Bibliography

Aho, A. V., Sethi, R. and Ullman, J. D. (1987). COMPILERS Principles, Techniques and Tools, Addison-Wesley Publishing Company.

Ahuja, S., Zhang, W., Lakshminarayana, A. and Shukla, S. K. (2010). A Methodology for Power Aware High-Level Synthesis of Co-processors from Software Algorithms, in Proc. of the VLSID '10, pp. 282–287, ISBN 978-0-7695-3928-7.

Ashar, P., Bhattacharya, S., Raghunathan, A. and Mukaiyama, A. (1998). Verification of RTL generated from scheduled behavior in a high-level synthesis flow, in Proc. of the 1998 IEEE/ACM international conference on Computer-aided design, ICCAD '98, pp. 517–524, ACM, ISBN 1-58113-008-2.

Bacon, D. F., Graham, S. L. and Sharp, O. J. (1994). Compiler transformations for high-performance computing, ACM Comput. Surv., vol. 26, pp. 345–420.

Barrett, C. W., Fang, Y., Goldberg, B., Hu, Y., Pnueli, A. and Zuck, L. D. (2005). TVOC: A Translation Validator for Optimizing Compilers, in CAV, pp. 291–295.

Bergamaschi, R. A., Lobo, D. and Kuehlmann, A. (1992). Control optimization in high-level synthesis using behavioral don't cares, in Proc. of the DAC, pp. 657–661, ISBN 0-89791-516-X.

Bharath, N., Nandy, S. and Bussa, N. (2005). Artificial Deadlock Detection in Process Networks for ECLIPSE, in IEEE International Conference on Application-Specific Systems, Architecture Processors 2005, pp. 22–27, IEEE Computer Society Press, ISBN 0-7695-2407-9.
http://portal.acm.org/citation.cfm?id=1097107.1097168

Borrione, D., Dushina, J. and Pierre, L. (2000). A compositional model for the functional verification of high-level synthesis results, IEEE Transactions on VLSI Systems, vol. 8, no. 5, pp. 526–530.

Bouchebaba, Y., Girodias, B., Nicolescu, G., Aboulhamid, E. M., Lavigueur, B. and Paulin, P. (2007). MPSoC memory optimization using program transformation, ACM Trans. Des. Autom. Electron. Syst., vol. 12, no. 4, pp. 43:1–43:39.

Brandolese, C., Fornaciari, W., Salice, F. and Sciuto, D. (2004). Analysis and Modeling of Energy Reducing Source Code Transformations, in Proc. of the conference on Design, automation and test in Europe, DATE '04, pp. 306–311, IEEE Computer Society, Washington, DC, USA, ISBN 0-7695-2085-5.
http://portal.acm.org/citation.cfm?id=968880.969253

Buck, J. T. (1993). Scheduling Dynamic Dataflow Graphs with Bounded Memory Using the Token Flow Model, Ph.D. thesis, University of California, EECS Dept., Berkeley, CA.

Camposano, R. (1991). Path-Based Scheduling for Synthesis, IEEE transactions on computer-Aided Design of Integrated Circuits and Systems, vol. Vol 10 No 1, pp. 85–93.

Carpenter, P. M., Ramirez, A. and Ayguade, E. (2010). Buffer sizing for self-timed stream programs on heterogeneous distributed memory multiprocessors, in 5th International Conference on High Performance Embedded Architectures and Compilers, HiPEAC 2010, pp. 96–110, Springer.

Castrillon, J., Velasquez, R., Stulova, A., Sheng, W., Ceng, J., Leupers, R., Ascheid, G. and Meyr, H. (2010). Trace-based KPN composability analysis for mapping simultaneous applications to MPSoC platforms, in Proceedings of the Conference on Design, Automation and Test in Europe, DATE '10, pp. 753–758, European Design and Automation Association, 3001 Leuven, Belgium, Belgium, ISBN 978-3-9810801-6-2.
http://portal.acm.org/citation.cfm?id=1870926.1871107

Catthoor, F., D., E. and Greff, S. S. (1998). HICSS. Custom Memory Management Methodology:Exploration of Memory Organisation for Embedded Multimedia System Design, Kluwer Academic Publishers.

Ceng, J., Castrillon, J., Sheng, W., Scharwächter, H., Leupers, R., Ascheid, G., Meyr, H., Isshiki, T. and Kunieda, H. (2008). MAPS: an integrated framework for MPSoC application parallelization, in Proceedings of the 45th annual Design Automation Conference, DAC '08, pp. 754–759, ACM, New York, NY, USA, ISBN 978-1-60558-115-6.
http://doi.acm.org/10.1145/1391469.1391663

Chandrakasan, A., Potkonjak, M., Mehra, R., Rabaey, J. and Brodersen, R. (1995a). Optimizing power using transformations, IEEE Transactions on CAD of ICS, vol. 14, no. 1, pp. 12 –31.

Chandrakasan, A., Sheng, S. and Brodersen, R. (1992). Low-power CMOS digital design, IEEE Journal of Solid-State Circuits, vol. 27, no. 4, pp. 473 –484.

Chandrakasan, A. P., Potkonjak, M., Mehra, R., Rabaey, J. and Brodersen, R. W. (1995b). Optimizing Power Using Transformations, IEEE Transactions on CAD of ICS, vol. 14, no. 1, pp. 12–31.

Chen, G., Kandemir, M. and Li, F. (2006a). Energy-aware computation duplication for improving reliability in embedded chip multiprocessors, in ASP-DAC '06: Proceedings of the 2006 conference on Asia South Pacific design automation, pp. 134–139, IEEE Press, Piscataway, NJ, USA, ISBN 0-7803-9451-8.

