Bibliography file of SCOPE project (papers related to code generation from statecharts)

Andrzej Wąsowski

IT University of Copenhagen

Following references were used during the project on compilation of statecharts (SCOPE). I have studied most of them. Some I was not able to reach unfortunately.

Only some papers on the list directly discusses statechart-like languages and even fewer is concerned with the problem of code synthesis from model description. Hopefully I will be able to filter those to separate list once the project is over.

References

1. *ACM SIGPLAN Languages, Compilers, and Tools for Embedded Systems (LCTES)*, Washington DC, USA, June 2004. ACM Press.
2. H. Abelson and G. Sussman. *Structure and Interpretation of Computer Programs*. MIT Press, Cambridge, Mass., 1985.
3. C. Ackad. Software synthesis from statechart models for real time systems. In F. Rammig, editor, *International Workshop on Distributed and Parallel Embedded Systems (DIPES'98)*, pages 73–81, Paderborn,Germany, 1998. Kluwer Academic Publishers.
4. S. S. Alhir. *UML in a Nutshell*. O'Reilly & Associates, Inc., 101 Moris Street, Sebastopol, CA 95472, 1998.
5. J. Ali and J. Tanaka. Converting statecharts into Java code. In *5th International Conference on Integrated Design and Process Technology (IDPT'99)*, Dallas,Texas, June 1999.
6. S. Alstrup and T. Rauhe. Improved labeling schemes for ancestor queries. Technical Report TR 2001-5, IT University of Copenhagen, Denmark, Aug. 2001.
7. R. Alur, S. Kannan, and M. Yannakakis. Communicating hierarchical state machines. In J. Wiedermann, P. van Emde Boas, and M. Nielsen, editors, *26th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1644 of *Lecture Notes in Computer Science*, pages 169–178, Prague, Czech Republic, July 1999. Springer-Verlag.
8. R. Alur and M. Yannakakis. Model checking of hierarchical state machines. *ACM Transactions on Programming Languages and Systems*, 23(3):273–303, May 2001.
9. P. Amagbégnon, L. Besnard, and P. L. Guernic. Implementation of the data-flow synchronous language SIGNAL. In *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, pages 163–173. ACM Press, 1995.
10. T. Amnell, E. Fersman, P. Pettersson, and W. Yi. Automatic code synthesis for timed automata. In *NWPT'01 – The 13th Nordic Workshop on Programming Theory*, Lyngby, Denmark, Oct. 2001.
11. C. André. SyncCharts: A visual representation of reactive behaviors. Technical Report TR 95-52, I3S, Sophia-Antipolis, France, Oct. 1995.
12. C. André. Representation and analysis of reactive behaviors: A synchronous approach. In *Proceedings of Computational Engineering of Systems Applications (CESA)*, pages 19–29, Lille, France, July 1996. IEEE-SMC.
13. C. André. Representation and analysis of reactive behaviours: A synchronous approach. Technical Report TR 96-28, Laboratoire Informatique, Signaux, Systèmes (I3S), Nice, France, 1996.
14. C. André, F. Boulanger, and A. Girault. Software implementation of synchronous programs. In *2nd International Conference on Application of Concurrency to System Design (ACSD)*, pages 133–142, Newcastle upon Tyne, UK, June 2001. IEEE Computer Society Press.
15. C. André and M.-A. Péraldi. Effective implementation of ESTEREL programs. In *5th EUROMICRO Workshop on Real-Time Systems*, pages 262–267, Oulu, Finland, June 1993. IEEE Computer Society Press.
16. A. W. Appel and G. J. Jacobson. The world's fastest scrabble program. *Communications of the ACM*, 31(5):572–578.585, 1988.
17. J.-R. Beauvais, T. Gautier, P. L. Guernic, R. Houdebine, and E. Rutten. A translation of statecharts into signal. In *International Conference on Application of Concurrency to System Design (ACSD'98)*, Aizu-Wakamatsu, Japan, Mar. 1998.
18. J.-R. Beauvais, R. Houdebine, P. L. Guernic, E. Rutten, and T. Gautier. A translation of Statecharts and Activitycharts into Signal equations. Technical Report 1182, Institut de Recherche en Informatique et Systemes Aléatoires (IRISA), Rennes Cedex, France, May 1998.
19. G. Behrmann, K. Kristoffersen, and K. G.Larsen. Code generation for hierarchical systems. In

NWPT'99 – The 11th Nordic Workshop on Programming Theory, Uppsala, Sweden, Sept. 1999.
