

Matt Luckcuck BSc (Hons), MSc, PhD(Ebor)

20 Lentworth Court, Liverpool, L17 6GD, UK

Email: m.luckcuck@tutanota.com

Summary

I'm a Post-Doctoral Researcher in the Department of Computer Science at Maynooth University, Ireland. My research interests include formal behavioural specification, model-checking, runtime verification, and safety verification and assurance for autonomous, automated, and other safety-critical systems.

Currently, I'm researching formal verification techniques for automated aircraft engine control systems, as part of the the Verification and Validation of Automated Systems' Safety and Security (VALU3S) project. This work will involve an extensive survey paper, and investigating integrating formal methods with Simulink models. My previous research work includes: building formal specifications from safety documents, investigating how to apply heterogeneous verification approaches to different components of an autonomous software system, and writing guidance for developing safe autonomous systems – with the UK's Office for Nuclear Regulation.

I received my PhD from the University of York UK, where I was supervised jointly by Professors Ana Cavalcanti and Andy Wellings. My PhD work involved developing a formal model of a new safety-critical language, Safety-Critical Java, using the state-rich process algebra *Circus*, which combines Z and CSP.

I have teaching experience at a variety of age-ranges and ability-levels, and have found it both enjoyable and rewarding. I have assisted in the teaching of several degree modules, at both undergraduate and postgraduate levels. Also, I have tutored an A-Level Computing student one-to-one, improving her result by two grade boundaries. Finally, I have taught school-age children basic programming skills, often using Scratch.

Selected Publications

- Rafael C. Cardoso, Marie Farrell, **Matt Luckcuck**, Angelo Ferrando, and Michael Fisher ‘*Heterogeneous Verification of an Autonomous Curiosity Rover*’ in NASA Formal Methods Symposium, 2020
- **Matt Luckcuck**, Marie Farrell, Louise Dennis, Clare Dixon, and Michael Fisher ‘*Formal Specification and Verification of Autonomous Robotic Systems: A Survey*’ in ACM Computing Surveys, 52(5) 2019
- Michael Fisher, Emily Collins, Louise A. Dennis, **Matt Luckcuck**, Matt Webster, Mike Jump, Vincent Page, Charles Patchett, Fateme Dinmohammadi, David Flynn, Valentin Robu, Xingyu Zhao ‘*Verifiable Self-Certifying Autonomous Systems*’ in ISSRE Workshops, 2018
- Marie Farrell, **Matt Luckcuck**, and Michael Fisher ‘*Robotics and Integrated Formal Methods: Necessity meets Opportunity*’ in iFM2018, 5-7th of September 2018
- **Matt Luckcuck**, Andy Wellings, and Ana Cavalcanti ‘*Safety-Critical Java: Level 2 in Practice*’ in Concurrency and Computation: Practice and Experience, 29(6) 2016
- **Matt Luckcuck**, Ana Cavalcanti, and Andy Wellings ‘*A Formal Model of the Safety-Critical Java Level 2 Paradigm*’ in Proceedings of the International Conference on Integrated Formal Methods, iFM 2016
- **Matt Luckcuck** ‘*A Formal Model for the Safety-Critical Java Level 2 Paradigm*’ in Proceedings of the Doctoral Symposium of Formal Methods 2015, DSFM 2015 [Honourable Mention]
- Andy Wellings, **Matt Luckcuck**, and Ana Cavalcanti ‘*Safety-critical Java Level 2: Motivations, Example Applications and Issues*’ in Proceedings of the 11th International Workshop on Java Technologies for Real-time and Embedded Systems, JTRES 2013

Education

PhD in Computer Science *University of York, UK* (2012 – 2016)

Safety-Critical Java Level 2: Applications, Modelling, and Verification

My PhD work models Safety-Critical Java (SCJ) using the state-rich process algebra *Circus*, which combines elements of Z and CSP. SCJ adopts a new programming paradigm for applications that must be certified. SCJ programs use a particular concurrency model and use region-based memory management (instead of garbage collection); specialised virtual machines are available to execute SCJ programs. It is organised into three compliance levels, of ascending complexity. My PhD focuses on the most complex compliance level, the programs of which are highly concurrent, potentially multi-processor, and make use of suspension and a variety of release patterns. My PhD provides the most complex compliance level of SCJ with its first semantics, enables further integration with other *Circus* semantics for SCJ, and provides automatic translation from SCJ to my model.

