Aubree: A Practical Topological Quantum Computing Simulator

Victory Omole

5/29/2017

Abstract

Describe your paper in 100-200 words, give or take. The command-line `wc` utility is really useful here! This particular sample paper is meant to demonstrate a variety of LaTeX directives for producing a well-structured, consistently-formatted scholarly document. The actual content and outline may vary according to the needs of your specific research topic.

Topological computing is the most ingored model of computation of computing because uing quantum gates is extemely popular because most Quantum Computer Scientists come from a clasical computing background, but the topological computing model is actually more elegant and if non-ablelian anyons can be found, this method is actually less prone to errors than the super conducting qubit methods of computing.

State the problem: There is no open source satisfactory software framework in which i can play around and test how topological quantum computers work

My Approach:Write software to get an idea of how toplogical computing because inaccesible concepts are accecible once the code is written.

My solution: I wrote a program that could simulate a topological quantum computer and make it easy to study the mathematics and the braid theory that comes along with it.

Background: I have built a couple of toy virtual machines and compilers in the past but I have started studying quantum computing fairly recently.

Motivation: Toplogogical quantum computing is the most difficult and fault tolerant method of doing quantum comuting, so i feel like it is the most correct model

Contents

1   Introduction

The problem: There are barerly any implementations of topological qunatum computing in software, the only one I found was about using using surfaces codes for error correction.

Why is it interesting or important? It is interesting because if a topological computer that can simulate one qubit is built, It will be very easy to scale it to a limitless number of qubits. This is because topological quantum computers, unlike other method of quantum computing, is fault toloreant. This is important because researchers are making progress in finding materials that give topological phases in matter. The person who won last year's nobel prize basical found, So even though non-ableian anyons have not been found, if they are found the implecationns of the qunatum computing capabilities that can be build from it will be astounding.

Why is it hard? (E.g., why do naive approaches fail?) : Topological Quantum Computing is hard because the is inaccesible from my point, a researcher needs to know about knot theory, absract algebra, lie groups, and quantum computing. It is an inter-diciplinary field that require a research to know high-elevel mathematics, physics, and computer science at once. This program is so that this field could be just a little more accesible.

Why hasn't it been solved before? (Or, what's wrong with previous proposed solutions? How does mine differ?) There have been a couple of solutings such as the papers Introduction to topological quantum computing and John Peskill's Quantum Computing Lectures, But there are points in these guides that the subject matter can be a little too theoretical. It does not give the student a opportunity to get his/her hands dirty with how the actual system would be implemented.

What are the key components of my approach and results? Also include any specific limitations.

The limitations is that we are simulating a topological quantum computer with a classical computer and even if we use the IBM quantum computer, that is just 5 qubits. The main component is the interface where you can write a quantum program using quantum gates and it shows you how the anyons would be braided to make your program which is beautiful mathematically. There is a program called QTop that does it, but only respect to the error codes.

Summary of contributions: 1.Create a software the simulates the architecture of a topological quantum computer 2.Every simulation creates a PNG file that shows how the Anyons were braided to create a specific program. 3.The topological quantum computer can be program by using quantum gates, and these gates will then be compiled down to the braiding of the anyons. 4 The configuration of these anyons can be then simulated to IBM's quantum computer. This makes it so that it is possible for the topological quantum computer to be simulated using gates.

2   Background, Preliminary, and Related Work

2.1   Quantum Computing

Quantum comptuing contains Qubits which are in a Superposition of two States

2.2   Universal Quantum Computing

There are only a small number of Quantum Gates that need to be applied to Qubits to perform any computation that can be computed. Church-Turing theiss says that a universal model of computation can simulate another universal model of computation.

List of models of computation, Quantum Circuit, One-Way-Quantum Computer, Topological Quantum Computer

2.3 Topological Quantum Computing

This is a fault-tolerant way of building a quantum computer.

Column 1	Column 2	Column 3
a	b	c
d	e	f
g	h	i

Table 1: A sample table

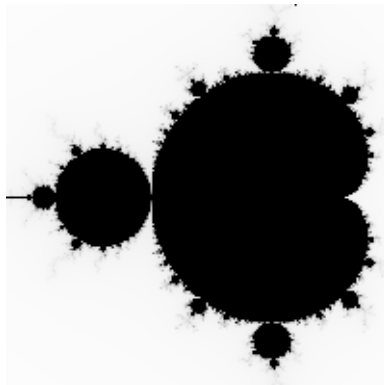Figure 1: A sample figure

3   Aubree by example

The outline after the introductory and background, preliminary, and related work sections is more dependent on the specific subject of your research. Remember to cite references where appropriate, organize the material so that it flows well and is clear to the reader.

3.1   Deutch Algorithm

LaTeX has support for up to three outline levels (`\section`, `\subsection`, and `\subsubsection`). It also recognizes `\paragraph` and `\subparagraph` directives, though those don't show up in the table of contents. All of these directives expect a title.

Note also the use of the `\verb` directive for inserting code-like labels or symbols. It was particularly needed here so that we can include the backslash character in the text.

3.2   Braid Generation

LaTeX has full support for tables and figures. Table 1 shows a sample table and Figure 1 shows a sample figure. Note the built-in support for captions and the automated numbering functionality. Lists of tables and figures can also be automatically generated, as seen at the beginning of this document.

One very important thing to remember about how LaTeX handles tables and figures by default: you don't have to worry about where they go exactly. The general rule is that you insert them in the source after your first reference to them, and LaTeX determines their final position. It also makes decisions on how much page space to devote to them. This all follows LaTeX's overall theme of focusing on the content of your paper, and not its format.

Just so you can see a second table, Table 2 is provided.

Column 1	Column 2	Column 3
a	b	c
d	e	f
g	h	i

Table 2: Another sample table

3.3 Optimizations

This is how the circuits are optimized with the braids

3.4 Simulating Aubree on IBM's quantum processor

These anyons can be theoretically simulated on IBM'S qunatum processor, so it gives a feel of how a topological quantum computer can be simulated using Quantum Gates

4 Conclusion

Wrap up your paper with an "executive summary" of the paper itself, reiterating its subject and its major points. If you want examples, just look at the conclusions from the literature.

5 Future Work

The biggest applications for Quantum Computing seem do be simulations for physics and chemistry. I could also have deep implications Material science. It would be greate if aubree contained a couple of the simulations algorithms. There should also be standard quantum algorithms like shors algorithm and grover's algorithm. Most researchers are not familiar with Lisp, so porting this program to a more accesible language like Python could be another opportunity.

6 Acknowledgements

I would like to thank Joesph Zambreno for guiding me in writing this report and proof-reading my work.

7 Appendix

I would like to thank Joesph Zambreno for guiding me in writing this report and proof-reading my work.

References

https://github.com/jacobmarks/QTop