Scientific Annals of Computer Science vol. 21, 2011, pp. 1–4

Concurrency and Interaction in Complex Systems

Sonia BEN MOKHTAR, Simon BLIUDZE, Roberto BRUNI
Alexandra SILVA and Angelo TROINA

Abstract

The authors emphasize the actual relevance and need of formal methods for the advancements of complex systems, and briefly present the other papers contained in this issue.

Mathematical techniques, nowadays widely referred to as "formal methods", have been advocated for and used in computer science since its beginnings. Historically, concurrent and distributed systems formed one the most important application domains for formal methods, due, in particular, to their inherent complexity. The importance of formal methods for the design and analysis of concurrent and distributed systems has been considerably boosted by the comparatively recent evolution of technologies, in particular, with the emergence of affordable multi-core computing platforms, web applications, service-oriented computing and ubiquitous computing targeting mobile devices. It is not incidentally that theoretical computer science contributed to a novel understanding of concurrency, distribution, interaction and modularity in complex systems, not available before in natural and social sciences where such phenomena intrinsically arise. One plausible explanation is that, citing Turing award winner and prominent British scientist Robin Milner (1934–2010): *"In natural science, concepts arise from the urge to understand observed phenomena. But in computer science, concepts arise as distillations of our design of systems."*

The current areas of interest of the European Commission research initiative in ICT within the Future and Emerging Technologies Proactive Framework (FP7)[1], ranging from Quantum ICT and Fundamentals of Collective Adaptive Systems to Dynamics of Multi-level Complex Systems and

[1] See http://cordis.europa.eu/fp7/ict/fet-proactive/areas_en.html.

Neuro-Bio-Inspired Systems are the best witnesses of the actual relevance and need of formal methods for the advancements of complex systems.

The use of formal methods in the domain of concurrent systems covers a broad spectrum of applications, which differ both in the number of agents and in the nature of interaction involved. The size of the system can go from only a few processes, as in the plant control case below, through several dozens of agents, as in complex distributed systems such as web-services, to a multitude of communicating entities, as for example, in many-core systems designed for high-performance computing or in sensor networks and in models for the description and analysis of biological systems.

Similarly, the nature of the interaction differs depending on the application domain. It can take place over dedicated (local) networks, which can be wired or wireless, or over channels shared with other systems, as in the case of web-services; the communication links can be fixed or established dynamically; the exchanged data can range from first-order values to symbolic and higher-order entities or from low-security level (public, unsecure) data to high-security level (confidential, non disclosable) information.

To some degree, the spectrum described above is represented in the papers comprising this volume.

The paper *"Desynchronisability of (Partial) Synchronous Closed Loop Systems"* by H. Beohar and P. Cuijpers discusses the properties of control systems, where a controller (or supervisor) interacts with a plant. The authors present some sufficient conditions for the desynchronizability of synchronous closed loop systems and then extend their results to partial synchronous closed loop systems, in which external actions are admitted. This is achieved by converting a synchronous closed loop system consisting of a plant and supervisor into an equivalent asynchronous system where the plant and supervisor communicate via messages. The correctness of the transformation is phrased in terms of the existence of a branching bisimulation between the original synchronous system and the transformed asynchronous system. The aim is not to synthesize a supervisor but rather to characterize when an existing correct system can be made asynchronous without affecting behaviour.

The paper *"Modular Verification of Interactive Systems with an Application to Biology"* by P. Drábik, A. Maggiolo-Schettini and P. Milazzo proposes an automata-based formalism for the description of biological systems and a modular verification technique that allows properties expressed in the universal fragment of CTL to be verified on suitably chosen fragments of models. rather than on whole models. Verification of properties is performed by using the NuSMV model checker and the modular verification technique proposed can verify properties for chosen fragments of the model in shorter times than those necessary to verify the same properties in the whole model.

The paper *"Programming in Biomolecular Computation: Programs, Self-Interpretation and Visualisation"* by L. Hartmann, N.D. Jones, J.G. Simonsen and S.B. Vrist introduces a programmable (and biologically plausible) model of computation. Its functioning is defined by a single and relatively small set of chemical-like reaction rules. The model is *stored-program*: programs are the same as data; *universal*: all computable functions can be computed (in natural ways and without arcane encodings of data and algorithm); *uniform*: new hardware is not needed to solve new problems; and *Turing complete* in a strong sense: a universal algorithm exists, that is able to execute any program, and is not asymptotically inefficient.

The paper *"Designing, Capturing and Validating History-Sensitive Security Policies for Distributed Systems"* by A.M. Hernandez, F. Nielson and H. Riis-Nielson addresses the security policies in distributed systems. More precisely, this paper introduces the possibility of expressing historical information, thus allowing to express policies that refer both to the future and to the past. The policies are meant to avoid information leakeage: each location has a security level and neither reading information of higher security level nor writing information to a location of lower security level is allowed. The authors also propose a logic for reasoning about a "global" security property of a distributed system. This logic allows validating such a property against the labelled transition system modelling the evolution of the system.

The paper *"An SCA-based Approach for Social and Pervasive Communications in Home Environments"* by R. Mélisson, D. Romero, R. Rouvoy and L. Seinturier addresses the issues of service mobility, event flow and user identity in pervasive home environments. More precisely, the authors propose three extensions to the Frascati platform (an SOA platform for the SCA standard). The first extension allows defining pervasive bindings, which enable service discovery using the UPnP service discovery protocol. The second extension allows defining social bindings, which enable event dissemination in home environments using the Twitter micro-blogging service. Finally, the third extension is a mechanism enabling the management of heterogeneous authentication processes according to home devices and services.

The papers assembled in this volume extend previous work presented by their respective authors at ICE, CS2Bio and CAMPUS workshops – satellite events of the DisCoTec'10 federated conference that took place in June 2010 in Amsterdam, The Netherlands. The Proceedings of ICE'10 have been published in the volume 38 of the open access, on-line series *Electronic Proceedings in Theoretical Computer Science*. The Proceedings of CS2Bio'10 have been published in the volume 268 of the on-line (ScienceDirect) series *Electronic Notes in Theoretical Computer Science*. The Proceedings of CAMPUS'10 have been published in the volume 28 of the open access, on-line series *Electronic Communication of the EASST*.

We would like to thank all the members of the Program Committee and reviewers of ICE, CS2Bio and CAMPUS workshops, who helped us in the selection of the papers and who helped the authors to improve their contributions in several ways. We want to thank all the authors who contributed to this volume for the quality of their work and for the efforts spent in the preparation and revision of their papers. We want to warmly thank Gabriel Ciobanu, the Editor-in-Chief of Scientific Annals of Computer Science, for helping us to prepare this issue, for his patience and constant support during the publication process.