Testing Transition Systems: An Annotated Bibliography

Ed Brinksma and Jan Tretmans

University of Twente *
Faculty of Computer Science, Formal Methods and Tools Group
P.O. Box 217, 7500 AE Enschede, The Netherlands
{brinksma,tretmans}@cs.utwente.nl
http://fmt.cs.utwente.nl

Abstract. Labelled transition system based test theory has made remarkable progress over the past 15 years. From a theoretically interesting approach to the semantics of reactive systems it has developed into a field where testing theory is (slowly) narrowing the gap with testing practice. In particular, new test generation algorithms are being designed that can be used in realistic situations whilst maintaining a sound theoretical basis. In this paper we present an annotated bibliography of labelled transition system based test theory and its applications covering the main developments.

1 Formal Testing Theory

Formal testing theory was introduced by Rocco De Nicola and Matthew Hennessy in their seminal paper [DNH84], further elaborated in [DN87,Hen88]. Their original motivation was to characterise interesting formalisations of the notion of *observable behaviour* for transition systems using an idealised but intuitive formalisation of testing. It contributed to the semantic theory of reactive systems, and was not intended as a theory about actual testing. One of the main behavioural preorders or *implementation relations* of [DNH84], so called *must*-testing, was in fact an alternative characterisation of the standard semantic model for CSP [Hoa85], the *failures* model (at least, in the absence of infinite internal computations). Their approach actually required the formal observability of *deadlock* behaviour. This assumption was further exploited by Phillips in [Phi87], and independently by Langerak [Lan90], to define testing scenarios that allow for the testing of alternative behaviour after the observation of deadlocks, the so called *refusals* model. Abramsky showed in [Abr87] that by the introduction of still stronger assumptions (e.g., the unbounded copying

* Supported by the Dutch Technology Foundation STW under project STW TIF.4111: *Côte de Resyste* – COnformance TEsting of REactive SYSTEms, and by the Netherlands Organization for Scientific Research NWO under the Van Gogh project for French-Dutch cooperation VGP 62-480: *Automatic Generation of Conformance Tests*.

F. Cassez et al. (Eds.): MOVEP 2000, LNCS 2067, pp. 187–195, 2001.

of behaviours under test), even classical behavioural equivalences like *bisimulation* can be characterised by testing scenarios. An overview of the theory of behavioural equivalences and preorders for transition systems, including the testing-based notions and their further extensions, can be found in the classical surveys by Van Glabbeek [Gla90,Gla93].

2 Test Frameworks

To arrive at a useful formal model for actual testing a well-defined framework of basic terminology, concepts and methods is needed. The ISO International Standard on conformance testing [ISO91] has been a very influential, informal source in this respect. A short overview can be found in [Ray87], a more critical a posteriori assessment in [Ray97]. A first attempt to come to a more formal interpretation can be found in [BAL+90], with subsequent elaborations in [TKB92,Tre94, Tre99]. This has led itself to standardisation activity culminating in [ISO96], with related expositions in [CFP94,HST96].

3 Formal Test Generation

The first attempts to use De Nicola-Hennessy testing theory for finding algorithms to derive tests automatically from formal specifications were made by Brinksma in [Bri87,Bri88]. This work, which used the specification language LOTOS [ISO89,BB87] as a defining notation for the transition systems, led to the so called *canonical tester* theory. This approach has led to a whole series of modifications and extensions, the main ones being [PF90,Tre90,Doo91,Dri92,Led92]. Publications on specific test generation algorithms and (in some cases) their implementations can be found in [Eer87,Ald90,Wez90,DAV93,TPB96]. All these approaches assume that the testing process can communicate *synchronously* with the system under test (SUT), i.e., that each communication can be viewed as a joint action of system and tester, as in most process algebras [Mil89,Hoa85, BK85,ISO89].

