 1988-89 may be called "Randomness in Computing." In [115], Levin and Venkatesan propose the first intractability results for random instances of NP problems. NP-complete problems should be hard on *some* (may be extremely rare) instances. Generic instances of many such problems proved to be easy. This paper shows the intractability of *random* instances of a graph coloring problem. Applications of average case intractability are considered in two other papers: [80, 92].

Blum and Micali [42] discovered permutations f with "hard-core" predicates $b(x)$ that cannot be efficiently guessed from $f(x)$ with a noticeable correlation. Both b, f are easy to compute. Yao [162] modifies any one-way permutation f into f^* which has a hard-core predicate. Its security may be lower than any constant power of the security of f and is too small for practical applications. Goldreich and Levin [80] prove that most linear predicates are hard-cores for every one-way function and have almost the same security. The result extends to multiple (up to the logarithm of security) hidden bits and has wide applicability to pseudorandomness, cryptography, etc.

Let an easily computable function f be one-way, i.e. for most x one cannot recover from $f(x)$ either (1) x by a polynomial time algorithm, or (2) an $x' \in f^{-1}(f(x))$ by a polynomial size circuit. In case (1), to exclude useless $f(x) = 0$, the difference between Shannon entropies of inputs and outputs of f is restricted to $O(1)$. Impagliazzo, Levin, and Luby [92] show, based on [80], that the existence of one-way functions in the sense (1) and (2) is necessary and sufficient for the existence of pseudo-random generators secure against feasible algorithms or circuits, respectively.

In [114], Levin compares probability distributions of computational objects. The usual distributions are concentrated on strings that differ little in any fundamental characteristic, except their informational size (Kolmogorov complexity). This property distinguishes a class of *homogeneous* probability measures suggesting various applications. In particular, it explains why the average case NP-completeness results are so measure independent, and offers their generalization to this wider and more invariant class of measures. It also demonstrates a sharp difference between pseudo-random strings and the objects known before.

Bruce Maggs

Bruce Maggs is studying the ability of a *host* network to emulate a possibly larger *guest* network [107]. His collaborators in this research are Richard Koch, Tom Leighton, Satish

Rao, and Arnold Rosenberg. An emulation is *work-preserving* if the work (processor-time product) performed by the host is at most a constant factor larger than the work performed by the guest. Such an emulation is efficient because it achieves optimal speedup over a sequential emulation of the guest. Many work-preserving emulations for particular networks have been discovered. For example, the N-node butterfly can emulate an $N \log N$ node shuffle-exchange graph and vice versa. On the other hand, a work-preserving emulation may not be possible unless the guest graph is much larger than the host. For example, a linear array cannot perform a work-preserving emulation of a butterfly unless the butterfly is exponentially larger than array. These positive and negative results provide a basis for comparing the relative power of different networks.

Bruce Maggs is also studying algorithms for routing packets on faulty bounded-degree networks. With Tom Leighton he has developed a scheme for routing N packets on an N-node multibutterfly network [159] in $O(\log N)$ steps even in the presence of many faulty nodes.

Yishay Mansour

Yishay Mansour has continued studying data transmission in communication networks. In a work with Schieber [122] they show lower bounds for communication over non-Fifo links. In a work with Herzberg and Goldriech [79] they give a randomized protocol for communication over non-FIFO links. In a work with Awerbuch and Shavit [15] they show how to achieve polynomial end to end communication.

In work with Linial and Nisan [116] they investigate constant depth circuit using the Fourier Transform. They are able to show a quasi-polynomial time algorithm for learning this class. Another work that is connected to learning is [23].

In a work with Schieber and Tiwari [124] they continue to develop techniques to prove lower bound for integer coumputations. The work with Schieber and Tiwari [123] tries to explore the complexity of approximating algebraic functions. In this work, techniques taken from Approximation Theory are used to derive lower and upper bound.

Mark J. Newman

Mark J. Newman continued work on fault-tolerant strategies for parallel computation. With Johan Hastad and Tom Leighton [89], he demonstrated algorithms for reconfiguring hypercubes with faulty components. After reconfiguration, the hypercubes retain all computational power (within constant factors). The algorithms are successful with high probability, given that nodes and edges fail independently and with constant probability. They also showed how to route permutations on hypercubes even if a constant fraction of the cube's components have failed.

With Tom Leighton, Abhiram Ranade and Eric Schwabe [111], Newman also showed how a dynamically changing binary tree can be embedded in a hypercube so that computational and communication overhead are low. Specifically, they produced randomized algorithms which embed any growing and shrinking binary tree so that the resulting simulation requires only constant factor overhead, with high probability.

Noam Nisan

Nisan arrived as a post-doc in the theory group in January '89. He has been working mainly on problems related to complexity theory.

Together with L. Babai and M. Szegedy [17] he proved lower bounds for the multiparty communication complexity of certain simple functions. These bounds were used to obtain a pseudorandom generator for Logspace without relying on any unproven assumptions.

In [133] Nisan obtained a full characterization of the parrallel time needed to compute any boolean function on a CREW PRAM in terms of the function's decision tree complexity.

In joint work with N. Linial [118], the question of obtaining approximate versions of the Inclusion-Exclusion formula is tackled. Tight upper and lower bounds are proved for several formulations of this question.

Nisan and J. Kilian [102] considered cryptographic protocols in the setting where all parties are space-bound. In this setting they design secure protocols for a wide spectrum of cryptographic problems. The security of these protocols is proved without relying on any unproven assumptions.

In his joint work with N. Linial and Y. Mansour [116], constant depth circuits are studied in terms of their Furier transform. It is shown that almost all of the power spectrum of a function in AC^0 lies in the low coefficients. This fact is used to obtain a learning algorithm for constant depth circuits, as well as several others results.

Marios C. Papaefthymiou

Papaefthymiou began his studies as a graduate student at MIT in September, 1988. He is working on his SM thesis under the supervision of Professor C. E. Leiserson.

His research focuses on the design of efficient algorithms for pipelining of combinational circuitry. A general framework for this problem has been given by C. E. Leiserson and J. Saxe [113].

Papaefthymiou has given an $O(E)$ optimal algorithm for minimum latency pipelining of combinational circuitry with constrained clock period. He also investigates methods for pipelining combinational circuitry using minimum number of registers.

James K. Park

James K. Park spent most of the last year collaborating with Alok Aggarwal (I.B.M., Yorktown Heights) and Dina Kravets on a number of problems relating to totally monotone arrays. Park's work with Aggarwal (described in [5],[6], and another manuscipt "Parallel Searching in Multidimensional Monotone Arrays," currently in preparation) centers on the problem of finding maximum entries in totally monotone arrays and applications of efficient sequential and parallel algorithms for this problem to problems in computational geometry, dynamic programming, string matching, and VLSI river routing. This work generalizes and

extends the results of [3]. (Park's master's thesis [137], finished in January, is also on this subject.) Park's work with Kravets (described in Kravets' master's thesis [109]) considers two more comparison problems — sorting and computing order statistics — in the context of totally monotone arrays and applications of efficient solutions to these problems.

In the coming year, Park plans to continue his research relating to totally monotone arrays and computational geometry.

Cynthia A. Phillips

Cynthia A. Phillips developed an $O(\lg^2 n)$-time $(n+e)/\lg n$-processor deterministic parallel algorithm to contract general n-node, e-edge graphs to a single node. This algorithm is used as a subroutine in an algorithm developed with Charles Leiserson to contract n-node bounded-degree graphs in $O(\lg n + \lg^2 \gamma)$ time with high probability where γ is the maximum genus of any connected component. A deterministic version runs in time $O(\lg n \lg^* n + \lg^2 \gamma)$. The algorithm for bounded-degree graphs uses $n/\lg n$ processors [139]. The contraction algorithm can be used to solve the connected-components, biconnected-components, and spanning-tree problems.

Cynthia Phillips with Stavros Zenios of U. Penn have completed a preliminary experimental study of the solution of large assignment problems on the Connection Machine (TM) multiprocessor. The assignment problem is also known as maximum-weight bipartite matching. They have developed heuristics to improve sequential "tail" behavior which seems to limit the usefulness of many current parallel algorithms for the assignment problem and related flow problems [140].

Phillips will be writing her thesis this summer. Among the new research that will probably be included is an analysis of the permutation distribution of the Benes network. In other words, how many distinct ways can the switches of a Benes network be set to yield a given permutation? If the permutations are well distributed, then pseudorandomly setting the switches of a Benes network may yield a good pseudorandom permutation network.

Satish B. Rao

In [107] Koch, Leighton, Maggs, Rao and Rosenberg study the problem of emulating T_G steps of an N_G-node guest network on an N_H-node host network. Although many isolated emulation results have been proved for specific networks in the past, and measures such as dilation and congestion were known to be important, the field has lacked a model within which general results and meaningful lower bounds can be proved. They attempt to provide such a model, along with corresponding general techniques and specific results in this paper.

Leighton and Rao have developed an approximate min-cut max-flow theorem for a type of multicommodity flow problem. This theorem yields an approximation algorithm for finding a separator in arbitrary graphs that costs at most a $O(\log^2 n)$ times the optimal. They also used the theorem to show that any permutation can be routed on an arbitrary network so that the congestion of any edge and path length of any message is within a $O(\log n)$ factor

of optimal. In joint work with Maggs, they explore the problem of scheduling messages on paths with given congestion and length so that the routing time is minimized.

Jon G. Riecke

Riecke continues to work in the area of semantics and logic of programming languages, with two primary interests: the semantics of continuations, and the theory of "lazy" (call-by-name) functional languages. Working jointly with Albert Meyer, he investigated some seemingly known—but undocumented—problems in the theory of continuations. More specifically, Meyer and Riecke showed that either programming with continuations explicitly or using special "continuation-accessing" operators (*e.g.*, Scheme's `call/cc`) leads one to conclude different facts about code; old equivalences between programs may no longer hold in a setting with continuations. The implications of these results, and their precise statements, are reported in [128] and in Riecke's S.M. thesis [144].

The theory of lazy languages, begun by Abramsky and Ong, has also become a focus of Riecke's work. Lazy functional languages pass arguments by name (that is, arguments are not evaluated before passing), but nevertheless stop evaluating higher-order expressions—functions—when they can build a *closure*. The usual Scott-style semantics do not predict this termination behavior correctly: a divergent functional and a closure that always diverges have the same meaning. Bard Bloom and Riecke [37] developed a model for a *typed* lazy language that accurately reflects the behavior of the interpreter. Stavros Cosmadakis and Riecke (in a forthcoming paper) used the model to develop principles for reasoning about lazy programs, and proved that equalities between terms in a fragment of the language are decidable.

In the past year, Riecke has also become interested in intuitionistic logic and type theory, and its applications to the theory of programming languages. He will continue his reading, as well as pursuing previous lines of research.

Phillip Rogaway

Rogaway is a third year graduate student working under Silvio Micali. He has been working on cryptography and complexity theory.

Rogaway's Master's thesis evolved into the CRYTO-88 paper which easily won the award for most coauthors, [24]. This paper establishes that an injective one-way function suffices to prove all of **IP** in computational zero-knowledge. It also shows that the "envelope model" for bit commitment suffices to show all of **IP** has perfect zero-knowledge proofs.

Rogaway has investigated generalized notions of knowledge complexity, e.g., protocols that release a "small" (but nonzero) amount of information. Recently he has been working on reducing the interaction required for secure distributed computation.

John Rompel

In January, Rompel completed his Master's thesis [148], based on approximation algorithms for graph coloring developed last year with Berger [28].

More recently, Rompel has been working on problems in the field of parallel algorithms. Rompel, together with Berger, [29] developed a general framework for removing randomness from randomized NC algorithms whose analysis uses only polylogarithmic independence. Previously no techniques were known to determinize those RNC algorithms depending on more than constant independence. One application of their techniques is an NC algorithm for the set discrepancy problem, which can be used to obtain many other NC algorithms, including a better NC edge coloring algorithm. As another application of their techniques, they provided an NC algorithm for a hypergraph coloring problem.

Rompel, working with Berger and Peter Shor [30], gave NC approximation algorithms for the unweighted and weighted set cover problems. Their algorithms use a linear number of processors and give a cover that has at most $\log n$ times the optimal size/weight, thus matching the performance of the best sequential algorithms. Previously, there were no known parallel algorithms for the general set cover problem. Berger, Rompel and Shor devised a randomized algorithm, depending on only pairwise independence, and then converted it to a deterministic one. Furthermore, they applied their set cover algorithm to learning theory, giving an NC algorithm to learn the concept class obtained by taking the closure under finite union or finite intersection of any concept class of finite VC-dimension which has an NC hypothesis finder. In addition, they gave a linear-processor NC algorithm for a variant of the set cover problem first proposed by Chazelle and Friedman, and used it to obtain NC algorithms for several problems in computational geometry.

Arie Rudich

Rudich began his first year as a graduate student at MIT in September, 1988. He is working on an SM thesis supervised by Meyer on Dataflow theory which should be complete by January, 1990.

His research aims to generalize recent results by Rabinovich and Trakhtenbrot [143] and by Lynch and Stark [121] which precisely delimit the classes of dataflow networks for which Kahn's "Least Fixed Point Principle" [97] applies, showing that Kahn's Principle fails precisely where Brock-Ackerman-like anomalies [51] begin. Rabinovich and Trakhtenbrot established this boundary without distinguishing completed and incompleted output streams. Rudich aims to show that the results carry over to the more conventional model where the completed/incompleted distinction is maintained.

Robert E. Schapire

Schapire has continued to work with Rivest on the problem of inferring an unknown finite-state automaton from its input/output behavior. In [145], they introduce a powerful new

technique, based on the inference of homing sequences, for solving this problem in the absence of a means of resetting the machine to a start state. Their inference procedures experiment with the unknown machine, and from time to time require a teacher to supply counterexamples to incorrect conjectures about the structure of the unknown automaton. In this setting, they describe a learning algorithm that, with probability $1 - \delta$, outputs a correct description of the unknown machine in time polynomial in the automaton's size, the length of the longest counterexample, and $\log(1/\delta)$. They present an analogous algorithm that makes use of a diversity-based representation of the finite-state system. Their algorithms are the first that are provably effective for these problems, in the absence of a "reset." They also present probabilistic algorithms for permutation automata which do not require a teacher to supply counterexamples. For inferring a permutation automaton of diversity D, they improve the best previous time bound by roughly a factor of $D^3/\log D$.

In January, Schapire led a team participating in the robot building contest of the AI Lab's "Winter Olympics." The goal of their project was to build a robot capable of performing some simple learning task. In particular, the robot they built, named S'bot (for Smartbot or Spotbot), was able to learn from experience how to avoid running into walls and other obstacles. Their team consisted of Amsterdam, Blum, Goldman, Moore, Rivest and Schapire.

Schapire has also been working with Goldman and Rivest on the problem of inferring a binary relation [78] between n objects of one kind and m of another. This can be viewed as the problem of inferring an $n \times m$ binary matrix. Their goal has been to minimize the number of prediction mistakes made by a learner presented with such a matrix one entry at a time. They have been able to prove numerous upper and lower mistake bounds for several variations of this problem.

Finally, Schapire has been looking at problems relevant to the distribution-free ("pac") learning model introduced by Valiant [160]. In [149], Schapire considers the problem of improving the accuracy of a hypothesis output by a learning algorithm. He shows that a model of learnability, called *weak learnability*, in which the learner is only required to perform slightly better than guessing is as strong as a model in which the learner's error can be made arbitrarily small. His result may have significant applications as a tool for efficiently converting a mediocre learning algorithm into one that performs extremely well.

Leonard Schulman

Schulman spent most of his time on coursework this year. In the spring he developed an algorithm for sorting n elements on an n-node ring of processors in the optimal time $n/2$. This requires only constant capacity at each node in the word model. Y. Mansour proved a closely related lower bound and these two results have been combined in a joint paper to be submitted shortly.

During the summer of 1989 Schulman intends to read under the guidance of M. Sipser.

Eric J. Schwabe

Schwabe has been working on problems involving the efficient implementation of dynamic structures on fixed-connection networks. In particular, he worked with Leighton, Newman and Abhiram Ranade (Berkeley) on the problem of dynamically embedding binary trees in butterfly and hypercube networks [111]. Randomized embedding algorithms were found for both networks which simultaneously optimize load (the maximum number of tree nodes mapped to a processor) and dilation (the maximum distance in the network between adjacent tree nodes) for trees which are a logarithmic factor larger than the host network. An improved algorithm for the hypercube was found which optimizes load and dilation for arbitrary binary trees, while also keeping congestion (the number of times a hypercube edge is 'traced over' by an embedded tree edge) low. Also, lower bounds were proved which show that deterministic algorithms cannot simultaneously optimize load and dilation.

Schwabe has also been studying the relative strengths of the butterfly and shuffle-exchange graphs as interconnection networks. He proved that normal hypercube algorithms (those which use only one dimension of hypercube edges at a time, and adjacent dimensions in consecutive time steps) can be simulated on a butterfly network with only a constant slowdown, a result which was previously known only for the shuffle-exchange graph. A version of this result is being prepared for journal submission. In addition, he recently discovered a one-to-one embedding of the butterfly into the shuffle-exchange graph with constant dilation and congestion, and expansion $2^{O(\sqrt{\log N})}$, improving a result of Koch et. al. [107].

Over the next year, Schwabe plans to work on relating the ideas in [111] to other problems in parallel memory management, and to continue his investigation of the shuffle-exchange vs. the butterfly.

Alan Sherman

Professor Alan T. Sherman (now at Tufts University) has completed a monograph on the PI System for placement and interconnect of custom VLSI circuits [150]. The PI System was designed and implemented at MIT under the leadership of Professor Ronald L. Rivest; Sherman was one of the key architects. The monograph is being published by Springer-Verlag. Beginning September 1989, Sherman will join the faculty at the University of Maryland Baltimore County.

