Challenges in Quantum Programming Languages

Peter Selinger
Dalhousie University, Halifax, Canada
selinger@mathstat.dal.ca
https://orcid.org/0000-0003-3161-856X

Abstract
In this talk, I will give an overview of some recent progress and current challenges in the design of quantum programming languages. Unlike classical programs, which can in principle be debugged by stopping the program at critical moments and examining the contents of variables, quantum programs are not amenable to traditional debugging because the state of a quantum system cannot usually be examined in a meaningful way. Therefore, we need other methods for ensuring the correctness of quantum programs, such as formal verification. For this reason, I advocate the use of strongly typed, functional programming languages for quantum computing. As far as functional quantum programming languages are concerned, there is currently a relatively wide gap between theory and practice. On the one hand, we have languages with strong theoretical foundations, such as the quantum lambda calculus, which operate at a relatively low level of abstraction and lack many features that would be useful to practical quantum programmers. On the other hand, we have practical functional quantum programming languages such as Quipper, which is implemented as an embedded language in Haskell, has many high-level features, and has been used in large-scale projects, but lacks a theoretical basis and a strong type system [1, 2, 3, 6]. We have recently attempted to narrow this gap through a family of languages called Proto-Quipper, which are designed to offer Quipper-like features while having sound theoretical foundations [5, 4]. I will give an overview of Quipper and its most useful features, report on the progress we made with formalizing fragments of Quipper, and outline several of the still remaining challenges.

2012 ACM Subject Classification Theory of computation → Quantum computation theory, Theory of computation → Program semantics

Keywords and phrases Quantum programming languages

Digital Object Identifier 10.4230/LIPIcs.FSCD.2018.3

Category Invited Talk

References
3:2 Challenges in Quantum Programming Languages

