RC 5767
(#25010)
12/17/75
Computer
Science

22 pages

BIBLIOGRAPHY ON PROGRAM OPTIMIZATION

F. E. Allen
IBM Thomas J. Watson Research Center
Yorktown Heights, New York 10598

RC 5767
(#25010)
12/17/75
Computer
Science

22 pages

BIBLIOGRAPHY ON PROGRAM OPTIMIZATION

F. E. Allen
IBM Thomas J. Watson Research Center
Yorktown Heights, New York 10598

ABSTRACT: A comprehensive bibliography of the English language documents related to program optimization is given. Relevant documents on program analysis, program verification techniques and graph theory are also listed. CACM refers to the Communications of the ACM, JACM refers to the Journal of the ACM, and JCSS refers to the Journal of Computer and System Science. Other abbreviations should be self-evident.

December 15, 1975

Abel, Norma E., and Bell, James R. Global optimization in compilers: a unified approach. Proc. First USA-Japan Computer Conf., AFIPS Press, Montvale, N.J., 1972, 437-441.

Abrahams, Paul. Compiler pessimization. Datamation, April 1, 1971, 32-33.

Abrams, P. S. An APL Machine. Doctoral Thesis, Stanford Univ., Jan. 1970.

Agresti, W. A. Nonserial dynamic programming for register allocation. 45th ORSA/TIMS Joint National Meeting, Boston, Mass., April 22, 1974.

Aho, A. V., and Johnson, S. C. Optimal code generation for expression trees. Proc. Seventh Annual Symposium on the Theory of Computing, Albuquerque, New Mexico (May 1975), 207-217.

Aho, A. V., Sethi, R., and Ullman, J. D. A formal approach to code optimization. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 86-100.

Aho, A. V., Sethi, Ravi, and Ullman, J. D. Code optimization and finite Church-Rosser systems. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1972, 89-105.

Aho, A. V., and Ullman, J. D. Transformations on straight line programs. Proc. Second Annual ACM Symposium on Theory of Computing, (May 1970), 136-148.

Aho, A. V., and Ullman, J. D. Optimization of straight line programs. SIAM J. of Computing 1, 1 (March 1972), 1-19.

Aho, A. V., and Ullman, J. D. Equivalence of programs with structured variables. JCSS 6,2 (1972), 125-137.

Aho, A. V., and Ullman, J. D. The Theory of Parsing, Translation, and Compiling, Vol. 2, Prentice-Hall, Englewood Cliffs, N. J., 1973.

Aho, A. V., and Ullman, J. D. Node listings for reducible flow graphs. Proc. Seventh Annual Symposium on the Theory of Computing, Albuquerque, New Mexico, (May 1975), 177-185.

Allard, R. W., Wolf, K. A., and Zemlin, R. A. Some effects of the 6600 computer on language structures. CACM 7, 2 (Feb. 1964), 112-119.

Allen, F. E. Program optimization. Annual Review in Automatic Programming, Vol. 5, Pergamon, New York, 1969, 239-307.

Allen, F. E. Control flow analysis. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5,7 (July 1970), 1-19.

Allen, Frances E. A basis for program optimization. Proc. IFIP Congress 71, North Holland Publishing Co., Amsterdam, 1971, 385-390.

Allen F. E. Interprocedural data flow analysis. Proc. IFIP Congress 74, North Holland Publishing Co., Amsterdam, 1974, 398-402.

Allen, F. E. A method for determining program data relationships. International Symposium on Theoretical Programming. Ershov, Andrei and Nepomniaschy, Valery A. (Eds.), Lecture Notes in Computer Science, Vol. 5, Springer-Verlag, Heidelberg, Germany, 1974, 299-308.

Allen, F. E. Interprocedural analysis and the information derived by it. Programming Methodology: Lecture Notes in Computer Science, VOL. 23, Springer-Verlag, Heidelberg, Germany, 1975, 291-321.

Allen, F. E., and Cocke, J. A catalogue of optimizing transformations. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1972, 1-30.

Allen, F. E., and Cocke, J. Graph-theoretic constructs for program control flow analysis. IBM Research Report RC3923, T. J. Watson Research Center, Yorktown Heights, N. Y., July 1972.

Allen, F. E. and Cocke, J. A program data flow analysis procedure. IBM Research Report RC5257, T. J. Watson Research Center, Yorktown Heights, N. Y., Feb. 1975.

Allen, F. E., Cocke, J., and Kennedy, K. Reduction of operator strength. Technical Report 476-093-6, Dept. of Mathematical Sciences, Rice Univ., Houston, Texas, August 1974.

Anderson, J. P. A note on some compiling algorithms. CACM 7, 3 (March 1964), 149-150.

Aron, E. Notes on compiler optimization. Technical Report 74-8, Dept. of Computer Science, Hebrew University, Jerusalem, Israel, 1974.

