

Curriculum Vitae

November 18, 2017

KC Sivaramakrishnan

Computer Laboratory
University of Cambridge
sk826@cl.cam.ac.uk

<http://kcsrk.info>

15 JJ Thomson Avenue
Cambridge CB3 0FD
Tel: +44 79828 42499

❖ Summary

I am interested in the design and implementation of concurrent programming languages targeting scalable platforms such as many-core processors and compute clouds. My research spans programming models, compilers, static analysis, schedulers, threading systems, and memory management.

❖ Education

PhD — Computer Science

Thesis Title: [Functional Programming Abstractions for Weakly Consistent Systems](#)
Advisor: Suresh Jagannathan

May 2011 – Dec 2014
Purdue University, USA

Master of Science — Computer Science

Aug 2008 – May 2011
Purdue University, USA

Bachelor of Engineering — Computer Science and Engineering

Aug 2004 – May 2008
PSG College of Technology
Anna University, India

❖ Experience

Senior Research Associate, University of Cambridge

Advisors: Alan Mycroft, Anil Madhavapeddy

Nov 2017 – present
Cambridge, UK

Technical director of OCaml Labs leading the development of [Multicore OCaml](#) project. Applying programming languages and program verification to solve extreme-scale parallelism and distribution.

Research Fellow, Royal Commission for the Exhibition of 1851

Oct 2015 – present

Research Fellow, Darwin College, Cambridge

Oct 2015 – present

Research Associate, University of Cambridge

Dec 2014 – Oct 2017

Research Assistant, Purdue University

Advisor: Suresh Jagannathan

Aug 2008 – Dec 2014
West Lafayette, IN, USA

My research focused on discovering new language abstractions and developing runtime system techniques to ease programming weakly consistent systems. To this end, I have built [MultiMLton](#), a parallel and distributed extension of MLton Standard ML compiler and runtime and [Quelea](#), a shallow extension of Haskell for declarative programming over eventually consistent data stores.

Teaching Assistant, Purdue University

Undergraduate C Programming (CS180)
Graduate Programming Languages (CS565)

West Lafayette, IN, USA
Aug 2012 – Dec 2012
Aug 2011 – Dec 2011

My tasks included designing and evaluating weekly projects, office hours for one-on-one instruction, and grading.

Research Intern, Microsoft Research, Cambridge

Advisors: Tim Harris, Simon Marlow, and Simon Peyton Jones

Feb 2012 – May 2012
Cambridge, UK

I developed a concurrency substrate for Glasgow Haskell Compiler (GHC) to allow programmers to modularly implement user-level schedulers and concurrency libraries for Haskell threads in Haskell, without having to re-engineer critical runtime system components. The concurrency substrate is built around one-shot continuations and uses transactional memory for coordination.

Research Intern, Samsung Information Systems America (R&D)

May 2010 – Aug 2010

Advisor: Daniel Waddington

San Jose, CA, USA

I was part of the core team that developed SNAPPLE programming language – a safe and concurrent extension of C++ targeted at many-core processors. The task involved designing language extensions for concurrency, compiler extensions for safety, and a runtime for executing large number of lightweight threads. SNAPPLE was implemented as a veneer on top of C++ using LLNL Rose source-to-source compiler.

Intern, Advanced Numerical Research and Analysis Group

Dec 2007 – Apr 2008

Advisor: Sankar Chnab

Hyderabad, India

As a part of the Compiler Engineering group, I ported Kaffe, an open source Java VM to an embedded microprocessor ANUPAMA and a desktop processor ABACUS. Developed a lightweight threading subsystem, and implemented a JIT backed for ABACUS.

