BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2024
http://excel.fit.vutbr.cz

Symbolic Quantum Circuit Simulation

Sára Jobranová*

Abstract
Classical quantum circuit simulation is a vital tool for understanding the potential of quantum computation. This work introduces a novel approach to decision diagram-based quantum circuit simulation that significantly outperforms the current state of the art, for example for circuits implementing Grover's search algorithm.

*xjobra01@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Quantum computing is a very intriguing field of computer science that leverages the principles of quantum mechanics to perform computations in ways that classical computers are unable to. The potential of quantum computers has exciting implications for many different fields, such as physics [1], chemistry [2], and finance [3]. Tools that can efficiently simulate quantum circuits on classical computers are essential for future research in this field for two reasons.

The first is that quantum computers are still not readily available, mainly because of the price of building such a system. The second reason is that in a real system, we need to measure a qubit to make observations about its state, which leads to the collapse of the state of the qubit (this operation is irreversible). This means that it is not possible to directly examine the probability amplitudes of a real system, which can only be done in a simulation. However, classical quantum circuit simulation is not a trivial computational task due to the significant difference between the size of a qubit state space and a classical bit state space.

Today, there are several different approaches to the simulation of quantum circuits. These approaches differ mainly in the underlying data structure used (currently, the most popular are different variants of decision diagrams). However, the current state of the art still leaves a lot of room for improvement in terms of performance, especially when it comes to more complex circuits with a larger number of qubits.

This work presents the usage of symbolic execution to significantly speed up the simulation of circuits with loops. The proposed method has been implemented in the MTBDD-based quantum circuit simulator `MEDUSA`. Also, this work provides an experimental comparison with the current state of the art. It is shown that `MEDUSA` vastly outperforms the current state of the art for various quantum circuits.

2. Preliminaries

This section introduces the necessary basics of quantum computing and MTBDDs.

2.1 Quantum Computing

A qubit's *quantum state* $|\psi\rangle$ can generally be in a linear combination called a *superposition* of the *computational basis states* $|0\rangle$ and $|1\rangle$

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,,$$

where $\alpha, \beta \in \mathbb{C}$ are the *probability amplitudes* for the respective basis states. A single qubit's state is therefore a *two-dimensional complex vector* (sometimes called a *state vector*).

A generally n-qubit system's state $|\psi'\rangle$ can be in a superposition of all the system's computational basis states

$$|\psi'\rangle = \sum_{i \in \{0,1\}^n} \alpha_i \cdot |i\rangle\,,$$

and therefore is a 2^n-dimensional unit complex vector (again, $\alpha_i \in \mathbb{C}$ are the probability amplitudes of the corresponding basis states).

Quantum gates are used to alter the system's quantum state. They can be conveniently represented as unitary matrices (see Figure 1). The update of the system's quantum state is simply carried out as a matrix multiplication of the gate matrix with the state vector.

2.2 Decision Diagrams

A *reduced ordered binary decision diagram (ROBDD)*, simply referred to as a *BDD*, is a data structure that can be efficiently used for encoding Boolean functions as was suggested by Bryant [4].

Multi-terminal binary decision diagrams (MTBDDs) are a generalised variant of BDDs — the only difference is that MTBDD's terminals can have an arbitrary value. Because of that, MTBDDs can represent any function $f(v_1, \dots, v_n) : \{0,1\}^n \rightarrow \mathbb{D}$, for any $\mathbb{D} \neq \emptyset$ with finitely representable elements.

3. MTBDD-based Quantum Circuit Simulation

The classic representation of quantum state as a vector is not very convenient, as the data structures (the state vector and gate matrices) grow exponentially in size w.r.t. the number of qubits in the circuit. Instead, we view the system's state as a function $f\colon \{0,1\}^n \rightarrow \mathbb{C}$, where the evaluation of input variables corresponds to a computational basis state and the value of this function is then the corresponding probability amplitude (see Figure 2). We also use an exact algebraic representation of complex numbers (see Equation 1) proposed in [5].

The gate application is performed as a single custom *Apply* for single qubit gates and controlled phase gates. For other multi-qubit gates, this is not possible and a sequence of operations over the MTBDD using the standard *Apply* procedure is performed instead. Based on these methods, the simulator `MEDUSA` was implemented.

