Swiss National Science Foundation

Lab for Automated Reasoning and Analysis

Automated Reasoning GrOup

Faculty of Mathematics

University of Belgrade

Workshop
Progress in Decision Procedures: From Formalizations to Applications

Belgrade, Serbia, March 30, 2013.

argo.matf.bg.ac.rs/pdp2013

Participants:

Milan Banković (MatF, University of Belgrade, Serbia)
Regis Blanc (EPFL, Switzerland)
Jelena Čolić Oravec (FTN, University of Novi Sad, Serbia)
Tihomir Gvero (EPFL, Switzerland)
Mirjana Ivanović (DMI, University of Novi Sad, Serbia)
Predrag Janičić (MatF, University of Belgrade, Serbia)
Chantal Keller (LIX, Palaiseau, France)
Etienne Kneuss (EPFL, Switzerland)
Filip Konecny (EPFL, Switzerland)
Damjan Krstajić (Research Centre for Cheminformatics, Serbia)
Viktor Kuncak (EPFL, Switzerland)
Ivan Kuraj (EPFL, Switzerland)
Vladimir Kurbalija (DMI, University of Novi Sad, Serbia)
Peter Lammich (TU Munich, Germany)
Tatjana Lutovac (ETF, University of Belgrade, Serbia)
Filip Marić (MatF, University of Belgrade, Serbia)
Bojan Marinković (MI SANU, Serbia)
Vesna Marinković (MatF, University of Belgrade, Serbia)
Mladen Nikolić (MatF, University of Belgrade, Serbia)
Filip Nikšić (MPI-SWS, Saarbrücken, Germany)
Aljoša Obuljen (Microsoft Development Center, Belgrade, Serbia
Jovana Obradović (FTN, University of Novi Sad, Serbia)
Jovanka Pantović (FTN, University of Novi Sad, Serbia)
Zoran Petrić (MI SANU, Serbia)
Danijela Petrović (MatF, University of Belgrade, Serbia)
Ivan Petrović (MatF, University of Belgrade, Serbia)
Pedro Quaresma de Almeida (University of Coimbra, Portugal)
Jovana Radenović (FTN, University of Novi Sad, Serbia)
Enric Rodríguez Carbonell (UPC, Barcelona, Spain)
Andrijana Stamenković (FTN, University of Novi Sad, Serbia)
Sana Stojanović (MatF, University of Belgrade, Serbia)
Mirko Stojadinović (MatF, University of Belgrade, Serbia)
Srđan Škrbić (DMI, University of Novi Sad, Serbia)
Predrag Tanović (MI SANU, Serbia)
Dmitriy Traytel (TU Munich, Germany)
Milan Todorović (MI SANU, Serbia)
Milena Vujošević-Janičić (MatF, University of Belgrade, Serbia)
Aleksandar Zeljić (Uppsala University, Sweden)
Dragiša Žunić (FIMEK, Novi Sad, Serbia)

Organization:
Automated Reasoning GrOup ARGO
Univeristy of Belgrade, Serbia
argo.matf.bg.ac.rs
and
Lab for Automated Reasoning and Analysis LARA
EPFL, Switzerland
lara.epfl.ch

Supported by:
SNF SCOPES IZ73Z0_127979 research grant

Host institution:
Faculty of Mathematics
University of Belgrade

Workshop
Progress in Decision Procedures: From Formalizations to Applications

http://argo.matf.bg.ac.rs/pdp2013

Book of Abstracts

and

Little Belgrade City Guide for Workshop Participants

March 30, 2013, Belgrade, Serbia

Preface

This booklet contains abstracts of the talks given at:

Workshop
Progress in Decision Procedures: From Formalizations to Applications

held at the Faculty of Mathematics, University of Belgrade on March 30, 2013. The workshop marks the end of successful bilateral joint research grant SNF SCOPES IZ73Z0_127979 between LARA group from EPFL and ARGO group from the University of Belgrade.

The meeting was attended by 39 participants coming from 15 research institutions from 7 European countries (France (1), Germany (3), Portugal (1), Serbia (26), Spain (1), Sweden (1), Switzerland (6)).

The workshop addressed a wide range of aspects of formal and automated theorem proving, with a special emphasis on decision procedures, their formalizations, implementations, and applications. The program consisted of 16 talks, divided (rather loosely) into the following categories: Overview Talks, Satisfiability Modulo Theory, Formalization of Decision Procedures, Decisions in Applications, Dealing with Proofs, and Software Verification.

The meeting was organized by the LARA group (`http://lara.epfl.ch/w/`) and the ARGO group (`http://argo.matf.bg.ac.rs`), and was supported by the research grant SNF SCOPES IZ73Z0_127979.