Bachmann, P. A contribution to the problem of the

optimization of programs. Proc. IFIP Congress 71, North Holland Publishing Co., Amsterdam, 1971, 397-401.

Backus, J. W., Beeber, R. J., Best, S., Goldberg, R., Haibt, L. M., Herrick, H. L., Nelson, R. A., Sayre, D., Sheridan, P. B., Hughes, R. A., and Nutt, R. The FORTRAN automatic coding system. Proceedings of the Western Joint Computer Conf. Los Angeles, Calif., Feb. 1957, 188-198. Also in Rosen, S. (Ed.) Programming Systems and Languages. McGraw-Hill, New York, N. Y., (1967), 29-47.

Baer, Jean-Loup, and Caughey, Robert. Segmentation and optimization of programs from cyclic structure analysis. Spring Joint Computer Conference-1972, AFIPS Press, Montvale, N. J., 40 (May 1972), 23-36.

Bagwell, John T., Jr. Local optimizations. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 52-66.

Ballard, A., and Tsichritzis, D. Transformations on programs. IFIP Congress 71, North Holland Publishing Co., Amsterdam, 1971, 89-93.

Bauer, F. L., and Eickel, J., (Eds.). Compiler Construction. Lecture Notes in Computer Science, Vol. 21, Springer-Verlag, Heidelberg, Germany, 1975.

Beatty, James C. An axiomatic approach to code optimization for expressions. JACM 19, 4 (October 1972), 613-640. Errata: JACM 20, 1 (Jan. 1973), 188 and JACM 20, 3 (July 1973), 538.

Beatty, J. C. A global register assignment algorithm. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1972, 65-88.

Beatty, J. C. An algorithm for tracing live variables based on a straightened program graph. IBM Technical Report TR 00.2503, Poughkeepsie, New York, December 1973.

Beatty, J. C. A register assignment algorithm for generation of highly optimized object code. IBM Journal of Res. and Dev. 18, 1 (January 1974), 20-39.

Blum, D., Brown, S. K., Calavano, A. G., Hempy, H. O., and Suez, J. Current technologies in FORTRAN object code optimization. IBM Technical Report TR 00.2240, Poughkeepsie, N. Y., 1971.

Bolas, B. J. Optimization problems in extensible compilers (abstract). Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 127.

Breuer, M. A. Generation of optimal code for expressions via factorization. CACM 12, 6 (June 1969), 333-340.

Bruno, J., and Lassagne, T. The generation of optimal code for stack machines. JACM 22, 3 (July 1975), 382-396.

Bruno, J., and Sethi, R. Register allocation for a one-register machine. Technical Report No. 157, Computer Science Dept., Pennsylvania State University, October 3, 1974.

Burkhardt, W. H. Automation of program speed-up on parallel-processor computers. Computing, 3, (1968), 297-310.

Burstall, R. M., and Darlington, J. Some transformations for developing recursive programs. International Conf. on Reliable Software, Los Angeles, Calif., April, 1975, 465-472.

Busam, V. A., and Englund, D. E. Optimization of expressions in FORTRAN. CACM 12, 12 (December 1969), 666-674.

Carter, J. L. A case study of a new compiling and code generation technique. IBM Research Report RC5666, T. J. Watson Research Laboratory, Yorktown Heights, N. Y., Oct., 1975.

Chandra, A. K. Efficient compilation of linear recursive procedures. IBM Research Report RC4517, T. J. Watson Research Laboratory, Yorktown Heights, N. Y., August 1973.

Cheatham, T. E., Jr., and Standish, Thomas A. Optimization aspects of compiler-compilers. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 37-51.

Chroust, G. Scope conserving expression evaluation. Proceedings IFIP 71, North Holland Publishing Co., Amsterdam, TA-3, 178-182.

Cocke, J. Global common subexpression elimination. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 20-24.

Cocke, J. On certain graph-theoretic properties of programs. IBM Research Report RC3391, T. J. Watson Research Center, Yorktown Heights, N. Y., June 1971.

Cocke, J., and Kennedy, K. An algorithm for reduction of operator strength. Technical Report 476-093-2, Dept. of

Mathematical Sciences, Rice Univ., Houston, Texas, March 1974.

Cocke, J., and Kennedy, K. Profitability computations on program flow graphs. Technical Report 476-093-3, Dept. of Mathematical Sciences, Rice Univ., Houston, Texas, May 1974.

Cocke, J., and Miller, R. E. Some analysis techniques for optimizing computer programs. Proc. 2nd International Conference of System Sciences, Hawaii, 1969, 143-146.

Cocke, J., and Schwartz, J. T. Programming Languages and Their Compilers: Preliminary Notes, Courant Institute of Mathematical Sciences, New York University, New York, N.Y., 1970.

Cohagan, W. L. Vector optimization for the ASC. Proc. 7th Annual Princeton Conf., (March 1973), 169-174.

