3-SAT Problem for RNA Codon Search in QISKIT Framework

Ruchipas Bavontaweepanya

Department of Physics, Faculty of Science, Rangsit University, Pathum Thani, 12000
Corresponding author, E-mail: ruchipas.b@rsu.ac.th

Abstract

The SAT problem has been widely used in different research areas, including database search. Many modern classical algorithms have been developed to obtain the target from the database. For an unstructured database, any classical algorithms will perform $O(N)$ steps to achieve the solutions. However, the quantum algorithm, called Grover's algorithm, can speed up search process quadratically. In this work, the trial SAT problem with 3 variables was developed to examine the RNA sequence search in Grover's search algorithm. The positions of RNA sequence were represented by the three-qubit state, which has 8 combinations of all possible states. The satisfying solutions can be achieved by implementing Grover's search algorithm in QISKIT quantum framework. The output states have 3 possible solutions, which are $|000>$, $|011>$ and $|101>$ with nearly equal probability. These output states correspond to the 3 families of RNA codon.

Keywords**:** *3-SAT problem, Grover's algorithm, Qubit, Quantum oracle, Unstructured database, Quantum algorithm*

1. Introduction

The satisfiability (SAT) problem is to determine the satisfying set of assignments that make the Boolean expression true. The expressions are usually presented in Conjunctive Normal Form (CNF), which is a set of clauses. In other words, the given Boolean expression requires the variables that can be replaced by TRUE or FALSE in such a way that the expression evaluates to TRUE. The SAT problem is widely applied to many research areas, such as machine learning (Belahcène, et al., 2018), biological process (Liu & Wang, 2010), transportation (Pellegrini, Marlière, & Rodriguez, 2017), bioinformatics (Yang & Yang, 2005) and so on.

Many research areas need the SAT problem to solve the problem of interest in the real situation. Many solvers have been introduced with various methods (Prestwich, 2003). However, when the problems become more complicated, the time for solving the problem is an important question. Many modern methods were introduced to solve the computing time problem (Jonsson, Lagerkvist, Nordh, & Zanuttini, 2017; Prestwich, 2003; Zaikin, 2017).

For the unstructured database containing N records, the SAT problem can also be employed. In order to find satisfying solutions, any classical algorithms will perform $O(N)$ steps to achieve the solutions. However, the quantum algorithms can perform $O(\sqrt{N})$ steps to obtain satisfying solutions. This algorithm, called Grover's algorithm, can speed up the computing time quadratically (Grover, 1996). Since a quantum system can be in multiple states simultaneously and perform multiple tasks at the same time, the target states or solutions in an unstructured list can be achieved quadratically over classical algorithms (Grover, 1996; Luan, Wang, & Liu, 2012). In this work, the trial SAT problem with 3 variables has been developed for RNA sequence search and evaluated with Grover's algorithm in QISKIT quantum framework. The variables in the 3-SAT problem relate to the positions of the RNA sequence. In the implementation of Grover's algorithm, these variables were represented by a qubit, which is a quantum state.

2. Objectives

This research is to propose the framework for RNA codon searching with 3-SAT database. This work can provide the framework for protein search in bioinformatics research and can develop this framework to advance research. The Grover's algorithm is the main tool to search the RNA codon corresponding to the prepared 3-SAT. Another objective is to study the properties of the output states from Grover's search algorithm in QISKIT.

3. Materials and Methods

3.1 Qubit Representation

A qubit or quantum bit is a quantum state that can carry information of classical bits simultaneously. As the classical bit is a fundamental unit of information in a classical computer, the qubit is a fundamental unit of information in a quantum computer.

While a classical bit can have value 0 or 1, which is a binary digit, a qubit can hold both 0 and 1 simultaneously, called quantum superposition. The conventional notation of qubit used to describe the quantum states is $|\psi>$. The classical bit 0 and 1 can be encoded in qubit as

$$|\psi> = c_1|0> + c_2|1>.$$

The c_1 and c_2 are probability coefficients satisfying the normalization condition $|c_1|^2 + |c_2|^2 = 1$. The $|c_1|^2$ is the probability that the outcome state is $|0>$, and the $|c_1|^2$ is the probability that the outcome state is $|1>$. The $|\psi>$ represents a single-qubit state.