20. G. Behrmann, K. G. Larsen, H. R. Andersen, H. Hulgaard, and J. Lind Nielsen. Verification of hierarchical state/event systems using reusability and compositionality. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, volume 1579 of *Lecture Notes in Computer Science*, pages 163–177, Amsterdam, The Netherlands, Mar. 1999. Springer-Verlag.
21. G. Behrmann, K. G. Larsen, H. R. Andersen, H. Hulgaard, and J. Lind Nielsen. Verification of hierarchical state/event systems using reusability and compositionality. *Formal Methods in System Design*, 21(2):225–244, 2002.
22. A. Benveniste, P. le Guernic, and C. Jacquemot. Programming with events and relations: the Signal language and its semantics. *Science of Computer Programming*, 16:103–149, 1991.
23. B. Bérard, M. Bidoit, A. Finkel, F. Laroussinie, A. Petit, L. Petrucci, P. Schnoebelen, and P. McKenzie. *Systems and Software Verification. Model-Checking Techniques and Tools*. Springer-Verlag, Berlin-Heidelberg, 2001.
24. G. Berry. The Esterel v5 language primer. version v5_91, July 2000.
25. G. Berry. The foundations of Esterel. In G. Plotkin, C. Stirling, and M. Tofte, editors, *Proof, Language and Interaction. Essays in Honour of Robin Milner*, Foundations of Computing Series, pages 425–454. The MIT Press, Cambridge, Massachusetts, 2000.
26. R. V. Binder. *Testing Object-Oriented Systems. Models, Patterns and Tools*. Addison-Wesley, 2000.
27. D. Björklund, J. Lilius, and I. Porres. Towards efficient code synthesis from statecharts. In A. Evans, R. France, and A. M. B. Rumpe, editors, *Practical UML-Based Rigorous Development Methods - Countering or Integrating the eXtremists. Workshop of the pUML-Group.*, Lecture Notes in Informatics P-7, Toronto,Canada, Oct. 2001. GI.
28. K. Bogdanov. *Automated testing of Harel's statecharts*. PhD thesis, The University of Sheffield, Jan. 2000.
29. K. Bogdanov and M. Holcombe. Statechart testing method for aircraft control systems. *Software Testing, Verification and Reliability*, 1(11):39–54, 2001.
30. K. Bogdanov and M. Holcombe. Properties of concurrently taken transitions of Harel statecharts. In *Workshop on Semantic Foundations of Engineering Design Languages (SFEDL)*, Grenoble, France, Apr. 2002.
31. G. W. Bond, F. Ivancic, N. Klarlund, and R. Trefler. Eclipse feature logic analysis. In *2nd IP-Telephony Workshop*, pages 100–107, New York City, USA, Apr. 2001.
32. G. Booch, J. Rumbaugh, and I. Jacobson. *The Unified Modelling Language: User Guide*. Addison-Wesley, 1999.
33. E. Brinksma and K. G. Larsen, editors. *14th International Conference on Computer Aided Verification (CAV)*, volume 2404 of *Lecture Notes in Computer Science*, Copenhagen, Denmark, July 2002. Springer-Verlag.
34. R. E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, 35(8):677–691, Aug. 1986.
35. M. A. Bulyonkov. Polyvariant Mixed Computation for Analyzer Programs. *Acta Informatica*, 21(5):473–484, Dec. 1984.
36. P. Caspi, D. Pilaud, N. Halbwachs, and J. Plaice. Lustre: a declarative language for programming synchronous systems. In *14th ACM Conference on Principles of Programming Languages (POPL)*, Munich, Jan. 1987. ACM Press.
37. M. Chiodo, P. Giusto, A. Jurecska, L. Lavagno, H. Hsieh, K. Suzuki, A. Sangiovanni-Vincentelli, and E. Sentovich. Synthesis of software programs for embedded control applications. In *32nd ACM/IEEE Design Automation Conference*, 1995.
38. E. M. Clarke. *Model Checking*. The MIT Press, Dec. 1999.
39. E. M. Clarke and W. Heinle. Modular translation of statecharts to SMV. Technical Report CMU-CS-00-XXX, School of Computer Science, Carnegie Mellon University, Apr. 2000.
40. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press, 2nd edition, 2001.
41. K. Czarnecki, T. Bednasch, P. Unger, and U. Eisenecker. Generative programming for embedded software: An industrial experience report. In D. Batory, C. Consel, and W. Taha, editors, *Generative Programming and Component Engineering (GPCE)*, volume 2487 of *Lecture Notes in Computer Science*, pages 156–172, Pittsburgh, PA, USA, Oct. 2002. Springer-Verlag.
