Author: Haastregt, Sven Joseph Johannes van
Title: Estimation and optimization of the performance of polyhedral process networks
Issue Date: 2013-12-17

Curriculum Vitae

Sven van Haastregt was born in Rijpwetering, the Netherlands, in 1985. He obtained his gymnasium diploma at Visser 't Hooft Lyceum, in Leiden, the Netherlands, in June 2003. In September 2003, Sven enrolled in the Computer Science track at Leiden University, the Netherlands. He received his BSc degree (with honors) in September 2006, and the MSc degree (with honors) in August 2008. After obtaining the MSc degree, Sven started his PhD research at the Leiden Embedded Research Center (LERC), part of the Leiden Institute of Advanced Computer Science (LIACS), Leiden University. He was involved in the MEDEA SoftSoC project on metamodeling of Hardware-dependent Software (HdS) and conducted research on high-level synthesis and performance estimation and optimization of polyhedral process networks. During his research, Sven (co-)authored various peer-reviewed conference and journal articles. He also contributed to the open-source PNgen and Espam tools, and initiated the PNtools software. From January 2010 to April 2010, Sven was a guest researcher at Xilinx Research Labs, in San Jose, California, United States of America. Besides his work as a researcher, Sven was involved as a teaching assistant in the Compiler Construction, Operating Systems, and Challenges in Computer Science Seminar courses which are part of the Computer Science bachelor program of Leiden University. The research culminated in the writing of this PhD dissertation in 2013. As of March 2013, Sven is working as a software engineer at ARM.

Acknowledgments

Back in 2006, when I was still an MSc. student, Sjoerd Meijer lured me into the Leiden Embedded Research Center (LERC). Unknowingly at the time, Sjoerd laid the foundation towards this dissertation, and for this I am very grateful. In addition, I would like to thank all present and former members of LERC for providing an inspiring working atmosphere. In particular I want to mention Todor Stefanov and fellow PhDs Teddy Zhai, Mohamed Bamakhrama, and Ana Balevic for the many interesting discussions that we had. I also acknowledge all other colleagues from the Leiden Institute of Advanced Computer Science (LIACS). In particular I want to mention André Deutz, Nies Huijsmans, Rudy van Vliet, and Hendrik Jan Hoogeboom, with whom I have enjoyed teaching courses on operating systems, compiler construction, and Android programming.

Over time I have supervised several BSc. and MSc. students who worked towards their graduation in the LERC group. I want to thank all of them, and in particular Eyal Halm and Wouter de Zwijger for their significant research contributions.

Sven Verdoolaege introduced me to the polyhedral model and developed a vast amount of software to tame this model. Upon reporting a bug or feature request, he often replied within a few hours with a source code patch that addressed my issue. This level of support is rare and I am therefore very grateful to him.

In 2008, Vinod Kathail and his team at Synfora introduced me to the world of high-level synthesis. In retrospect, this was an important first step towards this dissertation. Early 2010 I had the privilege of staying at Xilinx Research Labs in San Jose, CA, USA for a few months. This stay was made possible by Kees Vissers. I want to thank him for giving me this chance to work with state-of-the-art high-level synthesis tools, and for his support during my stay. I also want to thank Stephen Neuendorffer for sharing his extensive knowledge of high-level synthesis and FPGAs, and for reviewing my thesis and providing very detailed and useful feedback. I want to thank all colleagues from Xilinx with whom I worked, either directly or indirectly, in particular Juanjo Noguera, Chris Dick, and Ivo Bolsens.

This work was partly funded through the MEDEA+ SoftSoC project 2A714 about

Hardware-dependent Software for Systems on Chip. I would like to thank all persons with whom I have worked together in this project, in particular Ruud Derwig from Synopsys, Razvan Nane from TU Delft, Nikolay Kavaldjiev and Giuseppe Garcea from Compaan Design, and Mohammad Al Hissi from LERC.

Finally, I want to thank my parents, brother, family and friends, and all others who have shown interest in this work.

Bibliography

[ACDR09] K. Amiri, J. Cavallaro, C. Dick, and R. M. Rao. A High Throughput Configurable SDR Detector for Multi-user MIMO Wireless Systems. *Journal of Signal Processing Systems*, April 2009.

[AK87] R. Allen and K. Kennedy. Automatic Translation of FORTRAN Programs to Vector Form. *ACM Trans. on Programming Languages and Systems*, 9:pp. 491–542, 1987.

[Bal13] A. Balevic. *Exploiting Multi-Level Parallelism in Streaming Applications for Heterogeneous Platforms with GPUs*. Ph.D. thesis, Leiden University, 2013.

[Ban88] U. Banerjee. *Dependence Analysis for Supercomputing*. Kluwer, 1988.

[Ban97] U. Banerjee. *Dependence Analysis*. Loop Transformations for Restructuring Compilers. Kluwer, 1997.