4 Asynchronous Test Contexts

In practice the assumption of synchronous communication is seldom fulfilled. Initially, the problem was handled by applying existing theory on a transformation of the original formal specification. In this transformation the original transition system is ‘plugged’ into a context of input an output queues that enabled the asynchronous communication of inputs (test stimuli) and outputs (test responses) between tester and SUT [TV92,VTKB93]. In this way the existing theory could simply be ‘lifted’ to all sorts of asynchronous scenarios. The disadvantage of this approach, however, was that one had to have different transformations for each (static) communication interface. Moreover, as the communication queues were in most cases unbounded, they caused infinite state spaces even in the case of finite state specifications, which complicated the test generation problem considerably.

A major step forward was made by interpreting transition systems as descriptions of input/output-automata (I/O-automata) [LT89,Seg93]. In this model the property that test inputs cannot be blocked by the SUT is covered by the assumption of *input enabledness* (the I/O-condition), which says that in each state transitions for all input actions are defined. The impossibility to refuse test outputs is obtained by having the analogous requirement for the tester process. This semantic requirement on SUT and tester allows for a uniform and simplified treatment of whole classes of asynchronous testing scenarios and has related refusal and failure models that are in a certain sense simpler than their synchronous counterparts. These ideas were first explored by Phalippou and Tretmans [Pha94a,Pha94b,Tre96a,Tre96b]. Heerink has subsequently refined these ideas by refining the I/O-condition in the sense that either all or none of the input actions are enabled, allowing for the treatment of *bounded* communication media [HT97,Hee98]. This work also allows for the treatment of multiple *Points of Control and Observation* (PCO) that exist in practical situations. The multiple I/O-paradigm for testing turns out to be a natural saturation point of the whole theory, in the sense that all relevant other implementation relations can be obtained as special cases. An overview of the main ideas can be found in [BHT97].

5 Test Generation Tools

The ideas behind many of the reported test generation algorithms have been tried out in small, academic prototypes, but there are only a few examples of larger test tool developments linked to academic research in general, and transition system based testing in particular. An important point in this respect was the development of the test tool TVEDA [CGPT96]. Although not developed on the basis of a formal theory of testing, most of its underlying principles can be justified in terms of the I/O-theories of [Pha94b,Tre96b]. A more recent test tool that uses algorithms from the domain of model checking is TGV [FJJV96, FJJV97,JM99]. The ideas underlying TVEDA and TGV have been combined into TestComposer which is part of the commercial SDL tool set Object-Geode [KJG99].

In the Dutch *Côte de Resyste* project [STW96] the tool TorX is developed which is the first larger test tool that is completely based on a formal model of conformance testing [BFV⁺99,Tre99]. TorX accepts specifications in LOTOS and Promela; Promela is the input language for the model checker SPIN [Hol91,Spi]. The implementation of TorX uses the state-space exploration algorithms of SPIN for its Promela part [VT98], while for its LOTOS part it relies on the Cæsar-Aldebaran tool set [Gar98].

Among the other tools for formal transition system based test generation are VVT-RT which uses CSP as the input specification language [PS97], and SaMsTaG and Autolink which derive tests from SDL specifications but which have a slightly different and less formal approach [GSDD97,SEK⁺98]. The I/O testing paradigm is also used in hardware validation [HT99].

6 Current Developments

Current theory and tools can generate tests from transition systems based specifications, however, it is difficult to steer the test generation process and to know how much of the specification has been tested. Some of the tools use user-specified *test purposes* to steer the test generation (TGV, AUTOLINK), while others use a random approach (TORX). The problem of finding criteria for how to select the tests with the largest chance of detecting errors is one of the major research issues [Bri93,Pha94b,HT96,CV97,CG97]. Among the other research items of current interest are the representation and treatment of data aspects in test generation, the combination between conformance testing and performance testing, formal verification of test suites [JJM00], and distributed testing [JJKV98]. On the practical side we see increasing activities in applying the techniques and tools to realistic industrial case studies [KVZ98], while also comparison of different algorithms and tools is investigated in an experimental setting [BFV+99, DBRV+00,HFT00].

7 Other Formal Approaches to Testing

There are two other important 'schools' of formal methods based testing. The one with the longest tradition is based on Mealy-machines (also known as the FSM-approach); for overviews see [LY96,Pet00]. The link between this theory and transition systems based testing is studied in [Tan97].