Robert Sloan

Sloan's primary area of interest this past year was computational learning theory. His major activity for the year was preparing his doctoral dissertation [155]. Most of the other work in computational learning theory described here is also contained in that work.

Much of his work was within Valiant's model of probably approximately correct learning [160]. working with Helmbold and Warmuth while visiting U. C. Santa Cruz, he developed an algorithm for learning certain complex combinations of concept classes known to be learnable [91].

In [146] the problem of learning arbitrary boolean concepts in the Valiant model—by breaking them into pieces and learning one piece at a time is studied. In other work, Sloan studied the effects of different sorts of noise on learning in the Valiant model [153].

In [147] he explored an alternate model of inductive inference.

Sloan also remains interested in the subject of cryptography, and spent some time studying different definitions of zero knowledge [154].

Clifford Stein

Stein has been working with Shmoys on developing parallel algorithms for combinatorial optimization problems. Together with Philip Klein of Harvard, he has developed a parallel algorithm to find a maximal set of edge disjoint cycles in an undirected graph in $O(\log n)$ time using m processors on a CRCW PRAM. A maximal set of edge disjoint cycles is a set of cycles whose removal from the graph renders the graph acyclic. Stein and Klein have also been able to generalize this result to multi-graphs and obtain an algorithm which runs in $O(\log n \log C)$ time, where C is the largest multiplicity of any edge [104].

Using this algorithm, Stein has developed an algorithm which finds a *cycle cover* containing $O(m + n \log n)$ edges using $O(\log^2 n)$ time on m processors. A cycle cover is a set of cycles such that every edge in the graph appears in at least one cycle.

Stein has observed that the parallel matching algorithms of [132] and [98] can be combined with scaling to achieve *RNC* algorithms for the assignment problem which use a number of processors independent of the size of the largest number in the problem, by slowing down the running time by a factor proportional to the logarithm of the size of the largest number in the problem.

Stein has also been rewriting his undergraduate thesis [157] for publication. Together with Ravi Ahuja, Jim Orlin, and Bob Tarjan he has developed efficient algorithms for a wide variety of network flow problems in bipartite graphs. The main results are of the following form: given a bipartite graph with n nodes, but only n_1 nodes in the smaller half of the bipartition, an algorithm which runs in time $O(f(n, m))$ can be converted into an algorithm which runs in time $O(f(n_1, m) + n_1 m)$. This approach leads to an algorithm for bipartite maximum flow which runs in $O(n_1 m \log(\frac{{n_1}^2}{m} + 2))$ time, an algorithm for bipartite minimum cost circulation which runs in $(n_1 m \log n_1 \log(n_1 C))$ time, and an algorithm for parametric maximum flow which solves l bipartite maximum flow problems in $O(ln + n_1 m \log(\frac{l n_1 + {n_1}^2}{m} + 2))$ time.

Margaret C. Tuttle

Tuttle joined the theory group this year and has been working with Shmoys on approximation algorithms for the Mixed Postman Problem: Given a weighted graph G, find a least-cost tour of G which traverses each edge at least once. When G is totally directed or totally undirected, the problem can be solved in polynomial time. When G is a mixed graph (i.e.,

some edges are directed and some are undirected), the problem is NP-complete (shown by Papadimitriou in 1976).

This summer she will continue working with Shmoys.

Joel Wein

Wein has been working with Shmoys on parallel graph algorithms. He recently extended a result of Karloff's to obtain a Las Vegas *RNC* algorithm for minimum weight perfect matching, where the weights are represented in unary. This problem was shown to be in *RNC* by Karp, Upfal and Wigderson, but the algorithm was *Monte Carlo* in nature: it yielded a correct solution with high probability, but was unable to determine if the solution was indeed optimal. Wein developed a way to carry out this certification in *RNC*, yielding a robust *Las Vegas* algorithm that can verify optimality. The result utilizes a structure theorem of Sebö for the t-join problem and yields an *RNC* Las Vegas algorithm for that problem as well.

Over the summer Wein worked at Thinking Machines Corporation, developing practical Connection Machine implementations for various optimization problems. He intends to continue working on both practical and theoretical aspects of parallel computation.

Su-Ming Wu

Working with Tardos, Wu has developed an $O(n^2)$ algorithm for the problem of finding two edge-disjoint paths in a graph [161]. The basis for the algorithm is a graph-theoretic proof of P.D. Seymour (Bell Communications Research Laboratory, New Jersey).

4 Annotated References

[1] Y. Afek, B. Awerbuch, and H. Moriel. Overhead of resetting a communication protocol is independent of the size of the network. May 1989. Unpublished manuscript.

[2] Y. Afek and E. Gafni. End-to-end communication in unreliable networks. In *Proceedings of the 7^{th} Annual ACM Symposium on Principles of Distributed Computing, Toronto, Ontario, Canada*, pages 131–148, ACM SIGACT and SIGOPS, ACM, 1988.

[3] A. Aggarwal, M. Klawe, S. Moran, P. Shor, and R. Wilber. Geometric applications of a matrix-searching algorithm. *Algorithmica*, 2:209–233, 1987.

[4] A. Aggarwal and D. Kravets. A linear time algorithm for finding all farthest neighbors in a convex polygon. *Info. Proc. Lett.*, 31:16–20, 1989.

> Aggarwal et al. [A. Aggarwal, M.M. Klawe, S. Moran, P. Shor, R. Wilber, "Geometric Applications of a Matrix-Searching Algorithm," Algorithmica, Vol. 2, 1987, pp. 195-208] showed how to compute in $O(n)$ time *one farthest neighbor* for every vertex of a convex n-gon. In this note we extend this result to obtain a linear time algorithm for finding *all farthest neighbors* for every vertex of a convex polygon. Our algorithm yields a linear time solution to the *symmetric all-farthest neighbors* problem for simple polygons, thereby settling an open question raised by Toussaint in 1983 [G.T. Toussaint, "The Symmetric All-Farthest Neighbor Problem," Comp. and Math. Applications, Vol. 9, No. 6, 1983, pp. 747-753].

[5] A. Aggarwal and J. Park. Notes on searching in multidimensional monotone arrays. In 29^{th} *Symp. Found. Computer Sci.*, pages 597–512, IEEE, 1988.

[6] A. Aggarwal and J. Park. Sequential searching in multidimensional monotone arrays. 1989. Submitted for publication.

> A two-dimensional array is called *monotone* if the maximum entry in its i-th row lies directly below or to the right of the maximum entry in its $(i-1)$-st row. (If a row has several maxima, then we take the leftmost one.) A two-dimensional array is called *totally monotone* if every 2×2 subarray (*i.e.*, every 2×2 minor) is monotone. Totally monotone arrays were introduced by Aggarwal, Klawe, Moran, Shor, and Wilber, who showed that several problems in computational geometry could be reduced to the problem of finding the maximum entry in each row of a totally monotone array. They also gave a sequential algorithm for computing the row maxima of an $n \times n$ totally monotone array in $\Theta(n)$ time. In this paper, we generalize the notion of two-dimensional totally monotone arrays to multidimensional arrays, present sequential algorithms for finding maxima in such arrays, and exhibit a wide variety of problems (involving computational geometry, dynamic programming, and VLSI river routing) that can be solved efficiently using these array-searching algorithms.

[7] D. Angluin. Learning regular sets from queries and counterexamples. *Information and Computation*, 75:87–106, Nov. 1987.

[8] A. Appel and T. Jim. Continuation-passing, closure-passing style. In 16^{th} *Symposium on Principles of Programming Languages*, ACM, 1989.

[9] B. Awerbuch. On the effects of feedback in dynamic network protocols. In 29^{th} *Annual Symposium on Foundations of Computer Science*, pages 231–245, IEEE, Oct. 1988.

Describes an asynchronous adaptor, which converts any protocol for a static asynchronous network to work on a dynamic asynchronous network.Starts to embed results in a more general framework that is said to include open-loop systems, Finn's protocol, and the earlier local adaptor due to Afek, Awerbuch, and Gafni, by considering the effects of feedback.

[10] B. Awerbuch. Distributed shortest paths algorithms. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, pages 230–240, ACM SIGACT, ACM, May 1989.

[11] B. Awerbuch, A. Bar-Noy, N. Linial, and D. Peleg. Compact distributed data structures for adaptive network routing. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, pages 230–240, ACM SIGACT, ACM, May 1989.

[12] B. Awerbuch, A. Goldberg, M. Luby, and S. Plotkin. Network decomposition and locality in distributed computation. May 1989. Unpublished manuscript.

[13] B. Awerbuch, O. Goldreich, and A. Herzberg. A quantitative approach to dynamic networks. May 1989. Unpublished manuscript.

[14] B. Awerbuch, O. Goldreich, D. Peleg, and R. Vainish. *On the Message Complexity of Broadcast: Basic Lower Bound.* Technical Memo TM-365, MIT Lab. for Computer Science, July 1988. (Accepted for publication at Journal of the ACM.).

This paper concerns the message complexity of broadcast in arbitrary point-to-point communication networks.*Broadcast* is a task initiated by a*single* processor that wishes to convey a message to all processors in the network.We assume the widely accepted model of communication networks, in which each processor initially knows the identity of its neighbors, but does not know the entire network topology.Although it seems obvious that the number of messages required for broadcast in this model equals the number of links, no proof of this basic fact has been given before.

We show that the message complexity of broadcast depends on the exact complexity measure. If messages of unbounded length are counted at unit cost, then broadcast requires $\Theta(|V|)$ messages, where V is the set of processors in the network. We prove that if one counts messages of*bounded length* then broadcast requires $\Theta(|E|)$ messages, where E is the set of edges in the network.

The same results hold for the construction of spanning trees, and various other global tasks.

[15] B. Awerbuch, Y. Mansour, and N. Shavit. Polynomial end-to-end communication. In 30^{th} *Annual Symposium on Foundations of Computer Science*, IEEE, 1989. Submitted.

A dynamic communication network is one in which links may repeatedly fail and recover.In such a network, though it is impossible to establish a path of unfailed links, reliable communication is possible, if there is no cut of permanently failed links between a sender and receiver. We consider the basic task of of *end-to-end* communication, that is, delivery in finite time, of data items generated on-line at the sender, to the receiver, inorder without duplication, or omission. The best known previous solutions to this problem have exponential complexity. Moreover, it has been conjectured that a polynomial solution is impossible. This paper disproves this conjecture, presenting the first polynomial end-to-end protocol.The protocol uses techniques adopted from shared memory algorithms, and introduces novel techniques for fast load balancing in communication networks.

[16] B. Awerbuch and M. Sipser. Dynamic networks are as fast as static networks. In 29^{th} *Annual Symposium on Foundations of Computer Science*, pages 206–220, IEEE, Oct. 1988.

Presents a synchronous adaptor, which converts any protocol that runs on a static synchronous network to work on a dynamic asynchronous network. Based on a dynamic sychronizer.

[17] L. Babai, N. Nisan, and M. Szegedy. Multiparty protocols and pseudorandom generators for logspace. In *Proc. of the 21th STOC Symposium*, ACM, 1989.

Let $f(x_1, \ldots, x_k)$ be a Boolean function that k parties wish to collaboratively evaluate. The i'th party knows each input argument except x_i; and each party has unlimited computational power. They share a blackboard, viewed by all parties, where they can exchange messages. The objective is to minimize the number of bits written on the board.

We prove lower bounds of the form $\Omega(nc^{-k})$, for the number of bits that need to be exchanged in order to compute some (explicitly given) functions in P. Our bounds hold even if the parties only wish to have a 1% advantage at guessing the value of f on random inputs. We then give several applications of our lower bounds.

Our first application is a pseudorandom generator for Logspace.We explicitly construct (in polynomial time) pseudorandom sequences of length n from a random seed of length $\exp(c\sqrt{\log n})$ that no Logspace Turing machine will be able to distinguish from truly random sequences. As a corollary we give an explicit construction of universal traversal sequence of length $\exp(\exp(c\sqrt{\log n}))$ for arbitrary undirected graphs on n vertices.

We then apply the multiparty protocol lower bounds to derive several new time-space tradeoffs. We give a tight time-space tradeoff of the form $TS = \Theta(n^2)$, for general, k-head Turing-Machines; the bounds hold for a function that can be computed in linear time and constant space by a $k+1$-head Turing Machine.We also give a new length-width tradeoff for oblivious branching programs; in particular our bound implies new lower bounds on the size of arbitrary branching programs, or on the size of boolean formulas (over an arbitrary finite base).

[18] J. M. Barzdin and R. V. Frievald. On the prediction of general recursive functions. *Soviet Mathematics Doklady*, 13:1224–1228, 1972.

[19] D. Beaver and S. Goldwasser. Multi-party computation with faulty majority. March 1989. unpublished.

[20] M. Bellare and S. Goldwasser. New paradigms for digital signature schemes and message authentication based on non-interactive zero knowledge proofs. March 1989. unpublished.

[21] M. Bellare and S. Micali. Non-interactive oblivious transfer and applications. March 1989. unpublished.

[22] M. Bellare, S. Micali, and R. Ostrovsky. Parallelizing zero knowledge proofs and perfect completeness zero knowledge. Apr. 1989. unpublished.

[23] S. Ben-david, G. M. Benedek, and Y. Mansour. The passive student is really weaker. In *COLT*, 1989. Submitted.

We present a systematic framework for classifying comparing and defining models of computational learnability. Apart from the obvious uniformity parameters we present a novel "passiveness" notion that captures the difference between"Guess and Test" learning algorithms and learnability notions for which consistency with the samples guarantees success.

[24] M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway. Everything provable is provable in zero-knowledge. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

This paper shows how to prove IP in computational zero-knowledge assuming a 1-to-1 one-way function; and how to prove IP in perfect zero-knowlede under the "envelope" model for bit commitment.

[25] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Multi-prover interactive proof systems, removing intractibility assumptions. In *Proceedings of the* 20^{th} *STOC*, ACM, 1988.

[26] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Efficient identification schemes using two prover interactive proofs. March 1989. unpublished.

[27] B. Berger. *Data Structures for Removing Randomness.* Technical Report TR-436, MIT Lab. for Computer Science, Dec. 1988.

Karp, Wigderson, and Luby offer two main techniques for removing randomness from parallel algorithms: trying all sample points and zeroing in on a good sample point. We present a survey of these three papers, focusing on the above techniques. Furthermore, we provide new results which extend the technique of zeroing in on a good sample point to work for $(\log n)$-wise independence. We conclude by giving an application of our new technique to parallel edge coloring.

[28] B. Berger and J. Rompel. A better performance guarantee for approximate graph coloring. *Algorithmica*, 1988.

Approximate graph coloring takes as input a graph and returns a legal coloring which is not necessarily optimal. We improve the performance guarantee, or worst-case ratio between the number of colors used and the minimum number of colors possible, to $O(n(\log\log n)^3/(\log n)^3)$, an $O(\log n/\log\log n)$ factor better than the previous best known result.

[29] B. Berger and J. Rompel. Simulating $(\log^c n)$-wise independence in nc. In 30^{th} *Symp. on Found. of Computer Sci.*, IEEE, 1989. To appear. Also appeared as technical report MIT/LCS/TR-435.

We develop a general framework for removing randomness from randomized NC algorithms whose analysis uses only polylogarithmic independence. Previously no techniques were known to determinize those RNC algorithms depending on more than constant independence. One application of our techniques is an NC algorithm for the set discrepancy problem, which can be used to obtain many other NC algorithms, including a better NC edge coloring algorithm. As another application of our techniques, we provide an NC algorithm for the hypergraph coloring problem.

[30] B. Berger, J. Rompel, and P. Shor. Efficient nc algorithms for set cover with applications to learning and geometry. In 30^{th} *Symp. on Found. of Computer Sci.*, IEEE, 1989. To appear. Also appeared as technical report MIT/LCS/TR-444.

In this paper we give the first NC approximation algorithms for the unweighted and weighted set cover problems. Our algorithms use a linear number of processors and give a cover that has at most $\log n$ times the optimal size/weight, thus matching the performance of the best sequential algorithms. We apply our set cover algorithm to learning theory, giving an NC algorithm to learn the concept class obtained by taking the closure under finite union or finite intersection of any concept class of finite VC-dimension which has an NC hypothesis finder. In addition, we give a linear-processor NC algorithm for a variant of the set cover problem first proposed by Chazelle and Friedman, and use it to obtain NC algorithms for several problems in computational geometry.

[31] B. Berger and P. Shor. *Tight bounds for the acyclic subgraph problem.* Technical Report TR-413, MIT Lab. for Computer Science, June 1989. Submitted for publication.

Given a directed graph $G = (V, A)$, the *acyclic subgraph problem* is to find a subset A' of the arcs such that $G' = (V, A')$ is acyclic and A' has maximum cardinality. In this paper, we present polynomial-time and RNC algorithms which, when given any graph G without 2-cycles, find an acyclic subgraph of size at least $(1/2 + \Omega(1/\sqrt{\Delta(G)}))|A|$, where $\Delta(G)$ is the maximum degree of G. This bound is tight, in terms of $|A|$, since there exists a class of graphs that have an acyclic subgraph of size at most $(1/2 + O(1/\sqrt{\Delta(G)}))|A|$. These algorithms are based on a new technique which enables an improvement over the naive algorithms that find an acyclic subgraph of size at least $\frac{1}{2}|A|$, even for graphs which contain no 2-cycles. Furthermore, we show that considering graphs without 2-cycles is sufficient since 2-cycles can be dealt with optimally.

[32] G. Berry, P. Curien, and J. Lévy. Full abstraction for sequential languages: the state of the art. In M. Nivat and J. C. Reynolds, editors, *Algebraic Methods in Semantics*, pages 89–132, Cambridge University Press, 1985.

[33] D. Bertsimas and M. Grigni. On the space-filling curve heuristic for the euclidean traveling salesman problem. *Operations Research Letters*, 1989. To appear.

