Editorial

The need for theories, languages, methods, and tools through which software applications can be developed that meet the highest standards of quality keeps rising as the dependency of both economical and social progress on software-intensive systems increases. In order to meet this challenge, researchers and practitioners from academia, industry and government need to get together to advance the state of the art in formal methods, to scale up their application in software industry and to encourage their integration with practical engineering methods.

This special issue originates from the eighth IEEE International Conference on Software Engineering and Formal Methods held in Pisa, Italy, from the 13th to the 18th of September, 2010. After the conference, the authors of the highest-ranked papers and the invited speakers were invited to submit significantly extended papers for inclusion in this special issue; eight papers were submitted out of which five were accepted and have been included here.

“A Formal Approach to Adaptive Software: Continuous Assurance of Non-Functional Requirements”, by A. Filieri, C. Ghezzi and G. Tamburrelli, focuses on non-functional requirements and offers an approach based on formal (probabilistic) models and support tools that enable a holistic view of the design and run-time management of adaptive software systems.

“Reo + mCRL2: A Framework for Model-Checking Data flow in Service Compositions”, by N. Kokash, C. Krause and E. de Vink, addresses the verification of Reo using the mCRL2 toolset, which is based on a mapping of various semantic models previously developed for Reo (constraint automata, timed constraint automata, coloring semantics and the newly developed action constraint automata) to the process algebraic specification language of mCRL2.

“Assertion-based Slicing and Slice Graphs”, by J. Barros, D. da Cruz, P. Henriques and J. Sousa Pinto, shows how the forward propagation of preconditions and the backward propagation of postconditions can be combined in a new slicing algorithm that is more precise than the existing specification-based algorithms.

“Cut Set Analysis using Behavior Trees and Model Checking”, by P. A. Lindsay, N. Yatapanage and K. Winter, proposes the use of behavior trees and model checking to automate cut set analysis, i.e., the identification of combinations of component failures that can lead to hazardous system failures.

“Scalable Context-dependent Analysis of Emergency Egress Models”, by M. Massink, D. Latella, A. Bracciali, M. D. Harrison and J. Hillston, presents results on a novel usage of a scalable technique for the analysis of collective dynamic behaviour of people in pervasive environments.

“Using Fold-in and Fold-out in the Architecture Recovery of Software Systems”, by M. Risi, G. Scanniello and G. Tortora, presents an approach based on information retrieval and clustering techniques to automate the architecture recovery process of software systems.

We are grateful to 'Formal Aspects of Computing for allowing us to publish this collection of papers, and John Cooke in particular for his help throughout the editorial process. We are also grateful to all Program Committee members of SEFM 2010 and reviewers involved in selecting the papers included in this special issue.

José Luiz Fiadeiro
Stefania Gnesi
Tom Maibaum

Published online 17 February 2012