Darlington, J. A semantic approach to automatic program improvement. Doctoral Thesis, University of Edinburgh,, 1972.

Darlington, J. Application of program transformation to program synthesis. Proceedings IRIA Conference on Proving and Improving Programs, Rocquencourt, France, July 1975, 133-144.

Darlington, J., and Burstall, R. M. A system which automatically improves programs. Proc. 3rd Intl. Conf. on Artificial Intelligence, Stanford, California, August 1973, 537-542.

Davis, M. Interval analysis and program optimization. (unpublished notes) Harvard University, 1973.

Day, W. H. E. Compiler assignment of data items to registers. IBM Systems Journal, 9, 4 (1970), 281-317.

Earley, J. High level iterators and a method of automatically designing data structure representation. College of Engineering Memorandum ERL-M416, University of California, Berkeley, California, February 1974.

Earnest, C. Some topics in code optimization. JACM 21, 1 (January 1974), 76-102.

Earnest, C. P., Balke, K. G., and Anderson, J. Analysis of graphs by ordering of nodes. JACM 19, 1 (January 1972), 23-42.

Elfield, Mark H., and Cohn, Charles E. Improved subroutine efficiency through calling-sequence modification.

Software Age, 4, 2 (February 1970), 5-6.

Elson, M., and Rake, S. T. Code generation for large-language compilers. IBM Systems Journal 9, 3 (1970), 166-188.

Erickson, David B. Array processing on an array processor. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 17-24.

Ershov, A. P. Reduction of the problem of the memory allocation in programming to the problem of coloring the vertices of a graph. Soviet Mathematics 3 (1962), 163-165.

Ershov (also Yershov), A. P. ALPHA - an automatic programming system of high efficiency. JACM 13, 1 (January 1966), 17-24.

Ershov (also Yershov), A. P. (Ed.). The ALPHA Automatic Programming System. Academic Press, London and New York, 1971.

Ershov, A. P. A multilanguage programming system oriented to language description and universal optimization algorithms. ALGOL 68 Implementations, Peck J. E. L. (Ed.), North-Holland Publishing Co., Amsterdam, 1971, 143-162.

Ershov, A. P., Zmiyenskaya, L. L., Mishkovitch, R. D., and Trokhan, L. K. Economy and allocation of memory in the ALPHA-translator. The Alpha Automatic Programming System, Ershov (Yershov), A. P. (Ed.), Academic Press, London and New York, 1971.

Fateman, R. Optimal code for serial and parallel computation. CACM 12, 12 (December 1969), 694-695.

Finkelstein, M. A compiler optimization technique. The Computer Journal 11, 1 (May 1968), 22-25.

Floyd, R. W. An algorithm for coding efficient arithmetic operations. CACM 4, 1 (January 1961), 42-51.

Fong, Amelia, Kam, John, and Ullman, Jeffrey. Application of lattice algebra to loop optimization. Conf. Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 1-9.

Fosdick, L. D. BRNANL, a FORTRAN program to identify basic blocks in FORTRAN programs. Report 40, Dept. of Computer Science, University of Colorado, Boulder, Colorado, March 1974.

Frailey, Dennis. Expression optimization using unary complement operators. Proceedings of the Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 67-85.

Frailey, D. A Study of Optimization Using a General Purpose Optimizer. Doctoral thesis, Purdue University, Lafayette, Indiana, January 1971.

Freiburghouse, R. A. Register allocation via usage counts. CACM 17, 11 (Nov. 1974), 638-642.

Galler, B. A., and Perlis, A. J. Compiling matrix operations. CACM 5, 12 (Dec. 1962), 590-594.

Gear, C. W. High speed compilation of efficient object code. CACM 8, 8 (August 1965), 483-488.

Gerhart, Susan L. Correctness-preserving program transformations. Conference Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 54-66.

Geschke, Charles M. Global Program Optimizations. Doctoral thesis, Computer Science Department, Carnegie-Mellon University, Pittsburgh, Pa., 1972.

Giammo, C. S., and Siegel, S. Some aspects of global optimization in the Honeywell Series 6000 FORTRAN compiler. Honeywell Computer Journal, 7, 2 (1973), 144-150.

Goldberg, P. C. Compilers, (unpublished draft). IBM, Yorktown Heights, N. Y., 1971.

Goldberg, Patricia C. A comparison of certain optimization techniques. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1972, 31-50.

Graham, Susan L., and Wegman, Mark. A fast and usually linear algorithm for global flow analysis (extended abstract). Conference Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 22-34.

Gries, David. Compiler Construction for Digital Computers. John Wiley and Sons, New York, 1971.

Hall, Patrick A. V. Common subexpression identification in general algebraic systems. Technical Report UKSC 0060, IBM UK Scientific Centre, Neville Road, Peterlee, England, Nov. 1974.