3.2 Single-qubit Operation

Quantum computers can process the qubits by applying a set of quantum operators to the qubit state. This notion is similar to the electronic gate in a classical computer. In a quantum computer, the allowed operators to perform on a qubit must be unitary operators, i.e., the operator can be reversible. The quantum gates, unlike classical logic gates, have no finite set of primitive quantum operations. For example, the Hadamard gate *H*, which is a well-known gate in quantum computing, is defined by

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

The Hadamard gate can transform the state $|0>$ and $|1>$ to the superposition states with an equal weight of the basis state as given

$$H|0> = \frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$$
$$H|1> = \frac{1}{\sqrt{2}}|0> - \frac{1}{\sqrt{2}}|1>.$$

However, if the Hadamard gate is applied twice to the state $|0>$ and $|1>$, the final state will become the initial state

$$H\left(\frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>\right) = |0>$$
$$H\left(\frac{1}{\sqrt{2}}|0> - \frac{1}{\sqrt{2}}|1>\right) = |1>.$$

This means the Hadamard gate is a unitary operator, and it can recover the qubits to the initial states.

3.3 Multiple Qubits

In order to prepare the qubit state for solving the 3-SAT problem, one need three-qubit states. The quantum state of three qubits can be formed by the tensor products of three single-qubit states,

$$|\Psi> = |\psi_1> \otimes |\psi_2> \otimes |\psi_3>.$$

The explicit general form of three-qubit states is

$$|\Psi> = d_1|000> + d_2|001> + d_3|010> + d_4|011>$$
$$+ d_5|100> + d_6|101> + d_7|110> + d_8|111>,$$

where the coefficient d_{1-8} satisfy the normalization condition

$$|d_1|^2 + |d_2|^2 + |d_3|^2 + |d_4|^2 + |d_5|^2 + |d_6|^2 + |d_7|^2 + |d_8|^2 = 1.$$

The initial three-qubit state for solving the 3-SAT problem was prepared with equal weight superposition basis state,

$$|\Psi> = \frac{1}{2\sqrt{2}}|000> + \frac{1}{2\sqrt{2}}|001> + \frac{1}{2\sqrt{2}}|010> + \frac{1}{2\sqrt{2}}|011>$$
$$+ \frac{1}{2\sqrt{2}}|100> + \frac{1}{2\sqrt{2}}|101> + \frac{1}{2\sqrt{2}}|110> + \frac{1}{2\sqrt{2}}|111>.$$

This qubit state can be used to search for the solutions of the 3-SAT problem.

3.4 Grover's Algorithm

3.4.1 Unstructured Search

Grover's algorithm is the main tool in this work to search on the unstructured database. The unstructured search is searching on unsorted data. For example, searching on a DNA sequence is an application of unstructured search. The unsorted DNA sequence can be described by the unsorted list.

A	G	T	C	T	A	G	T	C	C

Figure 1 Unsorted DNA sequence

To find the base T - the red item - using the classical algorithm, one would need to search at the minimum of $\frac{N}{2} = 5$ iterations on this sequence. However, Grover's algorithm can search the target items in $\sqrt{2}$ steps. A quadratic speedup substantially saves evaluation time for searching the target item in unsorted long lists.

