NASA Monographs in Systems and Software Engineering

Series editor

Mike G. Hinchey, Limerick, Ireland

The **NASA Monographs in Systems and Software Engineering** series addresses cutting-edge and groundbreaking research in the fields of systems and software engineering. This includes in-depth descriptions of technologies currently being applied, as well as research areas of likely applicability to future NASA missions. Emphasis is placed on relevance to NASA missions and projects.

More information about this series at http://www.springer.com/series/7055

Mike G. Hinchey · Jonathan P. Bowen
Ernst-Rüdiger Olderog
Editors

Provably Correct Systems

Springer

Editors
Mike G. Hinchey
Lero–The Irish Software Research Centre
University of Limerick
Limerick
Ireland

Ernst-Rüdiger Olderog
Department für Informatik
Universität Oldenburg
Oldenburg
Germany

Jonathan P. Bowen
School of Engineering
London South Bank University
London
UK

ISSN 1860-0131 ISSN 2197-6597 (electronic)
NASA Monographs in Systems and Software Engineering
ISBN 978-3-319-48627-7 ISBN 978-3-319-48628-4 (eBook)
DOI 10.1007/978-3-319-48628-4

Library of Congress Control Number: 2016959748

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Foreword

The ProCoS Project (1989–1991) was funded by the European Community as a Basic Research Project, with a continuation (ProCoS II) also funded from 1992 to 1995. It was included in the ESPRIT programme of internationally collaborative research in Information Technology. The inspiration of the project was the recent completion of the Stack verification project, undertaken by Computational Logic, Inc. This was a start-up company founded and directed by Bob Boyer and J Moore, professors at the University of Texas at Austin. Both the European and the US projects sought to advance the technology of software verification by accepting the challenge of verification of components of a free-standing computer system. These included its operating system, its assembler, and its automatic verification aids, and even the hardware of a processor chip.

Many technological breakthroughs were triggered by these two challenge projects. The international collaboration which was forged by the ProCoS Project has continued under support of the individual national funding agencies. It has inspired and accelerated the automation of program verification. The resulting tools have found application in many advanced modern industries, including industrial giants in aerospace, electronics, silicon fabrication, automobiles, communications, advertising, social networks, retail sales, as well as suppliers of compilers and of operating systems and general software.

Many of the collaborators in the original ProCoS project, together with their students and followers, have contributed to this broadening of the application of the original basic research. A representative selection of their recent work was presented at the ProCoS Workshop in March 2015; and I welcome the publication of the proceedings in book form. I offer its authors and readers my best wishes for further progress in the understanding of the basic science, coupled with its broadest possible application.

Cambridge, UK — Tony Hoare
June 2016

Preface

ProCoS is the acronym for "Provably Correct Systems", a basic research project funded in two phases by the European Commission from 1989 to 1995. This project was planned by Tony Hoare (Oxford University), Dines Bjørner (DTU, Technical University of Denmark, Lyngby), and Hans Langmaack (University of Kiel). Its goal was to develop a mathematical basis for the development of embedded, real-time computer systems.

The survey paper on ProCoS presented at the conference FTRTFT (Formal Techniques in Real-Time and Fault-Tolerant Systems) 1994 states in its introduction:

> An embedded computer system is part of a total system that is a physical process, a plant, characterized by a state that changes over time. The role of the computer is to monitor this state through sensors and change the state through actuators. The computer is simply a convenient device that can be instructed to manipulate a mathematical model of the physical system and state. Correctness means that the program and the hardware faithfully implement the control formulas of the mathematical model of the total system, and nothing else. However, the opportunities offered by the development of computer technology have resulted in large, complex programs which are hard to relate to the objective of system control.

The ProCoS project developed a particular approach to mastering the complexity of such systems. Its emphasis was on proving system correctness across different abstraction layers. The inspiration for ProCoS stems from a sabbatical of Tony Hoare at the University of Austin at Texas in 1986. There he was impressed by the work of Robert S. Boyer and J Strother Moore on automated verification with their "Boyer-Moore" prover ACL2 at their company "Computational Logic, Inc." (CLI), in particular its application to a case study known as the "CLInc Stack". Discussing later with Dines Bjørner and Hans Langmaack, a project on the foundation of verification of many-layered systems was conceived: ProCoS. The different levels of abstraction studied in this project became known as the "ProCoS Tower". They comprised (informal) expectations, (formal) requirements, (formal) system specifications, programs (in the "occam" programming language), machine code (for the "transputer" microprocessor), and circuit diagrams (described using "netlists").

During the final deliverable for the first phase of ProCoS, Tony Hoare wrote in 1993:

> In summary, our overall goal is not to produce a single verified system or any particular verified language or compiler, but rather to advance the state of the art of systematic design of complex heterogeneous systems, including both hardware and software; and to concentrate attention on reducing the risk of error in the specification, design and implementation of embedded safety critical systems.