[Bas04] C. Bastoul. Code Generation in the Polyhedral Model Is Easier Than You Think. In *Parallel Architecture and Compilation Techniques (PACT)*, pp. 7–16. September 2004.

[BBK$^+$08] U. Bondhugula, M. Baskaran, S. Krishnamoorthy, J. Ramanujam, A. Rountev, and P. Sadayappan. Automatic Transformations for Communication-Minimized Parallelization and Locality Optimization in the Polyhedral Model. In *Proc. of Compiler Construction (CC)*, pp. 132–146. March–April 2008.

[BBS09] T. Bijlsma, M. J. G. Bekooij, and G. J. M. Smit. Inter-Task Communication via Overlapping Read and Write Windows for Deadlock-Free Execution of Cyclic Task Graphs. In *Proc. of Systems, Architectures, Modeling and Simulation*, SAMOS'09, pp. 140–148. 2009.

[BBW$^+$05] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bolcskei. VLSI Implementation of MIMO Detection using the Sphere Decoding Algorithm. *IEEE Solid-State Circuits*, 40(7):pp. 1566–1577, 2005.

[BEF$^+$94] W. Blume, R. Eigenmann, K. Faigin, J. Grout, J. Hoeflinger, *et al.* Polaris: Improving the Effectiveness of Parallelizing Compilers. In *Languages and Compilers for Parallel Computing*, pp. 141–154. 1994.

[BELP96] G. Bilsen, M. Engels, R. Lauwereins, and J. Peperstraete. Cyclo-Static Dataflow. *IEEE Trans. on Signal Processing*, 44(2):pp. 397–408, 1996.

[BHLM94] J. Buck, S. Ha, E. Lee, and D. Messerschmitt. Ptolemy: A Framework for Simulating and Prototyping Heterogeneous Systems. *Journal of Computer Simulation*, 4:pp. 155–182, April 1994.

[BM10] B. Bailey and G. Martin. *ESL Models and Their Application: Electronic System Level Design and Verification in Practice*. Springer, 2010.

[BSE04] A. Bahasi, B. Saltzberg, and M. Ergen. *Multi-Carrier Digital Communications Theory and Applications of OFDM*. Springer, 2004.

[BZNS12] M. Bamakhrama, J. Zhai, H. Nikolov, and T. Stefanov. A Methodology for Automated Design of Hard-Real-Time Embedded Streaming Systems. In *Design, Autom., Test Eur. (DATE)*, pp. 941–946. March 2012.

[Cal11] Calypto Design Systems. Catapult C. `http://www.calypto.com/catapult_c_synthesis.php`, 2011.

[CFGV09] P. Clauss, F. Fernández, D. Garbervetsky, and S. Verdoolaege. Symbolic Polynomial Maximization over Convex Sets and its Application to Memory Requirement Estimation. *IEEE Trans. on VLSI Systems*, 17(8):pp. 983–996, August 2009.

[CFH$^+$06] J. Cong, Y. Fan, G. Han, W. Jiang, and Z. Zhang. Platform-Based Behavior-Level and System-Level Synthesis. In *Proc. of the Int. SoC Conference*. September 2006.

[CGMT09] P. Coussy, D. Gajski, M. Meredith, and A. Takach. An Introduction to High-Level Synthesis. *IEEE Des. Test*, 26:pp. 8–17, July 2009.

[Che04] J. Chelikowski. Introduction: Silicon in all its Forms. In P. Siffert and E. F. Krimmel, editors, *Silicon: Evolution and Future of a Technology*. Springer, 2004.

[CJLZ09] J. Cong, W. Jiang, B. Liu, and Y. Zou. Automatic Memory Partitioning and Scheduling for Throughput and Power Optimization. In *Intl. Conf. on Computer-Aided Design (ICCAD)*. November 2009.

[Cla07] Clang team. Clang: a C Language Family Frontend for LLVM. `http://clang.llvm.org/`, 2007.

[CLN$^+$11] J. Cong, B. Liu, S. Neuendorffer, J. Noguera, K. Vissers, and Z. Zhang. High-Level Synthesis for FPGAs: From Prototyping to Deployment. *IEEE TCAD*, 30:pp. 473–491, April 2011.

[CM08] P. Coussy and A. Morawiec, editors. *High-Level Synthesis – From Algorithm to Digital Circuit*. Springer, 2008.

[DBC$^+$07] H. Devos, K. Beyls, M. Christiaens, J. van Campenhout, E. D'Hollander, and D. Stroobandt. Finding and Applying Loop Transformations for Generating Optimized FPGA Implementations. *Trans. on High Perf. Embedded Architectures and Compilers I*, 4050:pp. 159–178, July 2007.

