Ryan Kavanagh
Computer Science Department
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213-3891
United States of America

January 26, 2021

Subject: Application for the position of assistant professor

Dear Members of the Search Committee:

I am writing to apply for the position of assistant professor. I am an "all but dissertation" PhD student in computer science at Carnegie Mellon University advised by Professors Stephen Brookes and Frank Pfenning. I am defending in August 2021. I believe that my extensive research experience on programming languages and my teaching experience make me an ideal candidate for this position. I am a Canadian citizen.

My goal is to develop tools and techniques that help practitioners correctly specify, implement, and reason about concurrent and communicating systems. Concurrent and communicating systems are ubiquitous, but they are hard to get right, leading to expensive bugs and vulnerabilities. Programming languages research has a long and successful history of providing tools and techniques for reasoning about these systems and ensuring their correctness. In my dissertation, I study techniques for reasoning about programming languages that combine functional programming with a rich form of message-passing concurrency. Though we know how to reason separately about functional programs and about message passing concurrency, the ways in which these features interact are complex, especially in the presence of non-termination and of code transmission. However, if we are to reason about real-world programs and languages, we must be able to reason about these interactions. My dissertation gives a novel and rigorous framework for doing so. Concretely, I give the first **denotational semantics** for a language with these features. Importantly, denotational semantics can be used to reason modularly about programs, instead of needing to reason about whole programs at once. I also give the first **observed communication semantics** for a language with non-termination. Observed communication semantics provide an intuitively reasonable notion of "observation" for languages with message passing concurrency, and they act as the ground truth when we talk about programs behaving "the same way". To put my contributions on stable mathematical foundations, I develop new techniques for reasoning about fixed points in category theory, and I develop the first notion of **fairness** for this style of language. Fairness ensures that schedulers do not neglect portions of parallel or concurrent programs.

I envision a world in which communicating systems are guaranteed to respect their communication protocols, and in which protocol violations are automatically detected. Protocols are agreed-upon rules for communication, and they are critical for system interoperability. Sometimes systems violate these rules, resulting in bugs like the Heartbleed vulnerability, which allowed attackers to access private data on an estimated 24–55% of websites. To prevent this, we can implement

communicating systems using programming languages with *session types*. Session types specify what kinds of messages can be sent or received, analogously to the way data types specify which kinds of data can be returned or used by programs. Crucially, programs written in session-typed programming languages are *guaranteed* to obey their communication protocols. Unfortunately, traditional session types are insufficiently rich to describe many real-world communication protocols. Over the next five years, I will investigate the theoretical foundations of session-typed languages to extend them to support real-world protocols, and I will develop compilers and tools to bring these theoretical developments into practice. I believe this work will lead to interesting collaborations with colleagues who work on programming languages and systems. In the longer term, I will develop a research pipeline that spans from rigorous foundational work to tools and techniques with industrial impact.

As an educator, I am dedicated to exciting students about computer science, and to teaching them to think critically about course material so that they may apply it in new settings. As a teaching assistant for two courses, I had the opportunity to prepare and give lectures, recitations, and assignments that helped students to actively engage with course material and to apply it to new and interesting contexts. In both cases, I incorporated material related to ongoing research. I am eager to teach existing courses, especially introductory courses or those related to programming languages, logic, concurrency, or theory. I am also eager to develop courses related to these topics. My commitment to education does not end in the classroom: I believe that students can significantly deepen their understanding of computer science through research. As a result, **I am excited to mentor undergraduate research**, e.g., through undergraduate student research awards, and by supervising undergraduate honours theses.

I am eager to contribute to your university's mission to produce high-calibre research and teaching. I am enclosing my CV, my research and teaching statements, and samples of my recent publications. Thank you very much for your consideration.

Sincerely,
Ryan Kavanagh