Supervisors: Ana Cavalcanti and Andy Wellings

MSc with Merit in Computing *University of York, UK* (2011 – 2012)

Bsc (Hons) First Class in Computer Science *University of Wolverhampton, UK* (2007 – 2011)

Work Experience

Post-Doctoral Researcher: VALU3S *Maynooth University, Ireland* (2021 – Current)

I'm researching formal verification techniques for automated aircraft engine control systems, as part of the the Verification and Validation of Automated Systems' Safety and Security (VALU3S) project. My work on this project started with an extensive survey paper of formal verification for aerospace systems, and focused on integrating formal methods with the Simulink models provided by the project's use case provider.

Research Associate: RAIN Hub *University of Liverpool and University of Manchester, UK* (2017 – 2021)

My research focussed on formal verification for robotics and autonomous systems in nuclear environments, on the Robotics and AI in Nuclear (RAIN) Hub. Originally my group was based at the University of Liverpool, but we later moved to the University of Manchester. My work on this project began with an extensive survey paper, and then focussed on linking heterogeneous verification approaches applied across an autonomous software system, and runtime verification of an autonomous system's behaviour. I also lead a collaboration with the UK's Office for Nuclear Regulation on developing guidance for developers of autonomous systems that ensures their systems are amenable to robust verification and can provide useful assurance evidence.

Adult Education Tutor *York Learning, UK* (2017)

I delivered a 6 week adult-education course teaching introductory programming skills using Python. This involved presenting a lecture and then guiding the students through Python programming exercises designed to reinforce the material in the lecture and teach them the basic practical skills of programming. The lectures and resources were designed by me from scratch.

Tutor and Curriculum Design *York Maker Hub, UK* (2017)

Tutor I taught children (ranging from 8-14) basic programming skills, in a busy environment that allowed the children some freedom in choosing the platform they used. The day-long sessions provide the opportunity for the children to build and program various projects on a variety of platforms, including Scratch, Makey Makey, and Lego Mindstorms. This role also involved supervising the children during breaks and lunch.

Curriculum Design I also helped to develop a new pathway for older or more advanced children who wanted to learn programming. This involved reviewing the currently available projects, collaboratively designing new Python projects, and then writing up the project instructions sheets and resources.

Research Assistant *University of York, UK* (2015)

During my PhD I was employed, part-time, to build a software tool to automatically translated program code for Safety-Critical Java into a formal model written in the state-rich process algebra *Circus*. This role was closely related to my thesis, so I managed the work alongside my PhD.

Teaching and Outreach

- During my time at the University of Liverpool, I helped run an outreach activity for children aged 7–14 called *Lego Rovers*, which introduces the topic of autonomous systems. This involved running workshops and a ‘drop-in’ stand at larger events.
- Taught an 11 week adult-education course on basic programming skills, in Python. I designed the course and resources, gave lectures, then guided students through Python programming exercises.
- Taught basic programming skills using a range of platforms including Scratch and Lego Mindstorms. These were day-long sessions with a large, mixed age (8–15) and ability groups.
- Volunteered at a local Code Club, teaching 9–11 year olds basic programming skills using Scratch. The small group was of mixed ages and abilities. Each week we worked on a small game, each introducing increasingly complex techniques.
- Tutored an A-Level Computing student, one-to-one. We focussed on covering key topics, and improved the student’s exam grade by two grade boundaries. My contact time was 20 hours (1 hour per week).
- During my PhD (2013–2016) I was a post-graduate teaching assistant on several modules:
 - Workshops for two terms of a first year undergraduate module, introducing programming concepts: Python in the first term, where my contact time was 20 hours; and Java in the second, where my contact time was 28 hours. Each group contained around 65 students of very mixed ability.
 - Workshops for a \LaTeX module with around 12 PhD students and my contact time was 5 hours, and a \LaTeX for an MSc module, also with around 12 students.
 - Marked a first-year undergraduate programming module. There were 108 scripts for one exam and 130 for the other. This involved working to the mark scheme and keeping to the marking deadline.
 - Marked MSc Software Engineering group projects. There were eight groups of six students. This involved reading each group’s project report and grading it, along with supplementary material, against the module mark scheme; to a strict marking deadline.