❖ Journal Publications

- J6 **Concurrent System Programming with Effect Handlers** Nov 2017
Stephen Dolan, Spiros Eliopoulos, Daniel Hillerstrm, Anil Madhavapeddy, KC Sivaramakrishnan, Leo White
Post-proceedings of the Symposium on Trends in Functional Programming (TFP) (accepted)
- J5 **Eff directly in OCaml** Oct 2017
Oleg, Kiselyov, KC Sivaramakrishnan
Post-proceedings of the ML Workshop (accepted)
- J4 **Composable Scheduler Activations for Haskell** Jun 2016
KC Sivaramakrishnan, Tim Harris, Simon Marlow, Simon Peyton Jones
Journal of Functional Programming (JFP)
- J3 **Representation without Taxation: A Uniform, Low-Overhead, and High-Level Interface to Eventually Consistent Key-Value Stores** Mar 2016
KC Sivaramakrishnan, Gowtham Kaki, Suresh Jagannathan
IEEE Data Engineering Bulletin, 39(1): 52 – 64
- J2 **MultiMLton: A Multicore-aware Runtime for Standard ML** Nov 2014
KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
Journal of Functional Programming (JFP), 24(6): 613 – 674
- J1 **Efficient Sessions** Feb 2013
KC Sivaramakrishnan, Mohammad Qudeisat, Lukasz Ziarek, Karthik Nagaraj, Patrick Eugster
Science of Computer Programming (SCP), 78(2): 147 – 167
Invited paper

❖ Conference Publications

- C9 **Continuation Passing Style for Effect Handlers** Sep 2017
Daniel Hillerstrm, Sam Lindley, Robert Atkey, KC Sivaramakrishnan
International Conference on Formal Structures for Computation and Deduction (FSCD)
- C8 **DaLi : Database as a Library** May 2017
Gowtham Kaki, KC Sivaramakrishnan, Thomas Gazagnaire, Anil Madhavapeddy, Suresh Jagannathan
The 2nd Summit on Advances in Programming Languages (SNAPL)
Oral Presentation
- C7 **Declarative Programming over Eventually Consistent Data Stores** Jun 2015
KC Sivaramakrishnan, Gowtham Kaki, Suresh Jagannathan
International Conference on Programming Language Design and Implementation (PLDI)
- C6 **Rx-CML: A Prescription for Safely Relaxing Synchrony** Jan 2014
KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
Symposium on Practical Aspects of Declarative Languages (PADL)

- C5 [A Coherent and Managed Runtime for ML on the SCC](#) Nov 2012
 KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
Many-core Architecture Research Community Symposium (MARC)
Best paper award
- C4 [Eliminating Read Barriers through Procrastination and Cleanliness](#) Jun 2012
 KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
International Symposium on Memory Management (ISMM)
- C3 [Composable Asynchronous Events](#) Jun 2011
 Lukasz Ziarek, KC Sivaramakrishnan, Suresh Jagannathan
International Conference on Programming Language Design and Implementation (PLDI)
- C2 [Efficient Session Type Guided Distributed Interaction](#) June 2010
 KC Sivaramakrishnan, Karthik Nagaraj, Lukasz Ziarek, Patrick Eugster
International Conference on Coordination Models and Languages (COORDINATION)
- C1 [Partial Memoization of Concurrency and Communication](#) Sep 2009
 Lukasz Ziarek, KC Sivaramakrishnan, Suresh Jagannathan
International Conference on Functional Programming (ICFP)

❖ Workshop Publications

- W13 [A Memory Model for Multicore OCaml](#) Sep 2017
 Stephen Dolan and KC Sivaramakrishnan
OCaml Workshop
- W12 [Effectively Tackling the Awkward Squad](#) Sep 2017
 Stephen Dolan, Spiros Eliopoulos, Daniel Hillerstrm, Anil Madhavapeddy, KC Sivaramakrishnan, Leo White
OCaml Workshop
- W11 [Mergeable Types](#) Sep 2017
 Gowtham Kaki, KC Sivaramakrishnan, Samodya Abeysiriwardane, Suresh Jagannathan
ML Workshop
- W10 [Concurrent System Programming with Effect Handlers](#) Jun 2017
 Stephen Dolan, Spiros Eliopoulos, Daniel Hillerstrm, Anil Madhavapeddy, KC Sivaramakrishnan, Leo White
Symposium on Trends in Functional Programming (TFP)
- W9 [Eff directly in OCaml](#) Mar 2017
 Oleg Kiselyov and KC Sivaramakrishnan
JSSST Workshop on Programming and Programming Languages
- W8 [Lock-free programming for the masses](#) Sep 2016
 KC Sivaramakrishnan, Tho Laurent
OCaml Workshop
- W7 [Compiling Links Effect Handlers to the OCaml Backend](#) Sep 2016
 Daniel Hillestrm, Sam Lindley, KC Sivaramakrishnan
ML Workshop
- W6 [Eff Directly in OCaml](#) Sep 2016
 Oleg Kiselyov and KC Sivaramakrishnan
ML Workshop
- W5 [Effective Concurrency with Algebraic Effects](#) Sep 2015
 Stephen Dolan, Leo White, KC Sivaramakrishnan, Jeremy Yallop and Anil Madhavapeddy
OCaml Workshop