42. K. Czarnecki and U. W. Eisenecker. *Generative Programming: Methods, Tools, and Applications*. Addison-Wesley, 2000.
43. J. Daciuk. Experiments with automata compression. *Lecture Notes in Computer Science*, 2088:105–112, 2001.
44. D. Damian and O. Danvy. Static transition compression. In *Second International Workshop on Semantics, Applications, and Implementation of Program Generation (SAIG)*, volume 2196 of *Lecture Notes in Computer Science*, pages 92–107, Florence, Italy, Sept. 2001. Springer-Verlag.
45. O. Danvy, R. Glück, and P. Thiemann, editors. *Partial Evaluation*, volume 1110 of *Lecture Notes in Computer Science*, Dagstuhl Castle, Germany, Feb. 1996. Springer-Verlag.

46. A. David and M. O. Möller. From HUppaal to Uppaal. A translation from hierarchical timed automata to flat timed automata. Technical Report RS-01-11, BRICS, Mar. 2001.
47. A. David, M. O. Möller, and W. Yi. Formal verification of UML statecharts with real-time extensions. In R.-D. Kutsche and H. Weber, editors, *Fundamental Approaches to Software Engineering (FASE)*, volume 2306 of *Lecture Notes in Computer Science*, pages 218–232, Grenoble, France, Apr. 2002. Springer-Verlag.
48. A. David, M. O. Möller, and W. Yi. Verification of UML statechart with real-time extensions. Technical Report 2003-19, Department of Information Technology, Uppsala University, Feb. 2003.
49. L. de Alfaro and T. A. Henzinger. Interface automata. In *Proceedings of the Ninth Annual Symposium on Foundations of Software Engineering (FSE)*, pages 109–120, Vienna, Austria, Sept. 2001. ACM Press.
50. G. de Micheli, R. K. Brayton, and A. Sangiovanni-Vincentelli. Optimal state assignment for finite state machines. *IEEE Transactions on Computer-Aided Design*, CAD-4(3):269–285, 1985.
51. V. Diekert and Y. Métivier. Partial commutation and traces. In Rozenberg and Salomaa [151], pages 457–533.
52. K. Diethers, U. Goltz, and M. Huhn. Model checking UML statecharts with time. In Jürjens et al. [101], pages 35–51. TUM-I0208.
53. D. Drusinsky. *On Synchronized Statecharts*. PhD thesis, Department of Applied Mathematics, Weizmann Institute of Science, Rehovot, Israel, 1988.
54. D. Drusinsky and D. Harel. On the power of cooperative concurrency. In *Proceedings of Concurrency '88*, volume 335 of *Lecture Notes in Computer Science*, pages 74–103, New York, 1985. Springer-Verlag.
55. D. Drusinsky and D. Harel. Using statecharts for hardware description and synthesis. *IEEE Transactions on Computer-Aided Design*, 8(7):798–807, 1989.
56. D. Drusinsky and D. Harel. On the power of bounded concurrency I: Finite automata. *Journal of ACM*, 41(3):517–539, May 1994.
57. D. Drusinsky-Yoresh. Symbolic cover minimization of fully I/O specified finite state machines. *IEEE Transactions on Computer-Aided Design*, 9(7):779–781, 1990.
58. D. Drusinsky-Yoresh. Decision problems for interacting finite state machines. *IEEE Transactions on Computer-Aided Design*, 10(12):1576–1579, 1991.
59. D. Drusinsky-Yoresh. A state assignment procedure for single-block implementation of state charts. *IEEE Transactions on Computer-Aided Design*, 10(12):1569–1576, 1991.
60. G. Engels, R. Heckel, and J. M. Küster. Rule-based specification of behavioral consistency based on the UML meta-model. In Gogolla and Kobryn [71], pages 272–286.
61. J. Erns, W. Evans, C. W. Fraser, S. Lucco, and T. A. Proebsting. Code compression. *Proceedings of the ACM SIGPLAN'97*, 6:358–365, 1997.
62. E. Erpenbach. *Compilation, Worst-Case Execution Times and Schedulability Analysis of Statecharts Models*. PhD thesis, Department of Mathematics and Computer Science of the University of Paderborn, Apr. 2000.
63. E. Erpenbach and P. Altenbernd. Worst-case execution times and schedulability analysis of statecharts models. In *11th Euromicro Conference on Real Time Systems*, York, 1999.
64. E. Erpenbach and F. S. J. Stroop. Compilation and timing analysis of statecharts models for embedded systems. In *In The Second International Workshop on Compiler and Architecture Support for Embedded Systems (CASES)*, Washington, D.C, USA, Oct. 1999.
