

REFERENCES

- [1] M. Herlihy and J. E. B. Moss, "Transactional memory: Architectural support for lock-free data structures," in *Proceedings of the 20th Annual International Symposium on Computer Architecture*, ISCA '93, (New York, NY, USA), pp. 289–300, ACM, 1993.
- [2] I. Corporation, "Intel architecture instruction set extensions programming reference, chapter 8," 2013.
- [3] D. Dice, O. Shalev, and N. Shavit, "Transactional locking ii," in *Distributed Computing*, pp. 194–208, Springer, 2006.
- [4] R. Rajwar and J. R. Goodman, "Speculative lock elision: Enabling highly concurrent multithreaded execution," in *Proceedings of the 34th Annual ACM/IEEE International Symposium on Microarchitecture*, MICRO 34, (Washington, DC, USA), pp. 294–305, IEEE Computer Society, 2001.
- [5] L. Dalessandro, F. Carouge, S. White, Y. Lev, M. Moir, M. L. Scott, and M. F. Spear, "Hybrid NOrec: A case study in the effectiveness of best effort hardware transactional memory," in *Proceedings of the 16th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pp. 39–52, ACM, 2011.
- [6] T. David, R. Guerraoui, and V. Trigonakis, "Asynchronized concurrency: The secret to scaling concurrent search data structures," in *Proceedings of the Twentieth International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 631–644, ACM, 2015.
- [7] R. Guerraoui and V. Trigonakis, "Optimistic concurrency with OPTIK," in *Proceedings of the 21st ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP '16, (New York, NY, USA), pp. 18:1–18:12, ACM, 2016.
- [8] T. Brown, F. Ellen, and E. Ruppert, "Pragmatic primitives for non-blocking data structures," in *ACM Symposium on Principles of Distributed Computing*, PODC '13, Montreal, QC, Canada, July 22–24, 2013, pp. 13–22, 2013.
- [9] T. Skare and C. Kozyrakis, "Early release: Friend or foe," in *Workshop on Transactional Memory Workloads*, vol. 77, 2006.
- [10] S. K. Haider, W. Hasenplaugh, and D. Alistarh, "Lease/release: Architectural support for scaling contended data structures," in *Proceedings of the 21st ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP '16, (New York, NY, USA), pp. 17:1–17:12, ACM, 2016.
- [11] T. L. Harris, "A pragmatic implementation of non-blocking linked-lists," in *Proceedings of the International Conference on Distributed Computing (DISC)*, pp. 300–314, 2001.
- [12] M. M. Michael, "High performance dynamic lock-free hash tables and list-based sets," in *Proceedings of the fourteenth annual ACM symposium on Parallel algorithms and architectures*, pp. 73–82, ACM, 2002.
- [13] S. Heller, M. Herlihy, V. Luchangco, M. Moir, W. N. S. III, and N. Shavit, "A lazy concurrent list-based set algorithm," in *Proceedings of the 9th International Conference on Principles of Distributed Systems (OPODIS 2005)*, Revised Selected Papers (J. H. Anderson, G. Prencipe, and R. Wattenhofer, eds.), vol. 3974 of *Lecture Notes in Computer Science*, pp. 3–16, Springer, 2006.
- [14] M. Herlihy and N. Shavit, *The art of multiprocessor programming*. Morgan Kaufmann, 2008.
- [15] K. Fraser, "Practical lock-freedom," Tech. Rep. UCAM-CL-TR-579, University of Cambridge, Computer Laboratory, Feb. 2004.
- [16] M. Herlihy, Y. Lev, V. Luchangco, and N. Shavit, "A simple optimistic skiplist algorithm," in *Proceedings of the 14th international conference on Structural information and communication complexity*, SIROCCO'07, (Berlin, Heidelberg), pp. 124–138, Springer-Verlag, 2007.
- [17] D. Lea, 2007. <http://java.sun.com/javase/6/docs/api/java/util/concurrent/ConcurrentSkipListMap.html>.
- [18] M. Fomitchev and E. Ruppert, "Lock-free linked lists and skip lists," in *Proceedings of the 23rd annual ACM symposium on Principles of Distributed Computing (PODC'04)*, (New York, NY, USA), pp. 50–59, ACM Press, 2004.
- [19] F. Ellen, P. Fatourou, E. Ruppert, and F. van Breugel, "Non-blocking binary search trees," in *Proceedings of the 29th ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, PODC '10, (New York, NY, USA), pp. 131–140, ACM, 2010.
