Author: Jongmans, Sung-Shik T.Q.
Title: Automata-theoretic protocol programming : parallel computation, threads and their interaction, optimized compilation, [at a] high level of abstraction
Issue Date: 2016-03-03

Bibliography

[ABC+06] Krste Asanovic, Ras Bodik, Bryan Catanzaro, Joseph Gebis, Parry Husbands, Kurt Keutzer, David Patterson, William Plishker, John Shalf, Samuel Williams, and Katherine Yelick. The Landscape of Parallel Computing: A View from Berkeley. Technical Report UCB/EECS-2006-183, University of California, Berkeley, 2006.

(Pages 1 and 275.)

[ABD+09] Krste Asanovic, Rastislav Bodik, James Demmel, Tony Keaveny, Kurt Keutzer, John Kubiatowicz, Nelson Morgan, David Patterson, Koushik Sen, John Wawrzynek, David Wessel, and Katherine Yelick. A View of the Parallel Computing Landscape. *Communications of the ACM*, 52(10):56–67, 2009.

doi:10.1145/1562764.1562783.

(Page 17.)

[ABdB+05] Farhad Arbab, Christel Baier, Frank de Boer, Jan Rutten, and Marjan Sirjani. Synthesis of Reo Circuits for Implementation of Component-Connector Automata Specifications. In *COORDINATION 2005*, volume 3454 of *LNCS*, pages 236–251. Springer, 2005.

doi:10.1007/11417019_16.

(Page 140.)

[ABdBR07] Farhad Arbab, Christel Baier, Frank de Boer, and Jan Rutten. Models and temporal logical specifications for timed component connectors. *Software & Systems Modeling*, 6(1):59–82, 2007.

doi:10.1007/s10270-006-0009-9.

(Page 31.)

[ACH05] Siva Anantharaman, Jing Chen, and Gaétan Hains. A Synchronous Process Calculus for Service Costs. In *SEFM 2005*, pages 435–444. IEEE, 2005.

doi:10.1109/SEFM.2005.6.

(Page 147.)

[AdBO09] Krzysztof Apt, Frank de Boer, and Ernst-Rüdiger Olderog. While Programs. In *Verification of Sequential and Concurrent Programs*, Texts in Computer Science, chapter 3, pages 55–126. Springer, 3rd edition, 2009.

doi:10.1007/978-1-84882-745-5_3.

(Pages 204, 205, 206, 207, and 280.)

[AFF01] Giorgio Ausiello, Paolo Franciosa, and Daniele Frigioni. Directed Hypergraphs: Problems, Algorithmic Results, and a Novel Decremental Approach. In *ICTCS 2001*, volume 2202 of *LNCS*, pages 312–328. Springer, 2001.

doi:10.1007/3-540-45446-2_20.

(Page 216.)

[AHS93] Farhad Arbab, Ivan Herman, and Per Spilling. An overview of Manifold and its implementation. *Concurrency: Practice and Experience*, 5(1):23–70, 1993.

doi:10.1002/cpe.4330050103.

(Page 14.)

[AKM08] Farhad Arbab, Natallia Kokash, and Sun Meng. Towards Using Reo for Compliance-Aware Business Process Modeling. In *Leveraging Applications of Formal Methods, Verification and Validation (ISoLA 2008)*, volume 17 of *CCIS*, pages 108–123. Springer, 2008.

doi:10.1007/978-3-540-88479-8_9.

(Page 60.)

[AM08] Farhad Arbab and Sun Meng. Synthesis of Connectors from Scenario-Based Interaction Specifications. In *CBSE 2008*, volume 5282 of *LNCS*, pages 114–129. Springer, 2008.

doi:10.1007/978-3-540-87891-9_8.

(Page 60.)

[Amd67] Gene Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *AFIPS SJCC 1967*, pages 483–485. ACM, 1967.

doi:10.1145/1465482.1465560.

(Pages 3 and 275.)

[Apt09a] Krzysztof Apt. Introduction. In *Principles of Constraint Programming*, chapter 1, pages 1–7. Cambridge University Press, 2nd edition, 2009.

(Pages 115 and 217.)

[Apt09b] Krzysztof Apt. Some Complete Constraint Solvers. In *Principles of Constraint Programming*, chapter 4, pages 82–134. Cambridge University Press, 2nd edition, 2009.

(Page 217.)

[AR03] Farhad Arbab and Jan Rutten. A Coinductive Calculus of Component Connectors. In *WADT 2002*, volume 2755 of *LNCS*, pages 34–55. Springer, 2003.

doi:10.1007/978-3-540-40020-2_2.

(Page 31.)

[Arb96] Farhad Arbab. The IWIM Model for Coordination of Concurrent Activities. In *COORDINATION 1996*, volume 1061 of *LNCS*, pages 34–56. Springer, 1996.

doi:10.1007/3-540-61052-9_38.

(Page 14.)

[Arb98] Farhad Arbab. What Do You Mean, Coordination? *Nieuwsbrief van de Nederlandse Vereniging voor Theoretische Informatica*, pages 10–21, 1998.

(Page 14.)

[Arb04] Farhad Arbab. Reo: a channel-based coordination model for component composition. *Mathematical Structures in Computer Science*, 14(3):329–366, 2004.

doi:10.1017/S0960129504004153.

(Pages 18, 22, 51, 58, 60, and 277.)

[Arb05] Farhad Arbab. Abstract Behavior Types: a foundation model for components and their composition. *Science of Computer Programming*, 55(1–3):3–52, 2005.

doi:10.1016/j.scico.2004.05.010.

(Page 31.)

[Arb11] Farhad Arbab. Puff, The Magic Protocol. In *Talcott Festschrift*, volume 7000 of *LNCS*, pages 169–206. Springer, 2011.

doi:10.1007/978-3-642-24933-4_9.

(Pages 10, 13, 58, 60, and 277.)

[Atk10] Alistair Atkinson. A Dynamic, Decentralised Search Algorithm for Efficient Data Retrieval in a Distributed Tuple Space. In *AusPDC 2010*, pages 21–30. ACM, 2010.

(Page 141.)

[AVWW96] Joe Armstrong, Robert Virding, Claes Wikström, and Mike Williams. Concurrent Programming. In *Concurrent Programming in* ERLANG, chapter 5, pages 67–84. Prentice Hall, 2nd edition, 1996.

(Pages 5 and 275.)

[BB15] Eduard Baranov and Simon Bliudze. Offer semantics: Achieving compositionality, flattening and full expressiveness for the glue operators in BIP. *Science of Computer Programming*, 109:2–35, 2015.

doi:10.1016/j.scico.2015.05.011.

(Page 247.)

