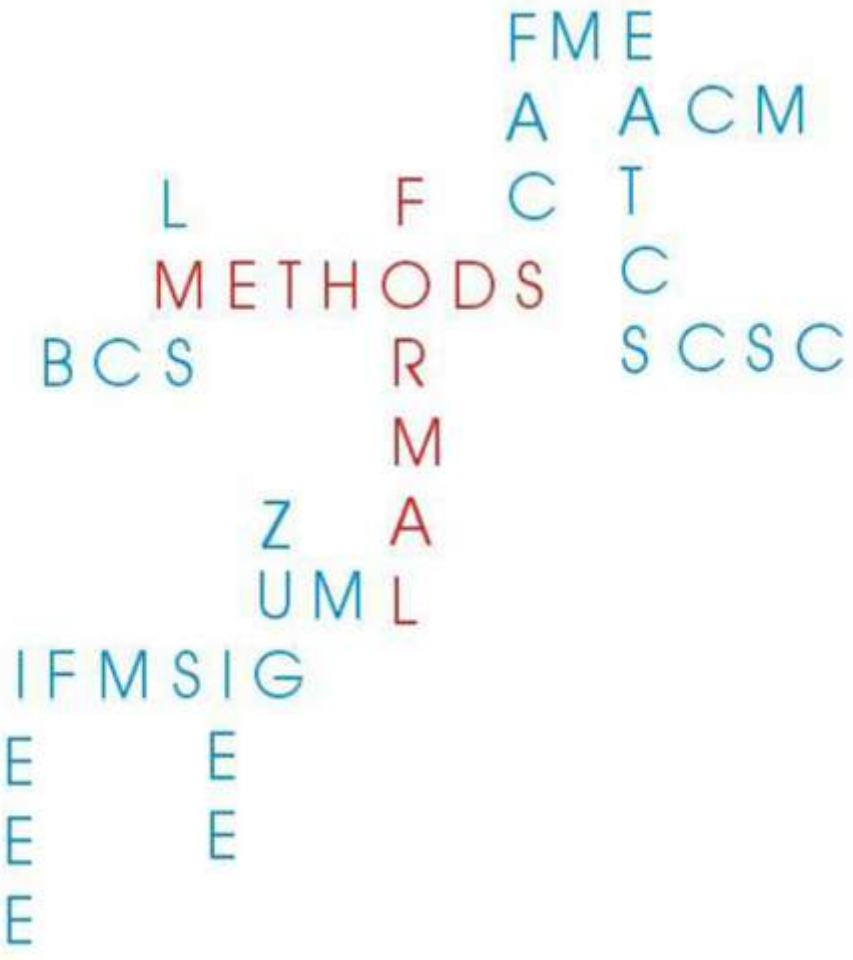
Issue 2025-2
July 2025

FACS FACTS



Formal Aspects of Computing Science Specialist Group

The Newsletter of the Formal Aspects of Computing Science (FACS) Specialist Group

ISSN 0950-1231

About FACS FACTS

FACS FACTS (ISSN: 0950-1231) is the newsletter of the BCS Specialist Group on Formal Aspects of Computing Science (FACS). FACS FACTS is distributed in electronic form to all FACS members.

Submissions to FACS FACTS are always welcome. Please visit the newsletter area of the BCS FACS website for further details at:

https://www.bcs.org/membership/member-communities/facs-formal-aspects-of-computing-science-group/newsletters/

Back issues of FACS FACTS are available for download from:

https://www.bcs.org/membership/member-communities/facs-formal-aspects-of-computing-science-group/newsletters/back-issues-of-facs-facts/

The FACS FACTS Team

Newsletter co-editors:

Tim Denvir **timdenvir@bcs.org**
Brian Monahan **brianqmonahan@googlemail.com**

Editorial team:

Jonathan Bowen, Tim Denvir, Keith Lines, Brian Monahan

Contributors to this issue:

Maurice ter Beek, Jonathan Bowen, Tim Denvir, Henri Habrias, Keith Lines, Zhiming Liu, Annabelle McIver, Brian Monahan, Andrei Popescu, André Videla

BCS-FACS websites:

BCS **https://facs.bcs.org**
LinkedIn **https://www.linkedin.com/groups/2427579/**
Wikipedia **https://en.wikipedia.org/wiki/BCS-FACS**

If you have any questions about BCS-FACS, please send these to Jonathan Bowen at: **jonathan.bowen@lsbu.ac.uk**.

Editorial

Dear reader,

Welcome to Issue 2025-2 of the FACS Newsletter. In this issue we have some particularly sad news to report, namely the passing of two well-known members of our FACS community: Jean-Raymond Abrial this year, and Mike Shields in 2023.

Jean-Raymond Abrial was well known to many of us in the community, both personally but also as the founder and progenitor of, not one but two, formal methods – the Z specification language and the B toolset. Jonathan Bowen and his close colleague, Henri Habrias, have contributed their appreciations of Jean-Raymond in the first article here.

Mike Shields may be less widely known in the FACS community, since his particular area of study was quite theoretical and concentrated on non-interleaving models of concurrency. This approach considers what some have more provocatively called "the study of true concurrency". His *Semantics of Parallelism* book is still in print and is considered by some to be the best and broadest account of non-interleaving semantics available. Mike's infamous alter ego, F. X. Reid, may be more familiar to some of us. Tim Denvir provides a personal reminiscence of Mike.

We are fortunate to have a rich crop of articles, reports and reviews for this issue. After the above memorials is a fluent, personal account, full of vitality, by Zhiming Liu, about his time working on the ProCoS project and, more particularly, on developing the Duration Calculus.

Next there is a report on Prof. Dr. Martin Fränzle and his Festschrift Symposium from Jonathan Bowen. This includes many anecdotal references to Jonathan's professional relationship with Martin Fränzle and their academic genealogies going back in history, way back to Hilbert, Gauss and even earlier.

That is followed by a report on the SETSS 2025 Workshop and School, Beijing, China, 17–23 May 2025, again by Jonathan Bowen. The proceedings of SETSS 2025, School on Engineering Trustworthy Software Systems, will be reported in an LNCS volume. Jonathan's account includes profiles of the principal speakers and his photographs indicate, among much else, that the delegates dined well!

FACS hosted a seminar in February this year on "Functional programming and dependent types for Metrology" by Keith Lines of NPL and André Videla of Strathclyde University. Brian Monahan gives a report on this.

Since the New Year, FACS has hosted a couple of collaborative seminars, one with Formal Methods Europe (FME) and the other with the London Mathematical Society (LMS), both enjoying a distinctly international flavour:

> Joint BCS-FACS / FME Seminar, "Formal Methods and Tools in Railways". The speaker was Maurice ter Beek, Pisa, Italy. Keith Lines provides a report.
>
> Joint BCS-FACS / LMS Seminar, "Probabilistic Datatypes" by Annabelle McIver of Macquarie University. Andrei Popescu reports on this.

Finally we have two reviews, both of which are somewhat out of the ordinary. The first is of a pure philosophy text called *The Structure of Pure Reason* by Danish philosopher Kai Sørlander. What initially intrigued us is that the work was translated from the original Danish by Dines Bjørner as a personal project. Dines will be well known to the majority of FACS readers, if not personally, then by repute.

The second review discusses an old, potentially overlooked, conference paper from 2013 containing an elegant application of formal methods.

Having said all this, we do greatly appreciate and look forward to contributions, including comments, from you, our readers.

We hope you enjoy FACS FACTS issue 2025-2.

Tim Denvir
Brian Monahan

Table of Contents

In memory of Jean-Raymond Abrial (1938–2025)

Henri Habrias
University of Nantes
France

Jonathan P. Bowen
London South Bank University
United Kingdom

A note on Jean-Raymond Abrial
by Jonathan Bowen

We have the sad news to report that Jean-Raymond Abrial passed away on 26 May 2025. Jean-Raymond was the progenitor and inspiration for not just one, but two major formal methods, namely the Z notation and the B-Method. He was a modest and independent researcher, admirably spanning theory and practice. He deserves much greater appreciation of his achievement than he has perhaps received during his lifetme. As a small tribute, I have updated Jean-Raymond's Wikipedia page and created a new page [7] for his 1996 magnum opus, *The B-Book* [1], after realising that I reviewed the book in 1997 [3].

We include an appreciation and biographical summary of Jean-Raymond's achievements below. For the future, we intend to have a longer set of memories about Jean-Raymond by his colleagues in the next issue of *FACS FACTS*, along the lines of the celebratory set of contributions for Tony Hoare's 90th birthday in 2024 [5]. Please contact the *FACS FACTS* editors if you would like to contribute.

An appreciation of Jean-Raymond Abrial
by Henri Habrias

Jean-Raymond Abrial died on 26 May 2025, the day before a conference held as part of the Journées Scientifiques at the University of Nantes, which focused on his work and its implementation in various fields.

J-R. Abrial was born in Versailles, France, in 1938. After studying at the Prytanée Militaire de la Flèche, he attended the Ecole Polytechnique. In 1960, he became a marine engineer. He was awarded a French government scholarship to Stanford University, then to the Centre de Programmation de la Marine, where he worked on a version of the LTR language (Real Time Language). It was there that Gérard Le Lann, who was part of the team that designed the Internet, met him. For him, J-R. Abrial is "one of the greatest French computer scientists!" Not inclined to embrace the "fashions" that more or less regularly agitate scientific communities. In Grenoble, at a university that was to be one of the first in France to develop computer science teaching and research, he and a team of three created the Socrate database management system in less than two years, starting in 1969. He applied the same approach to his subsequent work, specifying before programming. His efficiency was demonstrated by his rapid implementation.

In 1970–71, he gave an original course on "Data and program structure, existential point of view". At the Cargèse symposium in 1974, he published "Data Semantics", an article that would top bibliogaphies for decades. It was in Grenoble that he published the first papers on the Z formal specification method and notation. Z is the ultimate language and the first letter of Zermelo. He

used set theory, its notation and the language of predicates. Z from Grenoble was distributed first by Meyer Baudoin (programming methods) and then by Delobel Adiba (databases and relational systems). Socrate was immediately put to use in Grenoble in the IT sector and professionally via the Eca-Automation company, then in the army, gendarmerie, EDF and SNCF. This was followed by a new version, Clio, developed by Syseca, and many others such as CLIO/SQL, ORCHIS-Base, open to new query languages.

Tony Hoare, who had attended an Abrial course in the Alps, brought him to the Oxford Programming Research Group in 1979 for two years. It was here that he and others developed the well-known version of Z. The implementation of Z at IBM led to the restructuring and rewriting of parts of their CICS (Customer Information Control System) software. This earned PRG its first Queen's award for technology. Abrial was part of the CII-Honeywell-Bull Green team led by Jean Ichbiah, which was selected to define the language that would later be named Ada, in memory of Ada Lovelace, the daughter of Lord Byron, known as "the first programmer in history".

In his 2015 lecture at the Collège de France – which is still available online on the Collège de France website – Jean-Raymond recounts how he worked and what would become the B-Method. In a 1984 article for the Royal Society, "*Programming as a Mathematical Exercise*", he presents his method: "*One consequence of this viewpoint is that the activity of program construction becomes that of proof construction.*" He would remain an "independent consultant" for most of his career. He was also a professor at the CNAM (Conservatoire National des Arts et Métiers) in Paris. He shared his ongoing work with many lecturers and individuals preparing for an engineering degree at CNAM.

Abrial had also developed a tool to assist in development – the B-Tool – an aid for interactive theorem proving. I remember that he addressed the case of the 1986 proportional elections, using the approach presented in his article titled "*To Specify, or How to Master the Abstract*", which included a long quotation from Proust – quite unusual in academic circles. It was a very pedagogical article. The B-Method was in gestation: B because it came "before C" (the B-Toolkit generated ADA and C code).

During the 1980s, the RATP (Paris's public transport operator) called on Abrial for line A of the RER, where the first safety-critical control-command software in France was to be implemented. In his 2015 lecture at the Collège de France, he recalled:

> "Claude Hennebert told me, 'we would like to carry out a technical audit.' The audit lasted three weeks. I had to answer the question: 'Do the means implemented guarantee that the final product matches its original specification?' I said I couldn't answer because I hadn't seen a specification. I was told, 'There was no specification, so give us a course on specification.' "

In the mid-1980s, the RATP decided to embark on a driverless metro – Line 14 – whose critical components were developed using B. With Abrial's support, Alstom decided to develop its own toolset. In 1993, the first version of the B-Toolset was created, comprising a type checker, a proof obligation generator, and a theorem prover, suitable for industrial-scale software. Abrial then proposed that the company Digilog – later Steria, and now Clearsy – industrialise these tools.

Today, Atelier B is freely available via the Clearsy website. Two companies specialising in B were founded in Aix-en-Provence: Clearsy and Systerel.

In 1990, Abrial returned to Oxford, where BP (British Petroleum) developed the B-Toolkit. In B, one works with abstract machines and proves that operations preserve the invariant. For example, the invariant might specify that the generalised intersection of the envelopes of underground trains is always empty. One then needs to specify how these envelopes are computed, given that they evolve due to factors such as speed and passenger load. The process involves moving from a specification of abstract machines to machines that are closer to the software implementation, through what is called refinement. In this stage, abstract variables are linked to more concrete ones within the invariant. Refinement proof is then required. At the end of the process, a program is generated that will remain untouched. Proof is embedded in the construction of the program, rather than being done after the program is written.

Also in 1990, Guy Laffitte used B and the B-Tool at INSEE (the French National Institute of Statistics and Economic Studies) for the general population census. In 1993, Alstom delivered to the metro systems of Calcutta and Cairo, as well as to the SNCF and the RATP, the first train speed control systems incorporating software developed using B. Today, many metro systems around the world use B in their software. B was also used at the Gemplus research centre in Gémenos for smart card development.

The B-Book, published in 1996 [1], was subtitled *Assigning programs to meanings* – a nod to Robert Floyd's 1967 paper *Assigning meanings to programs* [6], which is one of the foundational sources of B. Also in 1996, at the first international B conference in Nantes, Abrial gave a talk on extending B without altering its foundations in order to develop distributed systems. This marked the beginning of Event-B. In 2010, *Modeling in Event-B: System and Software Engineering* [2] was published. In June 2025, the 18th conference on B has taken place in Düsseldorf as the combined ABZ 2025 conference, where the conference started with a tribute to Jean-Raymond and an article written in 2019 with Abrial was presented by his co-author, Dominique Cansell [4].

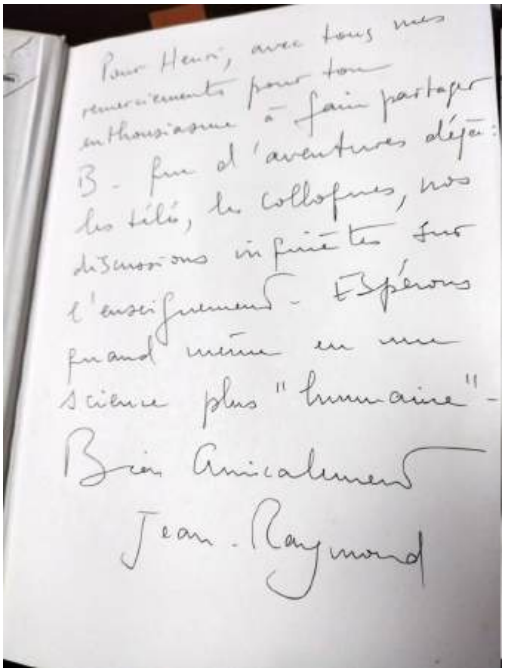
From 2004 to 2009, Abrial held a post at the Swiss Federal Institute of Technology in Zurich (ETH Zurich), where he and a team developed the Rodin platform. In 2006, he was inducted as a member of the Academia Europaea, and in 2008, awarded an honorary doctorate by the University of Sherbrooke. In 2017, in the presence of Chinese President Xi Jinping, he received the International Scientific and Technological Cooperation Award.

At the beginning of his Collège de France lecture, Jean-Raymond stated:

> "There are two types of researchers: the prolific and the monomaniacs.I belong to the latter, as I have always pursued the same kind of investigations – namely, the specification and verified construction of computerised systems."

Jean-Raymond was a researcher and a practitioner whose approach, pedagogy, and publications inspired many academics and practitioners around the world. Z, then B, became the *lingua franca* of many educators. Jean-Raymond Abrial was also a mountaineer and a hiker, whether trekking from Marseille to Cassis via the Calanques or across the Sahara.

Watercolour of Jean-Raymond Abrial, by Henri Habrias (2025).



Dedication by Jean-Raymond Abrial in Henri Habrias's copy of *The B-Book*.

References

[1] Abrial, J.-R. (1996). *The B-Book: Assigning Meanings to Programs*. Cambridge University Press. doi:10.1017/CBO9780511624162

[2] Abrial, J.-R. (2010). *Modeling in Event-B: System and Software Engineering*. Cambridge University Press. doi:10.1017/CBO9781139195881

[3] Bowen, J.P. (1997). B-hold the Future of Software Development. *The Times Higher Education Supplement*, **1267**(30), 14 February. Multimedia computer books.

