ANAIS PROCEEDINGS

CBSOFT * 2015

BRAZILIAN CONFERENCE ON SOFTWARE: THEORY AND PRACTICE

BELO HORIZONTE

SBMF 2015

18th BRAZILIAN SYMPOSIUM ON FORMAL METHODS

CBSOFT.ORG

Sponsors:

ThoughtWorks · AvenueCode · Google Brasil · take.net · RARO LABS · Avanti · IBM

Promotion:

SBC Sociedade Brasileira de Computação

Organizing Institutions:

UFMG · PUC Minas · CEFET-MG

CNPq Conselho Nacional de Desenvolvimento Científico e Tecnológico · CAPES · FAPEMIG

SBMF 2015

18th BRAZILIAN SYMPOSIUM ON FORMAL METHODS
September 21-22, 2015
Belo Horizonte – MG, Brazil

ANAIS | *PROCEEDINGS*

COORDENADORESDO COMITÊ DE PROGRAMA DO SBMF 2015
PROGRAM COMMITTEE CHAIRS OF SBMF 2015
Márcio Cornélio (UFPE, Brazil)
Bill Roscoe (University of Oxford, UK)

COORDENADORES GERAIS DO CBSOFT 2015 |*CBSOFT 2015 GENERAL CHAIRS*
Eduardo Figueiredo (UFMG)
Fernando Quintão (UFMG)
Kecia Ferreira (CEFET-MG)
Maria Augusta Nelson (PUC-MG)

REALIZAÇÃO | *ORGANIZATION*
Universidade Federal de Minas Gerais (UFMG)
Pontíficia Universidade Católica de Minas Gerais (PUC-MG)
Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)

PROMOÇÃO |*PROMOTION*
Sociedade Brasileira de Computação | Brazilian Computing Society

APOIO | *SPONSORS*
CAPES, CNPq, FAPEMIG, Google, RaroLabs, Take.net,
ThoughtWorks, AvenueCode, AvantiNegócios e Tecnologia.

APRESENTAÇÃO

O Simpósio Brasileiro de Métodos Formais (SBMF) 2015 é um evento que visa disseminar o desenvolvimento e uso de métodos formais para a construção e averificação de sistemas computacionais. É um evento consolidado, com reputação internacional. Sua primeira edição ocorreu no ano de 1998, indo para sua 18ª edição no ano de 2015.

Este volume contém os resumos dos artigos apresentados no 18º Simpósio Brasileiro de Métodos Formais (SBMF). A conferência ocorreu em Belo Horizonte, Brasil, como parte da 6ª Conferência Brasileira de Software: Teoria e Prática (CBSoft), que reúne outros três simpósios: Simpósio Brasileiro de Engenharia de Software (SBES), Simpósio Brasileiro de Linguagens de Programação (SBLP) e Simpósio Brasileiro de Componentes, Arquiteturas e Reutilização de Software (SBCARS).

O SBMF trouxe dois palestrantes convidados: Adenilso Simão(ICMC/USP, São Carlos, SP, Brasil) e Sumit Gulwani(Microsoft Research, EUA).

Um total de 12 artigos foram apresentados na conferência. Eles foram selecionados entre 25 submissões com autores de 10 países: África do Sul, Alemanha, Brasil, Canadá, Estados Unidos, França, Holanda, Inglaterra, Luxemburgo e Suécia.

O Comitê de Programa foi constituído por 41 membros da comunidade nacional e internacional de Métodos Formais. As deliberações do Comitê de Programa e a preparação dos Anais foram feitos com o auxílio do sistema EasyChair. Gostaríamos de agradecer ao Comitê de Programa e aos revisores adicionais pelas suas contribuições na avaliação das submissões e nas sugestões de melhorias. Em particular, agradecemos a Christiano Braga, Juliano Iyoda e Rohit Gheyi, co-coordenadores de edições anteriores do SBMF, que estiveram sempre disponíveis para nos ajudar e compartilhar de sua experiência.