Chen, X., Hsieh, H. and Balarin, F. (2006b). Verification approach of metropolis design framework for embedded systems, Int. J. Parallel Program., vol. 34, pp. 3–27.
http://portal.acm.org/citation.cfm?id=1182761.1182763

Chen, X., Hsieh, H., Balarin, F. and Watanabe, Y. (2003). Formal Verification for Embedded System Designs, Design Automation for Embedded Systems, vol. 8, pp. 139–153.

Cheung, E., Chen, X., Hsieh, H., Davare, A., Sangiovanni-Vincentelli, A. and Watanabe, Y. (2009). Runtime deadlock analysis for system level design, Design Automation for Embedded Systems, vol. 13, pp. 287–310.
http://dx.doi.org/10.1007/s10617-009-9046-2

Cheung, E., Hsieh, H. and Balarin, F. (2007). Automatic buffer sizing for rate-constrained KPN applications on multiprocessor system-on-chip, in Proceedings of the 2007 IEEE International High Level Design Validation and Test Workshop, pp.

37–44, IEEE Computer Society, Washington, DC, USA, ISBN 978-1-4244-1480-2. http://portal.acm.org/citation.cfm?id=1546679.1546842

Chiang, T.-H. and Dung, L.-R. (2007). Verification method of dataflow algorithms in high-level synthesis, J. Syst. Softw., vol. 80, no. 8, pp. 1256–1270.

Cimatti, A., Clarke, E. M., Giunchiglia, F. and Roveri, M. (2000). NuSMV: A New Symbolic Model Checker, International Journal on Software Tools for Technology Transfer, vol. 2, no. 4, pp. 410–425.

Clarke, E. M., Grumberg, O. and Peled, D. A. (2002). Model Checking, The MIT Press.

Cockx, J., Denolf, K., Vanhoof, B. and Stahl, R. (2007). SPRINT: a tool to generate concurrent transaction-level models from sequential code, EURASIP J. Appl. Signal Process., vol. 2007, no. 1, pp. 1–15.

Cordone, R., Ferrandi, F., Santambrogio, M. D., Palermo, G. and Sciuto, D. (2006). Using speculative computation and parallelizing techniques to improve scheduling of control based designs, in Proceedings of the 2006 Asia and South Pacific Design Automation Conference, ASP-DAC '06, pp. 898–904, IEEE Press, Piscataway, NJ, USA, ISBN 0-7803-9451-8. http://dx.doi.org/10.1145/1118299.1118502

Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C. (2001). Introduction to Algorithms, The MIT Press, Cambridge , Massachusetts London, England.

Davare, A., Zhu, Q. and Sangiovanni-Vincentelli, A. (2006). A Platform-based Design Flow for Kahn Process Networks, Tech. Rep. 2006-30, UC Berkeley.

de Kock, E. A. (2002). Multiprocessor mapping of process networks: a JPEG decoding case study, in ISSS '02: Proceedings of the 15th international symposium on System Synthesis, pp. 68–73, ISBN 1-58113-576-9.

Dos Santos, L. C. V., Heijligers, M. J. M., Van Eijk, C. A. J., Van Eijnhoven, J. and Jess, J. A. G. (2000). A code-motion pruning technique for global scheduling, ACM Trans. Des. Autom. Electron. Syst., vol. 5, no. 1, pp. 1–38.

Dos. Santos, L. C. V. and Jress, J. (1999). A reordering technique for efficient code motion, in Procs. of the 36th ACM/IEEE Design Automation Conference, DAC '99, pp. 296–299, ACM, New York, NY, USA, ISBN 1-58113-109-7.

E. Özer, A. P. N. and Gregg, D. (2003). Classification of Compiler Optimizations for High Performance, Small Area and Low Power in FPGAs, Tech. rep., Trinity College, Dublin, Ireland, Department of Computer Science.

Eveking, H., Hinrichsen, H. and Ritter, G. (1999). Automatic verification of scheduling results in high-level synthesis, in Proceedings of the conference on Design, automation and test in Europe, DATE '99, pp. 260–265, ACM, New York, NY, USA, ISBN 1-58113-121-6.

Fehnker, A., Huuck, R., Jayet, P., Lussenburg, M. and Rauch, F. (2007). Model Checking Software at Compile Time, in Proceedings of the First Joint IEEE/IFIP Symposium on Theoretical Aspects of Software Engineering, pp. 45–56, IEEE Computer Society, Washington, DC, USA, ISBN 0-7695-2856-2.

Fehnker, A., Huuck, R., Schlich, B. and Tapp, M. (2009). Automatic Bug Detection in Microcontroller Software by Static Program Analysis, in Proceedings of the 35th Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM '09, pp. 267–278, Springer-Verlag, Berlin, Heidelberg, ISBN 978-3-540-95890-1.

Fei, Y., Ravi, S., Raghunathan, A. and Jha, N. K. (2007). Energy-optimizing source code transformations for operating system-driven embedded software, Trans. on Embedded Computing Sys., vol. 7, no. 1, pp. 1–26.

Ferrandi, F., Fossati, L., Lattuada, M., Palermo, G., Sciuto, D. and Tumeo, A. (2007). Automatic Parallelization of Sequential Specifications for Symmetric MPSoCs, IFIP International Federation for Information Processing, vol. 231, no. 2, pp. 179–192.

Fisher, J. (1981). Trace Scheduling: A Technique for Global Microcode Compaction, IEEE Transactions on Computers, vol. C-30, no. 7, pp. 478 –490.