[BBB+91] David Bailey, Eric Barszcz, John Barton, David Browning, Russell Carter, Leonardo Dagum, Rod Fatoohi, Paul Frederickson, Thomas Lasinski, Robert Schreiber, Horst Simon, Venkat Venkatakrishnan, and Sisira Weeratunga. The Nas Parallel Benchmarks. *International Journal of High Performance Computing Applications*, 5(3):63–73, 1991.

doi:10.1177/109434209100500306.

(Pages 23, 80, and 285.)

[BBB+94] David Bailey, Eric Barszcz, John Barton, David Browning, Russell Carter, Leonardo Dagum, Rod Fatoohi, Paul Frederickson, Thomas Lasinski, Robert Schreiber, Horst Simon, Venkat Venkatakrishnan, and Sisira Weeratunga. The Nas Parallel Benchmarks. Technical Report RNR-94-007, NASA, 1994.

(Pages 23, 79, 80, 81, and 285.)

[BBJ+12] Borzoo Bonakdarpour, Marius Bozga, Mohamad Jaber, Jean Quilbeuf, and Joseph Sifakis. A framework for automated distributed implementation of component-based models. *Distributed Computing*, 25(5):383–409, 2012.

doi:10.1007/s00446-012-0168-6.

(Page 141.)

[BBK+10] Christel Baier, Tobias Blechmann, Joachim Klein, Sascha Klüppelholz, and Wolfgang Leister. Design and Verification of Systems with Exogenous Coordination Using Vereofy. In *ISoLA 2010*, volume 6416 of *LNCS*, pages 97–111. Springer, 2010.

doi:10.1007/978-3-642-16561-0_15.

(Pages 31, 38, and 69.)

[BBKK09a] Christel Baier, Tobias Blechmann, Joachim Klein, and Sascha Klüppelholz. A Uniform Framework for Modeling and Verifying Components and Connectors. In *COORDINATION 2009*, volume 5521 of *LNCS*, pages 247–267. Springer, 2009.

doi:10.1007/978-3-642-02053-7_13.

(Pages 31, 38, and 69.)

[BBKK09b] Christel Baier, Tobias Blechmann, Joachim Klein, and Sascha Klüppelholz. Formal Verification for Components and Connectors. In *FMCO 2008*, volume 5751 of *LNCS*, pages 82–101. Springer, 2009.

doi:10.1007/978-3-642-04167-9_5.

(Pages 31 and 69.)

[BBS06] Ananda Basu, Marius Bozga, and Joseph Sifakis. Modeling Heterogeneous Real-time Components in BIP. In *SEFM 2006*, pages 3–12. IEEE, 2006.

doi:10.1109/SEFM.2006.27.

(Pages 14 and 141.)

[BCS09] Marcello Bonsangue, Dave Clarke, and Alexandra Silva. Automata for Context-Dependent Connectors. In *COORDINATION 2009*, volume 5521 of *LNCS*, pages 184–203. Springer, 2009.

doi:10.1007/978-3-642-02053-7_10.

(Page 246.)

[BCS12] Marcello Bonsangue, Dave Clarke, and Alexandra Silva. A model of context-dependent component connectors. *Science of Computer Programming*, 77(66):685–706, 2012.

doi:10.1016/j.scico.2011.01.006.

(Page 246.)

[Ben06] Mordechai Ben-Ari. Semaphores. In *Principles of Concurrent and Distributed Programming*, chapter 6, pages 107–144. Addison-Wesley, 2nd edition, 2006.

(Page 15.)

[Bjo93] Robert Bjornson. *Linda on Distributed Memory Multiprocessors*. PhD thesis, Yale University, 1993.

(Page 141.)

[BKK10] Tobias Blechmann, Joachim Klein, and Sascha Klüppelholz. *Vereofy V1.1 - User Manual*. Technische Universität Dresden, 2010.

(Pages 72 and 73.)

[BKK11] Christel Baier, Joachim Klein, and Sascha Klüppelholz. Modeling and Verification of Components and Connectors. In *SFM 2011*, volume 6659 of *LNCS*, pages 114–147. Springer, 2011.

doi:10.1007/978-3-642-21455-4_4.

(Pages 31 and 69.)

[BKK14] Christel Baier, Joachim Klein, and Sascha Klüppelholz. Synthesis of Reo Connectors for Strategies and Controllers. *Fundamenta Informatica*, 130(1):1–20, 2014.

doi:10.3233/FI-2014-980.

(Page 140.)

[BMFL02] Christian Bessière, Pedro Meseguer, Eugene Freuder, and Javier Larrosa. On forward checking for non-binary constraint satisfaction. *Artificial Intelligence*, 141(1–2):205–224, 2002.

doi:10.1016/S0004-3702(02)00263-1.

(Pages 115 and 217.)

[BO05] Miguel Bugalho and Arlindo Oliveira. Inference of regular languages using state merging algorithms with search. *Pattern Recognition*, 38(9):1457–1467, 2005.

doi:10.1016/j.patcog.2004.03.027.

(Page 247.)

[BS08] Simon Bliudze and Joseph Sifakis. The Algebra of Connectors—Structuring Interaction in BIP. *IEEE Transactions on Computers*, 57(10):1315–1330, 2008.

doi:10.1109/TC.2008.26.

(Pages 38, 61, and 247.)

[BS10] Simon Bliudze and Joseph Sifakis. Causal semantics for the algebra of connectors. *Formal Methods in System Design*, 36(2):167–194, 2010.

doi:10.1007/s10703-010-0091-z.

(Pages 38, 61, and 247.)

[BSAR06] Christel Baier, Marjan Sirjani, Farhad Arbab, and Jan Rutten. Modeling component connectors in Reo by constraint automata. *Science of Computer Programming*, 61(2):75–113, 2006.

doi:10.1016/j.scico.2005.10.008.

(Pages 22, 38, 41, 48, 60, and 277.)

[BST89] Henri Bal, Jennifer Steiner, and Andrew Tanenbaum. Programming Languages for Distributed Computing Systems. *ACM Computing Surveys*, 21(3):261–322, 1989.

doi:10.1145/72551.72552.

(Page 95.)

[BV08] Jasper Berendsen and Frits Vaandrager. Compositional Abstraction in Real-Time Model Checking. In *FORMATS 2008*, volume 5215 of *LNCS*, pages 233–249. Springer, 2008.

doi:10.1007/978-3-540-85778-5_17.

(Page 147.)

[BvG87] Jos Baeten and Rob van Glabbeek. Merge and Termination in Process Algebra. In *FST&TCS 1987*, volume 287 of *LNCS*, pages 153–172. Springer, 1987.

doi:10.1007/3-540-18625-5_49.

(Page 147.)

[CBM+08] Călin Caşcaval, Colin Blundell, Maged Michael, Harold Cain, Peng Wu, Stefanie Chiras, and Siddhartha Chatterjee. Software Transactional Memory: Why is it Only a Research Toy? *Communications of the ACM*, 51(11):40–46, 2008.

doi:10.1145/1400214.1400228.

