1. Record Nr. UNINA9910495155003321

Titolo	Formal methods for industrial critical systems : 26th international conference, FMICS 2021, Paris, France, August 24-26, 2021 : proceedings / / Alberto Lluch Lafuente, Anastasia Mavridou (editors)
Pubbl/distr/stampa	Cham, Switzerland : , : Springer, , [2021] ©2021
ISBN	3-030-85248-2
Descrizione fisica	1 online resource (253 pages)
Collana	Lecture Notes in Computer Science ; ; v.12863
Disciplina	004.0151
Soggetti	Formal methods (Computer science) Software engineering Computer programs - Verification
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Intro -- Preface -- Organization -- Haunting Tales of Applied Formal Methods from Academia and Industry (Abstract of Invited Talk) -- Contents -- Verification -- Verification of Co-simulation Algorithms Subject to Algebraic Loops and Adaptive Steps -- 1 Introduction -- 2 Background -- 2.1 Simulation Units -- 2.2 Co-simulation Algorithms -- 2.3 Correct Co-simulation Algorithms -- 3 Related Work -- 4 Verifying Complex Co-simulation Algorithms -- 4.1 Verifying an Algorithm Using UPPAAL -- 4.2 Verifying Complex Simulation Scenarios in UPPAAL -- 4.3 Debugging Algorithm Errors -- 5 Validation -- 5.1 Motivation Example -- 5.2 Complex Scenario -- 6 Concluding Remarks -- References -- Automated Verification of Temporal Properties of Ladder Programs -- 1 Introduction -- 2 Introduction to Ladder Programming -- 3 Translation of Ladder Programs to WhyML -- 3.1 The Why3 Environment -- 3.2 Translation of Ladder Codes -- 3.3 The Ladder Loop, and the Encoding of Timing Charts -- 4 Implementation and Experimental Results -- 4.1 Overview of the Approach -- 4.2 Results on Correct Code -- 4.3 Results on Incorrect Code -- 5 Discussions, Related Work and Future Work -- References -- Spatial Model Checking for Smart Stations -- 1 Introduction and Outline -- 2 Industrial Context and Case Study: Station Lighting -- 3 Challenges in