Administration

- I co-chair the Formal Methods and Autonomous Systems (FMAS) workshop, which was started by me and a colleague. The first FMAS was held at Formal Methods 2019; the second was online due to Covid-19 disruptions. I take particular responsibility for organising the Programme Committee and advertising the workshop.
- In 2018/19 I organised a series of workshops on Safety Cases for Nuclear Robotics. The workshops drew attendees from academia, the nuclear industry, and the Office for Nuclear Regulation. Presentations and open discussions and these workshops focussed on ensuring that autonomous robotics used in the nuclear industry are safe.
- During my PhD, I was the Programme Committee chair for the York Doctoral Symposium on Computer Science and Electronics (YDS2014), a student conference organised at the University of York. I was responsible for recruiting the Programme Committee and ensuring that submissions were reviewed fairly. I also co-led the symposium’s interdepartmental Organising Committee.

Committee Membership

- **2019–Current:** Working Group Member IEEE P7009 Standard for Fail-Safe Design of Autonomous and Semi-Autonomous Systems

Reviewing

- Second Workshop on Formal Methods for Autonomous Systems (FMAS) 2020
- ACM Computing Surveys Journal (CSUR) 2020
- International Conference on Principles of Knowledge Representation and Reasoning (KR) 2020 [Subreviewer]

- International Joint Conference on Artificial Intelligence (IJCAI) 2020
- Workshop on the History of Formal Methods (HFM) 2020
- Journal of Autonomous Agents and Multi-Agent Systems (AAMAS) 2020
- International Conference on Integrated Formal Methods (iFM) 2019
- Formal Methods for Autonomous Systems Workshop (FMAS) 2019
- History of Formal Methods Workshop (HFM) 2019
- Science of Computer Programming (SCP) 2019
- International Journal of Adaptive Control and Signal Processing 2018 and 2019
- Journal of Software Testing, Verification and Reliability 2018
- York Doctoral Symposium (YDS) 2014, 2015, 2016, 2017
- International Workshop on Java Technologies for Real-time and Embedded Systems (JTRES) 2014
- Theoretical Aspects of Computing (ICTAC) 2013 [Subreviewer]

Conference and Workshop Presentations

- **2019:** International Conference on Integrated Formal Methods (iFM 2019)
- **2019:** Seminar on Autonomous Systems Verification at a Game Changers event, hosted by the National Nuclear Laboratory
- **2019:** Tutorial at CyPhyAssure Spring School, University of York
- **2016:** International Conference on Integrated Formal Methods (iFM 2016)
- **2015:** Doctoral Symposium at the International Symposium on Formal Methods (DSFM 2015)
- **2014:** Certifiable Java for Embedded Systems (CJ4ES)
- **2013:** Workshop on Java Technologies for Real-time and Embedded Systems (JTRES 2013)

Referees

Professor Michael Fisher — Current Supervisor
Department of Computer Science,
University of Manchester, Manchester,
M13 9PL, UK

Email: michael.fisher@manchester.ac.uk

Professor Ana Cavalcanti — PhD Supervisor
Department of Computer Science,
University of York, YO10 5GH, UK

Tel: +44 (0)1904 325478

Email: Ana.Cavalcanti@york.ac.uk