A Sociedade Brasileira de Computação (SBC) anualmente promove o SBMF. Somos gratos aos Coordenadores Gerais do CBSoft 2015, que trabalharam arduamente para que tudo funcionasse bem, Eduardo Figueiredo (UFMG), Fernando Quintão (UFMG), Kecia Ferreira (CEFET-MG) e Maria Augusta Nelson (PUC-MG). Em nome do Comitê de Programa, damos boas vindas aos participantes do SBMF 2015 e desejamos a todos um simpósio prolífico e uma semana produtiva em Belo Horizonte.

Belo Horizonte, Setembro de 2015.

Márcio Cornélio
Bill Roscoe

FOREWORD

The Brazilian Symposium on Formal Methods (SBMF) 2015 is the eighteenth of a series of events devoted to the development, dissemination and use of formal methods for the construction of high-quality computational systems. It is now a well-established event, with an international reputation. The first edition occurred in 1998.

This volume contains the abstracts of the papers presented at SBMF 2015: the 18th Brazilian Symposium on Formal Methods. The conference was held in Belo Horizonte, Brazil, colocated with CBSoft 2015, the 6th Brazilian Conference on Software: Theory and Practice, which integrates other three symposia: Brazilian Symposium on Software Engineering (SBES), Brazilian Symposium on Programming Languages (SBLP), and Brazilian Symposium on Software Components, Architectures and Reuse (SBCARS).

The conference included two invited talks given by Adenilso Simão(ICMC/USP, São Carlos, SP, Brazil) and Sumit Gulwani(Microsoft Research, USA).

A total of 12 papers were presented at the conference. They were selected from 25 submissions that came from 10 different countries: Brazil, Canada, France, Germany, Luxembourg, Netherlands, South Africa, Sweden, United Kingdom, and United States of America.

The Program Committee was composed by 41 members from the national and international community of formal methods.The deliberations of the Program Committee were handled by EasyChair. We are grateful to the Program Committee, and to the additional reviewers, for their hard work in evaluating submissions and suggesting improvements. In particular, special thanks go to Christiano Braga, Juliano Iyoda and Rohit Gheyi, co-chairsof previous editions of SBMF, who were always available to help us and to share his experience and wisdom.

The Brazilian Computer Society (SBC) annually promotes SBMF. We are very thankful tothe general chairs of CBSoft 2015, Eduardo Figueiredo (UFMG), Fernando Quintão (UFMG), Kecia Ferreira (CEFET-MG), and Maria Augusta Nelson (PUC-MG), who made everything possible for the conference to run smoothly. On behalf of the Program Committee, we welcome all the SBMF 2015 attendees, and wish a fruitful symposium and a productive week in Belo Horizonte.

Belo Horizonte, September 2015

Márcio Cornélio
Bill Roscoe

COMITÊ DE ORGANIZAÇÃO | *ORGANIZING COMMITTEE*

CBSOFT 2015 GENERAL CHAIRS

Eduardo Figueiredo (UFMG)
Fernando Quintão(UFMG)
Kecia Ferreira (CEFET-MG)
Maria Augusta Nelson (PUC-MG)

CBSOFT 2015 LOCAL COMMITTEE

Carlos Alberto Pietrobon (PUC-MG)
Glívia Angélica Rodrigues Barbosa (CEFET-MG)
Marcelo Werneck Barbosa (PUC-MG)
Humberto Torres Marques Neto (PUC-MG)
Juliana Amaral Baroni de Carvalho (PUC-MG)

WEBSITE AND SUPPORT

Diego Lima (RaroLabs)
Paulo Meirelles (FGA-UnB/CCSL-USP)
Gustavo do Vale (UFMG)
Johnatan Oliveira (UFMG)

COMITÊ TÉCNICO | *TECHNICAL COMMITTEE*

COORDENADORES DO COMITÊ DE PROGRAMA| *PROGRAM COMMITTEE CHAIRS*

Márcio Cornélio (UFPE, Brazil)
Bill Roscoe(University of Oxford, UK)

COMITÊ DIRETIVO | *STEERING COMMITTEE*

Rohit Gheyi (UFCG, Brazil)
David Naumann (Stevens Institute of Technology, USA)
Juliano Iyoda (UFPE, Brazil)
Leonardo de Moura (Microsoft Research, USA)
Christiano Braga (UFF, Brazil)
Narciso Martí-Oliet (Universidad Complutense de Madrid, Spain)
Márcio Cornélio (UFPE, Brazil)
Bill Roscoe (University of Oxford, UK)

