References

[1] E. H. L. Aarts and J. Korst. *Simulated Annealing and Boltzmann Machines*. John Wiley & Sons Ltd., 1989.

[2] J. Abello, A. L. Buchsbaum, and J. R. Westbrook. A functional approach to external graph algorithms. *Algorithmica*, 32(3):437–458, 2002.

[3] I. Abraham, D. Delling, A. V. Goldberg, and R. F. F. Werneck. A hub-based labeling algorithm for shortest paths in road networks. In *10th Symposium on Experimental Algorithms (SEA)*, volume 6630 of *LNCS*, pages 230–241. Springer, 2011.

[4] W. Ackermann. Zum hilbertschen Aufbau der reellen Zahlen. *Mathematische Annalen*, 99:118–133, 1928.

[5] G. M. Adel'son-Vel'skii and E. M. Landis. An algorithm for the organization of information. *Soviet Mathematics Doklady*, 3:1259–1263, 1962.

[6] A. Aggarwal and J. S. Vitter. The input/output complexity of sorting and related problems. *Communications of the ACM*, 31(9):1116–1127, 1988.

[7] A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley, 1974.

[8] A. V. Aho, B. W. Kernighan, and P. J. Weinberger. *The AWK Programming Language*. Addison-Wesley, 1988.

[9] R. K. Ahuja, R. L. Magnanti, and J. B. Orlin. *Network Flows*. Prentice Hall, 1993.

[10] R. K. Ahuja, K. Mehlhorn, J. B. Orlin, and R. E. Tarjan. Faster algorithms for the shortest path problem. *Journal of the ACM*, 3(2):213–223, 1990.

[11] M. Ajtai, J. Komlós, and E. Szemerédi. An $O(n \log n)$ sorting network. In *15th ACM Symposium on Theory of Computing (STOC)*, pages 1–9, 1983.

[12] Y. Akhremtsev and P. Sanders. Fast parallel operations on search trees. In *23rd IEEE Conference on High Performance Computing (HIPC)*, pages 291–300, 2016.

[13] M. Akra and L. Bazzi. On the solution of linear recurrence equations. *Computational Optimization and Applications*, 10(2):195–210, 1998.

[14] E. Alba and M. Tomassini. Parallelism and evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 6(5):443–462, 2002.

[15] N. Alon, M. Dietzfelbinger, P. B. Miltersen, E. Petrank, and E. Tardos. Linear hash functions. *Journal of the ACM*, 46(5):667–683, 1999.

[16] A. Andersson, T. Hagerup, S. Nilsson, and R. Raman. Sorting in linear time? *Journal of Computer and System Sciences*, pages 74–93, 1998.

[17] F. Annexstein, M. Baumslag, and A. Rosenberg. Group action graphs and parallel architectures. *SIAM Journal on Computing*, 19(3):544–569, 1990.

[18] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2007.

[19] L. Arge, M. T. Goodrich, M. Nelson, and N. Sitchinava. Fundamental parallel algorithms for private-cache chip multiprocessors. In *20th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 197–206, 2008.

[20] N. S. Arora, R. D. Blumofe, and C. G. Plaxton. Thread scheduling for multiprogrammed multiprocessors. In *10th ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 119–129, 1998.

[21] J. Arz, D. Luxen, and P. Sanders. Transit node routing reconsidered. In *12th Symposium on Experimental Algorithms (SEA)*, volume 7933 of *LNCS*, pages 55–66. Springer, 2013.

[22] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 1999.

[23] M. Axtmann, T. Bingmann, P. Sanders, and C. Schulz. Practical massively parallel sorting. In *27th ACM Symposium on Parallelism in Algorithms and Architectures, (SPAA)*, pages 13–23, 2015.

[24] M. Axtmann, A. Wiebigke, and P. Sanders. Lightweight MPI communicators with applications to perfectly balanced quicksort. In *32nd IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 254–265, 2018.

[25] M. Axtmann, S. Witt, D. Ferizovic, and P. Sanders. In-place parallel super scalar samplesort (IPSSSSo). In *25th European Symposium on Algorithms (ESA)*, pages 9:1–9:14, 2017. full paper at arXiv:1705.02257 [cs.DC].

[26] D. A. Bader and G. Cong. Fast shared-memory algorithms for computing the minimum spanning forest of sparse graphs. *Journal of Parallel and Distributed Computing*, 66:1366–1378, 2006.

[27] M. Bader. *Space-filling Curves – An Introduction with Applications in Scientific Computing*, volume 9 of *Texts in Computational Science and Engineering*. Springer, 2012.

[28] T. Balyo and P. Sanders. HordeSat: A massively parallel portfolio SAT solver. In *18th Conference on Theory and Applications of Satisfiability Testing (SAT)*, volume 9340 of *LNCS*, pages 156–172. Springer, 2015.

[29] H. Bast. Scheduling at twilight the easy way. In *19th Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 2285 of *LNCS*, pages 166–178. Springer, 2002.

[30] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck. Route planning in transportation networks. In

L. Kliemann and P. Sanders, editors, *Algorithm Engineering*, volume 9220 of *LNCS*, pages 19–80. Springer, 2016.

[31] H. Bast, S. Funke, P. Sanders, and D. Schultes. Fast routing in road networks with transit nodes. *Science*, 316(5824):566, 2007.

[32] H. Bast and T. Hagerup. Fast parallel space allocation, estimation and integer sorting. *Information and Computation*, 123(1):72–110, 1995.

[33] K. E. Batcher. Sorting networks and their applications. In *AFIPS Spring Joint Computing Conference*, pages 307–314, 1968.

[34] R. Bayer and E. M. McCreight. Organization and maintenance of large ordered indexes. *Acta Informatica*, 1(3):173–189, 1972.

[35] A. Beckmann, R. Dementiev, and J. Singler. Building a parallel pipelined external memory algorithm library. In *23rd IEEE International Symposium on Parallel and Distributed Processing (IPDPS)*, 2009.

[36] R. Beier and B. Vöcking. Random knapsack in expected polynomial time. *Journal of Computer and System Sciences*, 69(3):306–329, 2004.

[37] D. Belazzougui, F. C. Botelho, and M. Dietzfelbinger. Hash, displace, and compress. In *17th European Symposium on Algorithms (ESA)*, volume 5757 of *LNCS*, pages 682–693. Springer, 2009.

[38] R. Bellman. On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90, 1958.

[39] M. A. Bender, E. D. Demaine, and M. Farach-Colton. Cache-oblivious B-trees. In *41st IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 399–409, 2000.

[40] M. A. Bender, M. Farach-Colton, G. Pemmasani, S. Skiena, and P. Sumazin. Lowest common ancestors in trees and directed acyclic graphs. *Journal of Algorithms*, 57(2):75–94, 2005.