Floyd, R. W. (1967). Assigning Meaning to programs, in Schwartz, J. T. (Ed.), Proceedings the 19^{th} Symposium on Applied Mathematics, pp. 19–32, American Mathematical Society, Providence, R.I., mathematical Aspects of Computer Science.

Fraboulet, A., Kodary, K. and Mignotte, A. (2001). Loop fusion for memory space optimization, in ISSS '01: Proceedings of the 14th international symposium on Systems synthesis, pp. 95–100, ISBN 1-58113-418-5.

Freisleben, B. and Kielmann, T. (1995). Automated Transformation of Sequential Divide-and-Conquer Algorithms into Parallel Programs, Computers and Artificial Intelligence, vol. 14, pp. 579–596.

Fujita, M. (2005). Equivalence checking between behavioral and RTL descriptions with virtual controllers and datapaths, ACM Trans. Des. Autom. Electron. Syst., vol. 10, no. 4, pp. 610–626.

Gajski, D. D., Dutt, N. D., Wu, A. C. and Lin, S. Y. (1992). High-Level Synthesis: Introduction to Chip and System Design, Kluwer Academic Publishers.

Geilen, M. and Basten, T. (2003). Requirements on the execution of Kahn process networks, in Proceedings of the 12th European conference on Programming, ESOP'03, pp. 319–334, Springer-Verlag, Berlin, Heidelberg, ISBN 3-540-00886-1. http://portal.acm.org/citation.cfm?id=1765712.1765736

Gesellensetter, L., Glesner, S. and Salecker, E. (2008). Formal verification with Isabelle/HOL in practice: finding a bug in the GCC scheduler, in Proceedings of the 12th international conference on Formal methods for industrial critical systems, FMICS'07, pp. 85–100, Springer-Verlag, Berlin, Heidelberg, ISBN 3-540-79706-8, 978-3-540-79706-7. http://portal.acm.org/citation.cfm?id=1793603.1793613

Ghodrat, M., Givargis, T. and Nicolau, A. (2008). Control flow optimization in loops using interval analysis, in Proceedings of the 2008 international conference on Compilers, architectures and synthesis for embedded systems, CASES '08, pp. 157–166, ACM, New York, NY, USA, ISBN 978-1-60558-469-0. http://doi.acm.org/10.1145/1450095.1450120

Ghodrat, M., Givargis, T. and Nicolau, A. (2009). Optimizing control flow in loops using interval and dependence analysis, Design Automation for Embedded Systems, vol. 13, pp. 193–221, 10.1007/s10617-009-9043-5. http://dx.doi.org/10.1007/s10617-009-9043-5

Girkar, M. and Polychronopoulos, C. D. (1992). Automatic Extraction of Functional Parallelism from Ordinary Programs, IEEE Trans. Parallel Distrib. Syst., vol. 3, no. 2, pp. 166–178.

Graphics, M. (2006). Catapult C Synthesis. http://www.mentor.com/products/esl/high_level_synthesis/

Gries, D. (1987). The Science of Programming, Springer-Verlag New York, Inc., Secaucus, NJ, USA, ISBN 0387964800.

Gupta, R. and Soffa, M. (1990). Region scheduling: an approach for detecting and redistributing parallelism, IEEE Transactions on Software Engineering, vol. 16, no. 4, pp. 421 –431.

Gupta, S., Dutt, N., Gupta, R. and Nicolau, A. (2003a). Dynamic Conditional Branch Balancing during the High-Level Synthesis of Control-Intensive Designs, in Proceedings of DATE'03, pp. 270–275, IEEE Computer Society, Washington, DC, USA, ISBN 0-7695-1870-2.

Gupta, S., Dutt, N., Gupta, R. and Nicolau, A. (2003b). Dynamically Increasing the Scope of Code Motions during the High-Level Synthesis of Digital Circuits, IEE Proceedings: Computer and Digital Technique, vol. 150, no. 5, pp. 330–337.

Gupta, S., Dutt, N., Gupta, R. and Nicolau, A. (2003c). SPARK: a high-level synthesis framework for applying parallelizing compiler transformations, in Proc. of Int. Conf. on VLSI Design, pp. 461–466, IEEE Computer Society, Washington, DC, USA.

Gupta, S., Dutt, N., Gupta, R. and Nicolau, A. (2004a). Loop Shifting and Compaction for the High-level Synthesis of Designs with Complex Control Flow, in Proceedings of the DATE '04, vol. 1, pp. 114–119.

Gupta, S., Dutt, N., Gupta, R. and Nicolau, A. (2004b). Using global code motions to improve the quality of results for high-level synthesis, IEEE Transactions on CAD of ICS, vol. 23, no. 2, pp. 302–312.

Gupta, S., Gupta, R., Dutt, N. and Nicolau, A. (2004c). Coordinated Parallelizing Compiler Optimizations and High-Level Synthesis, ACM Transactions on Design Automation of Electronic Systems (TODAES), vol. 9, no. 4, pp. 1–31.

Gupta, S., Miranda, M., Catthoor, F. and Gupta, R. (2000). Analysis of high-level address code transformations for programmable processors, in Proceedings of the conference on Design, automation and test in Europe, DATE '00, pp. 9–13, ACM, New York, NY, USA, ISBN 1-58113-244-1.
http://doi.acm.org/10.1145/343647.343683

Gupta, S., Reshadi, M., Savoiu, N., Dutt, N., Gupta, R. and Nicolau, A. (2002). Dynamic common sub-expression elimination during scheduling in high-level synthesis, in Proceedings of the 15th international symposium on System Synthesis, ISSS '02, pp. 261–266, ACM, New York, NY, USA, ISBN 1-58113-576-9.

