The Systems Validation Centre in Retrospect

Henk Eertink (Telematica Instituut) Wan Fokkink (CWI)
Izak van Langevelde (CWI) Holger Hermanns (Univ. Twente)

Abstract

Solid theoretical foundations, state-of-the-art tools and industrial case studies were among the buzzwords of the blueprints of the Dutch Systems Validation Centre (SVC) when it was launched four years ago [11]. Now that the SVC has closed according to plan per December 31, 2002, it is time to take stock.

1 Introduction

In 1999, the Systems Validation Centre was initiated as a joint effort of Telematica Instituut, the Embedded Systems Group at CWI and the Formal Methods and Tools Group at the University of Twente, with support through industrial partners, including CMG, IBM, KPN and Lucent [11]. The rationale of this structure was to have the academic partners lay the theoretical foundations for the development of tools and techniques, and to have Telematica Instituut bring in, through its contacts, the industrial case studies.

This paper gives an overview of the SVC research on foundations, tools and cases. It contains a list of theses and publications in which SVC was involved.

2 Foundations

The theoretical spectrum of the SVC is spanned by two foundational cornerstones: process algebra with abstract data types and stochastic process algebras. The former is centered around the specification language μCRL, developed at CWI, while the latter is concentrated in the work on compositional Markov model generation and analysis, developed in Twente.

The fundamental work on μCRL concentrated on the transformation of a specification into an equivalent specification without parallel merge, communication, encapsulation and abstraction. It is this so-called linear format which is the pivot in the μCRL toolset, in the sense that all tools benefit from this format. The previously existing transformation [18] was restricted to a subset of μCRL, which was useful for a rich variety of system behaviour, but was not able to express, for instance, dynamic process creation. With the extension of this transformation to arbitrary μCRL specifications, the full spectrum of system behaviour is made accessible to the μCRL tools. This research was the main theme of Yaroslav Usenko's PhD thesis [3]; it was published in [35].

Work on compositional specification of stochastic models focused on the theory of interactive Markov chains [19,21] and of stochastic automata [13]. Both models constitute generalisations of different stochastic processes which occur in standard performance and discrete event simulation approaches, but is fully compositional. The stochastic automata model and the process algebra SPADES have been the core contribution of the PhD thesis of Pedro D'Argenio [1]. Related activities include algorithms to perform model checking of

stochastic processes [6,7], and to minimise such models [23], as well as results answering the question what constitutes an adequate form of composition of discrete stochastic models [12].

Further foundational research was performed on finite partial-order unfoldings for process algebra [26] and on flow analysis for model checking asynchronous systems [25].

3 Tools

The two main pillars of the tools developed in SVC directly rest on the two foundation stones of theoretical research: the μCRL toolset and the tools developed for compositional Markov model construction and model checking.

The μCRL toolset consists of efficient tools for linearisation, state space generation, optimisation, reduction, theorem proving and model checking [8]. One example of a promising development is the design and implementation of distributed tools for generation and reduction [9], which shifted the dimensions of systems that can be successfully analysed with several orders of magnitude. A second example worth mentioning is the fruitful combination of theorem proving and state space generation in a state space generator which benefits from the marking of confluent τ steps to generate a reduced state space [10]. Third, a number of important optimisation techniques for linear equations were implemented [16]. Fourth, a rewriter based on strategy annotations was developed [30,31]. A comparison of the generators in the μCRL and SPIN toolsets was made on the basis of a leader election protocol within the PIEEE 1394 Standard for Home Audio/Video Interoperability [34].

Major parts of the PhD thesis of Theo Ruys [2] are devoted to the *art of modelling* in the context of exhaustive verification techniques, using the SPIN tool, while [33] introduced a project management system for SPIN. The TIPPTOOL for modelling and evaluation of performance and dependability models was improved substantially within the SVC project [22]. The ETMCC model checker [20], the first verification tool for Markov models and continuous stochastic logic, was developed and successfully applied to case studies.

4 Case Studies

Firm foundations and state-of-the-art tools facilitated bridging the gap between theory and practice. In a number of industrial case studies, the expressive and analysing strength of theory and tools were assessed and improved.

Transaction Capabilities Procedures (TCAP) are part of the Signalling System No. 7, of which an optimisation was implemented in Erlang by Ericsson. Both the original TCAP and its optimisation were specified in μCRL, after which the μCRL (and the French Caesar/Aldebaran) tools were used to check equivalence of the two specifications. As a result, a number of small bugs were localised and fixed; one of these involved a memory leak which would have been hard to track down by conventional means [4,5].

