

BIBLIOGRAPHY

1. W. B. Ackerman. Data flow languages, February 1982.
2. T. L. Adam, K. M. Chandy, and J. R. Dickson. A comparison of list schedules for parallel processing systems. *Communications of the ACM*, 17:685–689, 1974.
3. I. Ahmad and M. K. Dhodhi. Multiprocessor scheduling in a genetic paradigm. *Parallel Computing*, 22:395–406, 1996.
4. I. Ahmad and Y.-K. Kwok. On exploiting task duplication in parallel program scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 9(8):872–892, September 1998.
5. I. Ahmad, Y.-K. Kwok, and M.-Y. Wu. Performance comparison of algorithms for static scheduling of DAGs to multiprocessors. In *Proceedings of the 7th IEEE Symposium on Parallel and Distributed Processing (SPDP'95)*, pages 185–192, Fremantle, Western Australia, September 1995.
6. I. Ahmad, Y.-K. Kwok, and M.-Y. Wu. Analysis, evaluation, and comparison of algorithms for scheduling task graphs on parallel processors. In G.-J. Li, D. F. Hsu, S. Horiguchi, and B. Maggs, editors, *Second International Symposium on Parallel Architectures, Algorithms, and Networks*, 1996, pages 207–213, June 1996.
7. I. Ahmad, Y.-K. Kwok, M.-Y. Wu, and W. Shu. Automatic parallelization and scheduling of programs on multiprocessors using CASCH. In *Proceedings of the 1997 International Conference on Parallel Processing (ICPP'97)*, pages 288–291, Bloomingdale, Illinois, USA, August 1997.
8. A. Aiken and A. Nicolau. Perfect pipelining: a new loop parallelization technique. In *Proceedings of 1988 European Symposium on Programming*, pages 221–235, 1988.
9. A. Alexandrov, M. Ionescu, K. E. Schauser, and C. Scheimann. LogGP: incorporating long messages into the LogP-model—one step closer towards a realistic model for parallel computation. In *7th Annual Symposium on Parallel Algorithms and Architectures*, pages 95–105. ACM Press, 1995.
10. H. H. Ali and H. El-Rewini. The complexity of scheduling interval orders with communication is polynomial. *Parallel Processing Letters*, 3(1):53–58, 1993.
11. R. Allen and K. Kennedy. Conversion of control dependence to data dependence. In *Proceedings of the 10th ACM Symposium on Principles of Programming Languages*, January 1983.
12. R. Allen and K. Kennedy. *Optimizing Compilers for Modern Architectures*. Morgan Kaufmann/Academic Press, 2002.

13. P. Banerjee, E. W. Hodges IV, D. J. Palermo, J. A. Chandy, J. G. Holm, S. Ramaswamy, M. Gupta, A. Lain, and E. Su. An overview of the paradigm compiler for distributed-memory multicomputers. *IEEE Computer*, 28(10):37–47, October 1995.
14. U. Banerjee. *Speedup of Ordinary Programs*. PhD thesis, University of Illinois at Urbana-Champaign, Department of Computer Science, October 1979.
15. U. Banerjee. *Dependence Analysis for Supercomputing*. Kluwer Academic Publishers, 1988.
16. U. Banerjee. An introduction to a formal theory of dependence analysis. *The Journal of Supercomputing*, 2(2):133–149, 1988.
17. U. Banerjee, R. Eigenmann, A. Nicolau, and D. A. Padua. Automatic program parallelization. *Proceedings of the IEEE*, 81(2):211–243, February 1993.
18. O. Beaumont, V. Boudet, and Y. Robert. The iso-level scheduling heuristic for heterogeneous processors. In *PDP'2002, 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing*. IEEE Computer Society Press, 2002.
19. O. Beaumont, V. Boudet, and Y. Robert. A realistic model and an efficient heuristic for scheduling with heterogeneous processors. In *HCW'2002, the 11th Heterogeneous Computing Workshop*. IEEE Computer Society Press, 2002.
20. M. S. T. Benten and S. M. Sait. Genetic scheduling of task graphs. *International Journal of Electronics*, 77(4):401–415, 1994.
21. C. Berge. *Graphs and Hypergraphs*, 2nd edition, North-Holland, 1976.
22. K. A. Berman and J. L. Paul. *Algorithms: Sequential, Parallel, and Distributed*. Thomson/Course Technology, 2005.
23. W. Blume and R. Eigenmann. The range test: a dependence test for symbolic, non-linear expressions. In *Proceedings Supercomputing '94*, pages 528–537, November 1994.
24. W. Blume, R. Eigenmann, J. Hoeflinger, D. A. Padua, P. M. Petersen, L. Rauchwerger, and P. Tu. *Automatic Detection of Parallelism: A Grand Challenge for High-Performance Computing*. Technical Report TR1348, Center for Supercomputing Research and Development, University of Illinois at Urbana-Champaign, 1994.
25. C. Boeres and V. E. F. Rebello. A versatile cost modelling approach for multicomputer task scheduling. *Parallel Computing*, 25(1):63–86, January 1999.