3.4.2 Quantum Oracles

A quantum oracle is a set of operators that are used to operate on the initial qubits to flip the sign of the target stages and amplify the amplitude of the target states at the amplification stage. The oracle encodes such a list in term of function $f(x)$, which is defined by

$$f(x) = \begin{cases} 1 & when\ x = x^* \\ 0 & when\ x \neq x^* \end{cases}$$

when x^* is a target item. When the quantum oracle operated on the prepared qubits, the output state will be measured. The process of Grover's search can be demonstrated by

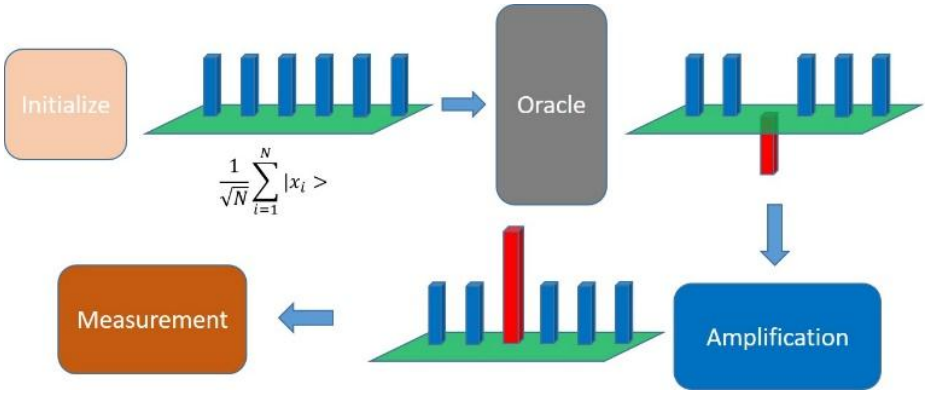

Figure 2 Grover's search algorithm process

3.5 3-SAT Problem

The satisfiability problem or SAT problem is the problem that consists of a logical expression in n variables and the requirement to find the Boolean values for each variable to make the expression true. For example, the given Boolean expression is

$$x_1 \wedge (x_2 \vee x_3) ,$$

where the $\wedge$ is AND operator, and the $\vee$ is OR operator. The SAT problem is to find the values of x_1, x_2 and x_3 that make the expression true. The choices can be $x_1 = TRUE,\ x_2 = TRUE,\ x_3 = TRUE$ or $x_1 = TRUE,\ x_2 = TRUE,\ x_3 = FALSE$ or $x_1 = TRUE,\ x_2 = FALSE,\ x_3 = TRUE$ or $x_1 = FALSE,\ x_2 = FALSE,\ x_3 = TRUE$, and so on.

The 3-SAT problem is a special case of SAT problem, where the Boolean expression should be divided into clauses such that each clause contains three variables. For example,

$$(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee -x_2 \vee x_3)$$

is the Boolean expression in 3-SAT form, which has 2 clauses and 3 variables. The " $-$ " is NOT operator, and the " $\wedge$ " and " $\vee$ " are AND and OR operator, respectively. The assignment of each variable becomes more complicated.

In this work, we will examine the simple framework to search on the RNA sequence which is composed of the nucleotide bases A, U, C and G. These four bases can form a sequence of nucleotide triplet, called codon, that will be translated to an amino acid. The target codon has to satisfy the following 3-SAT conditions,

$$(-x_1 \vee -x_2 \vee -x_3) \wedge (-x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee -x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee -x_3) \wedge (x_1 \vee -x_2 \vee -x_3) \qquad (1)$$

as an expression for searching the solutions by Grover's algorithm. This expression contains 5 clauses, which have 1 clause for all NOT values, 1 clause for two NOT values, and 3 clauses for one NOT values. The variable x_i determine the position of the base "U". The first condition means we would like to search for the codon that has no base "U" in all position in the sequence. The second, third, and fourth conditions mean the codon must have one position that cannot contain the base "U". The last condition means the 2nd and 3rd positions in the codon cannot contain the base "U" simultaneously.