Coordination architectures served as a fruitful domain for a number of cases. A detailed specification of Splice, developed by Thales was written in μCRL and model checked using Caesar/Aldebaran in order to establish soundness and completeness of read/write actions [14]. Other topics covered are transparent replication [24], JavaSpaces [32] and distribution of shared data spaces [28].

The verification of the IEEE P1394.1 Draft Standard for High Performance Serial Bus Bridges is a large case, in cooperation with the Technical University Eindhoven. This study targeted the correctness of the net update procedure. which is to guarantee that

the service of a network of buses connected through bridges is not interrupted by addition or removal of bridges. The relevant parts of the standard were specified in Promela and analysed using Spin, which revealed several shortcomings and one bug in the loop detection algorithm [27].

Other cases are the verification using μCRL of a distributed system for lifting trucks [17], an in-flight data acquisition unit for Lynx helicopters [15], and a cache coherence protocol for a Java Distributed Memory system [29].

Together with Lucent Technologies, the stability and control of an SDH data communication network, by measuring the performance of an experimental network at Lucent using both simulation and numerical methods. Although the main focus was on performance issues, as a bonus the measurements revealed system traces that do not adhere to the specification of the intended network behaviour.

5 Spin-off Projects

The SVC experience stimulated the start-up of a number of projects, guaranteeing the continuation of efforts initiated during SVC. At CWI, the PROGRESS projects "Improving the Quality of Embedded Systems by Formal Design and Systematic Testing" and "Formal Design, Tooling, and Prototype Implementation of a Real-Time Distributed Shared Data Space", and the NWO projects "Integrating Techniques for the Verification of Distributed Systems" and "Tools and Techniques for Integrating Performance Analysis and System Verification" build on the heritage of SVC. Among the spin-offs launched by the UT are the PROGRESS projects "Verification of Hard and Softly Timed Systems" and "Atom Splitting in Embedded Systems Testing", the NWO projects "Specification-Based Performability Checking" and "Systematic Testing of Real-Time Software Systems", and the NWO Vernieuwingsimpuls "Verification of Performance and Dependability".

Theses

1. D'Argenio, P.R., *Algebras and Automata for Timed and Stochastic Systems.* IPA Dissertation Series 1999-10, October 1999.

2. Ruys, Th.C., *Towards Effective Model Checking.* IPA Dissertation Series 2001-10, March 2001.

3. Usenko, Y.S., *Linearization in μCRL.* IPA Dissertation Series 2002-16, December 2002.

Selected Publications

4. Arts, Th. and Langevelde, I.A. van, How μCRL Supported a Smart Redesign of a Real-Life Protocol. In: *Proc. FMICS'99*, pp. 31–53, 1999.

5. Arts, Th. and Langevelde, I.A. van, Correct Performance of Transaction Capabilities. In: *Proc. ICACSD'01*, pp. 35–42, IEEE, 2001.

6. Baier, C., Katoen, J.P. and Hermanns, H., Approximate Symbolic Model Checking of Continuous-Time Markov Chains. In: *Proc. CONCUR'99*, LNCS 1664, pp. 146–162, Springer, 1999.

7. Baier, C., Katoen, J.P. and Hermanns, H., Model Checking Continuous-Time Markov Chains by Transient Analysis. In: *Proc. CAV'00*, LNCS 1855, pp. 358–372, Springer, 2000.

8. Blom, S.C.C., Fokkink, W.J., Groote, J.F., Langevelde, I.A. van, Lisser, B. and Pol, J.C. van de, μCRL: A Toolset for Analysing Algebraic Specifications. In: *Proc. CAV'01*, LNCS 2102, pp. 250–254, Springer, 2001.

9. Blom, S.C.C. and Orzan, S.M., A Distributed Algorithm for Strong Bisimulation Reduction of State Spaces. In: *Proc. PDMC'02*, ENTCS 68(4), Elsevier, 2002.

10. Blom, S.C.C. and Pol, J.C. van de, State Space Reduction by Proving Confluence. In: *Proc. CAV'02*, LNCS 2404, pp. 596–609, Springer, 2002.

11. Brinksma, E. and Groote, J.F., Validatietechnieken Houden Complexe Systemen Hanteerbaar. *Automatiseringgids* 19, 1999.

12. D'Argenio, P.R., Hermanns, H. and Katoen, J.P., On Generative Parallel Composition. ENTCS 22, Elsevier, 1999.

13. D'Argenio, P.R., Katoen, J.P. and Brinksma, E., Specification and Analysis of Soft Real-Time Systems: Quantity and Quality. In: *Proc. RTSS'99*, pp. 104–114, IEEE, 1999.

