BRNO UNIVERSITY OF TECHNOLOGY FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2023
http://excel.fit.vutbr.cz

MTBDD-based Quantum Circuit Simulation

Sára Jobranová*

Abstract
Quantum circuit simulation is well-known as a very challenging computational task due to the amount of possible quantum states. This work's aim is therefore introducing a novel and efficient way of simulating quantum circuits and comparing it with the current state of the art.

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

1. Introduction

Quantum computation is a very intriguing field of computer science which, however, is still full of ambiguity. Recent research confirms the ability of quantum computers to effectively solve problems that would be practically unsolvable on classical computers, or the so-called *quantum supremacy* [1]. Even though the technology has now advanced enough to allow quantum computing, this hardware still has a largely experimental character. It is therefore useful to simulate the behavior of quantum circuits on classical computers as it is an important tool for revealing the true power of quantum computers. However, this is not a trivial task as one needs to accurately and compactly represent complex numbers (which are necessary for the simulation) and efficiently carry out the needed calculations.

Today, several such simulators exist, however, when discussing the current state of the art in terms of performance, there is still a lot of room for improvement when it comes to complex circuits or circuits with a large number of qubits.

The aim of this work is then to present the implementation of such a simulator. Our simulator is based on an algebraic complex number representation in combination with the use of MTBDDs (multi-terminal binary decision diagrams) and MTBDD operations for the circuit representation. Finally, this work also provides an experimental comparison with the current state of the art and shows that our implementation's performance is superior for some types of simulated circuits, especially nontrivial circuits implementing the Grover's search algorithm.

2. Preliminaries

2.1 Quantum circuits

Quantum state. A qubit's state $|\psi\rangle$ can be in 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. Because a single qubit's state is a two-dimensional unit complex vector, generally n-qubit system's state $|\psi'\rangle$ can be in a superposition of all the system's basis states

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

and therefore is generally a 2^n-dimensional unit complex vector (again, $a_i \in \mathbb{C}$).

Quantum gates are used to alter the system's quantum state. They can be conveniently represented as unitary matrices (e.g. Figure 4). The update of the system's quantum state is simply carried out as a matrix multiplication of the gate matrix (or its size-adjusted modification) 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 [2].

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 $f(v_1, \ldots, v_n) : \{0,1\}^n \to \mathbb{D}$, for any $\mathbb{D} \neq \varnothing$ with finitely representable elements.

3. MTBDD-based quantum circuit representation

The following features make it possible for the simulator to meet the required speed with the use of reasonable computing power.

3.1 Algebraic representation of complex numbers

This work utilizes the algebraic representation (see (1)) presented in [3] and [4]. Even though this method does not cover the entire set of complex numbers, the subset is sufficient for a quantum circuit simulation without loss of generality as it suffices for certain universal gate sets.

3.2 Quantum gate representation using MTBDDs

Representation of quantum gates using matrices is not very convenient, as one needs a $2^n \times 2^n$ matrix for an n-qubit system. It is more compact to represent the system's quantum state using an MTBDD with leaves containing an algebraic representation of the probability amplitudes and then apply the gate operations as update formulae for this MTBDD. For all gates except X, Y, Z, S and T, the universal update formulae presented in [4] are used (Table 1). These five gates can use the permutation-based update formulae — they are less computationally demanding, but can be used universally only with single qubit gates that modify the values of the probability amplitudes in such a way that the new values can always be obtained directly from an already existing value of some probability amplitude (or by permutation of its coefficients in the case of multiplication with ω).

4. Implementation

Our simulator `MEDUSA` is implemented in C to maximise performance. `MEDUSA` only supports circuits specified in OpenQASM (Open Quantum Assembly Language). The supported set of gates is identical to the gates whose MTBDD representation is defined in Table 1.

`MEDUSA` is built on top of the `Sylvan` library [5]. `Sylvan` is a parallel BDD library providing, amongst others, custom MTBDDs and MTBDD operations. This framework can be also used conveniently in combination with the `GMP` [6] library, which is needed due to the character of the MTBDD representation of quantum gates introduced in Table 1 as the values of the integers needed for the algebraic representation increase exponentially.

5. Experimental results

The experiments consisted of measuring runtimes for various quantum circuits and they were carried out in comparison with the the BDD-based state-of-the-art simulator `SliQSim` [3]. In all of the conducted tests, the benchmark circuits of the quantum circuit verification tool `AutoQ` [4] were used. These circuits can be divided into eight different sets of circuits, worth mentioning are especially circuits implementing Bernstein–Vazirani's algorithm (BV), Grover's search algorithm (both single and multi-oracle) and circuits implementing multi-controlled Toffoli gates (MCToffoli).

As can be seen from Figure 5, our implementation `MEDUSA` noticeably outperforms `SliQSim` in a large portion of benchmarks. The difference is most obvious when it comes to non-trivial circuits implementing Grover's search algorithm — it is clear that MTBDD-based implementation is exponentially better (the exact comparison of some runtimes can be found in Table 2). On the other hand, `SliQSim` is mostly superior to `MEDUSA` when it comes to non-trivial random quantum circuits.

6. Conclusions

This submission introduces a new method for quantum circuit simulation based on the use of MTBDDs. This is possible thanks to the aforementioned accurate algebraic representation of complex numbers and the possibility to represent quantum gates universally using just MTBDDs and MTBDD operations. In comparison with the state-of-the-art simulator `SliQSim`, it is shown that when it comes to non-trivial circuits implementing Grover's search algorithm, `MEDUSA` is exponentially faster than `SliQSim`. However, in the case of non-trivial random quantum circuits, `SliQSim` is superior. It is therefore obvious that in the matter of future work on this subject, there certainly still is some room for improvement in the implementation itself, both in terms of performance and functionality.

Acknowledgements

I would like to thank my supervisor Ing. Ondřej Lengál, Ph.D. for his time and guidance. Furthermore, I would like to express my thanks to Bc. Michal Hečko for his valuable help.

References

[1] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, et al.

Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, Oct 2019.

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

[3] 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.

[4] Yu-Fang Chen, Kai-Min Chung, Ondřej Lengál, Jyun-Ao Lin, Wei-Lun Tsai, and Di-De Yen. An automata-based framework for verification and bug hunting in quantum circuits (technical report). Technical report, 2023.

[5] Tom van Dijk and Jaco van de Pol. Sylvan: multi-core framework for decision diagrams. *International Journal on Software Tools for Technology Transfer*, 19(6):675–696, Nov 2017.

[6] Torbjörn Granlund and the GMP development team. *GNU MP: The GNU Multiple Precision Arithmetic Library*, 5.0.5 edition, 2012.