Gupta, S., Savoiu, N., Dutt, N., Gupta, R. and Nicolau, A. (2001a). Conditional Speculation and its Effects on Performance and Area for High-Level Synthesis, in International Symposium on System Synthesis, pp. 171–176.

Gupta, S., Savoiu, N., Kim, S., Dutt, N., Gupta, R. and Nicolau, A. (2001b). Speculation techniques for high level synthesis of control intensive designs, in Proceedings of DAC'01, pp. 269–272.

Hall, M. W., Anderson, J. M., Amarasinghe, S. P., Murphy, B. R., Liao, S.-W., Bugnion, E. and Lam, M. S. (1996). Maximizing Multiprocessor Performance with the SUIF Compiler, Computer, vol. 29, no. 12, pp. 84–89.

Hoare, C. A. R. (1969). An axiomatic basis for computer programming, Commun. ACM, vol. 12, pp. 576–580.

Holzmann, G. J. (1997). The Model Checker SPIN, IEEE Trans. Softw. Eng., vol. 23, pp. 279–295.
http://portal.acm.org/citation.cfm?id=260897.260902

Howden, W. E. (1987). Functional program testing and analysis, McGraw-Hill, New York.

Hu, Y., Barrett, C., Goldberg, B. and Pnueli, A. (2005). Validating More Loop Optimizations, Electronic Notes in Theoretical Computer Science, vol. 141, no. 2, pp. 69–84, proceedings of the Fourth International Workshop on Compiler Optimization meets Compiler Verification (COCV 2005).

Hwu, W. M. W., Mahlke, S. A., Chen, W. Y., Chang, P. P., Warter, N. J., Bringmann, R. A., Ouellette, R. G., Hank, R. E., Kiyohara, T., Haab, G. E., Holm, J. G. and Lavery, D. M. (1993). The superblock: An effective technique for VLIW and superscalar compilation, The Journal of Supercomputing, vol. 7, pp. 229–248, 10.1007/BF01205185.
http://dx.doi.org/10.1007/BF01205185

Jain, R., Majumdar, A., Sharma, A. and Wang, H. (1991). Empirical evaluation of some high-level synthesis scheduling heuristics, in Proceedings of the 28th ACM/IEEE Design Automation Conference, DAC '91, pp. 686–689, ACM, New York, NY, USA, ISBN 0-89791-395-7.

Jiang, B., Deprettere, E. and Kienhuis, B. (2008). Hierarchical run time deadlock detection in process networks, in IEEE Workshop on Signal Processing Systems, 2008, pp. 239 –244.

Jiong, L., Lin, Z., Yunsi, F. and Jha, N. (2004). Register binding-based RTL power management for control-flow intensive designs, IEEE Transactions on CAD of ICS, vol. 23, no. 8, pp. 1175 – 1183.

Johnson, N. E. (2004). Code Size Optimization for Embedded Processors, Ph.D. thesis, University of Cambridge.

Kadayif, I. and Kandemir, M. (2005). Data space-oriented tiling for enhancing locality, Trans. on Embedded Computing Sys., vol. 4, no. 2, pp. 388–414.

Kahn, G. (1974). The Semantics of a Simple language for Parallel Programming, in Proceedings of IFIP Congress, pp. 471–475, North Holland Publishing Company.

Kandemir, M., Son, S. W. and Chen, G. (2005). An evaluation of code and data optimizations in the context of disk power reduction, in ISLPED '05: Proceedings of the 2005 international symposium on Low power electronics and design, pp. 209–214, ISBN 1-59593-137-6.

Kandemir, M., Vijaykrishnan, N., Irwin, M. J. and Ye, W. (2001). Influence of compiler optimizations on system power, IEEE Trans. Very Large Scale Integr. Syst., vol. 9, pp. 801–804.
http://portal.acm.org/citation.cfm?id=505592.505597

Kandemir, M. T. (2006). Reducing energy consumption of multiprocessor SoC architectures by exploiting memory bank locality, ACM Trans. Des. Autom. Electron. Syst., vol. 11, no. 2, pp. 410–441.

Karakoy, M. (2005). Optimizing Array-Intensive Applications for On-Chip Multiprocessors, IEEE Trans. Parallel Distrib. Syst., vol. 16, no. 5, pp. 396–411.

Karfa, C. (2007). Hand-in-hand Verification and Synthesis of Digital Circuits, MS Thesis, IIT Kharagpur.

Karfa, C., Reddy, J., Mandal, C. R., Sarkar, D. and Biswas, S. (2005). SAST: An interconnection aware high-level synthesis tool, in Proc. 9th VLSI Design and Test Symposium, Bangalore, pp. 285–292.

Karp, R. M., Miller, R. E. and Winograd, S. (1967). The Organization of Computations for Uniform Recurrence Equations, J. ACM, vol. 14, pp. 563–590. http://doi.acm.org/10.1145/321406.321418

Kastner, R., Gong, W., Hao, X., Brewer, F., Kaplan, A., Brisk, P. and Sarrafzadeh, M. (2006). Layout driven data communication optimization for high level synthesis, in Proceedings of the conference on Design, automation and test in Europe: Proceedings, DATE '06, pp. 1185–1190, European Design and Automation Association, 3001 Leuven, Belgium, Belgium, ISBN 3-9810801-0-6. http://portal.acm.org/citation.cfm?id=1131481.1131809

Kelly, W., Rosser, E., Pugh, B., Wonnacott, D., Shpeisman, T. and Maslov, V. (2008). The Omega Calculator and Library, version 2.1, available at `http://www.cs.umd.edu/projects/omega/`.