4. Results and Discussion

The given Boolean expression (1) was implemented with QISKIT, which is an open-source quantum framework. The initial three-qubit state

$$|\Psi> = \frac{1}{2\sqrt{2}}|000> + \frac{1}{2\sqrt{2}}|001> + \frac{1}{2\sqrt{2}}|010> + \frac{1}{2\sqrt{2}}|011>$$

$$+ \frac{1}{2\sqrt{2}}|100> + \frac{1}{2\sqrt{2}}|101> + \frac{1}{2\sqrt{2}}|110> + \frac{1}{2\sqrt{2}}|111>$$

was employed to search for the solution by Grover's search algorithm. The 1st, 2nd and 3rd digit in the qubit represent the 3rd, 2nd and 1st variable, respectively. State 1 refers to TRUE, and state 0 refers to FALSE. The number of iterations was varied from 100, 500, 1000, 1500, and 5000 iterations.

At 100 iterations, the output from Grover's search reveals 3 candidates for the solution, but the state $|000>$ is the winner with probability 0.38, as shown in Figure 3, and the others would be suppressed. This means that the Boolean expression (1) has multiple solutions.

However, there is a competition between 3 candidates. This is the clue to increase the number of iterations to study the effect of the number of iterations on the output state. When we implemented Grover's search with 500 iterations, the probability of $|000>$ became 0.348, but the probability of $|011>$ and $|101>$ increased to 0.310 and 0.304, respectively, as shown in Figure 4. The probability of all 3 candidates became roughly the same at 1000 iterations, and the state $|011>$ became the winner at 1500 iterations, as shown in Figure 5 and Figure 6. When we implemented Grover's algorithm at 5000 iterations, the winner became the state $|000>$, as shown in Figure 7. This competition of 3 candidate states happened when there are multiple solutions for the Boolean expression and the probabilities of each candidate approach to the uniform probability.

