

Preface

This volume contains the joint proceedings of MARS 2018, the third workshop on *Models for Formal Analysis of Real Systems*, and VPT 2018, the sixth international workshop on *Verification and Program Transformation*, held together on April 20, 2018 in Thessaloniki, Greece, as part of ETAPS 2018, the European Joint Conferences on Theory and Practice of Software.

The MARS workshop series addresses the formal modelling of realistic systems. Making a formal model of a system is a necessary prerequisite for its formal analysis and for formal verification of its correctness.

To show the applicability of tools and techniques for verification and analysis, toy examples or tiny case studies are typically presented in research papers. Since the theory needs to be developed first, this approach is reasonable. However, to show that a developed approach actually scales to real systems, large case studies are essential. The development of formal models of real systems usually requires a perfect understanding of informal descriptions of the systems sometimes found in RFCs or other standard documents which are usually just written in English. Based on the type of system, an adequate specification formalism needs to be chosen, and the informal specification needs to be translated into it. Examples for such formalisms include process and program algebra, Petri nets, variations of automata, as well as timed, stochastic and probabilistic extensions of these formalisms. Abstraction from unimportant details then yields an accurate, formal model of the real system.

The process of developing a detailed and accurate model usually takes a considerable amount of time, often months or years; without even starting a formal analysis. When publishing the results on a formal analysis in a scientific paper, details of the model usually have to be skipped due to lack of space, and often the lessons learnt from modelling are not discussed since they are not the main focus of the paper.

The MARS workshops aim at discussing exactly these unmentioned lessons. Examples are:

- Which formalism is chosen, and why?
- Which abstractions have to be made and why?
- How are important characteristics of the system modelled?
- Were there any complications while modelling the system?
- Which measures were taken to guarantee the accuracy of the model?

MARS emphasises modelling over verification. In particular, we invited papers that present full models of real systems, which may lay the basis for future comparison and analysis. The workshop thus intends to bring together researchers from different communities that all aim at verifying real systems and are developing formal models for such systems. An aim of the workshop is to present different modelling approaches and discuss pros and cons for each of them.

Full specifications of the contributed models are available at <http://mars-workshop.org> — often including executable models — so that their quality can be evaluated. Alternative formal descriptions are encouraged, which should foster the development of improved specification formalisms.

The aim of the VPT workshop series is to provide a forum where people from the areas of program transformation and program verification can fruitfully exchange ideas and gain a deeper understanding

of the interactions between those two fields. Recent research in those fields show many examples of mutual interactions and dependencies.

- In one direction, methods and tools developed in the field of program transformation, such as partial deduction, partial evaluation, fold/unfold transformations and supercompilation, have all been applied with success for the verification of systems, and in particular, the verification of infinite state and parameterised systems.
- In the other direction, methods developed in program verification, such as model checking, abstract interpretation, SAT and SMT solving, and automated theorem proving have been used to enhance program transformation techniques, thereby making these techniques more powerful and useful in practice.

There are two invited speakers:

- Xavier Leroy (INRIA, France) presents recent work on the Lustre reactive language (the core of the SCADE and to some extent Simulink modelling languages) and its verified compilation to imperative code.
- Kostis Sagonas (Uppsala University, Sweden) gives a tutorial and summary of recent work on progress on algorithms for stateless model checking.

We wish to express our gratitude to the authors who submitted papers, the speakers, the invited speakers and the programme committee members. Thanks are also due to the EasyChair organisation for supporting the various tasks related to the selection of the contributions and also EPTCS and arXiv for publishing and hosting the proceedings.

The body of this volume contains seven contributions, five for MARS and two for VPT. The submitted papers were carefully refereed by at least three members of the respective programme committee. The topics include:

- the Dolev-Yao intruder model, and MAA and TLS cryptographic protocols,
- the PageRank algorithm (imperative versus MapReduce Implementation),
- distributed cache coherent architectures, and
- avionic and railway transportation systems.

*John P. Gallagher
Rob van Glabbeek
Wendelin Serwe*

MARS 2018 Programme Committee:

Marsha Chechik	(University of Toronto, Canada)
Ansgar Fehnker	(University of Twente, The Netherlands)
Rob van Glabbeek (co-chair)	(Data61, CSIRO, Australia)
Jan Friso Groote	(Eindhoven University of Technology, The Netherlands)
Keijo Heljanko	(Aalto University, Finland)
Holger Hermanns	(Saarland University, Germany)
Eric Jenn	(IRT Saint Exupry, France)
Marjan Sirjani	(Reykjavik University, Iceland)
Wendelin Serwe (co-chair)	(INRIA, France)
Pamela Zave	(AT&T Laboratories, New Jersey, USA)

VPT 2018 Organisers:

John Gallagher	(Roskilde University, Denmark and IMDEA Software Institute, Spain)
Alexei Lisitsa	(The University of Liverpool, UK)
Andrei P. Nemytykh	(Program Systems Institute of RAS, Russia)

VPT 2018 Program Committee:

Emanuele De Angelis	(University G.d' Annunzio of Chieti-Pescara, Italy)
Olivier Danvy	(Yale-NUS College, Singapore)
John Gallagher (chair)	(Roskilde University, Denmark and IMDEA Software Institute, Spain)
Robert Glück	(University of Copenhagen, Denmark)
Geoff W. Hamilton	(Dublin City University, Republic of Ireland)
Bishoksan Kafle	(The University of Melbourne, Australia)
Julia Lawall	(INRIA Paris, France)
Alexei Lisitsa	(The University of Liverpool, UK)
Andrei P. Nemytykh	(Program Systems Institute of RAS, Russia)
Maurizio Proietti	(IASI-CNR, Rome, Italy)
C. R. Ramakrishnan	(Stony Brook University, USA)
Kostis Sagonas	(Uppsala University, Sweden)
Hirohisa Seki	(Nagoya Institute of Technology, Japan)