Rodrigo Rocha

Edinburgh, United Kingdom
+44 7729715356

rcor.cs@gmail.com

https://rcor.me

github.com/rcorcs

About

In addition to my research on compilers, I am also a collaborator to the open-source projects. For me, making research innovations accessible to the wider world is as important as developing them in the first place.

Research Interests

Compiler constructions
Compiler optimizations
Static binary translation
Runtime systems

Open Source

LLVM Core

microsoft/llvm-mctoll
Static Binary Translation

Mentoring

BSc & MSc Graduation Projects
University of Edinburgh
TU Munich
PUC Minas

Google Summer of Code
LLVM: Hot-Cold Code Splitting

Research Indicators

Publications: 29
Citations: 106
H-index: 7
i10-index: 3

Talks

EuroLLVM 2018
Look-Ahead SLP Vectorization

EuroLLVM 2019
Function Merging (SRC: first place)

Education

2017 - 2021
University of Edinburgh

PhD in Computer Science
Thesis: Reducing Code Size with Function Merging
Advisors: Hugh Leather, Murray Cole, Zheng Wang

2016 - 2017
University of Edinburgh

MSc by Research in Computer Science
Thesis: Online Iterative Compilation Guided by Profiling
Advisors: Hugh Leather, Murray Cole, Zheng Wang

2013 - 2015
UFMG
Brazil

MSc in Computer Science
Thesis: Stream processing with Hadoop for Big Data
Advisors: Dorgival Guedes, Wagner Meira

2016 - 2017
PUC Minas
Brazil

BSc in Computer Science
Outstanding Student Award
Summa cum laude

Work Experience

Since 2021
University of Edinburgh

Research Associate
Principal investigator: Michael O'Boyle
External collaborator: Pramod Bhatotia (TU Munich)

2015 - 2016
PUC Minas
Brazil

Assistant Professor
Sole lecturer for undergraduate courses
Computer Science and Information Systems

Publications

LCTES '21

HyFM: Function merging for free
Proposed HyFM, a function merging technique that works on the block level, reducing compile-time and memory usage.

PLDI '20

Effective function merging in the SSA form
Proposed SalSSA, a better function merging technique based on sequence alignment that fully supports the SSA form.

CC '20

Vectorization-aware loop unrolling with seed forwarding
Proposed VALU, a heuristic that takes vectorization into account when making its loop unrolling decisions.

CGO '19
Best Paper Award

Function merging by sequence alignment
Proposed the first function merging technique based on sequence alignment. This technique is can merge any two functions.

CGO '19

Super-Node SLP: Optimized vectorization for code sequences containing operators and their inverse
Improved SLP vectorization using algebraic properties on chains of commutative operations and their inverse elements.

PACT '18

VW-SLP: Auto-vectorization with adaptive vector width
Improved SLP vectorization using algebraic properties on chains of commutative operations and their inverse elements.

CGO '18

Look-Ahead SLP: Auto-vectorization in the presence of commutative operations
Improved SLP vectorization by reordering operands across a chain of commutative operations to uncover more isomorphic code.

SPC '18

Automatic parallelization of recursive functions with rewriting rules
Proposed rewriting rules based on algebraic properties for automatically parallelizing certain classes of recursive functions.

CCPE '17

TOAST: Automatic tiling for iterative stencil computations on GPUs
Proposed a framework that uses genetic algorithm for tuning the tiling of iterative stencil. It includes paging for large inputs.