NATIONAL ACADEMY OF SCIENCES

JACOB THEODORE SCHWARTZ
1930–2009

A Biographical Memoir by
MARTIN DAVIS AND EDMOND SCHONBERG

Any opinions expressed in this memoir are those of the authors and do not necessarily reflect the views of the National Academy of Sciences.

Biographical Memoir

AIP Emilio Segrè Visual Archives, Physics Today Collection.

JACOB THEODORE SCHWARTZ

January 9, 1930–March 2, 2009

BY MARTIN DAVIS AND EDMOND SCHONBERG

JACOB THEODORE ("JACK") SCHWARTZ was born on January 9, 1930, in the Bronx, New York City. His parents were secular Jews, immigrants to the United States, his mother from Germany and his father from Hungary. His father worked as a furrier, part of the vibrant women's clothing industry in New York. As a child Jack was a precocious omnivorous reader. He attended Stuyvesant High School, one of the special public schools provided in New York City for gifted students. During his high school years he became passionate about chemistry, and looked forward to a career as a scientist. He became close friends with the future physicist David Finkelstein, a friendship that was to endure for Jack's entire life. Soon after entering City College at the age of 15, his interest shifted decisively to mathematics. City College, with its tuition still free in those years, and classes reachable from home by a five-cent subway ride, provided a path to professional careers for many young people whose families were of very modest means. E. L. Post and B. P. Gill were inspiring teachers (despite very difficult working conditions), and there was real excitement about mathematics among the striking number of Jack's fellow students who were to go on to successful careers in the mathematical sciences. At the "mathematician's table" in the crowded, noisy student cafeteria, students were working on problems, setting challenges, playing chess, and teaching one another mathematics.

Jack attacked his new field with an intensity and gusto that were to become typical. Not content with the undergraduate curriculum, he began working his way through the newly published volumes in the American Mathematical Society's Colloquium series: Solomon Lefschetz's *Algebraic Topology* and Andre Weil's *Foundations of Algebraic Geometry*. In his later life he often repeated the pattern of rapidly absorbing the most advanced current literature of an area that was new to him and then making his own contribution to it.

Jack began his graduate studies at Yale in 1949. His research interest settled then and remained for some years in functional analysis, especially in the theory of linear operators. Nelson Dunford was his dissertation adviser and later his continuing collaborator. During his graduate student days, Jack married Sandra Wiener. He remained at Yale for several years after receiving the doctorate and moved to New York University in 1957, where he continued at the Courant Institute of Mathematical Sciences until his retirement in 2005. One fruit of Jack's collaboration with Dunford was the following Dunford-Schwartz Theorem:

Let T be a linear operator from L_1 to L_1 with $||T||_1 \leq 1$ and $||T||_\infty \leq 1$. Then, for every $f \in L_1$, $lim_{n \to \infty} \frac{1}{n} \sum_{k=0}^{n-1} T^k f$ exists almost everywhere.[1]

But their collaboration will especially be remembered for their monumental three-volume treatise *Linear Operators* (1958, 1963, 1970) for which the American Mathematical Society awarded them the Leroy P. Steele Prize for expository writing. The underlying conceptual framework of this work is a treatment of the basic structures of analysis in sufficiently general terms to enable applications of functional analysis to various aspects of classical analysis as well as enable a focus on contemporary topics. Thus the Lebesgue integral is developed for vector-valued functions, and the classical Cauchy complex analysis of analytic functions is generalized to analytic functions with values in a complex Banach algebra defined on a domain in complex n-space. A 140-page chapter in the first volume is devoted to investigation of the various properties of a menagerie of over two dozen function spaces. Each chapter is followed by a section of notes that fill out the historical background and include various excursions from the main topics. The list of more than 1,000 exercises, none of them routine, covers a huge range of applications.

It may not be amiss to quote from Béla Sz.-Nagy's reviews of the three volumes.