(Page 16.)

[CCA07] Dave Clarke, David Costa, and Farhad Arbab. Connector colouring I: Synchronisation and context dependency. *Science of Computer Programming*, 66(3):205–225, 2007.

doi:10.1016/j.scico.2007.01.009.

(Pages 99 and 137.)

[CHY12] Marco Carbone, Kohei Honda, and Nobuko Yoshida. Structured Communication-Centered Programming for Web Services. *ACM Transactions on Programming Languages and Systems*, 34(2):8·1–8·78, 2012.

doi:10.1145/2220365.2220367.

(Page 294.)

[CKA10] Behnaz Changizi, Natallia Kokash, and Farhad Arbab. A Unified Toolset for Business Process Model Formalization. In *Preproceedings of FESCA 2010*, pages 147–156, 2010.

(Page 60.)

[CL00] Sung-Eun Choi and Christopher Lewis. A Study of Common Pitfalls in Simple Multi-Threaded Programs. *ACM SIGCSE Bulletin (SIGCSE 2000)*, 32(3):325–329, 2000.

doi:10.1145/330908.331879.

(Page 10.)

[CO94] Rafael Carrasco and Jose Oncina. Learning stochastic regular grammars by means of a state merging method. In *ICGI 1994*, volume 862 of *LNCS*, pages 139–152. Springer, 1994.

doi:10.1007/3-540-58473-0_144.

(Page 247.)

[Col88] Murray Cole. *Algorithmic Skeletons: a Structured Approach to the Management of Parallel Computation*. PhD thesis, University of Edinburgh, 1988.

(Pages 16 and 276.)

[CP12] Dave Clarke and José Proença. Partial Connector Colouring. In *COORDINATION 2012*, volume 7274 of *LNCS*, pages 59–73. Springer, 2012.

doi:10.1007/978-3-642-30829-1_5.

(Pages 99, 137, and 279.)

[CPLA11] Dave Clarke, José Proença, Alexander Lazovik, and Farhad Arbab. Channel-based coordination via constraint satisfaction. *Science of Computer Programming*, 76(8):681–710, 2011.

doi:10.1016/j.scico.2010.05.004.

(Pages 38, 99, and 220.)

[Dij82] Edsger Dijkstra. On the Role of Scientific Thought. In *Selected Writings on Computing: A Personal Perspective*, Texts and Monographs in Computer Science, chapter EWD447, pages 60–66. Springer, 1982.

doi:10.1007/978-1-4612-5695-3_12.

(Page 14.)

[Dij02] Edsger Dijkstra. Cooperating Sequential Processes. In *The Origin of Concurrent Programming*, chapter 1, pages 65–138. Springer, 2002.

doi:10.1007/978-1-4757-3472-0_2.

(Pages 6 and 275.)

[DJAB15] Kasper Dokter, **Sung-Shik Jongmans**, Farhad Arbab, and Simon Bliudze. Relating BIP and Reo. In *ICE 2015*, volume 189 of *EPTCS*, pages 3–20. CoRR, 2015.

doi:10.4204/EPTCS.189.3.

(Page 61.)

[DM98] Leonardo Dagum and Ramesh Menon. OpenMP: An Industry Standard API for Shared-Memory Programming. *IEEE Computational Science & Engineering*, 5(1):46–55, 1998.

doi:10.1109/99.660313.

(Pages 5 and 275.)

[FGY94] Ming-Dong Feng, Yao-Qing Gao, and Chung-Kwong Yuen:. Distributed Linda Tuplespace Algorithms and Implementations. In *Parallel Processing: CONPAR 94 – VAPP VI*, volume 854 of *LNCS*, pages 581–592. Springer, 1994.

doi:10.1007/3-540-58430-7_51.

(Pages 140 and 141.)

[FSJY02] Michael Frumkin, Matthew Schultz, Haoqiang Jin, and Jerry Yan. Implementation of the NAS Parallel Benchmarks in Java. Technical Report NAS-02-009, NASA, 2002.

(Pages 80, 82, 83, and 291.)

[FSJY03] Michael Frumkin, Matthew Schultz, Haoqiang Jin, and Jerry Yan. Performance and Scalability of the NAS Parallel Benchmarks in Java. In *IPDPS 2003*, pages 139–44. IEEE, 2003.

doi:10.1109/IPDPS.2003.1213267.

(Pages 80, 285, and 291.)

[FWY96] Ming-Dong Feng, Weng-Fai Wong, and Chung-Kwong Yuen. BaLinda Lisp: Design and implementation. *Computer Languages*, 22(4):205–214, 1996.

doi:10.1016/S0096-0551(96)00016-1.

(Page 140.)

[Gel85] David Gelernter. Generative Communication in Linda. *Transactions on Programming Languages and Systems*, 7(1):80–112, 1985.

doi:10.1145/2363.2433.

(Page 140.)

[GL02] Dimitra Giannakopoulou and Flavio Lerda. From States to Transitions: Improving Translation of LTL Formulae to Büchi Automata. In *FORTE 2002*, volume 2529 of *LNCS*, pages 308–326. Springer, 2002.

doi:10.1007/3-540-36135-9_20.

(Page 247.)

[GL10] Horacio González-Vélez and Mario Leyton. A survey of algorithmic skeleton frameworks: high-level structured parallel programming enablers. *Software: Practice and Experience*, 40(12):1135–1160, 2010.

doi:10.1002/spe.1026.

(Pages 17 and 276.)

[GLPN93] Giorgio Gallo, Giustino Longo, Stefano Pallottino, and Sang Nguyen. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42(2–3):177–201, 1993.

doi:10.1016/0166-218X(93)90045-P.

(Page 213.)

[GPB+06] Brian Goetz, Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes, and Doug Lea. Task Execution. In *Java Concurrency in Practice*, chapter 6, pages 113–134. Addison-Wesley, 2006.

(Pages 5 and 275.)

[Hal11] Sean Halle. *A Study of Frameworks for Collectively Meeting the Productivity, Portability, and Adoptability Goals for Parallel Software*. PhD thesis, University of California, Santa Cruz, 2011.

(Page 292.)

[Hal12] Philipp Haller. On the Integration of the Actor Model in Mainstream Technologies. In *AGERE! 2012*, pages 1–6. ACM, 2012.

doi:10.1145/2414639.2414641.

(Pages 5 and 275.)

[HC13] Sean Halle and Albert Cohen. A Mutable Hardware Abstraction to Replace Threads. In *LCPC 2011*, volume 7146 of *LNCS*, pages 185–202. Springer, 2013.

doi:10.1007/978-3-642-36036-7_13.

(Page 292.)