[41] J. L. Bentley and M. D. McIlroy. Engineering a sort function. *Software Practice and Experience*, 23(11):1249–1265, 1993.

[42] J. L. Bentley and T. A. Ottmann. Algorithms for reporting and counting geometric intersections. *IEEE Transactions on Computers*, C-28(9):643–647, 1979.

[43] J. L. Bentley and R. Sedgewick. Fast algorithms for sorting and searching strings. In *8th ACM Symposium on Discrete Algorithms (SODA)*, pages 360–369, 1997.

[44] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[45] T. Bingmann, T. Keh, and P. Sanders. A bulk-parallel priority queue in external memory with STXXL. In *14th Symposium on Experimental Algorithms (SEA)*, volume 9125 of *LNCS*, pages 28–40. Springer, 2015.

[46] G. E. Blelloch, D. Ferizovic, and Y. Sun. Just join for parallel ordered sets. In *28th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 253–264, 2016.

[47] G. E. Blelloch, C. E. Leiserson, B. M. Maggs, C. G. Plaxton, S. J. Smith, and M. Zagha. A comparison of sorting algorithms for the connection machine

CM-2. In *3rd ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 3–16, 1991.

[48] M. Blum, R. W. Floyd, V. R. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7(4):448, 1972.

[49] N. Blum and K. Mehlhorn. On the average number of rebalancing operations in weight-balanced trees. *Theoretical Computer Science*, 11:303–320, 1980.

[50] Boost.org. Boost C++ Libraries. `www.boost.org`.

[51] O. Borůvka. O jistém problému minimálním. *Práce Moravské Přírodovědecké Společnosti*, 3:37–58, 1926. In Czech.

[52] F. C. Botelho, R. Pagh, and N. Ziviani. Practical perfect hashing in nearly optimal space. *Information Systems*, 38(1):108–131, 2013.

[53] G. S. Brodal. Worst-case efficient priority queues. In *7th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 52–58, 1996.

[54] G. S. Brodal and J. Katajainen. Worst-case efficient external-memory priority queues. In *6th Scandinavian Workshop on Algorithm Theory (SWAT)*, volume 1432 of *LNCS*, pages 107–118. Springer, 1998.

[55] G. S. Brodal, J. L. Träff, and C. D. Zaroliagis. A parallel priority queue with constant time operations. *Journal of Parallel and Distributed Computing*, 49(1):4–21, 1998.

[56] N. G. Bronson, J. Casper, H. Chafi, and K. Olukotun. A practical concurrent binary search tree. *ACM SIGPLAN Notices*, 45(5):257–268, 2010.

[57] M. R. Brown and R. E. Tarjan. Design and analysis of a data structure for representing sorted lists. *SIAM Journal of Computing*, 9:594–614, 1980.

[58] R. Brown. Calendar queues: A fast $O(1)$ priority queue implementation for the simulation event set problem. *Communications of the ACM*, 31(10):1220–1227, 1988.

[59] A. Buluc, H. Meyerhenke, I. Safro, P. Sanders, and C. Schulz. Recent advances in graph partitioning. In L. Kliemann and P. Sanders, editors, *Algorithm Engineering*, volume 9220 of *LNCS*, pages 117–158. Springer, 2014.

[60] C. K. Caldwell and Y. Cheng. Determining Mills' constant and a note on Honaker's problem. *Journal Integer Sequences*, 8(4):Article 05.4.1, 2005.

[61] J. L. Carter and M. N. Wegman. Universal classes of hash functions. *Journal of Computer and System Sciences*, 18(2):143–154, 1979.

[62] S. K. Cha, S. Hwang, K. Kim, and K. Kwon. Cache-conscious concurrency control of main-memory indexes on shared-memory multiprocessor systems. In *27th Conference on Very Large Data Bases (VLDB)*, pages 181–190, 2001.

[63] D. Chakrabarti, Y. Zhan, and C. Faloutsos. R-MAT: A recursive model for graph mining. In *SIAM International Conference on Data Mining (SDM)*, pages 442–446, 2004.

[64] R. Chandra. *Parallel Programming in OpenMP*. Morgan Kaufmann, 2001.

[65] Chazelle. A minimum spanning tree algorithm with inverse-Ackermann type complexity. *Journal of the ACM*, 47:1028–1047, 2000.

[66] B. Chazelle and L. J. Guibas. Fractional cascading: I. A data structuring technique. *Algorithmica*, 1(2):133–162, 1986.

[67] B. Chazelle and L. J. Guibas. Fractional cascading: II. Applications. *Algorithmica*, 1(2):163–191, 1986.

[68] J.-C. Chen. Proportion extend sort. *SIAM Journal on Computing*, 31(1):323–330, 2001.

[69] Y. Cheng. Explicit estimate on primes between consecutive cubes. *Rocky Mountain J. Math.*, 40:1, 117–153, 2010.

[70] J. Cheriyan and K. Mehlhorn. Algorithms for dense graphs and networks. *Algorithmica*, 15(6):521–549, 1996.

[71] B. V. Cherkassky, A. V. Goldberg, and T. Radzik. Shortest path algorithms: Theory and experimental evaluation. *Mathematical Programming*, 73:129–174, 1996.

[72] Y.-J. Chiang, M. T. Goodrich, E. F. Grove, R. Tamassia, D. E. Vengroff, and J. S. Vitter. External-memory graph algorithms. In *6th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 139–149, 1995.

[73] E. G. Coffman, M. R. G. Jr., , and D. S. Johnson. Approximation algorithms for bin packing: A survey. In D. Hochbaum, editor, *Approximation Algorithms for NP-Hard Problems*, pages 46–93. PWS, 1997.

[74] D. Cohen-Or, D. Levin, and O. Remez. Progressive compression of arbitrary triangular meshes. In *IEEE Conference on Visualization (VIS)*, pages 67–72, 1999.

[75] R. Cole. Parallel merge sort. *SIAM Journal on Computing*, 17(4):770–785, 1988.

[76] R. Cole, P. N. Klein, and R. E. Tarjan. Finding minimum spanning forests in logarithmic time and linear work using random sampling. In *8th ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 243–250, 1996.

[77] R. Cole, K. Ost, and S. Schirra. Edge-coloring bipartite multigraphs in $O(E \log D)$ time. *Combinatorica*, 21(1):5–12, 2000.

[78] S. A. Cook. *On the Minimum Computation Time of Functions*. PhD thesis, Harvard University, 1966.

[79] S. A. Cook. The complexity of theorem proving procedures. In *3rd ACM Symposium on Theory of Computing*, pages 151–158, 1971.

[80] A. Crauser, K. Mehlhorn, U. Meyer, and P. Sanders. A parallelization of Dijkstra's shortest path algorithm. In *23rd Symposium on Mathematical Foundations of Computer Science (MFCS)*, number 1450 in LNCS, pages 722–731. Springer, 1998.