ON VOLUME I

Although only the first of the planned…volumes has so far appeared, it can already be confirmed that the authors have created an extraordinarily important and valuable work that is distinguished in particular by its monumental completeness, clear organization, and attractive exposition.[2]

ON VOLUME II

[It] is clear that its study will from now on constitute a necessary stage for every analyst-to-be. The reader will be fascinated, whatever part of the book he studies, by the immense amount of information it offers, by the clearness and simplicity of the exposition, and by the harmony of classical and modern in the whole presentation.[3]

ON VOLUME III

The treatise of Dunford and Schwartz, now accomplished, will be, for many years to come, an inexhaustible source of information on modern analysis, and it certainly will (as the first two parts already did) very strongly influence further development in this huge area of mathematical knowledge.[4]

Peter Lax, a friend and colleague of Jack's, has contributed the following reminiscence:

Of all the people I have known, Jack Schwartz had the most powerful mind save one—John von Neumann, one of Jack's heroes. They had many traits in common: a very broad range of interest in mathematics but extending beyond mathematics, a kind of restlessness, a deep interest in computing, and a desire to make the world a better place.

Jack started his research career in functional analysis; his teacher was Nelson Dunford. The work resulted in the monumental encyclopedic three volumes of Dunford-Schwartz; it contained everything known, and many things not yet known, on linear functional analysis. Gian-Carlo Rota, Jack's most brilliant student, gives an amusing description of what writing the book was like, in an essay.[5]

Very much later when I wrote my book on functional analysis I consulted Dunford-Schwartz frequently. In particular I used their proof in Volume II of an important and deep result, Lidskii's trace formula. When I turned to the bibliography, I could not find Lidskii's paper cited there. I concluded that Jack had proved the trace formula independently. When pressed hard, Jack admitted it.

When Jack founded and joined the Computer Science Department at the Courant Institute he revealed that when he came to the Mathematics Department at Courant, he looked at our Bulletin of courses and decided that he will in time teach each of them, and that he had carried out his plan. I asked him how he taught a course on a subject he knew very little about. "The summer before, I took out of the library the leading books on the subject and read them,'' he answered.

In addition to his many deep technical works in an astonishingly wide variety of subjects, Jack also wrote some lighthearted essays.[6] In particular, the essay "The Pernicious Influence of Mathematics on Science" uttered in jest but meant seriously, points out blind spots of thinking mathematically.[7]

In order to say something about Jack's work on W* algebras, (also known as von Neumann algebras) we briefly review some of the underlying concepts. A W* algebra is an algebra A of bounded operators T on a separable Hilbert space H such that

1. A is closed under the adjoint map $T \to T^*$,
2. A is closed in the weak topology.

Von Neumann sought a structure theorem for such algebras, in effect an infinite dimensional analog of the Wedderburn Structure Theorem for finite dimensional rings. As basic components he singled out W* algebras whose centers are one dimensional, later called factors, and he showed that every W* algebra is a direct integral of factors. Factors of various types have been identified:

1. Type I_n, $n = 1, \ldots, \infty$, the Kronecker product of all bounded operators on an n-dimensional Hilbert space with all bounded operators on some other Hilbert space.
2. Type II_1 and type II_∞, W* algebra*s* not of type I that admit a linear functional tr with $tr(ST) = tr(TS)$.
3. Type III, factors not of any of the above types.

In 1963 Jack proved that in a given decomposition, the set of factors of type III is Borel, von Neumann having established this for factors of types I and II much earlier.

A W* algebra is called hyperfinite if it is generated by an increasing sequence of finite dimensional subalgebras. All hyperfinite factors of type II_1 are isomorphic. Jack Schwartz said that a W* algebra A has "Property P" if for every linear operator T on the Hilbert space, the closed convex hull of $\{UTU^* \mid U \in A\}$ contains some operator that commutes with all of A. This property, which has shown itself to be of considerable importance in the subject, turns out to be equivalent to hyperfiniteness and also to Connes notion of amenability.

In 1963 Jack found a new factor of type II_1 and also one of type III, and using Property P, proved for each of them, that it was not isomorphic to either of the two factors of the corresponding type then known to exist. In addition to his discoveries about W* algebras, Jack also contributed by writing a book on the subject.[8]