Bartholdi and Platzman proposed the spacefilling curve heuristic for the Euclidean Traveling Salesman Problem and proved that their heuristic returns a tour within an $O(\lg n)$ factor of optimal length. They conjectured that the worst-case ratio is in fact $O(1)$.In this note we exhibit a counterexample showing the $O(\lg n)$ upper bound is tight.

[34] B. Bloom, S. Istrail, and A. R. Meyer. Bisimulation can't be traced: preliminary report. In 15^{th} *Symp. Principles of Programming Languages*, pages 229–239, ACM, 1988. Final version in preparation for journal submission.

Shows how hard it is to reconcile the difference between Milner's bisimulation and Hoare-style trace congruence for synchronous CCS/CSP-like languages.

[35] B. Bloom and A. R. Meyer. Experimenting with process equivalence. Jan. 1989. 12 page extended abstract, to be submitted.

We investigate the foundations of the relation of bisimulation (a.k.a. "observational equivalence") which underlies the theory of CCS-like processes. Bisimulation is defined by an infinitary game; we consider the question of finding simpler and preferably finite observations which explain bisimulation. Following the suggestions of Hoare and Milner, we express these observations as experiments on black boxes. We consider several varieties of experiments, and discover that it is surprisingly difficult to obtain bisimulation as a plausible experimental equivalence. In our attempts to capture bisimulation, we

and other researchers have discovered a variety of intermediate equivalences. Several of these equivalences coincide, in a relation we call *ready simulation* which may be of independent interest.

[36] B. Bloom and A. R. Meyer. A remark on the bisimulation of probabilistic processes. In *Logic at Botic '89, Proceedings*, pages 26–40, Volume 363 of *Lect. Notes in Computer Sci.*, July 1989.

[37] B. Bloom and J. G. Riecke. LCF should be lifted. In *Proc. Conf. AMAST*, pages 133–136, 1989.

When observing termination of closed terms at all types in Plotkin's interpreter for PCF, the standard cpo model is not adequate. We define a new model with *lifted* functional types and prove its adequacy for this notion of observation. We prove that with the addition of a parallel conditional and a convergence testing operator to the language, the model becomes fully abstract; with the addition of an existential-like operator, the language becomes universal. Using the model as a guide, we develop a sound logic for the language.

[38] A. Blum. *On the Computational Complexity of Training Simple Neural Networks.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by Ron Rivest.

Presents NP-completeness results for training a variety of simple neural networks.Also shows that in certain cases, the complexity problems may be bypassed by suitable re-encoding of the input and by choosing an appropriate network on which to train.

[39] A. Blum. An $\tilde{O}(n^{0.4})$-approximation algorithm for 3-coloring (and improved approximation algorithms for k-coloring). In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, Seattle, Washington, May 1989.

Improves the provious best bound of $O(n^{0.5}/\sqrt{\log n})$ colors, breaking the "$\tilde{O}(n^{0.5})$" barrier.

[40] A. Blum and R. Rivest. Training a 3-node neural network is NP-Complete. In *Advances in Neural Information Processing Systems 1*, pages 494–501, Morgan Kaufmann, 1988. Also presented at the 1988 Workshop on Computational Learning Theory.

We present an proof of NP-completeness for training a very simple neural network.The proof also shows it to be NP-complete to decide whether two sets of boolean vectors in n-dimensional space can be separated by two hyperplanes.

[41] M. Blum, P. Feldman, and S. Micali. Noninteractive zero-knowledge proofs and their applications. In *Proceedings of the* 20^{th} *STOC*, ACM, 1988.

[42] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM J. Comput.*, 13(4):850–864, 1984.

[43] R. B. Boppana and M. Sipser. The complexity of finite functions. 1989. To appear in the Handbook of Theoretical Computer Science.

[44] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Comp. and Syst. Sci.*, 37:156–189, 1988.

> We describe very general techniques to prove the knowledge of some piece of data in a way that discloses the minimal amount of knowledge about it. This paper is a follow up to [46, 45].

[45] G. Brassard and C. Crépeau. Non-transitive transfer of confidence: a perfect zero-knowledge interactive protocol for SAT and beyond. In 27^{th} *Symp. of Found. of Computer Sci.*, pages 188–195, IEEE, 1986.

> Under the hypothesis that factoring is hard during the life time of the protocol, a new proof technique similar to [46] is introduced. This technique has the amazing property of giving total security on the privacy of the proof.

[46] G. Brassard and C. Crépeau. Zero-knowledge simulation of boolean circuits (extended abstract). In A. M. Odlyzko, editor, *Advances in Cryptology: Proceedings of Crypto '86*, pages 223–233, Volume 263 of *Lect. Notes in Computer Sci.*, Springer-Verlag, 1986.

> Under the hypothesis that factoring is hard, a technique is given to do efficient simulation of boolean circuits in zero-knowledge. The simulation of these circuits leads to zero-knowledge proofs for every languages in NP and beyond.

[47] G. Brassard and C. Crépeau. Sorting out zero-knowledge. In *Advances in Cryptology: Proceedings of Eurocrypt '89, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

> We introduce new formalism for the notion of interactive arguments and give definiton of zero-knowledge in a setting where the verifier is not necessarily time bounded.

[48] G. Brassard, C. Crépeau, and J. Robert. All-or-nothing disclosure of secrets (extended abstract). In A. M. Odlyzko, editor, *Advances in Cryptology: Proceedings of Crypto '86*, pages 234–238, Volume 263 of *Lect. Notes in Computer Sci.*, Springer-Verlag, 1986.

> Under cryptographic assumption, a protocol for the all-or-nothing disclosureof secrets problem is suggested.

[49] G. Brassard, C. Crépeau, and J. Robert. Information theoretic reductions among disclosure problems. In 27^{th} *Symp. of Found. of Computer Sci.*, pages 168–173, IEEE, 1986.

The all-or-nothing disclosure of secrets suggested in [48] is shown to be reduceable to a serie of simpler problems (ANNBP, AN2BP, 2BP). One of these simpler problems (AN2BP) is equivalent to "one-out-of-two oblivious transfer".

[50] G. Brassard, C. Crépeau, and M. Yung. Everything in NP can be argued in perfect zero-knowledge in a constant number of rounds. In 16^{th} *ICALP, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

Shown is how to achieve interactive arguments (notion similar to interactive proofs) that are perfect zero-knowledge with only a constant number of interactions.

[51] J. Brock and W. Ackerman. Scenarios: a model of non-determinate computation. In G. Goos and J. Hartmanis, editors, *Formalization of Programming Concepts*, pages 252–259, Volume 107 of *Lect. Notes in Computer Sci.*, Springer-Verlag, 1981.

[52] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms.* McGraw-Hill/MIT Press, 1989.

[53] S. Cosmadakis. Computing with recursive types (extended abstract). In 4^{th} *Symp. Logic in Computer Science*, IEEE, 1989. To appear, July.

[54] C. Crépeau. Equivalence between two flavours of oblivious transfers (abstract). In C. Pomerance, editor, *Advances in Cryptology: Proceedings of Crypto '87*, pages 350–354, Volume 293 of *Lect. Notes in Computer Sci.*, Springer-Verlag, 1987.

In this paper we show that the two known versions of obliviuos transfers (the standard oblivious transfer and the one-out-of-two oblivious transfer) are computationally equivalent.

[55] C. Crépeau. Verifiable disclosure of secrets and applications. In *Advances in Cryptology: Proceedings of Eurocrypt '89, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

A new primitive cryptographic protocol is introduced: verifiable 1-out-of-2 oblivious transfer. Shown is how to build a secure protocol for this primitive from standard 1-out-of-2 oblivious transfer and its applications.

[56] C. Crépeau and J. Kilian. Achieving oblivious transfer using weakened security assumptions. In 28^{th} *Symp. on Found. of Computer Sci.*, pages 42–52, IEEE, 1988.

A set of foundamental primitives sufficient to achieve Oblivious Transfer are presented. Reductions to primitives such as noisy channels and quantum transfer are provided in this paper.

[57] C. Crépeau and J. Kilian. Weakening security assumptions and oblivious transfer. In *Advances in Cryptology: Proceedings of Crypto '88, Lect. Notes in Computer Sci.*, Springer-Verlag, 1988. To appear.

We show that Oblivious Transfer can be achieved from some very weak version of this protocol and even more foundamental primitives such as noisy channels.

[58] P. Curien. *Categorical Combinators, Sequential Algorithms and Functional Programming.* John Wiley and Sons, 1986.

[59] A. De Santis, S. Micali, and P. Persiano. Bounded interaction zero-knowledge proofs. 1987. unpublished.

[60] P. Elias. Error-free coding. *IEEE Transactions on Information Theory*, PGIT-4:29–37, 1954. More accessible on pp. 39-47 in "Key Papers in the Development of Coding Theory", E. R. Berlekamp, Ed., IEEE Press, 1974.

[61] P. Elias. Zero error capacity under list decoding. *IEEE Transactions on Information Theory*, 34:1070–1074, 1988.

Shannon defined the zero error capacity C_0 of a channel and C_{0F}, its zero error capacity when noisless feedback of output symbols to the transmitter is available.He gave an algorithm for finding C_{0F}. Finding a general algorithm for C_0 is very hard and still open.Define the corresponding zero error list-of-L capacities $C_0(L)\,, C_{0F}(L)$, where the receiver lists L possible messages and is in error only if the correct message is not on the list.Both are nondecreasing in L: Shannon's case is $L = 1$ .The principal results are summarized by $\lim_{L\to\infty} C_{0F}(L) = C_{0F}(I-1) = \lim_{L\to\infty} C_0(L)$, where I is the size of the channel input alphabet.A simple algorithm finds $C_{0F}(I-1)$ and therefore the limit of $C_0(L)$ for large L.New combinatorial problems arise in finding $C_0(L)$ for finite L.

[62] P. Elias. *Error-Correcting Codes for List Decoding.* Technical Report TM-381, MIT Lab. for Computer Science, Feb. 1989. Submitted for publication.

In the list-of-L decoding of a block code the receiver of a noisy sequence lists L possible transmitted messages, and is in error only if the correct message is not on the list.This paper considers (n, e, L) codes, which correct all sets of e or fewer errors in a block of n bits under list-of-L decoding.New geometric relations between the number of errors corrected under list-of-1 decoding and the (larger) number corrected under list-of-L decoding of the same code lead to new lower bounds on the maximum rate of (n, e, L) codes.They show that a jammer who can change a fixed fraction $p < 1/2$ of the bits in an n-bit linear block code cannot prevent reliable communication at a positive rate using list-of-L decoding for sufficiently large n and an $L \leq n$.The new bounds are stronger for small n but weaker for fixed e/n in the limit of large n and L than known random coding bounds.

[63] M. D. Ernst. *Adequate Models for Recursive Program Schemes.* Bachelor's thesis, MIT Dept. of Electrical Engineering and Computer Science, May 1989. Supervised by Albert R. Meyer.

Adequacy relates the operational and denotational meanings of a term; it states that for any term of base type, the operational and denotational meanings are identical. Adequacy is typically proved in the continuous frame. This is a pedagogically questionable step; in order to prove adequacy (or some other property) of a pair of semantics, it would be desirable to show the property directly, without introducing superfluous notions. This difficulty is particularly acute because, in general, not all monotone functions are continuous.

This thesis attempts to work out the concept of adequacy for a class of monotone first-order recursive program schemes, using Vuillemin and Manna's method of "safe" computation rules. The attempt is very nearly successful, but at a crucial point the fact that the scheme-definable functions are, in fact, continuous as well as monotone must be used.

[64] L. Fortnow. *Complexity-Theoretic Aspects of Interactive Proof Systems.* Ph.D. thesis, MIT, 1989.

In 1985, Goldwasser, Micali and Rackoff formulated interactive proof systems as a tool for developing cryptographic protocols. Indeed, many exciting cryptographic results followed from studying interactive proof systems and the related concept of zero-knowledge. Interactive proof systems also have an important part in complexity theory merging the well established concepts of probabilistic and nondeterministic computation. This thesis will study the complexity of various models of interactive proof systems.

A perfect zero-knowledge interactive protocol convinces a verifier that a string is in a language without revealing any additional knowledge in an information theoretic sense. This thesis will show that for any language that has a perfect zero-knowledge proof system, its complement has a short interactive protocol. This result implies that there are not any perfect zero-knowledge protocols for NP-complete languages unless the polynomial-time hierarchy collapses. Thus knowledge complexity can show a language is easy to prove.

Interesting models of interactive proof systems arise by restricting the power of the verifier. This thesis examines the proof systems with a verifier required to run in logarithmic space as well as polynomial time. Relationships with circuit complexity and log-space Turing Machines are developed.

We can increase the power of interactive proof systems by allowing many provers that can not communicate among themselves during the protocol. This thesis shows the equivalence between this multi-prover model and probabilistic Turing machines with an untrustworthy oracle. We additionally give an oracle under which co-NP does not have multi-prover interactive protocols. This result implies an earlier result showing an oracle where co-NP does not have standard interactive protocols.

Another natural model occurs when the verifier has only linear time. Towards this direction, this thesis examines probabilistic machines and linear time. We show an oracle under which linear time probabilistic Turing machines can accept all BPP languages, an unusual collapse of a complexity

time hierarchy. We exhibit many other related relativized results. Finally we show probabilistic linear time does not contain all languages accepted by interactive proof systems.

[65] L. Fortnow and M. Sipser. Probabilistic computation and linear time. In 21^{st} *Symposium on Theory of Computing*, ACM, 1989. To appear.

We show some relativized results involving probabilistic and linear time computation.Under an appropriate oracle we show BPP is contained in probabilistic linear time, a rare collapse of a complexity time hierarchy.We also show that with an oracle Δ_2^P is also contained in probabilistic linear time and that BPP has linear size circuits.Finally, we note that probabilistic linear time can not contain both NP and BPP; implying that there are languages solvable by interactive proof systems that can not be solved in probabilistic linear time.

[66] M. Foster and R. I. Greenberg. Lower bounds on the area of finite-state machines. *Info. Proc. Lett.*, 30(1):1–7, Jan. 1989.

There are certain straightforward algorithms for laying out finite-state machines.This paper shows that these algorithms are optimal in the worst case for machines with fixed alphabets.That is, for any s and k, there is a deterministic finite-state machine with s states and k symbols such that *any* layout algorithm requires $\Omega(ks \lg s)$ area to lay out its realization. Similarly, any layout algorithm requires $\Omega(ks^2)$ area in the worst case for nondeterministic finite-state machines with s states and k symbols.

[67] M. Fredman and J. Komlos. On the size of separating systems and perfect hash functions. *SIAM J. Algebraic Discr. Meth.*, 5:61–68, 1984.

[68] J. Fried. Broadband module design: cost/performance tradeoffs. Lecture given at International Workshop on Physical Design of Broadband Switching and Multiplexing Equipment, Apr. 1989.

Provides an analytic model of the performance and cost of packet-routing modules for use in broadband networks.

[69] J. Fried. A VLSI chip set for burst and ATM switching. In *International Communications Conference*, 1989.

Describes the architecture, programming, and design of programmable switch intended for use in distributed computing and telecommunications networks.This design includes three custom chips, including some novel circuits for very fast locking of shared data structures

[70] J. Fried. *VLSI Processor Design for Communications Networks.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by C.E. Leiserson. Also appears as an MIT VLSI memo.

Two 'case study' designs for programmable switches useful in communications networks within distributed and parallel systems.

[71] J. Fried, D. Ghosh, and J. Daly. A novel content-addressable memory circuit. *Electronics Letters*, 1989.

A circuit design which implements a very fast CAM (4-ns match time) using less area than the design reported in [70]

[72] J. Fried and P. Kubat. Reliability models for facilities switching. 1989. Submitted to IEEE Transactions on Reliability.

[73] J. Fried and B. Kuszmaul. NAP (no ALU processor): the great communicator. In *Frontiers of Massively Parallel Computation*, 1988. An extended version of this paper has been submitted for publication in the Journal of Parallel and Distributed Computing.

Describes the architecture and programming of a specialized VLSI processor useful as a routing node within a wide variety of low-degree interconnection networks.

[74] A. V. Goldberg, S. Plotkin, D. B. Shmoys, and E. Tardos. Interior-point methods in parallel computation. 1989. submitted for publication.

In this paper, interior-point methods for linear programming, developed in the context of sequential computation, are used to obtain a parallel algorithm for the bipartite matching problem. The algorithm runs in $O^*(\sqrt{m})$ time, where n and m denote the number of nodes and edges of the input graph and an algorithm is said to run in $O^*(f(n))$ time if it runs in $O(f(n)\log^k(n))$ time for some constant k. The results extend to the weighted bipartite matching problem and to the zero-one minimum-cost flow problem, yielding $O^*(\sqrt{m}\log C)$ algorithms, where it is assumed that the weights are integers in the range $[-C \ldots C]$ and $C > 1$. These results improve previous bounds on these problems and introduce interior-point methods to the context of parallel algorithm design.

[75] A. V. Goldberg, E. Tardos, and R. E. Tarjan. Network flow algorithms. In *Flows, Paths and VLSI-layout*, Springer Verlag, 1989. To appear.

Network flow problems are central problems in operations research, computer science, and engineering and they arise in many real world applications.Starting with early work in linear programming and spurred by the classic book of Ford and Fulkerson the study of such problems has led to continuing improvements in the efficiency of network flow algorithms. In spite of the long history of this study, many substantial results have been obtained within the last several years. In this survey we examine some of these recent developments and the ideas behind them.

The chapters are: Introduction, Preliminaries, The maximum flow problem, The minimum-cost circulation problem: cost-scaling, Strongly polynomial algorithms based on cost-scaling, Capacity-scaling algorithms, The generalized flow problem.

[76] S. A. Goldman. A space efficient greedy triangulation algorithm. *Info. Proc. Lett.*, 1989. To appear in May. Earlier version available as MIT/LCS/TM–366.