[Dec03] J. Decaluwe. MyHDL. `http://www.myhdl.org/`, 2003.

[DG98] A. Dasdan and R. Gupta. Faster Maximum and Minimum Mean Cycle Algorithms for System Performance Analysis. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 17(10):pp. 889–899, October 1998.

[DGP$^+$08] R. Dömer, A. Gerstlauer, J. Peng, D. Shin, L. Cai, *et al.* System-on-Chip Environment: A SpecC-Based Framework for Heterogeneous MPSoC Design. *EURASIP Journal on Embedded Systems*, 2008(1), 2008.

[DHRT07] H. Dutta, F. Hannig, H. Ruckdeschel, and J. Teich. Efficient Control Generation for Mapping Nested Loop Programs onto Processor Arrays. *Journal of Systems Architecture*, 53(5–6):pp. 300–309, 2007.

[DRV01] A. Darte, Y. Robert, and F. Vivien. Loop Parallelization Algorithms. In *Compiler Optimizations for Scalable Parallel Systems Languages*, pp. 141–172. 2001.

[DTD$^+$09] C. Dick, M. Trajkovic, S. Denic, D. Vuletic, R. Rao, *et al.* FPGA Implementation of a Near-ML Sphere Detector for 802.16e Broadband Wireless Systems. In *Proc. of the SDR'09 Conference*. December 2009.

[DTZ$^+$05] S. Derrien, A. Turjan, C. Zissulescu, B. Kienhuis, and E. Deprettere. Deriving Efficient Control in Process Networks with Compaan/Laura. *International Journal of Embedded Systems*, 2005.

[DV97] A. Darte and F. Vivien. Optimal Fine and Medium Grain Parallelism Detection in Polyhedral Reduced Dependence Graphs. *Intl. Journal of Parallel Programming*, 25(6):pp. 447–496, December 1997.

[EJL$^+$03] J. Eker, J. Janneck, E. Lee, J. Liu, X. Liu, *et al.* Taming Heterogeneity—the Ptolemy Approach. *Proceedings of the IEEE*, 91(2), January 2003.

[EZL89] D. Eager, J. Zahorjan, and E. Lazowska. Speedup Versus Efficiency in Parallel Systems. *IEEE Trans. Computers*, 38(3):pp. 408–423, 1989.

[Fea88] P. Feautrier. Parametric Integer Programming. *RAIRO Recherche Opérationnelle*, 22(3):pp. 243–268, 1988.

[Fea91] P. Feautrier. Dataflow Analysis of Scalar and Array References. *Intl. Journal of Parallel Programming*, 20(1):pp. 23–53, 1991.

[Fea92a] P. Feautrier. Some Efficient Solutions to the Affine Scheduling Problem, Part I: One-Dimensional Time. *Intl. Journal of Parallel Programming*, 21(5):pp. 313–348, October 1992.

[Fea92b] P. Feautrier. Some Efficient Solutions to the Affine Scheduling Problem, Part II: Multi-Dimensional Time. *Intl. Journal of Parallel Programming*, 21(6):pp. 389–420, December 1992.

[Fea96] P. Feautrier. Automatic Parallelization in the Polytope Model. In *The Data Parallel Programming Model: Foundations, HPF Realization, and Scientific Applications*, pp. 79–103. Springer-Verlag, 1996.

[Fea06] P. Feautrier. Scalable and Structured Scheduling. *Intl. Journal of Parallel Programming*, 34(5):pp. 459–487, October 2006.

[Fin10] M. Fingeroff. *High-Level Synthesis: Blue Book.* Xlibris Corp., 2010.

[FK08] K. Fazel and S. Kaiser. *Multi-Carrier and Spread Spectrum Systems*. Wiley, 2008.

[GAGS09] D. Gajski, S. Abdi, A. Gerstlauer, and G. Schirner. *Embedded System Design: Modeling, Synthesis and Verification*. Springer, 2009.

[GCD92] R. Gupta, C. Coelho, and G. DeMicheli. Synthesis and Simulation of Digital Systems Containing Interacting Hardware and Software Components. In *Design Automation Conf. (DAC)*, pp. 225–230. June 1992.

[Gei09] M. Geilen. Reduction Techniques for Synchronous Dataflow Graphs. In *Design Automation Conference (DAC)*, pp. 911–916. July 2009.

[Gem96] A. van Gemund. *Performance Modeling of Parallel Systems*. Ph.D. thesis, Delft University of Technology, April 1996.

[GGS+06] A. H. Ghamarian, M. C. W. Geilen, S. Stuijk, T. Basten, B. D. Theelen, *et al.* Throughput Analysis of Synchronous Data Flow Graphs. In *Conference on Application of Concurrency to System Design (ACSD)*, pp. 25–36. June 2006.

