Quipper: A Scalable Quantum Programming Language

Mehmet Talha Kocaer
536459
mehmet.talha.kocaer@tu-clausthal.de

July 17, 2023

Abstract

Quipper, a quantum programming language based on the functional programming paradigm, is adapted to efficiently implement and optimise large-scale quantum algorithms [1]. Quipper has demonstrated superior efficiency in a comparison study with another quantum language, QCL. This paper discusses Quipper's architecture, features, optimisation capabilities and future research directions.

1 Introduction

Quipper differs from existing quantum programming languages by expressing complex, large-scale quantum algorithms and enabling their effective and efficient optimisation [1].

2 Performance Comparison

The performance of Quipper is proved by the implementation of the Binary Source Tree algorithm and compared with its execution in QCL [1]. The results showed the superiority of Quipper, which requires fewer gates and half the number of qubits compared to QCL.

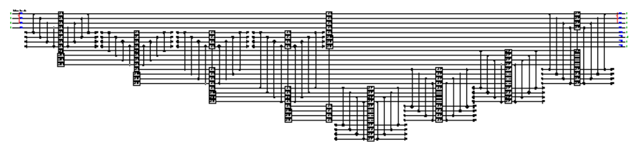

Figure 1: The circuit for o4 POW17

3 Scalability Test

Seven non-trivial quantum algorithms were implemented in Quipper to demonstrate its scalability [1]. This effort, involving 11 geographically dispersed programmers, resulted in functional representations suitable for realistic resource estimation.

4 Architecture and Capabilities

Quipper's architecture takes advantage of Haskell's type system to detect programming errors at compile time [1]. Other important capabilities include the automatic creation of quantum oracles via 'circuit removal' and the manipulation of all sub-circuits, making them user-friendly.

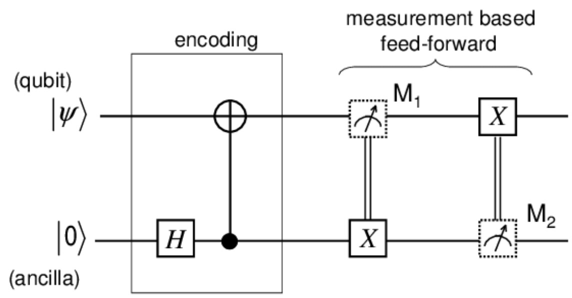

Figure 2: A visual representation of Quipper's architecture

5 Optimization

Quipper's optimisation capabilities are exemplified by the implementation of the Triangle Finding algorithm [1]. The algorithm was optimised by reducing the number of gates and qubits, and a significant advance in quantum computing was achieved.

6 Conclusion

As a quantum programming language, Quipper achieves scalability suitable for expressing complex, large-scale quantum algorithms and enabling their optimisation [1]. Future research aims to improve the functionality of Quipper, including further computational modelling, quantum error correction and integration with a quantum simulator.

References

[1] Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron. Quipper: A scalable quantum programming language. *CoRR*, abs/1304.3390, 2013.