Both the FSM and the transition system based approaches mainly deal with the dynamic aspects of system behaviour. An existing formal approach to testing the static aspects of systems, such as data structures and their operations, uses abstract data type (ADT) theory as its basis, see e.g., [Ber91,Gau95,LGA96] and [Mar95] for a corresponding tool. It is generally assumed that this approach can be combined with either of the control-oriented approaches [GJ98].

References

[Abr87] S. Abramsky. Observational equivalence as a testing equivalence. *Theoretical Computer Science*, 53(3):225–241, 1987.

[Ald90] R. Alderden. COOPER, the compositional construction of a canonical tester. In S.T. Vuong, editor, *FORTE'89*, pages 13–17. North-Holland, 1990.

[BAL+90] E. Brinksma, R. Alderden, R. Langerak, J. van de Lagemaat, and J. Tretmans. A formal approach to conformance testing. In J. de Meer, L. Mackert, and W. Effelsberg, editors, *Second Int. Workshop on Protocol Test Systems*, pages 349–363. North-Holland, 1990. Also: Memorandum INF-89-45, University of Twente, The Netherlands.

[BB87] T. Bolognesi and E. Brinksma. Introduction to the ISO specification language LOTOS. *Computer Networks and ISDN Systems*, 14:25–59, 1987.

[Ber91] G. Bernot. Testing against formal specifications: A theoretical view. In S. Abramsky and T. S. E. Maibaum, editors, *TAPSOFT'91, Volume 2*, pages 99–119. Lecture Notes in Computer Science 494, Springer-Verlag, 1991.

[BFV+99] A. Belinfante, J. Feenstra, R.G. de Vries, J. Tretmans, N. Goga, L. Feijs, S. Mauw, and L. Heerink. Formal test automation: A simple experiment. In G. Csopaki, S. Dibuz, and K. Tarnay, editors, *Int. Workshop on Testing of Communicating Systems 12*, pages 179–196. Kluwer Academic Publishers, 1999.

[BHT97] E. Brinksma, L. Heerink, and J. Tretmans. Developments in testing transition systems. In M. Kim, S. Kang, and K. Hong, editors, *Int. Workshop on Testing of Communicating Systems 10*, pages 143–166. Chapman & Hall, 1997.

[BK85] J.A. Bergstra and J.W. Klop. Algebra of communicating processes with abstraction. *Theoretical Computer Science*, 37(1):77–121, 1985.

[Bri87] E. Brinksma. On the existence of canonical testers. Memorandum INF-87-5, University of Twente, Enschede, The Netherlands, 1987.

[Bri88] E. Brinksma. A theory for the derivation of tests. In S. Aggarwal and K. Sabnani, editors, *Protocol Specification, Testing, and Verification VIII*, pages 63–74. North-Holland, 1988. Also: Memorandum INF-88-19, University of Twente, The Netherlands.

[Bri93] E. Brinksma. On the coverage of partial validations. In M. Nivat, C.M.I. Rattray, T. Rus, and G. Scollo, editors, *AMAST'93*, pages 247–254. BCS-FACS Workshops in Computing Series, Springer-Verlag, 1993.

[CFP94] A.R. Cavalli, J.P. Favreau, and M. Phalippou. Formal methods in conformance testing: Results and perspectives. In O. Rafiq, editor, *Int. Workshop on Protocol Test Systems VI*, number C-19 in IFIP Transactions, pages 3–17. North-Holland, 1994.

[CG97] O. Charles and R. Groz. Basing Test Coverage on a Formalization of Test Hypotheses. In M. Kim, S. Kang, and K. Hong, editors, *Int. Workshop on Testing of Communicating Systems 10*, pages 109–124. Chapman & Hall, 1997.

[CGPT96] M. Clatin, R. Groz, M. Phalippou, and R. Thummel. Two approaches linking test generation with verification techniques. In A. Cavalli and S. Budkowski, editors, *Eight Int. Workshop on Protocol Test Systems*. Chapman & Hall, 1996.