For the success of the meeting, we are grateful to all speakers and to all participants. We are also grateful to SNF (Swiss National Science Foundation) which funded the joint research project SNF SCOPES IZ73Z0_127979 and to the Faculty of Mathematics, University of Belgrade which was the host institution of the meeting.

More details about the meeting can be found online: `http://argo.matf.bg.ac.rs/pdp2013`

Viktor Kunčak
EPFL, Lausanne, Switzerland
and
Predrag Janičić
Faculty of Mathematics, University of Belgrade, Serbia

Participants

1. Milan Banković, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~milan`
2. Regis Blanc, EPFL, Switzerland `http://people.epfl.ch/regis.blanc`
3. Jelena Čolić Oravec, FTN, University of Novi Sad, Serbia `http://imft.ftn.uns.ac.rs/math/People/Jelena%C4%8Coli%C4%87`
4. Tihomir Gvero, EPFL, Switzerland `http://people.epfl.ch/tihomir.gvero`
5. Mirjana Ivanović, DMI, University of Novi Sad, Serbia `http://perun.pmf.uns.ac.rs/index.php?option=com_content&task=view&id=68&Itemid=41`
6. Predrag Janičić, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~janicic`
7. Chantal Keller, Laboratoire d'Informatique de Polytechnique (LIX), Palaiseau, France `http://www.lix.polytechnique.fr/~keller/`
8. Etienne Kneuss, EPFL, Switzerland `http://people.epfl.ch/etienne.kneuss`
9. Filip Konecny, EPFL, Switzerland `http://people.epfl.ch/filip.konecny`
10. Damjan Krstajić, Research Centre for Cheminformatics `Serbiahttp://www.rcc.org.rs/dkrstajic.html`
11. Viktor Kuncak, EPFL, Switzerland `http://lara.epfl.ch/~kuncak`
12. Ivan Kuraj, EPFL, Switzerland, `http://people.epfl.ch/ivan.kuraj`
13. Vladimir Kurbalija, DMI, University of Novi Sad, Serbia `http://perun.pmf.uns.ac.rs/kurbalija`
14. Peter Lammich, TU Munich, Germany, `http://www21.in.tum.de/~lammich/`
15. Tatjana Lutovac, ETF, University of Belgrade, Serbia `http://matematika.etf.bg.ac.rs/ljudi/t_lutovac.htm`
16. Filip Marić, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~filip`
17. Bojan Marinković, MI SANU, Serbia `http://www.mi.sanu.ac.rs/~bojanm`
18. Vesna Marinković, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~vesnap`
19. Mladen Nikolić, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~nikolic`
20. Filip Nikšić, MPI-SWS, Saarbrücken, Germany `http://www.mpi-sws.org/~fniksic`
21. Aljoša Obuljen, Microsoft Development Center, Belgrade, Serbia `http://www.microsoft.com/serbia/mdcs/`

22. Jovana Obradović, FTN, University of Novi Sad, Serbia `https://sites.google.com/site/jovanaftn/`

23. Jovanka Pantović, FTN, University of Novi Sad, Serbia, `http://imft.ftn.uns.ac.rs/~vanja/`

24. Zoran Petrić, MI SANU, Serbia, `http://www.mi.sanu.ac.rs/~zpetric`

25. Danijela Petrović, MatF, University of Belgrade, Serbia, `http://www.matf.bg.ac.rs/~danijela`

26. Ivan Petrović, MatF, University of Belgrade, Serbia `http://alas.matf.bg.ac.rs/~mr00006/`

27. Pedro Quaresma de Almeida, University of Coimbra, Portugal `http://www.mat.uc.pt/~pedro/`

28. Jovana Radenović, FTN, University of Novi Sad, Serbia `https://sites.google.com/site/jovanadradenovic/`

29. Enric Rodríguez Carbonell, Technical University of Catalonia (UPC), Barcelona, Spain `http://www.lsi.upc.edu/~erodri/`

30. Andrijana Stamenković, FTN, University of Novi Sad, Serbia

31. Sana Stojanović, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~sana`

32. Mirko Stojadinović, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~mirkos`

33. Srdan Škrbić, DMI, University of Novi Sad, Serbia `http://www.is.pmf.uns.ac.rs/skrbics/`

34. Predrag Tanović, MI SANU, Serbia

35. Dmitriy Traytel, TU Munich, Germany `http://home.in.tum.de/~traytel/`

36. Milan Todorović, MI SANU, Serbia

37. Milena Vujošević-Janičić, MatF, University of Belgrade, Serbia `http://www.matf.bg.ac.rs/~milena`

38. Aleksandar Zeljić, Uppsala University, Sweden `http://www.it.uu.se/katalog/aleze648?lang=en`

39. Dragiša Žunić, FIMEK, Novi Sad, Serbia `https://sites.google.com/site/dragisazunic/`

Program

Friday, March 29, 2013.	
17:15—18:00	Meeting at the Hotel Palace
18:00—19:30	Guided Walking Tour
19:30—22:00	Dinner (Restaurant ”Kalemegdanska terasa”)