[GHC+09] L. Gao, J. Huang, J. Ceng, R. Leupers, G. Ascheid, and H. Meyr. TotalProf: a Fast and Accurate Retargetable Source Code Profiler. In *Proc. of CODES+ISSS*, pp. 305–314. October 2009.

[GKM82] S. Graham, P. Kessler, and M. McKusick. gprof: a Call Graph Execution Profiler. In *Symposium on Compiler Construction*, pp. 120–126. 1982.

[GL97] M. Griebl and C. Lengauer. The Loop Parallelizer LooPo. In D. Sehr *et al.*, editors, *Languages and Compilers for Parallel Computing*, volume 1239 of *LNCS*, pp. 603–604. Springer, 1997.

[GNB08] Z. Guo, W. Najjar, and B. Buyukkurt. Efficient Hardware Code Generation for FPGAs. *ACM Trans. Archit. Code Optim.*, 5(1):pp. 6:1–6:26, May 2008.

[GQR03] A.-C. Guillou, P. Quinton, and T. Risset. Hardware Synthesis for Multi-Dimensional Time. In *Application-Specific Systems, Architectures and Processors (ASAP)*, pp. 40–50. 2003.

[Gri04] M. Gries. Methods for Evaluating and Covering the Design Space during Early Design Development. *Integration, the VLSI Journal*, 38(2):pp. 131–183, 2004.

[GTA06] M. Gordon, W. Thies, and S. Amarasinghe. Exploiting Coarse-Grained Task, Data, and Pipeline Parallelism in Stream Programs.

In *Arch. Support for Prog. Lang. and Operating Syst. (ASPLOS)*, pp. 151–162. 2006.

[GZA^{+}11] T. Grosser, H. Zheng, R. Aloor, A. Simbürger, A. Größlinger, and L.-N. Pouchet. Polly – Polyhedral Optimization in LLVM. In *IMPACT Workshop*. April 2011.

[HH96] J. Hennessy and M. Heinrich. Hardware/Software Codesign of Processors: Concepts and Examples. In G. D. Micheli and M. Sami, editors, *Hardware/Software Codesign*, volume 310 of *NATO ASI*, pp. 29–44. Kluwer, 1996.

[HHK10] S. van Haastregt, E. Halm, and B. Kienhuis. Cost Modeling and Cycle-Accurate Co-Simulation of Heterogeneous Multiprocessor Systems. In *Design, Automation and Test in Europe (DATE'10)*, pp. 1297–1300. March 2010.

[HK09] S. van Haastregt and B. Kienhuis. Automated Synthesis of Streaming C Applications to Process Networks in Hardware. In *Design, Automation and Test in Europe (DATE'09)*, pp. 890–893. April 2009.

[HK12] S. van Haastregt and B. Kienhuis. Enabling Automatic Pipeline Utilization Improvement in Polyhedral Process Network Implementations. In *Appl.-specific Systems, Architectures and Proc. (ASAP'12)*. July 2012.

[HKL^{+}08] S. Ha, S. Kim, C. Lee, Y. Yi, S. Kwon, and Y.-P. Joo. PeaCE: A Hardware-Software Codesign Environment for Multimedia Embedded Systems. *ACM Trans. Des. Autom. Electron. Syst.*, 12(3):pp. 24:1–24:25, May 2008.

[HNVK11] S. van Haastregt, S. Neuendorffer, K. Vissers, and B. Kienhuis. High Level Synthesis for FPGAs Applied to a Sphere Decoder Channel Preprocessor. In *Symposium on Field Programmable Gate Arrays (FPGA'11)*, p. 278. February–March 2011.

[Hoa85] C. Hoare. *Communicating Sequential Processes*. Prentice Hall, 1985.

[HRDT08] F. Hannig, H. Ruckdeschel, H. Dutta, and J. Teich. PARO: Synthesis of Hardware Accelerators for Multi-Dimensional Dataflow-Intensive Applications. In *Applied Reconf. Computing (ARC)*, pp. 284–289. 2008.

[HZ^{+}10] S. van Haastregt, J. Zhai, *et al.* pntools. `http://daedalus.liacs.nl/pntools`, 2010.

[IP95] K. Ito and K. K. Parhi. Determining the Minimum Iteration Period of an Algorithm. *VLSI Signal Processing*, 11(3):pp. 229–244, 1995.

[JS05] A. Jantsch and I. Sander. Models of Computation and Languages for Embedded System Design. *IEE Proc. on Computers and Digital Techniques*, 152(2):pp. 114–129, March 2005.

[Kah74] G. Kahn. The Semantics of a Simple Language For Parallel Programming. In *Proc. of the IFIP Congress*. North-Holland Publishing, 1974.

[KDWV02] B. Kienhuis, E. Deprettere, P. v. d. Wolf, and K. Vissers. A Methodology to Design Programmable Embedded Systems – The Y-Chart Approach. In *Systems, Architectures, Modeling, Sim. (SAMOS)*, pp. 18–37. 2002.

