Quantum Speedup for Protein Structure Prediction

Renata Wong and Weng-Long Chang

***Abstract*— Protein structure prediction (PSP) predicts the native conformation for a given protein sequence. Classically, the problem has been shown to belong to the NP-complete complexity class. Its applications range from physics, through bioinformatics to medicine and quantum biology. It is possible however to speed it up with quantum computational methods, as we show in this paper. Here we develop a fast quantum algorithm for PSP in three-dimensional hydrophobic-hydrophilic model on body-centered cubic lattice with quadratic speedup over its classical counterparts. Given a protein sequence of n amino acids, our algorithm reduces the temporal and spatial complexities to, respectively, $O(2^{\frac{n}{2}})$ and $O(n^2 \log n)$. With respect to oracle-related quantum algorithms for the NP-complete problems, we identify our algorithm as optimal. To justify the feasibility of the proposed algorithm we successfully solve the problem on IBM quantum simulator involving 21 and 25 qubits. We confirm the experimentally obtained high probability of success in finding the desired conformation by calculating the theoretical probability estimations.**

***Index Terms*— Molecular algorithms, NP-complete problems, protein structure prediction, quantum algorithms, quantum simulation, quantum speedup, quantum biology.**

I. Introduction

PROTEIN structure prediction (PSP) is one of the unsolved problems that in classical computation belong to the NP-complete complexity class [1]. The purpose of PSP is to predict the product of a protein's folding process. A protein (a linear sequence of amino acids) folds to assume its native three-dimensional conformation.

The thermodynamic hypothesis [2], [3] states that the native conformation of a protein is its thermodynamically most stable conformation, and, with some rare exceptions, it does not depend on whether the protein folds inside a cell or in a test tube [4]. This fact, combined with established connections between the protein's three-dimensional structure and its functionality, has the potential to accelerate drug discovery in medicine by enhancing the presently predominant trial-and-error experiments with computer simulations to detect a significant amount of drug issues *in silico*.

Manuscript received August 23, 2020; revised January 2, 2021; accepted March 7, 2021. Date of publication March 10, 2021; date of current version July 1, 2021. *(Corresponding author: Renata Wong.)*

Renata Wong is with the Department of Computer Science and Technology, Nanjing University, Nanjing 210023, China (e-mail: renata.wong@protonmail.com).

Weng-Long Chang is with the Department of Computer Science and Information Engineering, National Kaohsiung University of Science and Technology, Kaohsiung 80778, Taiwan (e-mail: changwl@cc.kuas.edu.tw).

Digital Object Identifier 10.1109/TNB.2021.3065051

Though PSP remains intractable classically, quantum computational methods can provide a significant speedup over their classical counterparts by utilizing quantum mechanical properties such as superposition and entanglement. It has been shown in [5] that the lower bound for quantum algorithms solving NP-complete problems of input size n is $\Omega(2^{\frac{n}{2}})$. This makes Grover's quantum search algorithm, which offers a quadratic improvement in performance over classical search algorithms, asymptotically the best achievable quantum speedup on hard, classically intractable problems [5]–[7] such as PSP. Grover's algorithm has been successfully implemented on various quantum computational devices and paradigms [8]–[11]. It is an integral part of the quantum counting algorithm [12], which has been tested experimentally as well [13]. It has also been successfully used to solve the maximal clique problem with a quadratic speedup [14]. It has been shown in [15] that finding the solution using Grover's algorithm can be made at zero failure rate for any input size by replacing phase inversion with phase rotation through a definite angle. The same holds for multiple solutions [16]. Among the various modified Grover algorithms (see e.g. in [17], [18]), the Grover-Long algorithm [15] has been proven to be optimal [18]. In fact, as pointed out in [19], Toyama *et al.* [18] have shown the Grover-Long algorithm to be exactly optimal. The algorithm has been experimentally demonstrated in a 3-qubit NMR system [20].

Some of the other remarkable advances in quantum algorithms solving hard problems include Shor's algorithm for integer factorization [21]. Very small instances of this algorithm have also been experimentally implemented on quantum devices [22], [23], albeit, like in the case of all quantum algorithms, the current number of commonly available quantum bits is still very limited (up to around 14 qubits) and doesn't allow for testing on a scale necessary for real-world applications. An insight into the capability of quantum computers can however be deduced from experiments conducted in 2019 using the Sycamore quantum processor. The experiments have shown that the processor takes about 200 seconds to sample one instance of a quantum circuit a million times, while a state-of-the-art classical supercomputer would require approximately 10,000 years for the same task [24]. Other notable quantum experiments solving quantum information problems, including that in quantum biology, have also been reported [25]–[30].

Here we develop a quantum algorithm for PSP in three dimensions under the HP model on a cubic lattice. The algorithm incorporates Grover's optimization algorithm to find

the desired solution, which is the most stable conformation for a given sequence of amino acids with respect to energy. With n being the length of an amino acid sequence, our algorithm achieves the asymptotic complexity $O(2^{\frac{n}{2}})$ for time and $O(n^2 \log n)$ for space. With respect to oracle-related quantum algorithms for NP-complete problems, we identify our algorithm as optimal.

A. Motivation

The hydrophobic-hydrophilic model considers all possible conformations and selects the one with the lowest energy as the native conformation. State-of-the-art classical supercomputers do not provide enough memory even to store the conformational space of an arbitrary amino acid sequence. The IBM OLCF-4 supercomputer for instance offers 250 petabytes (PB) of storage, where $1PB = 2^{50}$ bytes or 2^{53} bits. While this amount is enough to uniquely identify every conformation in the conformational space of some of the smallest protein sequences of ca. 50 amino acids in length, it is by far not enough to do that for a larger protein. More importantly, besides storing the conformational space, a large amount of additional bits are needed to process the data, including calculating the lattice coordinates, the energies and identifying the native conformation. Besides not having sufficient memory for the task, PSP has been shown to be NP-complete for classical computers. It is therefore clear that no classical supercomputer can solve the PSP problem in the HP model for any real world protein.

B. Main Contributions and Novelty

While it is impossible to solve the PSP problem in the HP model classically, quantum computers on the other hand promise to offer a solution once quantum computing becomes robust to errors and commercially feasible. As we show in this paper, a quantum counterpart of the classical algorithm performs better with respect to both time and space. In order to store the conformational space and compute all the necessary operations (such as coordinate and energy calculation, Grover's optimization) only a number of qubits that is polynomial ($O(n^2 \log n)$) in the input size is required, where the input is a sequence of n amino acids. With respect to time, a quadratic speedup can be achieved compared to the classical algorithm.