26. S. Booth, J. Fisher, and M. Bowers. *Introduction to the Cray T3E at EPCC*. Edinburgh Parallel Computing Centre, Scotland, UK, June 1999. <http://www.epcc.ed.ac.uk/t3d/documents/t3e-intro.html>.
27. P. Brucker. *Scheduling Algorithms*, 4th edition, Springer-Verlag, 2004.
28. P. Brucker, J. Hurink, and W. Kubiak. Scheduling identical jobs with chain precedence constraints on two uniform machines. *Mathematical Methods Operational Research*, 49(2):211–219, 1999.
29. P. Brucker and S. Knust. *Complexity Results for Scheduling Problems*. Technical Report, Mathematics Institute, University of Osnabrück, Germany, 2006-. <http://www.mathematik.uni-osnabrueck.de/research/OR/class/>.
30. P. Chrétienne. A polynomial algorithm to optimally schedule tasks over a virtual distributed system under tree-like precedence constraints. *European Journal of Operational Research*, 43:225–230, 1989.
31. P. Chrétienne. Task scheduling over distributed memory machines. In *Proceedings of the International Workshop on Parallel and Distributed Algorithms*. North-Holland, Amsterdam, 1989.

32. P. Chrétienne. Task scheduling with interprocessor communication delays. *European Journal of Operational Research*, 57:348–354, 1992.
33. P. Chrétienne. Tree scheduling with communication delays. *Discrete Applied Mathematics*, 49(1–3):129–141, 1994.
34. P. Chrétienne, E. G. Coffman, J. K. Lenstra, and Z. Liu, editors. *Scheduling Theory and Its Applications*. Wiley, 1995.
35. P. Chrétienne and C. Picouleau. Scheduling with communication delays: a survey. In P. Chrétienne, E. G. Coffman, J. K. Lenstra, and Z. Liu, editors, *Scheduling Theory and Its Applications*, pages 65–90. Wiley, 1995.
36. B. Cirou and E. Jeannot. Triplet: a clustering scheduling algorithm for heterogeneous systems. In *Proceedings of Workshop on Scheduling and Resource Management for Cluster Computing (ICPP 2001)*, pages 231–236, Valencia, Spain, September 2001. IEEE Press.
37. E. G. Coffman, editor. *Computer and Job-Scheduling Theory*. Wiley, 1976.
38. E. G. Coffman and R. L. Graham. Optimal scheduling for two-processor systems. *Acta Informatica*, 1:200–213, 1972.
39. M. Coli and P. Palazzari. Global execution time minimization by allocating tasks in parallel systems. In *Euromicro Workshop on Parallel and Distributed Processing, 1995*, pages 91–97, January 1995.
40. J.-Y. Colin and P. Chrétienne. CPM scheduling with small interprocessor communication delays and task duplication. *Operations Research*, 39(3):680–684, 1991.
41. S. A. Cook. The complexity of theorem proving procedures. In *Proceedings of 3rd ACM Symposium on Theory of Computing*, pages 151–158, 1971.
42. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*, 2nd edition, MIT Press, 2001.
43. R. C. Correa, A. Ferreira, and P. Rebreyend. Integrating list heuristics into genetic algorithms for multiprocessor scheduling. In *IEEE Symposium on Parallel and Distributed Processing 1996*, pages 462–469, 1996.
44. R. C. Correa, A. Ferreira, and P. Rebreyend. Scheduling multiprocessor tasks with genetic algorithms. *IEEE Transactions on Parallel and Distributed Systems*, 10(8):825–837, August 1999.
45. M. Cosnard and D. Trystram. *Parallel Algorithms and Architectures*. Thomson Computer Press, London, UK, 1995.
46. D. E. Culler, R. M. Karp, D. A. Patterson, A. Sahay, E. E. Santos, K. E. Schauer, R. Subramonian, and T. von Eicken. LogP: a practical model of parallel computation. *Communications of the ACM*, 39(11):78–85, November 1996.
47. D. E. Culler, R. M. Karp, D. A. Patterson, A. Sahay, K. E. Schauer, E. E. Santos, R. Subramonian, and T. von Eicken. LogP: towards a realistic model of parallel computation. *ACM SIGPLAN Notices, Proceedings of the Symposium on Principles and Practice of Parallel Programming*, 28(7):1–12, July 1993.
48. D. E. Culler and J. P. Singh. *Parallel Computer Architecture*. Morgan Kaufmann Publishers, 1999.
49. S. Darbha and D. P. Agrawal. Scalable scheduling algorithm for distributed memory machines. In *Eighth IEEE Symposium on Parallel and Distributed Processing, 1996*, pages 84–91, October 1996.

50. S. Darbha and D. P. Agrawal. Optimal scheduling algorithm for distributed-memory machines. *IEEE Transactions on Parallel and Distributed Systems*, 9(1):87–95, January 1998.
