

Wolfe, Michael; Banerjee, Utpal

Data dependence and its application to parallel processing. (English) Zbl 0639.68019

Int. J. Parallel Program. 16, No. 2, 137-178 (1987).

Summary: Data dependence testing is required to detect parallelism in programs. In this paper data dependence concepts and data dependence direction vectors are reviewed. Data dependence computation in parallel and vector constructs as well as serial 'do' loops is covered. Several transformations that require data dependence are given as examples, such as vectorization (translating serial code into vector code), concurrentization (translating serial code into concurrent code for multiprocessors), scalarization (translating vector or concurrent code into serial code for a scalar uniprocessor), loop interchanging and loop fusion. The details of data dependence testing including several data dependence decision algorithms are given.

MSC:

68Q60 Specification and verification (program logics, model checking, etc.)

68N25 Theory of operating systems

Cited in **2** Documents

Keywords:

Fortran; parallelism detection; vectorization; data dependence

Full Text: [DOI](#)

References:

- [1] Clifford N. Arnold, Vector Optimization on the Cyber 205, Proc. of the Int'l Conf. on Parallel Processing, (eds.), H. J. Siegel and Leah Siegel St. Charles, IL, IEEE Computer Society Press, Washington, DC, pp. 530-536. (August 1983).
- [2] American National Standards Institute, X3J3 Committee, Draft Proposed Revised American National Standard Programming Language Fortran, version S8.104, (September, 1987).
- [3] Mark D. Guzzi, CEDAR Fortran Programmer's Manual, Document no. 601, draft, Center for Supercomputing Research and Development, Univ. of Ill. at Urbana-Champaign, (March 1987).
- [4] D. Kuch, R. H. Kuhn, B. Leasure and M. Wolfe, The Structure of an Advanced Vectorizer for Pipelined Processors, Proc. of COMPSAC 80, The 4th Int'l Computer Software and Applications Conf., Chicago, IL, pp. 709-715 (October 1980).
- [5] David Kuck, Robert Kuhn, David Padua, Bruce Leasure, Michael Wolfe, Dependence Graphs and Compiler Optimizations, Proc. of the 8th ACM Symp. on Principles of Programming Languages (POPL), Williamsburg, VA, pp. 207-218, (January 1981).
- [6] John R. Allen and K. Kennedy, PFC: A Program to Convert Fortran to Parallel Form, Tech. Rpt. MASC TR82-6, Rice Univ., Houston, TX, (March 1982).
- [7] Michael Burke and Ron Cytron, Interprocedural Dependence Analysis and Parallelization, Proc. of the SIGPLAN '86 Symp. on Compiler Construction, Palo Alto, CA, (June 1986); also available as SIGPLAN Notices, 21(7):162-175 (July 1986).
- [8] David J. Kuck, Ahmed H. Sameh, Ron Cytron, Alexander V. Veidenbaum, Constantine D. Polychonopoulos, Gyungho Lee, Tim McDaniel, Bruce R. Leasure, Carol Beckman, James R. B. Davies, and Clyde P. Kruskal, The Effects of Program Restructuring, Algorithm Change and Architecture Choice on Program Performance, Proc. of the Int'l Conf. on Parallel Processing, (ed.), Robert M. Keller, St. Charles, IL, IEEE Computer Society Press, Washington, DC, pp. 129-138 (August 1984).
- [9] David Kuck, The Structure of Computers and Computations, Vol. I, John Wiley & Sons, Inc., New York, (1978).
- [10] Utpal Banerjee, Shyh-ching Chen, David Kuck, and Ross Towle, Time and Parallel Processor Bounds for Fortran-like Loops, IEEE Trans. on Computers, C-28, (9):660-670, (September 1979). · [Zbl 0419.68020](#) · [doi:10.1109/TC.1979.1675434](#)
- [11] John R. Allen and Ken Kennedy, Automatic Loop Interchange, Proc. of the ACM SIGPLAN '84 Symposium on Compiler Construction, Montreal, Canada; SIGPLAN NOTICES 19(6):233-246 (June 1984). · [doi:10.1145/502874.502897](#)
- [12] Michael Wolfe, Optimizing Supercompilers for Supercomputers, Ph.D. Thesis, Dept. of Comp. Sci. Rpt. No. 82-1105 Univ. of Illinois, Urbana, IL, Oct., 1982; available as document 83-03027 from University Microfilms, Ann Arbor, MI. · [Zbl 0699.68007](#)
- [13] Samuel P. Midkiff, and David A. Padua, Compiler Generated Synchronization for DO Loops, Proc. of the Int'l Conf. on Parallel Processing, St. Charles, IL, IEEE Computer Society Press, Wash., DC, pp. 544-551 (August 1986).
- [14] David A. Padua and Michael J. Wolfe, Advanced Compiler Optimizations for Supercomputers, Comm. of the ACM, 29(12):1184-1201 (December 1986). · [doi:10.1145/7902.7904](#)
- [15] John R. Allen, Dependence Analysis for Subscripted Variables and Its Application to Program Transformations, Ph.D. thesis, Rice Univ., Houston, TX, (April 1983); available as document 83-14916 from University Microfilms, Ann Arbor, MI.

- [16] Utpal Banerjee, Data Dependence in Ordinary Programs, Univ. of Ill. at Urb.-Champ. Dept. of Comp. Sci. Rpt. No. 76-837, (November 1976).
- [17] H. Griffin, Elementary Theory of Numbers, McGraw-Hill, New York, (1954). · [Zbl 0058.03203](#)
- [18] A. M. Kirch, Elementary Number Theory, Intext, New York, (1974). · [Zbl 0304.10001](#)
- [19] Utpal Banerjee, Speedup of Ordinary Programs, Univ. of Ill. at Urb.-Champ. Dept. of Comp. Sci. Rpt. No. 79-989, (October 1979); available as document 80-08967 from University Microfilms, Ann Arbor, MI.
- [20] William L. Cohagan, Vector Optimization for the ASC, Proc. of the Seventh Annual Princeton Conf. on Information Sciences and Systems, Princeton Univ., pp. 169-174 (March 1973).
- [21] Leslie Lamport, The Parallel Execution of DO Loops, Comm. of the ACM, 17(2):83-93 (February 1974). · [Zbl 0273.68012](#) · [doi:10.1145/360827.360844](https://doi.org/10.1145/360827.360844)
- [22] R. H. Kuhn, Optimization and Interconnection Complexity for: Parallel Processors, Single-Stage Networks, and Decision Trees, Ph.D. Thesis, Univ. of Ill. at Urb.-Champ., Dept. of Comp. Sci Rpt. No. 80-1009, (February 1980); available as document 80-26541 from University Microfilms, Ann Arbor, MI.

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.