We use quantum computational principles of superposition, entanglement and quantum phase interference to solve the problem of predicting the protein structure. The proposed algorithm is, to our best knowledge, the first quantum algorithm for PSP that is fully specified down to single quantum gates. The algorithm correctly predicts the solution, as defined under the HP model, and has a high probability of observing the solution upon a measurement (see IV). This experimentally obtained probability is corroborated by its theoretical estimation (see V). Our experimental implementation involves three different tests. One test is carried out for the problem in three dimensions, while the other two tests are for two dimensions. The currently available state-of-the-art quantum resources are limited, with most of them providing between a single qubit to up to 14 qubits (some lab developments include up to 79 qubits [24], [31]–[33]), and not yet fault-tolerant to the extent of facilitating reliable, purposeful computation at low error rates. In fact, with more qubits added and with increasing circuit depth, the dynamics of quantum devices becomes intractable [34]. Therefore, as an intermediate solution, IBM's quantum simulator provides a platform for reliable testing of quantum circuits of up to 32 qubits. The experimental evaluation of our algorithm was carried out on this simulator. Given the constraints on the qubit number, two amino acids are the maximal size that can be simulated in three dimensions, while three is the maximal size for two dimensions. Our circuit depths ranged between 1,624-3,328 quantum gates (see Table II) and were some of the largest quantum circuits ever simulated.

II. The HP Model

We denote a protein sequence consisting of n amino acids with $a = a_1 \cdots a_n$. Each a_e denotes the amino acid located at a position $1 \leq e \leq n$ in the sequence. The hydrophobic-hydrophilic (HP) model for protein structure prediction classified each amino acid either as hydrophobic (H) or hydrophilic/polar (P) based on parameters obtained in physical experiments [35]. The hydrophobicity status of an amino acid can be stored in a single qubit, where $|1\rangle$ stands for a hydrophobic and $|0\rangle$ stands for a hydrophilic amino acid.

Under the three-dimensional HP-model, the sequence is mapped into a cubic lattice. We use the body-centered cubic lattice (Fig. 1a) for the mapping. This choice of lattice is not arbitrary. It has been shown [36] that the body-centered cubic lattice performs best for the hydrophobic-hydrophobic interactions, which constitute the key physical component of the model. As such, this type of lattice is able to reproduce correctly both helices and sheets [36], the two canonical structures present in three-dimensional protein folds [37].

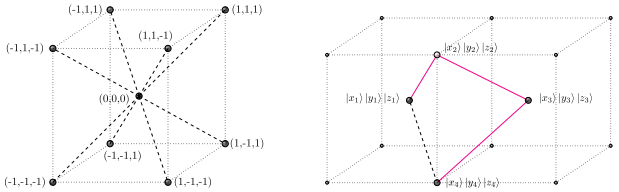
Finding the native conformation for a protein is usually subdivided into three steps. In the first step, the number of all possible conformations is determined. In the second step, the energy values for each possible conformation are calculated based on a scoring function that is particular to the given PSP model. And in the last step, the conformation with the lowest free energy is selected. For the HP model, evaluation of conformations is performed by a scoring function

$$V = -|(|a_i = 1\rangle, |a_j = 1\rangle)| \tag{1}$$

This function counts the number of all hydrophobic-hydrophobic contacts between amino acids a_i and a_j that are present in the lattice but not in the sequence (as represented by the dotted line in Fig. 1b). These are often referred to as loose contacts. Fig. 1b shows an example conformation of the sequence $\bigotimes |a_i\rangle$ for $1 \leq i \leq 4$ with such a loose contact present between the first and the last amino acid in the lattice. The two amino acids are not adjacent in the sequence. Here, in accordance with the usual convention, $\bigotimes$ stands for the tensor product.

III. Quantum Algorithm

The solution to the PSP problem is obtained in four steps. First, the quantum system is initialized to a superposition state



(a) (b)

Fig. 1. (a) Body-centered cubic lattice (bcc) with the eight coordinate transitions and their encoding with regard to the coordinates x, y, and z of each lattice point. The given encoding is used in our algorithm and is reproduced in Table I with the difference that a -1 in the lattice above corresponds to a 0 in Table I (-1, respectively) indicates that for a transition to take place, a 1 has to be added to (subtracted from) the existing coordinate value. The encoding of the transitions can be chosen arbitrarily. (b) Example conformation of length $n = 4$ embedded in bcc lattice with hydrophobic (•) and hydrophilic (◦) amino acids indicated. Solid lines indicate direct contacts in the conformation, while dotted lines indicate hydrophobic lattice contacts. $|x_i\rangle|y_i\rangle|z_i\rangle$ for $1 \leq i \leq 4$ are the respective amino acid coordinates.

that uniquely identifies each conformation. Second, conformations are mapped into a lattice and their coordinates are calculated in superposition. Third, the energy is computed for each conformation in superposition. The energy corresponds to the number of loose contacts. And finally, the conformation with the lowest energy (highest number of loose contacts) is selected by iterating Grover's algorithm. The following four subsections in detail describe the four steps above.

A. Preparing a Uniform Superposition State

A quantum register of length n stores the given amino acid sequence of length n. Each of the amino acids is denoted as $|a_e\rangle$ where $1 \leq e \leq n$. Correspondingly, each qubit $|a_e\rangle$ is initialized to either $|0\rangle$ (if it is hydrophilic), or $|1\rangle$ (if it is hydrophobic). Notation $|\beta^i\rangle$ is assumed throughout to indicate that a qubit $|\beta\rangle$ has a value $|i\rangle$ for $i \in \{0, 1\}$.

The Cartesian coordinates of e-th lattice site are: (x_e, y_e, z_e), where $1 \leq e \leq n$. Binary variables $w_{d,c}$ with $c \in \{1, 2, 3\}$ (Table I) encode the eight possible transitions in a bcc lattice from the $(d-1)$-th to the d-th lattice site for $2 \leq d \leq n$. The location of the first amino acid is fixed at the value $|x_1\rangle = |y_1\rangle = |z_1\rangle = 0$.

