Comparing Grover's Quantum Search Algorithm with Classical Algorithm on Solving Satisfiability Problem

Runqian Wang

Princeton International School of Mathematics and Science

Princeton, NJ

ray.wang@prismsus.org

Abstract – The emergence of quantum computing provides us the possibility of solving tasks that might take years classically in just a few minutes. For certain problems, quantum computing exhibits quantum supremacy, meaning that the quantum solution runs exponentially faster than classical algorithms and is able to completely take over classical computers. This high efficiency of quantum computing comes not only from the hardware but also the software, quantum algorithms. The algorithms utilize the qubits to make calculations in order to fulfill specific tasks with the lowest time complexity possible. One such algorithm is named the Grover's algorithm, which is able to perform database search in O(sqrt(N)), and it runs much faster than the traditional algorithm that takes O(N) time. For example, when the task is to find the even integers from N integers, traditional computation will need to run through all of the N integers one by one, making at least N steps of calculation, while by using Grover’s algorithm only around sqrt(N) calculations are needed. This exponential speed-up makes Grover's algorithm one of the most important quantum algorithms. Grover's algorithm has a wide application in many fields and is able to improve the time complexity exponentially. One area that can be solved using Grover's algorithm is the satisfiability problem. This type of problem asks the computer to find a set of values (commonly true or false) for several variables such that they satisfy certain constraints. This fundamental problem can be applied in areas including combinatorial equivalence checking, automatic test pattern generation, model checking, AI planning, and haplotyping. We use k-SAT problems to refer to satisfiability problems with k boolean variables to be determined. Grover’s algorithm can effectively solve the k-SAT problem by performing the database search on 2^N possible states of the variables. The algorithm’s square root optimization on searching helps to improve the efficiency of this solution significantly. Furthermore, this optimization of Grover's algorithm may play a more important role when k grows larger, and consequently the efficiency of the quantum solution could improve faster relative to the traditional solution. Yet this hypothesis is never tested due to the lack of a general k-SAT quantum algorithm. No quantum algorithms solving k-SAT problems where k is greater than 3 have been proposed, thus no test has been performed to compare the quantum solution and the classical solution on more general k-SAT problems. In this research, we formulate a general quantum solution for k-SAT problem and compare such solution with the best classical algorithm to determine whether and when the quantum algorithm performs better on satisfiability problems. The comparison will be done through both theoretical deduction as well as real-world implementation. At the end of this research, we will determine whether the proposed quantum algorithm outperforms the classical algorithm on solving k-satisfiability problems.

Keywords – quantum computing; Grover’s Algorithm; satisfiability problem; database search;