Saturday, March 30, 2013.	
09:45—09:50	Opening Remarks
Session *Overview Talks*; Session Chair: Predrag Janičić	
09:50—10:15	Filip Marić (MatF, University of Belgrade, Serbia): *Overview of Research Activities of the ARGO Group over the Last Three Years*
10:15—10:40	Viktor Kunčak (EPFL, Switzerland): *Implicit Programming: An Overview*
10:40—11:00	Coffee break
Session *Satisfiability Modulo Theories*; Session Chair: Predrag Janičić	
11:00—11:25	Enric Rodríguez Carbonell (Technical University of Catalonia (UPC), Spain): *Non-linear Arithmetic Solving for Termination Analysis*
11:25—11:40	Aleksandar Zeljić (Uppsala University, Sweden): *Towards SMT Style Reasoning about Floating-Point Arithmetic*
11:40—12:00	Coffee break
Session *Decision Procedures and Interactive theorem proving*; Session Chair: Predrag Janičić	
12:00—12:25	Chantal Keller (LIX, Palaiseau, France): *SMTCoq: Cooperation Between SAT/SMT Solvers and the Coq Proof Assistant through Proof Witnesses*
12:25—12:50	Dmitriy Traytel (TU-Munich, Germany): *A Verified Decision Procedure for MSO on Words Based on Derivatives of Regular Expressions*
12:50—14:50	Lunch break (Restaurant ”Jevrem”)
Session *Decisions in Applications*; Session Chair: Predrag Janičić	
14:50—15:15	Aljoša Obuljen (Microsoft Development Center Serbia): *Overview of Microsoft Development Center Serbia - Deliverables and Focus Areas*
15:15—15:40	Vladimir Kurbalija (DM, University of Novi Sad, Serbia): *Analysis of Constrained Time-Series Similarity Measures*
15:40—16:00	Coffee break
Session *Dealing with Proofs I*; Session Chair: Filip Marić	
16:00—16:25	Peter Lammich (TU-Munich, Germany): *The Isabelle Refinement Framework*
16:25—16:50	Pedro Quaresma (University of Coimbra, Portugal): *The Web Geometry Laboratory Project (Intelligent Geometric Tools)*
16:50—17:10	Coffee break
Session *Dealing with Proofs II*; Session Chair: Filip Marić	
17:10—17:35	Tatjana Lutovac (ETF, University of Belgrade, Serbia): *A Syntax Approach to Automated Detection of Some Redundancies in Linear Logic Sequent Derivations*
17:35—17:50	Jelena Čolić Oravec (FTN U Novi Sad, Serbia): *Incompletely Specified Operations and Their Clones*
17:50—18:10	Coffee break
Session *Software verification*; Session Chair: Filip Marić	
18:10—18:25	Filip Nikšić (MPI-SWS, Saarbrücken, Germany): *Incremental, Inductive Coverability*
18:25—18:40	Filip Konecny (EPFL, Switzerland): *Underapproximation of Procedure Summaries for Integer Programs*
18:40—18:55	Regis Blank (EPFL, Switzerland): *Verifying Functional Scala Programs in Leon*
18:55—19:10	Milena Vujošević-Janičić (MatF, University of Belgrade, Serbia): *System LAV and its Applications*
19:10—19:15	Closing Remarks
20:00—23:00	Dinner (Restaurant ”Teatroteka”)

Overview Talks

Session Chair: Predrag Janičić

1 Overview of Research Activities of the ARGO Group Over the Last Three Years

Filip Marić
MatF, University of Belgrade, Serbia

Abstract

ARGO group, based at the Department of Computer Science, Faculty of Mathematics, University of Belgrade is interested various areas of automated reasoning (SAT/SMT solving, automated theorem proving in geometry, interactive theorem proving, software verification, applications of automated reasoning in education, ...). During the last three years we have published around 25 publications in peer-reviewed journals, workshops and conferences. In this short talk, the published and our current research will be very briefly described.

2 Implicit Programming: An Overview

Viktor Kuncak
EPFL, Switzerland

Abstract

Implicit programming is a software development paradigm that we proposed to address long-standing bottlenecks of software construction. Today, even easy tasks require the work of professionals and experts, whose programming skills are in stark contrast to the remaining users of computing infrastructure. In the decade with billions of users of computing resources, blurring this contrast can have far-reaching economic and social consequences. Our implicit programming paradigm aims to make software construction substantially easier at several levels, from new declarative programming language constructs to new software development tools. We support implicit programming with a concept of synthesis procedures, which enhance algorithms in todays compilers by building on the advances in satisfiability modulo theories and the rapidly expanding field of software synthesis. We are also introducing new development tools, application manipulation interfaces, and techniques to handle ambiguity, with the goal of making development easier for both experts and non-experts.