65. E. Erpenbach, J. Stroop, and F. J. Rammig. On the compilation of statecharts models into target code. In *IEEE International Symposium on Computer Aided Control System Design Kohala Coast-Island of Hawai'i, USA*, pages 249–254. IEEE Computer Society Press, 1999.
66. R. Eshuis and R. Wieringa. Requirements level semantics for UML statecharts. In S. F. Smith and C. L. Talcott, editors, *Formal Methods for Open Object-Based Distributed Systems IV - Proc. FMOODS'2000, September, 2000, Stanford, California, USA*. Kluwer Academic Publishers, 2000.
67. S. Etalle and M. Gabbrieli. Partial evaluation of concurrent constraint languages. *ACM Computing Surveys*, 30(3es), Sept. 1998.
68. C. Fischer and R. LeBlanc. *Crafting a Compiler*. Addison-Wesley, 2nd edition, New revised edition is coming soon (early 2002).
69. J. Flinn, E. de Lara, M. Satyanarayanan, D. S. Wallach, and W. Zwaenepoel. Reducing the energy usage of office applications. *Lecture Notes in Computer Science*, 2218:252–272, 2001.
70. K. Friis Larsen and J. Lichtenberg. MuDDy 2.0 – SML interface to the binary decision diagrams package BuDDy. `http://www.itu.dk/research/muddy`.
71. M. Gogolla and C. Kobryn, editors. *4th International UML Conference – The Unified Modeling Language, Modeling Languages, Concepts, and Tools*, volume 2185 of *Lecture Notes in Computer Science*, Toronto, Canada, Oct. 2001. Springer-Verlag.
72. R. Gray. A generator for lexical analyzers that programmers can use. In *USENIX Conference*, pages 147–160, June 1988.

73. N. Halbwachs, P. Caspi, P. Raymond, and D. Pilaud. The synchronous data flow programming language LUSTRE. *Proceedings of IEEE*, 79(9):1305–1320, Sept. 1999.
74. D. Harel. Statecharts: A visual formalism for complex systems. *Science of Computer Programming*, 8:231–274, 1987.
75. D. Harel. Some thoughts on statecharts, 13 years later. In O. Grumberg, editor, *9th International Conference on Computer Aided Verification (CAV)*, volume 1254 of *Lecture Notes in Computer Science*, pages 226–231. Springer-Verlag, 1997.
76. D. Harel and E. Gery. Executable object modeling with statecharts. *IEEE Computer*, 30(7):31–42, 1997.
77. D. Harel and O. Kupferman. On object systems and behavioral inheritance. *IEEE Transactions on Software Engineering*, 28(9):889–903, Sept. 2002.
78. D. Harel and A. Naamad. The STATEMATE semantics of statecharts. *ACM Transactions on Software Engineering and Methodology*, 5(4):293–333, 1996.
79. D. Harel and A. Pnueli. On the development of reactive systems. In K. R. Apt, editor, *Logic and Model of Concurrent Systems*, volume 13 of *NATO ASI*, pages 477–498. Springer-Verlag, Oct. 1985.
80. D. Harel, A. Pnueli, J.P.Schmidt, and R.Sherman. On the formal semantics of statecharts. In *2nd IEEE Symposium on Logic in Computer Science*, pages 396–406, New York, 1988. IEEE Computer Society Press.
81. D. Harel and B. Rumpe. Modeling languages: Syntax, semantics and all that stuff. Technical report, The Weizmann Institute of Science, Rehovot, Israel, MCS00-16, 2000.
82. J. Hatcliff, T. A. Mogensen, and P. Thiemann, editors. *Partial Evaluation: Practice and Theory. DIKU 1998 International Summer School*, volume 1706 of *Lecture Notes in Computer Science*. Springer-Verlag, Copenhagen, Denmark, 1999.
83. T. A. Henzinger, B. Horowitz, and C. M. Kirsch. Giotto: A time-triggered language for embedded programming. In Henzinger and Kirsch [84], pages 166–184.
84. T. A. Henzinger and C. M. Kirsch, editors. *First International Workshop on Embedded Software (EMSOFT)*, volume 2211 of *Lecture Notes in Computer Science*, Tahoe City, CA, USA, Oct. 2001. Springer-Verlag.
85. T. A. Henzinger and C. M. Kirsch. The embedded machine: Predictable, portable real-time code. In *ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*, pages 315–326, Berlin,Germany, 2002.
86. C. Hoare. *Communicating Sequential Processes*. International Series in Computer Science. Prentice Hall, 1985.
87. J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 2nd edition, 2001.
88. J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
89. C. Huizing and W. de Roever. Introduction to design choices in the semantics of Statecharts. *Information Processing Letters*, 37:205–213, 1991.
