

Bibliography

- [1] Jennifer M. Anderson, Lance M. Berc, Jeffrey Dean, Sanjay Ghemawat, Monika R. Henzinger, Shun-Tak A. Leung, Richard L. Sites, Mark T. Vandevoorde, Carl A. Waldspurger, and William E. Weihl. Continuous profiling: Where have all the cycles gone? Technical Report 1997-16, Compaq Systems Research Lab., July 1997.
- [2] Vasanth Bala, Evelyn Duesterwald, and Sanjeev Banerjia. Dynamo: A transparent dynamic optimization system. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, 2000.
- [3] Thomas Ball and James R. Larus. Efficient path profiling. *Proceedings of the 29th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, December 1996.
- [4] Luiz A. Barroso, Kourosh Gharachorloo, Andreas Nowatzky, and Ben Verghese. Impact of chip-level integration on performance of oltp workloads. *Proceedings of the 6th Intl. Conference on High Performance Computer Architecture*, January 2000.
- [5] Luiz André Barroso, Kourosh Gharachorloo, and Edouard Bugnion. Memory system characterization of commercial workloads. *Proceedings of the 16th Annual Intl. Symposium on Computer Architecture*, pages 3–14, June 1998.
- [6] Bryan Black, Bohuslav Rychlik, and John Paul Shen. The block-based trace cache. *Proceedings of the 26th Annual Intl. Symposium on Computer Architecture*, May 1999.
- [7] D. Burger, T.M. Austin, and S.Bennett. Evaluating future microprocessors: the simplescalar tool set. Technical Report TR-1308, University of Winsconsin, July 1996.
- [8] Brad Calder and Dirk Grunwald. Fast & accurate instruction fetch and branch prediction. *Proceedings of the 21st Annual Intl. Symposium on Computer Architecture*, pages 2–11, 1994.
- [9] Brad Calder and Dirk Grunwald. Reducing branch costs via branch alignment. *Proceedings of the 6th Intl. Conference on Architectural Support for Programming Languages and Operating Systems*, pages 242–251, October 1994.
- [10] Brad Calder and Dirk Grunwald. Next cache line and set prediction. *Proceedings of the 22th Annual Intl. Symposium on Computer Architecture*, June 1995.
- [11] Po-Yung Chang, Marius Evers, and Yale N. Patt. Improving branch prediction accuracy by reducing pattern history table interference. *Proceedings of the Intl. Conference on Parallel Architectures and Compilation Techniques*, October 1996.

- [12] Po-Yung Chang, Eric Hao, Tse-Yu Yeh, and Yale N. Patt. Branch classification: a new mechanism for improving branch predictor performance. *Proceedings of the 27th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 22–31, 1994.
- [13] Robert Cohn, David Goodwin, P. Geoffrey Lowney, and Norman Rubin. Spike: an optimizer for alpha/nt executables. *USENIX*, pages 17–23, August 1997.
- [14] T. Conte, K. Menezes, P. Mills, and B. Patell. Optimization of instruction fetch mechanism for high issue rates. *Proceedings of the 22th Annual Intl. Symposium on Computer Architecture*, pages 333–344, June 1995.
- [15] Intel corporation. Ia64 application developer’s architecture guide. <http://developer.intel.com/design/ia64/index.htm>.
- [16] Standards Performance Evaluation Corporation. <http://www.spec.org/osg/cpu95/>.
- [17] Z. Cvetanovic and D.D. Donaldson. Performance characterization of the alpha 21264 microprocessor using tp and spec workloads. *Proceedings of the 21st Annual Intl. Symposium on Computer Architecture*, pages 60–70, April 1994.
- [18] A. N. Eden and Trevor N. Mudge. The yags branch prediction scheme. *Proceedings of the Intl. Conference on Parallel Architectures and Compilation Techniques*, pages 69–77, 1998.
- [19] Robert B. Garner et al. The scalable processor architecture (sparc). *Capcom’88*, pages 278–283, 1988.
- [20] Joseph A. Fisher. Trace scheduling: A technique for global microcode compaction. *IEEE Transactions on Computers*, 30(7):478–490, July 1981.
- [21] Joseph A. Fisher and Stefan M. Freudenberger. Predicting conditional branch directions from previous runs of a program. *Proceedings of the 5th Intl. Conference on Architectural Support for Programming Languages and Operating Systems*, pages 85–95, 1992.
- [22] Daniel Holmes Friendly, Sanjay Jeram Patel, and Yale N. Patt. Alternative fetch and issue techniques from the trace cache mechanism. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, December 1997.
- [23] Daniel Holmes Friendly, Sanjay Jeram Patel, and Yale N. Patt. Putting the fill unit to work: Dynamic optimization for trace cache microprocessors. *Proceedings of the 31st Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 173–181, November 1998.
- [24] Nikolas Gloy, Trevor Blackwell, Michael D. Smith, and Brad Calder. Procedure placement using temporal ordering information. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 303–313, December 1997.
- [25] Dirk Grunwald, Donald Lindsay, and Benjamin Zorn. Static methods in hybrid branch prediction. *Proceedings of the Intl. Conference on Parallel Architectures and Compilation Techniques*, pages 222–229, 1998.
- [26] L. Gwennap. Digital 21264 sets new standard. *Microprocessor Report*, pages 11–16, October 1996.