COMITÊ DE PROGRAMA | *PROGRAM COMMITTEE*

Aline Andrade (UFBA, Brazil)
Wilkerson Andrade (UFCG, Brazil)
Luis Barbosa (Universidade do Minho, Portugal)
Christiano Braga (UFF, Brazil)
Michael Butler (University of Southampton, UK)
Ana Cavalcanti (University of York, UK)
Simone Cavalheiro (UFPel, Brazil)
Márcio Cornélio (UFPE, Brazil), co-chair
Andrea Corradini (Università di Pisa, Italy)
Jim Davies (University of Oxford, UK)
David Deharbe (UFRN, Brazil)
Ewen Denney (RIACS/NASA, USA)
Clare Dixon (University of Liverpool, UK)
Adalberto Farias (UFCG, Brazil)
Rohit Gheyi (UFCG, Brazil)
Rolf Hennicker (Ludwig-Maximilians-Universitaet Muenchen, Germany)
Juliano Iyoda (UFPE, Brazil)
Peter Larsen (Aarhus University, Denmark)
Bruno Lopes (PUC-Rio, Brazil)
Anamaria Moreira (UFRJ, Brazil)
Patrícia Machado (UFCG, Brazil)
Narciso Martí-Oliet (Universidad Complutense de Madrid, Spain)
Tiago Massoni (UFCG, Brazil)

Ana Melo (USP, Brazil)
Alvaro Moreira (UFRGS, Brazil)
Alexandre Mota (UFPE, Brazil)
Leonardo Moura (Microsoft Research, USA)
Arnaldo Moura (Unicamp, Brazil)
Peter Müller (ETH Zürich, Switzerland)
David Naumann (Stevens Institute of Technology, USA)
Marcel Oliveira (UFRN, Brazil)
Jose Oliveira (Universidade do Minho, Portugal)
Leila Ribeiro (UFRGS, Brazil)
Bill Roscoe (University of Oxford, UK), co-chair
Augusto Sampaio (UFPE, Brazil)
Leila Silva (UFS, Brazil)
Adenilso Simão (USP, Brazil)
Sofiène Tahar (Concordia University, Canada)
Leopoldo Teixeira (UFPE, Brazil)
Heike Wehrheim (University of Paderborn, Germany)
Jim Woodcock (University of York, UK)

REVISORES EXTERNOS | *EXTERNAL REVIEWERS*

Luis Aguirre
Adilson Bonifácio
Alaeddine Daghar
Ahmed Hachani
Nuno Macedo
Alexandre Madeira
Isabel Pita
Valdivino Santiago Júnior
Gustavo Soares
Cláudia Tavares

PALESTRAS CONVIDADAS | *INVITED TALKS*

Applications of Formal Methods to Data Wrangling and Education

Sumit Gulwani (Microsoft Research, USA)

Abstract: Data is locked up in semi-structured formats such as spreadsheets, text/log files, webpages, pdf documents. Getting data out of these documents into structured formats that allow the data to be explored and analyzed is challenging. While data scientists spend 80% of their time cleaning data, programmatic solutions to data manipulation are beyond the expertise of 99% of end users who do not know programming. Programming by Examples (PBE) can make data wrangling a delightful experience for the masses. The first part of this talk will describe how formal methods can be used to address two key challenges in PBE: (a) efficient search algorithms to explore the huge state space of programs to find those that match the user-provided examples, and (b) effective ambiguity resolution techniques to deal with the inherent ambiguity in the examples. The second part of this talk will describe how formal methods can help automated two key tasks in Education, namely (a) problem generation and (b) feedback generation. I will illustrate this using recent research results that have been applied to various STEM subject domains including mathematics, programming, logic, and automata theory. These results advance the state-of-the-art in intelligent tutoring, and can play a significant role in enabling personalized and interactive education in both standard classrooms and MOOCs.