90. I-Logix Inc. Rhapsody® in MicroC. http://www.ilogix.com.
91. IAR Inc. IAR visualSTATE®. `http://www.iar.com/Products/VS/`.
92. IAR Inc. IAR visualSTATE® concept guide, 1999. Part of visualSTATE standard documentation.
93. IAR Inc. IAR visualSTATE® reference guide, Dec. 2000. Part of visualSTATE standard documentation.
94. IAR Inc. IAR visualSTATE® user guide, Oct. 2000. Part of visualSTATE standard documentation.
95. IEEE. *Fifteenth IEEE International Conference on Automated Software Engineering (ASE)*, Grenoble, France, Sept. 2000. IEEE Computer Society Press.
96. International standard. Programming Languages - C. Ref. ISO/IEC 9899:1999(E).
97. P. Jacobsen. Code generation for embedded systems. Master's thesis, Technical University of Denmark (Lyngby) and IT University of Copenhagen, Apr. 1999.
98. D. N. Jansen. Probabilistic UML statecharts for specification and verification: a case study. In Jürjens et al. [101]. TUM-I0208.
99. N. D. Jones, C. K. Gomard, and P. Sestoft. *Partial Evaluation and Automatic Program Generation*. International Series in Computer Science. Prentice Hall, 1993. `http://www.dina.kvl.dk/~sestoft/pebook`.
100. J.R. Burch, E.M. Clarke, K.L. McMillan, D.L. Dill, and L.J. Hwang. Symbolic Model Checking: 10^{20} States and Beyond. In *Fifth Annual IEEE Symposium on Logic in Computer Science*, pages 1–33, Washington, D.C., 1990. IEEE Computer Society Press.
101. J. Jürjens, M. V. Cengarle, E. B. Fernandez, B. Rumpe, and R. Sandner, editors. *pUML Group Workshop on Critical Systems Development with UML (CSDUML)*, Dresden, Germany, Sept. 2002. Technical University of Munich. TUM-I0208.
102. S. Kannan, M. Naor, and S. Rudich. Implicit representation of graphs. *SIAM J. DISC. MATH.*, 1992.
103. C. M. Kirsch. Principles of real-time programming. In Sangiovanni-Vincentelli and Sifakis [152], pages 61–75.
104. A. Knapp and S. Merz. Model checking and code generation for UML state machines and collaborations. In D. Haneberg, G. Schellhorn, and W. Reif,

editors, *5th Workshop on Tools for System Design and Verification*, Technical Report 2002-11, pages 59–64. Institut für Informatik, Universität Augsburg, 2002.

105. D. E. Knuth. *The Art of Computer Programming*, volume 1. Addison-Wesley, 3rd. edition, 1997.
106. T. Kowaltowski, C. L. Lucchesi, and J. Stolfi. Minimization of binary automata, 1993.
107. R. Laleau and A. Mammar. An overview of a method and its support tool for generating B specifications from UML notations. In IEEE [95], pages 269–272.
108. K. Lano, K. Androutsopoulos, and D. Clark. Structuring and design of reactive systems using RSDS and B. In T. S. E. Maibaum, editor, *Third International Conference Fundamental Approaches to Software Engineering (FASE)*, volume 1783 of *Lecture Notes in Computer Science*, pages 97–111, Berlin, Germany, 2000. Springer-Verlag.
109. K. Lano, D. Clark, K. Androutsopoulos, and P. Kan. Invariant-based synthesis of fault-tolerant systems. In M. Joseph, editor, *Formal Techniques in Real-Time and Fault-Tolerant Systems, 6th International Symposium (FTRTFT)*, Lecture Notes in Computer Science, pages 46–57, Pune, India, Sept. 2000. Springer-Verlag.
110. K. G. Larsen. *Context Dependent Bisumulation Between Processes*. PhD thesis, Edinburgh University, 1986.
111. M. Larsen. VisualSTATE rule base 4.0 specification (IAR Inc.). Internal Document of IAR Inc., 1998.
112. A.-F. Le Meur and C. Consel. Generic software component configuration via partial evaluation. In *Workshop on Product Line Architecture*, Denver, USA, Aug. 2000.
113. H. Ledang. Automatic translation from UML specifications to B. In *16th IEEE International Conference on Automated Software Engineering (ASE)*, page 436nn, San Diego, CA, USA, Nov. 2001. IEEE Computer Society Press.
114. H. Ledang and J. Souquières. Formalizing UML behavioral diagrams with B. In *Tenth OOPSLA Workshop on Behavioral Semantics: Back to Basics*, Tampa Bay, Florida, USA, Oct. 2001.