[4] Cansell, D. and Jean-Raymond Abrial, J.-R. (2025). The Proved Construction of a Protocol with an Example. *ABZ 2025 – 11th International Conference on Rigorous State Based Methods*, Düsseldorf, Germany, 11–13 June. **https://abz-conf.org/site/2025/program/**

[5] Denvir, T., He, J., Jones, C. B., Roscoe, A. W., Stoy, J., Sufrin, B., and Bowen, J. P. (2024). Tony Hoare @ 90. *FACS FACTS*, **2024**(2):5–42, July. BCS. **https://www.bcs.org/media/1wrosrpv/facs-jul24.pdf**

[6] Floyd, R. W. (1967). Assigning meanings to programs. Republished in Colburn, T. R., Fetzer, J. H., Rankin, T. L. (eds.) (1993), *Program Verification. Studies in Cognitive Systems*, vol. 14. Springer. doi:10.1007/978-94-011-1793-7_4

[7] Wikipedia (2025). The B-Book. *Wikipedia: The Free Encyclopedia*. **https://en.wikipedia.org/wiki/The_B-Book**

In memory of Mike Shields (1950–2023): A personal recollection

Tim Denvir

I first met Mike Shields at a two-week winter school in Copenhagen organised by Dines Bjørner and others in 1978. He gave a talk the following year at a conference on the Semantics of Concurrent Computation in Evian, France, titled *Adequate Path Expressions* [1]. I was impressed by his vibrant presentation style and by the fact that he started by saying that what he was really interested in was the Mathematics of the topic.

In the 1980s FACS organised Christmas Workshops at BCS. In 1988 Mike gave a talk at this workshop titled *Automata-Theoretic Models of Parallelism* [2]. After Mike's talk, all the other members of the FACS committee were wanting to get home, so I was delegated to take Mike out to an evening meal. I seem to remember that we went to an Italian restaurant.

In 1983 the Software Research Group at STL organised a Workshop on the Analysis of Concurrent Systems held at Clare College Cambridge. This consisted of four tutorial papers (the presenters included Robin Milner and J-R Abrial), and ten concurrency "problems", to which we invited solutions from well-known academics. Mike provided a solution to the fourth problem, which asked for a formal specification of a railway system where no section of track could have more than one train on it at a time. Other solutions were provided by Peter Lauer, B. Moszkowski, and Bill Roscoe. The proceedings of this workshop are recorded in *LNCS 207* [3] with Mike's solution at page 389. Mike was at the University of Kent, Canterbury, at the time.

As a result of these activities I and other members of the STL Software Research Group got to know many academics in Theoretical Computer Science. Robin Milner got in touch with me and explained about a new EPSRC funding initiative, designed to enhance liaison between academia and industry, to the benefit of both. Robin could engage an academic to come and work in our group at STL, we would reimburse his department for the visiting academic's salary, and his department would get some further EPSRC funding. I could see the advantage of this for us. The visiting academic would not be an employee of STL, so we would get them at a cost of just their salary, whereas if they were a STL employee, the STL administration would charge my department a loaded rate at three times their salary. It seemed beneficial all round, if slightly artificial in retrospect. I suggested Mike Smyth as a possibility, at which Robin said, "Well, if you can get Mike Smyth, that would be amazing!". He obviously had a great regard for Mike Smyth, but when I suggested it to him, Mike Smyth was not really interested, so when I reported back to Robin Milner, he suggested Mike Shields, who was then an RA in the Newcastle CS concurrency group under Peter Lauer, at which we at STL were most enthusiastic. The consequence was that Mike Shields worked at STL in our research group for about a year. One of the first things I did was to send Mike on a week's course on "Telecommunications Systems Planning", which we had all been on and which was run by a highly charismatic Englishman way past normal retirement age, who must have been very well paid because he had flats in central London and New York. I did wonder in advance what Mike would make of this course, but he declared that he found it "very stimulating".

There was a language used by the telecommunications industry called SDL – Systems Design

Language. This language was not itself all that well designed, but Mike set about writing a formal definition of it. To do so, Mike had to write typical mathematical text and the only way to do that in those days was to use an IBM Golf Ball typewriter. The "golf balls" were interchangeable and each would have a different typeface or set of mathematical and other symbols. It was before the era of advanced word-processors and way before LaTeX! So Mike monopolised my secretary's typewriter for a while, but I firmly believe that his efforts had a strong influence on improving the telecoms systems language, SDL.

After another talk that Mike gave to FACS in London in the mid 1990s, again we had dinner together. He had recently married Myra. They met each other through their shared enthusiasm for bell ringing. Myra lived in Malta and Mike moved to join her. Unfortunately and tragically Myra died after they had been married only 18 months or so. But before that, Mike wrote *Semantics of Parallelism* [4], which he dedicated to Myra and her two children, Ben and Rebecca. This was eventually published in 1997, and is still, in my opinion, possibly the best comparative study of all the different ways of formally modelling concurrency. A notable feature of this book, which illustrates Mike's broad cultural knowledge and wit, are the introductory quotes to each chapter, which range over James Joyce's Finnegan's Wake, James Thurber, Bob Dylan, J. S. Borges, G. B. Shaw, Jonathan Swift, Charles Darwin, not to mention his fictitious character, F. X. Reid, and many others. Mike had published one much earlier book, *An Introduction to Automata Theory*, in 1987 [5]. Mike's health later deteriorated and he was looked after well by his carer, Lee Dong-Mei. They married in, I think, 2019, but extensive efforts to contact Mike's second wife have failed.

Mike was in the process of writing three books when he died, on Logic, the Reals, and Whole Numbers. The latter is in the most complete state and I have the whole text of it. I hope at some time in the near future to get it published for him, but it needs a fair bit of editing. I have had useful editorial comments from FACS members, notably John Cooke.

I am grateful to Bernie Cohen for some input for this memorial. Mike will be missed by many colleagues, for his unique wit and for his piercing intellect.

References

[1] Shields, M.W. (1979), *Adequate path expressions*. In: Kahn, G. (eds) Semantics of Concurrent Computation. Lecture Notes in Computer Science, vol 70. Springer, Berlin, Heidelberg. doi:10.1007/BFb0022473

[2] Mike Shields (1988), *Automata-theoretic models of parallelism*, BCS-FACS Christmas Lectures, 1988, **automata-theoretic-models-of-parallelism-mike-shields.pdf**

[3] *The Analysis of Concurrent Systems*, Cambridge, September 1983, Ed. B.T. Denvir, W.T. Harwood, M.I. Jackson, M.J. Wray. Lecture Notes in Computer Science, vol 207. Springer, Berlin, Heidelberg, 1985, doi:10.1007/3-540-16047-7

[4] Shields, M.W. (1997), *Semantics of Parallelism*, Springer-Verlag, ISBN 3-540-76059-8.

[5] Shields, M.W. (1987), *An Introduction to Automata Theory*. Blackwell Scientific Publications. ISBN 978-0632017560.

The ProCoS project and Duration Calculus: A personal memoir

Zhiming Liu
School of Computer and Information Science
Southwest University, Chongqing, China

1 Background

I was involved with the European collaborative ESPRIT ProCoS Project [1] on "Provably Correct Systems" that ran during the early 1990s for only a few months. However, it was just after the viva of my PhD [15] and thus it gave me a powerful lift in my academic career development. Here I record some recollections of some events and my interactions with some friends, specially Dines Bjørner, Zhou Chaochen, Anders P. Ravn, and E.V. Sørensen.

2 A Viva, a Gas Burner, a Guest Scientist, and a Meeting

Between 1988 and 1991, I juggled life as a part-time PhD student and a full-time research assistant at the University of Warwick, funded by SERC – the Science and Engineering Research Council (now the more streamlined EPSRC). Under the supervision of Professor Mathai Joseph, who was well known in the ProCoS circle, I worked on *formal techniques for fault-tolerance*, a subject that proved as useful for surviving research life as for designing robust and resilient systems.

Driven by the ticking clock of project funding, I worked hard – or perhaps just frantically – to complete my PhD within three years. In a plot twist worthy of academia, I finished my thesis too quickly, only to be told by the postgraduate office that part-time PhDs were expected to serve a minimum of five years. Efficiency, it seems, was not part of the curriculum.

Fortunately, the first problem was resolved after Mathai wrote a convincing letter to the postgraduate office, and I managed to submit my thesis in July 1991. But just as one storm passed, another cloud appeared. I hadn't bothered applying for postdoctoral jobs, reassured by Mathai's confidence – he had two or three project proposals in the pipeline and was already worrying about where to find enough researchers to fill them. As luck (or bad timing) would have it, none of the projects got funded. I suddenly found myself not looking for a job, but for five or six months' worth of survival funding.

I decided to appeal to Zhou Chaochen at the Technical University of Denmark (DTU), wondering if there might be a few months of funding to be found there. Zhou was my MSc supervisor and a member of the ProCoS project's funding aristocracy – surely he could rescue a stranded academic for a modest fee! Zhou replied to my email very quickly and said Professor Anders Ravn[1] was

[1] Anders held the title of 'Docent' – a term that puzzled me at first, until I discovered it was the Scandinavian equivalent of a 'Reader' in the UK. Another academic rank learned, and this time, not from China.

happy to offer me a position of Guest Scientist, with quite a generous salary compared with that of an RA in the UK, which was funded by ProCoS.

At the time, I was a Chinese national and entitled to a tax exemption in Denmark. I remember one day, Dines glimpsed my payslip on my desk and remarked, "This isn't fair – you're earning more than me after tax!" Danish taxes were indeed high. After joining DTU, I learned there was even a church tax and a special tax to support employment. In fact, Dines's secretary was funded by that tax. So, in a way, I suppose I was partially responsible for her salary too – just not through my payslip.

My PhD viva took place on Friday the 13th of September, 1991 – an ominous date that lived up to its reputation. The examination was a record marathon, lasting over six hours in total, with a break for lunch in between. At the very end, one of the two examiners read from their report: "…, an ambitious piece of work has been done, …" – words that felt both like a verdict and a relief. Possibly noticing the look of confusion and mild panic on my face – as "ambitious" can carry mixed connotations in Chinese culture – Mathai, who had been courteously invited to attend (since both examiners had to be external due to my research staff status), leaned over and reassured me: "Don't worry – 'ambitious' is a compliment here." I decided to believe him.

The following Friday evening, Mathai and his wife, Anita, invited my wife, Hong, and me to their home for dinner. As I was chatting with Anita about my PhD viva, she turned to Mathai and exclaimed, "How could you schedule his viva on a Black Friday?" She knew that I had grown up in a poor village during the Cultural Revolution, a time when normal schooling was severely disrupted – we had few proper lessons, and even no books to read. I was among the first cohort of students admitted to university in 1978, just as Deng Xiaoping's reforms were beginning to reopen the gates of education. Turning back to me, Anita asked, "Do you know that Friday the 13th is considered an unlucky day?" I smiled and replied, "I've heard something about it – but to be honest, I'm still not quite sure why." In truth, before learning a little too much about Western culture, many Chinese people actually considered 13 a lucky number – and Friday the 13th just another perfectly ordinary day.

On 10 October 1991 – exactly 27 days after my viva – Hong and I arrived in Lyngby (a name I still struggle to pronounce correctly), Denmark, the second foreign country I had ever visited. I had come to join the ProCoS site team at DTU. The next day, I visited the department to see my new office and to meet Zhou and my other colleagues. As it happened, they were just preparing to leave for the ProCoS Symposium on the island of Fyn (Funen in English), about 30 kilometres from the birthplace of Hans Christian Andersen – a wonderfully Danish way to be introduced to the project.

I went to see Dines in his office – noticeably larger than the others, and rightly so, given the demands of his energetic work and the vast number of books needed to support it. We already knew each other well, largely through my connections to two of his closest friends: Zhou Chaochen, my MSc supervisor, and Mathai Joseph, my PhD advisor. It had been some time since we last met, and the reunion was warm and cheerful – a welcome moment of familiarity in a new environment.

After the warm greetings, Dines suddenly paused and said, with characteristic directness, "Oh, we've made a mistake – a serious mistake." Before I could begin to guess what kind of mistake, he had already summoned the three key members of the ProCoS site team – Anders Ravn, Hans

Rischel, and Michael R. Hansen – to his office. As they entered, he declared: "How could we make such a mistake and forget to arrange for Liu to attend the ProCoS meeting? Liu must go – and Hong should go with him. Please find the funds and book the train tickets immediately."

Anders replied without hesitation, "Yes, of course. Not a problem." The three left to carry out the mission, and Dines, satisfied, turned to me and said, "This is how things should be done – problems solved in real time."

On Saturday the 12th, Hong and I travelled with Anders by train from Copenhagen – a journey that took an unexpected turn when the entire train boarded a ship. It was the first time in my life that I had seen such a thing, and I wasn't entirely sure whether to admire the engineering or worry about buoyancy.

When we arrived at the hotel, we were surprised – and more than a little amused – to be given a large suite, complete with a grand working desk. As it turned out, our room was of the same calibre as those reserved for Professor Sir Tony Hoare of Oxford and Professor Dr Hans Langmaack of Kiel University (whom I had never met before). Meanwhile, other participants – including even Zhou – were assigned modest single rooms. Hong and I couldn't help but feel rather spoiled. So, while I tried to look appropriately academic, Hong, lent a hand to Dines's secretary with the conference logistics – partly to feel more at ease amidst such luxurious surroundings, and partly out of genuine gratitude for the warm and generous hospitality we had received.

It was at that ProCoS meeting that I first encountered several people who would later become lifelong friends, collaborators, colleagues – and even mentors. Among them was Jonathan Bowen, whom I initially spotted as the tallest person in the room, enthusiastically playing pool after dinner with the focus of a logician and the joy of someone who had momentarily escaped formal methods. Then there was He Jifeng, whom Zhou had praised to me even before I came out of China to the UK in 1988, especially for his remarkable research abilities. He wasn't particularly talkative, and I was admittedly too nervous to approach him at first – but Hong had no such reservations and seemed to engage him in several cheerful conversations with ease. Finally, I met Ernst-Rüdiger Olderog, whom I would come to know well through Anders Ravn – the two were long-standing close friends, and their camaraderie was as evident as their shared passion for specification and verification of real-time systems.

Among these friends, He Jifeng joined UNU-IIST as a Senior Research Fellow in 1998 and remained until 2005. I first spent an eight-month sabbatical at UNU-IIST in 2001, and then formally joined the institute in July 2002, staying until October 2013. This meant that Jifeng and I had several years of close collaboration, particularly between 2001 and 2005, during which we developed the relational semantics and refinement calculus for object-oriented programs [11], as well as the refinement calculus for object-oriented and component systems known as rCOS [10, 20].

When my UNU-IIST contract was unexpectedly terminated midway in 2013 due to restructuring by the UNU headquarters in Tokyo [2]. Jonathan Bowen kindly helped me secure a position at Birmingham City University (BCU). From that point on, he became a close collaborator and friend. Our collaboration continued even after we both left BCU in 2015, especially through co-organising the International School on Engineering Trustworthy Software Systems (SETSS), which we initiated

[2] UNU-IIST was later renamed the United Nations University Institute on Computing and Society (UNU-CS), and with that transformation, theoretical computer science and formal methods effectively disappeared from its agenda.

in 2014, e.g., [3].

As for Anders – I have devoted the final section of this memoir as a tribute to him.

At the time, Dines and Zhou were deeply involved in preparing for the launch of a brand-new institute: the United Nations University's International Institute for Software Technology (UNU-IIST) in Macao, which would officially open in 1992. As a result, Dines had to step down from his role as ProCoS project coordinator – a position that was then passed on to Tony Hoare.

The transition was formalised in true Dines style: at the workshop banquet. Both Dines and Tony gave speeches, full of warmth, humour[3], and just the right amount of academic sentimentality. I was surprised by Tony's speech – warm, emotional, even humorous – a marked contrast to the impression of an English gentleman he had given when Zhou's four former students, myself included, visited his home for lunch with his family in 1989.

Dines and Tony spoke of their collaboration, mutual support, and the shared effort of the wider ProCoS team – a community as rigorous in formal methods as it was informal over dinner. The mood was celebratory, but with a touch of poignancy – the kind you get when passing the baton between two masters of the field, under the gentle influence of good wine and even better camaraderie.

It was, for Hong and me, the first truly formal, multi-course Western banquet we had ever attended – and it left a lasting impression. From that evening on, we began to appreciate wine – not just for its taste, but for the elegance of its rituals. Hong would often recall with a smile, "That row of wine glasses in front of me – tall, short, fat, and thin – looked absolutely beautiful." Perhaps that's also why we stayed in academia: for the structured beauty of logical thinking, formal reasoning, and occasionally, a well-set table.