[CV97] J.A. Curgus and S.T. Vuong. Sensitivity analysis of the metric based test selection. In M. Kim, S. Kang, and K. Hong, editors, *Int. Workshop on Testing of Communicating Systems 10*, pages 200–219. Chapman & Hall, 1997.

[DAV93] K. Drira, P. Azéma, and F. Vernadat. Refusal graphs for conformance tester generation and simplification: A computational framework. In A. Danthine, G. Leduc, and P. Wolper, editors, *Protocol Specification, Testing, and Verification XIII*, number C-16 in IFIP Transactions. North-Holland, 1993.

[DBRV+00] L. Du Bousquet, S. Ramangalshy, C. Viho, A. Belinfante, and R.G. de Vries. Formal Test Automation: The Conference Protocol with TGV/TORX. In G. von Bochmann, R. Probert, and H. Ural, editors, *TestCom 2000*. Kluwer Academic Publishers, 2000. To appear.

[DN87] R. De Nicola. Extensional equivalences for transition systems. *Acta Informatica*, 24:211–237, 1987.

[DNH84] R. De Nicola and M.C.B. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.

[Doo91] P. Doornbosch. Test Derivation for Full LOTOS. Memorandum INF-91-51, University of Twente, Enschede, The Netherlands, 1991. Master's Thesis.

[Dri92] K. Drira. *Transformation et Composition de Graphes de Refus: Analyse de la Testabilité*. PhD thesis, Laboratoire d'Automatique et d'Analyse des Systemes du CNRS, Toulouse, France, 1992.

[Eer87] H. Eertink. The implementation of a test derivation algorithm. Memorandum INF-87-36, University of Twente, Enschede, The Netherlands, 1987.

[FJJV96] J.-C. Fernandez, C. Jard, T. Jéron, and C. Viho. Using on-the-fly verification techniques for the generation of test suites. In R. Alur and T.A. Henzinger, editors, *Computer Aided Verification CAV'96*. Lecture Notes in Computer Science 1102, Springer-Verlag, 1996.

[FJJV97] J.-C. Fernandez, C. Jard, T. Jéron, and C. Viho. An experiment in automatic generation of test suites for protocols with verification technology. *Science of Computer Programming – Special Issue on COST247, Verification and Validation Methods for Formal Descriptions*, 29(1–2):123–146, 1997.

[Gar98] H. Garavel. OPEN/CÆSAR: An open software architecture for verification, simulation, and testing. In B. Steffen, editor, *Fourth Int. Workshop on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'98)*, pages 68–84. Lecture Notes in Computer Science 1384, Springer-Verlag, 1998.

[Gau95] M.-C. Gaudel. Testing can be formal, too. In P.D. Mosses, M. Nielsen, and M.I. Schwartzbach, editors, *TAPSOFT'95: Theory and Practice of Software Development*, pages 82–96. Lecture Notes in Computer Science 915, Springer-Verlag, 1995.

[GJ98] M.-C. Gaudel and P.R. James. Testing algebraic data types and processes: A unifying theory. In J.F. Groote, B. Luttik, and J. van Wamel, editors, *Third Int. Workshop on Formal Methods for Industrial Critical Systems (FMICS'98)*, pages 215–230, Amsterdam, The Netherlands, 1998. CWI.

[Gla90] R.J. van Glabbeek. The linear time – branching time spectrum. In J.C.M. Baeten and J.W. Klop, editors, *CONCUR'90*, Lecture Notes in Computer Science 458, pages 278–297. Springer-Verlag, 1990.

[Gla93] R.J. van Glabbeek. The linear time – branching time spectrum II (The semantics of sequential systems with silent moves). In E. Best, editor, *CONCUR'93*, Lecture Notes in Computer Science 715, pages 66–81. Springer-Verlag, 1993.

[GSDD97] J. Grabowski, R. Scheurer, Z.R. Dai, and Hogrefe. D. Applying SaMsTaG to the B-ISDN protocol SSCOP. In M. Kim, S. Kang, and K. Hong, editors, *Int. Workshop on Testing of Communicating Systems 10*, pages 397–415. Chapman & Hall, 1997.