In the first phase, the ProCoS project comprised seven partners: Oxford University, Technical University of Denmark at Lyngby, Christian-Albrechts Universität Kiel, Universität Oldenburg, Royal Holloway and Bedford New College, Århus University, and the University of Manchester. In the second phase (ProCoS II), the team consisted of the first four original partners. The EU funding of ProCoS was relatively small. During the second phase only one researcher at each of the four partner sites was funded, but many more students and researchers at these sites contributed to the goals.

ProCoS was much influenced by the work of two Chinese scientists contributing to the project at Lyngby and Oxford: Zhou Chaochen and He Jifeng.

Zhou Chaochen and Anders P. Ravn initiated a major conceptual development of ProCoS: the Duration Calculus, an interval-based logic for specifying real-time requirements. The first paper on it was published by Zhou Chaochen, Tony Hoare and Anders P. Ravn in 1991. The types of durational properties that can be expressed in the Duration Calculus were motivated by the case study of a gas burner that was defined by E.V. Sørensen from DTU in collaboration with a Danish gas burner manufacturer. He Jifeng cooperated closely with Tony Hoare on a predicative approach to programming that led to the book "Unifying Theories of Programming" (UTP) published in 1998. The work on UTP has attracted a number of researchers and led to a series of symposiums on this topic.

To bridge the gap from requirements to programs, a combination of specification techniques for data and processes with transformation rules was developed by the group of E.-R. Olderog in Oldenburg. The topic of correct compilers, exemplified for the translation of an occam-like programming language to transputer machine code, was investigated in the group of Hans Langmaack in Kiel. Oxford contributed an algebraic approach to compiling verification.

Associated with the ProCoS project was an EU-funded ProCoS Working Group (1994–1997) of 25 academic and industrial partners interested in provably correct systems, arranging various meetings around Europe.

Other associated national projects in the United Kingdom included the "safemos" project (1989–1993), a UK EPSRC project on "Provably Correct Hardware/Software Co-design" (1993–1996), and an EPSRC Visiting Fellowship on "Provably Correct Real Time Systems" (1996–1997). Associated travel funding to encourage collaboration included ESPRIT/NSF ProCoS-US initiative on "Provably Correct Hardware Compilation" with Cornell University in the US, and KIT (Keep in Touch) grants with UNU/IIST in Macau (1993–1998) and PROCORSYS with the Federal University of Pernambuco in Brazil (1994–1997).

Impact

An extension of the Duration Calculus to cover continuous dynamical systems was led by Anders P. Ravn and Hans Rischel at DTU to contribute to the initial research on hybrid systems.

From 1992 until 1997, Dines Bjørner was the founding director of UNU-IIST, the International Institute for Software Technology of the United Nations University in Macau. Ideas from the ProCoS project flourished at the institute and were taken up by researchers from Asia working there. Also, a number of scientists associated with ProCoS visited UNU-IIST or had research posts for several years, including He Jifeng and Zhou Chaochen. From 1997 until 2002, Zhou Chaochen succeeded Dines Bjørner as the director of UNU-IIST, during the time of transition of Macau from a Portuguese to a Chinese city. Regrettably, in 2013 the United Nations decided to disband academic staff at UNU-IIST.

A number of young ProCoS contributors pursued academic careers. Martin Fränzle and Markus Müller-Olm, students in Kiel during the ProCoS project, are now professors at the universities of Oldenburg and Münster, respectively. Also, Debora Weber-Wulff and Bettina Buth, at Kiel during the ProCoS project, are now professors in Berlin and Hamburg, respectively. Michael Schenke, a ProCoS contributor at Oldenburg, is now a professor in Merseburg. Augusto Sampaio, during ProCoS working on his Ph.D. at Oxford on an algebraic approach to compilation during ProCoS, has became a professor at the University of Pernambuco, Brazil. Paritosh K. Pandya, working at Oxford during the ProCoS project, has become a professor at the Tata Institute of Fundamental Research in Mumbai, India. Zhiming Liu, who during ProCoS times spent a year as a postdoc at DTU and later was a researcher at UNI-IIST, is now professor at the Southwest University in Chongqing, China.

The collaborative project Verifix (*Construction and Architecture of Verifying Compilers*) directed by Gerhard Goos, Friedrich von Henke and Hans Langmaack and funded 1995–2004 by the German Research Foundation (DFG) deepened research on compiler correctness begun in the ProCoS project.

The large-scale Transregional Collaborative Research Center AVACS (*Automatic Verification and Analysis of Complex Systems*), directed by Werner Damm and funded by German Research Foundation (DFG) during the period 2004–2015, continued research pioneered in ProCoS but with emphasis on automation and for wider classes of systems. The collaborating sites were Oldenburg, Freiburg, and Saarbrücken. AVACS comprised of nine projects in the areas of real-time systems, hybrid systems, and systems of systems.