We show that the greedy triangulation of n points in the plane can be computed in $O(n^2 \log n)$ time and $O(n)$ memory and storage. In particular we show that by maintaining a generalized Delaunay triangulation, the next edge to add to the greedy triangulation can be found in $O(n)$ time.Furthermore, if the generalized Delaunay triangulation of a simple polygon could be computed in $O(n)$ time, our algorithm would compute the greedy triangulation in $O(n^2)$ time.

[77] S. A. Goldman and R. L. Rivest. Mistake bounds and efficient halving algorithms. 1989. Submitted.

We present a mistake bound model for evaluating polynomial-time prediction algorithms for concept learning.This model is particularly useful for concept classes with polynomial sized instance spaces.As an example we study the concept class k–BP, the class of n–bit patterns with at most k alternations.For this class we give an algorithm that only records mistakes and makes at most $k + k \lg(n-1)$ mistakes for any query sequence.

Since good mistake bounds are often obtained by the halving algorithm, we present techniques for obtaining efficient implementations of the halving algorithm.We discuss how mistake bounds are affected by the several methods for selecting the sequence of instances.Furthermore, we prove that in general the mistake bounds do not depend on whether the learner knows the size of the unknown target concept.

We define an *approximate halving algorithm* to be an algorithm that predicts in agreement with at least some constant fraction of the remaining concepts from $\mathcal{C}$.We prove that such an algorithm makes $O(\lg \mathcal{C})$ mistakes, and show how to use fully polynomial randomized approximation schemes (fpras) to efficiently implement randomized versions of such algorithms.These techniques are applied to the problem of learning a total order on n elements. We use a fpras due to [125] for counting extensions of a partial order to obtain an algorithm making $O(n \lg n)$ mistakes (except for an exponentially small fraction of its executions) when an adversary selects the query sequence.(The small probability of making $\omega(n \lg n)$ mistakes is taken over the coin flips of the learning algorithm and does not depend on the query sequence selected by the adversary.)

[78] S. A. Goldman, R. L. Rivest, and R. E. Schapire. Learning binary relations and total orders. 1989. To appear.

We study the problem of designing polynomial prediction algorithms to learn a binary relation.We represent the relation as an $n \times m$ binary matrix that is restricted to have at most k distinct row types.For this concept class the instances correspond to entries of the matrix and thus the instance space is only polynomially sized.As we shall see, the mistake bound model is particularly useful for studying such classes.However instead of just considering when an adversary selects the query sequence, we consider when the query sequence is selected by a helpful teacher or the learner.We also extend the mistake bound model to accommodate randomized algorithms.

For the concept class of binary matrices, we present an algorithm making at most $mk + (n - k) \lg k$ mistakes when the learner chooses the query sequence.We present an algorithm for $k = 2$ that makes at most $2m + n - 1$ mistakes when an adversary chooses the query sequence. For arbitrary k we present an algorithm making at most $km + \frac{k-1}{2k}n^2$ mistakes against an adversary selected query sequence.We also use the existence of projective geometries to prove an $\Omega(n^{3/2})$ lower bound on the worst case number of mistakes made by a large class of algorithms when an adversary chooses the query sequence.Finally, we describe a simple prediction rule that achieves an expected mistake bound of $O(k(n\sqrt{m} + m))$ when the query sequence is chosen at random.

Since good mistake bounds are often obtained by the halving algorithm, we also present techniques for obtaining efficient implementations of the halving algorithm.We illustrate these techniques for the problem of learning a total order on n elements.An instance corresponds to a comparison between two elements, and thus here too the instance space is polynomially sized.We use a fully polynomial randomized approximation scheme due to [125] for counting extensions of a partial order to obtain an algorithm making $O(n \lg n)$ mistakes (except for an exponentially small fraction of its executions) when an adversary selects the query sequence. Finally, we discuss how a majority algorithm (i.e.the halving algorithm) may be used to construct an efficient counting algorithm.

[79] O. Goldreich, A. Herzberg, and Y. Mansour. Source to destination communication in the presence of faults. In *8th Annual ACM Symposium on Principles of Distributed Computing*, 1989. To appear.

We present a protocol for reliable communication between twoprocessors via anunreliable, andpossibly evenmalicious, communicationmedia. Reliable communication means that all messages are accepted in thesame orderassent,with nomodifications,omissions,insertions or duplications. Our protocol isresilient to processorcrashes, in which theentire memoryof theprocessor iserased. The protocol is applicablebothtothe*data link layer* and to the *data transport layer* of the ISO model.

Our approach is probabilistic. Namely, for the worst case behavior of the underlying unreliable channel, our protocol guarantees reliable communica-

tion with very high probability. The probabilistic approach is justified in light of impossibility results concerning deterministic communication protocols.

[80] O. Goldreich and L. Levin. A hard-core predicate for all one-way functions. In 21^{th} *Symp. Theory of Computing*, pages 25–32, ACM, 1989.

[Blum Micali 82] discovered permutations f with "hard-core" predicates $b(x)$ which cannot be efficiently guessed from $f(x)$ with a noticeable correlation.Both b, f are easy to compute. [Yao 82] modifies any one-way permutation f into f^*, which has a hard-core predicate. Its security may be lower than any constant power of the security of f and is too small for practical applications. We prove that most linear predicates are hard-cores for every one-way function and have almost the same security. The result extends to multiple (up to the logarithm of security) hidden bits and has wide applicability to pseudorandomness, cryptography, etc.

[81] O. Goldreich, Y. Mansour, and M. Sipser. Interactive proof systems: provers that never fail and random selection. In *Proceedings of the* 27^{th} *FOCS*, pages 449–462, IEEE, 1987.

[82] S. Goldwasser and M. Sipser. Arthur-merlin games verses interactive proof systems. In *Proceedings of the* 18^{th} *STOC*, ACM, 1986.

[83] R. I. Greenberg. *Area-Universal Networks.* VLSI Memo 524, Massachusetts Institute of Technology, 1989.

An area-universal network is one which can efficiently simulate any other network of comparable area.This paper extends previous results on area-universal networks in several ways.First, it considers the size (amount of attached memory) of processors comprising the networks being compared.It shows that an appropriate universal network of area $\Theta(A)$ built from processors of size $\log A$ requires only $O(\log^2 A)$ slowdown *in bit-times* to simulate any network of area A, without any restriction on processor size or number of processors in the competing network.Furthermore, the universal network can be designed so that any message traversing a path of length d in the competing network need follow a path of only $O(d + \log A)$ length in the universal network.Thus, the results are almost entirely insensitive to removal of the unit wire delay assumption used in previous work.This paper also derives upper bounds on the slowdown required by a universal network to simulate a network of larger area and shows that all of the simulation results are valid even without the usual assumption that computation and communication of the competing network proceed in separate phases.

[84] R. I. Greenberg, A. T. Ishii, and A. L. Sangiovanni-Vincentelli. MulCh: A multi-layer channel router using one, two, and three layer partitions. In *IEEE International Conference on Computer-Aided Design (ICCAD-88)*, pages 88–91, IEEE Computer Society Press, 1988.

Multi-layer routing is becoming an important problem in the physical design of integrated circuits as technology evolves towards several layers of metallization.Several channel routers for three layers of interconnect have been proposed, but only one, CHAMELEON, has been implemented to accept specification of an arbitrary number of layers.CHAMELEON is based on a strategy of decomposing the multi-layer problem into two and three layer problems in which one of the layers is reserved primarily for vertical wire runs and the other layer(s) for horizontal runs.In some situations, however, it is advantageous to consider also layers that allow the routing of entire nets, using both horizontal and vertical wires. MULCH is a multi-layer channel router that extends the algorithms of CHAMELEON in this direction.MULCH can route channels with any number of layers and automatically chooses a good assignment of wiring strategies to the different layers. In test cases, MULCH shows significant improvement over CHAMELEON in terms of channel width, net length, and number of vias.

[85] M. Grigni and D. Peleg. *Tight Bounds on Minimum Broadcast Networks.* Technical Memo TM-374, MIT Lab. for Computer Science, Dec. 1988.

A *broadcast graph* is an n-vertex communication network that supports a broadcast from any one vertex to all other vertices in optimal time $\lceil \lg n \rceil$, given that each message transmission takes one time unit and a vertex participates in at most one transmission per time step.This paper establishes tight bounds for $B(n)$, the minimum number of edges of a broadcast graph, and $D(n)$, the minimum maxdegree of a broadcast graph.Let $L(n)$ denote the number of consecutive leading 1's in the binary representation of integer $n-1$.We show $B(n) = \Theta(L(n) \cdot n)$ and $D(n) = \Theta(\lg \lg n + L(n))$, and for every n we give a construction simultaneously within a constant factor of both lower bounds.For all n we also construct graphs with $O(n)$ edges and $O(\lg \lg n)$ maxdegree requiring at most $c \lg n + 1$ time units to broadcast.Our broadcast protocols may be implemented with local control and $O(\lg \lg n)$ bits overhead per message.

[86] L. A. Hall and D. B. Shmoys. Approximation schemes for constrained scheduling problems. 1989. submitted for publication.

In this paper, a polynomial approximation scheme is presented for the problem of scheduling jobs on parallel identical machines subject to release time in order to minimize the total lateness with respect to specified deadlines. If precedence constraints are added, then an algorithm is given that delivers a solution within a factor of two of optimal. In the special case wherethere is only one machine, simpler superior algorithms are obtained. The two-machine flow shop with release dates is also considered, and a polynomial approximation scheme is given. All of the approximation schemes are based on the notion of an outline, which is a restriction on the set of feasible schedules that still contains a near optimal schedule, and yet is restrictive enough

so that it is possible to use this information to compute such a schedule.

[87] L. A. Hall and D. B. Shmoys. Jackson's rule: making a good heuristic better. *Mathematics of OR*, 1989. To appear.

> We consider the scheduling problem in which jobs with release times and delivery times are to be scheduled on one machine.We present a 1/3-approximation algorithm for the problem with precedence constraints among the jobs, and two ϵ-approximation algorithms for the problem without precedence constraints.Finally, we prove a strong negative result concerning a restricted version of the problem with precedence constraints that indicates that precedence constraints make the problem considerably more difficult to solve.At the core of each of the algorithms presented is Jackson's Rule–a simple but seemingly robust heuristic for the problem.

[88] M. D. Hansen. Approximation algorithms for geometric embeddings in the plane with applications to parallel processing problems. 1989. Submitted to FOCS '89.

> The paper presents fast approximation algorithms for embedding d-dimensional grids into points in the plane within a factor of $O(\log N)$ times optimal cost for $d > 2$ and $O(\log^2 N)$ for $d = 2$. It also shows that any embedding of a hypercube, butterfly, or shuffle-exchange graph must be within an $O(\log N)$ factor of optimal cost. For certain restricted point sets, the paper presents polynomial time algorithms which can embed arbitrary weighted graphs with cost within an $O(\log^2 N)$ factor of optimal.
>
> These results are applied to give $O(\log^2 N)$-times optimal solutions to parallel processor performance optimization problems in the following areas: communication load balancing, dynamic allocation of jobs to processors, reconfiguring around faults, and simulating other architectures.

[89] J. Hastad, T. Leighton, and M. Newman. Fast computation using faulty hypercubes. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, ACM SIGACT, ACM, May 1989. To appear.

[90] M. Held and R. M. Karp. The traveling salesman problems and minimum spanning trees. *Oper. Res.*, 18:1138–1162, 1970.

[91] D. Helmbold, R. Sloan, and M. Warmuth. Learning nested differences of intersections closed concept classes. 1989. Submitted to COLT '89.

> This paper introduces a new framework for constructing learning algorithms. Our methods involve a master algorithm which uses learning algorithms for intersection closed concept classes as subroutines. For example, we give master algorithm capable of learning any concept class whose members can be expressed as nested differences (e.g. $c_1 - (c_2 - (c_3 - (c_4 - c_5))))$ of concepts from an intersection closed class.
>
> We show that our algorithms are optimal or nearly optimal with respect to several different criteria. These criteria include: the number of examples

needed to produce a good hypothesis with high confidence, the worst case total number of mistakes made, and the expected number of mistakes made in the first t trials.

[92] R. Impagliazzo, L. Levin, and M. Luby. Pseudo-random number generation from any one-way function. In 21^{th} *Symp. Theory of Computing*, pages 12–24, ACM, 1989.

Let an easily computable function f be one-way, i.e. for most x one cannot recover from $f(x)$ either (1) x by a polynomial time algorithm, or (2) an $x' \in f^{-1}(f(x))$ by a polynomial size circuit. In case (1), to exclude useless $f(x) = 0$, the difference between Shannon entropies of inputs and outputs of f is restricted to $O(1)$. Based on [Goldreich, Levin 89], we show that the existence of one-way functions in the sense (1) or (2) is necessary and sufficient for the existence of pseudo-random generators secure against feasible algorithms or circuits, respectively.

[93] R. Impagliazzo and M. Yung. Direct minimum knowledge computations. In *Advances in Cryptology*, 1987.

[94] A. T. Ishii. *A Digital Model for Level-Clocked Circuitry.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1988. Supervised by C.E. Leiserson.

This thesis presents the formal background for a mathematical model for level-clocked circuitry, in which latches are controlled by the levels (high or low) of clock signals rather than transitions (edges) of the clocks.Such level-clocked circuits are frequently used in MOS VLSI design.Our model maps continuous data-domains, such as voltage, into discrete, or *digital*, data domains, while retaining a continuous notion of time.A level-clocked circuit is represented as a graph $G = (V, E)$, where V consists of digital components—latches and functional elements—and E represents inter-component connections.

The majority of this thesis concentrates on developing lemmas and theorems that can serve as a set of "axioms" when analyzing algorithms based on the model.Key axioms include the fact that circuits in our model generate only well defined digital signals, and the fact that components in our model support and accurately handle the "undefined" values that electrical signals must take on when they make a transition between valid logic levels.In order to facilitate proofs for circuit properties, the class of *computational predicates* is defined.A circuit property can be proved by simply casting the property as a computational predicate.

[95] L. Jategaonkar and J. C. Mitchell. ML with extended pattern matching and subtypes (preliminary version). In *Symp. LISP and Functional Programming*, pages 198–211, ACM, 1988.

A representative fragment of the programming language ML extended with a more general form of record pattern matching and user-declared subtypes.These enhancements support programs in a restricted but relatively

natural object-oriented style.ML-style typing rules and an efficient type inference algorithm are presented.The algorithm is sound with respect to the typing rules, and it infers a most general typing for every typable expression.

[96] L. A. Jategaonkar. *ML with extended pattern matching and Subtypes.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by A. Meyer. To appear in September, 1989.

A representative fragment of the programming language ML extended with a more general form of record pattern matching and user-declared subtypes. These enhancements support programs in a restricted but relatively natural object-oriented style. ML-style typing rules and an efficient type inference algorithm are presented. The algorithm is sound with respect to the typing rules, and it infers a most general typing for every typable expression.

[97] G. Kahn. The semantics of a simple language for parallel programming. In J. L. Rosenfeld, editor, *Information Processing 74*, pages 471–475, North-Holland, 1974.

[98] R. Karp, E. Upfal, and A. Widgerson. Constructing a perfect matching is in random NC. *Combinatorica*, 6(1):35–48, 1986.

[99] J. Kilian. *Randomness in Algorithms and Protocols.* Ph.D. thesis, MIT Dept. of Mathematics, 1989. Supervised by Shafi Goldwasser.

We develop techniques for using randomness in algorithms and the design of secure protocols.First, we exhibit a probabilistic algorithm for generating large certified primes.Next, we give a round efficient reduction from secure circuit evaluation to oblivious transfer.Finally, we study the properties of a multi-prover generalization of interactive proof systems.

[100] J. Kilian. Efficient zero-knowledge proof systems with bounded interaction. Submitted to FOCS '89.

Exhibited is a simple technique for proving NP assertions in zero-knowledge with bounded interaction.The protocol obtained is more comunication efficient than the best known solutions with unbounded interaction.

[101] J. Kilian, S. Kipnis, and C. E. Leiserson. The organization of permutation architectures with bussed interconnections. *IEEE Trans. Computers*, 1989. To appear. Also appeared as technical memo MIT/LCS/TM-379 and VLSI memo 89-500. Earlier version appeared in 28^{th} IEEE Annual Symposium on Foundations of Computer Science (1987), 305–315.

Investigation of bussed interconnection schemes for realizing permutations among VLSI chips.Establishes a correspondence between uniform permutation architectures and difference covers.

[102] J. Kilian and N. Nisan. Space bounded cryptography. Submitted to FOCS '89.

Cryptographic protocols are considered in the scenario where the parties have space limitations.Some cryptographic protocols are presented whose correctness does not depend on any intractibility assumptions.

[103] S. Kipnis. *Three Methods for Range Queries in Computational Geometry.* Technical Memo TM-388, MIT Lab. for Computer Science, March 1989.

Survey of a variety of recent results addressing the problem of range queries in computational geometry.Identifies and focuses on three methods for range queries in computational geometry: random sampling, search-tree tables, and space-partition trees.

[104] P. Klein and C. Stein. *A Parallel Algorithm for Eliminating Cycles in Undirected Graphs.* Center for Research in Computing Technology Technical Report TR-01-89, Harvard University, March 1989. submitted to Inform. Processing Letters.

We give anparallel algorithm for finding a maximal set of edge-disjoint cycles in an undirected graph which runs in $O(\log n)$ time using $m$processors on a CRCW PRAM.The algorithm can be generalized to handle a weighted version of the problem.

[105] R. Koch. Increasing the size of a network by a constant factor can increase performance by more than a constant factor. In *29th FOCS*, pages 221–230, IEEE, Oct. 1988.

[106] R. Koch. *An Analysis of the Performance of Interconnection Networks for Multiprocessor Systems.* Ph.D. thesis, MIT Dept. of Mathematics, 1989. Supervised by F.T. Leighton.