Satisfiability Modulo Theories

Session Chair: Predrag Janičić

3 Non-linear Arithmetic Solving for Termination Analysis

Enric Rodríguez Carbonell
Technical University of Catalonia (UPC), Spain

Abstract

We consider the problem of solving formulas in non-linear arithmetic (NA). In previous work we proposed an approach based on reducing the problem to linear arithmetic (LA) and analizing unsatisfiable cores. In this talk we present an alternative MAX-SMT(LA)-based method, which is simpler to implement and which naturally extends to MAX-SMT(NA).

Most importantly, we show how MAX-SMT(NA) can be used for proving termination of imperative programs. MAX-SMT allows us to express the termination problem by giving different weights to the needed conditions, which provides a better notion of progress in comparison to previous approaches. The method has been implemented in a prototype that has successfully been tested on a wide sample of examples.

4 Towards SMT Style Reasoning about Floating-Point Arithmetic

Aleksandar Zeljić
Uppsala University, Sweden

Abstract

Software in many domains, including embedded systems, requires handling of real-valued quantities and relies on floating-point arithmetic (FPA) for that purpose. Since FPA operations involve rounding, results are often unintuitive, making analysis and verification of such software difficult. There is need for automatic, efficient, and bit-precise methods for reasoning about FPA. SMT solvers have proven to be a successful tool for reasoning in different domains (e.g., LIA, arrays, bit-vectors, etc.). They can be extended to other domains, such as FPA, by providing new theory solvers. The naive method for solving FPA constraints involves their encoding to fixed-size bit-vectors, followed by bit-blasting. In many cases, however, bit-blasting is extremely time and memory consuming, even if the subsequent reasoning is quick. We explore techniques to perform bit-blasting in a more lazy manner, by means of (under/over)-approximations that are refined by analysing satisfying assignments. Initial experiments are carried out in the context of the Z3 SMT solver.

Decision Procedures and Interactive Theorem Proving

Session Chair: Predrag Janičić

5 SMTCoq: Cooperation between SAT/SMT Solvers and the Coq Proof Assistant through Proof Witnesses

Chantal Keller
LIX, France

Abstract

A skeptical cooperation between automatic and interactive theorem provers is beneficial for both sides: the automatic provers grow in safety since their answers are checked /a posteriori/, and the interactive provers can enjoy more automation without compromising soundness. In this talk, I present SMTCoq, a cooperation tool between SAT/SMT solvers and the Coq proof assistant. It relies on a certified checker for SAT and SMT proof witnesses implemented in Coq, which is both efficient and modular: new SAT/SMT solvers as well as new decision procedures can be easily plugged-in. Finally, the use of Coq also offers the possibility to extract the code to obtain a certified checker running in general purpose programming languages.

---

6 A Verified Decision Procedure for MSO on Words Based on Derivatives of Regular Expressions

Dmitriy Traytel
TU Munich, Germany

Abstract

Monadic second-order logic on finite words (MSO) is a decidable yet expressive logic into which many decision problems can be encoded.

Since MSO formulas correspond to regular languages, equivalence of MSO formulas can be reduced to the equivalence of some regular structures (e.g. automata). However, formal verification of automata is a difficult task. Instead, the recursive data structure of regular expressions simplifies the formalization, notably by offering a structural induction principle.

Decision procedures of regular expression equivalence have been formalized before, usually based on Brzozowski derivatives. Yet, for a straightforward embedding of MSO formulas into regular expressions an extension of regular expressions with a projection operation is required. We prove total correctness

and completeness of an equivalence checker for regular expressions extended in that way. We also define a semantics-preserving translation of MSO formulas into regular expressions. Our results have been formalized and verified in the theorem prover Isabelle. Using Isabelle's code generation facility, this yields a formally verified algorithm that decides equivalence of MSO formulas.

Decisions in Applications

Session Chair: Filip Marić

7 Overview of Microsoft Development Center Serbia - Deliverables and Focus Areas

Aljoša Obuljen
Microsoft Development Center Serbia, Serbia

Abstract

Microsoft Development Center Serbia has been operating in Belgrade since 2005. This talk addresses Microsoft divisions with investment in the development site, technologies that were developed and center's initiative in broader informatics society, such as algorithmic competitions and internship programs. The divisions having representative teams in the center focus on core engineering of database systems, document layout analysis, image processing and accompanying engineering best practices and machine learning/heuristic approaches.