[81] G. B. Dantzig. Maximization of a linear function of variables subject to linear inequalities. In T. C. Koopmans, editor, *Activity Analysis of Production and Allocation*, pages 339–347. Wiley, 1951.

[82] C. Darwin. *The Origin of Species by Means of Natural Selection: or, the Preservation of Favored Races in the Struggle for Life*. John Murray, London, 1859.

[83] A. Davidson, D. Tarjan, M. Garland, and J. D. Owens. Efficient parallel merge sort for fixed and variable length keys. In *Innovative Parallel Computing (InPar), 2012*, pages 1–9. IEEE, 2012.

[84] A. De, P. P. Kurur, C. Saha, and R. Saptharishi. Fast integer multiplication using modular arithmetic. *SIAM Journal on Computing*, 42(2):685–699, 2013.

[85] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry Algorithms and Applications*. Springer, 2nd edition, 2000.

[86] J. Dean and S. Ghemawat. MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, January 2008.

[87] R. Dementiev, L. Kettner, J. Mehnert, and P. Sanders. Engineering a sorted list data structure for 32 bit keys. In *6th Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 142–151, 2004.

[88] R. Dementiev, L. Kettner, and P. Sanders. STXXL: standard template library for XXL data sets. *Softw., Pract. Exper.*, 38(6):589–637, 2008.

[89] R. Dementiev and P. Sanders. Asynchronous parallel disk sorting. In *15th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 138–148, 2003.

[90] R. Dementiev, P. Sanders, D. Schultes, and J. Sibeyn. Engineering an external memory minimum spanning tree algorithm. In *Exploring New Frontiers of Theoretical Informatics*, volume 155 of *IFIPAICT*, pages 195–208. Springer, 2004.

[91] L. Devroye. A note on the height of binary search trees. *Journal of the ACM*, 33(3):289–498, 1986.

[92] R. B. Dial. Shortest-path forest with topological ordering. *Communications of the ACM*, 12(11):632–633, 1969.

[93] M. Dietzfelbinger, T. Hagerup, J. Katajainen, and M. Penttonen. A reliable randomized algorithm for the closest-pair problem. *Journal of Algorithms*, 25(1):19–51, 1997.

[94] M. Dietzfelbinger, A. Karlin, K. Mehlhorn, F. Meyer auf der Heide, H. Rohnert, and R. E. Tarjan. Dynamic perfect hashing: Upper and lower bounds. *SIAM Journal of Computing*, 23(4):738–761, 1994.

[95] M. Dietzfelbinger and C. Weidling. Balanced allocation and dictionaries with tightly packed constant size bins. *Theoretical Computer Science*, 380(1–2):47–68, 2007.

[96] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[97] E. A. Dinic. Economical algorithms for finding shortest paths in a network. *Transportation Modeling Systems*, pages 36–44, 1978. In Russian.

[98] W. Domschke and A. Drexl. *Einführung in Operations Research*. Springer, 2007.

[99] J. R. Driscoll, N. Sarnak, D. D. Sleator, and R. E. Tarjan. Making data structures persistent. *Journal of Computer and System Sciences*, 38(1):86–124, 1989.

[100] M. Drmota and W. Szpankowski. A master theorem for discrete divide and conquer recurrences. *Journal of the ACM*, 60(3):16:1–16:49, 2013.

[101] S. Edelkamp and A. Weiss. BlockQuicksort: Avoiding branch mispredictions in quicksort. In *24th European Symposium on Algorithms (ESA)*, volume 57 of *LIPIcs*, pages 38:1–38:16, 2016.

[102] A. Elmasry, J. Katajainen, and M. Stenmark. Branch mispredictions don't affect mergesort. In *11th Symposium on Experimental Algorithms (SEA)*, volume 7276 of *LNCS*, pages 160–171. Springer, 2012.

[103] J. Fakcharoenphol and S. Rao. Planar graphs, negative weight edges, shortest paths, and near linear time. *Journal of Computer and System Sciences*, 72(5):868–889, 2006.

[104] L. K. Fleischer, B. Hendrickson, and A. Pınar. On identifying strongly connected components in parallel. In *Workshop on Solving Irregularly Structured Problems in Parallel (IPDPS Workshops)*, number 1800 in LNCS, pages 505–511. Springer, 2000.

[105] R. Fleischer. A tight lower bound for the worst case of Bottom-Up-Heapsort. *Algorithmica*, 11(2):104–115, 1994.

[106] R. Floyd. Assigning meaning to programs. In J. Schwarz, editor, *Mathematical Aspects of Computer Science*, pages 19–32. AMS, 1967.

[107] R. W. Floyd and R. L. Rivest. Expected time bounds for selection. *Communications of the ACM*, 18(3):165–172, 1975.

[108] L. R. Ford. Network flow theory. Technical Report P-923, Rand Corporation, Santa Monica, California, 1956.

[109] D. Fotakis, R. Pagh, P. Sanders, and P. Spirakis. Space efficient hash tables with worst case constant access time. *Theory of Computing Systems*, 38(2):229–248, 2005.

[110] W. D. Frazer and A. C. McKellar. Samplesort: A sampling approach to minimal storage tree sorting. *Journal of the ACM*, 17(3):496–507, 1970.

[111] E. Fredkin. Trie memory. *Communications of the ACM*, 3(9):490–499, 1960.

[112] M. L. Fredman. On the efficiency of pairing heaps and related data structures. *Journal of the ACM*, 46(4):473–501, 1999.

[113] M. L. Fredman, J. Komlós, and E. Szemerédi. Storing a sparse table with $O(1)$ worst case access time. *Journal of the ACM*, 31(3):538–544, 1984.

[114] M. L. Fredman, R. Sedgewick, D. D. Sleator, and R. E. Tarjan. The pairing heap: A new form of self-adjusting heap. *Algorithmica*, 1:111–129, 1986.

[115] M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Journal of the ACM*, 34(3):596–615, 1987.

[116] M. Frigo, C. E. Leiserson, H. Prokop, and S. Ramachandran. Cache-oblivious algorithms. In *40th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 285–298, 1999.

[117] M. Fürer. Faster integer multiplication. *SIAM Journal on Computing*, 39(3):979–1005, 2009.

[118] H. N. Gabow. Path-based depth-first search for strong and biconnected components. *Information Processing Letters*, 74(3–4):107–114, 2000.

[119] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.

[120] D. Gangal and A. Ranade. Precedence constrained scheduling in $(2-7/(3p-1))$ optimal. *Journal of Computer and System Sciences*, 74(7):1139–1146, 2008.

[121] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[122] B. Gärtner and J. Matoušek. *Understanding and Using Linear Programming*. Springer, 2006.

[123] R. Geisberger, P. Sanders, D. Schultes, and C. Vetter. Exact routing in large road networks using contraction hierarchies. *Transportation Science*, 46(3):388–404, 2012.