Looking back, I'm reminded not only of how closely knit the ProCoS team was, but also of Dines's extraordinary kindness to both me and Hong. Kari, one of the kindest ladies I've ever met, looked after Hong with genuine warmth. Hong thoroughly enjoyed doing patchwork with her, sharing tea, and exploring Copenhagen together.

On the lighter side – and still amusing to my very Chinese sensibilities – I remember Dines's party invitations, which read: "You are cordially invited to the party on . . . at . . . ; boyfriends, girlfriends, mistresses, and dogs are welcome."

It was in such moments that I felt a deep, almost fatherly warmth from Dines and Kari – a quiet, enduring generosity that made our time in Denmark all the more memorable and meaningful.

3 DC and Probabilistic DC

Duration Calculus [25], or simply DC, was one of the most innovative and influential outcomes of the ProCoS project. Zhou Chaochen and Anders Ravn encouraged me to explore a probabilistic extension of DC, in close collaboration with a senior professor at DTU, E.V. Sørensen. Known affectionately and efficiently as "EV" by everyone at DTU – a convenience I appreciated, as I could not manage to pronounce his full first name and never learned his middle one – he was both

[3] I remember that Tony said something like "Dines, our father, we are your children. You are leaving us . . . "

warm-hearted and deeply serious about research.

When Zhou suggested that I work with EV on probabilistic DC, he shared the story of how DC itself was originally inspired by a safety example that EV had presented at a ProCoS meeting. EV was an expert in the safety analysis of engineering systems, and he had a remarkable ability to illustrate abstract ideas through practical scenarios. One such example involved a gas burner system. Gas leaks, EV explained, were unavoidable – the result of various minor faults – but the system could still be considered safe as long as the gas concentration in the room never exceeded a critical threshold.

With the help of real analysis and a dash of calculus, EV showed that this safety requirement could be captured by a simple constraint:

Req: If the system is observed over any interval of at least one minute, the proportion of time spent in a leak state must be no more than one-twentieth of the total elapsed time.

To meet this requirement, EV proposed two corresponding design strategies:

Des-1: Any leak must be detected and stopped within one second.

Des-2: Leaks must be separated by at least 30 seconds – that is, a leak should not recur too soon after the previous one is resolved.[4]

It was a perfect illustration of how continuous-time reasoning could be turned into concrete, verifiable constraints – and how practical engineering concerns could give rise to elegant formal models.

Zhou recalled that it was at the end of EV's presentation that he – quite memorably – proposed a challenge: to develop a formal logic capable of specifying such safety requirements and designs, and of reasoning about their correctness. As I write this, I'm quietly amused to discover that the presentation wasn't just a vivid personal memory – it was, in fact, properly documented and cited as [22] in the DC book [24]. It's always reassuring when the historical record aligns with one's recollections – especially in academia, where memories are not always formally verifiable.

What impressed me even more, however, was Zhou's genuine excitement and gratitude for the opportunity that the ProCoS project had provided: a platform for rich, interdisciplinary collaboration. In many ways, DC stands as a masterpiece of his academic career – a shining example of how cross-disciplinary thinking and real-world problems can inspire deep theoretical innovation.

Zhou was a well-trained mathematician from Peking University – long regarded as China's premier Chinese institution for mathematics – and a logician who completed his master's degree under one of the most distinguished mathematical logicians in Chinese history, Professor Hu Shihua. When EV posed his challenge, Zhou took it up with the greatest enthusiasm I could imagine – the kind of enthusiasm that only a logician facing a beautifully structured problem could fully appreciate.

In just about a year, Zhou – working closely with Tony and Anders – developed the first version of Duration Calculus (DC). The core idea was elegant: to reason about systems using time intervals

[4] Strictly speaking, **Des-2** is more of an assumption than a design decision – and **Des-1** wouldn't be feasible without this assumption of stability.

and the durations of states. A state S was treated as a Boolean function over the time domain; its duration within a time interval was defined as the integral $\int S$ over that interval, while the length ℓ of the interval itself was simply the integral $\int 1$, that is, the duration of the *almost always-true state*.

The logic was modal, but minimal: it featured just one modality, ";", used for *chopping* a time interval into two adjacent subintervals. The proof system followed a Hilbert style, with just five axioms and one induction rule – compact, elegant, and rigorous.

Using DC, the safety requirement **Req** could be specified as:

$$\ell \geq 60 \Rightarrow 20 \int Leak \leq \ell$$

And the two design conditions could be formalised as:

$$\begin{array}{ll} \textbf{Des-1:} & \Box(\lceil Leak \rceil \Rightarrow \ell \leq 1) \\ \textbf{Des-2:} & \Box(\lceil Leak \rceil ; \lceil \neg Leak \rceil \Rightarrow \ell \geq 30) \end{array}$$

where $\lceil S \rceil$ abbreviates $(\int S = \int 1) \wedge (\ell \geq 0)$, i.e., the state S is *almost always true* throughout the interval.

It can then be formally proved that "the two design decisions imply the requirement" is a theorem of DC – a textbook example of how logic and calculus can combine to reason rigorously about real-world constraints. As Zhou once said with a smile, "finally, a use for integrals that isn't just in exams."

I still remember Zhou telling me – back when I was his MSc student – that he had always been proud of his performance in continuous mathematics at university, and much less so in algebra. Duration Calculus, it seems, gave him the perfect excuse to put that strength to brilliant use.

I must admit, I've often felt a bit embarrassed to call myself his student, having never come close to matching his clarity in exposition – the simplicity of language, the precision of concepts, the elegance and substance packed into his notation. And all this, despite the fact that he didn't command English as his native language.[5]

The work was submitted to the journal *Information Processing Letters* (IPL) before I joined DTU. One aspect of the academic culture there that I will never forget was the atmosphere of openness and collaboration. Colleagues were friendly, accessible, and always ready to exchange ideas. Office doors were (literally) open, and it was common for people to pop in for a quick chat, a technical discussion – or even a shared smoke break. (Anders often smoked his pipe, and I too was, at the time, a rather dedicated smoker.)

I remember one particular day when a rather animated discussion was taking place in Anders' office, involving Zhou, Anders, and possibly a few others. Voices were raised – not in anger, but in that unmistakable rhythm of urgent academic debate. After Zhou returned to his office, directly opposite mine, I went over to ask what had happened. He responded with the calm of someone already several steps into solving the problem: a serious flaw had just been discovered in the IPL

[5] Zhou studied Russian, not English, during his university years – which only makes his written English all the more impressive.

paper – the very one about to go to press. The issue was that the *induction rule* had been presented as a **theorem**, but further work on another paper had revealed it was, in fact, not valid in general. The rule in question was as follows:

Induction Rule 1 *Let X be a formula variable occurring in the formula $R(X)$, and let P be a state.*

1. *If $R(\lceil\ \rceil)$ holds, and both $R(X \vee (X;\lceil P\rceil) \vee (X;\lceil \neg P\rceil))$ can be proved from $R(X)$, then $R(true)$ holds.*
2. *If $R(\lceil\ \rceil)$ holds, and both $R(X \vee (\lceil P\rceil;X) \vee (\lceil \neg P\rceil;X))$ can be proved from $R(X)$, then $R(true)$ holds.*

To their credit, they had already worked out a possible fix – but no one seemed especially pleased with it. I went back to my desk to think it over. With a hint of hesitation (and feeling slightly naive), I returned to Zhou and suggested: "Well, if the rule isn't valid as a theorem, but your model needs it, why not just declare it as an axiom and call it the induction rule?"

Zhou paused, considered it, and then – with a characteristic blend of pragmatism and clarity – said, "Oh… why not? This is the solution then." That suggestion earned me an acknowledgement in the final published version of the paper – a small but memorable contribution to the foundations of Duration Calculus. It also left a good first impression on Anders, who was, after all, my boss at DTU – and not a bad person to impress early on.

I tremendously enjoyed the many discussions with Zhou about the potential impact and future challenges in the development of Duration Calculus. These conversations took place everywhere – in our offices, over lunch and dinner, and during after-dinner walks. At the time, Zhou was living alone in a university guest house, having left his family in Oxford. However, he maintained a firm rule: no work discussions after 8 p.m., so his mind could settle peacefully before bedtime – a habit I both respected and occasionally found hard to obey.

One issue we were particularly concerned with was how to relate DC to other temporal logics, such as LTL and TLA. These logics differ philosophically in how they view system behaviour. DC adopts a global perspective, observing changes in state at a given time point by considering its neighbourhood. In contrast, LTL and TLA focus on points of state change to describe global behaviour. This distinction is deeply connected to their applications: DC is well-suited to the specification and verification of low-level system models, while logics like LTL and TLA are better aligned with expressing high-level requirements, such as the safety condition **Req** in the gas burner example.

Years later, I proposed a two-dimensional model combining DC and LTL [17], showing that starting from a high-level specification like **Req** allows for a broader range of implementations than just **Des-1** and **Des-2**. In collaboration with Yifeng Chen – then a colleague at the University of Leicester – we further investigated the integration of DC with other modal logics from a semantic perspective [4, 5].

My main contribution to ProCoS was the development of a probabilistic extension to DC, "Probabilistic Duration Calculus", known as PDC, aimed at analysing the dependability of fault-prone

real-time systems – exemplified by a gas burner with an imperfect flame failure detector. This was proposed by Zhou. The background and motivation were obvious, and I accepted the suggestion without hesitation.

After all, my PhD focused on formal techniques for fault-tolerant systems. In fact, the version of my thesis originally submitted to the examiners began – just after the introduction – with a chapter on probabilistic characterisation of program execution under random faults. The outlined model was a probabilistic extension to *Hoare Logic* and introduced the notion of *probabilistic refinement*, without going into a full investigation.

One examiner, however, recommended removing the chapter from the final version [15], remarking that "a complete probabilistic extension to Hoare Logic can't be this simple." He was, of course, right – but that wasn't the point. The chapter was only intended to motivate my main work on an unquantified model of fault tolerance, and to propose a new research direction. As it turned out, that direction proved fruitful [21].

Back then, I was a modest and obedient young man. Regrettably, I deleted the chapter – and didn't even keep a copy. In hindsight, I rather wish I had kept the chapter in my thesis as probabilistic verification soon became a popular research area, especially now that probabilistic reasoning has become such a central topic in the age of AI.

PDC uses *probabilistic automata* to represent fault-prone implementations and assumes that requirements are expressed as DC formulas. The calculus includes seven axioms and one induction rule for computing the *satisfaction probability* $\mu(D)$ of a DC formula D with respect to a given automaton. Styled after DC, the axioms and rules offer a natural extension to probabilistic reasoning.

PDC was first presented at the two-day International Workshop on Responsive Systems, held at the impressive KDD Research and Development Laboratories in Kamifukuoka, Saitama, Japan, on 1–2 October 1992. The formal publication appeared in an edited volume [18].

The paper in [18] earned EV and me the travel to Japan in October, only a few months after UNU-IIST started. Dines and Zhou both suggested that we to fly (EV from Copenhagen and I from London) to Tokyo via Hong Kong and combine our journeys with a visit to UNU-IIST in Macao (its sovereignty was yet to be handed over to China). We were both very happy to accept, and I, in particular, felt overwhelmed by such an unexpected favour.

The work was later revised and extended to introduce a method for calculating satisfaction probabilities using classical probability matrices. Its usefulness was demonstrated through a dependability analysis of a communication protocol operating over an unreliable medium, as proposed in [8, 9]. The results were published in the inaugural volume of the journal *High Integrity Systems* [19].

We – Zhou in particular – were quite fond of this practical approach, as it pointed toward the automation of probabilistic verification, a topic just beginning to gain traction. Unfortunately, the journal ceased publication after its first volume – for reasons that remain a mystery – and the work never received the attention it might have deserved. PDC was later further developed by introducing parallel composition and probabilistic refinement [16]. PDC is for a discrete time domain, but it was extended to a PDC for continuous time by Dang Van Hung and Zhou [12],

when they were working at UNU-IIST.

4 UNU-IIST's Contribution to Formal Methods and DC

UNU-IIST, founded by two close friends – Dines as the Founding Director and Zhou as the Principal Research Fellow – certainly deserves a dedicated memoir for its significant contributions to research and education in formal methods, and to Duration Calculus in particular (and, personally, to my own development). The mission of the institute, as formally stated in its charter, was to *help developing countries*[6] *strengthen their software development capacity through training and collaboration*.

My first visit to UNU-IIST – and to Macao[7] and Hong Kong – was in October 1992, en route to Japan for the Responsive Systems workshop. I flew from London Heathrow to Hong Kong and arrived in the late morning, then took a taxi to the Hong Kong–Macau Ferry Terminal to catch a TurboJet. The taxi driver, apparently mistaking me for a naive "Mainland uncle" unfamiliar with English or exchange rates, tried to charge me in US dollars – at the same numerical rate as the Hong Kong dollars shown on the meter. When I firmly told him he was cheating and threatened to call the police, he quickly changed his tone, launching into a dramatic tale of hardship: he was just a part-time driver who had waited all day for a single fare. I decided not to argue further, paid the correct fare in §HK, and chalked it up as part of the international research experience.

The TurboJet took just over an hour. At the Macau Ferry Terminal, I was greeted by Dines – tall and unmistakable in his signature hat and suit – waving cheerfully from the crowd. He drove me to the hotel himself. Though I knew Macao covered just 9 square kilometres, it felt like a much larger city as we wound our way through its labyrinth of one-way streets. I shared my encounter with the cheeky Hong Kong taxi driver. Dines laughed and said a driver had once tried the same trick on him – to which he replied, "So, shall I pay you half price instead?" Classic Dines – sharp, witty, and impossible to fluster.

EV arrived a couple of hours later. We were both asked to give seminars the next afternoon – the very first seminars to be held at UNU-IIST since its founding. Zhou took the event very seriously and even had a notice published in the *Macau Daily* (in Traditional Chinese, of course). Formal methods – and theoretical computer science in general – were still largely unknown in Macao, and Zhou was anxious that our talks be accessible to the local audience. He was especially concerned about me – a freshly minted PhD with little experience in giving seminars and less in oral English. After dinner, Zhou came to my hotel and helped me rewrite many of my foils – the overhead projector transparencies we used in those pre-PowerPoint days.

UNU-IIST was then housed in a bank building, and the seminars took place there. As we ascended in the lift, Dines informed us with a conspiratorial grin that nightclubs occupied some floors above and below – which, he added, was why UNU-IIST fellows weren't allowed to work late in the office.

The seminar room was small but packed. After opening remarks by Dines and an introduction

[6] Now commonly referred to as the "Global South".
[7] Then commonly spelt "Macau" before the handover.

from Zhou, EV gave the first talk, followed by mine after a short break. The audience listened attentively – and, in line with local norms, refrained from asking questions. But the break and post-seminar discussions were lively and engaging. I still remember meeting You Xiaohong, a young man working at the Bank of China in Macao, who later became Zhou's MSc student – and thus, academically speaking, my younger brother.

We stayed one more day in Macao before continuing to Tokyo via Hong Kong.

UNU-IIST was always a small institute – the number of academic staff never exceeded ten, supported by six general staff. All academic members specialised in formal methods. The institute was informally split into two groups: the *upstairs group*, initially led by Zhou and later by He Jifeng after Zhou became Director; and the *downstairs group*[8], led for most of its time by Chris George. The upstairs group focused on theory – often referred to as the DC group – while the downstairs group worked on practical applications, especially using the RAISE method, hence the nickname RAISE group. UNU-IIST fellows – officially termed as such, though translating the word into Chinese always proved tricky – were selected from developing countries. They typically spent eight months at the institute, with some extending to a second term. The academic staff mentored these fellows through collaborative research, formal courses (both long and short), and organised training schools across the Global South. I taught at such schools in Brazil, China, Iran, Kazakhstan, Nigeria, Tunisia, South Africa, and Vietnam – both before and after I joined UNU-IIST in 2002. My colleagues reached even more countries, including Afghanistan, Bangladesh, Cuba, India, Iraq, Mongolia, Nepal, North Korea, Sri Lanka, and most of Africa, South America, and East Asia. It was the UNU-IIST staff who first ignited the spark of formal methods and theoretical computer science in many parts of the Global South.

Duration Calculus was a central theme in UNU-IIST's research and training – especially in its first decade, until Zhou's return to the Software Institute of the Chinese Academy of Sciences in late 2002. From 1992, UNU-IIST gradually became the epicentre of DC research, laying its theoretical foundation, including completeness and decidability results, and developing algorithms for linear *duration invariants*. It introduced new semantic models such as *finite divergence* and *superdense computation*, as well as new modalities (like *neighbourhoods*) for unbounded fairness and liveness, and higher-order operators to enhance expressiveness. These results are collected in the DC book [24], and include contributions from PhD theses by Xiaoshan Li [14] and Naijun Zhan [23], both former UNU-IIST fellows supervised by Zhou.