[KES$^+$00] E. de Kock, G. Essink, W. Smits, P. van der Wolf, J.-Y. Brunel, *et al.* YAPI: Application Modeling for Signal Processing Systems. In *Design Automation Conference (DAC'00)*, pp. 402–405. June 2000.

[Khr08] Khronos Group. OpenCL – The Open Standard for Parallel Programming of Heterogeneous Systems. `http://www.khronos.org/opencl/`, 2008.

[KKO$^+$06] T. Kangas, P. Kukkala, H. Orsila, E. Salminen, M. Hännikäinen, *et al.* UML-based Multiprocessor SoC Design Framework. *ACM Trans. Embed. Comput. Syst.*, 5(2):pp. 281–320, 2006.

[KSS$^+$09] J. Keinert, M. Streubuehr, T. Schlichter, J. Falk, J. Gladigau, *et al.* SystemCoDesigner—an Automatic ESL Synthesis Approach by Design Space Exploration and Behavioral Synthesis for Streaming Applications. *ACM Trans. Des. Autom. Electron. Syst.*, 14(1):pp. 1–23, 2009.

[Kum88] M. Kumar. Measuring Parallelism in Computation-Intensive Scientific/Engineering Applications. *IEEE Trans. on Computers*, 37(9):pp. 1088–1098, September 1988.

[LA04] C. Lattner and V. Adve. LLVM: A Compilation Framework for Lifelong Program Analysis & Transformation. In *Code Generation and Optimization (CGO)*, pp. 75–88. March 2004.

[Lam88] M. Lam. Software Pipelining: An Effective Scheduling Technique for VLIW Machines. In *Programming Language Design and Implementation (PLDI)*, pp. 318–328. June 1988.

[LC10] R. Leupers and J. Castrillon. MPSoC Programming using the MAPS Compiler. In *Asia/South Pacific Design Automation Conference (ASP-DAC)*, pp. 897–902. January 2010.

[Lei08] Leiden Embedded Research Center. Daedalus home. `http://daedalus.liacs.nl/`, 2008.

[LL98] A. Lim and M. Lam. Maximizing Parallelism and Minimizing Synchronization with Affine Partitions. *Parallel Computing*, 24(3-4):pp. 445–476, 1998.

[LM87] E. Lee and D. Messerschmitt. Static Scheduling of Synchronous Dataflow Programs for Digital Signal Processing. *IEEE Trans. on Computers*, C-36(1):pp. 24–35, January 1987.

[LM98] W. Luk and S. McKeever. Pebble: A Language for Parametrised and Reconfigurable Hardware Design. In *Field-Programmable Logic and Applications (FPL)*, pp. 9–18. 1998.

[LRS93] C. Leiserson, F. Rose, and J. Saxe. Optimizing Synchronous Circuitry by Retiming. In *Third Caltech Conference On VLSI*. 1993.

[LS11] E. Lee and S. Seshia. *Introduction to Embedded Systems – A Cyber-Physical Systems Approach*. Lee and Seshia, 2011.

[LSV98] E. Lee and A. Sangiovanni-Vincentelli. A Framework for Comparing Models of Computation. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 17(12):pp. 1217–1229, 1998.

[LSWD01] P. Lieverse, T. Stefanov, P. van der Wolf, and E. Deprettere. System Level Design with Spade: an M-JPEG Case Study. In *Int. Conference on Computer Aided Design (ICCAD'01)*, pp. 31–38. 2001.

[Mar06] G. Martin. Overview of the MPSoC Design Challenge. In *Design Automation Conference (DAC)*, pp. 274–279. July 2006.

[Mar11] P. Marwedel. *Embedded System Design*. Springer, 2011.

[MBBM07] A. Moonen, M. Bekooij, R. v. d. Berg, and J. v. Meerbergen. Practical and Accurate Throughput Analysis with the Cyclo Static Dataflow Model. In *Proc. of MASCOTS*, pp. 238–245. 2007.

[MBGS10] O. Moreira, T. Basten, M. Geilen, and S. Stuijk. Buffer Sizing for Rate-Optimal Single-Rate Data-Flow Scheduling Revisited. *IEEE Trans. Comput.*, 59(2):pp. 188–201, 2010.

[Mei10] S. Meijer. *Transformations for Polyhedral Process Networks*. Ph.D. thesis, Leiden University, 2010.

[MK88] G. D. Micheli and D. Ku. HERCULES – A System for High-Level Synthesis. In *Design Automation Conference (DAC)*, pp. 483–488. 1988.

[MKMT90] G. D. Micheli, D. Ku, F. Mailhot, and T. Truong. The Olympus Synthesis System. *IEEE Design & Test*, 7(5):pp. 37–53, 1990.