The bandwidth of the butterfly network is analyzed. In a dilated butterfly network nodes are connected by parallel edges instead of just one edge as in the usual butterfly network. He proves a previous conjecture that that the expected bandwidth of an N input dilated butterfly network is $\Theta(N(\log N)^{-\frac{1}{q}}$, where q is the number of parallel edges. He also explores some implications of his results for design tradeoffs. He also develops interesting techniques for finding asymptotics of nonlinear systems of recurrences

[107] R. Koch, T. Leighton, B. Maggs, S. Rao, and A. Rosenberg. Work-preserving emulations of fixed-connection networks. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, May 1989. To appear.

This paper examines the problem of emulating T_G steps of an N_G-node guest network on an N_H-node host network.An emulation is called *work-preserving* if the time required by the host, T_H, is $O(T_G N_G/N_H)$ because then both the guest and host networks perform the same total work, $\Theta(T_G N_G)$, to within a constant factor.An emulation is *real-time* if $T_H = O(T_G)$, because then the host emulates the guest with constant delay.Although many isolated emulation results have been proved for specific networks in the past, and measures such as dilation and congestion were known to be important, the

field has lacked a model within which general results and meaningful lower bounds can be proved.This paper attempts to provide such a model.Some of the more interesting and diverse conseqeuences of this work include

1. a proof that a linear array can emulate a (much larger) butterfly in a work-preserving fashion, but that a butterfly cannot emulate an expander (of any size) in a work-preserving fashion,

2. a proof that a mesh can be emulated in real time in a work-preserving fashion on a butterfly, even though any $O(1)$-to-1 embedding of a mesh in a butterfly has dilation $\Omega(\log N)$,

3.a proof that an $N \log N$-node butterfly can be emulated in a work-preserving fashion on an N-node shuffle-exchange graph, and vice-versa,

4. simple $O(N^2/\log^2 N)$-area and $O(N^{3/2}/\log^{3/2} N)$-volume layouts for the N-node shuffle-exchange graph, and 5) an algorithm for sorting N-numbers in $O(\log N)$ steps with high probability on an N-node shuffle-exchange graph with constant size queues.

[108] J. Körner and K. Marton. New bounds for perfect hashing via information theory. *European J. Combinatorics*, 9:523–530, 1986.

[109] D. Kravets. *Finding Farthest Neighbors in a Convex Polygon and Related Problems.* Technical Report TR-437, MIT Lab. for Computer Science, Jan. 1989.

Aggarwal et al. [A. Aggarwal, M.M. Klawe, S. Moran, P. Shor, R. Wilber, "Geometric Applications of a Matrix-Searching Algorithm," Algorithmica, Vol. 2, 1987, pp. 195-208] showed how to compute in $O(n)$ time *one farthest vertex* for every vertex of a convex n-gon. This thesis extends the results of Aggarwal et. al. by developing the following algorithms:

An optimal algorithm to find *all farthest vertices* for every vertex of a convex polygon.

An $O(kn \log k)$ time algorithm to find k *farthest vertices* for every vertex of a convex n-gon.

An $O(n^2)$ algorithm to sort the distances of all the vertices of a convex n-gon with respect to each vertex of the convex n-gon.

A worst-case optimal algorithm to sort a set of numbers given lower bounds on the ranks.

[110] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. Sequencing and scheduling: algorithms and complexity. In S. C. Graves, A. H. G. Rinnooy Kan, and P. Zipkin, editors, *The Handbooks of Operations Research and Management Science, Volume IV: Production Planning and Inventory*, North-Holland, 1989. To appear.

Scheduling theory is a field that has prospered since the mid 50's, and there is a wealth of literature in this area. This paper provides in-depth bibliographic coverage of results in this field, and each section is introduced with the complete explanation of one result that is a highlight for that particular area. The main sections are as follows: Preliminaries (Sequencing and scheduling problems, algorithms and complexity, a class of deterministic machine

scheduling problems), The single machine (Minmax criteria, Total weighted completion time, Weighted number of late jobs, Total tardiness and beyond), Parallel machines (Minsum criteria, Minmax criteria without preemption, Minmax criteria with preemption, Precedence constraints), Multi-operation models (Open shops, Flow shops, Job shops), More sequencing and scheduling (Resource-constrained project scheduling, Stochastic machine scheduling).

[111] T. Leighton, M. Newman, A. G. Ranade, and E. Schwabe. Dynamic tree embeddings in butterflies and hypercubes. In 1^{st} *Symp. on Parallel Algorithms and Architectures*, ACM, 1989.

We present simple randomized algorithms for dynamically embedding binary trees in either a butterfly or a hypercube network of processors. These algorithms are *dynamic* in the sense that the tree to be embedded may start as one node and grow by dynamically spawning children, where the nodes are incrementally embedded as they are spawned. Our embedding algorithms for the hypercube and butterfly simultaneously optimize load and dilation up to constant factors (with high probability) for trees which are a logarithmic factor larger than the host network. In addition, we present an improved algorithm for embedding in the hypercube which simultaneously optimizes load and dilation up to constant factors (with high probability) for arbitrary binary trees, while also keeping congestion low. We also prove a $\Omega(\sqrt{\log N})$ lower bound on dilation for determinstic embedding algorithms which achieve optimal load, implying that any embedding algorithm which simultaneously optimizes load and dilation must be randomized.

[112] T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In 29^{th} *Symp. on the Foundations of Computer Science*, pages 422–431, IEEE, 1988.

[113] C. E. Leiserson and J. B. Saxe. *Retiming Synchronous Circuitry.* Technical Memo TM-372, MIT Lab. for Computer Science, Oct. 1988.

[114] L. Levin. Homogeneous measures and polynomial time invariants. In 29^{th} *Symp. Found. Computer Sci.*, pages 36–41, IEEE, 1988.

The usual probability distributions are concentrated on strings which do not differ noticeably in any fundamental characteristic, except their informational size (Kolmogorov complexity). This property distinguishes a class of *homogeneous* probability measures suggesting various applications. In particular, it explains why the average case NP-completeness results are so measure independent and offers their generalization to this wider and more invariant class of measures. It also demonstrates a sharp difference between pseudo-random strings and the objects known before.

[115] L. Levin and R. Venkatesan. Random instances of a graph coloring problem are hard. In 20^{th} *Symp. Theory of Computing*, pages 217–222, ACM, 1988.

NP-complete problems should be hard on *some* (may be extremely rare) instances. On generic instances many such problems (especially related to random graphs) have been proven easy. Modifying the NP-completeness theorem, we show the intractability of *random* instances of a graph coloring problem.

[116] N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, Fourier transform and learnability. 1989. Submitted for publication.

This paper presents a study of boolean functions in AC^0 using the harmonic analysis of the cube. The main result is that an AC^0 boolean function has almost all of its "power spectrum" on the low-order coefficients. An important ingredient of the proof is Hastad's switching lemma.

This result implies several new properties of functions in AC^0: Functions in AC^0 have low "sensitivity"; they may be approximated well by a real polynomial of low degree; they cannot be strong pseudorandom function generators and their correlation with any polylog-wise independent probability distribution is small.

Perhaps the most interesting application is an $O(n^{polylog(n)})$ time algorithm for learning functions in AC^0.The algorithm observes the behavior of an AC^0 function on $O(n^{polylog(n)})$ randomly chosen inputs, and derives a good approximation for the Fourier transform of the function. This allows it to predict with high probability, the value of the function on other randomly chosen inputs.

[117] N. Linial, Y. Mansour, and R. L. Rivest. Results on learnability and the Vapnik-Chervonenkis dimension. In *Proceedings of the Twenth-Ninth Annual Symposium on Foundations of Computer Science*, pages 120–129, Oct. 1988.

[118] N. Linial and N. Nisan. Approximate inclusion-exclusion. 1989. Submitted for publication.

The Inclusion-Exclusion formula expresses the size of a union of a family of sets in terms of the sizes of intersections of all subfamilies. This paper considers approximating the size of the union when either intersection sizes are known for only some of the subfamilies or when these quantities are given to within some error or both.

In particular, we consider the case when all k-wise intersections are given for every $k \leq K$. It turns out that the answer changes in a significant way around $K = \sqrt{n}$: if $K << \sqrt{n}$ then any approximation may err by a factor of $\Theta(n/K^2)$, while if $K >> \sqrt{n}$ it is shown how to approximate the size of the union to within a multiplicative factor of $1 \pm e^{-\Omega(K/\sqrt{n})}$.

When the sizes of all intersections are only given approximately good bounds are derived on how well the size of the union may be approximated. Several applications for boolean function are mentioned in conclusion.

[119] R. Lipton and R. E. Tarjan. A separator theorem for planar graphs. In *Proc. A Conference on Theoretical Computer Science*, University of Waterloo, Aug. 1977.

[120] N. Littlestone. Learning when irrelevant attributes abound: a new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1988.

[121] N. A. Lynch and E. W. Stark. *A Proof of the Khan Principle for Input/Output Automata.* Technical Memo TM-349, MIT Lab. for Computer Science, Jan. 1988.

[122] Y. Mansour and B. Schieber. The intractability of bounded protocols for non-fifo channels. In *8th Annual ACM Symposium on Principles of Distributed Computing*, 1989. To appear.

> We discuss the efficiency of data link protocols for non-FIFO physical channels. We consider three resources: the amount of space required by the protocol, the number of headers, and the number of packets that have to be sent. We prove three lower bounds. First, we show that the space required by any protocol for delivering n messages using less than n headers can not be bounded by any function of n. Second, we prove that the number of packets that have to be sent by any data link protocol using a fixed number of headers in order to deliver a message is linear in the number of packets that are delayed on the channel at the time the message is sent. Finally, we introduce the notion of a probabilistic physical channel, in which a packet is lost with probability q. We prove an exponential lower bound, with overwhelming probability, on the number of packets that have to be sent by any data link protocol using a fixed number of headers, when it is implemented over a probabilistic physical channel.

[123] Y. Mansour, B. Schieber, and P. Tiwari. The complexity of approximating the square root. In 30^{th} *Annual Symposium on Foundations of Computer Science*, 1989. Submitted.

[124] Y. Mansour and B. S. P. Tiwari. Lower bounds for computations with the floor operation. In *Proceeding of ICALP 1989*, 1989. To appear.

> We prove an $\Omega(\sqrt{\log n})$ lower bound on the depth of any decision tree with operations $\{+, -, *, /, \lfloor\cdot, <\}$, that decides whether an integer is a perfect square, for any n-bit integer. We then extend the arguments to obtain the same lower bound on the time complexity of any RAM program with operations $\{+, -, *, /, \lfloor\cdot, <\}$ that solves the problem. Our proof technique can be used to derive lower bounds for many other problems.

[125] P. Mathews. Generating a random linear extension of a partial order. 1989. unpublished.

[126] A. R. Meyer. *Semantical Paradigms: Notes for an Invited Lecture, with Two appendices by Stavros Cosmodakis.* Technical Report MIT/LCS/TM353, MIT Lab. for Computer Science, July 1988.

[127] A. R. Meyer. Semantical paradigms: notes for an invited lecture, with two appendices by Stavros Cosmodakis. In 3rd *Symp. Logic in Computer Science*, pages 236–253, IEEE, 1988.

Comments on denotational semantics, highlighting "goodness of fit" criteria between semantic domains and symbolic evaluators. Two appendices provide the key parts of a long proof that Scott domains give a *computationally adequate* and *fully abstract* semantics for lambda calculus with simple *recursive* types.

[128] A. R. Meyer and J. G. Riecke. Continuations may be unreasonable. In *Proc. Conf. LISP and Functional Programming*, pages 63–71, ACM, 1988.

We show that two lambda calculus terms can be *observationally congruent* (*i.e.*, agree in all contexts) but their continuation-passing transforms may not be. We also show that two terms may be congruent in all untyped contexts but fail to be congruent in a calculus with call/cc operators.Hence, familiar reasoning about functional terms may be unsound if the terms use continuations explicitly or access them implicitly through new operators.We then examine one method of connecting terms with their continuized form, extending the work of Meyer and Wand.

[129] A. R. Meyer and M. A. Taitslin, editors. *Logic at Botic,'89: Proceedings of a Symposium on Logical Foundations of Computer Science.* Volume 363 of *Lect. Notes in Computer Sci.*, Springer-Verlag, July 1989.

[130] S. Micali and R. Ostrovsky. Simple non-interactive zero knowledge proofs with preprocessing oblivious transfer. Sep. 1988. To appear.

[131] S. Micali and A. Shamir. An improvement of the fiat-shamir identification. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[132] K. Mulmuley, U. Vazarani, and V. Vazarani. Matching is as easy as matrix inversion. In 19^{th} *Symp. of Theory of Computing*, pages 345–354, ACM, 1987.

[133] N. Nisan. Crew prams and decision trees. In *Proc. of the 21th STOC Symposium*, ACM, 1989.

This paper gives a full characterization of the time needed to compute a boolean function on a CREW PRAM with an unlimited number of processors.

The characterization is given in terms of a new complexity measure of boolean functions: the "block sensitivity". This measure is a generalization of the well know "critical sensitivity" measure (see [W], [CDR], [Si]). The block sensitivity is also shown to relate to the boolean decision tree complexity, and the implication is that the decision tree complexity also fully characterizes the CREW PRAM complexity.This solves an open problem of [W].

Our results imply that changes in the instruction set of the processors or in the capacity of the shared memory cells do not change by more than a constant factor the time required by a CREW PRAM to compute any boolean function.Moreover, we even show that a seemingly weaker version of a CREW PRAM, the CROW PRAM ([DR]), can compute functions as quickly as a general CREW PRAM.This solves an open problem of [DR].

Finally, our results have implications regarding the power of randomization in the boolean decision tree model.We show that in this model, randomization may only achieve a polynomial speedup over deterministic computation. This was known for Las-Vegas randomized computation; we prove it also for 1-sided error computation (a quadratic bound) and 2-sided error (a cubic bound).

[134] J. B. Orlin. *Genuinely Polynomial Simplex and Non-Simplex Algorithms for the Minimum Cost Flow Problem.* Technical Report No. 1615-84, Sloan School of Management, MIT, 1984.

[135] J. B. Orlin. A faster strongly polynomial minimum cost flow algorithm. In *Proc. of the 20th STOC Symposium*, pages 377–387, ACM, 1988.

[136] R. Ostrovsky. Noninteractive zero-knowledge proofs with pre-processing oblivious transfer. 1988. unpublished.

[137] J. Park. *Notes on Searching in Multidimensional Monotone Arrays.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, January 1989. Supervised by C. E. Leiserson.

[138] N. Perugini. *Neural Network Learning: Effects of Network and Training Set Size.* Master's thesis, MIT Department of Electrical Engineering and Computer Science, June 1989.

[139] C. Phillips. Parallel graph contraction. In 1^{st} *Symposium on Parallel Algorithms and Architectures*, ACM, 1989. To appear.

This paper shows how n-node, e-edge graphs can be *contracted* in a manner similar to the parallel tree contraction algorithm due to Miller and Reif.We give an $O((n+e)/\lg n)$-processor deterministic algorithm that contracts a graph in $O(\lg^2 n)$ time in the EREW PRAM model.We also give an $O(n/\lg n)$-processor randomized algorithm that with high probability can contract a bounded-degree graph in $O(\lg n + \lg^2 \gamma)$ time, where γ is the maximum genus of any connected component of the graph. (The algorithm can be made to run in deterministic $O(\lg n \lg^* n + \lg^2 \gamma)$ time using known techniques.)This algorithm does not require *a priori* knowledge of the genus of the graph to be contracted.The contraction algorithm for bounded-degree graphs can be used directly to solve the problem of region labeling in vision systems, *i.e.*, determining the connected components of bounded-degree planar graphs in $O(\lg n)$ time, thus improving the best previous bound of $O(\lg^2 n)$.

[140] C. Phillips and S. A. Zenios. Experiences with large scale network optimization on the connection machine. In *Impact of Recent Computer Advances on Operations REsearch*, Elsevier Science Publishing Co., New York, NY, 1989.

Network optimization problems appear in several areas of application from operations research, transportation, engineering design, financial planning and others. Such problems are characterized, quite often, by their very large size. Massively parallel computers like the Connection Machine (CM) appear to be well suited for both sparse and dense implementations of dual relaxation algorithms for network optimization. In this report we summarize recent experiences with the solution of large scale network optimization problems using the CM. We discuss key features of the implementation of parallel algorithms for *assignment* and *nonlinear network optimization* problems and present results with numerical experiments.

[141] G. D. Plotkin. LCF considered as a programming language. *Theoretical Computer Sci.*, 5:223–257, 1977.

[142] S. Plotkin and E. Tardos. Improved dual network simplex. 1989. submitted for publication.

This paper improves the number of pivot steps required for the dual network simplex algorithm.A simplified version of Orlin's[135] strongly polynomial minimum-cost flow algorithm is developed, and it is shown how to convert it to a dual network simplex. The pivoting strategy leads to an $O(m^2 \log n)$ bound on the number of pivots, which is better by a factor of m compared to the previously best pivoting strategy due to Orlin [134].Here n and m denotes the number of nodes and arcs in the input network.

[143] A. M. Rabinovich. *Nets and Processes.* Ph.D. thesis, Tel Aviv University, 1988. Supervised by B. A. Trakhtenbrot.

[144] J. G. Riecke. *Should a Function Continue?* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, Jan. 1989. Supervised by A.R. Meyer.

An extended version of [128].

[145] R. L. Rivest and R. E. Schapire. Inference of finite automata using homing sequences. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, Seattle, Washington, May 1989. To appear.

[146] R. L. Rivest and R. Sloan. Learning complicated concepts reliably and usefully. In *Proceedings AAAI-88*, pages 635–640, American Association for Artificial Intelligence, Aug. 1988.

This paper shows how to learn from examples (Valiant style) any concept representable as a boolean function, with the help of a teacher who breaks the

concept into subconcepts and teaches one subconcept per lesson.Each subconcept corresponds to a gate in a boolean circuit computing the unknown concept.The learner learns each subconcept from examples which have been randomly drawn according to an arbitrary probability distribution, and labeled as positive or negative instances of the subconcept by the teacher.The learning procedure runs in time polynomial in the size of the circuit.