While participating in a study of Karl Marx's classic *Das Capital* with a like-minded group in New York, Jack came to the conclusion that Marx had failed to confront adequately a contradiction between his economic analysis and empirical reality. Jack developed some simple economic models in an unavailing effort to convince the others in the group.[9] This was the beginning of a serious study of economic theory that led to books published in 1962 and 1965. Jack began with a theory of price ratios obtained using properties of eigenvalues of connected positive matrices and the Brouwer fixed-point theorem applied to Leontief-style input and output model economies. This was followed by a discussion of business cycles, emphasizing John Nash's suboptimal equilibria in *n*-person games in the context of Keynesian analysis. A striking feature is a model in which aspects of classical economic theory and Keynesian economics are permitted to coexist and concerning which Jack proved that the Keynesian phenomena dominate. All the models in the first book are in real terms; money plays no role. In the second smaller book, specifically about money, Jack is quite dubious concerning the monetarist theories of Milton Friedman then in vogue.[10]

In the mid-1960s Jack began to devote his creative energies to the emerging field of computer science. In 1969 he founded the Computer Science Department at New York University under the umbrella of the Courant Institute of

Mathematical Sciences. He served as chair of the department until 1977. Jack's directorship of the Information Science and Technology Office at the Defense Advanced Research Projects Agency during his two-year leave (1987-1989) from NYU enabled him to seriously influence the direction of research in computer science in the United States.

In Jack's own computer science research his initial focus was on questions of computer architecture and programming issues in connection with parallel computation. At a time when the needed technology still lay at least a decade in the future, he wrote a paper on the architecture of what he called Athena-class computers. These machines were to have a large number of separate processors operating independently but all having access to a common memory. Currently this architecture is well known under the name shared-memory MIMD (Multiple Instruction stream, Multiple Data stream).[11]

In 1978 Jack's interest returned to models of parallel computation. This time Jack took as a given that very large processing systems (involving thousands of identical processors) were not only feasible but would become available in short order, and that algorithms for such systems, as well as programming languages for them, had to be devised. In the ambitious paper he wrote[12] Jack focused on a particular architecture for the interconnection of a large number of processors: the perfect shuffle, so-called because it resembles the rearrangement of a deck of cards after half of it is inserted exactly within the other half. The advantages of this interconnection network for certain combinatorial algorithms had been known for some time. Jack showed how it could be used in numerous other areas of application (including set manipulation, matrix inversion and other linear algebra problems, and graph manipulation) and proposed programming constructs whose primitives reflected the basic operations of the architecture. In the same paper he discussed some of the physical details of a proposed ultracomputer containing 16,000 processors. This paper was the basis for the work of the NYU ultracomputer project, in which appropriate software and hardware was developed over the following decade under the guidance of Jack, together with Allan Gottlieb.

While visiting IBM, Jack worked with the IBM researchers John Cocke and Frances ("Fran") Allen in connection with the work on program optimization that they had pioneered. Later Jack and Fran married, Jack's marriage to Sandra having ended in divorce. Although Jack's second marriage was also eventually terminated by divorce, the two remained friends and admirers of each other's achievements. Jack's own interest in programming languages was triggered by working with Allen and Cocke. He collaborated with John Cocke on an encyclopedic treatment of concepts and techniques for compiler construction that, although published only as a Courant Institute Report, was quite influential. It contained the first systematic survey of parsing techniques, as well as code-generation algorithms for imperative and functional languages, and more recondite pattern-matching techniques. Far more important were optimization techniques that were truly seminal to the field of compiler development and its history. In addition to new techniques, a lasting framework for the subject was created. Compiler optimization as a subject of study began with this report, and as a result the Courant Institute became the place where the most important optimization techniques, and students, were produced. This work has been directly present in virtually every compiler ever written since, usually in much the way first laid out in the Cocke-Schwartz report.[13]

The algorithms developed at IBM for global data-flow analysis and program decomposition (interval analysis) have a natural set-theoretic expression, but these algorithms proved to be hard to implement in the programming language of choice at the time, namely Fortran. This led Jack to embark on a large effort to design and implement his programming language SETL based on set-theory, and to prove its usefulness as a specification language by recasting numerous algorithms in various areas of computer science into this language.[14] SETL underwent several implementations, and design improvements with substantial contributions from Robert Dewar and others.[15] The central feature of the language is the use of sets and mappings over arbitrary domains, as well as the use of universally and existentially quantified expressions to describe predicates and iterations over composite structures. This set-theoretic core is embedded in a conventional imperative language with familiar control structures, subprograms, recursion, and global state in order to make the language widely accessible. Conservative for its time, it did not include higher-order functions. The final version of the language incorporated a backtracking mechanism (with success and fail primitives) as well as database operations. The popular scripting and general purpose programming language Python is understood to be a descendent of SETL, and its lineage is apparent in Python's popularization of the use of mappings over arbitrary domains.