---

8 Analysis of Constrained Time-Series Similarity Measures

Vladimir Kurbalija
DMI, University of Novi Sad, Serbia

— *Joint work with Miloš Radovanović, Zoltan Geler, Mirjana Ivanović* —

Abstract

Time series represent the type of data where the values of some events are recorded over repeated measurements in time. Due to their applicability, time-series data became an inevitable part in many practical domains such as medical treatments, stock market analysis, observation of natural phenomena etc. However, working with time series is extremely difficult mainly because of their high dimensionality and the problem of defining a suitable similarity measure. Motivated by mentioned facts, the group at Department of Mathematics and Informatics, University of Novi Sad developed the multipurpose system FAP (Framework for Analysis and Prediction) which will be described in this paper. FAP is the framework for different aspects of time-series analysis which supports all three main concepts which need to be considered when dealing with time series: pre-processing transformation, time-series representation and similarity/distance measure. As the demonstration of the FAP system, we will investigate four main similarity/distance measures based on dynamic programming: Dynamic Time Warping (DTW) Longest Common Subsequence (LCS), Edit Distance with Real Penalty (ERP) and Edit Distance on Real sequence (EDR), and the effects of global constraints on them.

Dealing with Proofs

Session Chair: Filip Marić

9 The Isabelle Refinement Framework

Peter Lammich
TU Munich, Germany

Abstract

We provide a framework for program and data refinement in Isabelle/HOL. It is based on a refinement calculus for monadic expressions and provides tools to automate canonical tasks such as verification condition generation. It produces executable programs, from which Isabelle/HOL can generate verified, efficient code in various languages, including Standard ML, Haskell and Scala.

We have applied the framework to formalize various complex algorithms, among them Hopcroft's algorithm for automata minimization, the algorithm of Ilie, Navarro and Yu to compute simulation preorders on NFAs, Dijkstra's single-source shortest path algorithm, and a nested depth-first search algorithm for the emptiness check of Buchi automata.

---

10 The Web Geometry Laboratory Project (Intelligent Geometric Tools)

Pedro Quaresma
CISUC/Department of Mathematics, University of Coimbra, Portugal

Abstract

In this talk the Web Geometry Laboratory is introduced, its current status and the challenges that lie ahead.

The *Web Geometry Laboratory* (WGL) project's goal is, to build an adaptive and collaborative blended-learning Web-environment for geometry integrating DGSs and GATPs.

- In its current version (WGL1.1) it is already a collaborative blended-learning Web-environment integrating a dynamic geometry system (DGS) and with some adaptive features [3].
- The adaptative module will require a query mechanism for formal descriptions of geometric constructions [2].
- The loose integration of GATPs requires a common format capable of linking all the tools in the field of geometry, the I2GATP, an extension of the i2g format has been proposed in [1].

- The need for readable proofs leads to the implementation of semi-synthetic methods like the full-angle method. The production of “visual proofs” is also a goal to pursue.

In conclusion, if we want to provide an exciting and challenging learning environment for geometry many, more fundamental, questions have to be dealt with, and solved, before we can reach that goal.

[1] Pedro Quaresma. An XML-format for conjectures in geometry. In James Davenport, Johan Jeuring, Christoph Lange, and Paul Libbrecht, editors, *24th OpenMath Workshop, 7th Workshop on Mathematical User Interfaces (MathUI), and Intelligent Computer Mathematics Work in Progress*, number 921 in CEUR Workshop Proceedings, pages 54–65, Aachen, 2012.

[2] Pedro Quaresma and Yannis Haralambous. Geometry Constructions Recognition by the Use of Semantic Graphs. In *Proceedings of RecPad 2012*, 2012.

[3] Vanda Santos and Pedro Quaresma. Collaborative aspects of the WGL project. *Electronic Journal of Mathematics & Technology*, 2013. (to appear).

11 A Syntax Approach to Automated Detection of Some Redundancies in Linear Logic Sequent Derivations

Tatjana Lutovac
ETF, University of Belgrade, Serbia

Abstract

Sequent calculi proof search often involves managing information that later turns out to be redundant. This paper focuses on the development of automated techniques for the detection and elimination of certain kinds of redundant formulae from linear logic sequent derivations. A method is proposed for annotating sequent calculi rules and proofs with Boolean formulae and constraints describing which parts of the proof are essential. The proposed constraints are a result of purely syntactic observations of the structure and restrictions on the rule’s premise and/or conclusion. An algorithm is developed which makes it possible to detect and eliminate certain parts of the proofs/derivations that do not actively participate in the proof search. The proposed approach is independent of the search strategy used.