51. S. Darbha and D. P. Agrawal. SDBS: a task duplication based optimal scheduling algorithm. In *Proceedings of the Scalable High-Performance Computing Conference, 1994*, pages 756–763, May 1994.
52. A. Darte, Y. Robert, and F. Vivien. *Scheduling and Automatic Parallelization*. Brinkhäuser, Boston, USA, 2000.
53. A. L. Davis and R. M. Keller. Data flow program graphs. *IEEE Computer*, 15:26–41, February 1982.
54. L. Davis. *Handbook of Genetic Algorithms*. Van Nostrand-Reinhold, New York, USA, 1991.
55. M. K. Dhodhi, I. Ahmad, A. Yatama, and I. Ahmad. An integrated technique for task matching and scheduling onto distributed heterogeneous computing systems. *Journal of Parallel and Distributed Computing*, 62(9):1338–1361, September 2002.
56. J. Du, J. Y.-T. Leung, and G. H. Young. Scheduling chain-structured tasks to minimize makespan and mean flow time. *Information and Computation*, 92(2):219–236, 1991.
57. P.-F. Dutot, O. Sinnen, and L. Sousa. *A Note on the Complexity of Task Scheduling with Communication Contention*. Technical Report, University of Auckland, New Zealand, February 2005.
58. C. Eisenbeis and J.-C. Sogno. A general algorithm for data dependence analysis. In *International Conference on Supercomputing, Washington DC, USA*, pages 292–302. ACM Press, August 1992.
59. J. Eisenbiegler, W. Löwe, and A. Wehrenpfennig. On the optimization by redundancy using an extended LogP model. In *International Conference on Advances in Parallel and Distributed Computing (APDC'97)*, pages 149–155. IEEE Computer Society Press, 1997.
60. H. El-Rewini and M. Abd-El-Barr. *Advanced Computer Architecture and Parallel Processing*. Wiley, 2005.
61. H. El-Rewini and H. H. Ali. On considering communication in scheduling task graphs on parallel processors. *Journal of Parallel Algorithms and Applications*, 3:177–191, 1994.
62. H. El-Rewini and H. H. Ali. Static scheduling of conditional branching in parallel programs. *Journal of Parallel and Distributed Computing*, 24(1):41–54, 1995.
63. H. El-Rewini and T. G. Lewis. Scheduling parallel program tasks onto arbitrary target machines. *Journal of Parallel and Distributed Computing*, 9(2):138–153, June 1990.
64. H. El-Rewini and T. G. Lewis. *Distributed and Parallel Computing*. Manning, 1998.
65. H. El-Rewini, T. G. Lewis, and H. H. Ali. *Task Scheduling in Parallel and Distributed Systems*. Prentice Hall, 1994.
66. B. Falsafi and D. A. Wood. Scheduling communication on an SMP node parallel machine. In *Proceedings of IEEE International Symposium on High Performance Computer Architecture*, pages 128–138, 1997.
67. M. J. Flynn. Very high-speed computing systems. *Proceedings of the IEEE*, 54:1901–1909, 1966.
68. S. Fortune and J. Wyllie. Parallelism in random access machines. In *Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, pages 114–118, May 1978.

69. I. Foster. *Designing and Building Parallel Programs*. Addison-Wesley, 1995.
70. M. Fujii, T. Kasami, and K. Ninomiya. Optimal sequencing of two equivalent processors. *SIAM Journal of Applied Mathematics*, 17(3):784–789, 1969.
71. H. Gabow. An almost linear algorithm for two-processor scheduling. *Journal of the ACM*, 29(3):766–780, 1982.
72. M. R. Garey and D. S. Johnson. “Strong” NP-completeness results: motivation, examples and implications. *Journal of the ACM*, 25(3):499–508, 1978.
73. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, 1979.
74. M. R. Garey, D. S. Johnson, R. Tarjan, and M. Yannakakis. Scheduling opposing forests. *SIAM Journal of Algebraic and Discrete Methods*, 4(1):72–93, 1983.
75. A. Gerasoulis, J. Jiao, and T. Yang. A multistage approach for scheduling task graphs on parallel machines. In *Workshop on Parallel Processing of Discrete Optimization Problems*, pages 81–103. American Mathematical Society, 1994.
76. A. Gerasoulis and T. Yang. A comparison of clustering heuristics for scheduling DAGs on multiprocessors. *Journal of Parallel and Distributed Computing*, 16(4):276–291, December 1992.
77. A. Gerasoulis and T. Yang. On the granularity and clustering of directed acyclic task graphs. *IEEE Transactions on Parallel and Distributed Systems*, 4(6):686–701, June 1993.
78. D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
79. D. K. Goyal. *Scheduling Processor Bound Systems*. Technical Report CS-7-036, Computer Science Department, Washington State University, Pullman, 1996.
80. R. L. Graham. Bounds for multiprocessing timing anomalies. *SIAM Journal of Applied Mathematics*, 17(2):416–419, 1969.
81. R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, 5:287–326, 1979.
82. A. Grama, A. Gupta, G. Karypis, and V. Kumar. *Introduction to Parallel Computing*, 2nd edition, Pearson Addison Wesley, London, UK, 2003.
