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 [3, 1, 2, 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

1 Alexander Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. Quipper: a scalable quantum programming language. In *Proceedings of the 34th Annual ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2013, Seattle*, volume 48(6) of *ACM SIGPLAN Notices*, pages 333–342, 2013. Also available from `arXiv:1304.3390`. `doi:10.1145/2499370.2462177`.

2 Alexander Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. The Quipper language. Software implementation, available from `http://www.mathstat.dal.ca/~selinger/quipper/`, 2013.

3 Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. An introduction to quantum programming in quipper. In Gerhard W. Dueck and D. Michael Miller, editors, *Reversible Computation - 5th International Conference, RC 2013, Victoria, BC, Canada, July 4-5, 2013. Proceedings*, volume 7948 of *Lecture Notes in Computer Science*, pages 110–124. Springer, 2013. `doi:10.1007/978-3-642-38986-3_10`.

3rd International Conference on Formal Structures for Computation and Deduction (FSCD 2018).
Editor: Hélène Kirchner; Article No. 3; pp. 3:1–3:2
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

4 Francisco Rios and Peter Selinger. A categorical model for a quantum circuit description language. In Bob Coecke and Aleks Kissinger, editors, *Proceedings 14th International Conference on Quantum Physics and Logic, QPL 2017, Nijmegen, The Netherlands, 3-7 July 2017.*, volume 266 of *EPTCS*, pages 164–178, 2017. `doi:10.4204/EPTCS.266.11`.

5 Neil J. Ross. *Algebraic and Logical Methods in Quantum Computation.* PhD thesis, Department of Mathematics and Statistics, Dalhousie University, 2015. Available from `arXiv:1510.02198`.

6 Jonathan M. Smith, Neil J. Ross, Peter Selinger, and Benoît Valiron. Quipper: concrete resource estimation in quantum algorithms. Extended abstract for a talk given at the 12th International Workshop on Quantitative Aspects of Programming Languages and Systems, QAPL 2014, Grenoble. Available from `arXiv:1412.0625`, 2014.