12 Incompletely Specified Operations and Their Clones

Jelena Čolić Oravec
FTN, University of Novi Sad, Serbia

— *Joint work with Jovanka Pantović and Hajime Machida* —

Abstract

Incompletely specified operations are functions whose output values are specified for only some of the input values. Although this concept is to a certain extent similar to that of partial operations, incompletely specified operations are significantly different when it comes to composition of such operations. Namely, composition of partial operations f and g is undefined whenever f or g is undefined, while composition of incompletely specified operations f and g may be specified even if f or g is not specified. We define composition of incompletely specified operations and the corresponding clone, which we compare with clones of total, partial and hyperoperations.

Software verification

Session Chair: Filip Marić

13 Incremental, Inductive Coverability

Filip Nikšić
MPI-SWS, Saarbrücken, Germany

Abstract

Incremental, inductive coverability (IIC) is a new algorithm for checking coverability of well-structured transition systems. The algorithm generalizes IC3, a SAT-based algorithm for safety verification of finite-state systems, which has shown a great success in verification of hardware. IIC has been implemented for Petri nets and preliminary results indicate it sustains IC3's benefits. In this short talk, I will briefly present IIC and show how it runs on a simple Petri net example.

---

14 Underapproximation of Procedure Summaries for Integer Programs

Filip Konecny
EPFL, Switzerland

Abstract

We show how to underapproximate the procedure summaries of recursive programs over the integers using off-the-shelf analyzers for non-recursive programs. The novelty of our approach is that the non-recursive program we compute may capture unboundedly many behaviors of the original recursive program for which stack usage cannot be bounded. Moreover, we identify a class of recursive programs on which our method terminates and returns the precise summary relations without underapproximation. Doing so, we generalize a similar result for non-recursive programs to the recursive case. Finally, we present experimental results of an implementation of our method applied on a number of examples.

---

15 Verifying Functional Scala Programs in Leon

Regis Blank
EPFL, Switzerland

Abstract

We present the Leon verification system for a functional subset of the Scala programming language. In this system, both properties and programs are expressed using user-defined functions. We extend this initial subset with support for imperative constructs such as mutation or loops by compiling them into pure functions with additional arguments.