4. Proposed Symbolic Execution Extension

Symbolic execution consists of converting a classical representation into a symbolic representation, followed by symbolic simulation, and a final evaluation of all symbolic variables to convert back into the classical representation (see Figure 3).

This allows us to compute the *big-step semantics of loops* in the quantum circuit, which in turn leads to a significant acceleration of the calculation for circuits with loops with more than just a few iterations (there is no need to reevaluate the individual gates in each iteration). We represent the modification of the MTBDD caused by a single loop iteration with a symbolic update formulae in the form of a MTBDD. A second MTBDD is used to hold information about the value mapping of the classical MTBDD into the symbolic variables. Then `MEDUSA` computes the end result by repeated (corresponding to the number of iterations) substitution of the symbolic variables in this symbolic update MTBDD with the actual values of probability amplitude coefficients. This is particularly useful because loops are often a key element of quantum algorithms, e.g., algorithms that are based on amplitude amplification (Grover's algorithm) or phase estimation (Shor's algorithm).

5. Experimental Results

The experiments consisted of comparing both classical and symbolic modes of `MEDUSA` with the current state-of-the-art tools, namely decision diagram-based tools `SliQSim` [6], `MQT DDSIM` [7], `Quasimodo` [8], and CNF-based `Quokka#` [9]. Since `Quasimodo` supports multiple different decision diagram backends, all of them were tested separately. Benchmark circuits consist of circuits implementing Grover's algorithm, quantum counting (without inverse QFT), and period finding (also without inverse QFT). Both runtimes and peak memory usage (peak RSS) were compared.

As is shown in Figure 4, Figure 5 and Table 1 `MEDUSA`'s symbolic simulation significantly outperforms the state of the art for circuits implementing period finding and especially Grover's algorithm. Our symbolic simulation method still somewhat keeps up with the state-of-the-art tools in circuits implementing quantum counting (see Figure 6), however, the proposed approach could be easily extended for the best-performing `Quasimodo` backend options, which run faster for this type of circuit.

6. Conclusions

This work introduces a novel approach to quantum circuit simulation, which consists of combining symbolic execution with a regular decision diagram-based simulation method. In the conducted experiments, it is shown our approach greatly outperforms the state of the art when it comes to circuits implementing Grover's algorithm and quantum counting. Although some other state-of-the-art options performed better for circuits implementing period finding, our approach could be extended to these decision diagrams as well.

Acknowledgements

I would like to thank my supervisor Ing. Ondřej Lengál, Ph.D. for his time and guidance.

References

[1] Han-Sen Zhong, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020.

[2] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon C. Benjamin, and Xiao Yuan. Quantum computational chemistry. *Reviews of Modern Physics*, 92(1), March 2020.

[3] Dylan Herman, Cody Googin, Xiaoyuan Liu, Yue Sun, Alexey Galda, Ilya Safro, Marco Pistoia, and Yuri Alexeev. Quantum computing for finance. *Nature Reviews Physics*, 5(8):450–465, July 2023.

[4] Randal E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 8 1986.

[5] Philipp Niemann, Alwin Zulehner, Rolf Drechsler, and Robert Wille. Overcoming the trade-off between accuracy and compactness in decision diagrams for quantum computation. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, 39(12):4657–4668, 2020.

[6] Yuan-Hung Tsai, Jie-Hong R. Jiang, and Chiao-Shan Jhang. Bit-slicing the hilbert space: Scaling up accurate quantum circuit simulation. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*, pages 439–444, 2021.

[7] Alwin Zulehner and Robert Wille. Advanced simulation of quantum computations. *Trans. on CAD of Integrated Circuits and Systems*, 38(5):848–859, 2019.

[8] Meghana Sistla, Swarat Chaudhuri, and Thomas W. Reps. Symbolic quantum simulation with quasimodo. In Constantin Enea and Akash Lal, editors, *Computer Aided Verification - 35th International Conference, CAV 2023, Paris, France, July 17-22, 2023, Proceedings, Part III*, volume 13966 of *Lecture Notes in Computer Science*, pages 213–225. Springer, 2023.

[9] Jingyi Mei, Marcello M. Bonsangue, and Alfons Laarman. Simulating quantum circuits by model counting. *CoRR*, abs/2403.07197, 2024.