- W4 **Migrating MultiMLton to the Cloud** Sep 2013
 KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
ML Workshop
- W3 **Scalable Lightweight Task Management Schemes for MIMD Processors** Apr 2011
 Daniel G. Waddington, Chen Tian, KC Sivaramakrishnan
Workshop on Systems for Future Multi-Core Architectures (SFMA)
- W2 **The Design Rationale for MultiMLton** Sep 2010
 Suresh Jagannathan, Armand Navabi, KC Sivaramakrishnan, Lukasz Ziarek
ML Workshop
- W1 **Lightweight Asynchrony using Parasitic Threads** Jan 2010
 KC Sivaramakrishnan, Lukasz Ziarek, Raghavendra Prasad, Suresh Jagannathan
Workshop on Declarative Aspects of Multicore Programming (DAMP)

❖ Technical Reports and Drafts

- T1 **Featherweight Threads for Communication** Nov 2011
 KC Sivaramakrishnan, Lukasz Ziarek, Suresh Jagannathan
Purdue University Computer Science Technical Report – TR-11-018

❖ Teaching/Advising

- Guest Lectures:
 - Arrows, Advanced Functional Programming, University of Cambridge, Lent '16.
 - Debugging, Programming in C and C++, University of Cambridge, Michaelmas '15.
- Supervisions at University of Cambridge:
 - Databases, Lent '17, Michaelmas '17, Lent '16.
 - Concurrent and Distributed Systems, Lent '17, Michaelmas '17, Lent '16, Michaelmas '16, Lent '15.
 - Algorithms, Lent '15.
 - Object-oriented Programming, Michaelmas 2015–16.
- Teaching assistantships at Purdue University
 - Undergraduate C Programming (CS180), Aug 2012 – Dec 2012.
 - Graduate Programming Languages (CS565), Aug 2011 – Dec 2011.
- Projects supervised:
 - Matevz Polijanc, University of Cambridge, A Reactive Programming model in OCaml, Oct 2017 – Present.
 - Charlie Crisp, University of Cambridge, A Blockchain in Pure OCaml, Oct 2017 – Present.
 - Henry Mercer, University of Cambridge, Systematic Concurrency Testing for Multicore OCaml, Oct 2017 – Present.
 - Nicolas Assouad, ENS Paris, Hardware Support for Composable Lock-free Transactions, Mar 2017 – Jun 2017.
 - Matt Harrison, University of Cambridge, Secure Decentralized Apps, Sep 2016 – present.
 - Maxime Lesourd, ENS de Lyon, Verified CPS translation of handlers, Sep 2016 – Mar 2017.
 - Philip Dexter, Binghampton University, Approximate computing for OCaml, May 2016 – Aug 2016.
 - James Wright, University of Cambridge, Mechanized semantics of Algebraic Effects in OCaml, Sep 2015 – Mar 2016.
 - Armael Gueneau, ENS de Lyon, Algebraic Effects for js_of_ocaml, Sep 2015 – Mar 2016.
 - Theo Laurent, ENS, Reagents for Multicore OCaml, May 2015 – Aug 2015.
 - Guillain Potron, ENS de Lyon, Semantics of Irmin branch-consistent data store, March 2015 – Aug 2015.