DC's applications were also a focus – including defining semantics for specification languages and verifying case studies. Many publications were collaborative efforts between the DC and RAISE groups. After He Jifeng – another key ProCoS figure – joined UNU-IIST, substantial work was done integrating DC with CSP, Occam-like languages, and real-time semantics for Verilog and RAISE. These are cited in the Introduction chapter of the DC book.

Notably, many former UNU-IIST fellows became leading researchers and academic leaders in formal methods and software engineering, supervising numerous PhDs themselves. Though its global impact was rarely highlighted in public reports [9], UNU-IIST was often referred to as the

[8] These names came into use after the institute moved to a historic two-story building.

[9] Except for the a report that Jonathan Bowen wrote for the *Times Higher Education Supplement* after his visit [2].

Huangpu Military Academy[10] of software engineering.

The institute hosted many prominent international visitors, both for short and extended stays. These included Tony Hoare, Bill Roscoe, Willem-Paul de Roever, Ernst-Rˇdiger Olderog, Anders Ravn, Michael R. Hansen, Jonathan Bowen, Jim Woodcock, Paritosh Pandya, Augusto Sampaio, and Wang Yi – many of them former ProCoS collaborators. They gave lectures and collaborated with fellows. Of particular note is Paritosh Pandya, one of Mathai's earliest PhDs in India. Paritosh was also a former ProCoS researcher at Oxford's Programming Research Group (PRG), who has since carried the DC torch long after ProCoS and far beyond Europe and China.

Among the many events organised by UNU-IIST, one stands out: the 2007 Festschrift Conference in Macao, celebrating the 70th birthdays of Dines and Zhou [13]. Friends and colleagues from around the world – including many ProCoS alumni – gathered to honour their achievements.

5 Anders Ravn – a Friend and Mentor

I cannot complete this memoir without paying tribute to Anders – one of the best friends and mentors I've had, in both life and career.

Time at DTU: I felt instantly at ease when I first met Anders, after joining DTU in October 1991. He spoke softly and thoughtfully, his large smiling eyes behind thick lenses, his head slightly tilted and lowered in characteristic modesty. When my wife Hong fell ill shortly after our arrival in Lyngby, Anders walked us to the family doctor and ensured we had the right medicine from the pharmacy before seeing us safely back to the house we were renting, which was one of beautiful and straw-roofed houses in Lyngby.

We worked closely – and happily – with Zhou and EV on PDC. Anders hoped I could stay on at DTU, but he understood when I said I needed to return to the UK to fulfil my promise to Mathai and to simplify life for Hong and me, especially with the language.

Time at Warwick: Our friendship only deepened after I returned to Warwick in early 1992. We continued collaborating on PDC, and Anders carefully and diligently read and commented on all my fault-tolerance papers and research grant proposals. He invited me for anuber of short visits to DTU and a two-month stay, between 1992 and 1994.

In October 1992, he invited me to the *First Workshop on Hybrid Systems*, the proceedings of which appeared in 1993 [7]. That event later grew into the flagship *ACM International Conference on Hybrid Systems: Computation and Control*. I remember sitting with Anders, Zhou, and Professor Li Wei (later President of Beihang University) at the workshop banquet in Copenhagen City Hall. They were all aware of my uncertain career prospects at the time – and strongly urged me to stay abroad and seek a permanent position rather than return to China.

It was the era of the global recession, and jobs were scarce. For a five-year contract post, there were 99 applicants from 12 countries, including the US and Japan. Anders helped refine my CV and cover letters, offered detailed suggestions for presentations, and even prepped me for interviews –

[10] The Huangpu Military Academy, founded by Sun Yat-sen in 1924, trained revolutionary military leaders who shaped modern China.

including anticipating difficult questions.

Time at Leicester: Eventually, I landed a two-year lectureship in the Department of Mathematics and Computer Science at the University of Leicester, starting October 1st. The department kindly exempted me from teaching duties that term, so Anders invited me to spend it at DTU. There, he taught me how to lecture – not just the content, but also how to handle classes. Without that support from him, my position would not be made permanent just after I completed the first year.

During my eight years at Leicester, our collaboration continued with frequent visits, though Anders invited me more often than I invited him. In 1995, he visited Leicester to give a seminar. My second son, Edward, had just been born, and my parents were visiting from China to help. That evening, Anders came to our modest home – the first house I ever owned – to meet my parents and see the children. His gentleness, kindness, and warm smile left a lasting impression on my parents, who had spent their lives in a small Chinese village.

When Edward was born, I was quite anxious about raising two young children in a foreign country without close family nearby. I wrote to Anders about my worries. He replied with characteristic humour and reassurance: "I assure you, bringing up two will be no more than twice the trouble of bringing up one – especially when they're of the same type (both boys)." Anders and Annemette had two sons, Neils and Mads, of their own, so I knew he spoke from experience.

In the summer of 1999, I received an email from him: "Would you like to come to Lyngby for a couple of weeks?" I asked if he had a plan for collaboration. He replied, "No plan at all. How about just for fun?" Naturally, I agreed.

I stayed in a large house where the landlady kindly lent me a bicycle. I was working on a denotational semantics for object-oriented programs, and one night I struggled with a proof – returning to the office twice before finally completing it at dawn. That morning, Anders appeared and said, "Annemette and I would like to invite you for dinner." I hesitated, admitting I hadn't slept, washed, or brushed my teeth. He smiled and said, "Perfect! We'll offer you a hot shower, a fresh toothbrush, and good wine." It was a warm, cheerful evening, ending with a ride home in their little red Fiat, driven by Annemette – since Anders readily admitted he was a terrible driver.

Time at UNU-IIST: I officially joined UNU-IIST on 2 July 2002 – though amusingly, the travel day from the UK to Macao was counted as part of my working time. Within six months, we established a joint PhD programme with the University of Pisa, in addition to our fellow programme. I soon had a small group of two PhD students and several fellows. In May 2004, Macao Science and Technology Development Fund (FDCT) began supporting research projects. I was fortunate in a few applications and soon had my own funded projects, allowing me to support postdocs and invite collaborators for student supervision and joint research.

Anders visited Macao almost annually. Though the flights were long, he steadfastly refused to fly business class, saying, "That money could fund a couple of fellows to go to conferences." Even when the University of Macau offered him a business class ticket as a PhD examiner, he declined it.

Several members of my group collaborated with Anders, who was deeply respected and much loved by both fellows and PhD students. He was patient and generous with his time, even resorting to paper and pen when language proved a barrier – this was before AI could help with translations!

Our research focus was on the rCOS method, in collaboration with He Jifeng, and Anders made significant contributions to its theory and applications, as in [6].

Outside work, Anders and I shared many joyful moments – eating, hiking, and celebrating birthdays together. With him often alone and my family in the UK during my time in Macao, these times of companionship meant a great deal. Occasionally, his visits coincided with those of my wife Hong (Zhao) and our sons, Kim and Edward. We always enjoyed spending time together, and my family was very fond of him.

On one memorable occasion, Anders visited with his wife Annemette (Lind) and her mother, while my family happened to be in Macao as well. We walked the streets together and shared meals – lunch is commonly called tea in Macao, Hong Kong, and Guangdong Province. Annemette's mother was full of life – cheerful, healthy, and dressed in bright, colourful clothes. She passed away at the age of 97, after a long and happy life.

Final Meeting and Farewell: The last time I saw Anders was in October 2017, when I invited him to Changsha for Zhou's 80th birthday celebration, alongside Dines and Cliff Jones. It had been over four years since we'd last met, in Shanghai at He Jifeng's 70th birthday Festschrift. We were thrilled to reconnect, and I was surprised by Anders' DIY knowledge – he gave Hong tips on sealing windows in our Chongqing apartment. Having just retired, he was full of ideas, and we even discussed plans for him to visit regularly and co-author a few books together.

Sadly, those plans were not to be. Months later, I stopped receiving replies to my emails. In February 2018, en route to Norway with a transfer in Copenhagen, I sent him a quick note. Amazingly, an email from him came almost the same instant – a rare coincidence. But it carried devastating news: he had been diagnosed with a brain tumour.

Over the following months, he occasionally updated me about his treatment and recovery. Eventually, the emails stopped. Then came the saddest messages from Annemette, Dines, and Ernst-Rüdiger: Anders had passed away. I sent my condolences to Annemette, along with a small contribution to help with the funeral. She later wrote back, saying she had bought flowers and placed them at his gravestone.

Anders' passing was a profound personal loss – of a cherished friend, a collaborator, and a dream of future books never written.

Acknowledgement: Thank you to Jonathan Bowen for guidance, comments, and editing.

References

[1] Bjørner, D., Hoare, C.A.R., Bowen, J.P., He, J., Langmaack, H., Olderog, E.R., Martin, U., Stavridou, V., Nielson, F., Nielson, H.R., Barringer, H., Edwards, D., Løvengreen, H.H., Ravn, A.P., Rischel, H.: A ProCoS project description: ESPRIT BRA 3104. Bulletin of the European Association for Theoretical Computer Science **39**, 60–73 (1989), http://researchgate.net/publication/256643262

[2] Bowen, J.P.: Beijing takes the software pearl in Macau's crown. The Times Higher Education Supplement **1409**, 13 (5 Nov 1999)

[3] Bowen, J.P., Gomes, C., Liu, Z. (eds.): Engineering Trustworthy Software Systems: 6th International School, SETSS 2024, Chongqing, China, April 14–21, 2024, Tutorial Lectures, Lecture Notes in Computer Science, vol. 15584. Springer (2025). doi:10.1007/978-981-96-4656-2

[4] Chen, Y., Liu, Z.: From durational specifications to TLA designs of timed automata. In: Davies, J., Schulte, W., Barnett, M. (eds.) Formal Methods and Software Engineering, 6th International Conference on Formal Engineering Methods, ICFEM 2004, Seattle, WA, USA, November 8–12, 2004, Proceedings. Lecture Notes in Computer Science, vol. 3308, pp. 464–482. Springer (2004). doi:10.1007/978-3-540-30482-1_38

[5] Chen, Y., Liu, Z.: Integrating temporal logics. In: Boiten, E.A., Derrick, J., Smith, G. (eds.) Integrated Formal Methods, 4th International Conference, IFM 2004, Canterbury, UK, April 4–7, 2004, Proceedings. Lecture Notes in Computer Science, vol. 2999, pp. 402–420. Springer (2004). doi:10.1007/978-3-540-24756-2_22

[6] Chen, Z., Liu, Z., Ravn, A.P., Stolz, V., Zhan, N.: Refinement and verification in component-based model-driven design. Science of Computer Programming **74**(4), 168–196 (2009). doi:10.1016/J.SCICO.2008.08.003

[7] Grossman, R.L., Nerode, A., Ravn, A.P., Rischel, H. (eds.): Hybrid Systems, Lecture Notes in Computer Science, vol. 736. Springer (1992). doi:10.1007/3-540-57318-6

[8] Hansson, H., Jonsson, B.: A framework for reasoning about time and reliability. In: 10th IEEE Real-Time System Symposium, Santa Monica, Ca. pp. 102–111. IEEE (1989). doi:10.1109/REAL.1989.63561

[9] Hansson, H., Jonsson, B.: A logic for reasoning about time and reliability. Formal Aspects of Computing **6**(5), 512–535 (1994). doi:10.1007/BF01211866

[10] He, J., Li, X., Liu, Z.: Component-based software engineering. In: Hung, D.V., Wirsing, M. (eds.) Theoretical Aspects of Computing – ICTAC 2005, Second International Colloquium, Hanoi, Vietnam, October 17–21, 2005, Proceedings. Lecture Notes in Computer Science, vol. 3722, pp. 70–95. Springer (2005). doi:10.1007/11560647_5

[11] He, J., Li, X., Liu, Z.: rCOS: A refinement calculus of object systems. Theoretical Computer Science **365**(1–2), 109–142 (2006). doi:10.1016/J.TCS.2006.07.034

[12] Hung, D.V., Zhou, C.: Probabilistic duration calculus for continuous time. Formal Aspects of Computing **11**(1), 21–44 (1999). doi:10.1007/s001650050034

[13] Jones, C.B., Liu, Z., Woodcock, J. (eds.): Formal Methods and Hybrid Real-Time Systems, Essays in Honor of Dines Bjørner and Chaochen Zhou on the Occasion of Their 70th Birthdays, Papers presented at a Symposium held in Macao, China, September 24–25, 2007, Lecture Notes in Computer Science, vol. 4700. Springer (2007). doi:10.1007/978-3-540-75221-9

[14] Li, X.: A Mean Value Calculus. Ph.D. thesis, Software Institute, Chinese Academy of Sciences, Beijing, China (1993)

[15] Liu, Z.: Fault-tolerant Programming By Transformations. Ph.D. thesis, University of Warwick, United Kingdom (1991)

[16] Liu, Z., Nordahl, J., Sørensen, E.V.: Composition and refinement of probabilistic real-time systems. In: Mathematics of Dependable Systems, pp. 149–163. The Institute of Mathematics and its Applications Conference Series, Oxford University Press, Oxford (1995)

[17] Liu, Z., Ravn, A.P., Li, X.: Verifying duration properties of timed transition systems. In: Gries, D., de Roever, W.P. (eds.) Programming Concepts and Methods, IFIP TC2/WG2.2,2.3 International Conference on Programming Concepts and Methods (PROCOMET '98) 8–12 June 1998, Shelter Island, New York, USA. IFIP Conference Proceedings, vol. 125, pp. 327–345. Chapman & Hall (1998). doi:10.1007/978-0-387-35358-6_22

[18] Liu, Z., Ravn, A.P., Sørensen, E.V., Zhou, C.: A probabilistic Duration Calculus. In: Kopetz, H., Kakuda, Y. (eds.) Responsive Computer Systems, Dependable Computing and Fault-Tolerant Systems, vol. 7, pp. 29–52. Springer (1992). doi:10.1007/978-3-7091-9288-7_3

[19] Liu, Z., Ravn, A.P., Sørensen, E.V., Zhou, C.: Towards a calculus of system dependability. High Integrity Systems **1**(1), 49–75 (1994)

[20] Liu, Z.: Linking formal methods in software development – A reflection on the development of rCOS. In: Bowen, J.P., Li, Q., Xu, Q. (eds.) Theories of Programming and Formal Methods: Essays Dedicated to Jifeng He on the Occasion of His 80th Birthday. Lecture Notes in Computer Science, vol. 14080, pp. 52–84. Springer (2023). doi:10.1007/978-3-031-40436-8_3

[21] McIver, A., Morgan, C.: Abstraction, Refinement and Proof of Probabilstic Systems. Monographs in Computer Science, Springer (2005). doi:10.1007/b138392

[22] Sørensen, E.V., Ravn, A.P., Rischel, H.: Control program for a gas burner: Part 1: informal requirements, ProCoS case study 1. Tech. rep., Department of Computer Science, Technical University of Denmark (1990), ProCoS I, ESPRIT BRA 3104, Report No. ID/DTH EVS2

[23] Zhan, N.: Higher-order Duration Calculus and its Applications. Ph.D. thesis, Software Institute, Chinese Academy of Sciences, Beijing, China (2000), in Chinese

[24] Zhou, C., Hansen, M.R.: Duration Calculus: A Formal Approach to Real-Time Systems. Monographs in Theoretical Computer Science. An EATCS Series, Springer (2004). doi:10.1007/978-3-662-06784-0

[25] Zhou, C., Hoare, C.A.R., Ravn, A.P.: A calculus of durations. Information Processing Letters **40**(5), 269–276 (Dec 1991). doi:10.1016/0020-0190(91)90122-X

Report on Prof. Dr. Martin Fränzle and his Festschrift Symposium

Jonathan P. Bowen
London South Bank University
United Kingdom

I first met Martin Fränzle on the European ESPRIT ProCoS project on "Provably Correct Systems" (1989–1992) [1], when he was a bright young PhD student under the supervision of Prof. Dr. Hans Langmaack at the Christian-Albrechts-Universität zu Kiel in northern Germany.

We collaborated on presenting a ProCoS tutorial together in 1993 [4]. After his PhD studies, Martin joined Ernst-Rüdiger Olderog, also on the ProCoS project, at the Carl von Ossietzky Universität Oldenburg, not far from Kiel in Germany. It is here that Martin has dedicated his professional academic life to research into formal methods.

During 9–10 March 2015, I co-organized a two-day ProCoS reunion workshop at the BCS London offices, under the auspices of the BCS-FACS Specialist Group [2], with a formally published post-proceedings [8]. Martin presented a paper on constraint-solving techniques for the analysis of stochastic hybrid systems [7]. More recently, we have both been on the academic steering committee for the School on Engineerng Trustworthy Software Systems (SETSS), which has been held regularly in China since 2014, with a break during the COVID pandemic. This provides extended tutorials for postgraduate students with an associated post-proceedings [2]. Martin Fränzle has continued his interest in formal methods research, especially for hybrid discrete-continuous systems, throughout his academic career, and presented a tutorial on AI components for high-integrity, safety-critical human-cyber-physical systems [6] at SETSS 2024 in Chongqing [3].