Jack saw SETL in pragmatic terms as an executable specification language whose primary goals were conciseness and clarity, and where efficiency was to be obtained by means of a separate step of translation into a lower language. The spectacular inefficiency of the initial implementation reflected this approach. This led Jack to subsidiary research endeavors into optimization techniques and global program transformations.[16] Jack also introduced another fruitful research thread by suggesting the use of finite difference techniques to transform algorithms that manipulate sets into algorithms that perform pointwise operations on them.[17]

The preliminary material on sets and mappings in *Linear Operators* goes much further into abstract set theory than might have been expected; for example, there are proofs of Zorn's lemma and of Zermelo's well ordering theorem. This attests to Jack's appreciation of the significance of the expressibility of mathematical discourse in the language of set theory. As the work of Descartes and Fermat had shown that propositions of Euclid's geometry could be translated into the language of algebra, so the efforts of Russell, Zermelo, and von Neumann showed that propositions of the various branches of mathematics could be translated into the language of set theory. Jack had a grand fourfold vision of the role that set theory could play in computer science.

1. He imagined a robust set-theory based language in which mathematicians could comfortably express their proofs.

2. There was to be a computer program to automatically verify the correctness of such proofs.

3. To be included in the theorems that could be so verified were theorems whose validity implied the correctness of specified computer programs. In order to expedite this connection these programs should be written in a programming language itself based on set theory.

4. Formalized set theory itself being undecidable, one should seek substantial fragments of the language that could be dealt with algorithmically, hopefully enabling shortcuts in proofs being verified.

Of course the SETL project can be seen as being part of the framework of this vision. But it was the fourth item that provided the richest opportunities for significant research. Jack found a clue in the work of Heinrich Behmann who had provided an algorithm for the decision problem of second-order monadic predicate calculus. Jack saw that these methods could be extended to yield algorithms for certain propositions involving finite chains of the membership relation: $x_1 \in x_2 \in \ldots \in x_n$. Over a period of decades, working with a group of collaborators, all but one from Italy, who first were students at New York University and then became distinguished scientists in their own right, a surprising collection of nontrivial mathematics was found to lie within the scope of decidable fragments of set theory. This included the fundamental properties of von Neumann ordinal numbers.[18]

With some of these same Italian researchers a computer program was designed and implemented (at least as a prototype) that could verify the correctness of mathematical proofs presented in the language of set theory. Jack proposed to use this verifier to certify the correctness of a substantial body of the fundamentals of mathematical analysis. This was to include proofs of the basic properties of the real and complex number systems defined in set-theoretic terms, the fundamental properties of limits, continuity and the differential and integral calculus, and was to culminate in a proof of the Cauchy Integral Theorem of complex analysis. Although this goal has been accomplished for the bulk of elementary analysis, Jack died before the project was complete; a posthumous publication will include a progress report.[19]

In the early 1980s Jack's interests turned to robotics, and he created the Robotics Laboratory at NYU. In his usual comprehensive approach, he was interested not only in theoretical issues in motion planning, image recognition, and the like but also in pragmatic issues in robot programming and machine manufacturing. His work on motion planning, with Micha Sharir, led to a series of papers under the title "The Piano Movers' Problem" that founded a new field of algorithmic research. The piano movers' problem in its full generality can be described as follows:

Given an open subset U of n-dimensional Euclidean space, and two compact subsets C_0 and C_1 of U, where C_1 can be obtained from C_0 by a continuous motion, is it possible to move C_0 to C_1 while remaining entirely within U.

They first examined the problem of two-dimensional rigid polygonal bodies moving amidst polygonal barriers. They then reformulated the general problem as one in algebraic topology, and used Collins's decomposition and Tarski's decision procedure for the first order theory of real numbers to derive complexity bounds for the general problem. Among other results they showed that the problem for algebraic bodies having a fixed number of degrees of freedom can be solved in time polynomial in the number of geometric constraints present in the problem. The problem remains exponential in the number of degrees of freedom of these bodies. A subsequent paper tackled the simpler problem of two-dimensional circular bodies and polygonal barriers, and presented a $O(N^3)$ algorithm for two circles moving within N walls. There was further work on complexity in which John Hopcroft also collaborated.[20]

In addition to the Steele Prize from the American Mathematical Society for *Linear Operators* already mentioned and his election to the National Academy of Sciences in 1976, Jack Schwartz was the recipient of a number of honors: He was a Sloan Foundation fellow during 1961-1962. He received the Wilbur Cross Medal from Yale University and the Townsend Harris Medal from his Alma Mater, City College. He was awarded the Mayor's Medal for Contributions to Science and Technology, New York City, and was elected to the American Academy of Arts and Sciences. He had about 30 doctoral students in mathematics and computer science.