- [27] Linley Gwennap. Ppc 604 powers past pentium. *Microprocessor report*, 8(5):1–6, April 1994.
- [28] Amir H. Hashemi, David R. Kaeli, and Brad Calder. Efficient procedure mapping using cache line coloring. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 171–182, June 1997.
- [29] Amir H. Hashemi, David R. Kaeli, and Brad Calder. Procedure mapping using static call graph estimation. *Proc. Workshop on Interaction between Compiler and Computer Architecture*, February 1997.
- [30] Derek L. Howard and Mikko. H. Lipasti. The effect of program optimization on trace cache performance. *Proceedings of the Intl. Conference on Parallel Architectures and Compilation Techniques*, pages 256–261, October 1999.
- [31] Doug Hunt. Advanced performance features of the 64-bit pa-8000. *Capcom'95*, pages 123–128, 1995.
- [32] Wen-Mei Hwu and Pohua P. Chang. Achieving high instruction cache performance with an optimizing compiler. *Proceedings of the 16th Annual Intl. Symposium on Computer Architecture*, pages 242–251, June 1989.
- [33] Q. Jacobson and J. E. Smith. Instruction pre-processing in trace processors. *Proceedings of the 5th Intl. Conference on High Performance Computer Architecture*, pages 125–129, January 1999.
- [34] Quinn Jacobson, Eric Rotenberg, and James E. Smith. Path-based next trace prediction. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, December 1997.
- [35] Stephan Jourdan, Lihu Rappoport, Yoav Almong, Mattan Erez, Adi Yoaz, and Ronny Ronen. extended block cache. *Proceedings of the 6th Intl. Conference on High Performance Computer Architecture*, pages 61–70, January 2000.
- [36] John Kalamatianos and David R. Kaeli. Temporal-based procedure reordering for improved instruction cache performance. *Proceedings of the 4th Intl. Conference on High Performance Computer Architecture*, February 1998.
- [37] G. Kane and J. Heinrich. *MIPS RISC Architecture*. Prentice Hall, 1992.
- [38] Andreas Krall. Improving semi-static branch prediction by code replication. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 97–106, 1994.
- [39] Chih-Chieh Lee, I-Cheng K. Chen, and Trevor N. Mudge. The bi-mode branch predictor. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 4–13, December 1997.
- [40] J. Lee and A. Smith. Branch prediction strategies and branch target buffer design. *IEEE Computer*, 17(1):6–22, January 1984.

- [41] Donald Lindsay. Static methods in branch prediction. *Ph.D. thesis, Department of Computer Science, University of Colorado*, 1998.
- [42] Scott McFarling. Combining branch predictors. Technical Report TN-36, Compaq Western Research Lab., June 1993.
- [43] Scott McFarling and John Hennessy. Reducing the cost of branches. *Proceedings of the 13th Annual Intl. Symposium on Computer Architecture*, pages 396–403, 1986.
- [44] Wen mei W. Hwu, Thomas M. Conte, and Pohua P. Chang. Comparing software and hardware schemes for reducing the cost of branches. *Proceedings of the 16th Annual Intl. Symposium on Computer Architecture*, pages 224–233, June 1989.
- [45] Matthew C. Merten, Andrew R. Trick, Christopher N. George, John C. Gyllenhaal, and Wen mei Hwu. A hardware-driven profiling scheme for identifying program hot spots to support runtime optimization. *Proceedings of the 26th Annual Intl. Symposium on Computer Architecture*, pages 136–147, May 1999.
- [46] Pierre Michaud, Andre Seznec, and Richard Uhlig. Trading conflict and capacity aliasing in conditional branch predictors. *Proceedings of the 24th Annual Intl. Symposium on Computer Architecture*, pages 292–303, 1997.
- [47] Frank Mueller and David A. Whalley. Avoiding conditional branches by code replication. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 56–66, 1995.
- [48] Frank Mueller and David B. Whalley. Avoiding unconditional jumps by code replication. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 322–330, 1992.
- [49] Robert Muth. Alto: A platform for object code modification. *Ph.D. Dissertation, The University of Arizona*, August 1999.
- [50] Carlos Navarro, Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. Fetch engines and databases. *3rd Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW-3)*, January 2000.
- [51] Carlos Navarro, Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. On the performance of fetch engines running dss workloads. *Proceedings of the Intl. Euro-Par Conference*, August 2000.
- [52] Subbarao Palacharla, Norman P. Jouppi, and James E. Smith. Complexity-effective superscalar processors. *Proceedings of the 24th Annual Intl. Symposium on Computer Architecture*, June 1997.
- [53] Sanjay J. Patel, Tony Tung, Satarupa Bose, and Matthew M. Crum. Increasing the size of atomic instruction blocks using control flow assertions. *Proceedings of the 33rd Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 303–313, December 2000.