Keutzer, K., Malik, S., Newton, A. R., Rabaey, J. M. and Sangiovanni-Vincentelli, A. (2000). System-Level Design: Orthogonalization of Concerns and Platform-Based Design, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 19, no. 12, pp. 1523–43.

Kienhuis, B., Rijpkema, E. and Deprettere, E. F. (2000). Compaan: Deriving Process Networks from Matlab for Embedded Signal Processing Architectures, in CODES'00, pp. 13–17.

Kim, T. and Liu, X. (2010). A functional unit and register binding algorithm for interconnect reduction, Trans. Comp.-Aided Des. Integ. Cir. Sys., vol. 29, pp. 641–646. http://dx.doi.org/10.1109/TCAD.2010.2042903

Kim, Y., Kopuri, S. and Mansouri, N. (2004). Automated Formal Verification of Scheduling Process Using Finite State Machines with Datapath (FSMD), in Proceedings of the 5th International Symposium on Quality Electronic Design, ISQED '04, pp. 110–115, IEEE Computer Society, Washington, DC, USA, ISBN 0-7695-2093-6.

Kim, Y. and Mansouri, N. (2008). Automated formal verification of scheduling with speculative code motions, in Proceedings of the 18th ACM Great Lakes symposium

on VLSI, GLSVLSI '08, pp. 95–100, ACM, New York, NY, USA, ISBN 978-1-59593-999-9.
http://doi.acm.org/10.1145/1366110.1366134

King, J. C. (1980). Program correctness: On inductive assertion methods, IEEE Trans. on Software Engineering, vol. SE-6, no. 5, pp. 465–479.

Knoop, J., Ruthing, O. and Steffen, B. (1992). Lazy Code Motion, in PLDI, pp. 224–234.

Krol, T., van Meerbergen, J., Niessen, C., Smits, W. and Huisken, J. (1992). The Sprite Input Language-an intermediate format for high levelsynthesis, in Proceedings. [3rd] European Conference on Design Automation, pp. 186–192.

Kundu, S., Lerner, S. and Gupta, R. (2008). Validating High-Level Synthesis, in Proceedings of the 20th international conference on Computer Aided Verification, CAV '08, pp. 459–472, Springer-Verlag, Berlin, Heidelberg, ISBN 978-3-540-70543-7.

Kundu, S., Lerner, S. and Gupta, R. (2010). Translation Validation of High-Level Synthesis, IEEE Transactions on CAD of ICS, vol. 29, no. 4, pp. 566–579.

Kundu, S., Tatlock, Z. and Lerner, S. (2009). Proving optimizations correct using parameterized program equivalence, in Proceedings of the 2009 ACM SIGPLAN conference on Programming language design and implementation, PLDI '09, pp. 327–337, ACM, New York, NY, USA, ISBN 978-1-60558-392-1.
http://doi.acm.org/10.1145/1542476.1542513

Lakshminarayana, G., Khouri, K. and Jha, N. (1997). Wavesched: A novel scheduling technique for control-flow intensive behavioural descriptions, in Proc. of ICCAD, pp. 244–250.

Lakshminarayana, G., Raghunathan, A. and Jha, N. (2000). Incorporating Speculative Execution into Scheduling of Control-Flow-Intensive Design, IEEE Transactions on CAD of ICS, vol. 19, no. 3, pp. 308–324.

Lakshminarayana, G., Raghunathan, A., Jha, N. K. and Dey, S. (1999). Power management in high-level synthesis, IEEE Trans. Very Large Scale Integr. Syst., vol. 7, pp. 7–15.
http://dx.doi.org/10.1109/92.748195

Lam, M. S. and Wilson, R. P. (1992). Limits of control flow on parallelism, in Proceedings of the 19th annual international symposium on Computer architecture, ISCA '92, pp. 46–57, ACM, New York, NY, USA, ISBN 0-89791-509-7. http://doi.acm.org/10.1145/139669.139702

Landwehr, B. and Marwedel, P. (1997). A New Optimization Technique for Improving Resource Exploitation and Critical Path Minimization, in ISSS, pp. 65–72, ACM, New York, NY, USA.

Lee, J.-H., Hsu, Y.-C. and Lin, Y.-L. (1989a). A new integer linear programming formulation for the scheduling problem in data path synthesis, in Procs. of the International Conference on Computer-Aided Design, pp. 20 –23, IEEE Computer Society, Washington, DC, USA.

Lee, J.-H., Hsu, Y.-C. and Y-L, L. (1989b). A New Integer Linear Formulation for the Scheduling Program in High Level Synthesis, in Procs. of the IEEE Conference on Computer-Aided Design, pp. 20–23.

Li, F. and Kandemir, M. (2005). Locality-conscious workload assignment for array-based computations in MPSOC architectures, in Proceedings of DAC '05, pp. 95–100, ISBN 1-59593-058-2.

Li, P., Agrawal, K., Buhler, J. and Chamberlain, R. D. (2010). Deadlock avoidance for streaming computations with filtering, in Proceedings of the 22nd ACM symposium on Parallelism in algorithms and architectures, SPAA '10, pp. 243–252, ISBN 978-1-4503-0079-7. http://doi.acm.org/10.1145/1810479.1810526

Lin, Z. and Jha, N. (2005). Interconnect-aware low-power high-level synthesis, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 24, no. 3, pp. 336 – 351.

lp_solve (2010). http://lpsolve.sourceforge.net/5.5/

Majumdar, R. and Xu, R.-G. (2007). Directed test generation using symbolic grammars, in The 6th Joint Meeting on European software engineering conference and the ACM SIGSOFT symposium on the foundations of software engineering: companion papers, ESEC-FSE companion '07, pp. 553–556, ACM, New York, NY,

USA, ISBN 978-1-59593-812-1.
http://doi.acm.org/10.1145/1295014.1295039

Mandal, A., Mandal, C. and Reade, C. (2007). A System for Automatic Evaluation of C Programs: Features and Interfaces, International Journal of Web-Based Learning and Teaching Technologies, vol. 2, no. 4, pp. 24–39.

