

# Radu Grigore

December 2016

Email: [radugrigore@gmail.com](mailto:radugrigore@gmail.com)

Homepage: <http://rgrig.appspot.com/>

Blog: <http://rgrig.blogspot.com/>

## Selected Publications

- 2017 Radu Grigore. *Java Generics Are Turing Complete*. POPL
- 2016 Maria Bruna, Radu Grigore, Stefan Kiefer, Joël Ouaknine, James Worrell. *Proving the Herman-Protocol Conjecture*. ICALP
- 2016 Nikos Tzevelekos, Radu Grigore. *History-Register Automata*. LMCS (supersedes FoSSaCS2013)
- 2016 Radu Grigore, Hongseok Yang. *Abstraction Refinement Guided by a Learnt Probabilistic Model*. POPL
- 2015 Radu Grigore, Stefan Kiefer. *Tree Buffers*. CAV
- 2014 Xin Zhang, Ravi Mangal, Radu Grigore, Mayur Naik, Hongseok Yang. *On Abstraction Refinement for Program Analyses in Datalog*. PLDI (**distinguished paper award**)
- 2013 Mikoláš Janota, Radu Grigore, Joao Marques-Silva. *On QBF Proofs and Preprocessing*. LPAR
- 2013 Radu Grigore, Dino Distefano, Rasmus Lerchedahl Petersen, Nikos Tzevelekos. *Runtime Verification Based on Register Automata*. TACAS
- 2012 Thorsten Bormer, Marc Brockschmidt, Dino Distefano, Gidon Ernst, Jean-Christophe Filliâtre, Radu Grigore, Marieke Huisman, Vladimir Klebanov, Claude Marché, Rosemary Monahan, Wojciech Mostowski, Nadia Polikarpova, Christoph Scheben, Gerhard Schellhorn, Bogdan Tofan, Julian Tschannen, Mattias Ulbrich. *The COST IC0701 verification competition 2011*. FoVeOOS
- 2011 Matko Botinčan, Dino Distefano, Mike Dodds, Radu Grigore, Daiva Naudžiūnienė, Matthew J. Parkinson. *coreStar: The Core of jStar*. BOOGIE
- 2010 Mikoláš Janota, Goetz Boetterweck, Radu Grigore, Joao Marques-Silva. *How to Complete a Configuration Process?* SOFSEM
- 2009 Radu Grigore, Julien Charles, Fintan Fairmichael, Joseph Kiniry. *Strongest Postcondition of Unstructured Programs*. FTfJP
- 2007 Mikoláš Janota, Radu Grigore, Michał Moskal. *Reachability Analysis for Annotated Code*. SAVCBS
- 2005 Valentin Ștefan Gheorghiiță, Radu Grigore. *Constructing Checkers from PSL Properties*. CSCS

## Teaching

- 2016 *Introduction to Computer Systems* (Kent): lectures.
- 2016 *Systems Architecture* (Kent): lectures.
- 2016 *Operating Systems and Architecture* (Kent): lectures.
- 2016 *Theory of Computing* (Kent): classes.
- 2016 *Database Systems* (Kent): classes.
- 2013–2015 *Admission interviews* (Oxford, Worcester College).
- 2014 *Database Systems Implementation* (Oxford): I led problem solving sessions.
- 2012 *Computability* (QMUL): I marked the exams.
- 2011 *Specification and Verification* (QMUL): I led problem solving sessions.
- 2011 *Probabilities and Matrices* (QMUL): I demonstrated during practicals.
- 2011 *Algorithms and Data Structures in an Object-Oriented Framework* (QMUL): I demonstrated during practicals.
- 2009 *Unix Programming* (UCD): I **designed the course**, and I delivered it.
- 2007–2008 Coach of the team that represented UCD (and Ireland) in [ACM ICPC](#).
- 2008 *Foundations of Computing* (UCD): I demonstrated during practicals.
- 2008 *Operating Systems* (UCD): I **designed the course**, and I delivered it.
- 2006–2007 *Algorithmic Problem Solving* (UCD): I demonstrated during practicals.
- 2006 *Data Structures and Algorithms* (UCD): I demonstrated during practicals.
- 2004 *Telecommunications Software* (PUB): I demonstrated during practicals.
- 2003 *Data Networks* (PUB): I demonstrated during practicals.

## Education

- 2013–2015 Postdoctoral Research Assistant, Computer Science, University of Oxford.
- 2010–2013 Postdoctoral Research Assistant, Theory Group, EECS, Queen Mary, University of London.
- 2005–2010 PhD, CS, *The Design and Algorithms of a Verification Condition Generator*, University College Dublin.
- 1998–2003 BSc, EE, Politehnica University of Bucharest. My average grade is 9.75 out of 10. My diploma dissertation is on *Traffic Models for Data Networks*.
- 2001 Internship in the Darmstadt Institute for Microelectronics, funded by a [DAAD](#) scholarship.
- 1994–1998 “Gheorghe Șincai” highschool, București

## Industry

- 2008 I experimented with using kd-trees and R-trees to add orthogonal range queries on top of [BigTable](#) as a Software Engineer Intern in [Google](#).
- 2004–2005 I helped design and develop a translator between two languages used in hardware verification. The client was [Synopsys](#) and I was working in a team of ten within [NoBug Consulting](#).
- 2003–2004 I designed and developed a PSL frontend for the model checker [RuleBase](#) while working for [NoBug Consulting](#).
- 2003 I helped design and develop a translator from [PSL](#) to finite automata with counters, whose implementation is generated in a variety of procedural languages. I worked in a team of two within [NoBug Consulting](#).

## Open Source Contributions

- 2011... [TOPL](#) is a runtime verifier for temporal properties of Java programs.
- 2010... I help maintain the verifier [coreStar](#) and its frontend [jStar](#).
- 2007... [FreeBoogie](#) is a program verifier backend.
- 2007–2010 [AstGen](#) generates code from abstract grammars. It is part of [FreeBoogie](#).
- 2008–2010 [CLOPS](#) is a code generator for command line parsing.
- 2005–2009 I maintained [ESC/Java](#) and I implemented its reachability analysis.
- 2008 Part of my work in Google is included in [UzayGezen](#).
- 2005 [CFind](#) indexes and searches a hard-drive. I might improve it sometime.