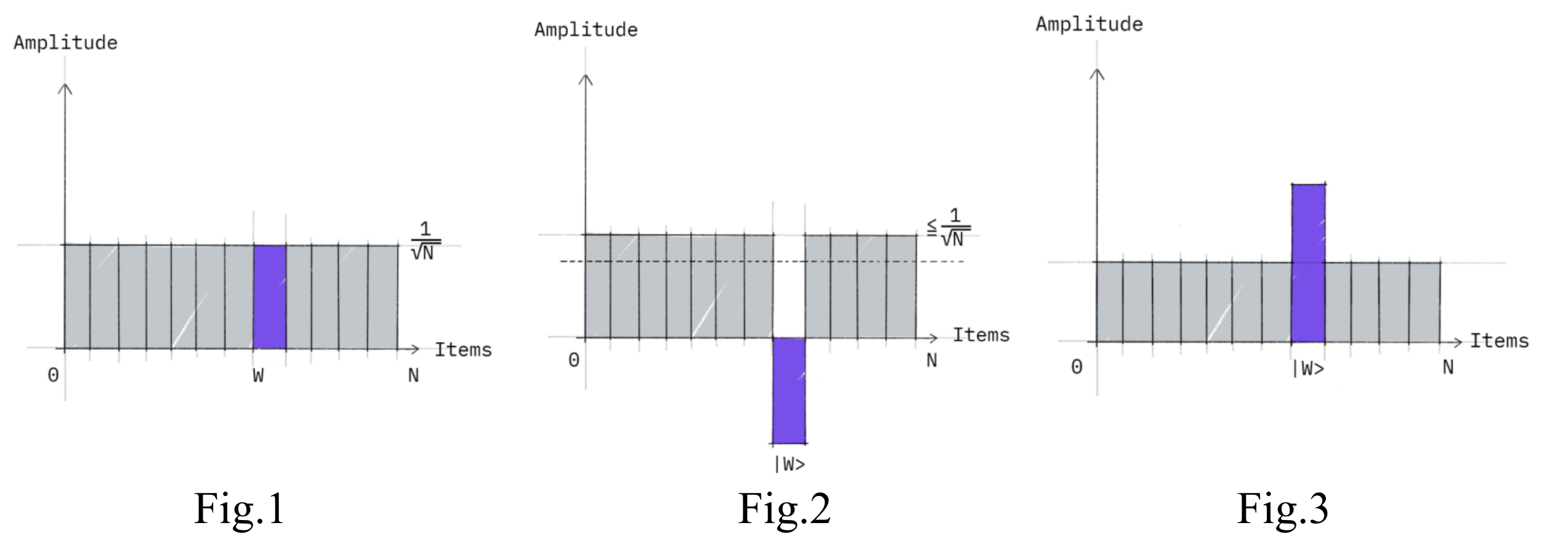
# Introduction

Solving classically impossible problems in just a few minutes, quantum computers can have substantial improvements over traditional computers on certain tasks [9]. However, there are not many quantum algorithms that can provide such improvement. One of the few useful quantum algorithms is Grover's algorithm [8], which already shows promising improvements in database searching and related problems. Yet the potential of this algorithm on solving satisfiability problems is not proven. Although there is a quantum solution for 3-SAT using Grover’s algorithm, it does not show any improvements as compared with the traditional solution [3][4][5]. That being said, no tests have been performed for k-SAT problems where k is greater than 3 due to the lack of a general quantum solution for SAT problem. It is possible that this optimization of Grover's algorithm may play a more important role when k grows larger, and consequently the efficiency of the quantum solution could improve faster relative to the traditional solution. This research aims to formulate a general quantum solution for the k-SAT problem and compare such a solution with the best classical algorithm to determine whether and when the quantum algorithm performs better on satisfiability problems.

The satisfiability problem asks the computer to find a set of values (commonly true or false) for several variables such that they satisfy certain constraints. K-SAT problems refer to satisfiability problems with k boolean variables to be determined. On solving the satisfiability problem, the only quantum algorithm proposed previously is focused on the 3-SAT problem and does not provide a general solution[3][4]. This proposed quantum solution utilizes Grover’s algorithm searching ability and traverse through all possible answer space, amplifying the correct states that satisfy the given constraints. This solution, however, runs even slower than the optimal 3-SAT classical solution[5]. It has a time complexity of O(1.414ᴺ) as compared with the optimal classical solution with time complexity of O(1.307ᴺ), where N is the number of boolean variables.

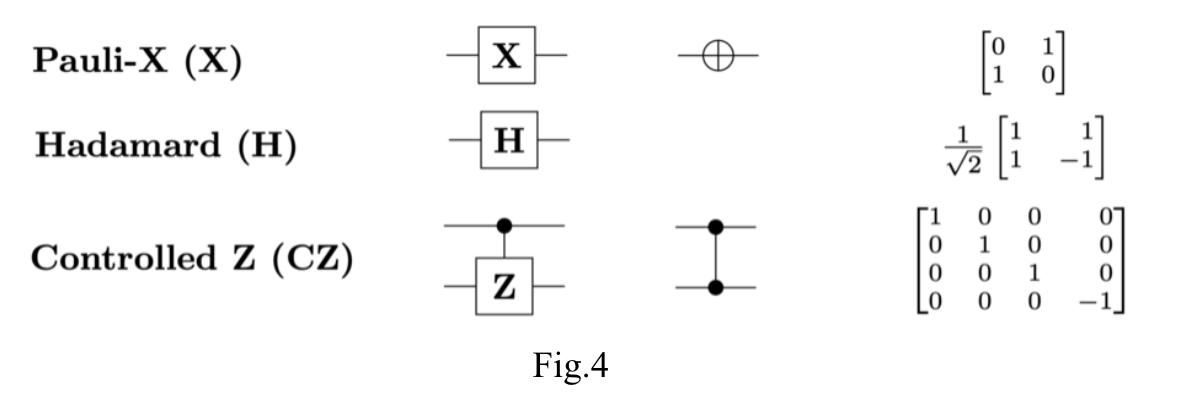
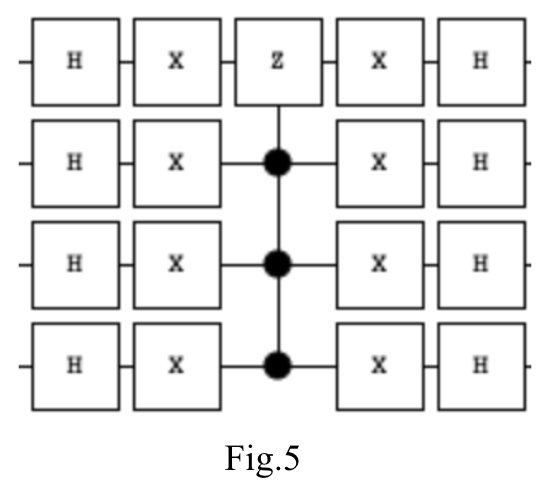
# Grover’s Algorithm