83. T. Hagras and J. Janeček. A high performance, low complexity algorithm for compile-time task scheduling in heterogeneous systems. *Parallel Computing*, 31(7):653–670, 2005.
84. C. Hamacher, Z. Vranesic, and S. Zaky. *Computer Organization*, 5th edition, McGraw-Hill, 2002.
85. C. Hanen and A. Munier Kordon. Minimizing the volume in scheduling an out-tree with communication delays and duplication. *Parallel Computing*, 28(11):1573–1585, November 2002.
86. C. Hanen and A. Munier. An approximation algorithm for scheduling dependent tasks on m processors with small communication delays. In *ETFA 95:INRIA/IEEE Symposium on Emerging Technology and Factory Automation*, pages 167–189. IEEE Press, 1995.
87. J. L. Hennessy and D. A. Patterson. *Computer Organization and Design: The Hardware/Software Interface*, 2nd edition. Morgan Kaufmann Publishers, San Francisco, USA, 1998.
88. J. L. Hennessy and D. A. Patterson. *Computer Architecture, A Quantitative Approach*, 3rd edition. Morgan Kaufmann Publishers, San Francisco, USA, 2003.

89. J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, USA, 1975.
90. J. A. Hoogeveen, J. K. Lenstra, and B. Veltman. *Three, Four, Five, Six or the Complexity of Scheduling with Communication Delays*. Technical Report BS-R9229, ISSN 0924-0659, Centre for Mathematics and Computer Science, The Netherlands, October 1992.
91. J. A. Hoogeveen, S. L. van de Velde, and B. Veltman. Complexity of scheduling multi-processor tasks with prespecified processor allocations. *Discrete Applied Mathematics*, 55(3):259–272, 1994.
92. E. S. H. Hou, N. Ansari, and H. Ren. Genetic algorithm for multiprocessor scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 5(2):113–120, February 1994.
93. T. Hu. Parallel sequencing and assembly line problems. *Operations Research*, 9(6):841–848, 1961.
94. J. J. Hwang, Y. C. Chow, F. D. Anger, and C. Y. Lee. Scheduling precedence graphs in systems with interprocessor communication times. *SIAM Journal of Computing*, 18(2):244–257, April 1989.
95. K. Hwang and F. A. Briggs. *Computer Architecture and Parallel Processing*. McGraw-Hill, London, UK, 1984.
96. IBM. *SP Switch2 Technology and Architecture*, March 2001. http://www-1.ibm.com/servers/eserver/pseries/hardware/whitepapers/sp_switch2.pdf.
97. H. Jung, L. M. Kirousis, and P. Spirakis. Lower bounds and efficient algorithms for multi-processor scheduling of directed acyclic graphs with communication delays. *Information and Computation*, 105(1):94–104, 1993.
98. T. Kalinowski, I. Kort, and D. Trystram. List scheduling of general task graphs under LogP. *Parallel Computing*, 26:1109–1128, 2000.
99. R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computation*, pages 85–104. Plenum Press, 1972.
100. R. M. Karp and R. E. Miller. Properties of a model for parallel computations: determinacy, termination, queueing. *SIAM Journal of Applied Mathematics*, 14(6):1390–1411, November 1966.
101. R. M. Karp, R. E. Miller, and S. Winograd. The organization of computations for uniform recurrence equations. *Journal of the ACM*, 14(3):563–590, July 1967.
102. H. Kasahara and S. Narita. Practical multiprocessor scheduling algorithms for efficient parallel processing. *IEEE Transactions on Computers*, C-33:1023–1029, November 1984.
103. A. A. Khan, C. L. McCreary, and M. S. Jones. A comparison of multiprocessor scheduling heuristics. In *Proceedings of International Conference on Parallel Processing*, Volume 2, pages 243–250, August 1994.
104. S. J. Kim and J. C. Browne. A general approach to mapping of parallel computation upon multiprocessor architectures. In *International Conference on Parallel Processing*, Volume 3, pages 1–8, 1988.
105. B. Kruatrachue. *Static Task Scheduling and Grain Packing in Parallel Processing Systems*. PhD thesis, Oregon State University, USA, 1987.
106. B. Kruatrachue and T. G. Lewis. Grain size determination for parallel processing. *IEEE Software*, 5(1):23–32, January 1988.

107. W. Kubiak. Exact and approximate algorithms for scheduling unit time tasks with tree-like precedence constraints. In *Abstracts EURO IX-TIMS XXVIII*, Paris, France, 1988.
108. V. Kumar, A. Grama, A. Gupta, and G. Karypis. *Introduction to Parallel Computing—Design and Analysis of Algorithms*. Benjamin/Cummings, 1994.
109. S. Y. Kung. *VLSI Array Processors*. Information and System Sciences Series. Prentice Hall, 1988.
110. Y.-K. Kwok and I. Ahmad. Efficient scheduling of arbitrary task graphs to multiprocessors using a parallel genetic algorithm. *Journal of Parallel and Distributed Computing*, 47(1):58–77, November 1997.
