

Curriculum Vitae

February 17, 2015

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

GPA: 3.94/4

Aug 2008 – May 2011
Purdue University, USA

Bachelor of Engineering — Computer Science and Engineering

GPA: 9.55/10

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

❖ Experience

Research Associate, University of Cambridge

Advisor: Anil Madhavapeddy

Dec 2014 – present
Cambridge, UK

I am working on a variety of projects under the OCaml Labs initiative in the Computer Laboratory.

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)

Advisor: Daniel Waddington

May 2010 – Aug 2010
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

Advisor: Sankar Chnab

Dec 2007 – Apr 2008

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

- 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

- 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

- 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

T2 [Composable Scheduler Activations for Haskell](#) Dec 2014
KC Sivaramakrishnan, Tim Harris, Simon Marlow, Simon Peyton Jones
Under consideration for Journal of Functional Programming (JFP)

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