Hammer, M. Optimization problems in very high level languages (unpublished memorandum). July 1974.

Han, J. C. Tree height reduction for parallel processing of blocks of FORTRAN assignment structures. National Technical Information Service PB-207985, 1972.

Hansen, G. J. Adaptive Systems for the Dynamic Run-Time Optimization of Programs. Doctoral Thesis, Carnegie-Mellon University, Pittsburgh, Pa., 1974.

Harrison, William. A class of register allocation algorithms. IBM Research Report RC5342, T. J. Watson Research Laboratory, Yorktown Heights, N. Y., March 1975.

Harrison, William. Compiler analysis of the value range of variables. IBM Research Report RC5544, T. J. Watson Research Laboratory, Yorktown Heights, N. Y., July 1975.

Hecht, M. S. Flow Analysis of Computer Programs. American Elsevier, New York, (to appear in 1976).

Hecht, M. S., and Ullman, J. D. Flow graph reducibility. SIAM J. Computing 1, 2 (June 1972), 188-202.

Hecht, M. S., and Ullman, J. D. Analysis of a simple algorithm for global flow problems. Conf. Record of ACM SIGACT/SIGPLAN Symposium on Principles of Programming Languages, Boston, Mass., October 1973, 202-217.

Hecht, Matthew S., and Ullman, Jeffrey D. A simple algorithm for global data flow analysis problems. SIAM J. of Computing, 4, 4 (Dec. 1975), 519-532.

Hecht, M. S., and Ullman, J. D. Characterizations of reducible flow graphs. JACM, 21, 3 (July 1974), 367-375.

Hill, V., Langmaack, H., Schwarz, H. R., and Seegmuller, G. Efficient handling of subscripted variables in ALGOL 60 compilers. Proceedings of the Rome Symposium on Symbolic Languages in Data Processing, Gordon and Breach, N. Y., 1962, 331-340.

Hoar, C. A. R. Subscript optimization and subscript checking. Algol Bulletin No. 29, Nov. 1968.

Hopgood, F. R. A. Compiling Techniques, American Elsevier, New York, 1969.

Hopkins, M. An optimizing compiler design. Proc. IFIP Congress 71, North Holland Publishing Co., Amsterdam,

1971, 391-396.

Horwitz, L. P., Karp, R. M., Miller, R. E., and Winograd, S. Index register allocation. JACM 13, 1 (January 1966), 43-61.

Huxtable, D. H. R. On writing an optimizing translator for ALGOL 60. Introduction to System Programming. Wegner, P., (Ed.), Academic Press Inc., New York, 1964, 137-155.

Ingalls, D. The execution time profile as a programming tool. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1971, 107-128.

Jasik, S. Monitoring Execution on the CDC 6000's. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall (1971), 129-136.

Kam, John B. Flow graph reducibility of certain subclasses of flow charts. Technical Report No. 141, Dept. of Electrical Engineering, Princeton Univ., Princeton, N. J., October 1973.

Kam, J. B., and Ullman, J. D. Global optimization problems and iterative algorithms. Technical Report No.146, Dept. of Electrical Engineering, Princeton Univ., Princeton, N. J., Jan. 1974.

Kam, John, and Ullman, Jeffrey D. Monotone data flow analysis frameworks. Technical Report No.169, Dept. of Electrical Engineering, Princeton Univ., Princeton, N. J., Jan. 1975.

Kasyanov, V. N. Some properties of fully reducible flow graphs. Information Processing Letters 2,4 (1973), 113-117.

Katz, Shmuel and Manna, Zohar. Logical analysis of programs. Technical Report, Dept. of Applied Mathematics, The Weizman Institute of Science, Rehovot, Israel, September 1974.

Kennedy, K. A global flow analysis algorithm. International J. Computer Math., 3 (December 1971), 5-15.

Kennedy, K. Safety of code motion. International J. Computer Math., 3 (December 1971), 117-130.

Kennedy, Ken. Index register allocation in straight line code and simple loops. Design and Optimization of Compilers, Rustin, R. (Ed.), Prentice-Hall, Englewood Cliffs, N. J., 1972, 51-63.

Kennedy, K. Global dead computation elimination. SETL Newsletter # 111, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., August, 1973.

Kennedy, K. An algorithm to compute compacted use-definition chains. SETL Newsletter # 112, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., August 1973.

Kennedy, K. A comparison of algorithms for global flow analysis. Technical Report 476-093-1, Dept. of Mathematical Sciences, Rice University, Houston, Texas, February 1974.

Kennedy, K. Variable subsumption with constant folding. SETL Newsletter # 123, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., February 1974.

Kennedy, K. Schaefer's node splitting algorithm. SETL Newsletter # 125, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., February 1974.

Kennedy, K. Edge listing data flow algorithms. SETL Newsletter # 127, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., March 1974.