111. Y.-K. Kwok and I. Ahmad. Benchmarking the task graph scheduling algorithms. In *Proceedings of International Parallel Processing Symposium/Symposium on Parallel and Distributed Processing (IPPS/SPDP-98)*, pages 531–537, Orlando, Florida, USA, April 1998.
112. Y.-K. Kwok and I. Ahmad. A comparison of parallel search-based algorithms for multiprocessors scheduling. In *Proceedings of the 2nd European Conference on Parallel and Distributed Systems (EURO-PDS'98)*, Vienna, Austria, July 1998.
113. Y.-K. Kwok and I. Ahmad. Static scheduling algorithms for allocating directed task graphs to multiprocessors. *ACM Computing Surveys*, 31(4):406–471, December 1999.
114. Y.-K. Kwok and I. Ahmad. Link contention-constrained scheduling and mapping of tasks and messages to a network of heterogeneous processors. *Cluster Computing*, 3(2):113–124, 2000.
115. Y.-K. Kwok and I. Ahmad. On multiprocessor task scheduling using efficient state space approaches. *Journal of Parallel and Distributed Computing*, 65:1515–1532, 2005.
116. L. Lamport. Time, clocks, and the ordering of events in a distributed system. *Communications of the ACM*, 21(7):558–565, July 1978.
117. C. Y. Lee, J. J. Hwang, Y. C. Chow, and F. D. Anger. Multiprocessor scheduling with interprocessor communication delays. *Operations Research Letters*, 7(3):141–147, 1988.
118. F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*. Morgan Kaufmann Publishers, 1992.
119. J. K. Lenstra, A. H. G. Rinnooy Kan, and P. Brucker. Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 1:343–362, 1977.
120. J. K. Lenstra, M. Veldhorst, and B. Veltman. The complexity of scheduling trees with communication delays. *Journal of Algorithms*, 20(1):157–173, 1996.
121. J. Y.-T. Leung, editor. *Handbook of Scheduling*. Chapman and Hall/CRC, 2004.
122. T. G. Lewis. *Foundations of Parallel Programming, A Machine-Independent Approach*. IEEE Press, 1993.
123. T. G. Lewis and H. El-Rewini. *Introduction to Parallel Computing*. Prentice Hall, 1992.
124. T. G. Lewis and H. El-Rewini. Parallax: a tool for parallel program scheduling. *IEEE Parallel and Distributed Technology: Systems and Applications*, 1(2):62–72, May 1993.
125. J.-C. Liou and M. A. Palis. A new heuristic for scheduling parallel programs on multiprocessor. In *1998 International Conference on Parallel Architectures and Compilation Techniques*, pages 358–365, October 1998.

126. G. Q. Liu, K. L. Poh, and M. Xie. Iterative list scheduling for heterogenous computing. *Journal of Parallel and Distributed Computing*, 65(5):654–665, May 2005.
127. J. W. S. Liu. *Real-Time Systems*. Prentice Hall, 2000.
128. Z. Liu. A note on Graham's bound. *Information Processing Letters*, 36:1–5, October 1990.
129. V. M. Lo. Temporal communication graphs: Lamport's process-time graphs augmented for the purpose of mapping and scheduling. *Journal of Parallel and Distributed Computing*, 16(4): 378–384, December 1992.
130. V. M. Lo, S. Rajopadhye, S. Gupta, D. Keldsen, M. A. Mohamed, B. Nitzberg, J. A. Telle, and X. Zhong. OREGAMI: tools for mapping parallel computations to parallel architectures. *International Journal of Parallel Programming*, 20(3):237–270, June 1991.
131. W. Löwe, W. Zimmermann, and J. Eisenbiegler. On linear schedules of task graphs for generalized LogP-machines. In *Euro-Par '97*, Volume 1300 of *Lecture Notes in Computer Science*, pages 895–904. Springer, 1997.
132. B. S. Macey and A. Y. Zomaya. A performance evaluation of CP list scheduling heuristics for communication intensive task graphs. In *Parallel Processing Symposium, 1998. Proceedings of IPPS/SPDP 1998*, pages 538–541, 1998.
133. K. F. Man, K. S. Tang, and S. Kwong. *Genetic Algorithms: Concepts and Designs*. Springer Verlag, 1999.
134. S. Manoharan. Effect of task duplication on the assignment of dependency graphs. *Parallel Computing*, 27(3):257–268, February 2001.
135. C. L. McCreary and H. Gill. Automatic determination of grain size for efficient parallel processing. *Communications of the ACM*, 32(9):1073–1078, September 1989.
136. C. L. McCreary, A. A. Khan, J. J. Thompson, and M. E. McArdle. A comparison of heuristics for scheduling DAGs on multiprocessors. In *Eighth International Parallel Processing Symposium, 1994*, pages 446–451, April 1994.
