Simon Cooksey

simon@graymalk.in
+44 (0) 7887 393086

Research

I study modern multi-processor computers to understand the behaviours admitted by their complex micro-architectural designs. My research has focussed particularly on so-called *weak memory behaviours*, which are exhibited on machines that allow out-of-order execution and caching.

These intricate behaviours present challenges for programmer reasoning. Formal models help to clarify precisely what is allowed, and my tools bridge the gap between mathematical formalisation and programmer intuition about computer behaviour. My tools make these formal models executable. This informs the development of the model by allowing quick refinement of definitions, and also permits a programmer to probe the behaviour of a model to grasp the precise meaning of a given program.

These models stand as cutting edge specifications for verification of concurrent programs executing on modern high-performance hardware.

PwTer *Pomsets with Predicate Transformers* (PwT) is a recent fully denotational semantics for weak memory consistency. It captures program dependencies by embedding a logic into predicate transformers which compose to create proof burdens necessary to remove dependencies which are syntactic only from list of dependencies actually present in a program. I developed an OCaml tool to automatically evaluate PwT over a series of litmus tests. *Appears in the proceedings of POPL 2022.*

MRD and MRDer *Modular Relaxed Dependencies* (MRD) is a partially denotational semantics for weak memory consistency in C and C++. As well as helping to define the denotation, I built an evaluation tool (MRDer) in OCaml to enable fast calculation of the semantics against a corpus of litmus tests. Further, I proved meta-theoretic properties about the semantics with respect to industry standard models of C/C++. *Appears in the proceedings of ESOP 2020.*

Mixed-Proxy Extensions for NVIDIA PTX An extension of NVIDIA's PTX ISA to support mixing load and store operations that target different *proxies* of memory. This gives well-defined semantics to programs with both, for example, constant loads and generic stores to the same address. The memory model extension was built in the relational model checking tool, Alloy, and we showed that our extension to the PTX model is sound and complete with respect to the existing industrial model. *To appear in the proceedings of ISCA 2022.*

PrideMM PrideMM is a tool written in OCaml which provides an API for building Second Order logical formulae and uses this API to express memory models. We use cutting edge *Quantified Boolean Formulae* (QBF) solvers to efficiently simulate a new class of memory model which solve the thin-air problem. We encode the problems in a high-level second order logic giving us flexibility in problem expression. This then gets translated into a QBF model checking problem for a solver to efficiently execute. *Appears in the post-proceedings of TAPAS 2019.*

Education

The University of Kent — Canterbury
Computer Science PhD Candidate, Passed subject to Minor Corrections — Sept. 2016 – Sept. 2021
Computer Science with a Year in Industry BSc (Hons), *First class* — Sept. 2012 – Jul. 2016

Internships & Employment

The University of Kent — Canterbury, United Kingdom
Research Associate — Jul. 2020 – Present
I have a position to continue my PhD research with collaborators at Kent. This work will focus on the impact of memory systems on program correctness on novel hardware platforms. This position is funded by a series of successful grant applications to which I was a co-author.

The University of Kent — Canterbury, United Kingdom
Assistant Lecturer — Sept. 2016 – Jun. 2020
I participated in teaching a selection of modules. This involved content delivery in seminars and terminals, as well as marking and providing feedback on student work.

NVIDIA — Santa Clara, California
Research Intern — Jul. 2018 – Dec. 2018
As an intern at NVIDIA I extended the Memory Consistency Model for NVIDIA's virtual instruction set (PTX) to support "memory proxies". This enables writing well-defined programs which mix generic load and store operations with specialised load and store operations for texture, surface and constant accesses. The result of this work now forms public NVIDIA documentation in the PTX manual.

XMOS — Bristol, United Kingdom
Development Intern — Aug. 2014 – Aug. 2015
At XMOS I built tools, libraries, and test infrastructure for the xCORE series of multicore processors.

Skills

- **OCaml.** Using OCaml to implement mathematical artefacts for mechanised evaluation.
- **Weak memory consistency.** Understanding and constructing mathematical models for multiprocessor systems, both in hardware and software.
- **Formal Hardware Specification.** Using internal hardware specifications and design manuals to capture a mathematical abstraction of machine behaviour, using *Alloy*.
- **Programming languages.** Regular user of C/C++, JavaScript, OCaml, and Python.

Publications

- *Mixed-Proxy Extensions for the NVIDIA PTX Memory Consistency Model*: *To appear,* 49th IEEE/ACM International Symposium on Computer Architecture, 2022. *Daniel Lustig, Simon Cooksey, Olivier Giroux*
- *The Leaky Semicolon: Compositional Semantic Dependencies for Relaxed-Memory Concurrency*: 49th ACM SIGPLAN Symposium on Principles of Programming Languages, 2022. *Alan Jeffery, James Riely, Mark Batty, Simon Cooksey, Ilya Kaysin, Anton Podkopaev*
- *Modular Relaxed Dependencies in Weak Memory Concurrency*: 29th European Symposium on Programming, 2020. *Marco Paviotti, Simon Cooksey, Anouk Paradis, Daniel Wright, Scott Owens, Mark Batty*
- *P1780 Modular Relaxed Dependencies: A new approach to the Out-Of-Thin-Air Problem*: ISO C/C++ Standards Committee meeting, Cologne, 2019. *Mark Batty, Simon Cooksey, Scott Owens, Anouk Paradis, Marco Paviotti, Daniel Wright*
- *PrideMM: Second Order Model Checking for Memory Consistency Models*: 10th Workshop on Tools for Automatic Program Analysis, 2019. *Simon Cooksey, Sarah Harris, Mark Batty, Radu Grigore, and Mikoláš Janota*

Prizes

- **Kent Postgraduate Prize**: (July 2020) Recognising the significant impact of my research.

Grants

- **Complementing Capabilities: introducing pointer-safe programming to DSbD tech**: *Researcher Co-Investigator* (£494,770) UK Innovation Funding. ISCF digital security by design - software ecosystem development
- CapC**: Capability C semantics, tools and reasoning**: *Named Researcher and Grant Co-Author* (£596,634) UKRI: Digital Security by Design
- **Fixing the thin-air problem: ISO dissemination**: *Named Researcher and Grant Co-Author* (£60,455) UK Research Institute: Verified Trustworthy Software Systems

Conference Attendance

- **POPL 2022**: Presenting work with collaborators.
- **ISO C++ Standards Committee**: Demonstrating a compositional semantics for C/C++ concurrency which avoids the out-of-thin-air problem to the C++ standards committee.
- **Aarhus Concurrency Workshop**: Explained the issues surrounding simulating the latest memory models and presented an early version of PrideMM.
- **PLMW / POPL 2017**: Attended the Programming Languages Mentoring Workshop at POPL'17 in Paris with a grant from the ACM.

Teaching

As an Assistant Lecturer, and Research Associate, I have taught on of the programming languages and systems modules in the University of Kent Computer Science course.

- Short lecture series on OCaml for programming language implementation.
- Guest lecture for Masters C++ students, introducing the C++ memory consistency model
- Seminar leader for:
  - CO545: Functional and Concurrent Programming
  - CO657: Internet of Things
  - CO658: Programming Language Implementation
  - CO661: Theory and Practice of Concurrency
  - CO663: Programming Languages: Application and Design
  - CO883: Systems Architecture

References

Available on request.