[MNS09] S. Meijer, H. Nikolov, and T. Stefanov. On Compile-time Evaluation of Process Partitioning Transformations for Kahn Process Networks. In *Proc. of CODES+ISSS*, pp. 31–40. October 2009.

[MNS10] S. Meijer, H. Nikolov, and T. Stefanov. Throughput Modeling to Evaluate Process Merging Transformations in Polyhedral Process Networks. In *Design, Automation, and Test in Europe (DATE)*. March 2010.

[MPC88] M. McFarland, A. Parker, and R. Camposano. Tutorial on High-Level Synthesis. In *Design Automation Conf. (DAC)*, pp. 330–336. 1988.

[Muc97] S. Muchnick. *Advanced Compiler Design and Implementation*. Morgan Kaufmann, 1997.

[MVBG$^+$12] W. Meeus, K. Van Beeck, T. Goedemé, J. Meel, and D. Stroobandt. An Overview of Today's High-Level Synthesis Tools. *Design Automation for Embedded Systems*, August 2012.

[Nad12] D. Nadezhkin. *Parallelizing Dynamic Sequential Programs using Polyhedral Process Networks*. Ph.D. thesis, Leiden University, 2012.

[NC10] R. Nikhil and K. Czeck. *BSV by Example*. Bluespec, Inc., 2010.

[NHS$^+$11] R. Nane, S. van Haastregt, T. Stefanov, B. Kienhuis, V. M. Sima, and K. Bertels. IP-XACT Extensions for Reconfigurable Computing. In *Application-specific Systems, Architectures and Processors (ASAP'11)*, pp. 215–218. September 2011.

[Nik09] H. Nikolov. *System-Level Design Methodology for Streaming Multi-Processor Embedded Systems*. Ph.D. thesis, Leiden University, 2009.

[NNH$^+$10] J. Noguera, S. Neuendorffer, S. van Haastregt, J. Barba, K. Vissers, and C. Dick. Sphere Detector for 802.16E Broadband Wireless Systems Implementation on FPGAs using High-level Synthesis Tools.

In *Conference on Software Defined Radio (SDR'10)*. November–December 2010.

[NNH+11] J. Noguera, S. Neuendorffer, S. van Haastregt, J. Barba, K. Vissers, and C. Dick. Implementation of Sphere Decoder for MIMO-OFDM on FPGAs using High-Level Synthesis Tools. *Analog Integr. Circuits Signal Process.*, 69(2-3):pp. 119–129, December 2011.

[NRD+09] H. Nikolov, A. Rao, E. Deprettere, S. Nandy, and R. Narayan. A H.264 Decoder: A Design Style Comparison Case Study. In *Asilomar Conf. on Signals, Systems, and Computers*, pp. 236–242. November 2009.

[NS07] N. Nethercote and J. Seward. Valgrind: A Framework for Heavyweight Dynamic Binary Instrumentation. In *Programming Language Design and Implementation (PLDI)*, pp. 89–100. 2007.

[NSD08a] H. Nikolov, T. Stefanov, and E. Deprettere. Automated Integration of Dedicated Hardwired IP Cores in Heterogeneous MPSoCs Designed with ESPAM. *EURASIP Journal on Embedded Systems*, 2008.

[NSD08b] H. Nikolov, T. Stefanov, and E. Deprettere. Systematic and Automated Multi-processor System Design, Programming, and Implementation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 27(3):pp. 542–555, March 2008.

[NV08] S. Neuendorffer and K. Vissers. Streaming systems in FPGAs. In *Systems, Architectures, Modeling, and Simulation (SAMOS)*. July 2008.

[Ope97] OpenMP ARB. The OpenMP API Specification for Parallel Programming. `http://www.openmp.org/`, 1997.

[Pag96] I. Page. Constructing Hardware-Software Systems from a Single Description. *Journal of VLSI Signal Processing*, 12(1):pp. 87–107, 1996.

[Pap91] M. Papaefthymiou. Understanding Retiming Through Maximum Average-Weight Cycles. In *Symposium on Parallel Algorithms and Architectures (SPAA)*, pp. 338–348. 1991.

[PBCC08] L.-N. Pouchet, C. Bastoul, A. Cohen, and J. Cavazos. Iterative Optimization in the Polyhedral Model: part II, Multidimensional Time. In *Programming Language Design and Implementation*, pp. 90–100. 2008.

[PCB$^+$06] S. Pop, A. Cohen, C. Bastoul, S. Girbal, P. Jouvelot, *et al.* GRAPHITE: Loop Optimizations Based on the Polyhedral Model for GCC. In *Proc. of GCC Developper's Summit*, pp. 179–198. June 2006.

[PD76] J. Patel and E. Davidson. Improving the Throughput of a Pipeline by Insertion of Delays. In *Symposium on Computer Architecture (ISCA)*, pp. 159–164. January 1976.