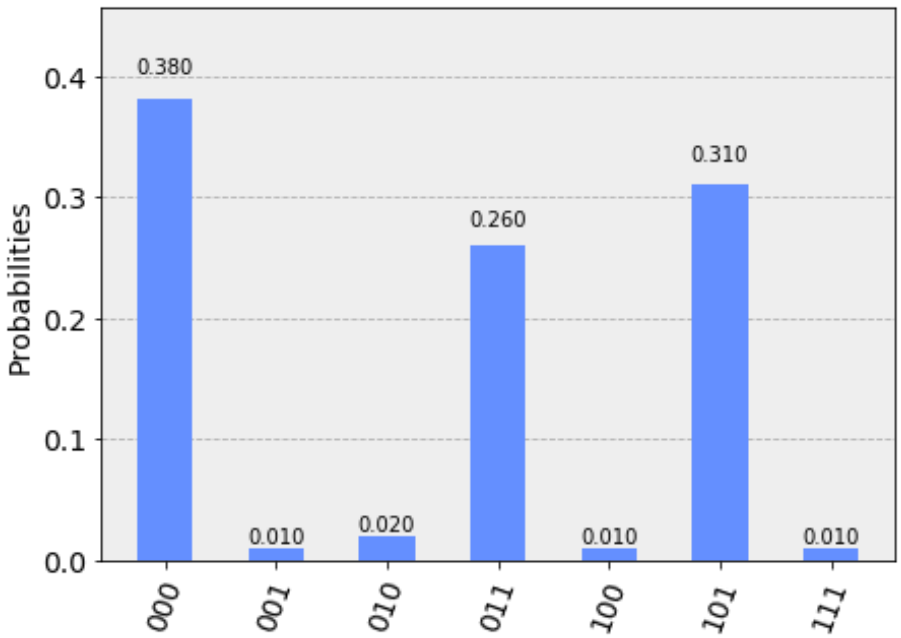

Figure 3 The output states of 100 iterations

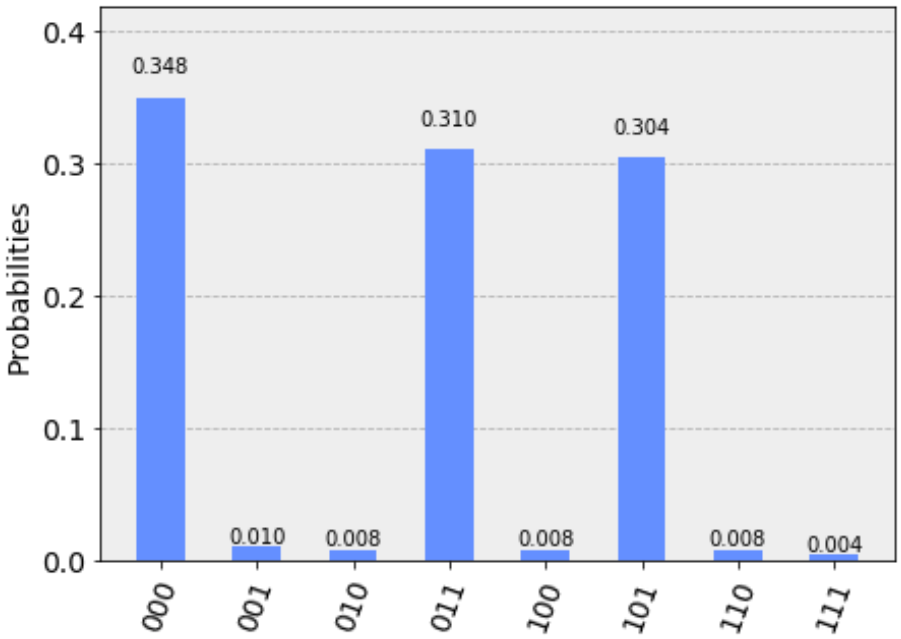

Figure 4 The output states of 500 iterations

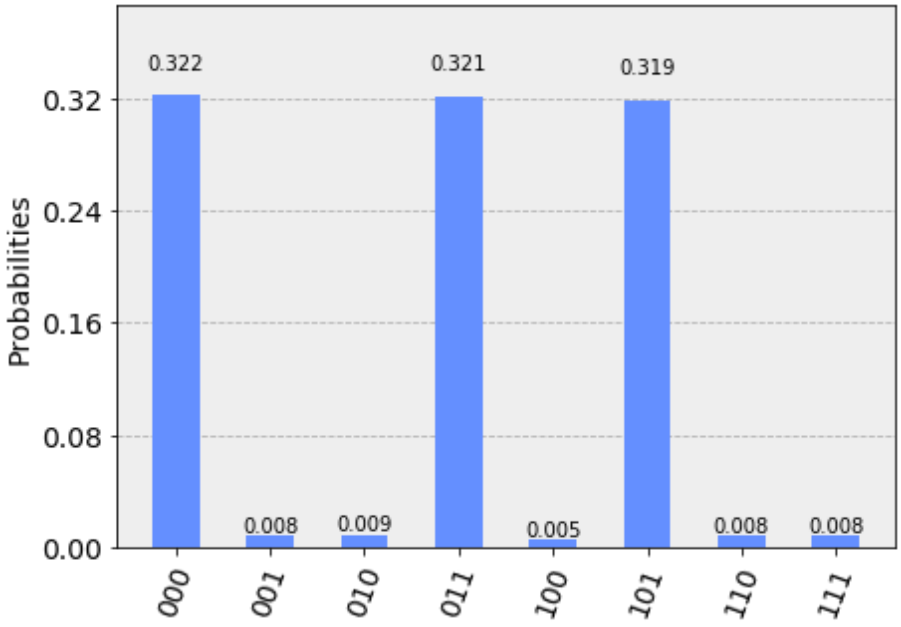

Figure 5 The output states of 1000 iterations

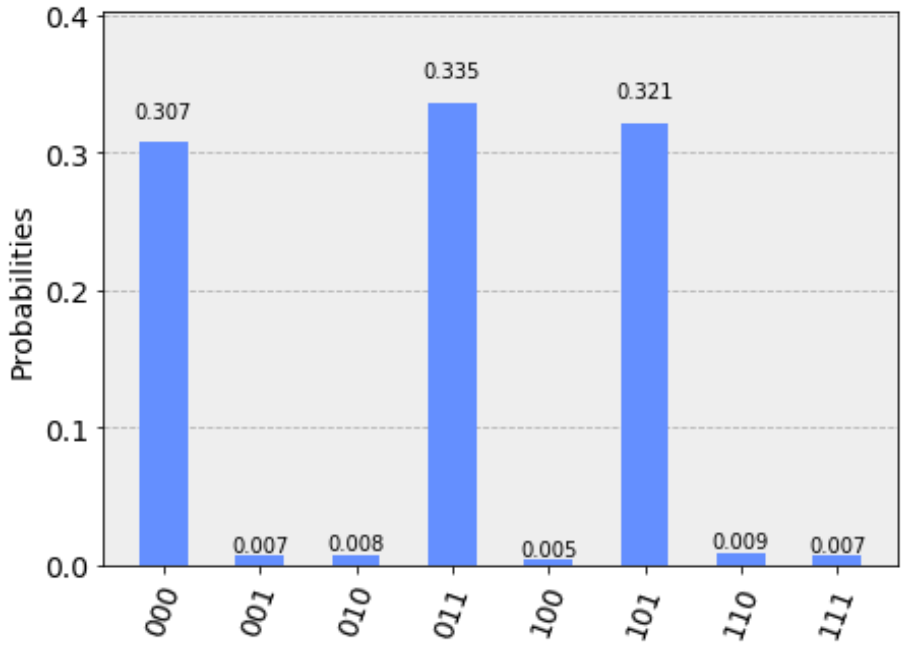

Figure 6 The output states of 1500 iterations

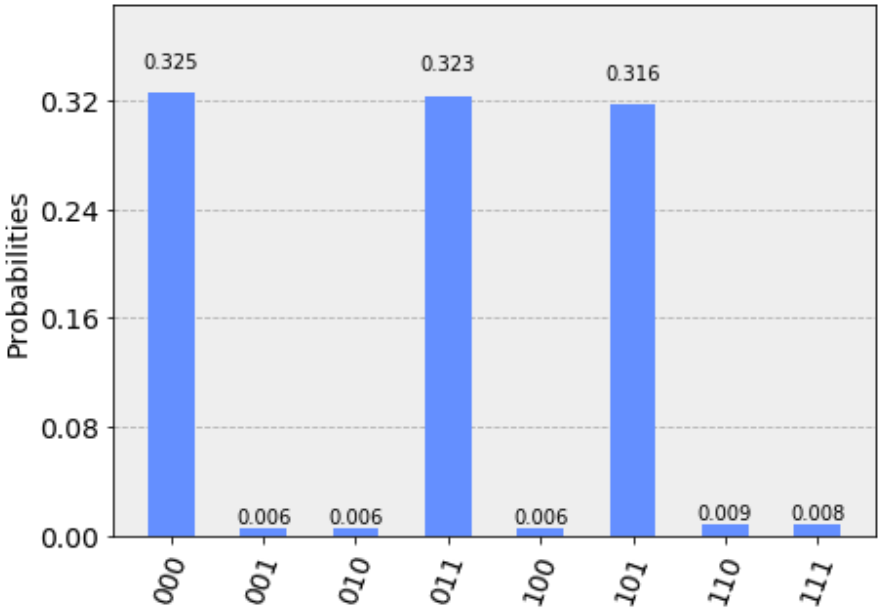

Figure 7 The output states of 5000 iterations

According to the results of 3-SAT problem searching, the non-solution states are suppressed, and the possible solutions, $|000>, |011>$ and $|101>$ dominate with equal probability. The output state $|000>$ means the target codon cannot occupy the base "U" in all three positions simultaneously. The second output state $|011>$ determines the codon that contains the base "U" in the 1st or 2nd positions in the codon and the 3rd position cannot contain the base "U". The last output state $|101>$ determines the codon that occupies the base "U" at the 1st or 3rd positions but not the 2nd position. This means we have 3 families of the target codon that satisfy the proposed 3-SAT condition. The lists of possible RNA codons satisfying the 3-SAT condition are shown in Table 1.