Jack Schwartz died on March 2, 2009, of metastasized prostate cancer that had been kept at bay by years of treatment. Jack didn't permit the disease to stop him from continuing to work up to almost the very end. He is survived by his wife of 23 years, Diana Robinson Schwartz; sister, Judith; two daughters; and two grandchildren.

NOTES

1. N. Dunford and J. T. Schwartz. Convergence almost everywhere of operator averages. *J. Rational Mech.* 5:129-178.

2. *Zbl. Math.* 84(1960):104. Although the reviews of volumes II and III are in English, that of volume I is in German. The original German of the excerpt given here is: *"Obwohl nur der erste der geplanten [zwei] Bände bisher erschienen ist, kann schon festgestellt werden, daβ die Verff. ein auβerordentlich groβes und wertvolle Werk geschaffen haben, das sich besonders durch seine monumentale Vollständigkeit, klare Struktur und anziehende Darstellungsweise auszeichnet."*

3. *Zbl. Math.* 128(1967):351. This review of volume II was coauthored with C. Foias.

4. *Zbl. Math.* 243(1970):317.

5. G.-C. Rota. *Indiscreet Thoughts.* Boston: Birkhäuser, 1997.

6. M. Kac, G.-C. Rota, and J. Schwartz. *Discrete Thoughts: Essays on Mathematics, Science and Philosophy.* Boston: Birkhäuser, 1992.

7. *Ibid.* This essay first appeared in *Proceedings of the 1960 International Congress on Logic, Philosophy, and the Methodology of Science,* pp. 356-360. Palo Alto, Calif.: Stanford University Press, 1961.

8. *W* Algebras.* New York: Gordon and Breach, 1967. This section on W* Algebras was written by Jacob Feldman to whom the authors wish to express their appreciation.

9. See M. Davis. Jack Schwartz meets Karl Marx. In *From Linear Operators to Computational Biology: Essays in Memory of Jacob T. Schwartz (1929-2009),* ed. E. Schoenberg. Springer, to appear.

10. J. T. Schwartz. *Lectures on the Mathematical Method in Analytical Economics.* New York: Gordon and Breach, 1962. *Theory of Money.* New York: Gordon and Breach, 1965.

11. Large parallel computers. *J. ACM* 13(1966):25-32.

12. Ultracomputers. *Trans. Program. Lang. Syst.* 2(1980):484-521.

13. J. Cocke and J. T. Schwartz. Programming languages and their compilers. *Courant Institute Lecture Notes,* New York, 1968. We are indebted to Josh Fisher for help with this section.

14. J. T. Schwartz. *On Programming: An Interim Report on the SETL Project. Installments I and II.* New York: Courant Institute Publications, New York University, 1975.

15. J. T. Schwartz, R. Dewar, E. Dubinsky, and E. Schonberg. *Programming with Sets: An Introduction to SETL.* New York: Springer-Verlag, 1986.

16. J. T. Schwartz. Optimization of very high level languages. *J. Program. Lang.* 1(1975):161-218. Automatic data structure choice in a language of very high level. *Commun. ACM* 18(1975):722-728.

17. R. Paige and J. T. Schwartz. Expression continuity and the formal differentiation of algorithms. *Proceedings of the 4th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages*, pp. 58-71. New York: ACM, 1977.

18. See, for example, D. Cantone, A. Ferro, and E. Omodeo. *Computable Set Theory.* New York: Clarendon Press, 1989.

19. J. T. Schwartz, D. Cantone, and E. G. Omodeo. *Computational Logic and Set Theory: Applying Formalized Logic to Analysis.* Springer, to appear.