Representation of lattice coordinates x, y, and z requires $2(n-1)+1$ values for each: the number 0, $n-1$ positive and $n-1$ negative numbers. The smallest number of bits necessary to represent each of the coordinate values is therefore $t = \lceil \log_2(2(n-1)+1) \rceil$. All three coordinates of e-th lattice site are encoded as: $\bigotimes |x_e\rangle$, $\bigotimes |y_e\rangle$, $\bigotimes |z_e\rangle$, where $1 \leq e \leq n$. Three binary variables $w_{d,1}$, $w_{d,2}$, and $w_{d,3}$ (Table I) encode the eight possible transitions in a bcc lattice from the $(d-1)$-th to d-th lattice site, where $2 \leq d \leq n$.

The system is set into a superposition state of $n+1$ qubits involving the $2^{3(n-1)}$) possible conformations and an auxiliary qubit $|g\rangle = |0\rangle$ used in oracle:

$$|\psi_1\rangle = \frac{1}{\sqrt{2^{3(n-1)}}} \sum_{w=0}^{2^{3(n-1)}-1} \frac{|w\rangle(|0\rangle + |1\rangle)}{\sqrt{2}} |a\rangle|x_w\rangle|y_w\rangle|z_w\rangle \quad (2)$$

Here, $|a\rangle$ represents the encoded amino acid sequence, while $|x_w\rangle$, $|y_w\rangle$, and $|z_w\rangle$ represent the registers holding the x, y, and z coordinate, respectively, for each conformation that is uniquely identified by the values of $|w\rangle$. All the coordinates have initially the default value of 0. For the sake of clarity, we have omitted smaller auxiliary registers that will be needed to store temporary values when conducting quantum addition.

TABLE I
ENCODING FOR TRANSITION DIRECTIONS

$\|w_{d,3}\rangle$	$\|w_{d,2}\rangle$	$\|w_{d,1}\rangle$	**from $(d-1)$st to dth site**
0	0	0	D_L (downward to the left)
0	0	1	D_B (downward into the page)
0	1	0	U_L (upward to the left)
0	1	1	U_B (upward into the page)
1	0	0	D_F (downward out of the page)
1	0	1	D_R (downward to the right)
1	1	0	U_F (upward out of the page)
1	1	1	U_R (upward to the right)

B. Calculating Coordinates

Coordinate calculation is carried out in two steps on each of the amino acids in a conformation starting from amino acid $d = 2$ and ending on $d = n$. In the first step the coordinates x_{d-1}, y_{d-1}, and z_{d-1} of the $(d-1)$-th amino acid are copied to the coordinates x_d, y_d, and z_d of the d-th amino acid. This operation can be implemented by the controlled Pauli-X gate: $|t_d\rangle = |t_{d-1} \oplus t_d^0\rangle$, where $t \in \{x, y, z\}$, and t_{d-1} is the control qubit while t_d^0 is the target qubit. In the second step, the respective coordinates of the d-th amino acid are adjusted based on the value of $|w\rangle$ (Table I). The encoding of directional transitions is arranged in such a way that the three values in $|w_{d,c}\rangle$ can be treated separately. Namely, $c = 1, 2, 3$ corresponds to the z, y, and x coordinate, respectively. Whenever $|w_{d,c}\rangle = |0\rangle$ the respective coordinate t_d is decremented by 1, while for $|w_{d,c}\rangle = |1\rangle$ t_d is incremented by 1.

After completing this operation, the coordinates of each amino acid in each conformation have been mapped onto their location in the lattice. With this, the $|x_w\rangle$, $|y_w\rangle$, and $|z_w\rangle$ registers in Equation 2 contain no longer all-zero values but in fact the coordinates for each single conformation.

C. Finding Energy Values

The energy value for each conformation is stored in the state $|\Psi_{k,j,p}\rangle$, where $1 \leq k, j \leq n$ and $0 \leq p \leq 7$. Every single qubit $|\Psi_{k,j,p}\rangle$ stores a $|1\rangle$ if and only if the coordinates of the k-th lattice site are adjacent to the coordinates of the j-th lattice site in the bcc lattice and both k and j are hydrophobic. Index p indicates which of the possible eight neighbors j is with respect to k in accordance with Table I (here we assume $p = w_3 w_2 w_1$). For instance, $|\Psi_{k,j,6}\rangle = |1\rangle$ indicates that the neighbor j is in the upper front corner of k (U_F).

Finding adjacent lattice contacts entails comparisons of coordinates. We assume that the x, y, and z coordinates of the eight adjacent sites j of a site k are denoted by x^*, y^* and z^*, respectively, and that they together form the state $|x^*_{k,j,p}\rangle|y^*_{k,j,p}\rangle|z^*_{k,j,p}\rangle$, where $1 \leq k, j \leq n$, and $0 \leq p \leq 7$.

The structure of bcc lattice prevents contacts between the first three amino acids in a sequence. Therefore, the comparison of amino acid k with other amino acids starts with the

fourth amino acid relative to k. Hence, here $1 \leq k \leq n-3$ while $k+3 \leq j \leq n$. The operation of finding the energy value for a conformation has three steps (1-3). For each ordered pair (k, j) (1) eight copies of coordinates x, y, and z of site k are stored in eight sets of states x^*, y^*, and z^* of site j by means of controlled Pauli-X gates. Each of the eight copies is for a different relative position of j with respect to k. For example, for $k = 1$ and $j = 4$, $|x_1\rangle|y_1\rangle|z_1\rangle$ are copied to $|x^*_{1,4,p}\rangle|y^*_{1,4,p}\rangle|z^*_{1,4,p}\rangle$, respectively, where $0 \leq p \leq 7$ refers to each of the eight possible adjacent lattice sites. We note that this copy operation does not violate the no-cloning theorem [38] because each time the target bits $x^* = y^* = z^* = 0$. In the case of target bits being zero, the controlled NOT operation is equivalent to copying values from control bits to target bits. In the case of target bits not being zero, the controlled Pauli-X gate will perform an exclusive disjunction operation. (2) Depending on the value of p ($p = w_3w_2w_1$, Table I), the coordinates x^*, y^*, and z^* of j are then modified to fit the respective potential eight neighbors of each k by either incrementing or decrementing them by 1. For example, for $p = 4$, $|x^*_{1,4,p}\rangle|y^*_{1,4,p}\rangle|z^*_{1,4,p}\rangle = |001\rangle|111\rangle|111\rangle$. (3) A controlled Pauli-X operation $|x_j \oplus x^*_{k,j,p}\rangle$ is carried out for each combination of k, j, p where the starred coordinates are the target bits. If both control and target bits are equal - implying that j is a neighbor of k in the lattice but not in the sequence - this operation will result in $|x^*_{k,j,p}\rangle|y^*_{k,j,p}\rangle|z^*_{k,j,p}\rangle$ having zero-only values. If additionally both k and j are hydrophobic, meaning that a loose contact has been found, the corresponding state $|\Psi_{k,j,p}\rangle$ is set to $|1\rangle$. Otherwise, $|\Psi_{k,j,p}\rangle = |0\rangle$.