[Hee98] L. Heerink. *Ins and Outs in Refusal Testing*. PhD thesis, University of Twente, Enschede, The Netherlands, 1998.

[Hen88] M. Hennessy. *Algebraic Theory of Processes*. Foundations of Computing Series. The MIT Press, 1988.

[HFT00] L. Heerink, J. Feenstra, and J. Tretmans. Formal Test Automation: The Conference Protocol with PHACT. In G. von Bochmann, R. Probert, and H. Ural, editors, *TestCom 2000*. Kluwer Academic Publishers, 2000. To appear.

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

[Hol91] G.J. Holzmann. *Design and Validation of Computer Protocols*. Prentice-Hall Inc., 1991.

[HST96] D. Hogrefe, Heymer. S., and J. Tretmans. Report on the standardization project "Formal Methods in Conformance Testing". In B. Baumgarten, H.-J. Burkhardt, and A. Giessler, editors, *Int. Workshop on Testing of Communicating Systems 9*, pages 289–298. Chapman & Hall, 1996.

[HT96] L. Heerink and J. Tretmans. Formal methods in conformance testing: A probabilistic refinement. In B. Baumgarten, H.-J. Burkhardt, and A. Giessler, editors, *Int. Workshop on Testing of Communicating Systems 9*, pages 261–276. Chapman & Hall, 1996.

[HT97] L. Heerink and J. Tretmans. Refusal testing for classes of transition systems with inputs and outputs. In T. Mizuno, N. Shiratori, T. Higashino, and A. Togashi, editors, *Formal Desciption Techniques and Protocol Specification, Testing and Verification FORTE X /PSTV XVII '97*, pages 23–38. Chapman & Hall, 1997.

[HT99] J. He and K.J. Turner. Protocol-Inspired Hardware Testing. In G. Csopaki, S. Dibuz, and K. Tarnay, editors, *Int. Workshop on Testing of Communicating Systems 12*, pages 131–147. Kluwer Academic Publishers, 1999.

[ISO89] ISO. *Information Processing Systems, Open Systems Interconnection, LOTOS - A Formal Description Technique Based on the Temporal Ordering of Observational Behaviour.* International Standard IS-8807. ISO, Geneve, 1989.

[ISO91] ISO. *Information Technology, Open Systems Interconnection, Conformance Testing Methodology and Framework.* International Standard IS-9646. ISO, Geneve, 1991. Also: CCITT X.290–X.294.

[ISO96] ISO/IEC JTC1/SC21 WG7, ITU-T SG 10/Q.8. *Information Retrieval, Transfer and Management for OSI; Framework: Formal Methods in Conformance Testing.* Committee Draft CD 13245-1, ITU-T proposed recommendation Z.500. ISO – ITU-T, Geneve, 1996.

[JJKV98] C. Jard, T. Jéron, H. Kahlouche, and C. Viho. Towards Automatic Distribution of Testers for Distributed Conformance Testing. In S. Budkowski, A. Cavalli, and Najm E., editors, *Formal Desciption Techniques and Protocol Specification, Testing and Verification FORTE X /PSTV XVII '98*, pages 353–368. Kluwer Academic Publishers, 1998.

[JJM00] C. Jard, T. Jéron, and P. Morel. Verification of Test Suites. In G. von Bochmann, R. Probert, and H. Ural, editors, *TestCom 2000*. Kluwer Academic Publishers, 2000. To appear.

[JM99] T. Jéron and P. Morel. Test generation derived from model-checking. In D. Halbwachs, N. nad Peled, editor, *Computer Aided Verification CAV'99*, pages 108–121. Lecture Notes in Computer Science 1633, Springer-Verlag, 1999.

[KJG99] A. Kerbrat, T. Jéron, and R. Groz. Automated Test Generation from SDL Specifications. In R. Dssouli, G. von Bochmann, and Y. Lahav, editors, *SDL'99, The Next Millennium – Proceedings of the* 9^{th} *SDL Forum*, pages 135–152. Elsevier Science, 1999.