[147] R. L. Rivest and R. Sloan. A new model for inductive inference. In M. Vardi, editor, *Proceedings of the 2nd Annual Theoretical Aspects of Reasoning about Knowledge Conference*, pages 13–27, Morgan Kaufmann, March 1988. Submitted to *Information and Computation*.

A new model for inductive inference is introduced. This model combines a Bayesian approach for representing the current state of knowledge with a simple model for the computational cost of making predictions from theories. This paper investigates the optimization problem: how should a scientist divide his time between doing experiments and deducing predictions for promising theories.

[148] J. Rompel. *A Better Performance Guarantee for Approximate Graph Coloring.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by Tom Leighton.

Approximate graph coloring takes as input a graph and returns a legal coloring which is not necessarily optimal. This thesis contains algorithms improvingthe performance guarantee, or worst-case ratio between the number of colors used and the minimum number of colors possible, to $O(n(\log\log n)^3/(\log n)^3)$, an $O(\log n/\log\log n)$ factor better than the previous best known result.

[149] R. E. Schapire. The strength of weak learnability. 1989. Submitted.

The problem is considered of improving the accuracy of a hypothesis output by a learning algorithm in the distribution-free ("pac") learning model introduced by Valiant. A concept class is *learnable* (or *strongly learnable*) if, given access to a source of examples from the unknown concept, the learner with high probability is able to output a hypothesis that is correct on all but an arbitrarily small fraction of the instances. The concept class is *weakly learnable* if the learner can produce a hypothesis that performs only slightly better than random guessing. In this paper, it is shown that these two notions of learnability are equivalent.

An explicit method is described for directly converting a weak learning algorithm into one that achieves arbitrarily high accuracy. This construction may have practical applications as a tool for efficiently converting a mediocre learning algorithm into one that performs extremely well. The result answers the *hypothesis boosting problem*, and adds both the weak learning and the

group learning models to the class of models equivalent to the basic pac-learning model. (A group learning algorithm need only output a hypothesis capable of classifying large groups of instances, all of which are either positive or negative.)

An interesting consequence of the construction presented is a proof that any strong learning algorithm that outputs hypotheses whose length (and thus whose time to evaluate) depends on the allowed error ϵ can be modified to output hypotheses whose length is only polynomial in $\log(1/\epsilon)$.

[150] A. T. Sherman. *VLSI Placement and Routing: The PI Project.* Springer-Verlag, New York, New York, 1989.

[151] D. B. Shmoys and E. Tardos. Computational complexity. In R. Graham, M. Grötschel, and L. Lovász, editors, *The Handbook of Combinatorics*, North Holland, Amsterdam, 1989. To appear.

This paper provides a general survey of complexity theory focusing on aspects that are related to combinatorial problems. The main sections are: Introduction, Complexity of Computational Problems (Computational problems, Models of computation, Complexity classes, Other theoretical models of efficiency), Shades of Intractability (Undecidability, Evidence of intractability: $\mathcal{NP}$-completeness, The polynomail-time hierarchy, Evidence of intractability: $\#\mathcal{P}$-completeness, Evidence of intractability: $\mathcal{PSPACE}$-completeness, Proof of intractability, Extensions of $\mathcal{NP}$: short proofs via randomization), Living with Intractability (The complexity of approximate solutions, Probabilistic analysis of algorithms, Cryptography),Inside $\mathcal{P}$ (Logarithmic space, The hardest problems in $\mathcal{P}$, Parallel computation), Attacks on the $\mathcal{P}$ versus $\mathcal{NP}$ Problem (Relativized complexity classes, Relating circuit complexity to Turing machine complexity, Constant-depth circuits, Monotone circuit complexity, Machine-based complexity classes).

[152] D. B. Shmoys and D. Williamson. Analyzing the Held-Karp TSP bound: a monotonicity property with application. 1989. Submitted to *Information Processing Letters.*

We consider the Held-Karp lower bound procedure for the Traveling Salesman Problem, and prove that the optimal Held-Karp bound is a monotonic property, in that the bound for a subset of the input is no more than for the entire input. A corollary of this is that the bound is at least 2/3 of the optimum TSP value for any input.

[153] R. Sloan. Types of noise in data for concept learning. In *First Workshop on Computational Learning Theory*, pages 91–96, Morgan Kaufmann, 1988.

This paper examines the effects of different sorts of noise on Valiant style learning. We show that basicallyonly two different sorts of noise need be considered:classification noise and attribute noise.

In particular, we show that the general upper bound on the tolerable noise rate for the case of malicious noise holds for any algorithm that works by minimizing disagreements in the case of random attribute noise with only slight weakening (from $\epsilon/(1+\epsilon)$ to ϵ).

In the other direction, we show that the information theoretic bound of one half which has been obtained for *random* classification noise is (information theoretically) achievable for any sort of classification noise. We also show that a particular efficient algorithm for learning kDNF which tolerates that amount of random classification noise can be modified to tolerate the same amount of arbitrary classification noise.

[154] R. Sloan. *All Zero-Knowledge Proofs are Proofs of Language Membership.* Technical Memo TM-385, MIT Lab. for Computer Science, Feb. 1989.

Several similar but distinct definitions of zero-knowledge proofshave beenproposed. They fall mainly in two categories: zero-knowledge proofs of language membership, and zero-knowledge proofs of possession of knowledge.

In this paper, we show that proofs of possession of knowledge are simply special cases of proofs of language membership in a very strong sense: All proofs of possession of knowledge are proofs of language membership, the natural subset of proofs of language membership are proofs of knowledge, and for every probabilistic polynomial time predicate $R(x,y)$ there is a zero-knowledge proof of possession of knowledgeof some y_0 such that $R(x,y_0)$ that is also a proof of membership of x in some MA language. Some of our results require the assumption of the existence of a secure probabilistic encryption scheme.

Thus, proofs of possession of knowledge have no independent interest from a complexity theory point of view. Nevertheless, they can be very useful conceptual tools for protocol design. Weexamine some of the applications of the various sorts of zero-knowledge proofs.

[155] R. Sloan. *Computational Learning Theory:New Models and Algorithms.* Ph.D. thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by R. L. Rivest.

In the past several years, there has been a surge of interest in computational learning theory—the formal (as opposed to empirical) study of learning algorithms. One major cause for this interest was the model of probably approximately correct learning, or pac learning, introduced by Valiant in 1988.

This thesis begins by presenting a new learning algorithm for a particular problem within that model:learning submodules of the free $\mathbf{Z}$-module $\mathbf{Z}^k$. We prove that this algorithm achieves probable approximate correctness, and indeed, that it is within a log log factor of optimal in a related, but more stringent model of learning, on-line mistake bounded learning.

We then proceed to examine the influence of noisy data on pac learning algorithms in general.Previously it has been shown that it is possible to tolerate large amounts of random classification noise, but only a very small amount of a very malicious sort of noise.We show that similar results can be obtained for models of noise in between the previously studied models:a large amount of malicious classification noise can be tolerated, but only a small amount of random attribute noise.

Next, we overcome a major limitation of the pac learning model by introducing a variant model with a more powerful teacher. Weshow how to learnany concept representable as a boolean function, with the help of a teacher who breaks the concept into subconcepts and teaches one subconcept per lesson. The learner outputs not the unknown boolean circuit, but rather a program which, on a given input, either produces the same answer as the unknown boolean circuit would, or else says "I don't know." Thus, unlike many learning programs, the output of this learning procedure is *reliable*. Furthermore, with high probability the output program is nearly always *useful* in that it says "I don't know" on only a small fraction of the domain.

Finally, we look at a new model for an older learning problem, inductive inference.This new model combines certain features of the traditional model of Gold for inductive inferencetogether with the concern of the Valiant model for efficient computation and also with notions of Bayesianism. The result is a model that captures certain qualitative aspects of the classic scientific method.

[156] W. Smith. *Ph.D. Thesis.* Ph.D. thesis, Department of Applied Mathematics, Princeton University, 1988.

[157] C. Stein. Efficient algorithms for bipartite network flow. 1987. Princeton University, unpublished manuscript.

[158] W. W. Tait. A realizability interpretation of the theory of species. In R. Parikh, editor, *Logic Colloqium, '73*, pages 22–37, Volume 453 of *Lect. Notes in Math.*, Springer-Verlag, 1975.

[159] E. Upfal. An $O(\log N)$ deterministic packet routing scheme. In *Proceedings of the 21st Annual ACM Symposium on Theory of Computing*, May 1989. To appear.

[160] L. G. Valiant. A theory of the learnable. *Comm. ACM*, 27(11):1134–1142, Nov. 1984.

[161] S. Wu. An efficient algorithm for finding two edge-disjoint paths in a graph. 1988. Submitted for publication.

In 1980, P.D. Seymour exhibited a necessary and sufficient condition for a pair $(G, \{s_1, t_1, s_2, t_2\})$, where G is a graph and the s_i, t_i are vertices of G, to have paths from s_1 to t_1 and from s_2 to t_2 which are edge-disjoint.From Seymour's proof, we derive an $O(n^3)$ algorithm for finding the two edge-disjoint paths, n being the number of vertices in G.Applying some additional

algorithmic techniques and providing a better analysis of the running time then yields an $O(n^2)$ algorithm for the 2 disjoint paths problem.This is better than the $O(m^2n)$ bound that is obtainable from Shiloach's $O(mn)$ algorithm by converting the input graph to its edge graph.

[162] A. C. Yao. Theory and applications of trapdoor functions. *IEEE Trans. Computers*, 80–91, 1982.

5 Publications '88–'89

[1] Y. Afek, B. Awerbuch, and H. Moriel. Overhead of resetting a communication protocol is independent of the size of the network. May 1989. Unpublished manuscript.

[2] A. Agarwal, G. E. Blelloch, R. L. Krawitz, and C. A. Phillips. Four vector-matrix primitives. In 1^{st} *Symposium on Parallel Algorithms and Architectures*, ACM, 1989.

[3] A. Aggarwal and D. Kravets. A linear time algorithm for finding all farthest neighbors in a convex polygon. *Info. Proc. Lett.*, 31:16–20, 1989.

[4] A. Aggarwal, T. Leighton, and K. Palem. Area-time optimal circuits for iterated addition in vlsi. Nov. 1988. Submitted to IEEE Trans. on Computers.

[5] A. Aggarwal and J. Park. Notes on searching in multidimensional monotone arrays. In 29^{th} *Symp. Found. Computer Sci.*, pages 597–512, IEEE, 1988.

[6] A. Aggarwal and J. Park. Sequential searching in multidimensional monotone arrays. 1989. Submitted for publication.

[7] A. Appel and T. Jim. Continuation-passing, closure-passing style. In 16^{th} *Symposium on Principles of Programming Languages*, ACM, 1989.

[8] B. Awerbuch. On the effects of feedback in dynamic network protocols. In 29^{th} *Annual Symposium on Foundations of Computer Science*, pages 231–245, IEEE, Oct. 1988.

[9] B. Awerbuch. Distributed shortest paths algorithms. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, pages 230–240, ACM SIGACT, ACM, May 1989.

[10] B. Awerbuch, A. Bar-Noy, N. Linial, and D. Peleg. Compact distributed data structures for adaptive network routing. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, pages 230–240, ACM SIGACT, ACM, May 1989.

[11] B. Awerbuch, A. Goldberg, M. Luby, and S. Plotkin. Network decomposition and locality in distributed computation. May 1989. Unpublished manuscript.

[12] B. Awerbuch, O. Goldreich, and A. Herzberg. A quantitative approach to dynamic networks. May 1989. Unpublished manuscript.

[13] B. Awerbuch, O. Goldreich, D. Peleg, and R. Vainish. *On the Message Complexity of Broadcast: Basic Lower Bound.* Technical Memo TM-365, MIT Lab. for Computer Science, July 1988. (Accepted for publication at Journal of the ACM.).

[14] B. Awerbuch, Y. Mansour, and N. Shavit. Polynomial end-to-end communication. In 30^{th} *Annual Symposium on Foundations of Computer Science*, IEEE, 1989. Submitted.

[15] B. Awerbuch and M. Sipser. Dynamic networks are as fast as static networks. In 29^{th} *Annual Symposium on Foundations of Computer Science*, pages 206–220, IEEE, Oct. 1988.

[16] L. Babai, N. Nisan, and M. Szegedy. Multiparty protocols and pseudorandom generators for logspace. In *Proc. of the 21th STOC Symposium*, ACM, 1989.

[17] F. Barahona and E. Tardos. Note on Weintraub's minimum cost flow algorithm. *SIAM Journal on Computing*, 1989.

[18] D. Beaver and S. Goldwasser. Multi-party computation with faulty majority. March 1989. unpublished.

[19] M. Bellare and S. Goldwasser. New paradigms for digital signature schemes and message authentication based on non-interactive zero knowledge proofs. March 1989. unpublished.

[20] M. Bellare and S. Micali. How to sign given any trapdoor function. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[21] M. Bellare and S. Micali. Non-interactive oblivious transfer and applications. March 1989. unpublished.

[22] M. Bellare, S. Micali, and R. Ostrovsky. Parallelizing zero knowledge proofs and perfect completeness zero knowledge. Apr. 1989. unpublished.

[23] S. Ben-david, G. M. Benedek, and Y. Mansour. The passive student is really weaker. In *COLT*, 1989. Submitted.

[24] M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway. Everything provable is provable in zero-knowledge. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[25] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Multi-prover interactive proof systems, removing intractibility assumptions. In *Proceedings of the* 20^{th} *STOC*, ACM, 1988.

[26] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Efficient identification schemes using two prover interactive proofs. March 1989. unpublished.

[27] B. Berger. *Data Structures for Removing Randomness.* Technical Report TR-436, MIT Lab. for Computer Science, Dec. 1988.

[28] B. Berger, M. Brady, D. Brown, and T. Leighton. Nearly optimal algorithms and bounds for multilayer channel routing. Feb. 1989. Submitted to JACM.

[29] B. Berger and J. Rompel. A better performance guarantee for approximate graph coloring. *Algorithmica*, 1988.

[30] B. Berger and J. Rompel. Simulating ($\log^c n$)-wise independence in nc. In 30^{th} *Symp. on Found. of Computer Sci.*, IEEE, 1989. To appear. Also appeared as technical report MIT/LCS/TR-435.

[31] B. Berger, J. Rompel, and P. Shor. Efficient nc algorithms for set cover with applications to learning and geometry. In 30^{th} *Symp. on Found. of Computer Sci.*, IEEE, 1989. To appear. Also appeared as technical report MIT/LCS/TR-444.

[32] B. Berger and P. Shor. *Tight bounds for the acyclic subgraph problem.* Technical Report TR-413, MIT Lab. for Computer Science, June 1989. Submitted for publication.

[33] F. Berman, D. Johnson, T. Leighton, P. Shor, and L. Snyder. Generalized planar matching. *J. Algorithms*, 1989. To appear.

[34] D. Bertsimas and M. Grigni. On the space-filling curve heuristic for the euclidean traveling salesman problem. *Operations Research Letters*, 1989. To appear.

[35] S. Bhatt, F. Chung, J. Hong, T. Leighton, and A. Rosenberg. Optimal simulations by butterfly networks. Sep. 1988. Submitted to JACM.

[36] S. Bhatt, F. Chung, T. Leighton, and A. Rosenberg. Efficient embedding of trees in hypercubes. Oct. 1988. Submitted to SIAM J. Computing.

[37] S. Bhatt, F. Chung, T. Leighton, and A. Rosenberg. Universal graphs for bounded-degree trees and planar graphs. *SIAM J Discrete Math*, 1989. To appear.

[38] B. Bloom. Can LCF be topped? In *Proceedings of LICS '88*, 1988. August 1988.

[39] B. Bloom, S. Istrail, and A. R. Meyer. Bisimulation can't be traced: preliminary report. In 15^{th} *Symp. Principles of Programming Languages*, pages 229–239, ACM, 1988. Final version in preparation for journal submission.

[40] B. Bloom and A. R. Meyer. Experimenting with process equivalence. Jan. 1989. 12 page extended abstract, to be submitted.

[41] B. Bloom and A. R. Meyer. A remark on the bisimulation of probabilistic processes. In *Logic at Botic '89, Proceedings*, pages 26–40, Volume 363 of *Lect. Notes in Computer Sci.*, July 1989.

[42] B. Bloom and J. G. Riecke. LCF should be lifted. In *Proc. Conf. AMAST*, pages 133–136, 1989.

[43] A. Blum. *On the Computational Complexity of Training Simple Neural Networks.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by Ron Rivest.

[44] A. Blum. An $\tilde{O}(n^{0.4})$-approximation algorithm for 3-coloring (and improved approximation algorithms for k-coloring). In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, Seattle, Washington, May 1989.

[45] A. Blum and R. Rivest. Training a 3-node neural network is NP-Complete. In *Advances in Neural Information Processing Systems 1*, pages 494–501, Morgan Kaufmann, 1988. Also presented at the 1988 Workshop on Computational Learning Theory.

[46] M. Blum, P. Feldman, and S. Micali. Proving security against chosen cyphertext attack. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[47] R. B. Boppana and M. Sipser. The complexity of finite functions. 1989. To appear in the Handbook of Theoretical Computer Science.

[48] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *J. Comp. and Syst. Sci.*, 37:156–189, 1988.

[49] G. Brassard and C. Crépeau. Sorting out zero-knowledge. In *Advances in Cryptology: Proceedings of Eurocrypt '89, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

[50] G. Brassard, C. Crépeau, and M. Yung. Everything in NP can be argued in perfect zero-knowledge in a constant number of rounds. In 16^{th} *ICALP, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

[51] K. B. Bruce and A. R. Meyer. The semantics of second-order polymorphic lambda calculus. In G. Kahn, D. B. MacQueen, and G. Plotkin, editors, *Semantics of Data Types*, pages 131–144, Volume 173 of *Lect. Notes in Computer Sci.*, Springer-Verlag, 1984. To appear in *Information and Computation*, January, 1990, coauthored with John C. Mitchell.