Grover’s Algorithm [8] can perform database search for specific targets quickly by using two major steps: reflection and amplification. The algorithm works as follows: first, the probabilities for all states are set to be equivalent, as shown in Fig.1; then, the reflection is performed, which reflects the amplitude of the desired states, as shown in Fig.2; afterward, the negative states are reflected across the probability of the original amplitude and is thus magnified, as shown in Fig.3. It can be proven that by repeating the last two steps, the amplitude will be magnified large enough after √N rounds. Notation-wise, the reflection function is defined as the Oracle function (Uf), which can identify those desired answers; and the amplification function is defined as the Us function.



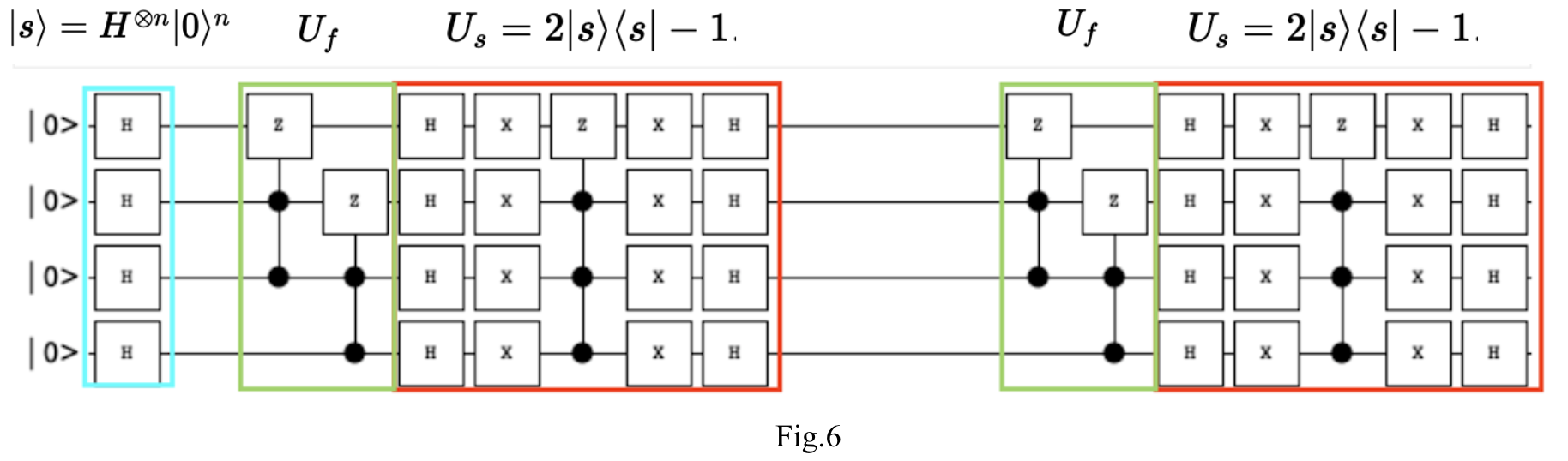
# Proposed Us Circuit

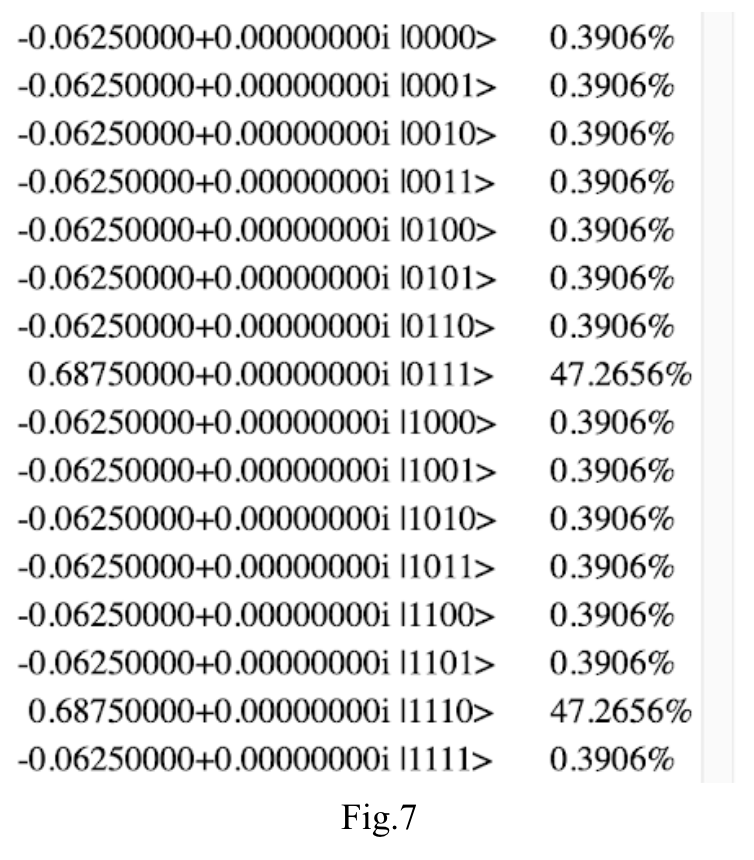
The Us function has been successfully designed and implemented in the quantum simulator. The three gates used in this section are the Pauli-X gate, Hadamard gate, and Controlled Z gate; the matrix representation of these gates is shown in Fig. 4. The example circuit for 4-SAT is shown in Fig.5. This 4-SAT example can be easily generalized to k-SAT by adding layers of {H, X, CZ, X, H} gates.

# Testing

To test the proposed Us circuit, it is placed in an example 4-SAT problem where the requirement is that either the first three bits are all true or the last three bits are all true. The Uf function is reversely engineered, which is easy given that the answers are “1110” and “0111”. Then the cycle is copied twice to guarantee accuracy. The completed circuit is shown in Fig.6 and the results are shown in Fig.7. In Fig.6, the initialization step, a straightforward H gate implementation, is marked by the blue circle; the reverse-engineered Uf Oracle function is marked by the green circle, and the proposed amplification circuit is marked by the red circle. As shown in Fig.7, the circuit successfully incremented the probability of the two correct answers to in total 94.53%, which is satisfactory.