The energy values stored in $|\Psi_{k,j,p}\rangle$ must subsequently be summed up for each conformation into a string of the form:

$$|s_w\rangle = \bigotimes_j |s_{n-3,7,j}\rangle_w \tag{3}$$

where $n-3$ is the highest value of k, 7 is the highest value of p, and j ranges therefore from $8(n-3)$ through 0. $|s_w\rangle$ is a string of 0's with a single digit being a 1. Its computation process is shown in the flow diagram in Fig. 2. The initial conditions are listed in the top element. The output (string $|s_w\rangle$) is obtained upon reaching the End element. $\wedge$, $\vee$, and the bar are the bitwise AND, OR, and negation operations, respectively. $\oplus$ stands for the controlled Pauli-X gate (addition modulo 2).

With the above, the quantum state of the system becomes:

$$|\psi_2\rangle = |\psi_1\rangle|x^*_w\rangle|y^*_w\rangle|z^*_w\rangle|\Psi_w\rangle|s_w\rangle \tag{4}$$

where $|x^*_w\rangle, |y^*_w\rangle, |z^*_w\rangle$ are auxiliary registers used for determining adjacency, $|\Psi_w\rangle$ stores the information on loose contacts, and $|s_w\rangle$ stores the number of loose contacts for each conformation.

D. Identifying the Conformation With the Minimal Energy

After obtaining the string $|s_w\rangle$ containing information on the number of hydrophobic lattice contacts for each conformation in superposition, a search algorithm must be executed to find the native conformation. Under the HP model, the native conformation is the one with the highest number of loose

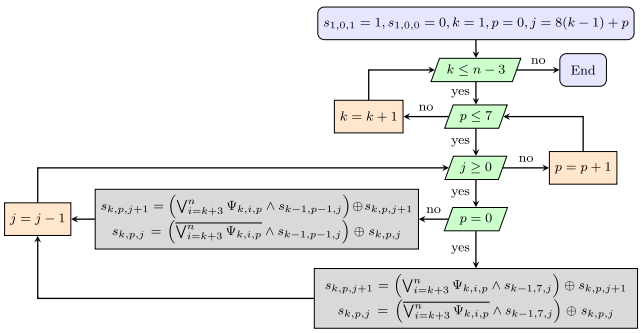


Fig. 2. The flow diagram for summing up the energy values into a single string.

contacts. One of the widely used algorithms is the Grover search algorithm [6] which amplifies the solution through multiple calls to an oracle. We use the Grover algorithm in our experimental validation of the PSP algorithm to find the native conformation. It should be noted that each of the $8(n-3)$ possible meaningful values of string $|s\rangle$ can be used to implement an oracle that compares this string with its actual value for each conformation in a superposition.

Grover's algorithm takes the quantum superposition state in Eq. 2 and amplifies the probability amplitude of the solution state so that upon measuring the state $|w\rangle$, the solution is found with a high probability. The prescribed number of iterations is $\frac{\pi}{4}\sqrt{2^{3(n-1)}}$. We denote the solution as $|w^*\rangle$. Applying oracle to the state $|\psi_1\rangle$ in Eq. 2 results in the phase inversion for the solution:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^{3(n-1)}}}\left(\sum_{w \neq w^*} \frac{|w\rangle(|0\rangle + |1\rangle)}{\sqrt{2}} + \frac{|w^*\rangle(|0\rangle - |1\rangle)}{\sqrt{2}}\right) \tag{5}$$

Here, for the sake of neatness, we omit the registers $|a\rangle, |x_w\rangle, |y_w\rangle, |z_w\rangle, |x^*_w\rangle|y^*_w\rangle|z^*_w\rangle|\Psi_w\rangle|s_w\rangle$. Register $|a\rangle$ does not change throughout computation, while for the remaining registers the index will be either $w = w^*$ or $w \neq w^*$ depending on whether they are part of the solution ($w = w^*$) or not ($w \neq w^*$).

After that, the diffusion operator carries out a reflection by the $|0\rangle$ vector by calculating

$$U_{diff} = \sum_{|w,g\rangle} (2\mu - \alpha_{|w,g\rangle})|w, g\rangle \tag{6}$$

where μ stands for the mean value of the amplitudes of all conformations in superposition.

E. Complexity Assessment

Given that an amino acid sequence is of length n, as the first main task, our algorithm not only encodes the sequence using n qubits but also uses additional qubits to encode directional transitions between amino acids, i.e. vector $|w\rangle$. After the encoding step, our algorithm then creates a superposition over the vector $|w\rangle$. This vector in superposition encodes all possible directional transitions from one amino acid to another in a sequence and constitutes therefore a unique identifier for each possible conformation. The number of bits in $|w\rangle$ before

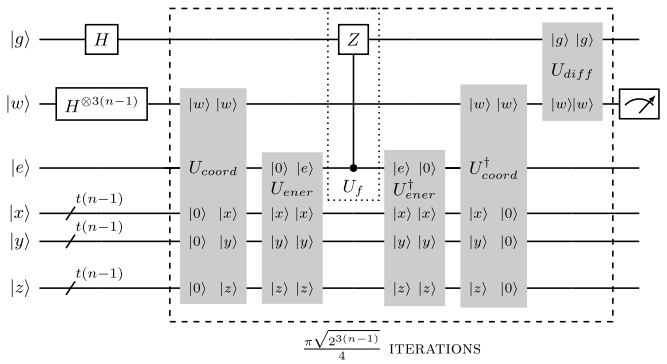


Fig. 3. A circuit schematic for the PSP algorithm for sequence length $n = 2$. U_{coord} is the unitary operation of coordinate calculation with its inverse being $U^\dagger_{coord}$. U_{ener} is the unitary operation of energy calculation with its inverse being $U^\dagger_{coord}$. U_f and U_{diff} represent, respectively, the oracle and the diffusion operator of Grover's optimization routine. Symbols inside operation boxes indicate which quantum registers participate in the operation and the values of these registers upon input/output (either $|0\rangle$ or calculated ($|g\rangle$, $|w\rangle$, $|e\rangle$, $|x\rangle$, $|y\rangle$, or $|z\rangle$)).

