Statement of Purpose

George Pîrlea

I am interested in building trustworthy, provably-correct computer systems. In particular, I focus on formally verifying protocols for distributed consensus, communications security, and anonymity. More broadly, my research interests include formal methods, programming languages, systems, and cryptography.

My motivation for pursuing a Ph.D. comes from my experience developing formally verified systems during 3 summers spent as a research intern: in a university department, at a leading industry research lab, and at a publicly funded independent research institute. I chose my internships deliberately, primarily with the goal of building up my understanding of formal methods, but also aiming to experience different research cultures. During that time, I came to appreciate the benefits that formal methods and strong type systems can bring to real computer systems.

Working with Ilya Sergey in the programming languages and verification group at UCL, I developed a model of blockchain-based distributed consensus mechanised in the Coq proof assistant. Our goal was to put this nascent world of blockchains on a strong formal foundation, opening up the possibility of developing a verified implementation of a real blockchain protocol. Such an implementation is sorely needed, as demonstrated by the many critical vulnerabilities discovered in cryptocurrencies, most recently the Bitcoin inflation bug which allowed miners to double-spend [1] between September 2017 and September 2018.

We released our Coq formalisation as open source software and published our paper, *Mechanising Blockchain Consensus* [2], at CPP 2018. For my Master thesis, I extended this formal model by removing assumptions and instantiated the generic model with a SHA256-based Nakamoto consensus protocol. Finally, I extracted the instantiated protocol from Coq to OCaml, thus obtaining the first formally verified implementation of a Nakamoto-style blockchain consensus protocol. While far from being realistic or performant, our implementation opened up a new avenue for research that others are already pursuing [3]. I presented this work at NUS in March 2019.

I continued working on consensus protocols as an intern in the confidential computing group at Microsoft Research Cambridge. Supervised by Christoph Wintersteiger and using the F* proof assistant, I developed a formalisation of Coco, Microsoft's confidential computing framework based on Intel SGX enclaves. While building the formal model, I discovered a critical concurrency bug in the Coco C++ implementation that would have allowed any (untrusted) user of the system to delete the entire replicated log. Coco is now available for public use on Microsoft Azure.

Most recently, I was an intern in the foundations of programming group at the Max Planck Institute for Software Systems, working on the RustBelt project [4] under the supervision of Derek Dreyer and Ralf Jung. RustBelt's goal is to formally verify Rust's

safety claims, i.e. that Rust programs are type-safe, memory-safe, and data-race-free. Standard techniques to show type system safety do not apply, since Rust allows the programmer to mix safe and unsafe code (there are over 1000 `unsafe` blocks in the standard library alone). To account for this safe-unsafe interaction, RustBelt employs a semantic type system, λ_{Rust}, in which types have logical meaning rather than being purely syntactic symbols — this allows unsafe code blocks to be typed even though they do not satisfy any syntactic typing rules.

During my internship, I extended the λ_{Rust} type system to account for *pinning*, an addition to Rust that allows the programmer to state (at the type level) that the location in memory of certain values (which have the `Pin` type) cannot change. Pinning imposes a major change in λ_{Rust} — the semantic definition of a type now depends on where in memory values of that type are stored! This is non-standard and difficult to reason about. In fact, an unsoundness issue was found in the `Pin` library in November 2019, due to an interaction between Rust's trait system and pinning. While my formalisation did not catch this (RustBelt does not model traits), our insights are proving invaluable in informing the debate on how to solve the unsoundness issue [5].

Through my experience, I realised that research has an immense potential to impact the world — that is what keeps me excited and motivated to continue despite the frustrations of research. And I have been heartened by the small ways in which my work has been impactful already. But I am ready to embark on a larger, more serious undertaking than what can be achieved during an internship. That is why I am pursuing a Ph.D. I want to advance our understanding of how to build dependable systems and make formal verification mainstream. Longer term, I aspire to lead an industry research lab, putting academic research into common industrial practice.

I want to continue my studies at the National University of Singapore due to its faculty's broad work in formal methods, programming languages, and systems, and the department's environment of cross-disciplinary collaboration. I had the honour of visiting NUS for 3 weeks in spring 2019 and had a wonderful time interacting with faculty and students, getting feedback on my work and exchanging ideas. I believe my skills and interests make me a good fit for NUS, and I am excited by the opportunity to collaborate with the department's researchers. In particular, the interests of Ilya Sergey, Prateek Saxena, Haifeng Yu, Seth Gilbert, and Aquinas Hobor mesh well with mine.

References

[1] Bitcoin. *CVE-2018-17144 Full Disclosure*. Sept. 2018. URL: `https://bitcoincore.org/en/2018/09/20/notice/`.

[2] George Pîrlea and Ilya Sergey. "Mechanising Blockchain Consensus". In: *7th ACM SIGPLAN International Conference on Certified Programs and Proofs* (Jan. 2018).

[3] Faria Kalim et al. "Kaizen: Building a Performant Blockchain System Verified for Consensus and Integrity". In: *Formal Methods in Computer Aided Design (FMCAD) 2019* (Oct. 2019).

[4] Ralf Jung et al. "RustBelt: Securing the Foundations of the Rust Programming Language". In: *Proceedings of the ACM on Programming Languages* 2.POPL (Jan. 2018).

[5] *Unsoundness in `Pin`*. Nov. 2019. URL: `https://internals.rust-lang.org/t/unsoundness-in-pin/11311`.