- [20] A. Natarajan and N. Mittal, "Fast concurrent lock-free binary search trees," in *ACM SIGPLAN Notices*, vol. 49, pp. 317–328, ACM, 2014.
- [21] M. M. Michael and M. L. Scott, "Simple, fast, and practical non-blocking and blocking concurrent queue algorithms," in *Proceedings of the Fifteenth Annual ACM Symposium on Principles of Distributed Computing*, PODC '96, (New York, NY, USA), pp. 267–275, ACM, 1996.
- [22] R. K. Treiber, "Systems programming: Coping with parallelism," Tech. Rep. RJ 5118, IBM Almaden Research Center, 1986.
- [23] N. Shavit and D. Touitou, "Elimination trees and the construction of pools and stacks: preliminary version," in *Proceedings of the seventh annual ACM symposium on Parallel algorithms and architectures*, pp. 54–63, ACM, 1995.
- [24] I. Lotan and N. Shavit, "Skiplist-based concurrent priority queues," in *Parallel and Distributed Processing Symposium, 2000. IPDPS 2000. Proceedings. 14th International*, pp. 263–268, IEEE, 2000.
- [25] D. Alistarh, J. Kopinsky, J. Li, and N. Shavit, "The spraylist: A scalable relaxed priority queue," in *Proceedings of the 20th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP 2015, (New York, NY, USA), pp. 11–20, ACM, 2015.
- [26] H. Rihani, P. Sanders, and R. Dementiev, "Brief announcement: Multiqueues: Simple relaxed concurrent priority queues," in *Proceedings of the 27th ACM on Symposium on Parallelism in Algorithms and Architectures*, SPAA '15, (New York, NY, USA), pp. 80–82, ACM, 2015.
- [27] R. Bayer and M. Schkolnick, "Readings in database systems," ch. Concurrency of Operations on B-trees, pp. 129–139, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.
- [28] L. Dalessandro, M. F. Spear, and M. L. Scott, "NOrec: streamlining STM by abolishing ownership records," in *Proceedings of the 15th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP 2010, Bangalore, India, January 9–14, 2010, pp. 67–78, 2010.
- [29] T. L. Harris, K. Fraser, and I. A. Pratt, "A practical multi-word compare-and-swap operation," in *Proceedings of the 16th International Conference on Distributed Computing*, DISC '02, (Berlin, Heidelberg), pp. 265–279, Springer-Verlag, 2002.
- [30] T. Brown, F. Ellen, and E. Ruppert, "A general technique for non-blocking trees," in *ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP '14, Orlando, FL, USA, February 15–19, 2014, pp. 329–342, 2014.
- [31] T. Brown, *Techniques for Constructing Efficient Lock-free Data Structures*. PhD thesis, University of Toronto, 2017.
- [32] J. E. Miller, H. Kasture, G. Kurian, C. Gruenwald III, N. Beckmann, C. Celio, J. Eastep, and A. Agarwal, "Graphite: A distributed parallel simulator for multi-cores," in *High Performance Computer Architecture (HPCA), 2010 IEEE 16th International Symposium on*, pp. 1–12, IEEE, 2010.
- [33] C. C. Minh, J. Chung, C. Kozyrakis, and K. Olukotun, "STAMP: stanford transactional applications for multi-processing," in *IISWC*, pp. 35–46, IEEE Computer Society, 2008.
- [34] O. Ostrovsky and A. Morrison, "Scaling concurrent queues by using HTM to profit from failed atomic operations," in *PPOPP '20: 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, San Diego, California, USA, February 22–26, 2020 (R. Gupta and X. Shen, eds.), pp. 89–101, ACM, 2020.
- [35] T. Nakaïke, R. Odaira, M. Gaudet, M. M. Michael, and H. Tomari, "Quantitative comparison of hardware transactional memory for blue gene/q, zenterprise ec12, intel core, and power8," in *Proceedings of the 42Nd Annual International Symposium on Computer Architecture*, ISCA '15, (New York, NY, USA), pp. 144–157, ACM, 2015.
- [36] N. Cohen, M. Herlihy, E. Petrank, and E. Wald, "The teleportation design pattern for hardware transactional memory," in *LIPICs-Leibniz International Proceedings in Informatics*, vol. 95, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.
- [37] D. J. Sorin, M. D. Hill, and D. A. Wood, *A Primer on Memory Consistency and Cache Coherence*. Morgan & Claypool Publishers, 1st ed., 2011.
- [38] R. Guerraoui and M. Kapalka, "On the correctness of transactional memory," in *Proceedings of the 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPOPP '08, (New York, NY, USA), p. 175–184, Association for Computing Machinery, 2008.