[Her14] Maurice Herlihy. The Multicore Transformation. *Ubiquity*, 2014(9):1–9, 2014.

doi:10.1145/2618405.

(Page 16.)

[HM93] Maurice Herlihy and Eliot Moss. Transactional Memory: Architectural Support for Lock-Free Data Structures. *ACM SIGARCH Computer Architecture News (ISCA 1993)*, 21(2):289–300, 1993.

doi:10.1145/173682.165164.

(Pages 16 and 276.)

[HMU06] John Hopcroft, Rajeev Motwani, and Jeffrey Ullman. Pushdown Automata. In *Introduction to Automata Theory, Languages and Computation*, chapter 6, pages 225–260. Addison-Wesley, 3rd edition, 2006.

(Page 39.)

[HO09] Philipp Haller and Martin Odersky. Scala Actors: Unifying thread-based and event-based programming. *Theoretical Computer Science*, 410(2–3):202–220, 2009.

doi:10.1016/j.tcs.2008.09.019.

(Pages 5 and 275.)

[Hoa69] Tony Hoare. An Axiomatic Basis for Computer Programming. *Communications of the ACM*, 12(10):576–580, 1969.

doi:10.1145/363235.363259.

(Pages 205 and 280.)

[Hoa74] Tony Hoare. Monitors: An Operating System Structuring Concept. *Communications of the ACM*, 17(10):549–557, 1974.

doi:10.1145/355620.361161.

(Pages 6 and 275.)

[HP11a] John Hennessy and David Patterson. Instruction-Level Parallelism and Its Exploitation. In *Computer Architecture: A Quantitative Approach*, chapter 3, pages 145–259. Elsevier, 5th edition, 2011.

(Page 1.)

[HP11b] John Hennessy and David Patterson. Memory Hierarchy Design. In *Computer Architecture: A Quantitative Approach*, chapter 2, pages 69–144. Elsevier, 5th edition, 2011.

(Page 165.)

[HP11c] John Hennessy and David Patterson. Vector Processors. In *Computer Architecture: A Quantitative Approach*, chapter G, pages G·1–G·34. Elsevier, 5th edition, 2011.

(Page 80.)

[HVK98] Kohei Honda, Vasco Vasconcelos, and Makoto Kubo. Language Primitives and Type Discipline for Structured Communication-Based Programming. In *ESOP 1998*, volume 1381 of *LNCS*, pages 122–138. Springer, 1998.

doi:10.1007/BFb0053567.

(Page 294.)

[HYC08] Kohei Honda, Nobuko Yoshida, and Marco Carbone. Multiparty Asynchronous Session Types. *ACM SIGPLAN Notices (POPL 2008)*, 43(1):273–284, 2008.

doi:10.1145/1328897.1328472.

(Page 294.)

[IBC11] Mohammad Izadi, Marcello Bonsangue, and Dave Clarke. Büchi automata for modeling component connectors. *Software & Systems Modeling*, 10(2):183–200, 2011.

doi:10.1007/s10270-010-0152-1.

(Page 31.)

[Iza11] Mohammad Izadi. *Model Checking of Component Connectors*. PhD thesis, Universiteit Leiden, 2011.

hdl:1887/18189.

(Page 31.)

[JA11] **Sung-Shik Jongmans** and Farhad Arbab. Correlating Semantic Models of Reo Connectors: Connector Coloring and Constraint Automata. In *ICE 2011*, volume 59 of *EPTCS*, pages 84–103. CoRR, 2011.

doi:10.4204/EPTCS.59.8.

(Page 25.)

[JA12] **Sung-Shik Jongmans** and Farhad Arbab. Overview of Thirty Semantic Formalisms for Reo. *Scientific Annals of Computer Science*, 22(1):201–251, 2012.

doi:10.7561/SACS.2012.1.201.

(Pages 25 and 60.)

[JA13a] **Sung-Shik Jongmans** and Farhad Arbab. Global Consensus through Local Synchronization. In *Advances in Service-Oriented and Cloud Computing (FOCLASA 2013)*, volume 393 of *CCIS*, pages 174–188. Springer, 2013.

doi:10.1007/978-3-642-45364-9_15.

(Pages 25 and 127.)

[JA13b] **Sung-Shik Jongmans** and Farhad Arbab. Modularizing and Specifying Protocols among Threads. In *PLACES 2012*, volume 109 of *EPTCS*, pages 34–45. CoRR, 2013.

doi:10.4204/EPTCS.109.6.

(Pages 25 and 95.)

[JA14] **Sung-Shik Jongmans** and Farhad Arbab. Toward Sequentializing Overparallelized Protocol Code. In *ICE 2014*, volume 166 of *EPTCS*, pages 38–44. CoRR, 2014.

(Pages 25, 95, and 127.)

[JA15a] **Sung-Shik Jongmans** and Farhad Arbab. Can High Throughput Atone for High Latency in Compiler-Generated Protocol Code? In *FSEN 2015*, volume 9392 of *LNCS*, pages 238–258. Springer, 2015.

doi:10.1007/978-3-319-24644-4_17.

(Page 25.)

[JA15b] **Sung-Shik Jongmans** and Farhad Arbab. Take Command of Your Constraints! In *COORDINATION 2015*, volume 9037 of *LNCS*, pages 117–132. Springer, 2015.

doi:10.1007/978-3-319-19282-6_8.

(Pages 25 and 203.)

[JA16] **Sung-Shik Jongmans** and Farhad Arbab. Global consensus through local synchronization: A formal basis for partially-distributed coordination. *Science of Computer Programming*, 115–116:199–224, 2016.

doi:10.1016/j.scico.2015.09.001.

(Pages 25, 95, and 127.)

[JCP12] **Sung-Shik Jongmans**, Dave Clarke, and Jose Proença. A Procedure for Splitting Processes and its Application to Coordination. In *FOCLASA 2012*, volume 91 of *EPTCS*, pages 79–96. CoRR, 2012.

doi:10.4204/EPTCS.91.6.

(Pages 25, 138, and 279.)

[JCP16] **Sung-Shik Jongmans**, Dave Clarke, and Jose Proença. A procedure for splitting data-aware processes and its application to coordination. *Science of Computer Programming*, 115–116:47–78, 2016.

doi:10.1016/j.scico.2014.02.017.

(Pages 25, 138, and 279.)

[JFY99] Haoqiang Jin, Michael Frumkin, and Jerry Yan. The OpenMP Implementation of NAS Parallel Benchmarks and Its Performance. Technical Report NAS-99-011, NASA, 1999.

(Page 80.)

[JHA14a] **Sung-Shik Jongmans**, Sean Halle, and Farhad Arbab. Automata-based Optimization of Interaction Protocols for Scalable Multicore Platforms. In *COORDINATION 2014*, volume 8459 of *LNCS*, pages 65–82. Springer, 2014.

doi:10.1007/978-3-662-43376-8_5.