Kennedy, K. Node listings applied to data flow analysis. Conference Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 10-21.

Kennedy, K. Use-definition chains with applications. Technical Report 476-093-9, Dept. of Mathematical Sciences, Rice Univ., Houston, Texas, 1975.

Kennedy, K., and Owens, P. An algorithm for use-definition chaining. SETL Newsletter # 37, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1971.

Kennedy, K., and Pfeiffer, P. A Markov model of program flow. Technical Report 476-093-4, Dept. of Mathematical Sciences, Rice University, Houston, Texas, May 1974.

Kernighan, Brian W. Optimal segmentation points for programs. Proceedings 2nd Annual ACM Symposium on Operating Systems Principles, Princeton, N. J., October 1969, 47-53.

Kernighan, B. W. Optimal sequential partitions of graphs. JACM 18, 1 (January 1971), 34-40.

Kildall, G. A code synthesis filter for basic block optimization. Technical Report TR 72-01-01, University of Washington, Computer Science Group, University of Washington, Seattle, Washington, January 1972.

Kildall, G. Global Expression Optimization During Compilation. Doctoral thesis, TR-72-06-02, Dept. of Computer Science, University of Washington, Seattle, Washington, June 1972.

Kildall, G. A. A unified approach to global program optimization, Record of ACM Symposium on Principles of Programming Languages, Boston, Mass., October 1973, 194-206.

King, James C. Symbolic execution and program testing. IBM Research Report RC5082, T. J. Watson Research Center, Yorktown Heights, N. Y., October 1974.

Kleir, R. L. and Ramamoorthy, C. V. strategies for optimizing microprograms. IEEE Transactions on Computers, C-20, 7 (July 1971), 783-794.

Knuth, D. E. An empirical study of FORTRAN programs. Software -- Practice and Experience 1, 2 (1971), 105-133.

Knuth, Donald E. and Stevenson, Francis R. Optimal measurement points for program frequency counts. BIT 13 (1973), 313-322.

Knuth, D. E. Structured programming with GO TO statements. ACM Computing Surveys 6, 4 (December 1974), 261-301.

Kosaraju, S. R. Analysis of structured programs. Journal of Computer and Systems Sciences 9,3 (1974), 232-255.

Kou, Lawrence T. On live-dead analysis for global data flow problems. IBM Research Report RC5278, T. J. Watson Research Center, Yorktown Heights, New York, February 1975.

Kral, J. One way of estimating frequencies of jumps in a program. CACM 11, 7 (July 1968), 475-480.

Lamport, Leslie. On programming parallel computers. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 25-33.

Letichevsky, A. A. Equivalence and optimization of programs. International Symposium on Theoretical Programming: Ershov, Andrei, and Nepomniaschy, Valery,

(Eds.), Lecture Notes in Computer Science, Vol. 5, Springer-Verlag, Heidelberg, Germany, 1974, 111-123.

Lomet, D. B. Data flow analysis in the presence of procedure calls. IBM Research Report RC5728, T. J. Watson Research Center, Yorktown Heights, N. Y. Nov., 1975.

Loveman, David B. and Faneuf, Ross A. Program opimization - theory and practice. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 97-102.

Loveman, David, Sattley, Kirk and Bearisto, David. Development of compiler optimization techniques. CADD-7407-2311, Mass. Computer Associates, Inc.

Low, James Richard. Automatic Coding: Choice of Data Structures. Doctoral thesis, Stanford University, August 1974, also Technical Report # 1, the Dept. of Computer Science, University of Rochester, Rochester, N. Y., 1974.

Lowry, E., and Medlock, C. W. Object code optimization. CACM 12, 1 (January 1969), 13-22.

Luccio, F. A comment on index register allocation. CACM 10, 9 (September 1967), 572-574.

Maggiolo-Schettini, A., and Strong, H. R. A graph-theoretic algorithm with applications for transforming recursive programs. AICA Convegno di Informatica Teoria, Pisa, Italy, March 1973.

Maggiolo-Schettini, A., Rosen, B. K., and Strong, H. R. Procedure linkage optimization: Working paper. Conference record of ACM Symposium on Principles of Programming Languages, Boston, Mass., October 1973, 183-193.

Markowsky, George. Lower bounds on lengths of basic and elementary node sequencies for reducible flow graphs. IBM Research Report RC5330, T. J. Watson Research Center, Yorktown Heights, N.Y., March 1975.

Markowsky, George and Tarjan, R. Endre. Lower bounds on the lengths of node sequencies in directed graphs. IBM Research Report RC5477, T. J. Watson Research Center, Yorktown Heights, N.Y., April 1975.

McCabe, Thomas J. Compiler optimization of vector oriented languages. (unpublished report) March 1972.

McKeeman, W. M. Peephole optimization. CACM 8, 7 (July

1965), 443-444.