[52] T. Bui, C. Heigham, C. Jones, and T. Leighton. Improving the performance of the kernighan-lin and simulated annealing graph bisection algorithms. In *DAC*, June 1989. To appear.

[53] B. Chor, M. Merritt, and D. B. Shmoys. Simple constant-time consensus protocols in realistic fault models. *J. ACM*, 1989. To appear. An earlier version of this appeared in the 4^{th} Symposium on Principles of Distributed Computing.

[54] E. Coffman, L. Flatto, and T. Leighton. First-fit allocation of queues: tight probabilistic bounds on wasted space. May 198p. Submitted to 1st ACM/SIAM Symposium on Discrete Algorithms.

[55] E. Coffman and T. Leighton. A provably efficient algorithm for dynamic storage allocation. *JCSS*, 38(1):2–35, Feb. 1989.

[56] T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms.* McGraw-Hill/MIT Press, 1989.

[57] C. Crépeau. Verifiable disclosure of secrets and applications. In *Advances in Cryptology: Proceedings of Eurocrypt '89, Lect. Notes in Computer Sci.*, Springer-Verlag, 1989. To appear.

[58] C. Crépeau and J. Kilian. Achieving oblivious transfer using weakened security assumptions. In 28^{th} *Symp. on Found. of Computer Sci.*, pages 42–52, IEEE, 1988.

[59] C. Crépeau and J. Kilian. Weakening security assumptions and oblivious transfer. In *Advances in Cryptology: Proceedings of Crypto '88, Lect. Notes in Computer Sci.*, Springer-Verlag, 1988. To appear.

[60] A. DeSantis, S. Micali, and G. Persiano. Non-interactive zero-knowledge proof-systems with auxiliary language. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[61] C. Dwork, D. Shmoys, and L. Stockmeyer. Flipping persuasively in constant expected time. *SIAM J. Computing*, 1989. To appear.

[62] P. Elias. Zero error capacity under list decoding. *IEEE Transactions on Information Theory*, 34:1070–1074, 1988.

[63] P. Elias. *Error-Correcting Codes for List Decoding.* Technical Report TM-381, MIT Lab. for Computer Science, Feb. 1989. Submitted for publication.

[64] M. D. Ernst. *Adequate Models for Recursive Program Schemes.* Bachelor's thesis, MIT Dept. of Electrical Engineering and Computer Science, May 1989. Supervised by Albert R. Meyer.

[65] M. D. Ernst. Ml typechecking is not efficient. In *Papers of the MIT ACM Undergraduate Conference*, Apr. 1989.

[66] L. Finkelstein, D. Kleitman, and T. Leighton. Applying the classification theorem for finite simple groups to minimize pin count in uniform permutation architectures. In *Proc. AWOC*, pages 247–256, 1988.

[67] L. Fortnow. *Complexity-Theoretic Aspects of Interactive Proof Systems.* Ph.D. thesis, MIT, 1989.

[68] L. Fortnow and M. Sipser. Are there interactive protocols for co-NP languages? *Info. Proc. Lett.*, 28:249–251, 1988.

[69] L. Fortnow and M. Sipser. Probabilistic computation and linear time. In 21^{st} *Symposium on Theory of Computing*, ACM, 1989. To appear.

[70] M. Foster and R. I. Greenberg. Lower bounds on the area of finite-state machines. *Info. Proc. Lett.*, 30(1):1–7, Jan. 1989.

[71] A. Frank, T. Nishizeki, N. Saito, H. Suzuki, and E. Tardos. Algorithms for routing around a rectangle. 1989. Submitted for publication.

[72] A. Frank and E. Tardos. An application of submodular flows. *Linear Algebra and its Applications*, 1989. To appear.

[73] J. Fried. A VLSI chip set for burst and ATM switching. In *International Communications Conference*, 1989.

[74] J. Fried. *VLSI Processor Design for Communications Networks.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by C.E. Leiserson. Also appears as an MIT VLSI memo.

[75] J. Fried. Yield modeling using the SPIROS redundancy planner. In *1st International Conference on Wafer-Scale Integration*, 1989.

[76] J. Fried, E. Daly, T. Lyszarcz, and M. Cooperman. A yield-enhanced crosspoint switch chip using e-beam restructuring. *IEEE Transactions on Solid-State Circuits*, 2, 1989.

[77] J. Fried, D. Ghosh, and J. Daly. A novel content-addressable memory circuit. *Electronics Letters*, 1989.

[78] J. Fried and P. Kubat. Reliability models for facilities switching. 1989. Submitted to IEEE Transactions on Reliability.

[79] J. Fried and B. Kuszmaul. NAP (no ALU processor): the great communicator. In *Frontiers of Massively Parallel Computation*, 1988. An extended version of this paper has been submitted for publication in the Journal of Parallel and Distributed Computing.

[80] A. V. Goldberg, S. Plotkin, D. B. Shmoys, and E. Tardos. Interior-point methods in parallel computation. 1989. submitted for publication.

[81] A. V. Goldberg, S. Plotkin, and E. Tardos. Combinatorial algorithms for the generalized circulation problem. In *29th FOCS Symposium*, pages 432–443, IEEE, 1988. Submitted for journal publication.

[82] A. V. Goldberg, E. Tardos, and R. E. Tarjan. Network flow algorithms. In *Flows, Paths and VLSI-layout*, Springer Verlag, 1989. To appear.

[83] S. A. Goldman. A space efficient greedy triangulation algorithm. *Info. Proc. Lett.*, 1989. To appear in May. Earlier version available as MIT/LCS/TM–366.

[84] S. A. Goldman and R. L. Rivest. Mistake bounds and efficient halving algorithms. 1989. Submitted.

[85] S. A. Goldman, R. L. Rivest, and R. E. Schapire. Learning binary relations and total orders. 1989. To appear.

[86] O. Goldreich, A. Herzberg, and Y. Mansour. Source to destination communication in the presence of faults. In *8th Annual ACM Symposium on Principles of Distributed Computing*, 1989. To appear.

[87] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *Society for Industrial and Applied Mathematics*, 18:186–208, 1989.

[88] R. I. Greenberg. *Area-Universal Networks.* VLSI Memo 524, Massachusetts Institute of Technology, 1989.

[89] R. I. Greenberg, A. T. Ishii, and A. L. Sangiovanni-Vincentelli. MulCh: A multi-layer channel router using one, two, and three layer partitions. In *IEEE International Conference on Computer-Aided Design (ICCAD-88)*, pages 88–91, IEEE Computer Society Press, 1988.

[90] R. I. Greenberg and C. E. Leiserson. A compact layout for the three-dimensional tree of meshes. *Applied Mathematics Letters*, 1(2):171–176, 1988.

[91] M. Grigni and D. Peleg. *Tight Bounds on Minimum Broadcast Networks.* Technical Memo TM-374, MIT Lab. for Computer Science, Dec. 1988.

[92] L. A. Hall and D. B. Shmoys. Approximation schemes for constrained scheduling problems. 1989. submitted for publication.

[93] L. A. Hall and D. B. Shmoys. Jackson's rule: making a good heuristic better. *Mathematics of OR*, 1989. To appear.

[94] M. D. Hansen. Approximation algorithms for geometric embeddings in the plane with applications to parallel processing problems. 1989. Submitted to FOCS '89.

[95] J. Hastad, T. Leighton, and M. Newman. Fast computation using faulty hypercubes. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, ACM SIGACT, ACM, May 1989. To appear.

[96] D. Helmbold, R. Sloan, and M. Warmuth. Learning nested differences of intersections closed concept classes. 1989. Submitted to COLT '89.

[97] D. S. Hochbaum and D. B. Shmoys. A polynomial approximation scheme for scheduling on uniform processors: using the dual approximation approach. *SIAM J. Computing*, 17:539–551, 1988.

[98] A. T. Ishii. *A Digital Model for Level-Clocked Circuitry.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1988. Supervised by C.E. Leiserson.

[99] L. Jategaonkar and J. C. Mitchell. ML with extended pattern matching and subtypes (preliminary version). In *Symp. LISP and Functional Programming*, pages 198–211, ACM, 1988.

[100] L. A. Jategaonkar. *ML with extended pattern matching and Subtypes.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by A. Meyer. To appear in September, 1989.

[101] J. Kilian. *Randomness in Algorithms and Protocols.* Ph.D. thesis, MIT Dept. of Mathematics, 1989. Supervised by Shafi Goldwasser.

[102] J. Kilian. Efficient zero-knowledge proof systems with bounded interaction. Submitted to FOCS '89.

[103] J. Kilian, S. Kipnis, and C. E. Leiserson. The organization of permutation architectures with bussed interconnections. *IEEE Trans. Computers*, 1989. To appear. Also appeared as technical memo MIT/LCS/TM-379 and VLSI memo 89-500. Earlier version appeared in 28^{th} IEEE Annual Symposium on Foundations of Computer Science (1987), 305–315.

[104] J. Kilian and N. Nisan. Space bounded cryptography. Submitted to FOCS '89.

[105] G. A. Kindervater, J. K. Lenstra, and D. B. Shmoys. The parallel complexity of TSP heuristics. *J. Algorithms*, 1989. To appear.

[106] S. Kipnis. *Three Methods for Range Queries in Computational Geometry.* Technical Memo TM-388, MIT Lab. for Computer Science, March 1989.

[107] P. Klein and C. Stein. *A Parallel Algorithm for Eliminating Cycles in Undirected Graphs.* Center for Research in Computing Technology Technical Report TR-01-89, Harvard University, March 1989. submitted to Inform. Processing Letters.

[108] R. Koch. Increasing the size of a network by a constant factor can increase performance by more than a constant factor. In *29th FOCS*, pages 221–230, IEEE, Oct. 1988.

[109] R. Koch. *An Analysis of the Performance of Interconnection Networks for Multiprocessor Systems.* Ph.D. thesis, MIT Dept. of Mathematics, 1989. Supervised by F.T. Leighton.

[110] R. Koch, T. Leighton, B. Maggs, S. Rao, and A. Rosenberg. Work-preserving emulations of fixed-connection networks. In *Proceedings of the* 21^{st} *Annual ACM Symposium on Theory of Computing, Seattle, Washington*, May 1989. To appear.

[111] D. Kravets. *Finding Farthest Neighbors in a Convex Polygon and Related Problems.* Technical Report TR-437, MIT Lab. for Computer Science, Jan. 1989.

[112] E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. Sequencing and scheduling: algorithms and complexity. In S. C. Graves, A. H. G. Rinnooy Kan, and P. Zipkin, editors, *The Handbooks of Operations Research and Management Science, Volume IV: Production Planning and Inventory*, North-Holland, 1989. To appear.

[113] T. Leighton. A $2d-1$ lower bound for 2-layer knock-knee channel routing. Nov. 1988. Submitted to SIAM J. Discrete Math.

[114] T. Leighton, C. Leiserson, and E. Schwabe. *Theory of Parallel and VLSI Computation.* Research Seminar Series MIT/LCS/RSS6, MIT Lab. for Computer Science, March 1989.

[115] T. Leighton and B. Maggs. Expanders might be practical:fast algorithms for routing around faults in multibutterflies. Submitted.

[116] T. Leighton, B. Maggs, and S. Rao. Universal packet routing algorithms. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 256–271, IEEE, Oct. 1988.

[117] T. Leighton, F. Makedon, and I. Tollis. A $2n-2$ step algorithm for routing in an nxn array with constant-size queues. In *ACM SPAA*, June 1989. To appear.

[118] T. Leighton, M. Newman, A. G. Ranade, and E. Schwabe. Dynamic tree embeddings in butterflies and hypercubes. In 1^{st} *Symp. on Parallel Algorithms and Architectures*, ACM, 1989.

[119] T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms (extended abstract). In 29^{th} *Symp. Found. Computer Sci.*, page 422, IEEE, 1988.

[120] T. Leighton and P. Shor. Tight bounds for minimax grid matching with applications to the average case analysis of algorithms. *Combinatorica*, 1989. To appear.

[121] C. Leiserson. VLSI theory and parallel supercomputing. In *Decennial Caltech Conference on VLSI*, pages 5–16, MIT Press, March 1989.

[122] C. Leiserson and B. Maggs. Communication-efficient parallel algorithms for distributed random-access machines. *Algorithmica*, 3:53–77, 1988. An early version appears as "Communication-efficient parallel graph algorithms," in *1986 International Conference on Parallel Processing*, August 1986, 861–868. (Received Most Original Paper Award at the conference.).

[123] C. Leiserson and J. Saxe. A mixed-integer linear programming problem which is efficiently solvable. *Journal of Algorithms*, 9:114–128, 1988. An early version appearsin *Twenty-First Annual Allerton Conference on Communication, Control, and Computing*, October 1983.

[124] J. K. Lenstra, D. B. Shmoys, and E. Tardos. Scheduling unrelated parallel machines. *Mathematical Programming*, 1989. To appear. An earlier version of this appeared in the 28^{th} Symposium on Foundations of Computer Science.

[125] N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, Fourier transform and learnability. 1989. Submitted for publication.

[126] N. Linial, Y. Mansour, and R. L. Rivest. Results on learnability and the Vapnik-Chervonenkis dimension. In *Proceedings of the Twenth-Ninth Annual Symposium on Foundations of Computer Science*, pages 120–129, Oct. 1988.

[127] N. Linial and N. Nisan. Approximate inclusion-exclusion. 1989. Submitted for publication.

[128] Y. Mansour and B. Schieber. The intractability of bounded protocols for non-fifo channels. In *8th Annual ACM Symposium on Principles of Distributed Computing*, 1989. To appear.

[129] Y. Mansour, B. Schieber, and P. Tiwari. The complexity of approximating the square root. In 30^{th} *Annual Symposium on Foundations of Computer Science*, 1989. Submitted.

[130] Y. Mansour and B. S. P. Tiwari. Lower bounds for computations with the floor operation. In *Proceeding of ICALP 1989*, 1989. To appear.

[131] A. R. Meyer. Semantical paradigms: notes for an invited lecture, with two appendices by Stavros Cosmodakis. In 3rd *Symp. Logic in Computer Science*, pages 236–253, IEEE, 1988.

[132] A. R. Meyer and J. G. Riecke. Continuations may be unreasonable. In *Proc. Conf. LISP and Functional Programming*, pages 63–71, ACM, 1988.

[133] A. R. Meyer and M. A. Taitslin, editors. *Logic at Botic,'89: Proceedings of a Symposium on Logical Foundations of Computer Science.* Volume 363 of *Lect. Notes in Computer Sci.*, Springer-Verlag, July 1989.

[134] S. Micali and R. Ostrovsky. Simple non-interactive zero knowledge proofs with preprocessing oblivious transfer. Sep. 1988. To appear.

[135] S. Micali and C. P. Schnorr. Super-efficient, perfect random number generators. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[136] S. Micali and A. Shamir. An improvement of the fiat-shamir identification. In *Proc. of CRYPTO-88*, Springer-Verlag, 1988.

[137] N. Nisan. Crew prams and decision trees. In *Proc. of the 21th STOC Symposium*, ACM, 1989.

[138] J. Park. *Notes on Searching in Multidimensional Monotone Arrays.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, January 1989. Supervised by C. E. Leiserson.

[139] C. Phillips. Parallel graph contraction. In 1st *Symposium on Parallel Algorithms and Architectures*, ACM, 1989. To appear.

[140] C. Phillips and S. A. Zenios. Experiences with large scale network optimization on the connection machine. In *Impact of Recent Computer Advances on Operations REsearch*, Elsevier Science Publishing Co., New York, NY, 1989.

[141] S. Plotkin and E. Tardos. Improved dual network simplex. 1989. submitted for publication.

[142] S. Rao. Finding near optimal separators in planar graphs. In 28th *Annual Symposium on Foundations of Computer Science, White Plains, New York*, page 225, IEEE, 1987.

[143] J. G. Riecke. *Should a Function Continue?* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, Jan. 1989. Supervised by A.R. Meyer.

[144] R. L. Rivest and R. E. Schapire. Inference of finite automata using homing sequences. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, Seattle, Washington, May 1989. To appear.

[145] R. L. Rivest and R. Sloan. Learning complicated concepts reliably and usefully. In *Proceedings AAAI-88*, pages 635–640, American Association for Artificial Intelligence, Aug. 1988.

[146] R. L. Rivest and R. Sloan. A new model for inductive inference. In M. Vardi, editor, *Proceedings of the 2nd Annual Theoretical Aspects of Reasoning about Knowledge Conference*, pages 13–27, Morgan Kaufmann, March 1988. Submitted to *Information and Computation.*

[147] J. Rompel. *A Better Performance Guarantee for Approximate Graph Coloring.* Master's thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by Tom Leighton.

[148] R. E. Schapire. The strength of weak learnability. 1989. Submitted.

[149] A. T. Sherman. *VLSI Placement and Routing: The PI Project.* Springer-Verlag, New York, New York, 1989.

[150] D. B. Shmoys and E. Tardos. Computational complexity. In R. Graham, M. Grötschel, and L. Lovász, editors, *The Handbook of Combinatorics*, North Holland, Amsterdam, 1989. To appear.

[151] D. B. Shmoys and D. Williamson. Analyzing the Held-Karp TSP bound: a monotonicity property with application. 1989. Submitted to *Information Processing Letters.*

[152] R. Sloan. Types of noise in data for concept learning. In *First Workshop on Computational Learning Theory*, pages 91–96, Morgan Kaufmann, 1988.

[153] R. Sloan. *All Zero-Knowledge Proofs are Proofs of Language Membership.* Technical Memo TM-385, MIT Lab. for Computer Science, Feb. 1989.