[KVZ98] H. Kahlouche, C. Viho, and M. Zendri. An industrial experiment in automatic generation of executable test suites for a cache coherency protocol. In A. Petrenko and N. Yevtushenko, editors, *Int. Workshop on Testing of Communicating Systems 11*, pages 211–226. Kluwer Academic Publishers, 1998.

[Lan90] R. Langerak. A testing theory for LOTOS using deadlock detection. In E. Brinksma, G. Scollo, and C. A. Vissers, editors, *Protocol Specification, Testing, and Verification IX*, pages 87–98. North-Holland, 1990.

[Led92] G. Leduc. A framework based on implementation relations for implementing LOTOS specifications. *Computer Networks and ISDN Systems*, 25(1):23–41, 1992.

[LGA96] P. Le Gall and A. Arnould. Formal specifications and test: Correctness and oracle. In O.-J. Dahl, O. Owe, and M. Haveraaen, editors, 11^{th} *ADT Workshop*. Lecture Notes in Computer Science, Springer-Verlag, 1996.

[LT89] N.A. Lynch and M.R. Tuttle. An introduction to Input/Output Automata. *CWI Quarterly*, 2(3):219–246, 1989. Also: Technical Report MIT/LCS/TM-373 (TM-351 revised), Massachusetts Institute of Technology, Cambridge, U.S.A., 1988.

[LY96] D. Lee and M. Yannakakis. Principles and methods for testing finite state machines – a survey. *The Proceedings of the IEEE*, 84, August 1996.

[Mar95] B. Marre. LOFT: A tool for assisting selection of test data sets from algebraic specifications. In P.D. Mosses, M. Nielsen, and M.I. Schwartzbach, editors, *TAPSOFT'95: Theory and Practice of Software Development*, pages 799–800. Lecture Notes in Computer Science 915, Springer-Verlag, 1995.

[Mil89] R. Milner. *Communication and Concurrency*. Prentice-Hall, 1989.

[Pet00] A. Petrenko. In *These Proceedings*, 2000.

[PF90] D. H. Pitt and D. Freestone. The derivation of conformance tests from LOTOS specifications. *IEEE Transactions on Software Engineering*, 16(12):1337–1343, 1990.

[Pha94a] M. Phalippou. Executable testers. In O. Rafiq, editor, *Int. Workshop on Protocol Test Systems VI*, number C-19 in IFIP Transactions, pages 35–50. North-Holland, 1994.

[Pha94b] M. Phalippou. *Relations d'Implantation et Hypothèses de Test sur des Automates à Entrées et Sorties*. PhD thesis, L'Université de Bordeaux I, France, 1994.

[Phi87] I. Phillips. Refusal testing. *Theoretical Computer Science*, 50(2):241–284, 1987.

[PS97] J. Peleska and M. Siegel. Test automation of safety-critical reactive systems. *South African Computer Jounal*, 19:53–77, 1997.

[Ray87] D. Rayner. OSI conformance testing. *Computer Networks and ISDN Systems*, 14:79–98, 1987.

[Ray97] D. Rayner. Future directions for protocol testing, learning the lessons from the past. In M. Kim, S. Kang, and K. Hong, editors, *Int. Workshop on Testing of Communicating Systems 10*, pages 3–17. Chapman & Hall, 1997.

[Seg93] R. Segala. Quiescence, fairness, testing, and the notion of implementation. In E. Best, editor, *CONCUR'93*, pages 324–338. Lecture Notes in Computer Science 715, Springer-Verlag, 1993.

[SEK+98] M. Schmitt, A. Ek, B. Koch, J. Grabowski, and D. Hogrefe. – AUTOLINK – Putting SDL-based Test Generation into Practice. In A. Petrenko and N. Yevtushenko, editors, *Int. Workshop on Testing of Communicating Systems 11*, pages 227–243. Kluwer Academic Publishers, 1998.

[Spi] Spin. On-the-Fly, LTL Model Checking with SPIN. URL: `http://netlib.bell-labs.com/netlib/spin/whatispin.html`.