(Pages 25, 239, 246, 292, and 293.)

[JHA14b] **Sung-Shik Jongmans**, Sean Halle, and Farhad Arbab. Reo: A Dataflow Inspired Language for Multicore. In *DFM 2013*, pages 42–50. IEEE, 2014.

doi:10.1109/DFM.2013.14 .

(Pages 25, 292, and 293.)

[JKA11] **Sung-Shik Jongmans**, Christian Krause, and Farhad Arbab. Encoding Context-Sensitivity in Reo into Non-Context-Sensitive Semantic Models. In *COORDINATION 2011*, volume 6721 of *LNCS*, pages 31–48. Springer, 2011.

doi:10.1007/978-3-642-21464-6_3.

(Pages 25 and 247.)

[JKA16] **Sung-Shik Jongmans**, Tobias Kappé, and Farhad Arbab. Composing Constraint Automata, State-by-State. In *FACS 2015*, volume 9539 of *LNCS*, chapter 12. Springer, 2016.

doi:10.1007/978-3-319-28934-2_12.

(Page 25.)

[**Jon**16] **Sung-Shik Jongmans**. Automata-Theoretic Protocol Programming (With Proofs). Technical Report FM-1601, Centrum Wiskunde & Informatica, 2016.

urn:nbn:nl:ui:18-24063.

(Pages 25, 61, 62, and 67.)

[JSA14] **Sung-Shik Jongmans**, Francesco Santini, and Farhad Arbab. Partially-Distributed Coordination with Reo. In *PDP 2014*, pages 697–706. IEEE, 2014.

doi:10.1109/PDP.2014.19.

(Pages 25 and 127.)

[JSA15] **Sung-Shik Jongmans**, Francesco Santini, and Farhad Arbab. Partially-Distributed Coordination with Reo and Constraint Automata. *Service Oriented Computing and Applications*, 9(3):311–339, 2015.

doi:10.1007/s11761-015-0177-y.

(Pages 25, 127, and 294.)

[JSS$^+$12] **Sung-Shik Jongmans**, Francesco Santini, Mahdi Sargolzaei, Farhad Arbab, and Hamideh Afsarmanesh. Automatic Code Generation for the Orchestration of Web Services with Reo. In *ESOCC 2012*, volume 7592 of *LNCS*, pages 1–16. Springer, 2012.

doi:10.1007/978-3-642-33427-6_1.

(Page 25.)

[JSS$^+$14] **Sung-Shik Jongmans**, Francesco Santini, Mahdi Sargolzaei, Farhad Arbab, and Hamideh Afsarmanesh. Orchestrating Web Services using Reo: From Circuits and Behaviors to Automatically Generated Code. *Service Oriented Computing and Applications*, 8(4):277–297, 2014.

doi:10.1007/s11761-013-0147-1.

(Page 25.)

[Kah62] Arthur Kahn. Topological Sorting in Large Networks. *Communications of the ACM*, 5(11):558–562, 1962.

doi:10.1145/368996.369025.

(Page 209.)

[KB09] Sascha Klüppelholz and Christel Baier. Symbolic model checking for channel-based component connectors. *Science of Computer Programming*, 74(9):688–701, 2009.

doi:10.1016/j.scico.2008.09.020.

(Pages 31, 38, and 69.)

[KB10] Sascha Klüppelholz and Christel Baier. Alternating-time stream logic for multi-agent systems. *Science of Computer Programming*, 75(6):398–425, 2010.

doi:10.1016/j.scico.2009.07.007.

(Page 31.)

[KC09] Christian Koehler and Dave Clarke. Decomposing Port Automata. In *SAC 2009*, pages 1369–1373. ACM, 2009.

doi:10.1145/1529282.1529587.

(Pages 38 and 140.)

[KGdV13] Christian Krause, Holger Giese, and Erik de Vink. Compositional and behavior-preserving reconfiguration of component connectors in Reo. *Journal of Visual Languages & Computing*, 24(3):153–168, 2013.

doi:10.1016/j.jvlc.2012.09.002.

(Pages 38 and 48.)

[KKSB11] Joachim Klein, Sascha Klüppelholz, Andries Stam, and Christel Baier. Hierarchical Modeling and Formal Verification. An Industrial Case Study Using Reo and Vereofy. In *FMICS 2011*, volume 6959 of *LNCS*, pages 228–243. Springer, 2011.

doi:10.1007/978-3-642-24431-5_17.

(Page 69.)

[Kla10] Martin Klazar. Some general results in combinatorial enumeration. In *Permutation Patterns*, volume 376 of *London Mathematical Society Lecture Note Series*, pages 3–40. Cambridge University Press, 2010.

(Page 130.)

[Kle12] Joachim Klein. *Compositional Synthesis and Most General Controllers*. PhD thesis, Technische Universität Dresden, 2012.

urn:nbn:de:bsz:14-qucosa-130654.

(Page 31.)

[Klü12] Sascha Klüppelholz. *Verification of Branching-Time and Alternating-Time Properties for Exogenous Coordination Models*. PhD thesis, Technische Universität Dresden, 2012.

urn:nbn:de:bsz:14-qucosa-86211.

(Pages 31, 38, 48, and 69.)

[KM03] Dietrich Kuske and Ingmar Meinecke. Branching Automata with Costs — A Way of Reflecting Parallelism in Costs. In *CIAA 2003*, volume 2759 of *LNCS*, pages 150–162. Springer, 2003.

doi:10.1007/3-540-45089-0_15.

(Page 149.)

[Kni86] Tom Knight. An Architecture for Mostly Functional Languages. In *LFP 1986*, pages 105–112. ACM, 1986.

doi:10.1145/319838.319854.

(Pages 16 and 276.)

[Knu97] Donald Knuth. Information Structures. In *Fundamental Algorithms*, volume 1 of *The Art of Computer Programming*, chapter 2, pages 232–465. Addison-Wesley, 3rd edition, 1997.

(Page 209.)

[KQCM09] David Kitchin, Adrian Quark, William Cook, and Jayadev Misra. The Orc Programming Language. In *FMOODS/FORTE 2009*, volume 5522 of *LNCS*, pages 1–25. Springer, 2009.

doi:10.1007/978-3-642-02138-1_1.

(Page 149.)

[KS08] Bartek Klin and Vladimiro Sassone. Structural Operational Semantics for Stochastic Process Calculi. In *FOSSACS 2008*, volume 4962 of *LNCS*, pages 428–442. Springer, 2008.

doi:10.1007/978-3-540-78499-9_30.

(Page 148.)