[124] J. Giacomoni, T. Moseley, and M. Vachharajani. Fastforward for efficient pipeline parallelism: A cache-optimized concurrent lock-free queue. In *13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP)*, pages 43–52, 2008.

[125] P. B. Gibbons. A more practical pram model. In *1st ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 158–168, 1989.

[126] P. B. Gibbons, Y. Matias, and V. Ramachandran. The QRQW PRAM: Accounting for contention in parallel algorithms. In *5th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 638–648, 1994.

[127] GMP (GNU Multiple Precision Arithmetic Library). `http://gmplib.org/`.

[128] A. V. Goldberg. Scaling algorithms for the shortest path problem. *SIAM Journal on Computing*, 24(3):494–504, 1995.

[129] A. V. Goldberg. A simple shortest path algorithm with linear average time. In *9th European Symposium on Algorithms (ESA)*, number 2161 in LNCS, pages 230–241. Springer, 2001.

[130] A. V. Goldberg and C. Harrelson. Computing the shortest path: A^* meets graph theory. In *16th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 156–165, 2005.

[131] G. H. Gonnet and R. Baeza-Yates. *Handbook of Algorithms and Data Structures: In Pascal and C*. Addison-Wesley, 2nd edition, 1991.

[132] M. T. Goodrich. Randomized Shellsort: A simple data-oblivious sorting algorithm. *Journal of the ACM*, 58(6):27:1–27:26, 2011.

[133] M. T. Goodrich. Zig-zag sort: A simple deterministic data-oblivious sorting algorithm running in $O(n \log n)$ time. In *46th ACM Symposium on Theory of Computing (STOC)*, pages 684–693, 2014.

[134] G. Graefe. A survey of B-tree locking techniques. *ACM Transactions on Database Systems (TODS)*, 35(3):16, 2010.

[135] G. Graefe and P.-A. Larson. B-tree indexes and CPU caches. In *17th International Conference on Data Engineering (ICDE)*, pages 349–358. IEEE, 2001.

[136] R. L. Graham. Bounds on multiprocessing timing anomalies. *SIAM Journal of Applied Mathematics*, 17(2):416–429, 1969.

[137] R. L. Graham, D. E. Knuth, and O. Patashnik. *Concrete mathematics*. Addison-Wesley, 2nd edition, 1994.

[138] J. F. Grantham and C. Pomerance. Prime numbers. In K. H. Rosen, editor, *Handbook of Discrete and Combinatorial Mathematics*, chapter 4.4, pages 236–254. CRC Press, 2000.

[139] R. Grossi and G. Italiano. Efficient techniques for maintaining multi-dimensional keys in linked data structures. In *26th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1644 of *LNCS*, pages 372–381. Springer, 1999.

[140] B. Haeupler, S. Sen, and R. E. Tarjan. Rank-pairing heaps. In *17th European Symposium on Algorithms (ESA)*, volume 5757 of *LNCS*, pages 659–670, 2009.

[141] S. Halperin and U. Zwick. Optimal randomized EREW PRAM algorithms for finding spanning forests and for other basic graph connectivity problems. In *7th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 438–447, 1996.

[142] Y. Han and M. Thorup. Integer sorting in $O\left(n\sqrt{\log\log n}\right)$ expected time and linear space. In *42nd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 135–144, 2002.

[143] G. Handler and I. Zang. A dual algorithm for the constrained shortest path problem. *Networks*, 10(4):293–309, 1980.

[144] T. D. Hansen, H. Kaplan, R. E. Tarjan, and U. Zwick. Hollow heaps. *ACM Transactions on Algorithms (TALG)*, 13(3):42:1–42:27, 2017.

[145] J. Hartmanis and J. Simon. On the power of multiplication in random access machines. In *5th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 13–23, 1974.

[146] M. Held and R. Karp. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18(6):1138–1162, 1970.

[147] M. Held and R. Karp. The traveling-salesman problem and minimum spanning trees, part II. *Mathematical Programming*, 1:6–25, 1971.

[148] J. L. Hennessy and D. A. Patterson. *Computer Architecture a Quantitative Approach*. Morgan Kaufmann, 5th edition, 2011.

[149] P. V. Hentenryck and L. Michel. *Constraint-Based Local Search*. MIT Press, 2005.

[150] M. Herlihy and N. Shavit. *The Art of Multiprocessor Programming, Revised Reprint*. Elsevier, 2012.

[151] C. A. R. Hoare. An axiomatic basis for computer programming. *Communications of the ACM*, 12(10):576–585, 1969.

[152] C. A. R. Hoare. Proof of correctness of data representations. *Acta Informatica*, 1(4):271–281, 1972.

[153] R. D. Hofstadter. Metamagical themas. *Scientific American*, 248(2):16–22, 1983.

[154] S. Hong, N. C. Rodia, and K. Olukotun. On fast parallel detection of strongly connected components (SCC) in small-world graphs. In *ACM Conference on High Performance Computing, Networking, Storage and Analysis (SC)*, pages 92:1–92:11, 2013.

[155] J. E. Hopcroft and J. D. Ullman. Set merging algorithms. *SIAM Journal on Computing*, 2(4):294–303, 1973.

[156] P. Høyer. A general technique for implementation of efficient priority queues. In *3rd Israeli Symposium on Theory of Computing and Systems*, pages 57–66, 1995.

[157] L. Hübschle-Schneider and P. Sanders. Communication efficient algorithms for top-k selection problems. In *30th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 659–668, 2016.

[158] S. Huddlestone and K. Mehlhorn. A new data structure for representing sorted lists. *Acta Informatica*, 17(2):157–184, 1982.

[159] J. Iacono. Improved upper bounds for pairing heaps. In *7th Scandinavian Workshop on Algorithm Theory (SWAT)*, volume 1851 of *LNCS*, pages 32–45. Springer, 2000.

[160] ©Intel. *Intel® 64 and IA-32 Architectures Optimization Reference Manual*, 2016. Order No. 248966-032.

[161] ISO/IEC. C++ programming language standard. Technical Report 14882:2014, ISO/IEC, 2014.

[162] A. Itai, A. G. Konheim, and M. Rodeh. A sparse table implementation of priority queues. In *8th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 115 of *LNCS*, pages 417–431. Springer, 1981.

[163] J. Jájá. *An Introduction to Parallel Algorithms*. Addison Wesley, 1992.

[164] V. Jarník. O jistém problému minimálním (Z dopisu panu O. Borůvkovi). *Práce Moravské Přírodovědecké Společnosti*, 6:57–63, 1930. In Czech.

[165] K. Jensen and N. Wirth. *Pascal User Manual and Report. ISO Pascal Standard*. Springer, 1991.

[166] jgrapht.org. JGraphT Java Graph Library. `http://jgrapht.org`.