[STW96] Dutch Technology Foundation STW. *Côte de Resyste* – COnformance TEsting of REactive SYSTEms. Project proposal STW TIF.4111, University of Twente, Eindhoven University of Technology, Philips Research Laboratories, KPN Research, Utrecht, The Netherlands, 1996. URL: `http://fmt.cs.utwente.nl/CdR`.

[Tan97] Q. Tan. *On Conformance Testing of Systems Communicating by Rendezvous*. PhD thesis, Université de Montréal, Montréal, Canada, 1997.

[TKB92] J. Tretmans, P. Kars, and E. Brinksma. Protocol conformance testing: A formal perspective on ISO IS-9646. In J. Kroon, R. J. Heijink, and E. Brinksma, editors, *Fourth Int. Workshop on Protocol Test Systems*, number C-3 in IFIP Transactions, pages 131–142. North-Holland, 1992. Extended abstract of Memorandum INF-91-32, University of Twente, The Netherlands, 1991.

[TPB96] Q.M. Tan, A. Petrenko, and G. von Bochmann. Modeling Basic LOTOS by FSMs for conformance testing. In P. Dembiński and M. Średniawa, editors, *Protocol Specification, Testing, and Verification XV*, pages 137–152. IFIP WG6.1, Chapman & Hall, 1996. Also: publication # 958, Université de Montréal, Département d'Informatique et de Recherche Opérationnelle.

[Tre90] J. Tretmans. Test case derivation from LOTOS specifications. In S.T. Vuong, editor, *FORTE'89*, pages 345–359. North-Holland, 1990. Also: Memorandum INF-90-21, University of Twente, The Netherlands.

[Tre94] J. Tretmans. A formal approach to conformance testing. In O. Rafiq, editor, *Int. Workshop on Protocol Test Systems VI*, number C-19 in IFIP Transactions, pages 257–276. North-Holland, 1994.

[Tre96a] J. Tretmans. Conformance testing with labelled transition systems: Implementation relations and test generation. *Computer Networks and ISDN Systems*, 29:49–79, 1996.

[Tre96b] J. Tretmans. Test generation with inputs, outputs and repetitive quiescence. *Software—Concepts and Tools*, 17(3):103–120, 1996. Also: Technical Report No. 96-26, Centre for Telematics and Information Technology, University of Twente, The Netherlands.

[Tre99] J. Tretmans. Testing concurrent systems: A formal approach. In J.C.M Baeten and S. Mauw, editors, *CONCUR'99 – 10^{th} Int. Conference on Concurrency Theory*, volume 1664 of *Lecture Notes in Computer Science*, pages 46–65. Springer-Verlag, 1999.

[TV92] J. Tretmans and L. Verhaard. A queue model relating synchronous and asynchronous communication. In R.J. Linn and M.Ü. Uyar, editors, *Protocol Specification, Testing, and Verification XII*, number C-8 in IFIP Transactions, pages 131–145. North-Holland, 1992. Extended abstract of Memorandum INF-92-04, University of Twente, Enschede, The Netherlands, 1992, and Internal Report, TFL RR 1992-1, TFL, Hørsholm, Denmark.

[VT98] R.G. de Vries and J. Tretmans. On-the-Fly Conformance Testing using Spin. In G. Holzmann, E. Najm, and A. Serhrouchni, editors, *Fourth Workshop on Automata Theoretic Verification with the* Spin *Model Checker*, ENST 98 S 002, pages 115–128, Paris, France, November 2, 1998. Ecole Nationale Supérieure des Télécommunications. Also to appear in *Software Tools for Technology Transfer*.

[VTKB93] L. Verhaard, J. Tretmans, P. Kars, and E. Brinksma. On asynchronous testing. In G. von Bochmann, R. Dssouli, and A. Das, editors, *Fifth Int. Workshop on Protocol Test Systems*, IFIP Transactions. North-Holland, 1993. Also: Memorandum INF-93-03, University of Twente, The Netherlands.

[Wez90] C. D. Wezeman. The CO-OP method for compositional derivation of conformance testers. In E. Brinksma, G. Scollo, and C. A. Vissers, editors, *Protocol Specification, Testing, and Verification IX*, pages 145–158. North-Holland, 1990.