115. H. Ledang and J. Souquières. Contributions for modelling UML state-charts in B. In M. J. Butler, L. Petre, and K. Sere, editors, *Third International Conference on Integrated Formal Methods (IFM)*, volume 2335 of *Lecture Notes in Computer Science*, pages 109–127, Turku, Finland, 2002. Springer-Verlag.
116. J. Lilius and I. P. Paltor. The semantics of UML state machines. Technical Report No 273, Turku Centre for Computer Science, Àbo Akademi University, Finland, May 1999.
117. J. Lind Nielsen. BuDDy – a binary decision diagram package version 2.0. http://www.it-c.dk/research/buddy.
118. J. Lind-Nielsen, H. R. Andersen, H. Hulgaard, G. Behrmann, K. Kristoffersen, and K. G. Larsen. Verification of large state/event systems using compositionality and dependency analysis. *Formal Methods in System Design*, 18(1):5–23, Jan. 2001.
119. J. B. Lind Nielsen. *Verification of Large/State Event Systems*. PhD thesis, Technical University of Denmark, Apr. 2000.
120. N. Lynch. I/O automata: A model for discrete event systems. In *Annual Conference on Information Sciences and Systems*, pages 29–38, Princeton University, Princeton, N.J., 1988.
121. N. A. Lynch and M. R. Tuttle. An introduction to input/output automata. Technical Report MIT/LCS/TM-373, The MIT Press, Nov. 1988.
122. J. H. M. Ganapathi, C.N. Fisher. Retargetable compiler code generation. *ACM Computing Surveys*, 14(4), 1982.
123. J. Magee and J. Kramer. *Concurrency. State Models and Java Programs*. John Wiley & Sons, Chichester, 1999.
124. F. Maraninchi. The Argos language: Graphical representation of automata and description of reactive systems. In *IEEE Workshop on Visual Languages*, Kobe, Japan, Oct. 1991.
125. F. Maraninchi and N. Halbwachs. Compiling ARGOS into boolean equations. In *Proc. 4th Int. School and Symposium on Formal Techniques in Real Time and Fault Tolerant Systems (FTRTFT)*, volume 1135 of *Lecture Notes in Computer Science*, Uppsala, Sweden, Sept. 1996. Springer-Verlag.
126. F. Maraninchi and Y. Rémond. Applying formal methods to industrial cases: The language approach (the production-cell and mode-automata). In *5th International Workshop on Formal Methods for Industrial Critical Systems (FMICS)*, Berlin,Germany, Apr. 2000.
127. F. Maraninchi and Y. Rémond. Argos: an automaton-based synchronous language. *Computer Languages*, 27(1–3):61–92, 2001.
128. P. Marwedel. Code generation for embedded processors: An introduction, 1995. Introductory article in book: Code Generation for Embedded Processors.
129. P. Marwedel and G. Goosens. *Code Generation for Embedded Processors*. Kluwer Academic Publishers, Boston/London/Dordrecht, 1995.
130. C. Meinel and C. Stangier. Data structures for boolean functions. BDDs – foundations and application. *Lecture Notes in Computer Science*, 2122:61–78, 2001.
131. M. Mendler, editor. *Workshop on Semantic Foundations of Engineering Design Languages (SFEDL)*. Elsevier Science Publishers, 2004. Accepted. To be published in ENTCS.

132. E. Mikk, Y. Lakhnech, C. Petersohn, and M. Siegel. On formal semantics of statecharts as supported by STATEMATE. In *2nd BCS-FACS Northern Formal Methods Workshop*. Springer-Verlag, 1997.
133. E. Mikk, Y. Lakhnech, and M. Siegel. Hierarchical automata as model for statecharts. In R. K. Shyamasundar and K. Ueda, editors, *Third Asian Computing Science Conference (ASIAN)*, volume 1345 of *Lecture Notes in Computer Science*, pages 181–196, Kathmandu, Nepal, Dec. 1997. Springer-Verlag.
134. E. Mikk, Y. Laknech, M. Siegel, and G. J. Holzmann. Implementing statecharts in PROMELA/SPIN. In *2nd IEEE Workshop on Industrial Strength Formal Specification Techniques*, pages 90–101, Boca Raton, Florida, Oct. 1998. IEEE Computer Society Press.
135. R. Milner. *Communication and Concurrency*. Prentice Hall International Series in Computer Science. Prentice Hall, 1989.
136. J. B. Møller. *Symbolic Model Checking of Real-Time Systems Using Difference Decision Diagrams*. PhD thesis, Department of Innovation, IT University of Copenhagen, Apr. 2002.