[167] T. Jiang, M. Li, and P. Vitányi. Average-case complexity of Shellsort. In *26th International Colloquium on Automata, Languages and Programming (ICALP)*, number 1644 in LNCS, pages 453–462. Springer, 1999.

[168] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon. Optimization by simulated annealing: Experimental evaluation; part II, graph coloring and number partitioning. *Operations Research*, 39(3):378–406, 1991.

[169] N. L. Johnson, S. Kotz, and A. W. Kemp. *Univariate Discrete Distributions*. Wiley, 1989.

[170] S. L. Johnsson and C. T. Ho. Optimum broadcasting and personalized communication in hypercubes. *IEEE Transactions on Computers*, 38(9):1249–1268, 1989.

[171] L. V. Kalé and A. B. Sinha. Information sharing mechanisms in parallel programs. In *8th International Parallel Processing Symposium (IPPS)*, pages 461–468. IEEE, 1994.

[172] K. Kaligosi and P. Sanders. How branch mispredictions affect quicksort. In *14th European Symposium on Algorithms (ESA)*, volume 4168 of *LNCS*, pages 780–791. Springer, 2006.

[173] H. Kaplan and R. E. Tarjan. New heap data structures. Technical Report TR-597-99, Princeton University, 1999.

[174] A. Karatsuba and Y. Ofman. Multiplication of multidigit numbers on automata. *Soviet Physics Doklady*, 7(7):595–596, 1963.

[175] D. Karger, P. N. Klein, and R. E. Tarjan. A randomized linear-time algorithm for finding minimum spanning trees. *Journal of the ACM*, 42(1):321–329, 1995.

[176] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[177] J. Katajainen and B. B. Mortensen. Experiences with the design and implementation of space-efficient deques. In *5th Workshop on Algorithm Engineering (WAE)*, volume 2141 of *LNCS*, pages 39–50. Springer, 2001.

[178] I. Katriel, P. Sanders, and J. L. Träff. A practical minimum spanning tree algorithm using the cycle property. In *11th European Symposium on Algorithms (ESA)*, number 2832 in LNCS, pages 679–690. Springer, 2003.

[179] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack problems*. Springer, 2004.

[180] L. Khachiyan. A polynomial time algorithm in linear programming (in Russian). *Soviet Mathematics Doklady*, 20(1):191–194, 1979.

[181] T. Kieritz, D. Luxen, P. Sanders, and C. Vetter. Distributed time-dependent contraction hierarchies. In *9th Symposium on Experimental Algorithms (SEA)*, volume 6049 of *LNCS*, pages 83–93. Springer, 2010.

[182] V. King. A simpler minimum spanning tree verification algorithm. *Algorithmica*, 18(2):263–270, 1997.

[183] S. Knopp, P. Sanders, D. Schultes, F. Schulz, and D. Wagner. Computing many-to-many shortest paths using highway hierarchies. In *9th Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 36–45, 2007.

[184] D. E. Knuth. *The Art of Computer Programming—Sorting and Searching*, volume 3. Addison Wesley, 2nd edition, 1998.

[185] D. E. Knuth. *MMIXware: A RISC Computer for the Third Millennium*, volume 1750 of *LNCS*. Springer, 1999.

[186] P. Konecny. Introducing the Cray XMT. In *Cray User Group meeting (CUG)*, 2007.

[187] D. König. Über Graphen und ihre Anwendung auf Determinantentheorie und Mengenlehre. *Mathematische Annalen*, 77(4):453–465, 1916.

[188] R. E. Korf. Depth-first iterative-deepening: An optimal admissible tree search. *Artificial Intelligence*, 27(1):97–109, 1985.

[189] B. Korte and J.Vygen. *Combinatorial Optimization: Theory and Algorithms*. Springer, 2000.

[190] J. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):48–50, 1956.

[191] V. Kumar, A. Grama, A. Gupta, and G. Karypis. *Introduction to Parallel Computing. Design and Analysis of Algorithms*. Benjamin/Cummings, 1994.

[192] A. LaMarca and R. E. Ladner. The influence of caches on the performance of heaps. *ACM Journal of Experimental Algorithmics (JEA)*, 1:4:1–4:32, 1996.

[193] E. L. Lawler, J. K. Lenstra, A. H. G. Rinooy Kan, and D. B. Shmoys. *The Traveling Salesman Problem*. Wiley, 1985.

[194] LEDA (Library of Efficient Data Types and Algorithms). www.algorithmic-solutions.com.

[195] L. Q. Lee, A. Lumsdaine, and J. G. Siek. *The Boost graph library: user guide and reference manual*. Addison-Wesley, 2002.

[196] P. L. Lehman and S. B. Yao. Efficient locking for concurrent operations on b-trees. *ACM Transactions on Database Systems (TODS)*, 6(4):650–670, 1981.

[197] T. Leighton. *Introduction to Parallel Algorithms and Architectures*. Morgan Kaufmann, 1992.

[198] N. Leischner, V. Osipov, and P. Sanders. GPU sample sort. In *24th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2010. see also arXiv:0909.5649.

[199] C. E. Leiserson. Fat-trees: universal networks for hardware-efficient supercomputing. *IEEE Transactions on Computers*, 100(10):892–901, 1985.

[200] LEMON. LEMON C++ Graph Library. http://lemon.cs.elte.hu.

[201] G. Lev, N. Pippenger, and L. Valiant. A fast parallel algorithm for routing in permutation networks. *IEEE Transactions on Computing*, 30(2):93–100, 1981.

[202] L. Levin. Universal search problems (in Russian). *Problemy Peredachi Informatsii*, 9(3):265–266, 1973.

[203] I. Lustig and J.-F. Puget. Program does not equal program: Constraint programming and its relationship to mathematical programming. *Interfaces*, 31(3):29–53, 2001.

[204] T. Maier and P. Sanders. Dynamic Space Efficient Hashing. In *25th European Symposium on Algorithms (ESA)*, volume 87 of *LIPIcs*, pages 58:1–58:14, 2017.

[205] T. Maier, P. Sanders, and R. Dementiev. Concurrent hash tables: Fast *and* general?(!). *CoRR*, arXiv:1601.04017 [cs.DS], 2016. short version in PPoPP 2016.

[206] S. Martello and P. Toth. *Knapsack Problems – Algorithms and Computer Implementations*. Wiley, 1990.

[207] C. Martínez and S. Roura. Optimal sampling strategies in Quicksort and Quickselect. *SIAM Journal on Computing*, 31(3):683–705, 2002.

[208] F. Mattern. Algorithms for distributed termination detection. *Distributed Computing*, 2(3):161–175, 1987.

[209] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, and J. Ramirez-Alfonsin, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 195–247. Springer, 1998.

[210] C. McGeoch, P. Sanders, R. Fleischer, P. R. Cohen, and D. Precup. Using finite experiments to study asymptotic performance. In *Experimental Algorithmics — From Algorithm Design to Robust and Efficient Software*, volume 2547 of *LNCS*, pages 93–126. Springer, 2002.