❖ Talks

A deep dive into Multicore OCaml Garbage Collector System Research Group Seminar	<i>Jul 2017</i> Computer Laboratory, University of Cambridge
Multicore OCaml GC JaneStreet Group	<i>Jun 2017</i> New York, NYC
Composable lock-free programming for Multicore OCaml ABCD Meeting	<i>Nov 2016</i> University of Edinburgh
Practical Algebraic Effect Handlers in Multicore OCaml LFCS Seminar	<i>Nov 2016</i> University of Edinburgh
Effective Concurrency and Parallelism in Multicore OCaml PL Seminar	<i>Nov 2016</i> Indian Institute of Technology, Madras
Effective Concurrency and Parallelism in Multicore OCaml PL Seminar	<i>Nov 2016</i> Indian Institute of Technology, Bombay
Effective parallelism with Reagents Facebook Faculty Summit	<i>Sep 2016</i> London, UK
Multicore OCaml and Programming with Reagents LDN Functionals	<i>Aug 2016</i> Jane Street UK, London
Effect handlers in Multicore OCaml Dagstuhl Seminar	<i>Mar 2016</i> Dagstuhl, Germany
Arrows and Reagents Invited Lecture, Advanced Functional Programming	<i>Mar 2016</i> Cambridge, UK
Concurrent and Multicore OCaml: A deep dive Facebook Tech Talk	<i>Jan 2016</i> Menlo Park, CA
OCaml Platform: Update OCaml Consortium Meeting	<i>Nov 2015</i> Paris, France
Multicore OCaml: Update OCaml Developer's Meeting	<i>Nov 2015</i> Paris, France
Silence is Golden: Controlling Communication and Coordination in Distributed Databases Darwin College Science Seminar	<i>Oct 2015</i> Cambridge, UK
Effective Concurrency with Algebraic Effects OCaml Workshop 2015	<i>Sep 2015</i> Vancouver, Canada
Quelea: Declarative Programming over Eventually Consistent Data Stores Computer Laboratory, University of Cambridge	<i>Apr 2015</i> Cambridge, UK
Functional Programming Abstractions for Weakly Consistent Systems PhD Defense	<i>Dec 2014</i> Purdue University
Functional Abstractions for Practical and Scalable Concurrent Programming Invited Lecture	<i>Mar 2014</i> Microsoft Research, Cambridge, UK
Rx-CML: A Prescription for Safely Relaxing Synchrony PADL 2014	<i>Jan 2014</i> San Diego, CA
Migrating MultiMLton to the Cloud ML Workshop 2013	<i>Sep 2013</i> Boston, MA
A Coherent and Managed Runtime for ML on the SCC MARC 2012	<i>Nov 2012</i> RWTH Aachen

Eliminating Read Barriers through Procrastination and Cleanliness

ISMM 2012, Beijing
Wrestling Wednesdays, Microsoft Research, Cambridge

Jun 2012
May 2012

Lightweight Concurrency in GHC

Wrestling Wednesdays

May 2012
Microsoft Research, Cambridge

Efficient Session Type guided Distributed Interaction

COORDINATION 2012

Jun 2012
CWI Amsterdam

❖ Service

- Organizer, [Dagstuhl Seminar on "Algebraic Effect Handlers go Mainstream"](#), Apr 2018.
- Program Committee member: PMLDC@ECOOP 2017, Off-the-beaten track (OBT) 2017, OCaml Workshop 2016, SPLASH-MARC symposium, 2013.
- Artifact Evaluation Committee member: PLDI 2015, PPOPP/CGO 2016.
- Reviewer: ECOOP, TODS, JFP, POPL, ICFP, ASPLOS, TLDI, Concurrency and Computation: Practice and Experience, Software: Practice and Experience.
- Organizer for Darwin College Science Seminar Series, Oct 2015 – May 2017.

❖ Awards and Recognitions

- Research Fellowship, Royal Commission for the Exhibition of 1851, 2015–2018, £102,000.
- Research Fellowship, Darwin College, Cambridge, 2015–2018, £900.
- Maurice H. Halstead Memorial Award for outstanding research in Software Engineering, Purdue University, 2014, \$4,000.
- Best paper award at Many-core Architecture Research Symposium at RWTH-Aachen, 2012, \$1,000.
- Invited paper in Science of Computer Programming, Vol. 78, Iss. 2 (Feb 2013).
- Glasgow Haskell Compiler (GHC) Committer.
- SIGPLAN PAC travel grant for PLDI 2012 and POPL 2014, \$1,500 each.
- NSF travel grant for ICFP 2013, \$2,000.