Table 1 All possible RNA codon satisfying the 3-SAT condition

$\|000>$	$\|011>$	$\|101>$
CCC	UUC	UCC
CCA	UUA	UCA
CCG	UUG	UCG
ACC	CUC	UCU
ACA	CUA	CCU
ACG	CUG	ACU
GCC	AUC	UAA
GCA	AUA	UAG
GCG	AUG	GCU
CAC	GUC	UAU
CAA	GUA	UAC
CAG	GUG	UGU
AAC	UCC	UGC
AAA	UCA	UGA
AAG	UCG	UGG
GAA	UGC	CGU
GAG	UGA	GCU
GAC	UGG	AAU
CGC	UAA	AGU
CGA	UAG	GAU
CGG	UAC	GGU
AGA		CAU
AGG		
AGC		
GGC		
GGA		
GGG		

5. Conclusion

The RNA codons satisfying the trial 3-SAT condition were searched by Grover's algorithm. The positions of each base were represented by the qubit states. The search algorithm was performed with QISKIT, which an open-source quantum framework. The initial state contains 8 possible configurations corresponding to the logical value in the Boolean expression. The search was implemented with 100, 500, 1000 and 1500 iterations to study the effect on the output states. The output quantum state showed that the given expression has multiple solutions, and the candidate states were $|000>$, $|011>$ and $|101>$ with nearly equal probability at 1500 iterations. This means that the quantum search algorithm can find all multiple solutions simultaneously. Furthermore, the probability of each candidate state tends to be equally likely the same, and the non-solution states would be strongly suppressed when the number of iterations increased. These solution states revealed 3 families of satisfying RNA codons. This would be useful in bioinformatics research to search for the target RNA codon. For future work, we will modify Grover's search algorithm and the 3-SAT problem to search on DNA sequences or to design protein structure.

6. Acknowledgments

I am really thankful for the quantum framework from IBM to provide many useful tools for performing quantum computing.

7. References

Belahcène, K., Labreuche, C., Maudet, N., Mousseau, V., & Ouerdane, W. (2018). An efficient SAT formulation for learning multiple criteria non-compensatory sorting rules from examples. *Computers & Operations Research*, *97*, 58–71. https://doi.org/10.1016/j.cor.2018.04.019

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96* (pp. 212–219). Philadelphia, Pennsylvania, United States: ACM Press. https://doi.org/10.1145/237814.237866

Jonsson, P., Lagerkvist, V., Nordh, G., & Zanuttini, B. (2017). Strong partial clones and the time complexity of SAT problems. *Journal of Computer and System Sciences*, *84*, 52–78. https://doi.org/10.1016/j.jcss.2016.07.008

Liu, X., & Wang, S. (2010). Development of an in vivo computer for the SAT problem. *Mathematical and Computer Modelling*, *52*(11), 2043–2047. https://doi.org/10.1016/j.mcm.2010.06.006

Luan, L., Wang, Z., & Liu, S. (2012). Progress of Grover Quantum Search Algorithm. *Energy Procedia*, *16*, 1701–1706. https://doi.org/10.1016/j.egypro.2012.01.263

Pellegrini, P., Marlière, G., & Rodriguez, J. (2017). RECIFE-SAT: A MILP-based algorithm for the railway saturation problem. *Journal of Rail Transport Planning & Management*, *7*(1), 19–32. https://doi.org/10.1016/j.jrtpm.2017.08.001

Prestwich, S. (2003). SAT problems with chains of dependent variables. *Discrete Applied Mathematics*, *130*(2), 329–350. https://doi.org/10.1016/S0166-218X(02)00410-9

Yang, C.-N., & Yang, C.-B. (2005). A DNA solution of SAT problem by a modified sticker model. *Biosystems*, *81*(1), 1–9. https://doi.org/10.1016/j.biosystems.2005.01.001

Zaikin, O. (2017). A parallel SAT solving algorithm based on improved handling of conflict clauses. *Procedia Computer Science*, *119*, 103–111. https://doi.org/10.1016/j.procs.2017.11.166