[211] MCSTL: The Multi-Core Standard Template Library. http://algo2.iti.uni-karlsruhe.de/singler/mcstl/.

[212] K. Mehlhorn. A faster approximation algorithm for the Steiner problem in graphs. *Information Processing Letters*, 27(3):125–128, Mar. 1988.

[213] K. Mehlhorn. Amortisierte Analyse. In T. Ottmann, editor, *Prinzipien des Algorithmenentwurfs*, pages 91–102. Spektrum Lehrbuch, 1998.

[214] K. Mehlhorn and U. Meyer. External-memory breadth-first search with sublinear I/O. In *10th European Symposium on Algorithms (ESA)*, volume 2461 of *LNCS*, pages 723–735. Springer, 2002.

[215] K. Mehlhorn and S. Näher. Bounded ordered dictionaries in $O(\log\log N)$ time and $O(n)$ space. *Information Processing Letters*, 35(4):183–189, 1990.

[216] K. Mehlhorn and S. Näher. Dynamic fractional cascading. *Algorithmica*, 5(2):215–241, 1990.

[217] K. Mehlhorn and S. Näher. *The LEDA Platform for Combinatorial and Geometric Computing*. Cambridge University Press, 1999.

[218] K. Mehlhorn, S. Näher, and P. Sanders. Engineering DFS-based graph algorithms. *arXiv preprint arXiv:1703.10023*, 2017.

[219] K. Mehlhorn, V. Priebe, G. Schäfer, and N. Sivadasan. All-pairs shortest-paths computation in the presence of negative cycles. *Information Processing Letters*, 81(6):341–343, 2002.

[220] K. Mehlhorn and P. Sanders. Scanning multiple sequences via cache memory. *Algorithmica*, 35(1):75–93, 2003.

[221] K. Mehlhorn and S. Saxena. A still simpler way of introducing the interior-point method for linear programming. *Computer Science Review*, 22:1–11, 2016.

[222] K. Mehlhorn and M. Ziegelmann. Resource constrained shortest paths. In *8th European Symposium on Algorithms (ESA)*, volume 1879 of *LNCS*, pages 326–337. Springer, 2000.

[223] R. Mendelson, R. E. Tarjan, M. Thorup, and U. Zwick. Melding priority queues. In *9th Scandinavian Workshop on Algorithm Theory (SWAT)*, volume 3111 of *LNCS*, pages 223–235. Springer, 2004.

[224] *Meyers Konversationslexikon*. Bibliographisches Institut, 1888.

[225] B. Meyer. *Object-Oriented Software Construction*. Prentice-Hall, second edition, 1997.

[226] U. Meyer. Average-case complexity of single-source shortest-path algorithms: lower and upper bounds. *Journal of Algorithms*, 48(1):91–134, 2003. preliminary version in SODA 2001.

[227] U. Meyer and P. Sanders. Δ-stepping: A parallelizable shortest path algorithm. *Journal of Algorithms*, 49(1):114–152, 2003.

[228] U. Meyer, P. Sanders, and J. Sibeyn, editors. *Algorithms for Memory Hierarchies*, volume 2625 of *LNCS Tutorial*. Springer, 2003.

[229] G. L. Miller and J. H. Reif. Parallel tree contraction and its application. In *26st IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 478–489, 1985.

[230] B. M. E. Moret and H. D. Shapiro. An empirical analysis of algorithms for constructing a minimum spanning tree. In *2nd Workshop on Algorithms and Data Structures (WADS)*, volume 519 of *LNCS*, pages 400–411. Springer, 1991.

[231] R. Morris. Scatter storage techniques. *Communications of the ACM*, 11(1):38–44, 1968.

[232] S. S. Muchnick. *Advanced Compiler Design and Implementation*. Morgan Kaufmann, 1997.

[233] I. Müller, P. Sanders, A. Lacurie, W. Lehner, and F. Färber. Cache-efficient aggregation: Hashing is sorting. In *ACM SIGMOD Conference on Management of Data*, pages 1123–1136, 2015.

[234] I. Müller, P. Sanders, R. Schulze, and W. Zhou. Retrieval and perfect hashing using fingerprinting. In *13th Symposium on Experimental Algorithms (SEA)*, volume 8504 of *LNCS*, pages 138–149. Springer, 2014.

[235] S. Näher and O. Zlotowski. Design and implementation of efficient data types for static graphs. In *10th European Symposium on Algorithms (ESA)*, volume 2461 of *LNCS*, pages 748–759. Springer, 2002.

[236] M. Naor and O. Reingold. On the construction of pseudorandom permutations: Luby-Rackoff revisited. *Journal of Cryptology*, 12(1):29–66, 1999.

[237] G. Navarro. *Compact Data Structures – A Practical Approach*. Cambridge University Press, 2016.

[238] G. Nemhauser and Z. Ullmann. Discrete dynamic programming and capital allocation. *Management Science*, 15(9):494–505, 1969.

[239] G. Nemhauser and L. Wolsey. *Integer and Combinatorial Optimization*. Wiley, 1988.

[240] J. Nešetřil, H. Milková, and H. Nešetřilová. Otakar Borůvka on minimum spanning tree problem: Translation of both the 1926 papers, comments, history. *Discrete Mathematics*, 233:3–36, 2001.

[241] K. S. Neubert. The Flashsort1 algorithm. *Dr. Dobb's Journal*, pages 123–125, February 1998.

[242] J. v. Neumann. First draft of a report on the EDVAC. Technical report, University of Pennsylvania, 1945.

[243] J. Nievergelt and E. Reingold. Binary search trees of bounded balance. *SIAM Journal on Computing*, 2(1):33–43, 1973.

[244] K. Noshita. A theorem on the expected complexity of Dijkstra's shortest path algorithm. *Journal of Algorithms*, 6(3):400–408, 1985.

[245] V. Osipov, P. Sanders, and J. Singler. The filter-Kruskal minimum spanning tree algorithm. In *10th Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 52–61, 2009.

[246] A. Pagh, R. Pagh, and M. Ružić. Linear probing with constant independence. *SIAM Journal on Computing*, 39(3):1107–1120, 2009.

[247] R. Pagh and F. Rodler. Cuckoo hashing. *Journal of Algorithms*, 51(2):122–144, 2004.

[248] M. Patrascu and M. Thorup. On the k-independence required by linear probing and minwise independence. In *37th International Colloquium on Automata, Languages and Programming (ICALP, Part I)*, volume 6198 of *LNCS*, pages 715–726, 2010.

[249] W. J. Paul, P. Bach, M. Bosch, J. Fischer, C. Lichtenau, and J. Röhrig. Real PRAM programming. In *8th Euro-Par*, volume 2400 of *LNCS*, pages 522–531. Springer, 2002.