137. D. A. Menascé, D. Saha, S. C. S. S. Porto, V. A. F. Almeida, and S. K. Tripathi. Static and dynamic processor scheduling disciplines in heterogeneous parallel architectures. *Journal of Parallel and Distributed Computing*, 28(1):1–18, July 1995.
138. Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard*, June 1995. <http://www.mpi-forum.org/docs/docs.html>.
139. H. Oh and S. Ha. A static scheduling heuristic for heterogeneous processors. In *Proceedings of Europar'96*, Volume 1124 of *Lecture Notes in Computer Science*. Springer-Verlag, 1996.
140. M. A. Palis, J.-C. Liou, and D. S. L. Wei. Task clustering and scheduling for distributed memory parallel architectures. *IEEE Transactions on Parallel and Distributed Systems*, 7(1):46–55, January 1996.
141. C. H. Papadimitriou and M. Yannakakis. Scheduling interval ordered tasks. *SIAM Journal of Computing*, 8:405–409, 1979.
142. C. H. Papadimitriou and M. Yannakakis. Towards an architecture-independent analysis of parallel algorithms. *SIAM Journal of Computing*, 19(2):322–328, April 1990.
143. B. Parhami. *Introduction to Parallel Processing: Algorithms and Architectures*. Plenum Press, 1999.
144. K. K. Parhi. Algorithm transformation techniques for concurrent processors. *Proceedings of the IEEE*, 77(12):1879–1895, December 1989.
145. K. K. Parhi. *VLSI Digital Signal Processing*. Wiley, 1999.

146. K. K. Parhi and D. G. Messerschmitt. Static rate-optimal scheduling of iterative data-flow programs via optimum unfolding. *IEEE Transactions on Computers*, 40(2):178–195, December 1991.
147. D. A. Patterson. A case for NOW (networks of workstations). In *Proceedings of the 14th Annual ACM Symposium on Principles of Distributed Computing (PODC '95)*, pages 17–19, New York, August 1995. ACM.
148. P. M. Petersen and D. A. Padua. Static and dynamic evaluation of data dependence analysis techniques. *IEEE Transactions on Parallel and Distributed Systems*, 7(11):1121–1132, November 1996.
149. C. Picouleau. *Two New NP-Complete Scheduling Problems with Communication Delays and Unlimited Number of Processors*. Technical Report 91-94, IBP, Université Pierre et Marie Curie, France, April 1991.
150. C. Picouleau. New complexity results on scheduling with small communication delays. *Discrete Applied Mathematics*, 60(1–3):331–342, 1995.
151. M. Pinedo. *Scheduling: Theory, Algorithms, and Systems*. Prentice Hall, 2002.
152. C. D. Polychronopoulos. *Parallel Programming and Compilers*. Kluwer Academic Publishers, 1988.
153. C. D. Polychronopoulos, M. Girkar, M. Reza Haghighat, C.-L. Lee, B. Leung, and D. Schouten. Parafrase-2: a new generation parallelizing compiler. *International Journal of High Speed Computing*, 1(1):45–72, May 1989.
154. J. Protić, M. Tomašević, and V. Milutinović. Distributed shared memory: concepts and systems. *IEEE Transactions on Parallel and Distributed Technology*, pages 63–79, 1996.
155. W. Pugh. The Omega test: a fast and practical integer programming algorithm for dependence analysis. *Communications of the ACM*, 8:102–114, August 1992.
156. M. J. Quinn. *Parallel Programming in C with MPI and OpenMP*. McGraw-Hill, 2004.
157. A. Radulescu and A. J. C. van Gemund. Low-cost task scheduling for distributed-memory machines. *IEEE Transactions on Parallel and Distributed Systems*, 13(6):648–658, 2002.
158. V. J. Rayward-Smith. UET scheduling with unit interprocessor communication delays. *Discrete Applied Mathematics*, 18:55–71, 1987.
159. V. J. Rayward-Smith, F. W. Burton, and G. J. Janacek. Scheduling parallel programs assuming preallocation. In P. Chrétienne, E. G. Coffman, J. K. Lenstra, and Z. Liu, editors, *Scheduling Theory and Its Applications*, pages 145–165. Wiley, 1995.
160. P. Rebreyend, F. E. Sandnes, and G. M. Megson. *Static Multiprocessor Task Graph Scheduling in the Genetic Paradigm: A Comparison of Genotype Representations*. Research Report 98-25, Ecole Normale Supérieure de Lyon, Laboratoire de Informatique du Parallelisme, Lyon, France, 1998.
161. C. R. Reeves and J. E. Rowe. *Genetic Algorithms: Principles and Perspectives: A Guide to GA Theory*. Kluwer Academic Publishers, 2003.
162. R. Reiter. Scheduling parallel computations. *Journal of the ACM*, 15(4):590–599, October 1968.
163. P. Sadayappan, F. Ercal, and J. Ramanujam. Cluster partitioning approaches to mapping parallel programs onto a hypercube. *Parallel Computing*, 13:1–16, 1990.