[KV08] Bernhard Korte and Jens Vygen. Spanning Trees and Arborescences. In *Combinatorial Optimization: Theory and Algorithms*, volume 21 of *Algorithms and Combinatorics*, chapter 6, pages 127–150. Springer, 4th edition, 2008.

doi:10.1007/978-3-540-71844-4.

(Page 213.)

[Lam86] Leslie Lamport. The Mutual Exclusion Problem: Part I—A Theory of Interprocess Communication. *Journal of the ACM*, 33(2):313–326, 1986.

doi:10.1145/5383.5384.

(Page 6.)

[Lea11] Diana Lea. The Dictionary. In *Oxford Advanced American Dictionary*, pages 1–1736. Oxford University Press, 2011.

(Page 27.)

[Lee06] Edward Lee. The Problem with Threads. *Computer*, 39(5):33–42, 2006.

doi:10.1109/MC.2006.180.

(Pages 6 and 10.)

[LNZ14] Edward Lee, Stephen Neuendorffer, and Gang Zhou. Synchronous-Reactive Models. In *System Design, Modeling, and Simulation using Ptolemy II*, chapter 5, pages 158–185. Ptolemy.org, 1st edition, 2014.

(Page 220.)

[LPP98] Kevin Lang, Barak Pearlmutter, and Rodney Price. Results of the Abbadingo One DFA Learning Competition and a New Evidence-Driven State Merging Algorithm. In *Grammatical Inference*, volume 1433 of *LNCS*, pages 1–12. Springer, 1998.

doi:10.1007/BFb0054059.

(Page 247.)

[LSB09] Daan Leijen, Wolfram Schulte, and Sebastian Burckhardt. The Design of a Task Parallel Library. *ACM SIGPLAN Notices (OOPSLA 2009)*, 44(10):227–242, 2009.

doi:10.1145/1639949.1640106.

(Pages 5 and 275.)

[MAB11] Sun Meng, Farhad Arbab, and Christel Baier. Synthesis of Reo circuits from scenario-based interaction specifications. *Science of Computer Programming*, 76(8):651–680, 2011.

doi:10.1016/j.scico.2010.03.002.

(Page 60.)

[Mil89] Robin Milner. Strong Bisimulation and Strong Equivalence. In *Communication and Concurrency*, chapter 4, pages 84–105. Prentice Hall, 1989.

(Pages 40 and 277.)

[Moo98] Gordon Moore. Cramming More Components onto Integrated Circuits. *the IEEE*, 86(1):114–117, 1998.

doi:10.1109/JPROC.1998.658762.

(Page 1.)

[MRR12] Michael McCool, Arch Robinson, and James Reinders. Introduction. In *Structured Parallel Programming*, chapter 1, pages 1–38. Elsevier, 2012.

(Page 17.)

[MSM05] Timothy Mattson, Beverly Sanders, and Berna Massingill. A Pattern Language for Parallel Programming. In *Patterns for Parallel Programming*, SPS, chapter 1, pages 1–6. Addison-Wesley, 2005.

(Page 17.)

[MT03] Ronaldo Menezes and Robert Tolksdorf. A New Approach to Scalable Linda-systems Based on Swarms. In *SAC 2003*, pages 375–379. ACM, 2003.

doi:10.1145/952532.952607.

(Page 141.)

[NCY15] Nicholas Ng, Jose Coutinho, and Nobuko Yoshida. Protocols by Default: Safe MPI Code Generation based on Session Types. In *CC 2015*, volume 9031 of *LNCS*, pages 212–232. Springer, 2015.

doi:10.1007/978-3-662-46663-6_11.

(Page 294.)

[NY14] Nicholas Ng and Nobuko Yoshida. Pabble: Parameterised Scribble for Parallel Programming. In *PDP 2014*, pages 707–714. IEEE, 2014.

doi:10.1109/PDP.2014.20.

(Page 294.)

[PA98] George Papadopoulos and Farhad Arbab. Coordination Models and Languages. *Advances in Computers*, 46:329–400, 1998.

doi:10.1016/S0065-2458(08)60208-9.

(Page 141.)

[Par72] David Parnas. On the Criteria To Be Used in Decomposing Systems into Modules. *Communications of the ACM*, 15(12):1053–1058, 1972.

doi:10.1145/361598.361623.

(Page 14.)

[PC13a] José Proença and Dave Clarke. Data Abstraction in Coordination Constraints. In *Advances in Service-Oriented and Cloud Computing (FOCLASA 2013)*, volume 393 of *CCIS*, pages 159–173. Springer, 2013.

doi:10.1007/978-3-642-45364-9_14.

(Pages 38, 99, and 220.)

[PC13b] José Proença and Dave Clarke. Interactive Interaction Constraints. In *COORDINATION 2013*, volume 7890 of *LNCS*, pages 211–225. Springer, 2013.

doi:10.1007/978-3-642-38493-6_15.

(Pages 38, 99, and 220.)

[PCdVA11] José Proença, Dave Clarke, Erik de Vink, and Farhad Arbab. Decoupled execution of synchronous coordination models via behavioural automata. In *FOCLASA 2011*, volume 58 of *EPTCS*, pages 65–79. CoRR, 2011.

doi:10.4204/EPTCS.58.5.

(Pages 99, 137, and 279.)

[PCdVA12] José Proença, Dave Clarke, Erik de Vink, and Farhad Arbab. Dreams: a framework for distributed synchronous coordination. In *SAC 2012*, pages 1510–1515. ACM, 2012.

doi:10.1145/2245276.2232017.

(Pages 99, 137, and 279.)

[Pos14] Raphael Poss. Multicore architectures and their software landscape. In *Computer Handbook*, chapter 24, pages 24·1–24·17. CRC Press, 3rd edition, 2014.

(Pages 1 and 8.)

[Pro11] José Proença. *Synchronous Coordination of Distributed Components*. PhD thesis, Universiteit Leiden, 2011.

hdl:1887/17624.

(Pages 38, 99, 137, and 279.)

[PSAB12] Bahman Pourvatan, Marjan Sirjani, Farhad Arbab, and Marcello Bonsangue. Decomposition of Constraint Automata. In *FACS 2010*, volume 6921 of *LNCS*, pages 237–258. Springer, 2012.

doi:10.1007/978-3-642-27269-1_14.

(Pages 38, 48, and 140.)

[PSHA12] Bahman Pourvatan, Marjan Sirjani, Hossein Hojjat, and Farhad Arbab. Symbolic execution of Reo circuits using constraint automata. *Science of Computer Programming*, 77(7–8):848–869, 2012.

doi:10.1016/j.scico.2011.04.001.

(Pages 38 and 48.)

[Rau10a] Wolfgang Rautenberg. First-Order Logic. In *A Concise Introduction to Mathematical Logic*, Universitext, chapter 2, pages 41–90. Springer, 3rd edition, 2010.

doi:10.1007/978-1-4419-1221-3_2.