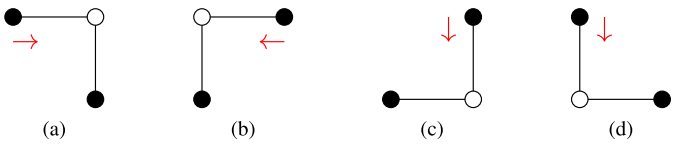


Fig. 4. Four possible conformations with adjacent hydrophobic-hydrophobic contacts for amino acid sequence $|101\rangle$ of length $L = 3$. Each edge is of unit length. Arrows indicate the first amino acid and the transition direction from the first amino acid to the second amino acid. The sequences of transition directions are as follows. R stands for transition rightwards, D downwards, L leftwards, and U upwards. (a) $|w\rangle = |RD\rangle = |0110\rangle$. (b) $|w\rangle = |LD\rangle = |1110\rangle$. (c) $|w\rangle = |DL\rangle = |1001\rangle$. (d) $|w\rangle = |DR\rangle = |1011\rangle$.

superposition is $3 \times (n-1)$. For that reason, the number of elements that are the inputs to our algorithm is $N = 2^{3\times(n-1)}$), i.e. the magnitude of $|w\rangle$ after applying Hadamard gates to create a superposition. The number of elements in $|w\rangle$ in superposition is used to estimate the optimal number of Grover's iterations for labelling the answer(s) and completing the amplification of the amplitude of the answer(s), as also shown in Fig. 3. This gives us the asymptotic time complexity of $O(2^{n/2})$.

Next, as the second main task, our algorithm uses additional quantum bits to encode all of the sequence's possible coordinates, i.e. the conformational space of an amino acid sequence. Encoding of coordinates takes $3\times n\times\log(2\times(n-1)+1)$ qubits. Next, as the third main task, our algorithm uses auxiliary quantum bits to conduct adjacency determination between any two amino acids. Encoding of the auxiliary quantum bits for adjacency determination takes $7 \times (n^2 \times \log(2 \times (n-1)+1))$ qubits. Other small number of quantum bits are negligible with respect to asymptotic spatial complexity. Taken together, this gives a spatial complexity of $O(n^2 \times \log n)$. A simple example is shown in the experimental implementation in Section IV.

IV. Experimental Implementation

A smallest instance of the PSP algorithm requires several hundred logical qubits, which is more than are currently available on any quantum device or simulator. Therefore, we simulate a simplified version of the algorithm to confirm its correctness. The simulations were executed on IBM simulator which allows tests of quantum circuits of up to 32 qubits. Given the presently available quantum computing resources, $n = 2$ (requires 21 qubits) was the maximal length that could be tested on the bcc lattice. Additionally, we also include simulations conducted on a two-dimensional square lattice for $n = 3$ (requires 25 qubits, maximal for this dimensionality). Three computations were successfully carried out: (A) identifying a single solution $|w\rangle = |110\rangle$ for amino acid sequence $|11\rangle$ in three dimensions, (B) identifying a single solution $|w\rangle = |1001\rangle$ for amino acid sequence $|101\rangle$ (see Fig. 4c), and (C) identifying four solutions $|w\rangle = |0110\rangle$, $|1110\rangle$, $|1001\rangle$, and $|1011\rangle$ for amino acid sequence $|101\rangle$ (see Fig. 4). The results of all three computations confirm the validity of our algorithm and demonstrate a high probability of finding the solution(s): 0.954 for (A) (Fig. 5b), 0.999 for (B) (Fig. 5d), and $0.222 - 0.263$ for each of the four solutions in (C) (Fig. 5f). All three experimentally obtained values correspond closely to the theoretical estimations of the probability of finding the correct conformation under the respective model given in Fig. 5.

Length $n = 2$ required 21 qubits and was the maximal length that could be simulated without exceeding the 32 qubit limit. A sequence $|a\rangle = |11\rangle$ was assumed implicitly, while the arbitrarily chosen conformation was $|w\rangle = |110\rangle$.

With this in mind, each of the simplified instances of the algorithm consists of the following steps:

1) Register initialization, including setting the system into a superposition over vectors $|w\rangle$, which encodes the directional transitions for all candidate conformations, and $|g\rangle$, which is an ancillary qubit storing temporarily information about the energy of each conformation.
2) Calculation of three-dimensional Cartesian coordinates $|x\rangle$, $|y\rangle$ and $|z\rangle$ (for (A) only) for each conformation.
3) Calculation of energy values for each conformation. The energy value is stored in quantum register $|e\rangle$. $|e\rangle = |1\rangle$ if and only if the respective conformation is a solution.
4) Transfer of the energy value from vector $|e\rangle$ to $|g\rangle$ and flipping the sign of $|g\rangle$ whenever $|g\rangle = |1\rangle$. This step fulfills a twofold function. On the one hand, it calculates the energy of each conformation and, on the other hand, it inverts the phase of the solution(s). The latter corresponds in its function to the oracle in Grover's algorithm.
5) Uncomputing the energies of each conformation. This step is required for the proper functioning of Grover's diffusion operator in step 7.
6) Uncomputing the coordinates of all conformations by executing the coordinate calculation step in reverse. This step results in all coordinates being reset to all-zero values and is required for the correct performance of Grover's algorithm in the last step of our algorithm.
7) Application of Grover's diffusion operator to registers $|w\rangle$ and $|g\rangle$.

Steps 2. through 7. are iterated over as many times as the Grover algorithm requires for a given number of solutions and

TABLE II
RESOURCES REQUIRED FOR EACH CASE

Gates (per iteration)	A	B	C
Hadamard gates	12	15	15
Pauli-X gates	49	67	59
Controlled-X (CX) gates	28	24	24
Toffoli (CCX) gates	509	724	712
Controlled-Z (CZ) gates	2	2	2
Total (per iteration)	600	832	812
Total	1,800	3,328	1,624

candidate conformations. The required number of iterations is

$$\frac{\pi}{4}\sqrt{\frac{N}{M}} \tag{7}$$

where N is the number of candidate conformations and M is the number of solutions.