Figure 1: Oration by Hans Langmaack with Martin Fränzle and his wife in the audience.

Most recently, a Festschrift Symposium was organized at the Carl von Ossietzky Universität Oldenburg in tribute to Martin Fränzle to celebrate his 60th birthday, held on 28 February 2025. I

Figure 2: Oration by Ernst-Rüdiger Olderog and response by Martin Fränzle.

Figure 3: Martin Fränzle and his close colleagues.

attended, and it was good to meet old colleagues from as far back as the ProCoS project, including Hans Langmaack, Ernst-Rüdiger Olderog, and Michael R. Hansen. Hans and Ernst-Rüdiger gave a wonderful oration celebrating Martin's achievements (see Figure 1 and Figure 2, in German), although I later helped Hans to translate his script into English. Many of Martin's close colleagues and students over the years were in attendance (see Figure 3).

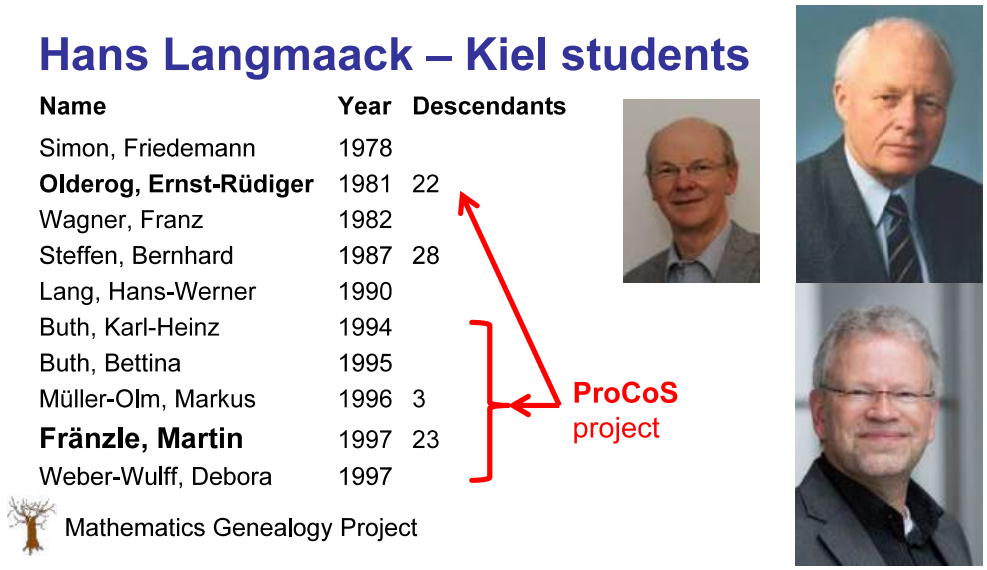


Figure 4: Students of Hans Langmaack.

I presented an abbreviated talk on "Formal Methods: Whence and Whither?",[1] including mention of Martin's presentation at SETSS 2024 and an exploration of his academic heritage in terms of his "family tree" of advisors. A significant number of Hans Langmaack's doctoral students were involved with the ProCoS project, including Martin Fränzle and Ernst-Rüdiger Olderog (see Figure 4). Tracing Martin's academic heritage back by four generations of supervision on the Mathematics Genealogy Project, one reaches David Hilbert (1862–1943), one of the most influential mathematicians of the early 20th century (see Figure 5).

Five generations back from Hilbert, one reaches Carl Friedrich Gauss (1777–1855), another giant of the field of mathematics and a polymath (see Figure 6). Another five generations back, one reaches Otto Mencke (1644–1707), a philosopher and scientist who founded *Acta Eruditorum*, the first German scientific journal, in 1682 (see Figure 7). Mencke's advisor was Jakob Thomasius (1622–1684), who was also advisor of Gottfried Wilhelm Leibniz (1646–1716). Thomasius's advisor was Friedrich Leibniz (1597–1652), the father of Gottfried Leibniz (see Figure 8). Projecting forwards from Abraham Gotthelf Kästner (1719–1800), "grandfather" of Gauss and a Fellow of the Royal Society, one eventually even reaches the Bowen family through a line of chemists (see Figure 9).

[1] An extended and adapted version of the talk is planned for the Annual FACS Peter Landin Semantics Seminar on 18 December 2025 at the BCS London office.

Martin Fränzle – whence?

(Dr. rer. nat., Universität zu Kiel, 1997)

Dissertation: *Controller Design from Temporal Logic: Undecidability Need Not Matter*

- **Hans Langmaack** (Dr. rer. nat., Münster, 1960)
- Heinrich Adolph Behnke (Dr. phil., Hamburg, 1923)
- Erich Hecke (Ph.D., Göttingen, 1910)
- **David Hilbert** (Ph.D., Königsberg, 1885)

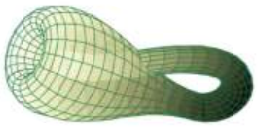
"great great grandfather"

Mathematics Genealogy Project

Figure 5: Supervisor line for Martin Fränzle to David Hibert.

David Hilbert

- Ferdinand von Lindemann (Erlangen-Nürnberg, 1873)
  – transcendantal numbers
- Felix Klein (Bonn, 1868)
  – Klein bottle



- Julius Plücker (Marburg, 1823)
  – winner of the 1866 Royal Society Copley Medal for "analytical geometry, magnetism, & spectral analysis"
- Christian Ludwig Gerling (Göttingen, 1812)
  – geodetic triangulations
- **Carl Friedrich Gauß** (Helmstedt, 1799)
  – mathematician, astronomer, geodesist, and physicist

Mathematics Genealogy Project

Figure 6: Supervisor line for David Hilbert to Carl Gauss.

Carl Friedrich Gauß

Möbius strip

- Johann Friedrich Pfaff (Göttingen, 1786)
  – "Pfaffian" systems, also advisor of **August Möbius**
- Abraham Gotthelf Kästner (Leipzig, 1739)
  – textbooks & encyclopedias, Fellow of the Royal Society
- Christian August Hausen (Halle-Wittenberg, 1713)
  – electrical phenomena
- Johann Christoph Wichmannshausen (Leipzig, 1685)
  – philology & philosophy
- **Otto Mencke** (Leipzig, 1665)
  – founder of *Acta Eruditorum*, first German scientific journal

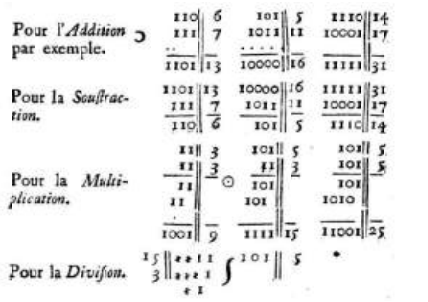
Mathematics Genealogy Project

Figure 7: Supervisor line for Carl Gauss to Otto Mencke.

Otto Mencke

- Jakob Thomasius (Leipzig, 1643)
  – also advisor of **Gottfried Wilhelm Leibniz**
- Friedrich Leibniz (Leipzig, 1739)
  – father of **Gottfried Wilhelm Leibniz**

Mathematics Genealogy Project



Binary arithmetic by Gottfried Leibniz

Figure 8: Supervisor connections for Otto Mencke with the Leibniz family.

Figure 9: Supervisor connection of Abraham Kästner with the Bowen family!

In conclusion, we congratulate Prof. Dr. Martin Fränzle on his contributions to the field of formal methods over the years and his excellent academic pedigree. A post-proceedings Festschrift volume is due out in the Springer Lecture Notes in Computer Science series. Happy belated birthday!

References

[1] Bjørner D., Hoare, C.A.R., Bowen, J.P., He, J., Langmaack, H., Olderog, E.-R., Martin, U.H., Stavridou, V., Nielson, F., Nielson, H.R., Barringer, H. Edwards, D., Løvengreen, H.H., Ravn, A.P., Rischel, H.S.: A ProCoS Project Description. Bulletin of the European Association for Theoretical Computer Science (EATCS), **39**:60–73, October (1989). **http://researchgate.net/publication/256643262**

[2] Bowen, J.P. (2016). BCS-FACS – ProCoS Workshop on Provably Correct Systems. *FACS FACTS*, **2016**(1):14–34, March. BCS. **https://www.bcs.org/media/3087/facs-mar16.pdf**

[3] Bowen, J.P. (2024). Report on the SETSS 2024 Spring School. *FACS FACTS*, **2024**(2):60–71, July. BCS. **https://www.bcs.org/media/1wrosrpv/facs-jul24.pdf**

[4] Bowen, J.P., Fränzle, M., Olderog, E.-R., Ravn, A.P.: Developing correct systems. In: 5th Euromicro Workshop on Real-Time Systems, Oulu, Finland, pp. 176–187. IEEE Computer Society Press (1993). doi:10.1109/EMWRT.1993.639088

[5] Bowen, J.P., Gomes, C., Liu. Z. (eds): Engineering Trustworthy Software Systems: 6th International School, SETSS 2024, Chongqing, China, April 15–21, 2024. LNCS, vol. 15584. Springer, Cham (2025). doi:10.1007/978-981-96-4656-2

[6] Fränzle, M.: AI Components for High Integrity, Safety-Critical Human-Cyber-Physical Systems – A Challenge for Formal Methods In: [2], pp. 102–133 (2025). doi:10.1007/978-981-96-4656-2_4

[7] Fränzle, M., Geo, Y., Gerwinn, S.: Constraint-solving techniques for the analysis of stochastic hybrid systems. In: [8], pp. 9–38 (2017). doi:10.1007/978-3-319-48628-4_2

[8] Hinchey, M.G., Bowen, J.P., Olderog, E.-R. (eds) Provably Correct Systems. NASA Monographs in Systems and Software Engineering. Springer, Cham (2017). doi:10.1007/978-3-319-48628-4

Report on SETSS 2025 workshop and school, Beijing, China, 17–23 May 2025

Jonathan P. Bowen
London South Bank University
United Kingdom

SETSS Overview

The Seventh International School on Engineering Trustworthy Software Systems (SETSS 2025) and an associated workshop took place at the Institute of Software, Chinese Academy of Sciences (ISCAS, see Figures 1 and 2), Beijing, China, during 17–23 May 2025.

Figure 1: The Institute of Software Chinese Academy of Sciences (ISCAS) building sign.

From 2014 to 2019, five annual consecutive editions of the International School on Engineering Trustworthy Software Systems (SETSS) took place, with international speakers presenting extended tutorials. These events were popular within the Chinese software engineering community, particularly among young researchers (mainly postgraduates) interested in formal methods for trustworthy software. The 6th edition was suspended due to the COVID pandemic. However, the School resumed in 2024, with the postponed 6th edition taking place during 15–21 May 2024, at Southwest University in Chongqing, China [1, 2]. It is intended that SETSS will become an annual event again.

Figure 2: Jonathan Bowen and a SETSS 2025 banner at ISCAS.

The academic program of SETSS 2025 offered five days of extended tutorial courses (in two to four 90-minute sessions), along with a two-day workshop immediately beforehand. This year's thematic focus was on the modelling, specification, and verification of human-cyber-physical systems, with a particular attention to robotics and AI. The courses were led by international experts in the field, invited by the Steering Committee of SETSS 2025 (see Figure 5). The associated workshop provided an opportunity for school participants to exchange research ideas. This included presentations on proposed research questions, ongoing work, and new research results.

In addition to the academic schedule, informal discussions about research, teaching, and potential collaborations were also encouraged. Besides the scientific aspects, participants also had the opportunity to enjoy the historical and cultural sights of Beijing, a city with many cultural heritage attractions, include the restored Ming Dynasty (1505) section of the Great Wall of China at Badaling (see Figure 3), within a short 20-minute fast train ride from Beijing.

Figure 3: Evening view of the Great Wall of China at Badaling.

All the previous editions have resulted in Springer LNCS volumes as post-proceedings after the event. It has been agreed with Springer Beijing that SETSS 2025 proceedings will appear in a similar way in the LNCS Tutorial series, in time for SETSS 2026.

SETSS 2025 Workshop, 17–18 May 2025

There were seven workshop speakers, all from China, who delivered shorter presentations:

Figure 4: The start of the SETSS 2025 Workshop.

- *Certifying Adversarial Robustness in Quantum Machine Learning: From Theory to Physical Validation* by Ji Guan, Institute of Software
- *On Temporal Verification of Stateful P4 Programs* by Fei He, Tsinghua University
- *Formal Synthesis of Neural Controllers for Autonomous Systems* by Wang Lin, Zhejiang Sci-Tech University
- *Runtime Backdoor Detection for Federated Learning via Representational Dissimilarity Analysis* by Meng Sun, Peking University
- *Neural Code Generation Models with Programming Language Knowledge* by Yingfei Xiong, Peking University
- *Stochastic System Safety: Probabilistic Invariance Verification* by Bai Xue, Institute of Software
- *New Progress in Computational Number Theory and Number-Theoretic Cryptography* by Song Y. Yan, Zhejiang Normal University

The Workshop was chaired by Wanwei Liu of the National University of Defense Technology, China.

SETSS 2025 School, 19–23 May 2025

The School was introduced by Lijun Zhang of ISCAS and Zhiming Liu of Southwest University, including mention of the SETSS Steering Committee (see Figure 5), the local organizers (see Figure 6), and the tutorial presenters (see Figure 7). The following tutorials were presented by invited speakers at SETSS 2025, consisting of two to four 90-minute sessions.

Figure 5: The SETSS 2025 Steering Committee, introduced by Zhiming Liu.

Figure 6: The SETSS 2025 local organizers, introduced by Lijun Zhang.

Figure 7: The SETSS 2025 tutorial presenters.

Advances on SAT and SMT Solving

Shaowei Cai, ISCAS, Beijing, China

Shaowei Cai is a Professor at Institute of Software, Chinese Academy of Sciences, leader of the Constraint Solving group. He received his PhD degree from Peking University with Distinguished Doctoral Dissertation Award. His research interests include constraint solving and formal methods. Particularly, he has regularly won Gold/Silver medals in recent SAT Competitions, including 5 Gold medals in Parallel tracks won by the PRS solver in his team. The SMT solvers Z3++ and Parti-Z3++ has won "largest contribution" award and "largest leading" awards in Model Validation track and Cloud track in SMT Competitions. He received the Best/Distinguished Paper Award at SAT 2021, CP 2024 ,and CAV 2024. He has given tutorials at FMCAD and SoCS conferences.

Building Safe Autonomous Systems using Imperfect Components

Samarjit Chakraborty, The University of North Carolina at Chapel Hill, USA

Samarjit Chakraborty is a Kenan Distinguished Professor of Computer Science at UNC Chapel Hill. He is also an adjunct professor of Mathematics at UNC. Prior to coming here, he was a professor of Electrical Engineering at the Technical University of Munich in Germany, where he held the Chair of Real-Time Computer Systems for 11 years. Before that, he was an assistant professor of Computer Science at the National University of Singapore. He obtained his PhD degree from ETH Zurich. His research interests cover all aspects of designing hardware and software for embedded computers, with an emphasis on cyber-physical systems design, sustainable computing, and sensor network-based information processing. He serves/d on the editorial boards of several journals, including the ACM Transactions on Cyber-Physical Systems and the ACM Journal on Autonomous Transportation Systems. He and his students have won several best paper awards for their work, including the 2019 ACM Transactions on Design Automation of Electronic Systems

Best Paper Award for their work on automotive security, and the 2021 ACM Transactions on Embedded Computing Systems Best Paper Award for their work on energy modeling of the Bluetooth Low Energy protocol. He is a Fellow of the IEEE and was offered a Humboldt Professorship from Germany in 2023.

Model Checking, Monitoring, Performance Analysis, Synthesis and Learning for Cyber-Physical Systems

Kim Guldstrand Larsen, Aalborg University, Denmark

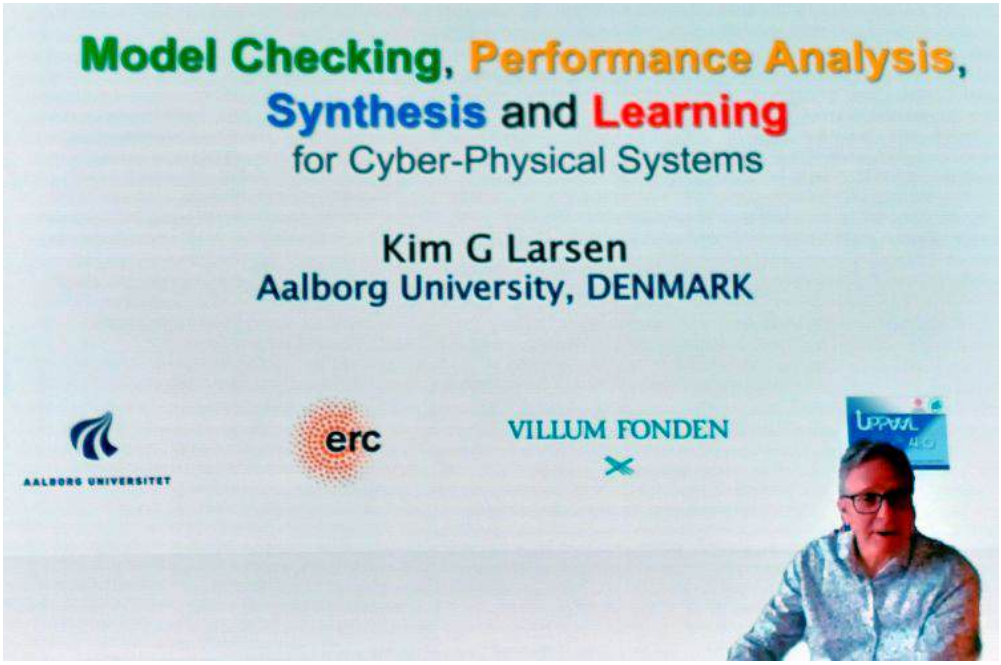


Figure 8: Kim Larsen presenting online from Demark.