164. F. E. Sandnes and G. M. Megson. Improved static multiprocessor scheduling using cyclic task graphs: a genetic approach. *Proceedings of the International Conference on Parallel*

- Computing: Fundamentals, Applications and New Directions* (Parco'97), pages 703–710, Bonn, Germany, 1997.
165. F. E. Sandnes and G. M. Megson. An evolutionary approach to static taskgraph scheduling with task duplication for minimised interprocessor traffic. In *Proceedings of the International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT 2001)*, pages 101–108, Taipei, Taiwan, July 2001. Tamkang University Press.
 166. F. E. Sandnes and O. Sinnen. A new strategy for multiprocessor scheduling of cyclic task graphs. *International Journal of High Performance Computing and Networking*, 3(1):62–71, 2005.
 167. V. Sarkar. *Partitionning and Scheduling Parallel Programs for Execution on Multiprocessors*. MIT Press, 1989.
 168. R. Sethi. Scheduling graphs on two processors. *SIAM Journal of Computing*, 5(1):73–82, 1976.
 169. G. C. Sih and E. A. Lee. A compile-time scheduling heuristic for interconnection-constrained heterogeneous processor architectures. *IEEE Transactions on Parallel and Distributed Systems*, 4(2):175–186, February 1993.
 170. H. Singh and A. Youssef. Mapping and scheduling heterogeneous task graphs using genetic algorithms. In *Proceedings of the Heterogeneous Computing Workshop (HCW'96)*, pages 86–97, Honolulu, HI, April 1996. IEEE Computer Society.
 171. O. Sinnen. *Experimental Evaluation of Task Scheduling Accuracy*. Tese de Mestrado (Master's thesis), Instituto Superior Técnico, Technical University of Lisbon, Portugal, December 2001.
 172. O. Sinnen. *Accurate Task Scheduling for Parallel Systems*. PhD thesis, Instituto Superior Técnico, Technical University of Lisbon, Portugal, April 2003.
 173. O. Sinnen and L. Sousa. *A Classification of Graph Theoretic Models for Parallel Computing*. Technical Report RT/005/99, INESC-ID, Instituto Superior Técnico, Technical University of Lisbon, Portugal, May 1999.
 174. O. Sinnen and L. Sousa. A comparative analysis of graph models to develop parallelising tools. In *Proceedings of 8th IASTED International Conference on Applied Informatics (AI 2000)*, pages 832–838, Innsbruck, Austria, February 2000.
 175. O. Sinnen and L. Sousa. A platform independent parallelising tool based on graph theoretic models. In *Vector and Parallel Processing—VECPAR 2000, Selected Papers*, Volume 1981 of *Lecture Notes in Computer Science*, pages 154–167. Springer-Verlag, 2001.
 176. O. Sinnen and L. Sousa. Experimental evaluation of task scheduling accuracy: implications for the scheduling model. *IEICE Transactions on Information and Systems*, E86-D(9):1620–1627, September 2003.
 177. O. Sinnen and L. Sousa. List scheduling: extension for contention awareness and evaluation of node priorities for heterogeneous cluster architectures. *Parallel Computing*, 30(1):81–101, January 2004.
 178. O. Sinnen and L. Sousa. On task scheduling accuracy: evaluation methodology and results. *The Journal of Supercomputing*, 27(2):177–194, February 2004.
 179. O. Sinnen and L. Sousa. Communication contention in task scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 16(6):503–515, June 2005.
 180. O. Sinnen, L. Sousa, and F. E. Sandnes. Toward a realistic task scheduling model. *IEEE Transactions on Parallel and Distributed Systems*, 17(3):263–275, 2006.

181. T. Sterling, D. Savarese, D. J. Becker, J. E. Dorband, U. A. Ranawake, and C. V. Packer. BEOWULF: a parallel workstation for scientific computation. In *International Conference on Parallel Processing, Volume 1: Architecture*, pages 11–14. Boca Raton, USA, August 1995. CRC Press.
182. H. S. Stone. Multiprocessor scheduling with the aid of network flow algorithms. *IEEE Transactions on Software Engineering*, SE-3(1):85–93, January 1977.
183. A. Tam and C. L. Wang. Contention-aware communication schedule for high speed communication. *Cluster Computing*, 6(4):339–353, 2003.
184. S. Telford. *BOBCAT User Guide*. Edinburgh Parallel Computing Centre, Scotland, UK, May 2000. <http://www.epcc.ed.ac.uk/sun/documents/introdoc.html>.
185. S. Tongsima, E. H.-M. Sha, and N. L. Passos. Communication-sensitive loop scheduling for DSP applications. *IEEE Transactions on Signal Processing*, 45(5):1309–1322, May 1997.
186. The 500 most powerful computer systems. Web site, TOP500 Supercomputer Sites, <http://www.top500.org/>.
187. H. Topcuoglu, S. Hariri, and M.-Y. Wu. Task scheduling algorithms for heterogenous machines. In *Proceedings of Heterogeneous Computing Workshop*, pages 3–14, 1999.