Fig. 3 shows the general schematic of a simplified instance of the algorithm. Only the working qubits are shown. Initialization involves setting the system into a superposition over $|w\rangle$ and $|g\rangle$. Then each conformation has its coordinates calculated in superposition depending on the value of $|w\rangle$ with a controlled quantum adder. With the coordinates known, the energy value for each conformation is calculated. The energy $|e\rangle = |1\rangle$ if and only if the conformation has the desired contacts in the lattice. Otherwise, $|e\rangle = |0\rangle$. For the coordinate and energy results see Fig. 5a. Next, qubit $|g\rangle$ is used in the oracle to invert the phase of the conformation that has energy value $|e\rangle = |1\rangle$, resulting in

$$|\psi_3'\rangle = \frac{1}{\sqrt{2^{3(n-1)}}} \sum_{w=0}^{2^{3(n-1)}-1} \frac{|w\rangle(|0\rangle + (-1)^{e+1}|1\rangle)}{\sqrt{2}} \tag{8}$$

in accordance with the state $|\psi_3\rangle$ in Eq. 5.

The energy values and the coordinates must then be uncomputed before the diffusion operator can be applied. The measurement is carried out on the state $|w\rangle$, which uniquely identifies each conformation. The experimental results confirm the validity of our algorithm and demonstrate a high probability of finding the solution(s). The experimental results correspond closely to the theoretical estimations of the probability of finding the correct conformation calculated below.

A. Assessment of Resources

Case (A) required 21 qubits, while both (B) and (C) required 25 qubits each. Table II lists the number of gates used for each of the three cases.

B. Data Availability

The experimental validation was carried out on IBM's qasm simulator using the QISKit software development kit [39]. The Python source files are available on request from the corresponding author.

V. THEORETICAL PROBABILITY OF IDENTIFYING THE MINIMUM ENERGY CONFORMATION

We verify the experimental results obtained in the previous section by estimating the theoretical probability of obtaining a solution, which corresponds to the respective native conformation under the HP model. We show the detailed calculation for the case (A) only. For the cases (B) and (C), we provide only the respective mean value and the probability of identifying correctly the right conformation for each iteration. The number of iterations is obtained from Formula 7.

For the case (A), the number of iterations is therefore $\pi \approx 3$. The four qubits that are the inputs to the diffusion operator of the PSP algorithm are $|w\rangle$ and $|g\rangle$. Let $|k\rangle$ denote $|w\rangle|g\rangle$.

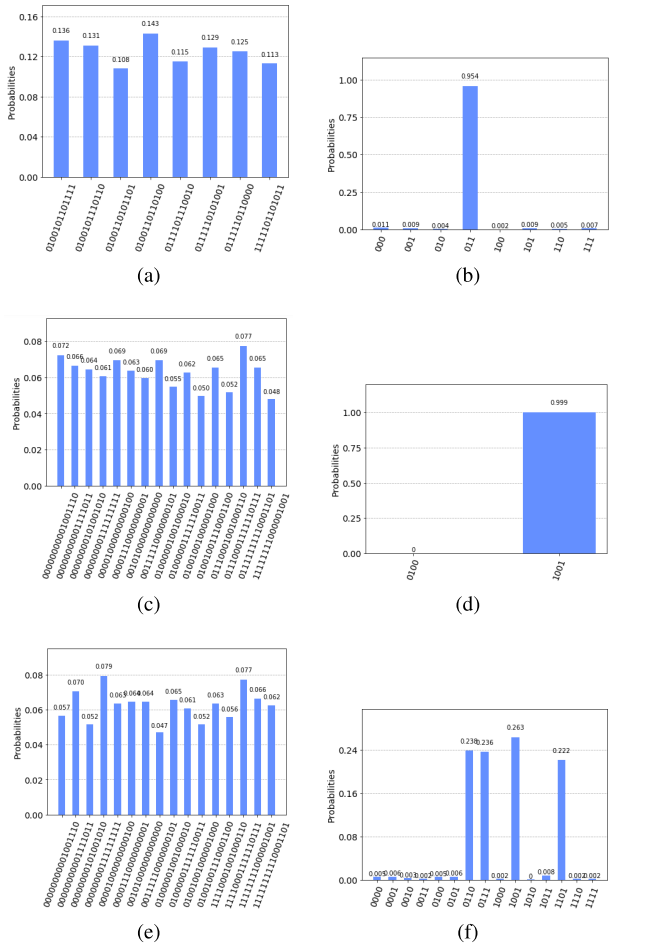


Fig. 5. Outputs of the PSP algorithm, to be read from top to bottom. (a) Calculation of coordinates and energies for (A). The first three bits represent $|w\rangle$, the next nine represent the coordinates x_2, y_2, and z_2, of the second amino acid (three bits for each coordinate). The coordinates x_1, y_1, and z_1 of the first amino acid are fixed at all-zero values. The last bit represents the energy value e. Only the conformation $|w\rangle = |110\rangle$ correctly shows $|e\rangle = |1\rangle$. (b) Solution $|w\rangle = |110\rangle$ is observed with prob. 0.954 after three iterations. The result corresponds closely to the theoretical estimation of 0.9613. (c) Calculation of coordinates and energies for (B). The first four bits represent $|w\rangle$, the next twelve represent the coordinates x_2, x_3, y_2 and y_3, of the second and the third amino acid, respectively (three bits for each coordinate). The coordinates x_1 and y_1 of the first amino acid are fixed at all-zero values. The last bit represents the energy value e. Only the conformation $|w\rangle = |1001\rangle$ correctly shows $|e\rangle = |1\rangle$. (d) Solution $|w\rangle = |1001\rangle$ is observed with prob. 0.999 after four iterations. The result corresponds closely to the theoretical estimation of 0.9993. (e) Calculation of coordinates and energies for (C). The first four bits represent $|w\rangle$, the next twelve represent the coordinates x_2, x_3, y_2 and y_3, of the second and the third amino acid, respectively (three bits for each coordinate). The coordinates x_1 and y_1 of the first amino acid are fixed at all-zero values. The last bit represents the energy value e. Only the four conformations $|w\rangle = |1001\rangle$, $|0110\rangle$, $|1110\rangle$, and $|1011\rangle$ have $|e\rangle = |1\rangle$. (f) The four solutions are observed after one iteration with probability between $0.222 - 0.263$ each. This result corresponds closely to the expected theoretical probability 0.2363 of finding each of the solutions.