Sumit Gulwani is a principal researcher at Microsoft Research (in Redmond, USA), and an adjunct faculty in the Computer Science Department at IIT Kanpur (India). He has expertise in formal methods and automated program analysis and synthesis techniques. His recent research interests are in the cross-disciplinary areas of automating end-user programming and building intelligent tutoring systems. His programming-by-example work led to the Flash Fill feature of Microsoft Excel 2013 that is used by hundreds of millions of people. He was awarded the ACM SIGPLAN Robin Milner Young Researcher Award in 2014. He obtained his PhD in Computer Science from UC-Berkeley in 2005, and was awarded the ACM SIGPLAN Outstanding Doctoral Dissertation Award. He obtained his BTech in Computer Science and Engineering from IIT Kanpur, and was awarded the President's Gold Medal.

To test or not to test, that is formal question

Adenilso Simão(ICMC/USP, São Carlos, SP, Brazil)

Abstract: The demand of highly dependable software has greatly motivated the research of two important areas of software engineering, namely, formal methods and software testing. Both areas have matured considerably in the last years, to the point of being mainstream approaches in the development of critical systems. However, despite some fruitful exchange of ideas between them, formal methods and software testing have advanced somehow isolated from each other. In this talk, we review the achievements related to the combination of formal methods and software testing. We will discuss, for instance, how testing can be formal and how formal methods can be aided by testing. The main goal of this talk is to identify opportunities to strengthen the exchange between these two exciting and important areas.

Adenilso Simão received the BS degree in computer science from the State University of Maringa (UEM), Brazil, in 1998, and the MS and PhD degrees in computer science from the University of Sao Paulo (USP), Brazil, in 2000 and 2004, respectively. Since 2004, he has been a professor of computer science at the Computer System Department of USP. From August 2008 to July 2010, he has been on a sabbatical leave at Centre de Recherche Informatique de Montreal (CRIM), Canada. He has received best paper awards in several important conferences. He has also received distinguishing teacher awards in many occasions. His research interests include software testing and formal methods.

INDICE | *TABLE OF CONTENTS*

ABSTRACTS

Hard-wiring CSP Hiding: Implementing Channel Abstraction to Generate Verified Concurrent Hardware

F. J. S. MacárioandM. V. M. Oliveira

Departamento de Informática e Matemática Aplicada, UFRN, Brazil

Throughout the development of concurrent systems, complexity may easily grow exponentially yielding a very complex and error-prone process. By using formal languages like CSP we may simplify this task increasing the level of confidence on the resulting system. Unfortunately, such languages are not executable: the gap between the specification language and an executable program must be solved. In previous work, we presented a tool, csp2hc, that translates a considerable subset of CSP into Handel-C source code, which can itself be converted to produce files to program FPGAs. This subset restricts the use of data structures and CSP hiding. In this paper, we present an extension to csp2hc that includes sequences in the set of acceptable data structures and completely deals with the CSP hiding operator. Finally, we validate our extension by applying the translation approach to a industrial scale case study, the steam boiler.

Instantiation Reduction in Iterative Parameterised Three-Valued Model Checking

Nils Timm and Stefan Gruner
Department of Computer Science, University of Pretoria, South Africa
{ntimm, sgruner}@cs.up.ac.za

We introduce an enhanced approach to parameterised three-valued model checking (PMC) based on iterative parameterisation. The model is parameterised until it is precise enough for a definite verification result. Results from past iterations are reused to reduce the number of parameter instances in future iterations. Our approach is based on a SAT encoding. In the initial iteration we construct an over-approximation of all possible instances in later iterations. For this over-approximation we compute the set of all satisfying interpretations. All subsequent iterations are then accomplished by validating whether for each instance one of the precomputed interpretations is satisfying as well, which is less costly than solving each SAT instance from scratch. Our iterative parameterisation approach leads to a substantial speed-up of PMC.