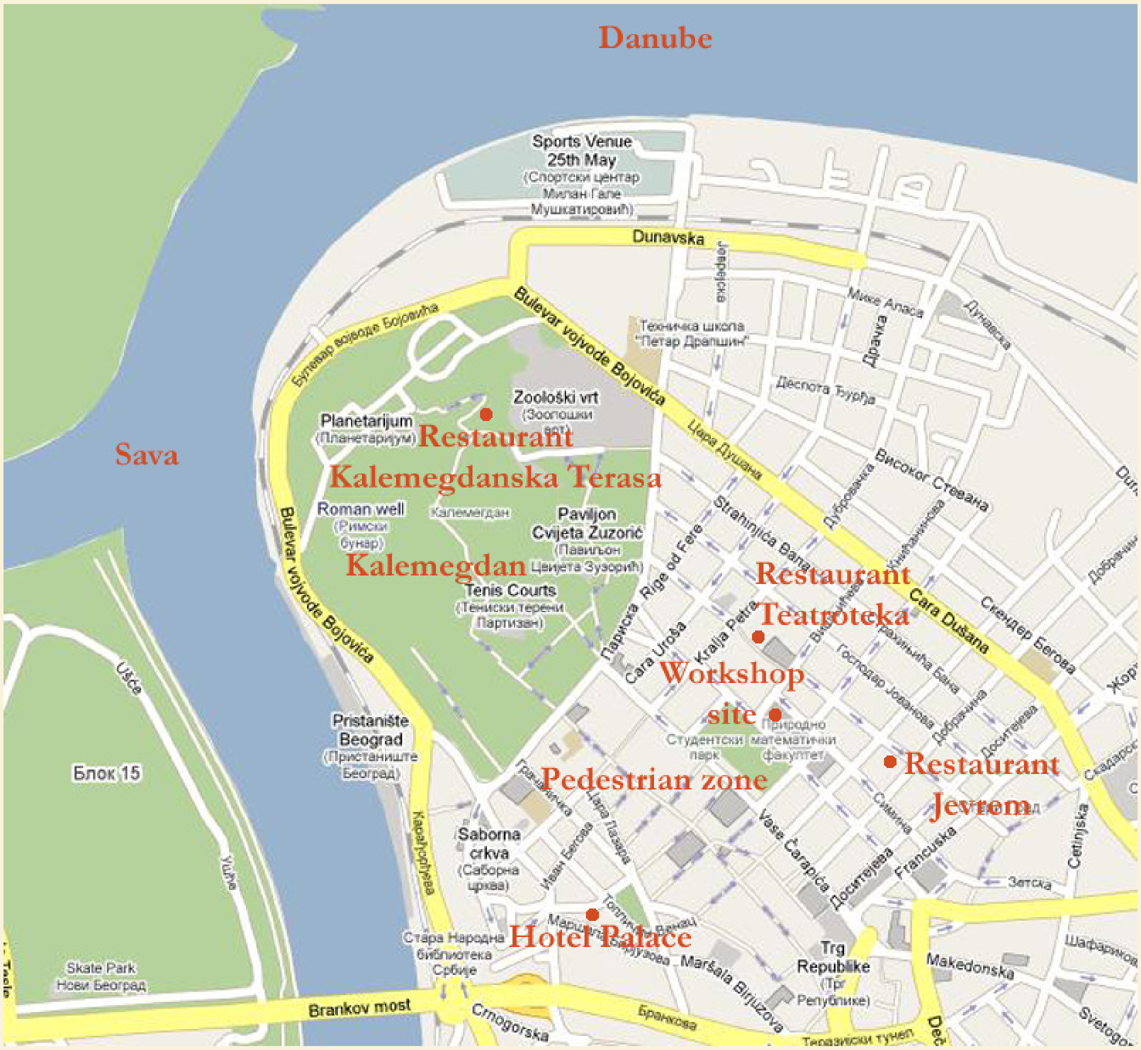
16 System LAV and its Applications

Milena Vujošević Janičić
MatF, University of Belgrade, Serbia

Abstract

In this talk we will briefly present LAV, an open-source tool for statically verifying program assertions and locating bugs such as buffer overflows, pointer errors and division by zero. We will also present its application within a framework for automated evaluation of students' assignments. The framework is based on merging information from three different evaluation methods: automated testing, automated bug finding, and control flow graph similarity measurement. Our empirical evaluation shows that the synergy of proposed methods improves the quality and precision of automated grading and that automatically generated grades are highly correlated with instructor-assigned grades.

Little Belgrade City Guide for Workshop Participants



Workshop Site

The workshop site is the building of the faculties of sciences of the University of Belgrade. It is located in the very city centre and close to Kalemegdan fortress, and the rivers Danube and Sava. The workshop site is just 300m from the Knez Mihajlova street and the surrounding pedestrian zone, with a large number of impressive buildings and mansions built in XIX and XX century in the style of neoclassicism, academism, secession, and art-deco. Just 200m from the workshop site are remains of large Roman termes (built in III centrury) and 100m away is Sheikh Mustapha's turbeh (Turkish mausoleum; erected in XVIII century over the tomb of this religious figure), to name just a few intersting sights that are nearby.

Brief History of Belgrade

Belgrade, a city of very turbulent history, is one of the oldest cities in Europe. Its history lasts full 7000 years. The area around two great rivers, the Sava and the Danube has been inhabited as early as palaeolithic period. Remains of human bones and skulls of Neanderthals, found in Belgrade date back to the early Stone Age. The founding of Singidunum (the ancient name of Belgrade) is attributed to the Celtic tribe, the Scordiscs. Singidunum was mentioned for the first time in 279 B.C. The first part of the word - Singi - means ”round” and dunum means ”fortress” or ”town”. The Romans conquered Belgrade in the beginning of the I century A.D. and it has been under their rule for full four centuries. The Huns captured the town and completely destroyed it in 441. After the fall of the Huns, the town became a part of the Byzantine Empire in 454, but it was soon conquered by the Sarmatians, and later the Eastern Goths. In 488, it became a Byzantine town again. Around 630, the Serbian settlers come to this area. The town was first mentioned under the Slavic name Beograd (White Town - probably because of the walls made of white limestone) in 878. The Serbian rule over Belgrade began in 1284. but during some periods it was under Hungarians again. After almost a century of resisted sieges and attacks, Belgrade fell to Turks's rule in 1521. The town, getting more and more oriental look, counted in XVII population of 100000 and was the second-largest town in the Empire, right after Istanbul. The Austrians conquered Belgrade in 1688. When in 1739 it was captured again by the Turks, it was exposed to a heavy destruction. After two Serbian insurrections (started in 1804 in 1815) and the period of weakening of their power in Serbia, the Turks left Belgrade for good in 1867. In World War I, the Austrian army conquered the city in October 1915. The Serbian army and parts of the Allies' army liberated Belgrade in 1918. During WWI, Serbia lost 28% of its whole population, while Belgrade was the most destroyed town in Serbia. After the liberation, Belgrade became the capital of the newly-created Kingdom of the Serbs, Croats and Slovenes (later called Yugoslavia). In April 1941, Belgrade became the target of a terrible destruction by German air force. Belgrade also had to undergo losses in the Allies' bombing, especially in 1944. During World War II Belgrade lost about 50000 citizens and suffered inestimable damage. Belgrade was liberated by the units of the National Liberation Army of Yugoslavia and the Red Army on October 20, 1944. The monarchy in Yugoslavia was abolished in 1945 when the communist rule of Josip Broz Tito started. Thanks to a specific policy of Yugoslavia, Belgrade became an important international, political, cultural, sports, and economic center, linking East and West, North and South. Many unsolved national problems led to disintegration of Yugoslavia in 1991 and since 2006, the Republic of Serbia is independent state with Belgrade as its capital.