14. Dechering, P.F.G. and Langevelde, I.A. van, The Verification of Coordination. In: *Proc. COORDINATION'00*, LNCS 1906, pp. 335–340, Springer, 2000.

15. Fokkink, W.J., Ioustinova, N., Kesseler, E., Pol, J.C. van de, Usenko, Y.S. and Yushtein, Y., Refinement and Verification Applied to an In-Flight Data Acquisition Unit. In: *Proc. CONCUR'02*, LNCS 2421, pp. 1–23, Springer, 2002.

16. Groote, J.F. and Lisser, B. Computer Assisted Manipulation of Algebraic Process Specifications. In: *Proc. VCL'02*, Report DSSE-TR-2002-5, University of Southampton, 2002.

17. Groote, J.F., Pang, J. and Wouters, A.G., Analysis of a Distributed System for Lifting Trucks. *Journal of Logic and Algebraic Programming* 55(1/2), pp. 21–56, 2003.

18. Groote, J.F., Ponse, A. and Usenko, Y.S., Linearization of Parallel pCRL. *Journal of Logic and Algebraic Programming* 48(1/2), pp. 39–72, 2001.

19. Hermanns, H., *Interactive Markov Chains and the Quest for Quantified Quality.* LNCS 2428, Springer, 2002.

20. Hermanns, H., Katoen, J.P., Meyer-Kayser, J. and Siegle, M., A Markov Chain Model Checker. In: *Proc. TACAS'00*, LNCS 1785, pp. 347–362. Springer, 2000.

21. Hermanns, H., Herzog, U. and Katoen, J.P., Process algebra for performance evaluation. *Theoretical Computer Science* 274(1/2), pp. 43–87, 2002.

22. Hermanns, H., Herzog, U., Klehmet, U., Mertsiotkis, V., and Siegle, M., Compositional Performance Modelling with the TIPPtool. *Performance Evaluation* 39(1/4), pp. 5–35, 2000.

23. Hermanns, H. and Siegle, M., Bisimulation Algorithms for Stochastic Process Algebras and their BDD-based Implementation. In: *Proc. ARTS'99*, LNCS 1601, pp. 144–264, Springer 1999.

24. Hooman, J. and Pol, J.C. van de, Formal Verification of Replication on a Distributed Data Space Architecture. In: *Proc. SAC'02*, pp. 351–358, ACM, 2002.

25. Ioustinova, N., Sidorova, N. and Steffen, M., Abstraction and Flow Analysis for Model Checking Open Asynchronous Systems. In: *Proc. APSEC'02*, pp. 227–235, IEEE, 2002.

26. Langerak, R. and Brinksma, E., A Complete Finite Prefix for Process Algebra. In: *Proc. CAV'99*, LNCS 1633, pp. 184–195, Springer, 1999.

27. Langevelde, I.A. van, Romijn, J.M.T. and Goga, N., Founding FireWire Bridges through Promela Prototyping. In: *Proc. FMPPTA'03*, IEEE, 2003.

28. Orzan, S.M. and Pol, J.C. van de, Distribution of a Simple Shared Dataspace Architecture. In: *Proc. FOCLASA'02*, ENTCS 68(3), Elsevier, 2002.

29. Pang, J., Fokkink, W.J., Hofman, R. and Veldema, R., Model Checking a Cache Coherence Protocol for a Java DSM Implementation. In: *Proc. FMPPTA'03*, IEEE, 2003.

30. Pol, J.C. van de, Just-In-Time: On Strategy Annotations. In: *Proc. WRS'01*, ENTCS 57, Elsevier, 2001.

31. Pol, J.C. van de, JITty: A Rewriter with Strategy Annotations. In: *Proc. RTA'02*, LNCS 2378, pp. 367–370, Springer, 2002.

32. Pol, J.C. van de and Valero Espada, M., Formal Specification of JavaSpaces Architecture using μCRL. In: *Proc. COORDINATION'02*, LNCS 2315, pp. 274–290, Springer, 2002.

33. Ruys, Th.C., Xspin/Project – Integrated Validation Management for Xspin, In: *Proc. SPIN'99*, LNCS 1680, pp. 108–119, Springer, 1999.

34. Usenko, Y.S., State Space Generation for the HAVi Leader Election Protocol. *Science of Computer Programming* 43(1), pp. 1–33, 2002.

35. Usenko, Y.S., Linearization of μCRL Specifications. In: *Proc. VCL'02*, Report DSSE-TR-2002-5, University of Southampton, 2002.