Mobile CSP

Jim Woodcock , Andy Wellings, and Ana Cavalcanti
Department of Computer Science, University of York

We describe an extension of imperative CSP with primitives to declare new event names and to exchange them by message passing between processes. We give examples in Mobile CSP to motivate the language design, and describe its semantic domain, based on the standard failures-divergences model for CSP, but also recording a dynamic event alphabet. The traces component is identical to the separation logic semantics of Hoare & O'Hearn. Our novel contribution is a semantics for mobile channels in CSP, described in Unifying Theories of Programming, that supports: compositionality with other language paradigms; channel faults, nondeterminism, deadlock, and livelock; multi-way synchronisation; and many-to-many channels. We compare and contrast our semantics with other approaches, including the π-calculus, and consider implementation issues. As well as modelling reconfigurable systems, our extension to CSP provides semantics for techniques such as dynamic class-loading and the full use of dynamic dispatching and delegation.

Evaluating the Assignment of Behavioral Goals to Coalitions of Agents

Christophe Chareton[1], Julien Brunel[2], David Chemouil[2]
[1] École Polytechnique de Montréal, Montreal (Quebec), Canada
[2] Onera/DTIM, Toulouse, France

We present a formal framework for solving what we call the “assignment problem”: given a set of behavioral goals for a system and a set of agents described by their capabilities to make the system evolve, the problem is to find a “good” assignment of goals to (coalitions of) agents. To do so, we define Kore, a core modelling framework as well as its semantics in terms of a strategy logic called USL. In Kore, agents are defined by their capabilities, which are pre- and post-conditions on the system variables, and goals are defined in terms of temporal logic formulas. Then, an assignment associates each goal with the coalition of agents that is responsible for its satisfaction. Our problem consists in defining and checking the correctness of this assignment. We define different criteria for modelling and formalizing this notion of correctness. They reduce to the satisfaction of USL formulas in a structure derived from the capabilities of agents. Thus, we end up with a procedure for deciding the correctness of the assignment. We illustrate our approach using a toy example featuring exchanges of resources between a provider and two clients.

Towards Reasoning in Dynamic Logics with Rewriting Logic: the Petri-PDL Case

Christiano Braga and Bruno Lopes
Instituto de Computação, Universidade Federal Fluminense, Niterói, Brazil
cbraga@ic.uff.br, bruno@ic.uff.br

Safety is a desired property in software to ensure that no unforeseen scenarios will be achieved and in concurrent systems the variety of scenarios increase the complexity. Dynamic Logics (DL) present a large body of techniques to reason about systems and certify systems. Modelling and assessing concurrent systems with a formal semantics leads to the possibility of proving that they comply with their specification. Petri nets fulfill the requirements as a formal modelling language comprising a wide body of tools and an intuitive graphical interpretation. Petri-PDL puts together DL with Petri nets, providing a theoretical background to reason about Petri nets, inheriting their properties with all the techniques available for DL. This work presents a prototype implementation, in the Rewriting Logic language Maude, of a bounded model checker for Petri-PDL. The Petri-PDL model checker is formally designed following the representation of Kripke models as rewrite theories defined for the Linear Temporal Logic model checker available in Maude.

Refinement strategies for Safety-Critical Java

Alvaro Miyazawa and Ana Cavalcanti
Department of Computer Science, University of York, York, UK

Safety-Critical Java (SCJ) is a version of Java that supports the development of real-time, embedded, safety-critical software. SCJ introduces abstractions that enforce a simpler architecture, and simpler concurrency and memory models, to support easier certi cation. In this paper, we detail a re nement strategy that takes a state-rich process algebraic design specification that adheres to a cyclic executive pattern and produces an SCJ design that can be automatically translated to code. We then show how this refinement strategy can be extended to support more complex patterns that include non-terminating and multiple missions.

Verifying Transformations of Java programs using Alloy

Tarciana Dias da Silva, Augusto Sampaio, and Alexandre Mota
Universidade Federal de Pernambuco - Centro de Informática, Pernambuco, Brazil
{tds, acas, acm}@cin.ufpe.br

In this paper we verify Java transformations by using a fourth–stage strategy. Initially we embed models in Alloy: a metamodel for a subset of the Java, a model for each program transformation being investigated, and another one for a program called Validator that exercises methods of each side of the transfomation. Secondly, we use the Alloy Analyzer to find valid instances, corresponding to pairs (left and right-hand sides of a program transformation) and instances of the Validator. If instances can be found, this means they describe well– formed programs as long as transformation conditions, structural and type constraints are formally stated in our models. Thirdly we developed a tool that translates the Alloy instances to Java; finally, these are executed and the results used to verify whether there are any dynamic semantic problems in the resulting programs.