(Pages 32, 33, 36, 176, and 254.)

[Rau10b] Wolfgang Rautenberg. Propositional Logic. In *A Concise Introduction to Mathematical Logic*, Universitext, chapter 1, pages 1–40. Springer, 3rd edition, 2010.

doi:10.1007/978-1-4419-1221-3_1.

(Page 177.)

[Rei85] Wolfgang Reisig. Introductory Examples and Basic Definitions. In *Petri Nets: An Introduction*, volume 4 of *EATCS Monographs on Theoretical Computer Science*, chapter 1, pages 3–16. Springer, 1985.

(Page 33.)

[Rei07] James Reinders. Why Threading Building Blocks? In *Intel Threading Building Blocks*, chapter 1, pages 1–6. O'Reilly, 2007.

(Pages 5 and 275.)

[Rob13] Arch Robison. Composable Parallel Patterns with Intel Cilk Plus. *Computing in Science and Engineering*, 15(2):66–71, 2013.

doi:10.1109/MCSE.2013.21.

(Pages 5 and 275.)

[RW96] Antony Rowstron and Alan Wood. An Efficient Distributed Tuple Space Implementation for Networks of Workstations. In *Euro-Par 1996*, volume 1123 of *LNCS*, pages 510–513. Springer, 1996.

doi:10.1007/3-540-61626-8_69.

(Page 141.)

[Seg95] Roberto Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, Massachusetts Institute of Technology, 1995.

(Page 147.)

[SGG13] Abraham Silberschatz, Peter Galvin, and Greg Gagne. Threads. In *Operating System Concepts*, chapter 4, pages 163–201. Wiley, 9th edition, 2013.

(Page 8.)

[SJBA06] Marjan Sirjani, Mohammad-Mahdi Jaghoori, Christel Baier, and Farhad Arbab. Compositional Semantics of an Actor-Based Language Using Constraint Automata. In *COORDINATION 2006*, volume 4038 of *LNCS*, pages 281–297. Springer, 2006.

doi:10.1007/11767954_18.

(Page 65.)

[SO93] Andreas Stolcke and Stephen Omohundro. Hidden Markov Model Induction by Bayesian Model Merging. In *NIPS 1992*, pages 11–18. Morgan-Kaufmann, 1993.

(Page 247.)

[ST97] Nir Shavit and Dan Touitou. Software transactional memory. *Distributed Computing*, 10(2):99–116, 1997.

doi:10.1007/s004460050028.

(Pages 16 and 276.)

[Sut05] Herb Sutter. The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software. *Dr. Dobbs Journal*, 30(3), 2005.

(Page 1.)

[TDJ13] Samira Tasharofi, Peter Dinges, and Ralph Johnson. Why Do Scala Developers Mix the Actor Model with other Concurrency Models? In *ECOOP 2013*, volume 7920 of *LNCS*, pages 302–326. Springer, 2013.

doi:10.1007/978-3-642-39038-8_13.

(Pages 9 and 275.)

[TVMS08] Samira Tasharofi, Mohsen Vakilian, Roshanak Zilouchian Moghaddam, and Marjan Sirjani. Modeling Web Service Interactions Using the Coordination Language Reo. In *WS-FM 2007*, volume 4937 of *LNCS*, pages 108–123. Springer, 2008.

doi:10.1007/978-3-540-79230-7_8.

(Page 60.)

[Vaj11] András Vajda. Practical Many-Core Programming. In *Programming Many-Core Chips*, chapter 9, pages 175–211. Springer, 2011.

doi:10.1007/978-1-4419-9739-5_9.

(Page 8.)

[vDKV00] Arie van Deursen, Paul Klint, and Joost Visser. Domain-Specific Languages: An Annotated Bibliography. *ACM SIGPLAN Notices*, 35(6):26–36, 2000.

doi:10.1145/352029.352035.

(Page 17.)

[vdN15] Mathijs van de Nes. In preparation. Master's thesis, Universiteit Leiden, 2015.

(Pages 20, 83, and 291.)

[Vra97] Jos Vrancken. The algebra of communicating processes with empty process. *Theoretical Computer Science*, 177(2):287–328, 1997.

doi:10.1016/S0304-3975(96)00250-2.

(Page 147.)

[WM95] William Wulf and Sally McKee. Hitting the Memory Wall: Implications of the Obvious. *ACM SIGARCH Computer Architecture News*, 23(1):20–24, 1995.

doi:10.1145/216585.216588.

(Page 1.)

[WMLF98] Peter Wyckoff, Stephen McLaughry, Tobin Lehman, and Daniel Ford. T Spaces. *IBM Systems Journal*, 37(3):454–474, 1998.

doi:10.1147/sj.373.0454.

(Page 140.)

[Woe92] Gerhard Woeginger. The complexity of finding arborescences in hypergraphs. *Information Processing Letters*, 44(3):161–164, 1992.

doi:10.1016/0020-0190(92)90057-3.

(Page 216.)

[Yeh93] Wei-Jen Yeh. *Controlling State Explosion in Reachability Analysis*. PhD thesis, Purdue University, 1993.

(Page 148.)

[YMG14] Leonid Yavits, Amir Morad, and Ran Ginosar. The effect of communication and synchronization on AmdahlâĂŹs law in multicore systems. *Parallel Computing*, 40(1):1–16, 2014.

doi:10.1016/j.parco.2013.11.001.

(Pages 3 and 275.)

Index

Titles in the IPA Dissertation Series since 2013

H. Beohar. *Refinement of Communication and States in Models of Embedded Systems*. Faculty of Mathematics and Computer Science, TU/e. 2013-01

G. Igna. *Performance Analysis of Real-Time Task Systems using Timed Automata*. Faculty of Science, Mathematics and Computer Science, RU. 2013-02

E. Zambon. *Abstract Graph Transformation – Theory and Practice*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2013-03

B. Lijnse. *TOP to the Rescue – Task-Oriented Programming for Incident Response Applications*. Faculty of Science, Mathematics and Computer Science, RU. 2013-04

G.T. de Koning Gans. *Outsmarting Smart Cards*. Faculty of Science, Mathematics and Computer Science, RU. 2013-05

M.S. Greiler. *Test Suite Comprehension for Modular and Dynamic Systems*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2013-06

L.E. Mamane. *Interactive mathematical documents: creation and presentation*. Faculty of Science, Mathematics and Computer Science, RU. 2013-07

M.M.H.P. van den Heuvel. *Composition and synchronization of real-time components upon one processor*. Faculty of Mathematics and Computer Science, TU/e. 2013-08

J. Businge. *Co-evolution of the Eclipse Framework and its Third-party Plugins*. Faculty of Mathematics and Computer Science, TU/e. 2013-09

S. van der Burg. *A Reference Architecture for Distributed Software Deployment*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2013-10

