Fundamenta Informaticae 108 (2011) i-iii
DOI 10.3233/FI-2011-410
IOS Press

Application of Concurrency to System Design, the Eighth Special Issue

Preface

The ninth International Conference on the Application of Concurrency to System Design (ACSD) was held in July 2009 in Augsburg, Germany. Following a tradition, Fundamenta Informaticae publishes a special issue with revised and extended versions of a selection of the best papers from ACSD. The current issue is the eighth special issue devoted to ACSD.

ACSD serves as a forum for disseminating theoretical results with application potential and advanced methods and tools for the design of complex concurrent systems. The conference aims at cross-fertilizing both theoretical and applied research on the following topics:

- design methods, tools and techniques based on models of computation and concurrency (dataflow models, communicating automata, Petri nets, process algebras, state charts, MSCs, etc.), (performance) analysis, verification, testing and synthesis;
- hardware / software co-design, platform-based design, component-based design, refinement techniques, hardware / software abstractions, co-simulation and verification;
- synchronous and asynchronous design, asynchronous circuits, globally asynchronous locally synchronous systems, interface design, multi-clock systems, functional and timing verification;
- concurrency issues in systems-on-chip, massively parallel architectures, networks on chip, task and communication scheduling, resource, memory and power management, fault-tolerance and quality-of-service issues;
- (industrial) case studies of general interest, gaming applications, consumer electronics and multimedia, automotive systems, (bio-)medical applications, Internet and grid computing, etc.;
- concurrency issues in *ad hoc*, mobile and wireless networking, sensor networks, communication protocols, cross-layer optimization, resource and power management, fault-tolerance, concurrency-related security issues;
- business process modeling, simulation and verification, (distributed) workflow execution, business process (de-)composition, inter-organizational and heterogeneous workflow systems, computer-supported collaborative work systems, web services; and

- synthesis and control of concurrent systems, (compositional) modeling and design, (modular) synthesis and analysis, distributed simulation and implementation, (distributed) controller synthesis, adaptive systems, supervisory control.

The seven papers in this special issue have been selected from the papers presented at the ninth ACSD conference. ACSD 2009 had over 30 submissions. Every submission was reviewed by four specialists in the subject area of the paper. Eighteen submissions (including two tool papers) were selected for presentation at the conference and publication in the IEEE CS Press proceedings. The selection for this special issue was based on the reviews of the ACSD program committee.

The authors of the selected papers were invited to significantly expand their papers (by at least 25%) to include new results and insights, and to revise them, taking into account the ACSD reviews and comments received during the conference. The extended papers have each been reviewed, usually by three reviewers, who were asked to check if the comments made by the ACSD reviewers had been taken into account. None of the papers were rejected in this round of reviewing, however some of the authors were asked to further revise and expand their papers, based on the reviews. These papers then underwent another round of reviews and revision

This special issue presents the papers according to the alphabetical order of the last names of the first authors.

In the first paper, Josep Carmona, Jorge Júlvez, Jordi Cortadella, and Michael Kishinevsky present a new technique for implementing latency intensive (a.k.a. synchronous elastic) circuits, which are designed in the standard synchronous style but built as distributed functional units connected by buffers able to cope with communication delays. While such circuits are usually implemented by adding many handshaking signals, the authors propose a static scheduling technique that allows for completely distributed control, eliminating the need and overhead of handshaking.

The next paper, by Silvia Crafa, Francesco Ranzato and Francesco Tapparo presents two methods to synthesize Petri nets, in particular, place/transition nets from finite partial languages or sets of labeled partial orders. Partial orders truly represent the concurrency among the events of a system. The work is the first to provide an effective solution to the synthesis problem for place/transition nets when starting from their partial-order behavior. The techniques have a wide application potential, ranging from hardware design to workflow management.

In the third paper, Niels Lohmann and Karsten Wolf improve the concept of an operating guideline developed earlier. An operating guideline of a service characterizes all its strategies, i.e. all those user behaviours that ensure that in combination with the service, deadlocks are avoided and either successful termination is reached or repeated communication takes place (responsiveness). The operating guideline is an automaton where the states are annotated with Boolean formulae, and a user behaviour is a strategy if and only if a kind of simulation by the operating guideline exists such that the formulae are satisfied in some sense. The new contribution is that it suffices to store two extra bits per state of the automaton instead of a formulae; the simulation is adapted accordingly. A number of experiments give evidence of the significant space reduction this ensues.

Andrey Mokhov, Victor Khomenko and Alex Yakovlev address fault diagnosis by static and dynamic observers in the fourth paper. The basic problem of diagnosability is the question of whether an observer is capable of identifying a fault given a set of sensed events. The main results of the paper are related to the minimization of the number of sensed events required for diagnosability for a static observer that cannot change the set of observed events during execution, and to the synthesis of dynamic observers that can switch sensors on and off during execution, thus dynamically changing the set of observed events. Solutions to these problems are presented, and several complexity results are proven.

In the fifth paper, Dumitru Potop Butucaru, Robert de Simone, Yves Sorel, and Jean-Pierre Talpin present a technique for transforming a synchronous specification into a globally asynchronous one with identical input/output behavior. Their basic goal is to eliminate the need (and hence the cost of transmitting) for pure synchronization messages to keep the asynchronous system's behavior consistent with its original synchronous specification.

Jean-Baptiste Raclet, Eric Badouel, Albert Benveniste, Benoit Caillaud, Axel Legay and Roberto Passerone unify interface automata and modal specifications with a language-based approach called modal interface theory. In their theory, conjunction and quotiening are the most important operators and special attention is payed to specifications with different alphabets. Interface automata, modal specifications and modal interface theory are compared in detail, and an error in some earlier paper by other authors is corrected.

Finally, in the last paper, Karsten Wolf, Christian Stahl, Daniela Weinberg, Janine Ott and Robert Danitz contribute to the issue of service discovery in Service Oriented Architectures: how can a user find a suitable published service? Suitability is defined here by weak termination, i.e. a service is suitable if under some fairness assumption the composition of user and service terminates successfully. The authors suggest to precompute so-called fragments in the publication phase such that time can be saved in the time-critical find phase when checking the state space of the composed system for weak termination. Also, specific reduction rules for fragments are given, and the methodology is validated with experiments.

We thank the authors of the seven papers of this special issue for their careful consideration and implementation of the reviewers' comments to produce interesting and useful contributions. We are grateful to the reviewers, both of the conference versions of the papers and of the extended journal versions published in this issue, for their careful reviews and valuable suggestions. Their commitment has been invaluable in ensuring the high quality of the papers in this issue.

Editors

Stephen A. Edwards

Department of Computer Science, Columbia University
New York, NY 10027 USA
sedwards@cs.columbia.edu

Ryszard Janicki

Department of Computing and Software, McMaster University
Hamilton, Ontario, Canada L8S 4K1
janicki@cas.mcmaster.ca

Walter Vogler

Institut für Informatik, Augsburg University
86135 Augsburg, Germany
vogler@informatik.uni-augsburg.de