A Mechanized Textbook Proof of a Type Unification Algorithm

Rodrigo Ribeiro[1] and Carlos Camarão[2]
[1] Universidade Federal de Ouro Preto, João Monlevade, Minas Gerais, Brazil
rodrigo@decsi.ufop.br
[2] Universidade Federal de Minas Gerais, Belo Horizonte, Minas Gerais, Brazil
camarao@dcc.ufmg.br

Unification is the core of type inference algorithms for modern functional programming languages, like Haskell. As a first step towards a formalization of a type inference algorithm for such programming languages, we present a formalization in Coq of a type unification algorithm that follows classic algorithms presented in programming language textbooks.

Automatic generation of test cases and test purposes from natural language

Sidney Nogueira[1,2], Hugo L. S. Araujo[1], Renata B. S. Araujo[1], Juliano Iyoda[1], and Augusto Sampaio[1]
[1] Centro de Informática, Universidade Federal de Pernambuco
[2] DEINFO, Universidade Federal Rural de Pernambuco

Use cases are widely used for requirements description in the software engineering practice. As a use case event flow is often written in natural language, it lacks tools for automatic analysis or processing. In this paper, we extend previous work that proposes an automatic strategy for generating test cases from use cases written in a Controlled Natural Language (CNL), which is a subset of English that can be processed and translated into a formal representation. Here we propose a state-based CNL for describing use cases. We translate state-based use case descriptions into CSP processes from which test cases can be automatically generated. In addition, we show how a similar notation can be used to specify test selection via the definition of state-based test purposes, which are also translated into CSP processes. Test generation and selection are mechanised by running refinement checking verifications using the CSP processes for use cases and test purposes. All the steps of the strategy are integrated into a tool that provides a GUI for authoring use cases and test purposes described in the proposed CNL, so the formal CSP notation is totally hidden from the test designer. We illustrate our tool and techniques with a running example.

Time Performance Formal Evaluation of Complex Systems

Valdivino Alexandre de Santiago Júnior[1] and Sofiène Tahar[2]
[1] Instituto Nacional de Pesquisas Espaciais (INPE)
Av. dos Astronautas, 1758, SãoJosé dos Campos, São Paulo, SP, Brazil.
valdivino.santiago@inpe.br
[2] Concordia University
1455 De Maisonneuve Blvd. West, Montreal, QC, Canada
tahar@ece.concordia.ca

Formal verification methods, such as Model Checking, have been used for addressing performance/dependability analysis of systems. Such formal methods have several advantages over traditional techniques aiming at performance/dependability analysis such as the use of a single computational technique for evaluation of any measure and all complex numerical computation steps are hidden to the user. This paper reports on the use of Probabilistic Model Checking for time performance eval- uation of complex systems. We propose an approach, TPerP, that allows a professional to clearly address time performance analysis based on Continuous-Time Markov Chain (CTMC). Our approach takes into consideration several types of delay analyzes. We applied it to a balloon-borne high energy astrophysics scientific experiment where we dealt with CTMCs that had around 30 million reachable states and 75 million transitions, and we concluded that the current definition used in the balloon system is inadequate in terms of performance.

Test Case Generation from Natural Language Requirements using CPN Simulation

Bruno Cesar F. Silva, Gustavo Carvalho, and Augusto Sampaio
Universidade Federal de Pernambuco - Centro de Informática, Brazil
{bcfs,ghpc,acas}@cin.ufpe.br

We propose a test generation strategy from natural language (NL) requirements via translation into Colored Petri Nets (CPN), an extension of Petri Nets that supports model structuring and provides a mature theory and powerful tool support. This strategy extends our previous work on the NAT2TEST framework, which involves syntactic and semantic analyses of NL requirements and the generation of Data Flow Reactive Systems (DFRS) as an intermediate representation, from which target formal models can be obtained for the purpose of test case generation. Our contributions include a systematic translation of DFRSs into CPN models, besides a strategy for test generation. We illustrate our overall approach with a running example.