Briefly About Modern Belgrade

Belgrade is the capital and the largest city of Serbia. The city lies at the confluence of Sava and Danube rivers. With a population of almost two million, Belgrade is the third largest city in Southeastern Europe. The architecture of Belgrade is a mirror of different cultural and historical periods, influences and styles: from old Oriental influences, across baroque architecture, secession, academism and neoclassicism, socialist and industrial features from post WW2 period, to modern architecture and layout of New Belgrade with wide boulevards. Knez Mihajlova Street is the main walking street in Belgrade. It is a pedestrian zone, protected by law as one of the most valuable monumental complexes of the city. Belgrade has many beautiful parks and the biggest one is Kalemegdan, with an old fortress, comprising remains from Ancient and Byzantine times to Turkish and Austro-Ugrian periods. Belgrade has more than 20 theaters and two opera houses and it is home to a number of film, theater, and music festivals. There are many excellent restaurants, cafés and pubs, and British Times proclaimed Belgrade as Europe's best nightlife city.

Knez Mihajlova Street

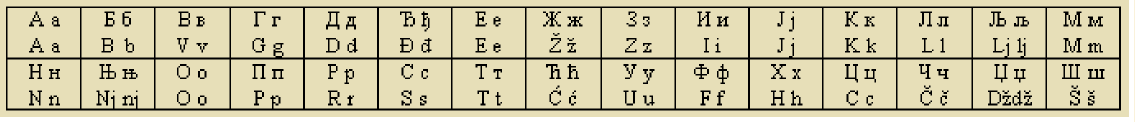
Knez Mihajlova Street, pedestrian precinct and main city street, now protected by law, is one of the oldest and most valuable city environments, with a whole range of impressive buildings and town houses which sprung up at the end of the 1880's. It is generally believed that as early as Roman times this was the centre of the settlement of Singidunum, while during Turkish rule the streets went through the gardens, fountains and mosques that stood in this part of town. Today it is the main business area of Belgrade and the headquarters of many national institutions (such as the Serbian Academy of Science and Arts, Belgrade City Library and the Belgrade Cultural Centre).

Kalemegdan Park and Fortress

Kalemegdan is the core and the oldest section of the urban area of Belgrade and for centuries the city population was concentrated only within the walls of the fortress, thus its history, until most recent history, equals the history of Belgrade itself. The name Kalemegdan derives from two Turkish words, kale (fortress) and megdan (battleground) (literally, ”battlefield fortress”). Kalemegdan fortress is the most important cultural-historical complex in the city, standing above the Sava-Danube confluence. Since its construction the Belgrade fortress has been constantly attacked and defended, destroyed and renovated. Chronicles trace a history of about 40 to 60 devastations of the fortress. The landscaping of the wide plateau around the fortress was begun on the order of Prince Mihailo Obrenovic after the fortress had been handed over from the Turks to the Serbs in 1867. and it was converted into a park in the 1880's. Today, Kalemegdan park is the largest and loveliest park in Belgrade with an area of 52 hectares. There is a number of monuments, Sahat Tower, the Military Museum, the statue of Belgrade Victor, the Zoo.

Serbian Alphabet

Serbian is a South Slavic language. Both Latin and Cyrillic alphabets are used to write Serbian. Serbian is an example of synchronic digraphia. The orthography, introduced by the language reform led by Vuk Karadžić in mid XIX century, is very consistent: it is an approximation of the principle ”one letter per sound”. The following table gives 30 letters used in Serbian, both in Cyrillic and in Latin alphabet.



А а A a	Б б B b	В в V v	Г г G g	Д д D d	Ђ ђ Đ đ	Е е E e	Ж ж Ž ž	З з Z z	И и I i	Ј ј J j	К к K k	Л л L l	Љ љ Lj lj	М м M m
Н н N n	Њ њ Nj nj	О о O o	П п P p	Р р R r	С с S s	Т т T t	Ћ ћ Ć ć	У у U u	Ф ф F f	Х х H h	Ц ц C c	Ч ч Č č	Џ џ Dž dž	Ш ш Š š