[PEP06] A. Pimentel, C. Erbas, and S. Polstra. A Systematic Approach to Exploring Embedded System Architectures at Multiple Abstraction Levels. *IEEE Trans. on Computers*, 55(11):pp. 99–112, February 2006.

[PGS$^+$09] A. Papakonstantinou, K. Gururaj, J. Stratton, D. Chen, J. Cong, and W.-M. Hwu. FCUDA: Enabling Efficient Compilation of CUDA Kernels onto FPGAs. In *Appl. Specific Proc. (SASP)*, pp. 35–42. July 2009.

[PK89] P. Paulin and J. Knight. Scheduling and Binding Algorithms for High-Level Synthesis. In *Design Automation Conf. (DAC)*, pp. 1–6. 1989.

[PP12] R. Piscitelli and A. Pimentel. Design Space Pruning through Hybrid Analysis in System-level Design Space Exploration. In *Design, Automation and Test in Europe (DATE)*, pp. 781–786. March 2012.

[Pug91] W. Pugh. Uniform Techniques for Loop Optimization. In *International Conference on Supercomputing (ICS)*, pp. 341–352. ACM, 1991.

[Pug92] W. Pugh. Definitions of Dependence Distance. *ACM Lett. Program. Lang. Syst.*, 1(3):pp. 261–265, September 1992.

[PW86] D. Padua and M. Wolfe. Advanced compiler optimizations for supercomputers. *Commun. ACM*, 29(12):pp. 1184–1201, December 1986.

[Qui84] P. Quinton. Automatic Synthesis of Systolic Arrays from Uniform Recurrent Equations. In *ISCA*, pp. 208–214. 1984.

[RDK00] E. Rijpkema, E. Deprettere, and B. Kienhuis. Deriving Process Networks from Nested Loop Algorithms. *Parallel Processing Letters*, 10(2/3):pp. 165–176, 2000.

[Rij02] E. Rijpkema. *Modeling Task Level Parallelism in Piece-wise Regular Programs*. Ph.D. thesis, Leiden University, 2002.

[Row94] J. A. Rowson. Hardware/Software Co-Simulation. In *Design Automation Conference (DAC)*, pp. 439–440. 1994.

[SB00] S. Sriram and S. Bhattacharyya. *Embedded Multiprocessors: Scheduling and Synchronization.* Marcel Dekker, Inc., 2000.

[SD03] T. Stefanov and E. Deprettere. Deriving Process Networks from Weakly Dynamic Applications in System-Level Design. In *Hardware/software codesign (CODES)*, pp. 90–96. October 2003.

[SEJ11] M. Sackmann, P. Ebraert, and D. Janssens. A Model-Driven Approach for Software Parallelization. In *MoDELS Workshop*, pp. 276–290. 2011.

[SGB06] S. Stuijk, M. Geilen, and T. Basten. SDF^3: SDF For Free. In *Application of Concurrency to System Design (ACSD)*, pp. 276–278. June 2006.

[SGB08] S. Stuijk, M. Geilen, and T. Basten. Throughput-Buffering Trade-Off Exploration for Cyclo-Static and Synchronous Dataflow Graphs. *IEEE Trans. Comput.*, 57(10):pp. 1331–1345, October 2008.

[SHP12] H. Shen, M. Hamayun, and F. Pétrot. Native Simulation of MPSoC Using Hardware-Assisted Virtualization. *IEEE Trans. on CAD of Integrated Circuits and Systems (TCAD)*, 31(7):pp. 1074–1087, 2012.

[SKD02] T. Stefanov, B. Kienhuis, and E. Deprettere. Algorithmic Transformation Techniques for Efficient Exploration of Alternative Application Instances. In *HW/SW codesign (CODES)*, pp. 7–12. May 2002.

[SM93] T. Shepherd and J. McWhirter. *Systolic Adaptive Beamforming*, volume 25 of *Springer Series in Information Sc.* Springer-Verlag, 1993.

[Spe97] R. Spence. *The Acquisition of Insight.* Royal College of Art, 1997.

[Ste04] T. Stefanov. *Converting Weakly Dynamic Programs to Equivalent Process Network Specifications.* Ph.D. thesis, Leiden University, 2004.

[Syn10] Synopsys. SynphonyC. `http://www.synopsys.com/Systems/BlockDesign/HLS`, 2010.

[Sys05] SystemC. IEEE Std 1666, 2005.

[SZT$^+$04] T. Stefanov, C. Zissulescu, A. Turjan, B. Kienhuis, and E. Deprettere. System Design using Kahn Process Networks: The Compaan/Laura Approach. In *Design, Automation and Test in Europe (DATE'04)*, pp. 340–345. February 2004.