# Future Work

An Us circuit is not the full solution; an Oracle function is still required. However, due to time limitations, only the processes mentioned above are finished. In the coming dates, this procedure will be followed to continue the work: discover a general way of making the Oracle function; test the entire proposed quantum algorithm on a quantum simulator; test the proposed algorithm on a real IBM quantum computer; compare it with the optimal classic algorithm; summarize and analyze the result.

##### References

[1] D. Monroe, "Bouncing Balls and Quantum Computing," ''Communications of the ACM'', vol. 63, no. 10, Oct., pp. 10-12, 2020.

[2] Srinivasan K, Satyajit S, Behera BK, Panigrahi PK. Efficient quantum algorithm for solving travelling salesman problem: An IBM quantum experience. arXiv preprint arXiv:1805.10928. 2018 May 28.

[3] Fernandes, Diogo, and Inês Dutra. "Using Grover's search quantum algorithm to solve Boolean satisfiability problems: Part I." XRDS: Crossroads, The ACM Magazine for Students 26.1 (2019): 64-66.

[4] Fernandes, Diogo, Carla Silva, and Inês Dutra. "Using Grover's search quantum algorithm to solve Boolean satisfiability problems, part 2." XRDS: Crossroads, The ACM Magazine for Students 26.2 (2019): 68-71.

[5] Hansen, Thomas Dueholm, et al. "Faster k-sat algorithms using biased-ppsz." Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing. 2019.

[6] Ambainis, Andris, et al. "Quantum speedups for exponential-time dynamic programming algorithms." Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms. Society for Industrial and Applied Mathematics, 2019.

[7] Srinivasan, Karthik, Bikash K. Behera, and Prasanta K. Panigrahi. "Solving linear systems of equations by gaussian elimination method using Grover’s search algorithm: an IBM quantum experience." arXiv preprint arXiv:1801.00778 (2018).

[8] Kwiat, P. G., et al. "Grover's search algorithm: an optical approach." Journal of Modern Optics 47.2-3 (2000): 257-266.

[9] Ashley Montanaro, "Quantum algorithms: an overview," Nature, Jan. 12, 2016. [Online]. Available: https://www.nature.com/articles/npjqi201523. [Accessed Oct.7, 2020]