Mandal, C. and Chakrabarti, P. P. (2003). Genetic Algorithms for High-Level Synthesis in VLSI Design, Materials and Manufacturing Processes, vol. 18, no. 3, pp. 355–383.

Mandal, C. and Zimmer, R. M. (2000). A Genetic Algorithm for the Synthesis of Structured Data Paths, in 13th International Conference on VLSI Design, pp. 206–211, IEEE Computer Society Press, ISBN 0-7695-0487-6.

Manna, Z. (1974). Mathematical Theory of Computation, McGraw-Hill Kogakusha, Tokyo.

Mansouri, N. and Vemuri, R. (1998). A Methodology for Automated Verification of Synthesized RTL Designs and Its Integration with a High-Level Synthesis Tool, in Proceedings of the Second International Conference on Formal Methods in Computer-Aided Design, FMCAD '98, pp. 204–221, Springer-Verlag, London, UK, ISBN 3-540-65191-8.

Mansouri, N. and Vemuri, R. (1999). Accounting for various register allocation schemes during post-synthesis verification of RTL designs, in Proceedings of the conference on Design, automation and test in Europe, DATE '99, pp. 223–230, ACM, New York, NY, USA, ISBN 1-58113-121-6.

Marwedel, P. (2006). Embedded System Design, Springer(India) Private Limited, New Delhi, India.

Meijer, S., Kienhuis, B., Turjan, A. and de Kock, E. (2007). Interactive presentation: A process splitting transformation for Kahn process networks, in DATE '07: Proceedings of the conference on Design, automation and test in Europe, pp. 1355–1360, ISBN 978-3-9810801-2-4.

Meijer, S., Nikolov, H. and Stefanov, T. (2009). On Compile-time Evaluation of Process Partitioning Transformations for Kahn Process Networks, in Proc. of

IEEE/ACM/IFIP Int. Conf. on HW/SW Codesign and System Synthesis (CODES-ISSS'09), pp. 31–40.

Meijer, S., Nikolov, H. and Stefanov, T. (2010a). Combining process splitting and merging transformations for Polyhedral Process Networks, in 8th IEEE Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia), 2010, pp. 97 –106.

Meijer, S., Nikolov, H. and Stefanov, T. (2010b). Throughput Modeling to Evaluate Process Merging Transformations in Polyhedral Process Networks, in Proc. of Int. Conf. Design, Automation and Test in Europe (DATE'10), pp. 747–752.

Menon, V., Pingali, K. and Mateev, N. (2003). Fractal symbolic analysis, ACM Trans. Program. Lang. Syst., vol. 25, no. 6, pp. 776–813.

Moon, S.-M. and Ebcioğlu, K. (1992). An efficient resource-constrained global scheduling technique for superscalar and VLIW processors, in Proceedings of the 25th annual international symposium on Microarchitecture, MICRO 25, pp. 55–71, IEEE Computer Society Press, Los Alamitos, CA, USA, ISBN 0-8186-3175-9. http://dx.doi.org/10.1145/144953.145000

Muchnick, S. S. (1997). Advanced compiler design and implementation, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, ISBN 1-55860-320-4.

Musoll, E. and Cortadella, J. (1995). High-level synthesis techniques for reducing the activity of functional units, in Proceedings of the ISLPED '95, pp. 99–104, ISBN 0-89791-744-8. http://doi.acm.org/10.1145/224081.224099

Nicolau, A. and Novack, S. (1993). Trailblazing: A Hierarchical Approach to Percolation Scheduling, in ICPP 1993. International Conference on Parallel Processing, 1993, vol. 2, pp. 120 –124.

Nikolov, H., Stefanov, T. and Deprettere, E. (2008). Systematic and Automated Multiprocessor System Design, Programming, and Implementation, IEEE Transactions on CAD of Integrated Circuits and Systems, vol. 27, no. 3, pp. 542–555.

Olson, A. and Evans, B. (2005). Deadlock detection for distributed process networks, in IEEE International Conference on Acoustics, Speech, and Signal Processing, 200, vol. 5, pp. v/73 – v/76 Vol. 5.

Palkovic, M., Catthoor, F. and Corporaal, H. (2009). Trade-offs in loop transformations, ACM Trans. Des. Autom. Electron. Syst., vol. 14, pp. 22:1–22:30. http://doi.acm.org/10.1145/1497561.1497565

Panda, P. and Dutt, N. (1995). 1995 high level synthesis design repository, in Proceedings of the 8th international symposium on System synthesis, ISSS '95, pp. 170–174, ACM, New York, NY, USA, ISBN 0-89791-771-5.