[154] R. Sloan. *Computational Learning Theory:New Models and Algorithms.* Ph.D. thesis, MIT Dept. of Electrical Engineering and Computer Science, 1989. Supervised by R. L. Rivest.

[155] E. Tardos. An intersection theorem for supermatroids. *Journal of Combinatorial Theory, B*, 1989. To appear.

[156] S. Wu. An efficient algorithm for finding two edge-disjoint paths in a graph. 1988. Submitted for publication.

6 Public Lectures '88–'89 (Annotated)

[1] B. Bloom. LCF can't be topped. Lecture given at LICS 1988, June 1988.

[2] G. Brassard, D. Chaum, C. Crépeau, and I. Damgard. Conversations that don't say much! Lecture given at McGill University, Nov. 1988.

Presentation at McGill University (Montreal) of a survey talk about cryptographic protocols

[3] G. Brassard and C. Crépeau. Sorting out zero-knowledge. Lecture given at Eurocrypt, Apr. 1989.

[4] G. Brassard, C. Crépeau, and M. Yung. Everything in np can be argued in perfect zero-knowledge in a constant number of rounds. Lecture given at Eurocrypt, Apr. 1989.

[5] D. Chaum, C. Crépeau, and I. Damgård. Multiparty unconditionally secure protocols. Lecture given at SIAM meetings on Discrete Mathematics, June 1988.

[6] C. Crépeau. From photons to secret computations. Lecture given at Aarhus University, Apr. 1989.

Presentation at Aarhus University of a survey talk about reductions among cryptographic protocols.

[7] C. Crépeau. Verifiable disclosure of secrets and applications. Lecture given at Eurocrypt, Apr. 1989.

[8] C. Crépeau and J. Kilian. Achieving oblivious transfer using weakened security assumptions. Lecture given at 27th Foundation of Computer Science, Oct. 1988.

[9] C. Crépeau and J. Kilian. Cryptographic protocols based on nature's random sources. Lecture given at IBM Almaden Research Center, Aug. 1988.

Presentation at IBM ARC of [10]

[10] C. Crépeau and J. Kilian. Weakening security assumptions and oblivious transfer. In *Advances in Cryptology: Proceedings of Crypto '88, Lect. Notes in Computer Sci.*, Springer-Verlag, 1988. To appear.

We show that Oblivious Transfer can be achieved from some very weak version of this protocol and even more foundamental primitives such as noisy channels.

[11] M. D. Ernst. Polymorphic typechecking is exponential. Lecture given at Massachusetts Institute of Technology, Apr. 1989.

[12] J. Fried. Broadband module design: cost/performance tradeoffs. Lecture given at International Workshop on Physical Design of Broadband Switching and Multiplexing Equipment, Apr. 1989.

Provides an analytic model of the performance and cost of packet-routing modules for use in broadband networks.

[13] A. V. Goldberg, S. Plotkin, and E. Tardos. Combinatorial algorithms for the generalized circulation problem. In *29th FOCS Symposium*, pages 432–443, IEEE, 1988. Submitted for journal publication.

The paper considers a generalization of the maximum flow problem, in which there is no conservation of flow on the arcs of the network. More precisely, only a constant fraction $x(e)\gamma(e)$ of the flow $x(e)$ that enters a given arc e, reaches the other end. For instance, nodes of the graph can correspond to different currencies, with the multipliers being the exchange rates.

Maximizing the amount of flow into the sink is a special case of linear programming, and therefore can be solved in polynomial time by interior point methods (such as ellipsoid or Karmarkar). This presents the first polynomial time combinatorial algorithms for the generalized circulation problem. The algorithms are simple and intuitive.

[14] S. A. Goldman. Learning binary relations and total orders. Lecture given at Center for Intelligent Control Systems Machine Learning Workshop, May 1989.

[15] R. I. Greenberg. Area-universal networks. Lecture given at Polytechnic University, Princeton University, and University of Southern California, February–March 1989.

This talk will describe area-universal networks based on Leiserson's *fat-tree* architecture. Area-universality refers to the ability of a parallel computer to efficiently simulate any other machine of comparable physical size. With only modest assumptions, it is possible to build a machine of area $O(A)$ which can simulate any machine of area A in time greater only by a factor polylogarithmic in A. The emphasis on area instead of processor count is an attempt to capture the real-world constraints of wiring up large networks.

Showing a parallel machine to be area-universal involves two major steps: demonstrating a means of mapping competing network components to the universal machine so that the simulation burden is modest and providing an algorithm to efficiently route messages on the universal machine. This talk will describe how certain network structures facilitate this mapping and will discuss techniques for accomplishing the message routing. For example, a very simple circuit-switched network with a very simple randomized routing algorithm can be used to build a parallel machine of area A which can simulate any other, with $O(lg^3 A)$ degradation in bit-times.

[16] R. I. Greenberg. Efficient multi-layer channel routing. Lecture given at Georgia Institute of Technology and University of Maryland, March–April 1989.

The traditional channel routing problem assumes that two layers of material are available for routing. Advances in manufacturing technology, however, have made it practical to use three or even four layers for interconnections,

and it is possible that in the near future more interconnect layers will be available. Thus, it is important to extend channel routing to handle multi-layered regions. This extension is also meaningful for well-established technologies such as printed circuit boards and hybrid circuits.

The first part of this talk will describe the heuristics used to implement a multi-layer channel router, MulCh, which builds on ideas of the Chameleon system, developed at Berkeley. The basic approach of both programs is to partition the channel routing problem into essentially independent problems of at most three layers, but MulCh is unique in allowing one-layer partitions. In test cases, MulCh shows significant improvement over Chameleon in terms of channel width, net length, and number of vias. The work on MulCh is joint with Alex Ishii of MIT and Alberto Sangiovanni-Vincentelli of Berkeley.

The second part of this talk will discuss potential improvements to the running time of MulCh by describing efficient algorithms to carry out certain subtasks. In particular, the minimum separation for any single-layer channel routing problem can be determined in linear time, generalizing previous results for river routing. Also, since MulCh follows a greedy approach of assigning nets one at a time to what appears to be the best partition, there are computations on problem partitions which must be performed each time a net is added. Some of these computations can be performed more efficiently by taking advantage of the incremental nature of the problem. Some of this work is joint with Miller Maley of Princeton.

[17] R. I. Greenberg. MulCh: A multi-layer channel router using one, two, and three layer partitions. Lecture given at Massachusetts Institute of Technology, May 1989.

Multi-layer routing is becoming an important problem in the physical design of integrated circuits as technology evolves towards several layers of metallization. Several channel routers for three layers of interconnect have been proposed, but only one, CHAMELEON, has been implemented to accept specification of an arbitrary number of layers. CHAMELEON is based on a strategy of decomposing the multi-layer problem into two and three layer problems in which one of the layers is reserved primarily for vertical wire runs and the other layer(s) for horizontal runs. In some situations, however, it is advantageous to consider also layers that allow the routing of entire nets, using both horizontal and vertical wires. MULCH is a multi-layer channel router that extends the algorithms of CHAMELEON in this direction. MULCH can route channels with any number of layers and automatically chooses a good assignment of wiring strategies to the different layers. The algorithms have been devised so that MULCH is expected to always perform at least as well as CHAMELEON in terms of area occupied by the routing. In test cases, MULCH shows significant improvement over CHAMELEON in terms of channel width, net length, and number of vias.

[18] L. A. Hall and D. B. Shmoys. Approximation schemes for constrained scheduling problems. 1989. submitted for publication.

In this paper, a polynomial approximation scheme is presented for the problem of scheduling jobs on parallel identical machines subject to release time in order to minimize the total lateness with respect to specified deadlines. If precedence constraints are added, then an algorithm is given that delivers a solution within a factor of two of optimal. In the special case wherethere is only one machine, simpler superior algorithms are obtained. The two-machine flow shop with release dates is also considered, and a polynomial approximation scheme is given. All of the approximation schemes are based on the notion of an outline, which is a restriction on the set of feasible schedules that still contains a near optimal schedule, and yet is restrictive enough so that it is possible to use this information to compute such a schedule.

[19] L. A. Hall and D. B. Shmoys. Jackson's rule: making a good heuristic better. *Mathematics of OR*, 1989. To appear.

We consider the scheduling problem in which jobs with release times and delivery times are to be scheduled on one machine.We present a 1/3-approximation algorithm for the problem with precedence constraints among the jobs, and two ϵ-approximation algorithms for the problem without precedence constraints.Finally, we prove a strong negative result concerning a restricted version of the problem with precedence constraints that indicates that precedence constraints make the problem considerably more difficult to solve.At the core of each of the algorithms presented is Jackson's Rule–a simple but seemingly robust heuristic for the problem.

[20] A. T. Ishii. MulCh: A multi-layer channel router using one, two, and three layer partitions. Lecture given at iccad88, Nov. 1988.

Multi-layer routing is becoming an important problem in the physical design of integrated circuits as technology evolves towards several layers of metallization. Several channel routers for three layers of interconnect have been proposed, but only one, CHAMELEON, has been implemented to accept specification of an arbitrary number of layers. CHAMELEON is based on a strategy of decomposing the multi-layer problem into two and three layer problems in which one of the layers is reserved primarily for vertical wire runs and the other layer(s) for horizontal runs. In some situations, however, it is advantageous to consider also layers that allow the routing of entire nets, using both horizontal and vertical wires. MULCH is a multi-layer channel router that extends the algorithms of CHAMELEON in this direction. MULCH can route channels with any number of layers and automatically chooses a good assignment of wiring strategies to the different layers. The algorithms have been devised so that MULCH is expected to always perform at least as well as CHAMELEON in terms of area occupied by the routing. In test cases, MULCH shows significant improvement over CHAMELEON in terms of channel width, net length, and number of vias.

[21] L. Jategaonkar. Ml with extended pattern matching and subtypes. Lecture given at New York University, New York, NY; IBM Research, Hawthorne, NY, Sep. 1988.

> We extend a fragment of the programming language Standard ML to incorporate some ideas associated with "object-oriented" programming. This extended language, ML+, is an experimental step towards the longer-term goal of ML++, an ML-style language with inheritance. Some features of ML+ are polymorphic functions, function definition by pattern matching, automatic type inference, subtyping of base types, and a simple form of inheritance. In keeping with the framework of ML, a set of typing rules is developed and an algorithm for inferring most general typings is presented.

[22] J. Kilian. Theory and practice of cryptographic primitives. Lecture given at University of California, Berkeley, and Stanford University, Apr. 1989.

[23] T. Leighton. Flows, paths and VLSI layout. Lecture given at Bonn Workshop on Flows, Paths and VLSI Layout; and AWOC, June 1988.

[24] T. Leighton. Dynamic tree embeddings in butterflies and hypercubes. Lecture given at ICSI Berkeley, Jan. 1989.

[25] T. Leighton. Fast computation using faulty hypercubes. Lecture given at ACM STOC, May 1989.

[26] T. Leighton. Survey talk on networks, parallel computation and VLSI design. Lecture given at Trento School on VLSI Computation; ICALP, (July), 1988; NCUBE and Univ. of Oregon, (Jan.); Dartmouth, (May), 1989.

[27] T. Leighton. Survey talk on packet routing algorithms. Lecture given at IDA SRC, (June); U. British Columbia Distinguished Lecture Series; Stanford, (Dec.), 1988; ICSI Berkeley; IBM Almaden, (Jan.); MIT Center for Intelligent Control, (Mar.); DARPA Contractors Meeting; NSF Industry-University Symposium, (April); DIMACS Symposium Invited Lecture, (May), 1989.

[28] T. Leighton, M. Newman, A. G. Ranade, and E. Schwabe. Dynamic tree embeddings in butterflies and hypercubes. Lecture given at MIT VLSI Research Review, May 1989.

> See [29] for abstract.

[29] T. Leighton, M. Newman, A. G. Ranade, and E. Schwabe. Dynamic tree embeddings in butterflies and hypercubes. In 1^{st} *Symp. on Parallel Algorithms and Architectures*, ACM, 1989.

> We present simple randomized algorithms for dynamically embedding binary trees in either a butterfly or a hypercube network of processors. These algorithms are *dynamic* in the sense that the tree to be embedded may start as one node and grow by dynamically spawning children, where the nodes are incrementally embedded as they are spawned. Our embedding algorithms

for the hypercube and butterfly simultaneously optimize load and dilation up to constant factors (with high probability) for trees which are a logarithmic factor larger than the host network. In addition, we present an improved algorithm for embedding in the hypercube which simultaneously optimizes load and dilation up to constant factors (with high probability) for arbitrary binary trees, while also keeping congestion low. We also prove a $\Omega(\sqrt{\log N})$ lower bound on dilation for determinstic embedding algorithms which achieve optimal load, implying that any embedding algorithm which simultaneously optimizes load and dilation must be randomized.

[30] C. E. Leiserson. Very large scale computing. Lecture given at Project Mac 25^{th} Anniversary Symposium, MIT LCS, Oct. 1988.

[31] C. E. Leiserson. VLSI theory and parallel supercomputing. Lecture given at Decennial Caltech Conference on VLSI, California Institute of Technology, Pasadena, California, (Mar.); Thinking Machines Corporation, Cambridge, Massachusetts (Apr.), 1989.

[32] J. K. Lenstra, D. B. Shmoys, and E. Tardos. Scheduling unrelated parallel machines. *Mathematical Programming*, 1989. To appear. An earlier version of this appeared in the 28^{th} Symposium on Foundations of Computer Science.

In the minimum makespan problem for unrelated parallel machines, there are n jobs to be scheduled on m machines, where p_ij is the time to process job j on machine i, and the aim is to minimize the maximum completion time. A polynomial-time algorithm is given that always delivers a schedule that has maximum completion time no more than twice the optimum. It is also proved that finding $3/2-\epsilon$-approximate solutions for any fixed positive ϵ is NP-hard.

[33] B. Maggs. Universal packet routing algorithms. Lecture given at IBM Thomas J. Watson Research Center, Apr. 1989.

[34] A. R. Meyer. Observing concurrent processes: dataflow. Lecture given at MIT, Project Mac 25^{th} Anniversary Symposium, October 1988.

[35] A. R. Meyer. Semantical paradigms. Lecture given at 3^{rd} IEEE Symp. Logic in Computer Science, Edinburgh, Scotland, Invited Lecture (July); Mitre Corporation, MA (Dec.), July 1988.

[36] A. R. Meyer. An ultimate "Kahn Principle" for dataflow semantics. Lecture given at IBM Research Lab, Hawthorne, NY, Distinguished Lecture (Jan.); University of Maryland, MD (Feb.), 1989.

[37] J. Park. Notes on searching in multidimensional monotone arrays. Lecture given at 29^{th} Symp. Found. Computer Sci., October 1988.

[38] J. G. Riecke. Observing termination in Scott-style semantics. Lecture given at IBM Research, Hawthorne, NY, Nov. 1988.

The denotational semantics most appropriate for a programming language depends crucially upon the *observations* one makes about computations. Classic results by Plotkin and Sazanov for the simply typed functional language PCF show that Scott-style semantics works well for observing that the outcome of a computation is a base constant. But the standard Scott model is not adequate for observing that an evaluation terminates, because the evaluation of a lambda abstraction terminates immediately using standard interpreters, while such interpreters diverge on other terms with the same meaning.

We define a new semantical model with *lifted* functional types and prove its adequacy for observing *termination* of closed terms. We prove that with the addition of a parallel conditional and a convergence testing operator to the language, the model becomes fully abstract; with the addition of an existential-like operator, the language becomes universal. We discuss some recent extensions to these results by Stavros Cosmadakis, and highlight some preliminary results on a logic for reasoning a fragment of the language.

[39] R. L. Rivest. Inference of finite automata using homing sequences. Lecture given at Boston University, March 1989.

[40] R. L. Rivest. Learning theory: what's easy and what's hard. Lecture given at MIT, Oct. 1989.

[41] R. E. Schapire. Diversity-based inference of finite automata. Lecture given at GTE Laboratories, American Control Conference, 1988.

[42] R. E. Schapire. Inference of finite automata using homing sequences. Lecture given at Twenty-First Annual Symposium on Theory of Computing, May 1989.

[43] R. E. Schapire. The strength of weak learnability. Lecture given at Northeastern University, Center for Intelligent Control Systems Machine Learning Workshop, 1989.

[44] D. B. Shmoys. Jackson's rule: making a good heuristic better. Lecture given at CWI, EURO/TIMS, 1988.

See [19] for a complete abstract.

[45] D. B. Shmoys. Approximation schemes for constrained scheduling problems. Lecture given at Stanford, Oberwolfach, Cornell, 1989.

See [18] for a complete abstract.

[46] D. B. Shmoys. Using linear programming in the design and analysis of approximation algorithms. Lecture given at Princeton (DIMACS Theory Day), Stanford, Columbia, 1989.

See [32] and [47] for a complete abstract.

[47] D. B. Shmoys and D. Williamson. Analyzing the Held-Karp TSP bound: a monotonicity property with application. 1989. Submitted to *Information Processing Letters.*

We consider the Held-Karp lower bound procedure for the Traveling Salesman Problem, and prove that the optimal Held-Karp bound is a monotonic property, in that the bound for a subset of the input is no more than for the entire input. A corollary of this is that the bound is at least 2/3 of the optimum TSP value for any input.

[48] C. Stein. Improved algorithms for bipartite network flow. Lecture given at MIT Lab. for Computer Science, Apr. 1989.

[49] E. Tardos. Combinatorial algorithms for the generalized circulation problem. Lecture given at University of Waterloo, Mathematical Programming Symposium in Tokyo, Rutgers University, Stanford, Cornell University, SIAM workshop on optimization, 1988.

The talks were based on the paper [13].

[50] E. Tardos. Recent advances in the theory of network flow algorithms. Lecture given at Summer school on "Paths, Flows and VLSI-layout" at the Operations Research Institute, Bonn F.R.G., 1988.

This series of three talks gave a survey of recent developments in network flow theory.