An Introduction to Quantum Computing for Computer Scientists, with SAT

Francesco Piro

March 15, 2020

Abstract

This is the paper I have produced during my study of quantum computing for solving the SAT problem...

Introduction

Quantum computation is a wide area involving several disciplines that is having always more success in nowadays applications thanks in particular to the development of the technology and its incredible results. Quantum physics' principles are the fundamentals on which the entire theory is based: thanks to their properties, new architectures allow to define devices that can solve classical problems in surprisingly reduced time and space complexities. However we always have a trade-off to consider, in particular now that these technology are still emerging.

This paper aims at giving a description for computer scientists of what a quantum computer is and which are its real impacts and advantages with respect to the classical ones. Hence I will try to provide a description of the state of the art of quantum computing with an approach that allows to understand how from the basic principles of quantum mechanics we are able to have an algorithm that is faster with respect to its classic counterpart. In order to do so in the first chapter (1) I will start with the basic linear algebra needed to study the quantum physics' principles we need to define a quantum computer. The definition of the *quantum computer* is fundamental both to understand how and algorithm is executed but also to have a comparison with the classic Turing machine that allows us to determine conclusions on computational complexity (chapter 2). In the first chapter we will also have practical examples realized with the **qiskit library** in order to clarify also with some lines of code the concepts. Once the background on quantum computation and computational theory are well consolidated by the reader, the last chapters provide a practical example that is used to prove the speedup for the particular **satisfiability problem.** I have used an efficient classical solver for the $k\text{-}SAT$ as I could compare it with my quantum implementation, both realized in python. In the

end I provide some important conclusions for quantum computing that I was able to conclude thanks to my study, in particular in the papers listed in the references.

1 Quantum Computing

This chapter aims at providing first the fundamentals needed in order to deal with quantum mechanics and second at defining a quantum computer thanks to the principles previously identified. With the quantum device we will be able to make a comparison with the classical one to understand with examples how the basic operations are realized in order to use them to implement complete algorithms. Further in the chapter, a section is completely dedicated to the main quantum search algorithm that is fundamental to solve the SAT with a quantum algorithm. Also here we will start from a classical version to compare it with its quantum counterpart. All the arguments related in this chapter, together with the ones in the next (2) are fundamental to give a significant interpretation to the comparison between the classic and quantum implementation of the algorithm able to solve the **satisfiability problem** (2.1).

1.1 Fundamentals

The study of quantum computers requires the knowledge of the decimal and binary representation of integers, probability notions and in particular linear algebra fundamental definitions like the ones of: *vectors, spaces, bases, linear systems, tensor product...*. In this section are presented the basic concepts needed to face the quantum physics principles that we need in order to realize our quantum computer.

1.1.1 Linear Algebra

In order to realize a quantum computer we need understand how to define a state that is able to contain information that can be used to obtain a certain objective. As we will see in the next section, the state of a quantum device is a quantum state, thus a mathematical model that lives into a specific *vector state* whose dimension depends on the amount of information it needs to take care of. Typically, significant states contain information that results from the composition of several spaces combined together thanks to a particular operator called tensor product.

Definition (Tensor Product): *Given two vector spaces V and W over a field K with bases $e_1, ..., e_m$ and $f_1, ..., f_n$ respectively, the tensor product $V \otimes W$ is another vector space over K of dimension mn. The tensor product space is equipped with a bilinear operation $\otimes : V \times W \to V \otimes W$. The vector space $V \otimes W$ has basis $e_i \otimes f_j \forall i = 1, ..., m, j = 1, ..., n$.*

Typically we are going to work with complex Euclidean vector spaces of the form $\mathbb{C}^n$ and, by choosing the standard basis in the origin vector spaces, then the tensor product is nothing more than the Kronecker product.

Definition (Kronecker Product): *Given $A \in \mathbb{C}^{m \times n}$, $B \in \mathbb{C}^{p \times q}$, the Kronecker product $A \otimes B$ is the matrix $D \in \mathbb{C}^{mp \times nq}$ defined as:*

$$D = A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ a_{21}B & \cdots & a_{2n}B \\ \vdots & \vdots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

1.1.2   Quantum Mechanics

1.1.3   The Quantum Computer

1.2   Grover's Algorithm

2   Computational Theory

2.1   SAT Problem

3   SAT Implementation

3.1   Classical

3.2   Quantum

3.3   Classical vs. Quantum

4   Conclusions

References

[1] Qiskit textbook. `https://qiskit.org/textbook/`.

[2] Scott Aaronson. Np-complete roblems and physical reality. 2005.

[3] Stephen A. Cook. The complexity of theorem-proving procedures. 1971.

[4] V. Kreinovich E. Dantsin and A. Wolpert. On quantum versions of record-breaking algorithms for sat. 2005.

[5] Richard P. Feynman. Quantum mechanical computers. 1986.

[6] Lov K. Grover. A fast quantum mechanical algorithm for database search. 1996.

[7] T. Hertli and R. A. Moser. Improving ppsz for 3-sat using critical variables. 2018.

[8] S. Tamaki K. Makino and M. Yamamoto. Derandomizing hssw algorithm for 3-sat. 2011.

[9] K. Kutzkov and D.Scheder. *Using Constraint Satisfaction To Improve Deterministic 3-SAT*. 2018.

[10] Giacomo Nannicini. An introduction to quantum computing, without the physics. 2020.

[11] K. Y. Chwa O. Cheong and K. Park. *Algorithms and Computations*. Springler, 2010.

[12] M. Ohya and N.Masuda. Np problem in quantum algorithm. 2018.

[13] M. Ohya and Igor V. Volovich. New quantum algorithm for studying np-complete problems. 2018.