A series of conferences called VSTTE (*Verified Systems—Theories, Tools and Experiments*), was initiated by a vision for a Grand Challenge project formulated by Tony Hoare and Jay Misra in July 2005.

The ProCoS project and its related initiatives have inspired a number of books, including the following:

- He Jifeng, *Provably Correct Systems—Modelling of Communicating Languages and Design of Optimized Compilers*, McGraw-Hill, 1994.

- Jonathan P. Bowen (ed.), *Towards Verified Systems*, Elsevier Science, Real-Time Safety Critical Systems Series, 1994.
- Mike G. Hinchey and Jonathan P. Bowen (eds.), *Applications of Formal Methods*, Prentice Hall, Series in Computer Science, 1995.
- C.A.R. Hoare and He Jifeng, *Unifying Theories of Programming*, Prentice Hall, Series in Computer Science, 1998.
- Jonathan P. Bowen and Mike G. Hinchey, *High-Integrity System Specification and Design*, Springer, FACIT Series, 1999.
- Zhou Chaochen and Michael R. Hansen, *Duration Calculus—A Formal Approach to Real-Time Systems*, Springer, 2004.
- E.-R. Olderog and Henning Dierks, *Real-Time Systems—Formal Specification and Automatic Verification*, Cambridge University Press, 2008.

Structure of this Book

In September 2013, Jonathan Bowen and Ernst-Rüdiger Olderog met at the Festschrift Symposium for He Jifeng in Shanghai and discussed the possibility of having a workshop celebrating 25 years of ProCoS. This idea materialized with the help of Mike Hinchey in March 2015, when a two-day ProCoS Workshop with around 40 invited researchers and 25 presentations on the topic of "Provably Correct Systems" took part in the rooms of the BCS in London. This book consists of 13 chapters mainly describing recent advances on "Provably Correct Systems", based on presentations at that workshop. Each paper has been carefully reviewed by three to five reviewers. The chapters address the following topics:

- Historic Account,
- Hybrid Systems,
- Correctness of Concurrent Algorithms,
- Interfaces and Linking,
- Automatic Verification,
- Run-time Assertions Checking,
- Formal and Semi-formal Methods, and
- Web-Supported Communities in Science.

Historic Account

In the note "ProCoS: How it all Began—as seen from Denmark", Dines Bjørner opens his diary and shows entries by Tony Hoare during a meeting of IFIP Working Group 2.3 at Château du Pont d'Oye in Belgium in 1987. The author explains that this was a first draft on the content of ProCoS.

Hybrid Systems

Martin Fränzle, Yang Gao, and Sebastian Gerwinn review in Chap. "Constraint-Solving Techniques for the Analysis of Stochastic Hybrid Systems" definitions of (parametric) stochastic hybrid automata as needed for reliability evaluation. The authors then discuss automatic verification and synthesis methods based on arithmetic constraint solving. The chapters are able to solve step-bounded stochastic reachability problems and multi-objective parameter synthesis problems, respectively.

Mingshuai Chen, Xiao Han, Tao Tang, Shuling Wang, Mengfei Yang, Naijun Zhan, Hengjun Zhao, and Liang Zou introduce in Chap. "MARS: A Toolchain for Modelling, Analysis and Verification of Hybrid Systems" the toolchain MARS for Modelling, Analysing and verifying hybrid Systems. Using MARS, they build executable models of hybrid systems using the industrial standard environment Simulink/Stateflow, which facilitates analysis by simulation. The toolchain includes a translation of Simulink/Stateflow models to Hybrid CSP and verification using an interactive prover for Hybrid Hoare Logic.

Correctness of Concurrent Algorithms

John Derrick, Graeme Smith, Lindsay Groves, and Brijesh Dongol study in Chap. "A Proof Method for Linearizability on TSO Architectures" the correctness of non-atomic concurrent algorithms on a weak memory model, the TSO (Total Store Order) model. They show how linearizability is defined on TSO, and how one can adapt a simulation-based proof method for use on TSO. Their central result is a proof method that simplifies simulation-based proofs of linearizability on TSO.

Interfaces and Linking

E.-R. Olderog, A.P. Ravn, and R. Wisniewski investigate in Chap. "Linking Discrete and Continuous Models, Applied to Traffic Manoeuvrers" the interplay between discrete and continuous dynamical models, and combine them with linking predicates. The topic of linking system specifications at different levels of abstraction was central to the ProCoS project. However, here the application area is more advanced: traffic manoeuvrers of multiple vehicles on highways.