- [54] Sanjay Jeram Patel, Marius Evers, and Yale N. Patt. Improving trace cache effectiveness with branch promotion and trace packing. *Proceedings of the 25th Annual Intl. Symposium on Computer Architecture*, pages 262–271, June 1998.
- [55] Sanjay Jeram Patel, Daniel Holmes Friendly, and Yale N. Patt. Critical issues regarding the trace cache fetch mechanism. Technical Report CSE-TR-335-97, University of Michigan, May 1997.
- [56] Harish Patil and Joel Emer. Combining static and dynamic branch prediction to reduce destructive aliasing. *Proceedings of the 6th Intl. Conference on High Performance Computer Architecture*, pages 251–262, January 2000.
- [57] Jason R. C. Patterson. Accurate static branch prediction by value range propagation. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 67–78, 1995.
- [58] Alexander Peleg and Uri Weiser. Dynamic flow instruction cache memory organized around trace segments independent of virtual address line. *U.S. Patent Number 5.381.533*, January 1995.
- [59] Karl Pettis and Robert C. Hansen. Profile guided code positioning. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 16–27, June 1990.
- [60] Alex Ramirez, Luiz Barroso, Kourosh Gharachorloo, Robert Cohn, Josep L. Larriba-Pey, Geoffrey Lawney, and Mateo Valero. Code layout optimizations for transaction processing workloads. *Proceedings of the 28th Annual Intl. Symposium on Computer Architecture*, July 2001.
- [61] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. The effect of code reordering on branch prediction. *Proceedings of the Intl. Conference on Parallel Architectures and Compilation Techniques*, pages 189–198, October 2000.
- [62] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. Semi-static branch prediction for optimized code layouts. *3rd Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW-3)*, January 2000.
- [63] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. A stream processor fron-end. *IEEE TCCA Newsletter*, pages 10–13, June 2000.
- [64] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. Branch prediction using profile data. *Proceedings of the Intl. Euro-Par Conference*, August 2001.
- [65] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. Fetching instruction streams. Technical Report UPC-DAC-2001-38, Universitat Politecnica de Catalunya, November 2001.
- [66] Alex Ramirez, Josep L. Larriba-Pey, and Mateo Valero. Instruction fetch architectures and code layout optimizations. *Proceedings of the IEEE*, 89(11):1588–1609, November 2001.

- [67] Alex Ramirez, Josep Ll. Larriba-Pey, Carlos Navarro, Xavi Serrano, Josep Torrellas, , and Mateo Valero. Optimizing instruction fetch for decision support workloads. *2nd Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW-2)*, January 1999.
- [68] Alex Ramirez, Josep Ll. Larriba-Pey, Carlos Navarro, Xavi Serrano, Josep Torrellas, and Mateo Valero. Optimization of instruction fetch for decision support workloads. *Proceedings of the Intl. Conference on Parallel Processing*, pages 238–245, September 1999.
- [69] Alex Ramirez, Josep Ll. Larriba-Pey, Carlos Navarro, Josep Torrellas, and Mateo Valero. Software trace cache. *Proceedings of the 13th Intl. Conference on Supercomputing*, June 1999.
- [70] Alex Ramirez, Josep Ll. Larriba-Pey, and Mateo Valero. Trace cache redundancy: Red & blue traces. *Proceedings of the 6th Intl. Conference on High Performance Computer Architecture*, January 2000.
- [71] Glenn Reinman, Todd Austin, and Brad Calder. A scalable front-end architecture for fast instruction delivery. *Proceedings of the 26th Annual Intl. Symposium on Computer Architecture*, pages 234–245, May 1999.
- [72] Glenn Reinman, Brad Calder, and Todd Austin. Fetch directed instruction prefetching. *Proceedings of the 32nd Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 16–24, 1999.
- [73] M. Rosenblum, E. Bugnion, S. A. Herrod, and S. Devine. Using the simos machine simulator to study complex computer systems. *ACM Transactions on Modeling and Computer Simulation*, 7(1):78–103, January 1997.
- [74] E. Rotenberg, S. Bennett, and J. E. Smith. Trace cache: a low latency approach to high bandwidth instruction fetching. *Proceedings of the 29th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 24–34, December 1996.
- [75] E. Rotenberg, Q. Jacobson, Y. Sazeides, and J. E. Smith. Trace processors. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 138–148, December 1997.
- [76] Eric Rotenberg, Steve Bennett, and James E. Smith. A trace cache microarchitecture and evaluation. *IEEE Transactions on Computers, Special Issue on Cache Memory*, February 1999.
- [77] Andre Sez nec and Pierre Michaud. D-aliased hybrid branch predictors. Technical Report PI-1229, IRISA, February 1999.
- [78] R. L. Sites. *Alpha Architecture Reference Manual*. Digital Press, Burlington, MA, 1992.
- [79] J. E. Smith and G. S. Sohi. The microarchitecture of superscalar processors. *Proceedings of the IEEE*, 83(12), December 1995.