Kim Guldstrand Larsen (see Figures 8 and 9) is Professor in Computer Science at Aalborg University, Denmark. His field of research includes modeling, validation and verification, performance analysis, and synthesing of real-time, embedded, and cyber-physical systems utilizing and contributing to concurrency theory, model checking and model checking. Kim Guldstrand Larsen is co-founder and main contributor to the tool UPPAAL. UPPAAL received the prestigious CAV Award in 2013 as the foremost tool for modelling and verification of real-time systems. Kim Guldstrand Larsen won the ERC Advanced Grant in 2015 and won a Villum Investigator grant in 2021. In 2022 he received the CONCUR Test-of-Time Award 2022.

Kim Guldstrand Larsen is a member of Royal Danish Academy of Sciences and Letters, elected fellow and digital expert (vismand) in the Danish Academy of Technical Sciences and Knight of the Order of the Dannebrog (2007). Moreover he is Honorary Doctor, Uppsala University (1999), Honorary Doctor, École normale supérieure Paris-Saclay, Paris (2007), Foreign Expert of China, Distinguished Professor, Northeastern University (2018). He has published around 500

Figure 9: David N. Jansen of ISCAS (a local organizer) chairing the online session with Kim Larsen.

peer-reviewed papers and received the Thomson Scientific Award as the most cited Danish computer scientist 1990–2004.

Hardware-Software Leakage Contracts for Side-Channel Security

Jan Reineke, Saarland University, Germany

Jan Reineke is a professor of computer science at Saarland University. Before joining Saarland University in 2012, he was a postdoctoral scholar at UC Berkeley in the Ptolemy group from 2009 to 2011. He completed his MSc and PhD degrees in Computer Science at Saarland University in 2005 and 2008, respectively, and his BSc in Computing Science at the University of Oldenburg in 2003. His research centers around problems at the boundary between hardware and software.

In the area of real-time systems, he is particularly interested in principles for the design of timing-predictable hardware and in precise and efficient timing-analysis techniques for multi-core architectures. His recent results include the design of the first provably timing-predictable pipelined processor design (RTSS 2018) and the first exact analyses for LRU caches (CAV 2017, POPL 2019, RTSS 2019). Another focus of his work are security vulnerabilities of hardware-software systems. Recent results include the development of automatic techniques to detect information leaks introduced by speculative execution (Spectector, S&P 2020), techniques to quantify the information leakage through cache side channels (ACM TISSEC 2015), and automatic methods to obtain highly detailed performance models for modern microarchitectures (uops.info, ASPLOS 2019).

Pedro Ribeiro, University of York, United Kingdom

Model-based Software Engineering for Robotics

Figure 10: Pedro Ribeiro used interactive software via smartphones to enable audience participation.

Pedro Ribeiro (see Figure 10) is a Lecturer (Assistant Professor) in Computer Science at the University of York, UK. Previously, he was a Research Fellow in the School of Physics Engineering and Technology at York, and before that a Research Associate in Computer Science. He completed his PhD degree at York on the treatment of angelic nondeterminism for process calculi. He has over a decade of experience with formal approaches to software engineering relevant to robotics and cyber-physical systems more generally. His research interests span the breadth of the engineering lifecycle, spanning from design and development of domain-specific notations and their semantics, to testing and formal reasoning using automated mathematical proof techniques. He is a member of the York's RoboStar centre for Excellence in Software Engineering for Robotics, and a founding member of Formal Methods Europe's communications committee.

Simulation Testing of Autonomous Driving Systems Based on Safety-Critical Scenario Generation

Yinxing Xue, University of Science and Technology of China, Hefei, China

Xue Yinxing is currently a Research Professor at the Institute of Artificial Intelligence for Industries, specializing in software security. He received his PhD degree from the National University of Singapore (NUS) and subsequently conducted defense-related research at the Temasek Laboratories of National University of Singapore and Nanyang Technological University in Singapore. In

2018, he was recruited through the CAS Talent Program to join the University of Science and Technology of China (USTC). He has led six national and provincial/ministerial-level research projects. Notably, he has spearheaded two international standards for intelligent connected vehicles (ICVs) at ISO/ITU, with one ITU standard and one IEEE standard successfully being approved and under development. His recent academic achievements include: 24 first/corresponding-author publications in the past six years (19 in CCF Rank A venues), two-time recipient of the ACM SIGSOFT Distinguished Paper Award.

Safeguarding Deep Reinforcement Learning Systems via Formal Methods: From Safety-by-Design to Runtime Assurance

Min Zhang, East China Normal University, Shanghai, China

Min Zhang is a full professor at the Software Engineering Institute, East China Normal University. He earned his PhD degree from the Japan Advanced Institute of Science and Technology in 2011, and joined East China Normal University in 2014. From 2019 to 2021, he served as a senior visiting professor at Nice University. His research interests primarily focus on formal methods for safety-critical systems, including real-time and AI-empowered systems. More recently, he has concentrated on the formal verification of deep neural networks and intelligent systems. He has co-authored over 80 papers, published in top-tier conferences such as CAV, TACAS, ASE, ICSE, NeurIPS, CVPR.

Figure 11: Lijun Zhang (left) of ISCAS, who introduced the School, and Jim Woodock (right) of Southwest University (a SETSS Steering Committee member), who summed up at the end of the School.

Acknowledgements and Further Information

Lijun Zhang (see Figure 11) of ISCAS was in overall charge of SETSS 2025 and Andrea Turrini, also of ISCAS, was the main organizer. SETSS 2025 was supported by the China Computer Federation (CCF), ISCAS, and Southwest University.

ISCAS was a wonderful host for the event, treating the tutorial speakers and Steering Committee members to dinner each evening (see Figures 12, 13, and 14).

Figure 12: Lijun Zhang cooking hot pot, Beijing style, which is not quite as hot as Chongqing style!

For further information on SETSS 2025, including online copies of slides for the tutorials and workshop presentations, see:

https://tis.ios.ac.cn/SETSS2025/

References

[1] Bowen, J.P. (2024). Report on the SETSS 2024 Spring School. *FACS FACTS*, **2024**(2):60–71, July. BCS. **https://www.bcs.org/media/1wrosrpv/facs-jul24.pdf**

[2] Bowen, J.P., Gomes, C., Liu. Z. (eds): Engineering Trustworthy Software Systems: 6th International School, SETSS 2024, Chongqing, China, April 15–21, 2024. LNCS, vol. 15584. Springer, Cham (2025). doi:10.1007/978-981-96-4656-2

Figure 13: A more formal dinner with Chinese “Great Wall” red wine. Left to right: Schmuel Tyszberowicz (Steering Committee member), Jan Reineke(tutorial speaker), Andrea Turrini (local organizer), and Pedro Ribeiro (tutorial speaker).

Figure 14: Schmuel Tyszberowicz, Jonathan Bowen, and Andrea Turrini, during a post-prandial walk.

Functional programming and dependent types for Metrology

Keith Lines
Data Science
National Physical Laboratory
Teddington

André Videla
Mathematically Structured Programming Group
University of Strathclyde

(**Reported by:** Brian Monahan)

Keith Lines, NPL
Photo by Jonathan Bowen

André Videla, Strathclyde
Photo by Jonathan Bowen

This presentation summarised a collaborative project between the Data Science Department of the National Physical Laboratory and the Mathematically Structured Programming Group of the University of Strathclyde. The aim was to investigate how functional programming and dependent typing in particular could increase the trustworthiness of software used in measurement science (that is, *metrology*).

Code written in a functional language can provide a more self-evidently correct implementation of an underlying mathematical model than code written in a more traditional imperative language. Dependent types can provide helpful contextual information, useful for type-checking and review. For example, a matrix containing measurement values could use a column index to determine the unit of measurement associated with the values held in that column. Such features are built into the type system itself, and no extra coding is required.

An important challenge was to determine how to make such topics more accessible to metrologists, a technically knowledgeable audience but not one consisting of specialists in either computer science or functional programming. Several case studies demonstrated advantages that functional programming languages, including those that provide dependent types like **Idris**, can provide when compared to more commonly used languages such as **Python**.

As someone with some background in functional programming, I thought the talk provided a helpful concrete introduction to the more recent developments in functional programming. It did this by examining several helpful concrete domain examples all drawn from the area of metrology to illustrate the ideas. André and Keith avoided elaborate technical introductions to **Idris**, the functional programming language used in this study. This was wise, given that modern-day functional languages tend to be somewhat specialist – and not for the faint of heart!

By doing so, this kept the focus on the application area, not letting the sophistication of the technology overpower and dominate the presentation. At the same time, those of us who no doubt will have been intrigued to learn more about **Idris** will surely be quite capable of seeking that out for ourselves [1]. Overall, keeping the focus on concrete application rather than the finesse of the technology helps establish a degree of credability for the dependent types approach, and to that extent the talk was a clear success.

An full report of this work from NPL is available online [2], and a YouTube video of the talk can be found here.

References

[1] *Idris: A Language for Type-Driven Development*
https://www.idris-lang.org

[2] Videla, A; Lines, K (2025)
Functional Programming (with some Type Theory) for Metrology.
NPL Report. MS 60, doi:10.47120/npl.MS60
http://eprintspublications.npl.co.uk/10181/1/MS60.pdf

Joint BCS-FACS / FME Seminar
Formal Methods and Tools in Railways

Maurice ter Beek
CNR-ISTI
Pisa, Italy

(**Reported by:** Keith Lines)

Abstract

Formal methods and tools have been successfully applied to the development of safety-critical systems for decades, in particular in the transport domain, without a single language or tool emerging as the dominant solution for system design.

Formal methods are highly recommended by the current safety standards in the railway industry, but railway engineers often lack the knowledge to transform their semi-formal models into formal models, with a precise semantics, to serve as input to formal methods tools.

We share the results of performing empirical studies in the railway domain, including usability analyses of formal methods tools involving railway practitioners. We discuss, in particular with respect to railway systems and their modelling, our experiences in applying formal methods and tools to a variety of case studies, for which we interacted with a number of railway companies. We report on lessons learned from these experiences and provide pointers to steer future research.

Figure 1: Maurice ter Beek was introduced by Ana Cavalcanti

1 Introduction

In March FACS had the pleasure of hosting Formal Methods Europe (FME) [1] at the BCS London offices, for their committee meeting and annual general meeting. The event was concluded with an excellent presentation by Maurice ter Beek, who was introduced by Ana Cavalcanti of the University of York (also FME chair and a FACS committee member).

Maurice heads the Formal Methods and Tools Laboratory (FMT) [2] of the Istituto di Scienza e Tecnologie dell'Informazione "Alessandro Faedo" (ISTI). ISTI is part of the National Research Council of Italy (CNR), Italy's largest public research institution.

For over 10 years railways have provided a favourite, and fruitful, area of research for FMT. The extremely safety-critical nature of railway systems makes use of formal methods highly recommended in railway-related standards such as EN 50128 [3].

2 Formal methods and tools: A systematic evaluation

An overview was presented of an evaluation of the use of formal methods and tools in railways. It was interesting to note that the main application of formal methods lies within specification-based activities. Details of the evaluation are published in [4].

3 Success stories

Arguably, Jean-Raymond Abrial's B-method provides the most widely known examples of the successful application of formal methods within the rail industry [5].

FMT has used formal methods and related tools to investigate deadlock avoidance in train scheduling, next generation railway signalling systems, synthesis of autonomous driving strategies, smart railway systems and stations of the future. A variety of languages and tools, such as UPPAAL [6], were used.

4 Future challenges

Big data analytics can help with the ongoing challenge of predictive maintenance in railways. For example, FMT has collaborated with a rail operator for the Lombardy region (Trenord) to develop data-driven strategies for monitoring on-board equipment.

Research in the use of AI-based systems in railways is at an early stage. Formal methods and tools must be able to help analyse such systems.

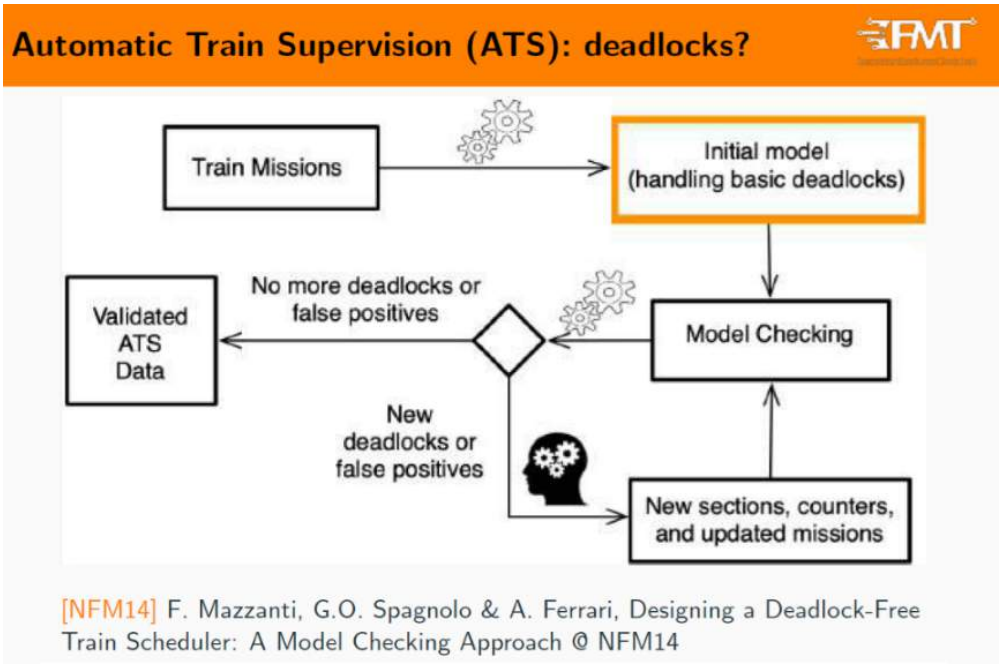


Figure 2: Modelling to ensure deadlock avoidance

5 Key take-away points

- Models defined using semi-formal methods, such as UML, provide a popular and accessible entry point to the use of more formal techniques (i.e., those with formal semantics).
- Formal methods education is key to the future of these techniques [7].

6 Question and answers

The presentation ended with a Q+A session chaired by Ana Cavalcanti. Topics covered included whether Dines Bjørner's domain engineering would have value for this work. Maurice was not aware of any application of this approach in the railway domain, but it would be worth exploring. Also, has there been cross-influence with other safety-critical domains such as aerospace? The comparison with aerospace is interesting. Maurice noted an example he was involved with where the aerospace vehicles had greater freedom of movement (i.e., not restricted to tracks) and were nowhere near as close to each other as would be the case with railways.

Biography

Maurice ter Beek is a director of research at CNR-ISTI (Pisa, Italy) where he heads the Formal Methods and Tools lab. He obtained his PhD at Leiden University (The Netherlands) and has authored over 150 peer-reviewed papers, edited over 35 proceedings and special issues of journals.

He serves on the editorial boards of several journals, has recently been appointed co-Editor-in-Chief of Formal Aspects of Computing, and is a board member of Formal Methods Europe (FME).

His research interests are formal methods and model-checking tools for the specification and verification of safety-critical software systems and communication protocols, focusing on applications in software product line engineering and railway systems.