[250] H.-O. Peitgen and P. H. Richter. *The Beauty of Fractals*. Springer-Verlag, 1986.

[251] W. W. Peterson. Addressing for random access storage. *IBM Journal of Research and Development*, 1(2), Apr. 1957.

[252] S. Pettie. Towards a final analysis of pairing heaps. In *46th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 174–183, 2005.

[253] S. Pettie and V. Ramachandran. An optimal minimum spanning tree algorithm. In *27th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1853 of *LNCS*, pages 49–60. Springer, 2000.

[254] J. Pinkerton. *Voyages and Travels*, volume 2. 1808.

[255] P. J. Plauger, A. A. Stepanov, M. Lee, and D. R. Musser. *The C++ Standard Template Library*. Prentice-Hall, 2000.

[256] R. C. Prim. Shortest connection networks and some generalizations. *Bell Systems Technical Journal*, 36(6):1389–1401, 1957.

[257] W. Pugh. Skip lists: A probabilistic alternative to balanced trees. *Communications of the ACM*, 33(6):668–676, 1990.

[258] M. Pătraşcu and M. Thorup. The power of simple tabulation hashing. *Journal of the ACM*, 59(3):14:1–14:50, 2012.

[259] M. Rahn, P. Sanders, and J. Singler. Scalable distributed-memory external sorting. In *26th IEEE International Conference on Data Engineering (ICDE)*, pages 685–688, 2010.

[260] S. Rajasekaran and J. H. Reif. Optimal and sublogarithmic time randomized parallel sorting algorithms. *SIAM Journal on Computing*, 18(3):594–607, 1989.

[261] O. Ramaré and Y. Saouter. Short effective intervals containing primes. *Journal on Number Theory*, 98(1):10–33, 2003.

[262] A. Ranade, S. Kothari, and R. Udupa. Register efficient mergesorting. In *7th Conference on High Performance Computing (HIPC)*, volume 1970 of *LNCS*, pages 96–103. Springer, 2000.

[263] J. H. Reif. Depth-first search is inherently sequential. *Information Processing Letters*, 20(5):229–234, 1985.

[264] N. Robertson, D. P. Sanders, P. Seymour, and R. Thomas. Efficiently four-coloring planar graphs. In *28th ACM Symposium on Theory of Computing (STOC)*, pages 571–575. ACM Press, 1996.

[265] G. Robins and A. Zelikwosky. Improved Steiner tree approximation in graphs. In *11th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 770–779, 2000.

[266] S. Roura. Improved master theorems for divide-and-conquer recurrences. *Journal of the ACM*, 48(2):170–205, 2001.

[267] P. Sanders. *Lastverteilungsalgorithmen für parallele Tiefensuche*. PhD thesis, University of Karlsruhe, 1996.

[268] P. Sanders. On the competitive analysis of randomized static load balancing. In S. Rajasekaran, editor, *First Workshop on Randomized Parallel Algorithms*, Honolulu, Hawaii, 16 April, 1996. `http://algo2.iti.kit.edu/sanders/papers/rand96.pdf`.

[269] P. Sanders. Random permutations on distributed, external and hierarchical memory. *Information Processing Letters*, 67(6):305–310, 1998.

[270] P. Sanders. Randomized priority queues for fast parallel access. *Journal Parallel and Distributed Computing, Special Issue on Parallel and Distributed Data Structures*, 49(1):86–97, 1998.

[271] P. Sanders. Fast priority queues for cached memory. *ACM Journal of Experimental Algorithmics*, 5, 2000.

[272] P. Sanders. Randomized receiver initiated load balancing algorithms for tree shaped computations. *The Computer Journal*, 45(5):561–573, 2002.

[273] P. Sanders, S. Lamm, L. Hübschle-Schneider, E. Schrade, and C. Dachsbacher. Efficient random sampling – parallel, vectorized, cache-efficient, and online. *ACM Transactions on Mathematical Software*, 44(3), 2018.

[274] P. Sanders, S. Schlag, and I. Müller. Communication efficient algorithms for fundamental big data problems. In *IEEE Conference on Big Data*, pages 15–23, 2013.

[275] P. Sanders and D. Schultes. Highway hierarchies hasten exact shortest path queries. In *13th European Symposium on Algorithms (ESA)*, volume 3669 of *LNCS*, pages 568–597. Springer, 2005.

[276] P. Sanders and C. Schulz. Distributed evolutionary graph partitioning. In *14th Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 16–29. SIAM, 2012.

[277] P. Sanders, J. Speck, and J. L. Träff. Two-tree algorithms for full bandwidth broadcast, reduction and scan. *Parallel Computing*, 35(12):581–594, 2009.

[278] P. Sanders and J. Wassenberg. Engineering a multi-core radix sort. In *17th Euro-Par*, volume 6853 of *LNCS*, pages 160–169. Springer, 2011.

[279] P. Sanders and S. Winkel. Super scalar sample sort. In *12th European Symposium on Algorithms (ESA)*, volume 3221 of *LNCS*, pages 784–796. Springer, 2004.

[280] P. Sanders and T. Worsch. *Parallele Programmierung mit MPI – ein Praktikum*. Logos Verlag Berlin, 1997. ISBN 3-931216-76-4.

[281] R. Santos and F. Seidel. A better upper bound on the number of triangulations of a planar point set. *Journal of Combinatorial Theory Series A*, 102(1):186–193, 2003.

[282] N. Satish, M. Harris, and M. Garland. Designing efficient sorting algorithms for manycore GPUs. In *23rd IEEE International Symposium on Parallel and Distributed Processing (IPDPS)*, 2009.

[283] R. Schaffer and R. Sedgewick. The analysis of heapsort. *Journal of Algorithms*, 15(1):76–100, 1993.

[284] A. Schönhage. Storage modification machines. *SIAM Journal on Computing*, 9(3):490–508, 1980.