137. M. Murakami. Partial evaluation of reactive communciating processes using temporal logic formulas. In *Workshop on Algebraic and Object-Oriented Approaches to Software Science*, 1995.
138. I. A. Niaz and J. Tanaka. Code generation from UML statecharts. In *7th IASTED International Conference on Software Engineering and Applications (SEA 2003)*, Marina del Rey, CA, USA, Nov. 2003.
139. J. Noble and C. Weir. *Small Memory Software. Patterns for systems with limited memory*. Addison-Wesley, 2001.
140. Object Management Group. OMG Unified Modelling Language specification, 1999. `http://www.omg.org`.
141. Object Management Group. Meta object facility (MOF) specification, 2002. `http://www.omg.org`.
142. Object Management Group. OMG XML metadata interchange (XMI) specification, Jan. 2002. http://www.omg.org.
143. D. L. Parnas. On the design and development of program families. *IEEE Transactions on Software Engineering*, Vol. SE-2(No. 1):1–9, Mar. 1976.
144. A. Pnueli and M. Shalev. What is in a step: On the semantics of statecharts. In T. Ito and A. R. Meyer, editors, *International Conference on Theoretical Aspects of Computer Software (TACS)*, volume 526 of *Lecture Notes in Computer Science*, pages 244–264, Sendai, Japan, Sept. Septmeber 24-27, 1991. Springer-Verlag.
145. A. Pnueli, M. Siegel, and O. Shtrichman. The Code Validation Tool (CVT) – Automatic verification of code generated from synchronous languages. In *The International Workshop on Software Tools for Technology Transfer*, volume NS-98-4 of *BRICS notes series*, pages 7–12, Aalborg, Denmark, July 1998. BRICS.
146. S. K. Rajamani and J. Rehof. Conformance checking for models of asynchronous message passing software. In Brinksma and Larsen [33], pages 166–179.
147. S. Ramesh. Efficient translation of statecharts into hardware circuits. In *12th International Conference on VLSI Design*, pages 384–389. IEEE Computer Society Press, Jan. 1999.
148. Rational Software Corp. Rational Rose® Real Time (RoseRT). http://www.rational.com/products/rosert/.
149. M. Riebisch, I. Philippow, and M. Götze. UML-based statistical test case generation. In M. Aksit, M. Mezini, and R. Unland, editors, *Objects, Components, Architectures, Services, and Applications for a Networked World. NetObjectDays, Revised Papers*, volume 2591 of *Lecture Notes in Computer Science*, pages 394–411, Erfurt, Germany, Oct. 2002. Springer-Verlag.
150. E. E. Roubtsova, J. van Katwijk, R. C. M. de Rooij, and H. Toetenel. Transformation of UML specification to XTG. In D. Bjørner, M. Broy, and A. V. Zamulin, editors, *Perspectives of System Informatics (PSI), 4th International Andrei Ershov Memorial Conference, Revised Papers*, volume 2244 of *Lecture Notes in Computer Science*, pages 249–256, Akademgorodok, Novosibirsk, July 2001. Springer-Verlag.
151. G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages*, volume 3. Springer-Verlag, 1997.
152. A. Sangiovanni-Vincentelli and J. Sifakis, editors. *2nd International Conference on Embedded Software (EMSOFT)*, volume 2491 of *Lecture Notes in Computer Science*, Grenoble,France, Oct. 2002. Springer-Verlag.
153. S.A.Seshia, R. Shyamasundar, A. Bhattacharjee, and S. Dhodapkar. A translation of statecharts to Esterel. In J. M. Wing, J. Woodcock, and J. Davies, editors, *FM'99 - Formal Methods, World Congress on Formal Methods in the Development of Computing Systems*, volume 1709 of *Lecture Notes in Computer Science*, pages 983–1007, Toulouse, France, Sept. 1999. Springer-Verlag.
154. Z. Sawa. Equivalence checking of non-flat systems is EXPTIME-hard. In FIXME, editor, *International Conference on Concurrency Theory (CONCUR)*, Lecture Notes in Computer Science, page FIXME, Marseille, France, Sept. 2003. Springer-Verlag.
155. SCOPE: A statechart compiler, 2003. `http://www.mini.pw.edu.pl/~wasowski/scope`.
156. K. Scott and N. Ramsey. When do match-compilation heuristics matter? Technical Report

CS-2000-13, Department of Computer Science, University of Virginia, May 2000.
157. E. Sekerinski. Graphical design of reactive systems. In D. Bert, editor, *Second International B Conference. Recent Advances in the Development and Use of the B Method*, volume 1393 of *Lecture Notes in Computer Science*, pages 182–197, Montpellier, France, Apr. 1998. Springer-Verlag.