Mendicino, S., and Zwackenberg, R. A FORTRAN code optimizer for the CDC 6600. UCRL-14162, Lawrence Radiation Lab, Livermore, California, April 1965.

Meyers, W. S. Optimization of computer code. (unpublished memorandum) G. E. Research Center, Schenectady, N. Y. 1965.

Millstein, R. E. Optimization for an array computer. (abstract) Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 129.

Nakata, Ikuo. On compiling algorithms for arithmetic expressions, CACM 10, 8 (August 1967), 492-494.

Neel D. and Amirchahy, M. Semantic attributes and improvement of generated code. Proceedings ACM Annual Conf., San Diego, Calif., (Nov. 1974), 1-10.

Neel D. and Amirchahy, M. Removal of invariant statements from nested-loops in a single effective compiler pass. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines. SIGPLAN Notices 10,3 (March 1975), 87-96.

Nelson, H. L. Program optimizing techniques for the CDC 6600 central processor. Technical Report, April 1965.

Nievergelt, J. On the automatic simplification of computer programs. CACM 8, 6 (June 1965), 366-370.

Osterweil, Leon J., and Fosdick, Lloyd D. Data flow analysis as an aid in documentation assertion generation, validation, and error detection. Report #CU-CS-055-74, Dept. of Computer Science, University of Colorado, Boulder, Colorado, September 1974.

Osterweil, Leon, and Fosdick, L. D. DAVE, a comprehensive data flow analysis system for FORTRAN programs. (in preparation)

Painter, James A. Effectiveness of an optimizing compiler for arithmetic expressions. Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 101-126.

Pottosin, I. V. Economy of expressions in the ALPHA-translator. The Alpha Automatic Programming System, Ershov (Yershov), A. P. (Ed.), Academic Press, London and New York, 1971, 149-159.

Presberg, David L. and Johnson, Neil W. The paralyzer:

IVTRAN's parallelism analyzer and synthesizer. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 9-16.

Prosser, R. T. Applications of Boolean matrices to the analysis of flow diagrams. Proc. AFIPS 1959 Eastern Joint Computer Conference, Spartan Books, Washington, D.C., 133-138.

Purdom, P. W., and Moore, E. F. Immediate predominators in a directed graph. CACM, 15, 8 (August 1972), 777-778.

Ramamoorthy, C. V. Analysis of graphs by connectivity considerations. JACM 13, 2 (April 1966), 211-222.

Ramamoorthy, C. V. and Gonzalez, M. J. Subexpression ordering in the execution of arithmetic expressions. CACM 14, 7 (July 1971), 479-485.

Redziejowski, R. R. On arithmetic expressions and trees. CACM 12, 2 (Feb. 1969), 81-84.

Rogers, A. H. Optimization in PL/I and its effect on language development. Technical Report TR.12.112, IBM United Kingdom, Hursley Park, Winchester, England, June 1973.

Rohl, J. S., and Linn, J. A. A note on compiling arithmetic expressions. Computer Journal, 15 (1972), 13-14.

Rosen, B. K. Data flow analysis for recursive PL/I programs. IBM Research Report RC5211, T. J. Watson Research Center, Yorktown Heights, N.Y., January 1975.

Rosen, Barry K. Note on semantics and optimization. IBM Research Report RC4935, T. J. Watson Research Center, Yorktown Heights, N. Y., July 1974.

Rosen, Barry K. Correctness of parallel programs: the Church-Rosser approach. Proceedings IRIA Conference on Proving and Improving Programs, Rocquencourt, France, July 1975, 115-132.

Rosen, B. K. How to preclude optimization. IBM Research Report RC5468, T. J. Watson Research Center, Yorktown Heights, N. Y., June 1975.

Rosen, B. K. High level data flow analysis, part 1. IBM Research Report RC5598, T. J. Watson Research Center, Yorktown Heights, N. Y., August 1975.

Rosen, B. K. High level data flow analysis, part 2. IBM

Research Report RC5744, T. J. Watson Research Center, Yorktown Heights, N. Y., December 1975.

Rustin, R., (Ed.). Design and Optimization of Compilers, Courant Computer Science Symposium 5, Prentice-Hall, Englewood Cliffs, N. J., 1972.

Ryan, J. A direction-independent algorithm for determining the forward and backward computer points for a term or subscript during compilation. Computer Journal 9, 2 (August 1966), 157-160.

Ryland, Christopher. Some optimization techniques for an extensible language. Conference Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 35 (+ handout).

Schaefer, M. A Mathematical Theory of Global Program Optimization, Prentice-Hall, Englewood Cliffs, New Jersey, 1973.

Schneck, Paul B. A survey of compiler optimization techniques, Report NASA-TM-X-69499, Goddard Institute for Space Studies.

Schneck, Paul B. Automatic recognition of vector and parallel operators in a higher level language. Proceedings ACM Annual Conf., Boston, Mass., 1972, 772-779.