Panda, P. R., Catthoor, F., Dutt, N. D., Danckaert, K., Brockmeyer, E., Kulkarni, C., Vandercappelle, A. and Kjeldsberg, P. G. (2001). Data and memory optimization techniques for embedded systems, ACM Trans. Des. Autom. Electron. Syst., vol. 6, pp. 149–206. http://doi.acm.org/10.1145/375977.375978

Parker, A. C., Pizarro, J. T. and Mlinar, M. (1986). MAHA: a program for datapath synthesis, in Proceedings of the 23rd ACM/IEEE Design Automation Conference, DAC '86, pp. 461–466, IEEE Press, Piscataway, NJ, USA, ISBN 0-8186-0702-5. http://portal.acm.org/citation.cfm?id=318013.318087

Parks, T. M. (1995). Bounded Scheduling of Process Networks, Ph.D. thesis, EECS Department, University of California, Berkeley. http://www.eecs.berkeley.edu/Pubs/TechRpts/1995/2926.html

Paulin, P. G. and Knight, J. P. (1987). Force–Directed Scheduling in Automatic Data Path Synthesis, Procs. of the 24th Design Automation Conference.

Paulin, P. G. and Knight, J. P. (1989). Scheduling and binding algorithms for high-level synthesis, in Proceedings of the 26th ACM/IEEE Design Automation Conference, DAC '89, pp. 1–6, ACM, New York, NY, USA, ISBN 0-89791-310-8.

Potkonjak, M., Dey, S., Iqbal, Z. and Parker, A. (1993). High performance embedded system optimization using algebraic and generalized retiming techniques, in Proc. of ICCD, pp. 498 –504, IEEE Computer Society, Washington, DC, USA.

Qi, D., Roychoudhury, A., Liang, Z. and Vaswani, K. (2009). Darwin: an approach for debugging evolving programs, in Proceedings of the the 7th joint meeting of the European software engineering conference and the ACM SIGSOFT symposium on The foundations of software engineering, ESEC/FSE '09, pp. 33–42, ACM, New York, NY, USA, ISBN 978-1-60558-001-2. http://doi.acm.org/10.1145/1595696.1595704

Qiu, M., Sha, E. H. M., Liu, M., Lin, M., Hua, S. and Yang, L. T. (2008). Energy minimization with loop fusion and multi-functional-unit scheduling for multidimensional DSP, J. Parallel Distrib. Comput., vol. 68, no. 4, pp. 443–455.

Radhakrishnan, R., Teica, E. and Vermuri, R. (2000). An approach to high-level synthesis system validation using formally verified transformations, in Proceedings of the IEEE International High-Level Validation and Test Workshop (HLDVT'00), HLDVT '00, pp. 80–85, IEEE Computer Society, Washington, DC, USA, ISBN 0-7695-0786-7.

Raghavan, V. (2010). Principles of Compiler Design, Tata McGraw Hill Education Private Limited, New Delhi.

Raghunathan, A., Dey, S. and Jha, N. (1999). Register transfer level power optimization with emphasis on glitch analysis and reduction, IEEE Transactions on CAD of ICS, vol. 18, no. 8, pp. 1114 –1131.

Rajan, S. P. (1995). Correctness of Transformations in High Level Synthesis, in CHDL '95: 12th Conference on Computer Hardware Description Languages and their Applications, pp. 597–603, Chiba, Japan.

Raudvere, T., Sander, I. and Jantsch, A. (2008). Application and Verification of Local Nonsemantic-Preserving Transformations in System Design, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 27, no. 6, pp. 1091 –1103.

Rim, M., Fann, Y. and Jain, R. (1995). Global scheduling with code motions for high-level synthesis applications, IEEE Transactions on VLSI Systems, vol. 3, no. 3, pp. 379–392.

Ruthing, O., Knoop, J. and Steffen, B. (2000). Sparse Code Motion, in IEEE POPL, pp. 170–183.

Sampath, P., Rajeev, A. C., Ramesh, S. and Shashidhar, K. C. (2008). Behaviour Directed Testing of Auto-code Generators, in Proceedings of the 2008 Sixth IEEE International Conference on Software Engineering and Formal Methods, pp. 191–200, IEEE Computer Society, Washington, DC, USA, ISBN 978-0-7695-3437-4. http://portal.acm.org/citation.cfm?id=1475696.1476187

Samsom, H., Franssen, F., Catthoor, F. and De Man, H. (1995). System level verification of video and image processing specifications, in Proceedings of the 8th international symposium on System synthesis, ISSS '95, pp. 144–149, ACM, New York, NY, USA, ISBN 0-89791-771-5.
http://doi.acm.org/10.1145/224486.224533

Sarkar, D. and De Sarkar, S. (1989). Some inference rules for integer arithmetic for verification of flowchart programs on integers, IEEE Trans Software. Engg., vol. 15, no. 1, pp. 1–9.

Shashidhar, K. C. (2008). Efficient Automatic Verification of Loop and Data-flow Transformations by Functional Equivalence Checking, Ph.D. thesis, Department of Computer Science, Katholieke Universiteit Leuven, Belgium.

Shashidhar, K. C., Bruynooghe, M., Catthoor, F. and Janssens, G. (2002). Geometric Model Checking: An Automatic Verification Technique for Loop and Data Reuse Transformations, Electronic Notes in Theoretical Computer Science (ENTCS), vol. 65, no. 2, pp. 71–86.

Shashidhar, K. C., Bruynooghe, M., Catthoor, F. and Janssens, G. (2005a). Functional Equivalence Checking for Verification of Algebraic Transformations on Array-Intensive Source Code, in Proc. of DATE'05, pp. 1310–1315, ISBN 0-7695-2288-2.

Shashidhar, K. C., Bruynooghe, M., Catthoor, F. and Janssens, G. (2005b). Verification of Source Code Transformations by Program Equivalence Checking, in 14th International Conference on Compiler Construction (CC'05), pp. 221–236.