158. E. Sekerinski and R. Zurob. iState: A statechart translator. In Gogolla and Kobryn [71], pages 376–390.
159. E. Sekerinski and R. Zurob. Translating statecharts to B. In M. J. Butler, L. Petre, and K. Sere, editors, *Third International Conference on Integrated Formal Methods (IFM)*, Lecture Notes in Computer Science, pages 128–144, Turku, Finland, May 2002. Springer-Verlag.
160. A. J. H. Simons. The compositional properties of UML statechart diagrams. In C. J. van Rijsbergen, editor, *Third Electronic Workshop on Rigorous Object-Oriented Methods*. British Computer Society, 2000.
161. *Software Product Line Conference (SPLC)*, Lecture Notes in Computer Science, Boston, USA, August/September 2004. Springer-Verlag.
162. J. Staunstrup, H. R. Andersen, H. Hulgaard, J. Lind-Nielsen, K. G. Larsen, G. Behrmann, K. J. Kristoffersen, A. Skou, H. Leerberg, and N. B. Theilgaard. Practical verification of embedded software. *IEEE Computer*, 5(33):68–75, 2000.
163. R. E. Tarjan and A. C.-C. Yao. Storing a sparse table. *Communications of the ACM*, 22(11):606–611, 1979.
164. F. Tip. A survey of program slicing techniques. *Journal of Programming Languages*, 3(3):121–189, Sept. 1995.
165. H. Toetenel, E. Roubtsova, and J. van Katwijk. A timed automata semantics for real-time UML specifications. In *IEEE Symposia on Human-Centric Computing Languages and Environments*, pages 88–95, Stresa, Italy, Sept. 2001. IEEE Computer Society Press.
166. M. Tofte. *Compiler Generators - What They Can Do, What They Might Do and What They Probably Never Do*, volume 19. Springer-Verlag, 1990.
167. M. Tofte. Code generation using standard ML. Probably never published and unobtainable, 1991.
168. I. Traoré. An outline of PVS semantics for UML statecharts. *Journal of Universal Computer Science*, 6(11):1088–1108, 2000.
169. A. K. Tsakalidis. Maintaing order in a generalized linked list. *Acta Informatica*, 21(1):101–112, 1984.
170. M. von der Beeck. A comparison of statecharts variants. In *ProCoS: Third International Symposium on Formal Techiniques in Real Time and Fault-Tolerant Systems.*, volume 863 of *Lecture Notes in Computer Science*, pages 128–148. Springer-Verlag, 1994.
171. Y. Wang, J.-P. Talpin, A. Benveniste, and P. L. Guernic. Pre-order semantics of UML statemachines. Technical Report No 3958, INRIA, June 2000.
172. J. Warmer and A. Kleppe. *The Object Constraint Language. Precise Modelling with UML.* Addison-Wesley, 1999.
173. A. Wąsowski. On Efficient Program Synthesis from Statecharts. In *ACM SIGPLAN Languages, Compilers, and Tools for Embedded Systems (LCTES)*, San Diego, USA, June 2003. ACM Press.
174. A. Wąsowski. Automatic generation of program families by model restrictions. In SPLC:2004 [161]. Accepted.
175. A. Wąsowski. Flattening Statecharts without Explosions. In *ACM SIGPLAN Languages, Compilers, and Tools for Embedded Systems (LCTES)* [1].
176. A. Wąsowski. On Succinctness of Hierarchical State Diagrams in Absence of Message Passing. In Mendler [131]. Accepted. To be published in ENTCS.
177. A. Wąsowski and P. Sestoft. Compile-time scope resolution for statecharts transitions. In Jürjens et al. [101], pages 133–145. TUM-I0208.
178. A. Wąsowski and P. Sestoft. On the formal semantics of visualSTATE statecharts. Technical Report TR-2002-19, IT University of Copenhagen, Sept. 2002.
179. C. W.Fraser and A. L. Wendt. Integrating code generation and optimization. *SIGPLAN Notices*, 21(7):242–248, June 1986.
180. N. Wirth. The programming language Oberon. *Software – Practice and Experience*, 18(7):671–690, July 1988.
181. N. Wirth. Embedded systems and real-time programming. In Henzinger and Kirsch [84], pages 486–492.
182. T. Ziadi, J.-M. Jézéquel, and F. Fondement. Product line derivation with UML. In *Software Variability Management Workshop*, University of Groningen Departement of Mathematics and Computing Science, Feb. 2003.
183. A. Zündorf. Rigorous object oriented software development with Fujaba. Unpublished Draft, 2000.