188. H. Topcuoglu, S. Hariri, and M.-Y. Wu. Performance-effective and low complexity task scheduling for heterogeneous computing. *IEEE Transactions on Parallel and Distributed Systems*, 13(3):260–274, 2002.
189. R. A. Towle. *Control and Data Dependence for Program Transformations*. PhD thesis, University of Illinois, Urbana-Champaign, Department of Computer Science, March 1976.
190. E. V. Trichina and J. Oinonen. Parallel program design in visual environment. In *IEEE International Conference on High Performance Computing*, pages 198–203. Bangalore, India, December 1997.
191. T. Tsuchiya, T. Osada, and T. Kikuno. Genetic-based multiprocessor scheduling using task duplication. *Microprocessors and Microsystems*, 22:197–207, 1998.
192. J. D. Ullman. NP-complete scheduling problems. *Journal of Computing System Science*, 10:384–393, 1975.
193. A. J. van der Steen and J. J. Dongarra. *Overview of Recent Supercomputers*. Technical Report, TOP500 Supercomputer Sites, <http://www.top500.org/ORSC/> 1996.
194. T. A. Varvarigou, V. P. Roychowdhury, and T. Kailath. Scheduling in and out forests in the presence of communication delays. In *Proceedings of 7th International Parallel Processing Symposium*, pages 222–229, 1993.
195. B. Veltman. *Multiprocessor Scheduling with Communication Delays*. PhD thesis, CWI, Amsterdam, The Netherlands, 1993.
196. B. Veltman, B. J. Lageweg, and J. K. Lenstra. Multiprocessor scheduling with communication delays. *Parallel Computing*, 16(2–3):173–182, 1990.
197. T. von Eicken, D. E. Culler, S. C. Goldstein, and K. E. Schauer. Active Messages: a mechanism for integrated communication and computation. In *Proceedings of the 19th Annual International Symposium on Computer Architecture*, pages 256–266. Gold Coast, Australia, May 1992.
198. L. Wang, H. J. Siegel, V. P. Roychowdhury, and A. A. Maciejewski. Task matching and scheduling in heterogeneous computing environments using a genetic-algorithm-based approach. *Journal of Parallel and Distributed Computing*, 47:8–22, November 1997.

199. J. White III and S. Bova. Where's the overlap? An analysis of popular MPI implementations. In *Proceedings of MPIDC 1999*, 1999.
200. B. Wilkinson and C. M. Allen. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*, 2nd edition, Prentice Hall, 2005.
201. G. Wirtz. Developing parallel programs in a graph-based environment. In D. Trystram, editor, *Proceedings of Parallel Computing 93, Grenoble, France*, pages 345–352. Amsterdam, September 1993. Elsevier Science Publishing North Holland.
202. M. Wolfe. *Optimizing Supercompilers for Supercomputers*. MIT Press, 1989.
203. M. Wolfe. Data dependence and program restructuring. *The Journal of Supercomputing*, 4(4):321–344, January 1991.
204. M. Wolfe. *High Performance Compilers for Parallel Computing*. Addison-Wesley, 1996.
205. S.-H. Woo, S.-B. Yang, S.-D. Kim, and T.-D. Han. Task scheduling in distributed computing systems with a genetic algorithm. In *High Performance Computing on the Information Superhighway, 1997. HPC Asia '97*, pages 301–305, April 1997.
206. A. S. Wu, H. Yu, S. Jin, K. Lin, and G. Schiavone. An incremental genetic algorithm approach to multiprocessor scheduling. *IEEE Transactions on Parallel and Distributed Systems*, 15(9):824–834, September 2004.
207. M. Y. Wu and D. D. Gajski. Hypertool: a programming aid for message-passing systems. *IEEE Transactions on Parallel and Distributed Systems*, 1(3):330–343, July 1990.
208. T. Yang and C. Fu. Heuristic algorithms for scheduling iterative task computations on distributed memory machines. *IEEE Transactions on Parallel and Distributed Systems*, 8(6):608–622, June 1997.
209. T. Yang and A. Gerasoulis. PYRROS: static scheduling and code generation for message passing multiprocessors. In *Proceedings of 6th ACM International Conference on Supercomputing*, pages 428–437, Washington, DC, August 1992.
210. T. Yang and A. Gerasoulis. List scheduling with and without communication delays. *Parallel Computing*, 19(12):1321–1344, 1993.
211. T. Yang and A. Gerasoulis. DSC: scheduling parallel tasks on an unbounded number of processors. *IEEE Transactions on Parallel and Distributed Systems*, 5(9):951–967, September 1994.
212. A. Yazici and T. Terzioglu. A comparison of data dependence analysis tests. In M. Valero, E. Onate, M. Jane, J. L. Larriba, and B. Suarez, editors, *Parallel Computing and Transputer Applications*, pages 575–583. IOS Press, Amsterdam, 1992.
213. A. Y. Zomaya, C. Ward, and B. S. Macey. Genetic scheduling for parallel processor systems: comparative studies and performance issues. *IEEE Transactions on Parallel and Distributed Systems*, 10(8):795–812, August 1999.