Preface for the special issue on tool papers of the 23rd International Conference on Coordination Models and Languages, COORDINATION 2021

(Article begins on next page)

Preface for the Special Issue on Tool Papers of the 23rd International Conference on Coordination Models and Languages, COORDINATION 2021

Giorgio Audrito

Università di Torino, Italy

Omar Inverso

Gran Sasso Science Institute, Italy

Hugo Torres Vieira

Evidence Srl, Italy

This special issue reports on the tools selected out of the tool track of the 23rd International Conference on Coordination Models and Languages (COORDINATION 2021), originally programmed in the University of Malta and held online, on June 14-18, 2021. Since 2019 the conference welcomes submissions describing technological artefacts, including innovative prototypes, for modelling, analysing, simulating or testing systems in the scope of the research topics of COORDINATION. The prescribed format of the submissions includes both a short abstract describing the tool and a short video to showcase the presentation to be carried out at the conference, an innovative feature that has been highly appreciated by the members of the program committee.

There was a total of 8 submissions to the tool track (about 25% of the total submissions), and 7 were accepted for publication (about 88% of the total of tool track submissions and 40% of the number of overall accepted contributions) as full papers extending the submitted version. For this special issue, the authors of the more mature contributions were invited to submit their

Email addresses: `giorgio.audrito@unito.it` (Giorgio Audrito), `omar.inverso@gssi.it` (Omar Inverso), `hugo.t.vieira@evidence.eu.com` (Hugo Torres Vieira)

tool as an Original Software Publication, a novel format that attends to the need of disseminating valuable implementation efforts. A total of 6 invitations were issued to COORDINATION 2021 authors, and all 6 contributions are included in this special issue. Each tool submission was reviewed by three experts, one of which chosen from the COORDINATION PC members who reviewed the conference version, and the remaining two among recognised experts in the specific area of the submission, including also reviewers suggested by the authors themselves.

The contributions included in this special issue are the following:

- Lorenzo Bacchiani, Mario Bravetti, Marco Giunti, João Mota, and António Ravara. *A Java typestate checker supporting inheritance.* This contribution presents a tool that allows to perform typestate analysis over Java programs, recognized by the reviewers as an easy to use tool and including well documented code. The accompanying paper includes an illustrative running example that allows the reader to go through the usage steps while grasping underlying concepts.

- Davide Basile and Maurice H. ter Beek. *Contract Automata Library.* Contract automata facilitate the specification, composition, and synthesis of behavioural contracts. This contribution presents Contract Automata Library (CATLib), a software API supporting the specification of contract automata. The accompanying paper discusses the library architecture, details a usage example and presents recent qualitative improvements of the software.

- Riccardo Bianchini and Francesco Dagnino. *QueryAGT: asynchronous global types in co-logic programming.* QueryAGT aims at high-level specification of protocols and analysis thereof. It implements in co-logic programming a novel formalism of global types with asynchronous communication. The user defines sessions and global types in a simple language with primitives for type checking. The tool can check different safety and liveness properties, either in interactive or batch mode.

- Christian Bartolo Burlò, Adrian Francalanza, Alceste Scalas, Catia Trubiani and Emilio Tuosto. *PSTMonitor: Monitor Synthesis from Probabilistic Session Types.* This contribution presents PSTMonitor, a tool for the run-time verification of quantitative specifications of message-passing applications. Crucially, PSTMonitor uses probabilistic

session types to detect executions that deviate from expected probabilistic behaviour. The accompanying paper presents the architecture and operation of the tool, while assessing the runtime overheads it induces.

- Eva Maria Kuehn. *The Peer-Model Tool-Chain.* This paper presents an open-source implementation of the blackboard-based Peer Model tool chain. The tool-chain aims at supporting model-driven development of coordination software and identifying potential issues at an early stage. A demo video linked from the accompanying paper is available to further illustrate the proposed methodology.

- Rudolf Schlatte, Einar Broch Johnsen, Eduard Kamburjan and S. Lizeth Tapia Tarifa. *The ABS Simulator Toolchain.* This contribution presents an executable specification language, for behavioural modelling of distributed, time- and resource- sensitive communicating systems. The language provides asynchronous method calls, a resource model, and support for software product lines. The accompanying paper includes a review of several case studies available online.