References

[1] Formal Methods Europe: **https://www.fmeurope.org**

[2] Formal Methods and Tools Laboratory, CNR-ISTI: **https://www.fmt.isti.cnr.it**

[3] CENELEC EN 50128:2011+A2 Railway applications – Communication, signalling and processing systems – Software for railway control and protection systems. **https://landingpage.bsigroup.com/LandingPage/...**

[4] A. Ferrari, F. Mazzanti, D. Basile and M. H. ter Beek, "Systematic Evaluation and Usability Analysis of Formal Methods Tools for Railway Signalling System Design". *IEEE Transactions on Software Engineering*, vol. 48, no. 11, pp. 4675–4691, 1 Nov. 2022, doi:10.1109/TSE.2021.3124677

[5] A. Ferrari and M. H. ter Beek, " Formal Methods in Railways: A Systematic Mapping Study". *ACM Computing Surveys*, vol. 55, no. 4, pp. 69:1–69:37, 21 Nov. 2022, doi:10.1145/3520480

[6] UPPAAL tool homepage: **https://uppaal.org**

[7] M. ter Beek, M. Broy and B. Dongol, "The Role of Formal Methods in Computer Science Education". *ACM Inroads* vol. 15, no. 4, pp. 58–66, 11 Nov. 2024, doi:10.1145/3702231

Joint BCS-FACS / LMS Seminar
Probabilistic Datatypes

Annabelle McIver
Macquarie University, Australia

(**Reported by:** Andrei Popescu)

Annabelle discussed a novel probabilistic semantics that she and her coauthors recently introduced in order to accommodate a correct notion of refinement for probabilistic datatypes, such that inlining (replacing a call to an operation with its actual code) is a correct program transformation. The problem with the standard semantics based on Markov Decision Processes (MDPs) is that intuitively correct implementations of probabilistic behavior may rely on information being kept hidden from the caller, and this abstraction could be broken by inlining. Annabelle illustrated this on a simple coin toss example, where for efficiency the implementor decides to "pre-flip" the coin while waiting for the function to be called – which in the MDP-based semantics cannot be recognized as a correct implementation, due to the noncommutative interaction between demonic (nondeterministic) choice and probabilistic choice; roughly speaking, the outcome of pre-flipping the coin can be "picked up" by a subsequent demonic choice, altering the expected probabilities.

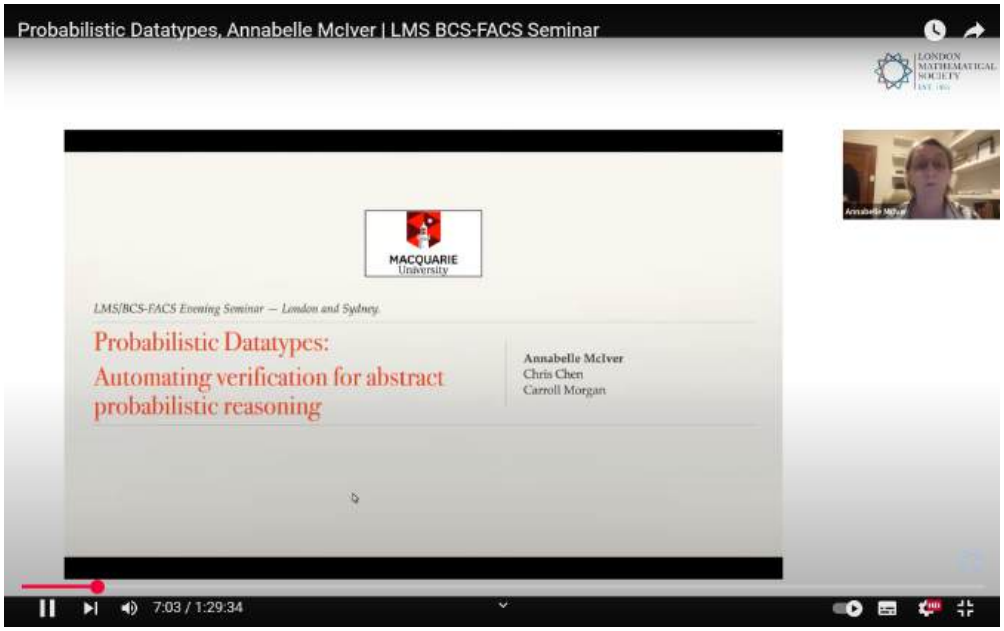


Figure 1: Automating verification for abstract probabilistic reasoning

A solution to this problem, Annabelle showed, comes from the world of quantitative information flow – specifically, by employing a semantics that distinguishes between hidden and visible state, based on *Partially* Observable Markov Decision Processes (POMDPs), and only allowing demonic choice to consider the visible state. Annabelle demonstrated how, under this semantics, the intuitive notions of refinement and simulation work smoothly in order to validate correct hidden-state-based implementations. But there is a price: It is the implementor's responsibility that information

about the hidden state is only revealed when necessary. This leads to an interesting situation where functional correctness depends on information flow security. More details can be found in a recent paper by Annabelle and coauthors [1].

Judging by the large number of questions, the (online) audience seemed highly engaged with Annabelle's talk. They wanted to learn more about the underlying probabilistic computational model, the interplay between probabilistic behavior and information flow, and the possibility of extending the results to a functional programming paradigm.

Biography

Annabelle McIver is a Professor of Computer Science at Macquarie University and co-director of the Future Communications Research Centre. She studied mathematics at the University of Cambridge and earned a doctorate in mathematics from the University of Oxford. A leading figure in formal methods, she specializes in the verification of probabilistic programs and the foundations of quantitative information flow. She has co-authored two influential books in the field, which have become standard references [2, 3].

The video of this talk may be found **here**, and where the slides for the talk may be found **here**.

References

[1] Chris Chen, Annabelle McIver, Carroll Morgan (2024)
Source-Level Reasoning for Quantifying Information Leaks,
in *Principles of Verification: Cycling the Probabilistic Landscape - Essays Dedicated to Joost-Pieter Katoen on the Occasion of His 60th Birthday, Part I*, (eds) Nils Jansen, Sebastian Junges, Benjamin Lucien Kaminski, et al, LNCS 15260, pp98–127, Springer doi:10.1007/978-3-031-75783-9_5

[2] Annabelle McIver, Carroll Morgan, (2005)
Abstraction, Refinement and Proof for Probabilistic Systems
Springer, Berlin

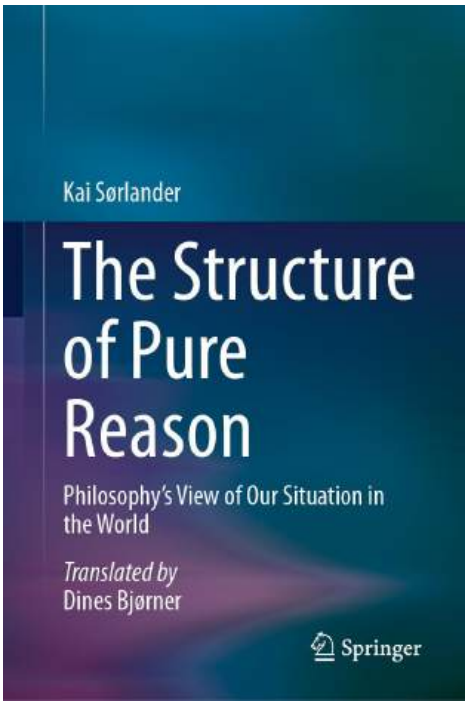
[3] A. McIver, M. S. Alvim, C. Palamidessi, K. Chatzikokolakis, C. Morgan, and G. Smith, (2020)
The Science of Quantitative Information Flow
Springer, Berlin

Book review: The Structure of Pure Reason

Philosophy's View of Our Situation in the World

Kai Sørlander

(**Reviewed by:** Tim Denvir and Brian Monahan)



The Structure of Pure Reason: Philosophy's View of Our Situation in the World, by Kai Sørlander, Translated by Dines Bjørner, 117 pages, Published by Springer Nature, 2024, ISBN:978-3031833007

This is an unusual book review for a newsletter dedicated to supporting Formal Methods of software development. Notably, this book is a work of (pure) philosophy and doesn't contain any particular references to software, computer systems or even mathematical theorems – not even one equation!

However, what brought Danish philosopher Kai Sørlander's work to our attention was the interesting fact that Dines Bjørner, a computer scientist (likely well-known to the majority of FACS readers, if not personally, at least by repute) took on the work of actively translating this from the original Danish into English, clearly making it something of a personal project.

In his Translator's foreword, Bjørner starts by introducing himself, giving a very brief selected account of some of the events in software engineering, including requirements engineering, relating that "Kai Sørlander's Philosophy as outlined in five of his books". Bjørner further states that he devoted a chapter of his latest book to a summary of Sørlander's philosophical work, which he first came across in 2014 [1]. Since Kai Sørlander's works were all in Danish, Dines Bjørner decided to, in 2023, translate the original Structure of Pure Reason into English [2]. All in all, this has piqued our interest sufficiently to explore and write this review.

Dines Bjørner
Photo by Kari Bjørner

At this stage, we should say that your book reviewers have no particular background in academic philosophy, apart from that gained by a general acquaintance with various philosophical ideas, reached through reading around the subject. What follows here is therefore a broad sketch of the work's overall content in general terms and our attempt to convey an appreciation of the argument given therein.

The book opens with three preliminary parts, a Foreword, a Translator's Foreword by Dines Bjørner, and the author's Preface. The main body of the work consists of seven quite substantial chapters, each with its subdivisions. All of this is completed by a brief Conclusion chapter. Unfortunately,the book has no index, and a query to the publisher revealed that the print edition did not have one either. It is not so necessary for an online edition because one can search for

words or phrases using the 'find' facility, but an index is more necessary in a hard-copy.

Curiously, the Foreword by Georg Henrik von Wright given here is from an earlier book by Sørlander [3], published over thirty years previously in 1993/1994. However, the reason for this is likely due to this foreword providing considerable support and useful context for the reader, as the current book is a close development of that predecessor. Written in 1993, von Wright's foreword reveals a rich history of association with Sørlander from the latter's student days, yet surprisingly, without ever having met him. From this foreword, Sørlander comes out as a Platonist, an Aristotelian believer[1] that there is a state of the world which "is the case", to echo the famous opening lines from Wittgenstein's Tractatus.

But what is this book really all about? A good pointer here is found in the author's brief Preface, where he states, "Philosophy asks for the deepest truth about the world and our situation in the world: that which under no circumstances could be otherwise". The author begins with this essentially metaphysical, Platonist/Aristotelian stance, but then immediately states that the widespread view today is that "there is no ultimate truth, but that there are only particular and local truths". Sørlander rebuts this position as self-contradictory, and reverts to the earlier Platonist view that: "we have to start all over again and ask the philosophical question from the ground up". The author asserts that the book gives "a definitive answer to the philosophical question [of what is the truth about the world and our situation in the world]". A bold claim indeed!

This position is further emphasised in the author's brief 280-word introduction, entitled **The Task of Philosophy**, which strongly reiterates Sørlander's opposition to the widely held "no ultimate truth" position, along with his notion of inescapable truth. He draws an analogy with Nietzsche's idea that one can determine one's own spiritual strength by defining one's own moral truth, which is condemned as a misunderstanding. By contrast, Sørlander approves of Kant but declares that in this book, he will go deeper. Sørlander lists three of his previous books, since they form a progression along the same theme, especially [3].

We now turn to a brief summary of each of the seven chapters that make up this book, plus the conclusion. In **Chapter 1, Philosophy and Rationality**, Sørlander proclaims that "our philosophical discourse must be rational, based on conceptual logic. Philosophy begins with a wonder, but from there it develops into a rational question about what is so necessary that it could not be under any circumstances different." The author's primary task in this chapter seems to be to establish some basic ground rules for his exploration and to motivate the central question: "what is so necessary that it could not rationally be otherwise?". The author continues, writing on originality, independence and fallibility: Philosophical originality implies thinking independently and innovatively, but is not in itself infallible; philosophical errors can lead to significant advances when they are later thought through. Sørlander refers to the ancient Greek philosophers, including Plato, Aristotle, and the pre-Socratics, but does not provide an analysis of their ideas at this point.

In Section 1.7, the author criticises again what he sees as the present-day position, namely more of a relativistic standpoint, represented, as he sees it, by Heidegger and postmodernism; he condemns current philosophical teaching in universities; asserting that they, and "the whole of the societal culture that they represent" have gone astray, a bold claim indeed.

[1] Or even a "post-Wittgensteinian Kantianite" as von Wright says in his foreword.

Although it is stated early on that Sørlander wrote a gold medal thesis on Wittgenstein's early philosophy, it is quite surprising that there is no reference by Sørlander to Wittgenstein himself in the whole of the text of this book. Sørlander must surely have read Wittgenstein's *On Certainty*, which has as its (depressing) thesis a severely contradictory view to Sørlander's own.

Turning now to **Chapter 2: Kant and Hume on Causality**, this is where the author gives an account of Kant's reaction to Hume's previous ideas on causality, favouring Kant's position but then finding that too is deficient, and finally positing Sørlander's own analysis. Further research with various resources confirms that Sørlander appears to have a sound understanding of Hume's and Kant's philosophies.

Chapter 3: What Philosophy Can Learn from the Special Development of Physics is an intriguing title for a chapter in a philosophy text, especially one dealing with metaphysics. Generally speaking, within philosophy, it is the general case which is seen to embody logically prior, overarching rational principles, so that the particular case would then appear as some kind of instance of that general case. But here we see that, instead, philosophy itself should heed generally how the physical sciences have themselves developed.

Sørlander gives a quick summary of Isaac Newton's classical theory of mechanics. This is then followed, even more quickly, by a summary of Einstein's Special and General Theories of Relativity. Next, the principles of Quantum Mechanics as developed by Niels Bohr and then Max Planck are briefly explored. Both these developments are contrasted with Newton's classical theory, whilst insisting that the classical theory need not be rejected: it remains a paradigm. Although some variant of the "standard model" might one day unify quantum theory with Einstein's special relativity, work would still need to be done to incorporate the general theory. The overall message for philosophy appears to be that, despite radically differing approaches being taken, a deeper exploration can yield significant unifying benefits in the long run. This chapter concludes with some reflections on the philosophy of science, including references to the work of Thomas Kuhn.

We now arrive at **Chapter 4: The Given System of Sciences**. This chapter and the next are among the longest in the book, and as such, these chapters present Sørlander's primary arguments. Within this chapter, Sørlander takes a considered look at the development of mathematics and the physical sciences, all the way through natural science to the humanities, to generally provide evidence for the view that, at root, philosophy itself provides unifying conceptual patterns within the development of these areas. This is not to claim that these areas are themselves in any way similar in their content. On the contrary, Sørlander is at pains to say that the subject areas are, unsurprisingly, entirely distinctive in their content. Instead, Sørlander is stating that these areas all appear to share common conceptual basis typified by a focus on rational consideration and the need for evidence.

For example, in Section 1, the author writes: "Physics is a spectacular example of a scientifically specified conceptual system", and goes on to say that this also applies to "simpler" conceptual systems such as, logic, number theory and geometry. He seems to regard these concepts as a description of reality. In Section 2, Logic, he gives a brief cultural-historic analysis from classical Greek times as well as a summary of logical connectives, 'and', 'or', implication etc. that the Greeks formulated in their "first flowering". Nothing much happened for a long time after that until the axiomatisation of propositional logic in the nineteenth century. He records that the Greeks

in classical times also touched upon modal logic, with concepts of 'possible', 'necessary' and so forth. Finally in this Section he discusses the relationship between divalent logic with two truth values, 'true' and 'false', and polyvalent logics. The next few sections discuss the science of numbers, geometry and physics. The chapter finally embarks on Biology, Psychology, Linguistics, Social Science and [the study of] History. There is much that readers will find contentious in these sections, especially in sections 4.7, on Psychology, and 4.8, Linguistics. However, those rather contentious observations can stimulate readers to private thoughts and analyses which one might not otherwise have embarked upon.

In the very final Section 4.11, Philosophy and the System of Sciences, Sørlander starts by summarising the chapter so far. But then he draws some extraordinary conclusions. "... we must also conclude that this conceptual system must basically apply for all human languages. It follows since all languages in principle must be mutually translatable". This seems to be in direct contradiction to the Sapir-Whorf hypothesis, also known as *linguistic relativity*[2]. The author goes further: "all human beings should bow to the same logic. They should recognize the same arithmetic, geometry, physics, biology, and psychology. And they should recognize the same method in social science and in the study of history". The author makes no reference to the Sapir-Whorf hypothesis, despite implicitly condemning it, but strongly rejects almost any form of relativism. Overall, Sørlander is returning to the initial thesis posited in the Introduction and the beginning of Chapter 1.

Concluding this chapter, Sørlander says "Rationally speaking, we must thus state that we cannot justify any alternative to – and thus any way out of – this system of basic concepts for describing reality. But at the same time, we are confronted with a question that cannot be answered on the basis that we have used so far: the actual existence of the sciences. It is the question of why this system of basic concepts is as it is at all. Or: Why do we have precisely the system of sciences that we actually have? Could it not be otherwise?"

At this point, we now reach the conceptual culmination of this book, somewhat portentously titled: **Chapter 5: The Transcendental Deduction of the Conceptual Conditions for Any Possible Description of Reality**. It is indeed one of the two longest chapters in the book, the other being the final chapter. Not only is this chapter lengthy, but it is also somewhat challenging to read and comprehend. For instance, the key phrase, "transcendental deduction", occurs numerous times in this chapter, and was first mentioned in Chapter 2. However, despite its frequent occurrence here in this chapter, no real hint is given by Sørlander that the phrase "transcendental deduction" will likely be very familiar to students of Kant's *Critique* of Pure Reason. So the phrase "transcendental deduction" might reasonably be safely taken to mean "an absolutely undeniable and necessary inference from premise to conclusion".