- [80] James E. Smith. A study of branch prediction strategies. *Proceedings of the 8th Annual Intl. Symposium on Computer Architecture*, pages 135–148, 1981.
- [81] S. Peter Song, Marving Denman, and Joe Chang. The powerpc 604 risc microprocessor. *IEEE Micro*, 14(5):8–17, October 1994.
- [82] Eric Sprangle, Robert S. Chappell, Mitch Alsup, and Yale N. Patt. The agree predictor: A mechanism for reducing negative branch history interference. *Proceedings of the 24th Annual Intl. Symposium on Computer Architecture*, pages 284–291, 1997.
- [83] Amitabh Srivastava and Alan Eustace. Atom - a system for building customized program analysis tools. *Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation*, pages 196–205, June 1994.
- [84] Amitabh Srivastava and David W. Wall. A practical system for intermodule code optimization at link-time. *Journal of Programming Languages*, 1(1):1–18, December 1992.
- [85] Jared Stark, Paul B. Racunas, and Yale N. Patt. Reducing the impact of icache misses by writing instructions into the reservation stations out of order. *Proceedings of the 30th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 34–43, December 1997.
- [86] Robert Stets, Luiz A. Barroso, Kourosh Gharachorloo, and Ben Verghese. A detailed comparison of tpc-c versus tpc-b. *3rd Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW-3)*, January 2000.
- [87] Josep Torrellas, Chun Xia, and Russell Daigle. Optimizing instruction cache performance for operating system intensive workloads. *Proceedings of the 1st Intl. Conference on High Performance Computer Architecture*, pages 360–369, January 1995.
- [88] Transaction Processing Performance Council (TPC). Tpc benchmark c (oltp). Standard Specification.
- [89] Transaction Processing Performance Council (TPC). Tpc benchmark b (database stress test). Standard Specification, 1990.
- [90] Transaction Processing Performance Council (TPC). Tpc benchmark d (decision support). Standard Specification, Revision 1. 2. 3, 1993–1997.
- [91] Pedro Trancoso, Josep Ll. Larriba-Pey, Zheng Zhang, and Josep Torrellas. The memory performance of dss commercial workloads in shared-memory multiprocessors. *Proceedings of the 3rd Intl. Conference on High Performance Computer Architecture*, February 1997.
- [92] T. Y. Yeh and Y. N. Patt. Two-level adaptive branch prediction. *Proceedings of the 24th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 51–61, 1991.
- [93] T.-Y. Yeh and Y. N. Patt. A comprehensive instruction fetch mechanism for a processor supporting speculative execution. *Proceedings of the 25th Annual ACM/IEEE Intl. Symposium on Microarchitecture*, pages 129–139, December 1992.

- [94] Tse-Yu. Yeh, Deborah T. Marr, and Yale N. Patt. Increasing the instruction fetch rate via multiple branch prediction and a branch address cache. *Proceedings of the 7th Intl. Conference on Supercomputing*, pages 67–76, July 1993.
- [95] Tse-Yu Yeh and Yale N. Patt. Alternative implementations of two-level adaptive branch prediction. *Proceedings of the 19th Annual Intl. Symposium on Computer Architecture*, pages 124–134, 1992.
- [96] Cliff Young and Michael D. Smith. Improving the accuracy of static branch prediction using branch correlation. *Proceedings of the 6th Intl. Conference on Architectural Support for Programming Languages and Operating Systems*, pages 232–241, October 1994.