User-Centric Design of Smart Stations -- 4 Methodology -- 4.1 Spatial Model Checking -- 4.2 Statistical Spatio-Temporal Model Checking -- 5 Conclusion and Outlook -- References -- Program Safety and Education -- Parametric Faults in Safety Critical Programs -- 1 Introduction -- 2 Background -- 3 Identifying Incorrect Parameters -- 4 Case Study -- 5 Discussion -- 6 Related Works -- 7 Conclusion -- References -- Modular Transformation of Java Exceptions Modulo Errors -- 1 Introduction -- 2 Background -- 2.1 Abrupt Termination -- 2.2 VerCors.
3 Related Work -- 4 Semantics of Exceptions -- 4.1 Errors and Sources of Errors -- 4.2 Ideal Semantics -- 4.3 Semantics Modulo Errors -- 5 The finally Encoding Problem -- 5.1 Candidate Encodings -- 6 Evaluation -- 6.1 Common Exception Patterns in Commercial Software -- 6.2 Verification with VerCors -- 7 Discussion -- 7.1 Backend Requirements -- 7.2 Performance -- 8 Conclusion -- References -- On Education and Training in Formal Methods for Industrial Critical Systems -- 1 Introduction -- 2 Terminology -- 3 Roles in Formal Methods for Industrial Critical Systems: [Engineer] and [Engineer] -- 3.1 FMICS Roles and Activities -- 3.2 Consequences on Education and Training -- 4 Learning Objectives: vs. -- 5 Curriculum and Course Construction -- 6 Exemplary Implementation -- 7 Conclusion -- References -- (Event-)B Modeling and Validation -- Improving SMT Solver Integrations for the Validation of B and Event-B Models -- 1 Introduction -- 2 Former Z3 Integration -- 2.1 High-Level Translation -- 2.2 Workflow -- 3 New Z3 Integration -- 3.1 High-Level Translation -- 3.2 New Workflow -- 4 Empirical Evaluation -- 4.1 Weaknesses of the Integration of Z3 -- 4.2 Strengths of the Integration of Z3 -- 4.3 Symbolic Model Checking -- 5 Related Work -- 6 Future Work -- 7 Conclusion -- References -- Standard Conformance-by-Construction with Event-B -- 1 Introduction -- 2 Certification and Conformance -- 3 Event-B -- 3.1 Contexts and Machines (Tables1b and 1c) -- 3.2 Event-B Extensions with Theories -- 4 Case Study: ARINC 661 + Multi-purpose Interactive Application -- 4.1 ARINC 661 Standard Specification: An Extract -- 4.2 Multi-purpose Interactive Application and Weather Radar System -- 5 Standards Formalised as Ontologies ((1) on Fig.2) -- 6 Our Approach -- 6.1 Domain Standards as Ontology-Based Theories ((2) on Fig.2) -- 6.2 Standard Theory Instantiation ((3) on Fig.2).
6.3 Model Annotation for Conformance ((4) on Fig.2) -- 7 Standard Conformance-by-Construction: The Case of ARINC 661 -- 7.1 ARINC 661 Standard Formalisation ((2) on Fig.2) -- 7.2 System-Specific Concepts Describing WXR Widgets ((3) on Fig.2) -- 7.3 Annotated Event-B Model of WXR Application ((4) on Fig.2) -- 8 Assessment -- 9 Conclusion -- References -- Formal Analysis -- Randomized Reachability Analysis in Uppaal: Fast Error Detection in Timed Systems -- 1 Introduction -- 2 Randomized Reachability Analysis -- 3 New Results on Herschel-Planck -- 4 More Schedulability -- 5 Gossiping Girls -- 6 Scalability Experiments -- 7 Conclusion -- 8 Future Work -- References -- Verifying the Mathematical Library of an UAV Autopilot with Frama-C -- 1 Introduction -- 2 The Paparazzi Autopilot -- 3 Proving the Absence of Runtime Errors -- 4 Functional Verification Using Automatic Provers -- 5 Functional Verification Using Interactive Provers -- 6 Conclusion -- References -- Formal Analysis of the UNISIG Safety Application Intermediate Sub-layer -- 1 Introduction -- 2 Background -- 3 The Model -- 4 The Analysis -- 5 Conclusion -- References -- Tools -- ProB2-UI: A Java-Based User Interface for ProB -- 1 Introduction and Motivation -- 2 Features of ProB2-UI -- 3 Related Work -- 4 Conclusion -- References -- Intrepid: A Scriptable

and Cloud-Ready SMT-Based Model Checker -- 1 Introduction -- 2 Constructing Models -- 2.1 Translating Industrially-Relevant Models -- 3 Simulating Models -- 4 Model Checking -- 4.1 A Comparison of the Engines -- 5 Sample Applications -- 5.1 Equivalence Checking for Clock-Gating -- 5.2 Automated Test Generation of MC/DC -- 6 A REST API for Model Checking -- 7 Conclusion -- References -- Merit and Blame Assignment with Kind 2 -- 1 Introduction -- 2 Running Example -- 3 The New Features -- 4 Implementation Details -- References. Test Generation and Probabilistic Verification -- PSY-TaLiRo: A Python Toolbox for Search-Based Test Generation for Cyber-Physical Systems -- 1 Introduction -- 2 Architecture -- 3 Interface -- 3.1 System Under Test (SUT) -- 3.2 Specifications -- 3.3 Optimizers -- 3.4 Options -- 4 Examples -- 4.1 MATLAB/Simulink -- 4.2 PX4 -- 5 Conclusions -- References -- Probabilistic Verification for Reliability of a Two-by-Two Network-on-Chip System -- 1 Introduction -- 2 Motivation -- 3 Concrete Formal Model for NoC -- 4 Need for Abstraction -- 4.1 Predicate Abstraction to Simplify Complex Data Structures -- 4.2 Probabilistic Choice Abstraction -- 4.3 Boolean Queue Abstraction -- 5 Results -- 5.1 Every Other Cycle Flit Injection -- 5.2 Burst Flit Injection -- 5.3 Minimizing PSN with Flit Generation Pattern -- 5.4 Results Summary and Discussion -- 6 Conclusion -- References -- Author Index.