Schneck, Paul B. Movement of implicit parallel and vector expressions out of loops. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 103-106.

Schneck, P. B., and Angel, Ellinor. A FORTRAN to FORTRAN optimizing compiler. The Computer Journal 16, 4 (Nov. 1973), 322-330.

Schneider, V. On the number of registers needed to evaluate arithmetic expressions. BIT 11, 1 (1971), 84-93.

Schwartz, J. T. Reduction in strength (or Babbage's difference engine in modern dress), (unpublished memorandum) IBM, Menlo Park, California, 1967.

Schwartz, J. T. Various graph-theoretical algorithms related to the optimization of compiler-generated code. (draft), Courant Institute of Mathematical Sciences, New York, N.Y., 1971.

Schwartz, J. T. More detailed suggestions concerning "data strategy" elaborations in SETL programs, SETL NEWSLETTER

#39, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., 1972.

Schwartz, J. T. More local and semi-local SETL optimizations, SETL Newsletter, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., January 1974.

Schwartz, J. T. Automatic and semiautomatic optimization of SETL. SIGPLAN Notices 9, 4 (April 1974), 43-49.

Schwartz, J. T. A few peephole optimizations applicable to iterators. SETL Newsletter #122A, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. Still more miscellaneous optimizations, SETL Newsletter #122B, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. Deducing relationships of inclusion and membership in SETL programs. SETL Newsletter #130, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., May 1974.

Schwartz, J. T. More on copy optimization of SETL programs. SETL Newsletter #131, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., June 1974.

Schwartz, J. T. Some optimizations using type information. SETL Newsletter #132, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., June 1974.

Schwartz, J. T. A higher-level control diction. SETL Newsletter #133, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., June 1974.

Schwartz, J. T. Additional pursue block examples. SETL Newsletter #133A, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. Inter-procedural optimization. SETL Newsletter #134, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. Introductory lecture at the June 28 informal optimization symposium. SETL Newsletter #135, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. Structureless programming, or the notion of 'rubble', and the reduction of programs to rubble. SETL Newsletter #135A, Courant Institute of Mathematical

Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. T. A framework for certain kinds of high-level optimization. SETL Newsletter #136, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. Additional thoughts concerning automatic data structure choice. SETL Newsletter #137, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., July 1974.

Schwartz, J. Optimization by set suppression. SETL Newsletter #138A, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., August 1974.

Schwartz, J. T. Automatic data structure choice in a language of very high level. Conference Record of the Second ACM Symposium on Principles of Programming Languages, Palo Alto, California, January 1975, 36-40.

Schwartz, J. T. Optimization of very high level languages-I. Value transmission and its corollaries. Computer Languages, 1, 2 (June 1975), 161-194.

Schwartz, J. T. Optimization of very high level languages-II. Deducing relationships of inclusion and membership. Computer Languages, 1, 3 (Sept. 1975), 197-218.

Sethi, R. Complete register allocation problems. SIAM J. of Computing, 4, 3 (Sept. 1975), 226-248.

Sethi, Ravi. A note on implementing parallel assignment instructions. Inf. Proc. Letters 2 (1973), 91-95.

Sethi, R. Testing for the Church-Rosser property. JACM 21 (1974), 671-679.

Sethi, R., and Ullman, J. D. The generation of optimal code for arithmetic expressions. JACM 17, 4 (October 1970), 715-728.

Sheridan, P. B. The arithmetic translator-compiler of the IBM FORTRAN automatic coding system. CACM 2,2 (Feb. 1959), 9-21.

Sites Richard L. Some thoughts on proving clean termination of programs. Stanford Tech. Report STAN-CS-74-417, Computer Sciences Dept., Stanford Univ., Stanford, Calif., May 1974.

Sites, R. L. Proving That Programs Terminate Cleanly. Doctoral thesis, STAN-CS-74-418, Computer Sciences Dept.,

Stanford University, Stanford, California, May 1974.

Spillman, T. C. Exposing side effects in a PL/I optimizing compiler. Proc. IFIP Congress 71, North Holland Publishing Co., Amsterdam, 1971, 376-381.

Stockhausen, P. F. Adapting optimal code generation for arithmetic expressions to the instruction sets available on present-day computers. CACM 16, 6 (June 1973), 353-354. Erratum: CACM 16, 10 (Oct. 1974), 591.

Strong, H. R. Translating recursive equations into flowcharts. JCSS 5 (1971), 254-285.

Strong, H. R. Jr. Maggiolo-Schettini, A., and Rosen, B. K., Recursion structure simplification. SIAM J. Comput. 4, 3 (Sept. 1975), 307-320.

Tapscott, Robert P. ADS: the source listing annotator. IBM Research Report RC5065, T. J. Watson Research Center, Yorktown Heights, N. Y., October 1974.