Zhiming Liu and Xin Chen discuss in Chap. "Towards Interface-Driven Design of Evolving Component-Based Architectures" how software design for complex evolving systems can be supported by an extension of the rCOS method for

refinement of component and object systems. It shows the need for a suitable interface theory and of multi-modelling notations for the description of multi-viewpoints of designs. This requires a theoretical foundation in the style of Unifying Theories of Programming as proposed by Tony Hoare and He Jifeng.

Automatic Verification

J Strother Moore presents in Chap. "Computing Verified Machine Address Bounds During Symbolic Exploration of Code" an abstract interpreter for machine address expressions that attempts to produce a bounded natural number interval guaranteed to contain the value of the expression. The interpreter has been proved correct by the ACL2 theorem prover. The author discusses the interpreter, what has been proved about it by ACL2, and how it is used in symbolic reasoning about machine code.

Shilpi Goel, Warren A. Hunt, Jr., and Matt Kaufmann describe in Chap. "Engineering a Formal, Executable x86 ISA Simulator for Software Verification" a formal, executable model of the x86 instruction-set architecture (ISA). They use this model to reason about x86 machine-code programs. Validation of the x86 ISA model is done by co-simulating it regularly against a physical x86 machine.

Jens Otten and Wolfgang Bibel present in Chap. "Advances in Connection-Based Automated Theorem Proving" calculi to automate theorem proving in classical and some important non-classical logics, namely first-order intuitionistic and first-order modal logics. These calculi are based on the connection method. The authors present details of the leanCoP theorem prover, a very compact PROLOG implementation of the connection calculus for classical logics. leanCoP has also been adapted to non-classical logics by integrating a prefix unification algorithm.

Run-Time Assertion Checking

Frank S. de Boer and Stijn de Gouw extend in Chap. "Run-Time Deadlock Detection" run-time assertions by attribute grammars for specifying properties of message sequences. These assertions are used in a method for detecting deadlocks at run-time in both multi-threaded Java programs and systems of concurrent objects.

Tim Todman and Wayne Luk present in Chap. "In-Circuit Assertions and Exceptions for Reconfigurable Hardware Design" a high-level approach to adding assertions and exceptions in a hardware design targeting FPGAs (Field Programmable Gate Arrays). They allow for imprecise assertions and exceptions to trade performance for accurate location of errors.

Formal and Semi-formal Methods

Bettina Buth reports in Chap. "From ProCoS to Space and Mental Models – A Survey of Combining Formal and Semi-Formal Methods" on work influenced by the ProCoS project. Systems from the application areas of space and aerospace are analysed using suitable abstractions to CSP specifications and the FDR model checker.

Web-Supported Communities in Science

Jonathan P. Bowen studies in Chap. "Provably Correct Systems: Community, Connections, and Citations" the building and support of scientific communities and collaboration, especially online, visualized graphically and formalized using the Z notation, including the concept of a "Community of Practice". His examples are drawn from the ProCoS project.

In summary, we hope that you enjoy this volume, providing a selection of research developments and perspectives since the original ProCoS initiatives of the 1990s. Further ProCoS-related information can be found online under:

http://formalmethods.wikia.com/wiki/ProCoS

Limerick, Ireland — Mike G. Hinchey
London, UK — Jonathan P. Bowen
Oldenburg, Germany — Ernst-Rüdiger Olderog

Acknowledgements

The following reviewed papers in these proceedings:
Wolfgang Bibel, Darmstadt University of Technology, Germany
Simon Bliudze, EPFL, Switzerland
Jonathan P. Bowen, London South Bank University, UK
Bettina Buth, HAW Hamburg, Germany
Michael Butler, University of Southampton, UK
Ana Cavalcanti, University of York, UK
Frank de Boer, CWI, The Netherlands
Willem-Paul de Roever, University of Kiel, Germany
John Derrick, Unversity of Sheffield, UK
Martin Fränzle, University of Olderburg, Germany
Anthony Hall, Independent consultant, UK
Jifeng He, East China Normal University, China
Warren Hunt, University of Texas, USA
Cliff Jones, Newcastle University, UK
Zhiming Liu, Southwest University, China
Annabelle McIver, Macquarie University, Australia
Dominique Méry, University of Lorraine, LORIA, France
Ernst-Rüdiger Olderog, University of Olderburg, Germany
Jan Peleska, University of Bremen, Germany
Anders P. Ravn, Aalborg University, Denmark
Augusto Sampaio, Federal university of Pernambuco, Brazil
Elizabeth Scott, University of London, UK
Jianqi Shi, National University of Singapore, Singapore
Marina Waldén, Åbo Akademi University, Finland
Jim Woodcock, University of York, UK
Naijun Zhan, Institute of Software, Chinese Academy of Sciences, China
Huibiao Zhu, East China Normal University, China

Contents