The state of these four qubits is initially

$$|\phi_{A,1}\rangle = \frac{1}{4}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{1}{4}|k\rangle \tag{9}$$

where $|k^*\rangle$ is the single solution, i.e. the conformation that should be output after executing the PSP algorithm. The first application of the oracle negates the phase of the solution:

$$|\phi_{A,2}\rangle = -\frac{1}{4}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{1}{4}|k\rangle \tag{10}$$

The mean in the first iteration of the algorithm is $\mu_1 = 0.2187$. The diffusion operator is

$$U_{diff} = \sum_k (2\mu - \alpha_k)|k\rangle \tag{11}$$

and its application results in the following state for the first iteration:

$$|\phi_{A,3}\rangle = \frac{11}{16}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{3}{16}|k\rangle \tag{12}$$

This corresponds to the probability of outputting the solution upon measurement being 0.4726.

In the second iteration, the oracle inverts the phase of the solution resulting in the following state:

$$|\phi_{A,4}\rangle = -\frac{61}{64}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{5}{64}|k\rangle \tag{13}$$

The mean is $\mu_3 = 0.0136$ for this iteration. And therefore the diffusion operator amplifies the phase of the solution as follows:

$$|\phi_{A,5}\rangle = \frac{251}{256}|k^*\rangle - \sum_{|k\rangle\neq|k^*\rangle}\frac{293}{256}|k\rangle \tag{14}$$

Hence, upon measuring the state after the third iteration of the PSP algorithm, the probability of seeing the solution is 0.9613.

The probability calculations of finding the solution(s) for the cases (B) and (C) can be carried out along the same lines. In both cases, the five qubits that are the inputs to Grover's routine are $|w\rangle$ and $|g\rangle$. Let $|k\rangle$ denote $|w\rangle|g\rangle$. For (B), the number of iterations is $\sqrt{2}\pi \approx 4$ and the state of these five qubits is initially

$$|\phi_{B,1}\rangle = \frac{1}{\sqrt{32}}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{1}{\sqrt{32}}|k\rangle \tag{15}$$

where $|k^*\rangle$ is the single solution, i.e. the conformation that should be output after executing the PSP algorithm. Table III shows the mean value and the probability of observing the desired solution upon a measurement after each of the four iterations.

For the case (C), the number of iterations is $\frac{\sqrt{2}}{2}\pi \approx 2$. The five qubits $|w\rangle$ and $|g\rangle$ are initially in the state

$$|\phi_{C,1}\rangle = \sum_{|k^*\rangle}\frac{1}{\sqrt{32}}|k^*\rangle + \sum_{|k\rangle\neq|k^*\rangle}\frac{1}{\sqrt{32}}|k\rangle \tag{16}$$

where $|k^*\rangle$ are the four solutions, i.e. the conformations that should be output after executing the PSP algorithm. Table IV shows the mean value and the probability of observing each of the four solutions after each iteration.

TABLE III
THEORETICAL ESTIMATION OF PROBABILITY FOR CASE (B)

Iteration	Mean	Probability of finding the solution
1	0.1657	0.2583
2	0.1339	0.6025
3	0.0854	0.8971
4	0.0262	0.9993

TABLE IV
THEORETICAL ESTIMATION OF PROBABILITY FOR CASE (C)

Iteration	Mean	Probability of finding the solution
1	0.1325	0.1953
2	0.0220	0.2363

All three probability estimations presented in this section correspond closely to the experimentally obtained probabilities and corroborate thereby the correctness of our PSP algorithm.

VI. CONCLUSION

We have proposed a quantum algorithm for the problem of protein structure prediction in three-dimensional hydrophobic-hydrophilic model on body-centered cubic lattice that offers a quadratic speedup over its classical counterparts and has polynomial space requirements. We have also demonstrated the correctness of our algorithm by conducting a simulation on the IBM quantum simulator. We have further confirmed the correctness of the experimental results by calculating the theoretical estimation of finding the solution. Given the recent progress in the development of quantum computing devices, we hope that our algorithm could be implemented and tested on real life proteins in foreseeable future.

We note that the database containing all possible conformations can be seen as a structured one, especially with respect to the string $|s_w\rangle$ which contains the energy value for each conformation. However, in order to calculate this string, one needs to apply Grover's algorithm first. Therefore, since the database becomes sorted only after Grover's algorithm has been applied to it, the search is effectively executed on an unstructured database.

Despite being the best possible under Grover's algorithm, we note that a square root improvement over 2^n still renders the problem intractable. The quadratic advantage just pushes the exponential wall along a bit. We also would like to stress that with the quadratic advantage the improvement in the time required to execute the algorithm is better for larger n, i.e. for longer amino acid sequences than for shorter ones.

ACKNOWLEDGMENT

The authors wish to thank Prof. Mang Feng, Prof. Athanasios Vasilakos as well as two anonymous reviewers for their comments, which have made the paper stronger.

REFERENCES

[1] B. Berger and T. Leighton, "Protein folding in the hydrophobic–hydrophilic (HP) model is NP-complete," *J. Comput. Biol.*, vol. 5, no. 1, pp. 27–40, Jan. 1998.
[2] C. B. Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, no. 4096, pp. 223–230, Jul. 1973.