Tarjan, R. E. Depth-first search and linear graph algorithms. SIAM J. Comput. 1, 2 (September 1972), 146-160.

Tarjan, R. E. Finding dominators in directed graphs. SIAM J. Comput., 3, 1 (March 1974), 62-89.

Tarjan, Robert. Testing flow graph reducibility. Proc. of Fifth Annual ACM Symposium in Theory of Computing, Austin, Texas, April 1973, 96-107.

Tennenbaum, A. Type Determination for Very High Level Languages. Doctoral Thesis, Courant Computer Science Report No. 3, Courant Institute of Mathematical Sciences, New York University, N.Y., N.Y., 1974.

Tennenbaum, A. Compile time type determination in SETL. Proceedings of the ACM Annual Conference, San Diego, California, November 1974, 95-100,

Thompson, J. R. Some topics on the optimization of PL/I. Technical Report TR.12.118, IBM United Kingdom Laboratories Limited, Hursley Park, Winchester, England, March 1973.

Udin, D. On register allocation in the presence of common subexpressions. (unpublished notes) Harvard University, 1973.

Ullman, J. D. Fast algorithms for the elimination of common subexpressions. Acta Informatica, 2, 3 (January 1974), 191-213.

Ullman, J. D. A survey of data flow analysis techniques. 2nd USA-Japan Computer Conf. Proceedings, AFIPS Press, Montvale, N. J., 1975, 335-342.

Urschler, G. Complete redundant expression elimination in flow diagrams. IBM Research Report RC4965, T. J. Watson Research Center, Yorktown Heights, N. Y., August 1974.

Vuillemin, J. Correct and optiminal implementations of recursion in a simple programming language. JCSS 9, 3 (Dec. 1974), 332-354.

Vyssotsky, V., and Wegner, P. A graph theoretical FORTRAN source language analyzer. (unpublished) Bell Telephone Laboratories Report., September 1963.

Wagner, R. A. Some Techniques for Algebraic Optimization with Application to Matrix Arithmetic Expressions. Doctoral Thesis, Carnegie-Mellon University, June 1968.

Waite, W. M. Optimization. Compiler Construction: Lecture Notes in Computer Science Vol. 21, Springer-Verlag, Heidelberg, Germany, 1974, 549-602.

Walker, S. A., and Strong, H. R. Characterizations of flowchartable recursions. JCSS 7 (1973), 407-447.

Warshall, S. A theorem on boolean matrices. JACM 9, 1 (Jan. 1962), 11-12.

Wasilew, S. G. A Compiler Writing System with Optimization Capabilities for Complex Order Structures. Doctoral thesis, Northwestern University, Evanston, Illinois, 1971.

Wedel, Dorothy. FORTRAN for the Texas Instruments ASC system. Proc. of a Conf. on Programming Languages and Compilers for Parallel and Vector Machines, SIGPLAN Notices 10,3 (March 1975), 119-132.

Wegbreit, Ben. The ECL programming system. Proceedings Fall Joint Computer Conference, AFIPS Press, Montvale, N.J., 1971, 253-262.

Wegbreit, Ben. Mechanical program analysis. CACM 18, 9 (Sept. 1975), 528-539.

Wegbreit, Ben. Property extraction in well-founded property sets. IEEE Transactions on Software Engineering. (Sept. 1975), 270-285.

Wegbreit, Ben. State description model of peephole optimization. (unpublished notes) Harvard Univ., 1973.

Wegbreit, Ben. Goal-directed program transformation. (unpublished report) Sept. 1975.

Wegman, Mark. General and Efficient Methods for Global Code Improvement. Doctoral Thesis, Univ. of Calif. at Berkley, Berkley, Calif. (to be published).

Wulf, William A., Johnsson, Richard K., Weinstock, Charles B., Hobbs, Steven O. and Geschke, Charles M. The Design of an Optimizing Compiler. American Elsevier, New York, 1975.

Yershov, A. P. (see Ershov, A. P.)

Yhap, E. Global register assignment using interval partition and piecewise processing. IBM Research Report RC5015, T. J. Watson Research Center, Yorktown Heights, N. Y., September 1974.

Yhap, E. F. General register assignment in presence of data flow. IBM Research Report RC5645, T. J. Watson Research Center, Yorktown Heights, N. Y., Sept. 1975.

Zelkowitz, M. V. and Bail, W. G. Optimization of structured programs. Software Practice and Experience 4 (1974), 51-57.

Zwakenberg, R. G. CDC 6600/7600 optimization. (abstract) Proc. ACM SIGPLAN Symposium on Compiler Optimization, SIGPLAN Notices 5, 7 (July 1970), 130.

Acknowlegements. The author wishes to thank Al Aho, Bruce Daniels, Matt Hecht, Brian Marks, Barry Rosen, Paul Schneck, Ravi Sethi, and Jeff Ullman for the many corrections and additions they made to the preliminary version of this bibliography.