[285] A. Schönhage and V. Strassen. Schnelle Multiplikation großer Zahlen. *Computing*, 7(3–4):281–292, 1971.
[286] A. Schrijver. *Combinatorial Optimization (3 Volumes)*. Springer Verlag, 2003.
[287] R. Sedgewick. Analysis of Shellsort and related algorithms. In *4th European Symposium on Algorithms (ESA)*, volume 1136 of *LNCS*, pages 1–11. Springer, 1996.
[288] R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms*. Addison-Wesley, 1996.
[289] R. Seidel and C. Aragon. Randomized search trees. *Algorithmica*, 16(4–5):464–497, 1996.
[290] R. Seidel and M. Sharir. Top-down analysis of path compression. *SIAM Journal of Computing*, 34(3):515–525, 2005.
[291] J. Sewall, J. Chhugani, C. Kim, N. Satish, and P. Dubey. PALM: Parallel Architecture-Friendly Latch-Free Modifications to B+ Trees on Many-Core Processors. *PVLDB*, 4(11):795–806, 2011.
[292] M. Sharir. A strong-connectivity algorithm and its applications in data flow analysis. *Computers and Mathematics with Applications*, 7(1):67–72, 1981.
[293] J. C. Shepherdson and H. E. Sturgis. Computability of recursive functions. *Journal of the ACM*, 10(2):217–255, 1963.
[294] J. Shun and G. E. Blelloch. Phase-concurrent hash tables for determinism. In *26th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 96–107, 2014.
[295] J. Shun, G. E. Blelloch, J. T. Fineman, and P. B. Gibbons. Reducing contention through priority updates. In *25th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 152–163, 2013.
[296] J. Shun, G. E. Blelloch, J. T. Fineman, P. B. Gibbons, A. Kyrola, H. V. Simhadri, and K. Tangwongsan. The problem based benchmark suite. In *24th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 68–70, 2012. `http://www.cs.cmu.edu/~pbbs`.
[297] J. Singler, P. Sanders, and F. Putze. MCSTL: The multi-core standard template library. In *13th Euro-Par*, volume 4641 of *LNCS*, pages 682–694. Springer, 2007.
[298] M. Sipser. *Introduction to the Theory of Computation*. MIT Press, 1998.
[299] D. D. Sleator and R. E. Tarjan. A data structure for dynamic trees. *Journal of Computer and System Sciences*, 26(3):362–391, 1983.
[300] D. D. Sleator and R. E. Tarjan. Self-adjusting binary search trees. *Journal of the ACM*, 32(3):652–686, 1985.
[301] D. Spielman and S.-H. Teng. Smoothed analysis of algorithms: why the simplex algorithm usually takes polynomial time. *Journal of the ACM*, 51(3):385–463, 2004.
[302] M. Stephan and J. Docter. Jülich Supercomputing Centre. JUQUEEN: IBM Blue Gene/Q Supercomputer System at the Jülich Supercomputing Centre. *Journal of large-scale research facilities*, A1, 1 2015.

[303] A. Stivala, P. J. Stuckey, M. G. de la Banda, M. Hermenegildo, and A. Wirth. Lock-free parallel dynamic programming. *Journal of Parallel and Distributed Computing*, 70(8):839–848, 2010.

[304] G. L. Taboada, S. Ramos, R. R. Expósito, J. Touriño, and R. Doallo. Java in the high performance computing arena: Research, practice and experience. *Science of Computer Programming*, 78(5):425–444, 2013.

[305] R. E. Tarjan. Depth first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[306] R. E. Tarjan. Efficiency of a good but not linear set union algorithm. *Journal of the ACM*, 22(2):215–225, 1975.

[307] R. E. Tarjan. Shortest paths. Technical report, AT&T Bell Laboratories, 1981.

[308] R. E. Tarjan. Amortized computational complexity. *SIAM Journal on Algebraic and Discrete Methods*, 6(2):306–318, 1985.

[309] R. E. Tarjan and U. Vishkin. An efficient parallel biconnectivity algorithm. *SIAM Journal on Computing*, 14(4):862–874, 1985.

[310] M. Thorup. Undirected single source shortest paths in linear time. *Journal of the ACM*, 46(3):362–394, 1999.

[311] M. Thorup. Even strongly universal hashing is pretty fast. In *11th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 496–497, 2000.

[312] M. Thorup. Compact oracles for reachability and approximate distances in planar digraphs. *Journal of the ACM*, 51(6):993–1024, 2004.

[313] M. Thorup. Integer priority queues with decrease key in constant time and the single source shortest paths problem. *Journal of Computer and System Sciences*, 69(3):330–353, 2004.

[314] M. Thorup and U. Zwick. Approximate distance oracles. *Journal of the ACM*, 52(1):1–24, 2005.

[315] A. Toom. The complexity of a scheme of functional elements realizing the multiplication of integers. *Soviet Mathematics Doklady*, 150(3):496–498, 1963.

[316] P. Tsigas and Y. Zhang. A simple, fast parallel implementation of quicksort and its performance evaluation on SUN enterprise 10000. In *11th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP)*, pages 372–381. IEEE, 2003.

[317] Unknown. *Der Handlungsreisende – wie er sein soll und was er zu thun hat, um Auftraege zu erhalten und eines gluecklichen Erfolgs in seinen Geschaeften gewiss zu sein – Von einem alten Commis-Voyageur*. 1832.

[318] L. Valiant. A bridging model for parallel computation. *Communications of the ACM*, 33(8):103–111, 1990.

[319] P. van Emde Boas. Preserving order in a forest in less than logarithmic time. *Information Processing Letters*, 6(3):80–82, 1977.

[320] R. Vanderbei. *Linear Programming: Foundations and Extensions*. Springer, 2001.

[321] P. J. Varman, S. D. Scheufler, B. R. Iyer, and G. R. Ricard. Merging multiple lists on hierarchical-memory multiprocessors. *Journal on Parallel & Distributed Computing*, 12(2):171–177, 1991.

[322] V. Vazirani. *Approximation Algorithms*. Springer, 2000.
[323] C. Vetter. Fast and exact mobile navigation with openstreetmap data. Master's thesis, KIT, 2010.
[324] J. Vuillemin. A data structure for manipulating priority queues. *Communications of the ACM*, 21(4):309–314, 1978.
[325] L. Wall, T. Christiansen, and J. Orwant. *Programming Perl*. O'Reilly, 3rd edition, 2000.
[326] J. Wassenberg, W. Middelmann, and P. Sanders. An efficient parallel algorithm for graph-based image segmentation. In *13th Conference on Computer Analysis of Images and Patterns (CAIP)*, pages 1003–1010, 2009.
[327] I. Wegener. BOTTOM-UP-HEAPSORT, a new variant of HEAPSORT beating, on an average, QUICKSORT (if n is not very small). *Theoretical Computer Science*, 118(1):81–98, 1993.
[328] I. Wegener. *Complexity Theory: Exploring the Limits of Efficient Algorithms*. Springer, 2005.
[329] R. Wickremesinghe, L. Arge, J. S. Chase, and J. S. Vitter. Efficient sorting using registers and caches. *ACM Journal of Experimental Algorithmics*, 7(9), 2002.
[330] R. Wilhelm and D. Maurer. *Compiler Design*. Addison-Wesley, 1995.
[331] J. W. J. Williams. Algorithm 232: Heapsort. *Communications of the ACM*, 7(6):347–348, 1964.
[332] A. Yasin. A top-down method for performance analysis and counters architecture. *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, pages 35–44, 2014.
[333] W. Zhou. A practical scalable shared-memory parallel algorithm for computing minimum spanning trees. Master's thesis, Karlsruhe Institute of Technology, 2017.
[334] A. L. Zobrist. A new hashing method with application for game playing. Technical Report TR88, Computer Sciences Department, University of Wisconsin, Madison, 1970.

Index