[TCL05] T. Todman, J. G. F. Coutinho, and W. Luk. Customisable Hardware Compilation. *The Journal of Supercomputing*, 32(2):pp. 119–137, 2005.

[TKD03] A. Turjan, B. Kienhuis, and E. Deprettere. Realizations of the Extended Linearization Model. In *Domain-Specific Embedded Multiprocessors*, chapter 9, pp. 171–191. Marcel Dekker, Inc., 2003.

[TKD07] A. Turjan, B. Kienhuis, and E. Deprettere. Classifying Interprocess Communication in Process Network Representation of Nested-Loop Programs. *ACM Trans. Embed. Comput. Syst.*, 6(2), May 2007.

[TS09] L. Thiele and N. Stoimenov. Modular Performance Analysis of Cyclic Dataflow Graphs. In *EMSOFT'09*, pp. 127–136. October 2009.

[Tur07] A. Turjan. *Compiling Nested Loop Programs to Process Networks*. Ph.D. thesis, Leiden University, 2007.

[Ver03a] S. Verdoolaege. Barvinok. `http://freecode.com/projects/barvinok`, 2003.

[Ver03b] S. Verdoolaege. Integer Set Analysis tool set. `http://repo.or.cz/w/isa.git`, 2003.

[Ver08] S. Verdoolaege. Integer Set Library. `http://repo.or.cz/w/isl.git`, 2008.

[Ver10] S. Verdoolaege. Polyhedral Process Networks. In S. Bhattacharrya, E. Deprettere, R. Leupers, and J. Takala, editors, *Handbook on Signal Processing Systems*, pp. 931–965. Springer, 2010.

[Viv02] F. Vivien. On the Optimality of Feautrier's Scheduling Algorithm. In *Proc. of Euro-Par Conference*, pp. 299–308. Springer-Verlag, 2002.

[VNS07] S. Verdoolaege, H. Nikolov, and T. Stefanov. PN: a Tool for Improved Derivation of Process Networks. *EURASIP Journal on Embedded Systems*, 2007.

[VSB$^+$07] S. Verdoolaege, R. Seghir, K. Beyls, V. Loechner, and M. Bruynooghe. Counting Integer Points in Parametric Polytopes using Barvinok's Rational Functions. *Algorithmica*, 48(1):pp. 37–66, May 2007.

[WL91] M. Wolf and M. Lam. A Loop Transformation Theory and an Algorithm to Maximize Parallelism. *IEEE Trans. Parallel Distributed Systems*, 2:pp. 452–471, October 1991.

[Xil02] Xilinx, Inc. System Generator. http://www.xilinx.com/tools/sysgen, 2002.

[Xil11] Xilinx, Inc. AutoESL. http://www.xilinx.com/tools/autoesl.htm, 2011.

[Xil13] Xilinx, Inc. Vivado HLS. http://www.xilinx.com/products/design-tools/vivado/integration/esl-design, 2013.

[YBK+07] Y. Yankova, K. Bertels, G. Kuzmanov, G. Gaydadjiev, Y. Lu, and S. Vassiliadis. DWARV: DelftWorkBench Automated Reconfigurable VHDL Generator. In *Field Progr. Logic and Appl. (FPL)*, pp. 697–701. 2007.

[ZDG97] J. Zhu, R. Dömer, and D. Gajski. Syntax and Semantics of the SpecC Language. In *Proc. of the SASIMI Workshop*, pp. 75–82. 1997.

[ZI08] C. Zissulescu-Ianculescu. *Synthesis of a Parallel Data Stream Processor from Data Flow Process Networks*. Ph.D. thesis, Leiden University, 2008.

[ZKD04] C. Zissulescu, B. Kienhuis, and E. Deprettere. Increasing Pipelined IP core Utilization in Process Networks using Exploration. In *Field-Programmable Logic and Applications (FPL)*, pp. 690–699. 2004.

[ZNS11] J. Zhai, H. Nikolov, and T. Stefanov. Modeling Adaptive Streaming Applications with Parameterized Polyhedral Process Networks. In *Design Automation Conference (DAC)*, pp. 116–121. 2011.

[ZSKD03] C. Zissulescu, T. Stefanov, B. Kienhuis, and E. Deprettere. LAURA: Leiden Architecture Research and Exploration Tool. In *Field Programmable Logic and Applications (FPL)*, pp. 911–920. 2003.

[ZTKD02] C. Zissulescu, A. Turjan, B. Kienhuis, and E. Deprettere. Solving Out of Order Communication using CAM Memory; an Implementation. In *Circuits, Systems and Signal Processing (ProRISC)*. 2002.

[Zwi12] W. de Zwijger. *Increasing Explicit Parallelism in Polyhedral Process Networks using Transformations*. Master's thesis, LIACS, Leiden University, June 2012.

INDEX