J.J.A. Keiren. *Advanced Reduction Techniques for Model Checking*. Faculty of Mathematics and Computer Science, TU/e. 2013-11

D.H.P. Gerrits. *Pushing and Pulling: Computing push plans for disk-shaped robots, and dynamic labelings for moving points*. Faculty of Mathematics and Computer Science, TU/e. 2013-12

M. Timmer. *Efficient Modelling, Generation and Analysis of Markov Automata*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2013-13

M.J.M. Roeloffzen. *Kinetic Data Structures in the Black-Box Model*. Faculty of Mathematics and Computer Science, TU/e. 2013-14

L. Lensink. *Applying Formal Methods in Software Development*. Faculty of Science, Mathematics and Computer Science, RU. 2013-15

C. Tankink. *Documentation and Formal Mathematics — Web Technology meets Proof Assistants*. Faculty of Science, Mathematics and Computer Science, RU. 2013-16

C. de Gouw. *Combining Monitoring with Run-time Assertion Checking*. Faculty of Mathematics and Natural Sciences, UL. 2013-17

J. van den Bos. *Gathering Evidence: Model-Driven Software Engineering in Automated Digital Forensics*. Faculty of Science, UvA. 2014-01

D. Hadziosmanovic. *The Process Matters: Cyber Security in Industrial Control Systems*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2014-02

A.J.P. Jeckmans. *Cryptographically-Enhanced Privacy for Recommender Systems*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2014-03

C.-P. Bezemer. *Performance Optimization of Multi-Tenant Software Systems*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2014-04

T.M. Ngo. *Qualitative and Quantitative Information Flow Analysis for Multi-threaded Programs*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2014-05

A.W. Laarman. *Scalable Multi-Core Model Checking*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2014-06

J. Winter. *Coalgebraic Characterizations of Automata-Theoretic Classes*. Faculty of Science, Mathematics and Computer Science, RU. 2014-07

W. Meulemans. *Similarity Measures and Algorithms for Cartographic Schematization*. Faculty of Mathematics and Computer Science, TU/e. 2014-08

A.F.E. Belinfante. *JTorX: Exploring Model-Based Testing*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2014-09

A.P. van der Meer. *Domain Specific Languages and their Type Systems*. Faculty of Mathematics and Computer Science, TU/e. 2014-10

B.N. Vasilescu. *Social Aspects of Collaboration in Online Software Communities*. Faculty of Mathematics and Computer Science, TU/e. 2014-11

F.D. Aarts. *Tomte: Bridging the Gap between Active Learning and Real-World Systems*. Faculty of Science, Mathematics and Computer Science, RU. 2014-12

N. Noroozi. *Improving Input-Output Conformance Testing Theories*. Faculty of Mathematics and Computer Science, TU/e. 2014-13

M. Helvensteijn. *Abstract Delta Modeling: Software Product Lines and Beyond*. Faculty of Mathematics and Natural Sciences, UL. 2014-14

P. Vullers. *Efficient Implementations of Attribute-based Credentials on Smart Cards*. Faculty of Science, Mathematics and Computer Science, RU. 2014-15

F.W. Takes. *Algorithms for Analyzing and Mining Real-World Graphs*. Faculty of Mathematics and Natural Sciences, UL. 2014-16

M.P. Schraagen. *Aspects of Record Linkage*. Faculty of Mathematics and Natural Sciences, UL. 2014-17

G. Alpár. *Attribute-Based Identity Management: Bridging the Cryptographic Design of ABCs with the Real World*. Faculty of Science, Mathematics and Computer Science, RU. 2015-01

A.J. van der Ploeg. *Efficient Abstractions for Visualization and Interaction*. Faculty of Science, UvA. 2015-02

R.J.M. Theunissen. *Supervisory Control in Health Care Systems*. Faculty of Mechanical Engineering, TU/e. 2015-03

T.V. Bui. *A Software Architecture for Body Area Sensor Networks: Flexibility and Trustworthiness*. Faculty of Mathematics and Computer Science, TU/e. 2015-04

A. Guzzi. *Supporting Developers' Teamwork from within the IDE*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2015-05

T. Espinha. *Web Service Growing Pains: Understanding Services and Their Clients*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2015-06

S. Dietzel. *Resilient In-network Aggregation for Vehicular Networks*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2015-07

E. Costante. *Privacy throughout the Data Cycle*. Faculty of Mathematics and Computer Science, TU/e. 2015-08

S. Cranen. *Getting the point — Obtaining and understanding fixpoints in model checking*. Faculty of Mathematics and Computer Science, TU/e. 2015-09

R. Verdult. *The (in)security of proprietary cryptography*. Faculty of Science, Mathematics and Computer Science, RU. 2015-10

J.E.J. de Ruiter. *Lessons learned in the analysis of the EMV and TLS security protocols*. Faculty of Science, Mathematics and Computer Science, RU. 2015-11

Y. Dajsuren. *On the Design of an Architecture Framework and Quality Evaluation for Automotive Software Systems*. Faculty of Mathematics and Computer Science, TU/e. 2015-12

J. Bransen. *On the Incremental Evaluation of Higher-Order Attribute Grammars*. Faculty of Science, UU. 2015-13

S. Picek. *Applications of Evolutionary Computation to Cryptology*. Faculty of Science, Mathematics and Computer Science, RU. 2015-14

C. Chen. *Automated Fault Localization for Service-Oriented Software Systems*. Faculty of Electrical Engineering, Mathematics, and Computer Science, TUD. 2015-15

S. te Brinke. *Developing Energy-Aware Software*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2015-16

R.W.J. Kersten. *Software Analysis Methods for Resource-Sensitive Systems*. Faculty of Science, Mathematics and Computer Science, RU. 2015-17

J.C. Rot. *Enhanced coinduction*. Faculty of Mathematics and Natural Sciences, UL. 2015-18

M. Stolikj. *Building Blocks for the Internet of Things*. Faculty of Mathematics and Computer Science, TU/e. 2015-19

D. Gebler. *Robust SOS Specifications of Probabilistic Processes*. Faculty of Sciences, Department of Computer Science, VUA. 2015-20

M. Zaharieva-Stojanovski. *Closer to Reliable Software: Verifying functional behaviour of concurrent programs*. Faculty of Electrical Engineering, Mathematics & Computer Science, UT. 2015-21

R.J. Krebbers. *The C standard formalized in Coq*. Faculty of Science, Mathematics and Computer Science, RU. 2015-22

R. van Vliet. *DNA Expressions – A Formal Notation for DNA*. Faculty of Mathematics and Natural Sciences, UL. 2015-23

S.-S.T.Q. Jongmans. *Automata-Theoretic Protocol Programming*. Faculty of Mathematics and Natural Sciences, UL. 2016-01