From the Introduction, Sørlander's primary focus in the book has been with this question: "What is thus necessary is that it couldn't be different?". According to von Wright's foreword and also his own preface, Sørlander has generally tackled this question throughout his published work, starting

[2]See e.g., Whorf, Benjamin Lee (2012), *Language, thought, and reality : selected writings of Benjamin Lee Whorf*, John B. Carroll, Stephen C. Levinson, Penny Lee (2nd ed.), Cambridge, Mass.: The MIT Press, ISBN 978-0-262-51775-1; Sapir, Edward (1983), David G. Mandelbaum (ed.), *Selected Writings of Edward Sapir in Language, Culture, and Personality*, University of California Press.
Also: **https://en.wikipedia.org/wiki/Linguistic_relativity**.

in his first book [3]. Very broadly speaking, Sørlander explains the difference to Kant's original approach as being: "Where Kant here carries out a transcendental deduction of that system of categories, which must be presupposed by the possibility of self-awareness, I shall try to carry out a transcendental deduction of the system of basic concepts, which must be presupposed by every possible description of reality."

Sørlander claims to obtain this transcendental deduction since "when we have to find a basis which is deeper than that on which Kant builds, it must be the principle of contradiction. Then it can be nothing but the principle, that the truth of a proposition implies the falsity of the negation of the same proposition." Sørlander continues: "Here we must remember that the principle of contradiction does not stand entirely alone. It presupposes a theory of meaning. It presupposes that the designations with which propositions can be expressed must be defined by relations of implication to other designations." In this way, Sørlander tries to resolve the apparent dilemma posed by seeking a system of concepts for describing reality, but without also presupposing it to start with.

As one might expect, the author naturally makes much reference to Kant, but also to more modern writers, Zinkernagel (1988) and David Favrholdt (1999), although mainly to contradict their positions. There is a discussion of self-awareness and consciousness, in the context of Kant's following Descartes' starting point. However, in his Meditations on First Philosophy, not actually referenced or referred to by Sørlander, Descartes asserts that *Cogito ergo sum, Je pense, donc je suis*. Since I am aware of myself thinking, I must exist. Sørlander examines this and finds fault with its foundations. Accordingly, Sørlander contends that consciousness cannot be the foundation for a "transcendental deduction of what is so necessary that it could not be otherwise under any circumstances", as Kant claimed, since it amounts to a form of solipsistic self-belief, and moreover, such an empirical fact cannot be taken as a necessary consequence – because it could of course be contingent.

Sørlander further examines the logical basis of Kant's and Descartes' arguments in minute detail, cruising through "The Logic of Propositions" and "The Logical Framework for the Possibility of Propositions About Reality". Numbers, Time, Space, Geometry and Causality all receive the author's attention. The author rapidly embarks on a increasingly wide ranging series of somewhat technical studies throughout the remainder of this chapter.

One gets the impression of a surging exploration from a basis of enquiry into logical necessities to reach some questionable conclusions. FACS readers, especially those with an interest in philosophy, will find this chapter full of challenging interest, but not all will accord with its conclusions, or indeed, its argumentative process. It is worth noting that despite containing a subsection on Knowledge and Rationality, the author nowhere makes reference to Imre Lakatos, the champion of rationality, one might say [4, 5].

We finally arrive at the last two chapters, namely **Chapter 6: Pure Reason in Ethics** and **Chapter 7: Pure Reason in Politics**. In these chapters, Sørlander takes on the issue of human nature with all its profound difficulties, providing ample opportunity for commentary across a vast range of general topics, from ethics and social justice to religion and democracy. Broadly speaking, Sørlander's frequent target here lies with postmodernism and the incessant relativisation of what counts as truth. At this point, although much more could be said in our review, it is already

substantial in length, and therefore, time to close.

Overall, in this book and despite its relatively short length, Kai Sørlander has largely attempted a substantial review of Kant's metaphysical foundational approach, basing it not on self-conciousness but on a foundational rational understanding of the nature of truth (the principle of contradiction). That being said, it is certainly not for our review to claim how successful Sørlander has been in this endeavour – only time will tell. Finally, we cannot close here without again thanking Dines Bjørner for his extraordinary translation efforts of what must have been a thoroughly challenging project.

References

[1] Bjørner. D (2024). *Domain Science and Engineering*, Springer Nature, 2024 (See: Chapter 2 "Domain Philosophy", pp 11–18)

[2] Sørlander, K. (2022). Den rene fornufts struktur [*The structure of pure reason*]. Ellekær.

[3] Sørlander, K. (1994). Det Uomgængelige – Filosofiske Deduktioner [*The inevitable – Philosophical deductions*, with a foreword by Georg Henrik von Wright] (168 pp.). Munksgaard. Rosinante.

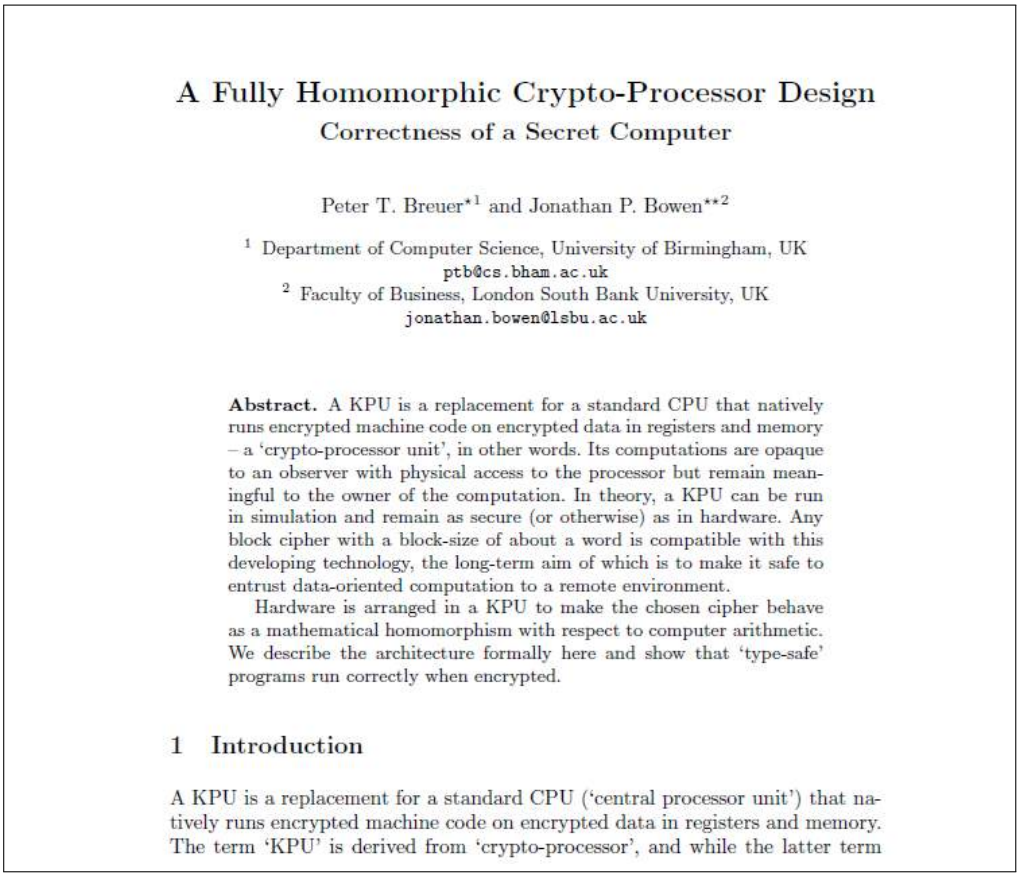
[4] Lakatos, I., (1976). *Proofs and Refutations*, Cambridge University Press.

[5] Lakatos, I., (1978). *The Methodology of Scientific Research Programmes*, Cambridge University Press

An interesting conference paper from 2013

(**Reviewed by:** Brian Monahan)

Quite by chance, I recently came across this old conference paper by Peter Breuer and Jonathan Bowen from early 2013. It was this:



A Fully Homomorphic Crypto-Processor Design
Correctness of a Secret Computer

Peter T. Breuer*[1] and Jonathan P. Bowen**[2]

[1] Department of Computer Science, University of Birmingham, UK
ptb@cs.bham.ac.uk
[2] Faculty of Business, London South Bank University, UK
jonathan.bowen@lsbu.ac.uk

Abstract. A KPU is a replacement for a standard CPU that natively runs encrypted machine code on encrypted data in registers and memory – a 'crypto-processor unit', in other words. Its computations are opaque to an observer with physical access to the processor but remain meaningful to the owner of the computation. In theory, a KPU can be run in simulation and remain as secure (or otherwise) as in hardware. Any block cipher with a block-size of about a word is compatible with this developing technology, the long-term aim of which is to make it safe to entrust data-oriented computation to a remote environment.

Hardware is arranged in a KPU to make the chosen cipher behave as a mathematical homomorphism with respect to computer arithmetic. We describe the architecture formally here and show that 'type-safe' programs run correctly when encrypted.

1 Introduction

A KPU is a replacement for a standard CPU ('central processor unit') that natively runs encrypted machine code on encrypted data in registers and memory. The term 'KPU' is derived from 'crypto-processor', and while the latter term

A fully homomorphic crypto-processor design: correctness of a secret computer
Peter T. Breuer, and Jonathan P. Bowen
Proc. of ESSoS'13, Paris, France, February 2013

I was unaware of this paper when it first came out in 2013 – at least. I don't recall having read it back then. If I had, I would almost certainly have taken serious note of it at the time.

Cloud computing and virtualisation

Back in the early 2000s, many industrial research groups were tackling all manner of projects dedicated to making the dream of cloud computing a reality. At that time, I was working as a researcher for HP Labs in Bristol, within a department calling itself "Trusted E-Services". Our

broad remit was to investigate and explore options for the future concerning cybersecurity and related areas.

One of the topics of general interest for HP in the mid 2000s was the issue of cloud computing and the question of how secure this is. Along with all other major players in the industry, HP wanted to emphasise the safety and security of their cloud offering to customers. Also around that time, commercial *virtualisation* hypervisor server platforms started to emerge beginning with **VMware**, followed later by **Hyper-V** from Microsoft and **VirtualBox**, firstly from Innotek, then Sun Microsystems and finally Oracle. Research tools and frameworks such as **QEMU**, **KVM** and the **Xen** hypervisor had also become fairly commonplace as well. Clearly, virtualisation was here to stay!

This approach was all very well for efficient data center deployment of cloud computing workflows, but was it also sufficiently secure for commercial deployment? The debate there mostly concerned issues such as privileged access to client memory from either the host or, potentially, from other clients. At the time, such concerns were largely resolved through consideration of the low-level security architecture of the processors in use at the time and support for memory isolation.

Nevertheless, a number of us still wondered if virtualisation itself could play more of a direct role in providing security assurances. Memory isolation is an essential capability provided by virtualisation and this clearly involves memory mapping. Perhaps mappings like that could be strengthened and enhanced in some manner to increase barriers to unauthorised access? This seemed hard to achieve reliably at the time – but one couldn't help wondering how virtualisation technology could enhance security for guest virtual machines (VMs) more fundamentally.

An elegant approach

It was quite a bit later when the Breuer and Bowen paper [1] showed how that enhanced security could plausibly be achieved. Without going into any specific technicalities here, the paper outlines a model hardware architecture in which not only the data but also the code is encoded in a key-dependent manner. This means that, in theory, only the owner of that code and data will be able to gain access to whatever is processed using such a platform. An elegant design point here is that only the Arithmetic Logic Unit needs to be changed to provide the desired behaviour, while keeping the remaining processor architecture conventional. The key result of the paper is to present a formal model of this system in order to demonstrate that conforming *type-safe* programs evaluate correctly, albeit computing over appropriately encoded data.

The tricky issue of deployment might plausibly be provided through virtualization technology such as **QEMU** etc. to emulate, albeit slowly, the hardware design in software, as alluded to in the original abstract at the time.

A second paper by Breuer and Bowen et al., [2] published three years later, refined this approach by providing an ALU coprocessor to operate alongside a more conventionally architected CPU. The objective achieved by this later study was to concretely examine the feasibility of this refinement and to validate practical performance and system integration capabilities.

Overall, it seems clear that formal methods played an important role here in initially establishing the coherence for the approach, demonstrating that the idea could work conceptually in principle. With that having been done, the second paper then had the licence to refine and reify the design to demonstrate practical feasibility. Naturally enough, the second paper probably could not have done that as effectively as it did without the first paper having already established the conceptual coherence of the approach.

A final thought – perhaps its worth acknowledging that no one should be too surprised that some way of devising a key-dependent computational model should be possible, particularly in the light of the Church-Turing thesis? After all, Turing himself formulated his own conceptual model of computation as a concrete idealisation of how people conduct computations on paper. This involves writing down symbols and so on, all of which are highly amenable to mapping and specific notations. Who is to say that any one set of symbols and notations have any particular status? All that matters is that those symbols and notations are used consistently in order to convey information. I can't help feeling that Turing might have strongly approved of the outcome of these papers.

References

[1] Breuer, P.T. and Bowen, J.P., (2013)
A fully homomorphic crypto-processor design: correctness of a secret computer, Proc. of the 5th International Conference on Engineering Secure Software and Systems (ESSoS'13), pages 123–138, Paris, France, Springer, doi:10.1007/978-3-642-36563-8_9, PDF

[2] Breuer, P.T., Bowen, J.P., Palomar, E. and Liu, Z., (2016)
A Practical Encrypted Microprocessor, In Proc. of the 13th International Joint Conference on e-Business and Telecommunications (ICETE 2016) - Volume 4: SECRYPT, pages 239–250, ISBN: 978-989-758-196-0, doi:10.5220/0005955902390250 PDF

Forthcoming Events

If you have suggestions for future FACS seminar speakers or other events, especially if you are willing to help with co-organisation or even give a talk, please contact Alvaro Miyazawa on **Alvaro.Miyazawa@york.ac.uk**.

Events Venue (unless otherwise specified):

BCS, The Chartered Institute for IT
Ground Floor, 25 Copthall Avenue, London, EC2R 7BP

The nearest tube station is Moorgate, but Bank and Liverpool Street are within walking distance as well. The new Elizabeth Line is now very convenient for the BCS London office, by alighting at the Liverpool Street stop and leaving via the Moorgate exit.

Date and Time	Title
December 2025 4:00pm–8:30pm	Hybrid: FACS AGM and Peter Landin Semantics Seminar Jonathan P. Bowen, Emeritus Professor of Computing, London South Bank University ***Talk:Formal Methods: Whence and Whither***

Details of all forthcoming events can be found online here:

https://www.bcs.org/membership/member-communities/facs-formal-aspects-of-computing-science-group/

Please revisit this site for updates as and when further events are confirmed.

FACS Committee

Formal Aspects of Computing Science Specialist Group

Jonathan Bowen
FACS Chair and BCS Liaison

John Cooke
Treasurer and Publications

Roger Carsley
Secretary and Vice-Treasurer

Alvaro Miyazawa
Seminar Organiser and Vice-Secretary

Margaret West
Inclusion Officer

Keith Lines
Vice-Chair

Tim Denvir
Newsletter Co-editor

Brian Monahan
Newsletter Co-editor

Andrei Popescu
LMS Liaison

Ana Cavalcanti
FME Liaison

FACS is always interested to hear from its members and keen to recruit additional helpers. Presently we have vacancies for officers to help with fund raising, to liaise with other specialist groups such as the Requirements Engineering group and the European Association for Theoretical Computer Science (EATCS), and to maintain the FACS website. If you are able to help, please contact the FACS Chair, Professor Jonathan Bowen at the contact points below:

BCS-FACS
c/o Professor Jonathan Bowen (Chair)
London South Bank University
Email: jonathan.bowen@lsbu.ac.uk
Web: http://www.jpbowen.com

You can also contact the other Committee members via this email address.

Mailing Lists

As well as the official BCS-FACS Specialist Group mailing list run by the BCS for FACS members, there are also two wider mailing lists on the Formal Aspects of Computer Science run by JISCmail.

The main list: **facs@jiscmail.ac.uk** can be used for relevant messages by any subscriber. An archive of messages is accessible under:

http://www.jiscmail.ac.uk/lists/facs.html

including facilities for subscribing and unsubscribing.

The additional list: **facs-event@jiscmail.ac.uk** is specifically for announcement of relevant events.

Similarly, an archive of announcements is accessible under:

http://www.jiscmail.ac.uk/lists/facs-events.html

including facilities for subscribing and unsubscribing.

BCS-FACS announcements are normally sent to these lists as appropriate, as well as the official BCS-FACS mailing list, to which BCS members can subscribe by officially joining FACS after logging onto the BCS website.