[3] E. Haber and C. B. Anfinsen, "Side-chain interactions governing the pairing of half-cystine residues in ribonuclease," *J. Biol. Chem.*, vol. 237, pp. 1839–1844, Jun. 1962.
[4] K. A. Dill, S. B. Ozkan, M. S. Shell, and T. R. Weikl, "The protein folding problem," *Annu. Rev. Biophys.*, vol. 37, pp. 289–316, Jun. 2008.
[5] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, "Strengths and weaknesses of quantum computing," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1510–1523, Oct. 1997.
[6] L. K. Grover, "Quantum computers can search rapidly by using almost any transformation," *Phys. Rev. Lett.*, vol. 80, no. 19, pp. 4329–4332, May 1998.
[7] L. K. Grover, "Synthesis of quantum superpositions by quantum computation," *Phys. Rev. Lett.*, vol. 85, no. 6, pp. 1334–1337, Aug. 2000.
[8] I. L. Chuang, N. Gershenfeld, and M. Kubinec, "Experimental implementation of fast quantum searching," *Phys. Rev. Lett.*, vol. 80, no. 15, pp. 3408–3411, Apr. 1998.
[9] I. L. Chuang, L. M. K. Vandersypen, X. Zhou, D. W. Leung, and S. Lloyd, "Experimental realization of a quantum algorithm," *Nature*, vol. 393, no. 6681, pp. 143–146, May 1998.
[10] M. O. Scully and M. S. Zubairy, "Quantum optical implementation of Grover's algorithm," *Proc. Nat. Acad. Sci. USA*, vol. 98, no. 17, pp. 9490–9493, 2001.
[11] C. Figgatt, D. Maslov, K. A. Landsman, N. M. Linke, S. Debnath, and C. Monroe, "Complete 3-Qubit Grover search on a programmable quantum computer," *Nature Commun.*, vol. 8, no. 1, p. 1918, Dec. 2017.
[12] G. Brassard, P. Høyer, and A. Tapp, "Quantum counting," in *Automata, Languages and Programming*, K. G. Larsen, S. Skyum, and G. Winskel, Eds. Berlin, Germany: Springer, 1998, pp. 820–831.
[13] J. A. Jones and M. Mosca, "Approximate quantum counting on an NMR ensemble quantum computer," *Phys. Rev. Lett.*, vol. 83, no. 5, pp. 1050–1053, Aug. 1999.
[14] W.-L. Chang, Q. Yu, Z. Li, J. Chen, X. Peng, and M. Feng, "Quantum speedup in solving the maximal-clique problem," *Phys. Rev. A, Gen. Phys.*, vol. 97, no. 3, Mar. 2018, Art. no. 032344.
[15] G. L. Long, "Grover algorithm with zero theoretical failure rate," *Phys. Rev. A, Gen. Phys.*, vol. 64, no. 2, Jul. 2001, Art. no. 022307.
[16] G. Long and Y. Liu, "Search an unsorted database with quantum mechanics," *Frontiers Comput. Sci. China*, vol. 1, no. 3, pp. 247–271, Jul. 2007.
[17] L. K. Grover and J. Radhakrishnan, "Is partial quantum search of a database any easier?" in *Proc. 17th Annu. ACM Symp. Parallelism Algorithms Archit. (SPAA)*, 2005, pp. 186–194.
[18] F. M. Toyama, W. van Dijk, and Y. Nogami, "Quantum search with certainty based on modified Grover algorithms: Optimum choice of parameters," *Quantum Inf. Process.*, vol. 12, no. 5, pp. 1897–1914, May 2013.
[19] G. Castagnoli, "Highlighting the mechanism of the quantum speedup by time-symmetric and relational quantum mechanics," *Found. Phys.*, vol. 46, no. 3, pp. 360–381, Mar. 2016.
[20] Y. Liu and F. H. Zhang, "First experimental demonstration of an exact quantum search algorithm in nuclear magnetic resonance system," *Sci. China Phys. Mech. Astron.*, vol. 58, no. 7, 2015, Art. no. 070301.
[21] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1484–1509, Oct. 1997.
[22] B. P. Lanyon *et al.*, "Experimental demonstration of a compiled version of Shor's algorithm with quantum entanglement," *Phys. Rev. Lett.*, vol. 99, no. 25, Dec. 2007, Art. no. 250505.
[23] T. Monz *et al.*, "Realization of a scalable Shor algorithm," *Science*, vol. 351, no. 6277, pp. 1068–1070, Mar. 2016.
[24] F. Arute *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.
[25] T. P. Xiong *et al.*, "Experimental verification of a Jarzynski-related information-theoretic equality using a single trapped ion," *Phys. Rev. Lett.*, vol. 121, no. 8, Aug. 2018, Art. no. 010601.
[26] Z. Li *et al.*, "Quantum simulation of resonant transitions for solving the eigenproblem of an effective water Hamiltonian," *Phys. Rev. Lett.*, vol. 122, no. 9, Mar. 2019, Art. no. 090504.
[27] P. Wang, C. Chen, X. Peng, J. Wrachtrup, and R.-B. Liu, "Characterization of arbitrary-order correlations in quantum baths by weak measurement," *Phys. Rev. Lett.*, vol. 123, no. 5, Aug. 2019, Art. no. 050603.
[28] B. C. Britt, "Modeling viral diffusion using quantum computational network simulation," *Quantum Eng.*, vol. 2, no. 1, p. e29, Mar. 2020.
[29] M. Tao, M. Hua, N. Zhang, W. He, Q. Ai, and F. Deng, "Quantum simulation of clustered photosynthetic light harvesting in a superconducting quantum circuit," *Quantum Eng.*, vol. 2, no. 3, p. e53, Sep. 2020.
[30] J. Pearson, G. Feng, C. Zheng, and G. Long, "Experimental quantum simulation of avian compass in a nuclear magnetic resonance system," *Sci. China Phys., Mech. Astron.*, vol. 59, no. 12, Dec. 2016, Art. no. 120312.
[31] *IBM Quantum*. Accessed: Mar. 10, 2021. [Online]. Available: https://quantum-computing.ibm.com/
[32] *Rigetti*. Accessed: Mar. 10, 2021. [Online]. Available: https://www.rigetti.com/
[33] *Google Quantum AI*. Accessed: Mar. 10, 2021. [Online]. Available: https://quantumai.google/
[34] A. D. Corcoles *et al.*, "Challenges and opportunities of near-term quantum computing systems," *Proc. IEEE*, vol. 108, no. 8, pp. 1338–1352, Aug. 2020.
[35] R. Wolfenden, "Experimental measures of amino acid hydrophobicity and the topology of transmembrane and globular proteins," *J. Gen. Physiol.*, vol. 129, no. 5, pp. 357–362, May 2007.
[36] A. Onofrio *et al.*, "Distance-dependent hydrophobic–hydrophobic contacts in protein folding simulations," *Phys. Chem. Chem. Phys.*, vol. 16, no. 35, pp. 18907–18917, 2014.
[37] L. Pauling, R. B. Corey, and H. R. Branson, "The structure of proteins: Two hydrogen-bonded helical configurations of the polypeptide chain," *Proc. Nat. Acad. Sci. USA*, vol. 37, no. 4, pp. 205–211, Apr. 1951.
[38] J. L. Park, "The concept of transition in quantum mechanics," *Found. Phys.*, vol. 1, no. 1, pp. 23–33, 1970.
[39] H. Abraham *et al.*, "Qiskit: An open-source framework for quantum computing," 2019, doi: 10.5281/zenodo.2562110.