Sllame, A. and Drabek, V. (2002). An efficient list-based scheduling algorithm for high-level synthesis, in Euromicro Symposium on Digital System Design, 2002, pp. 316 – 323.

Strehl, K. and Thiele, L. (2000). Interval diagrams for efficient symbolic verification of process networks, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 19, no. 8, pp. 939 –956.

Synopsys (2011). Synphony C Compiler.
www.synopsys.com/Systems/BlockDesign/HLS/Pages/SynphonyC-Compiler.aspx

Trickey, H. (1987). Flamel: A High Level Hardware Compiler, IEEE Trans. on CAD., vol. 6, pp. 259–269.

Tsai, F.-S. and Hsu, Y.-C. (1992). STAR - An Automatic Data Path Allocator, IEEE Trans. on CAD., vol. 11, no. 9, pp. 1053–1064.

Tseng, C. J. and Siewiorek, D. P. (1986). Automated Synthesis of Data Paths in Digital-Systems, IEEE Trans. on CAD., vol. 5, no. 3, pp. 379–395.

Turjan, A. (2007). Compiling Nested Loop Programs to Process Networks, Ph.D. thesis, Leiden University.

Turjan, A., Kienhuis, B. and Deprettere, E. (2004). Translating affine nested-loop programs to process networks, in CASES '04: Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems, pp. 220–229, ISBN 1-58113-890-3.

Verdoolaege, S., Janssens, G. and Bruynooghe, M. (2009). Equivalence Checking of Static Affine Programs Using Widening to Handle Recurrences, in Proceedings of CAV '09, pp. 599–613, ISBN 978-3-642-02657-7.

Verdoolaege, S., Palkovič, M., Bruynooghe, M., Janssens, G. and Catthoor, F. (2010). Experience with Widening Based Equivalence Checking in Realistic Multimedia Systems, J. Electron. Test., vol. 26, no. 2, pp. 279–292.

Viswanath, V., Vasudevan, S. and Abraham, J. (2009). Dedicated Rewriting: Automatic Verification of Low Power Transformations in RTL, in 22nd International Conference on VLSI Design, 2009, pp. 77 –82.

Šimunić, T., Benini, L., De Micheli, G. and Hans, M. (2000). Source code optimization and profiling of energy consumption in embedded systems, in Proceedings of the 13th international symposium on System synthesis, ISSS '00, pp. 193–198, IEEE Computer Society, Washington, DC, USA, ISBN 1-58113-267-0. http://dx.doi.org/10.1145/501790.501831

Wakabayashi, K. and Yoshimura, T. (1989). A resource sharing and control synthesis method for conditional branches, in IEEE International Conference on Computer-Aided Design, 1989 (ICCAD-89), pp. 62 –65, IEEE Computer Society, Washington, DC, USA.

Wolf, M. E. and Lam, M. S. (1991). A Loop Transformation Theory and an Algorithm to Maximize Parallelism, IEEE Trans. Parallel Distrib. Syst., vol. 2, no. 4, pp. 452–471.

Xing, X. and Jong, C. C. (2007). A look-ahead synthesis technique with backtracking for switching activity reduction in low power high-level synthesis, Microelectron. J., vol. 38, pp. 595–605.
http://portal.acm.org/citation.cfm?id=1265997.1266263

Xue, L., Ozturk, O. and Kandemir, M. (2007). A memory-conscious code parallelization scheme, in Proceedings of the 44th annual Design Automation Conference, DAC '07, pp. 230–233, ACM, New York, NY, USA, ISBN 978-1-59593-627-1.
http://doi.acm.org/10.1145/1278480.1278536

Yices (2010). http://yices.csl.sri.com.

ZamanZadeh, S., Najibi, M. and Pedram, H. (2009). Pre-synthesis Optimization for Asynchronous Circuits Using Compiler Techniques, in Advances in Computer Science and Engineering, vol. 6 of *Communications in Computer and Information Science*, pp. 951–954, Springer Berlin Heidelberg, ISBN 978-3-540-89985-3.

Zhang, C. and Kurdahi, F. (2007). Reducing off-chip memory access via stream-conscious tiling on multimedia applications, Int. J. Parallel Program., vol. 35, no. 1, pp. 63–98.

Zhu, H. W. and Jong, C. C. (2002). Interconnection optimization in data path allocation using minimal cost maximal flow algorithm, Microelectronics Journal, vol. 33, no. 9, pp. 749 – 759.
http://www.sciencedirect.com/science/article/pii/S0026269202000484

Zhu, Y., Magklis, G., Scott, M. L., Ding, C. and Albonesi, D. H. (2004). The Energy Impact of Aggressive Loop Fusion, in PACT '04: Proceedings of the 13th International Conference on Parallel Architectures and Compilation Techniques, pp. 153–164, ISBN 0-7695-2229-7.

Zory, J. and Coelho, F. (1998). Using Algebraic Transformations to Optimize Expression Evaluation in Scientific Code, in Proceedings of the 1998 International Conference on Parallel Architectures and Compilation Techniques, PACT '98, pp. 376–384, IEEE Computer Society, Washington, DC, USA, ISBN 0-8186-8591-3.
http://portal.acm.org/citation.cfm?id=522344.825711

Zuck, L., Pnueli, A., Goldberg, B., Barrett, C., Fang, Y. and Hu, Y. (2005). Translation and Run-Time Validation of Loop Transformations, Form. Methods Syst. Des., vol. 27, no. 3, pp. 335–360.

Zuck, L., Pnueli, A., Y. Fang and Goldberg, B. (2003). VOC: A translation validator for optimizing compilers, Journal of Universal Computer Science, vol. 9, no. 3, pp. 223–247.