20. J. T. Schwartz and M. Sharir. On the piano movers' problem. I. The special case of a rigid polygonal body moving amidst polygonal barriers. *Commun. Pure Appl. Math.* 36(1983):345-398. II. General techniques for computing topological properties of real algebraic manifolds. *Adv. Appl. Math.* 4(1983):298-351. III. Coordinating the motion of several independent bodies. The special case of circular bodies moving amidst polygonal barriers. *Robotic. Res.* 2(1983):46-75. IV. The case of a rod moving in three-dimensional space amidst polyhedral obstacles. *Commun. Pure Appl. Math.* 37(1984):815-848.
J. E. Hopcroft, J. T. Schwartz, and M. Sharir. On the complexity of motion planning for multiple independent objects; PSPACE hardness of the "warehouseman's problem." *Int. J. Robotic. Res.* 3(1984):76-88.
Efficient detection of intersections among spheres. *Int. J. Robotic. Res.* 4(1985):77-80.

SELECTED BIBLIOGRAPHY

1955

De Rham's theorem for arbitrary spaces. *Am. J. Math.* 77:29-44.

1956

With N. Dunford. Convergence almost everywhere of operator averages. *J. Rational Mech.* 5:129-178.

With W. G. Bade. On Mautner's eigenfunction expansion. *Proc. Natl. Acad. Sci. U. S. A.* 42:519-525.

1958

With N. Dunford. *Linear Operators,* vol. I. *General Theory*. New York: Interscience. (Reprinted Wiley, 1988.)

1962

Lectures on the Mathematical Method in Analytical Economics. New York: Gordon and Breach.

1963

With N. Dunford. *Linear Operators,* vol. II. *Spectral Theory*. New York: Interscience. (Reprinted Wiley, 1988.)

Two finite, non-hyperfinite, non-isomorphic factors. *Commun. Pure Appl. Math.* 16:19-26.

Non-isomorphic factors of type III. *Commun. Pure Appl. Math.* 16:111-120.

With H. S. Shapiro and D. J. Newman. Generators of the convolution algebra L_1. *Trans. Am. Math. Soc.* 107:466-484.

1966

Large parallel computers. *J. ACM* 13:25-32.

1967

*W*Algebras*. New York: Gordon and Breach.

1969

With J. Cocke. *Programming Languages and Their Compilers*. New York: Courant Institute.

1970

With N. Dunford. *Linear Operators,* vol. III. *Spectral Operators.* New York: Interscience. (Reprinted Wiley, 1988.)

1975

Automatic data structure choice in a language of very high level. *Commun. ACM* 18:722-728.

1979

With M. Davis. Metamathematical extensibility for theorem verifiers and proof-checkers. *Comput. Math. Applic.* 5:217-230.

1980

Fast probabilistic algorithms for verification of polynomial identities. *J. ACM* 27:701-717.

With A. Ferro, M. Breban, and E. Omodeo. Decision procedures for elementary sublanguages of set theory. Part I. Multi-level syllogistic and its extensions. *Comm. Pure Appl. Math.* 33:599-608.

1981

With A. Ferro, M. Breban, and E. Omodeo. Decision procedures for elementary sublanguages of set theory. Part II. A decidable class of quantified formulae. *Commun. Pure Appl. Math.* 34:177-195.

1982

With A. Gottlieb. Networks and algorithms for large-scale parallel computation. *IEEE Comput. Mag.* 15:27-36.

1983

With M. Sharir. On the piano movers' problem. I. The special case of a rigid polygonal body moving amidst polygonal barriers. *Commun. Pure Appl. Math.* 36:345-398.

1984

With M. Sharir. On the piano movers' problem. IV. The case of a rod moving in three-dimensional space amidst polyhedral obstacles. *Commun. Pure Appl. Math.* 37:815-848.

With J. E. Hopcroft and M. Sharir. On the complexity of motion planning for multiple independent objects; PSPACE hardness of the "warehouseman's problem." *Int. J. Robotic. Res.* 3:76-88.

1985

With J. E. Hopcroft and M. Sharir. Efficient detection of intersections among spheres. *Int. J. Robotic. Res.* 4:77-80.

1986

With E. Schonberg, E. Dubinsky, and R. Dewar. *Higher Level Programming: An Introduction to the Set-Theoretic Language SETL.* New York: Springer-Verlag.

1995

With A. Policriti. T-theorem proving. I. *J. Symb. Comput.* 20:315-342.