# New encoding for implementing lattice protein folding problems on a quantum annealer

Ryuji Takagi[1, 2]

[1] Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[2] Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

(Dated: May 18, 2018)

We introduce a new encoding scheme that maps a lattice protein model to a problem Hamiltonian for the quantum annealing protocol. Our encoding scheme is a space-based encoding approach inspired by the diamond encoding method, which allocates the qubits to the locations that $i$th amino acid can possibly reach. Compared to the diamond encoding, our method uses the exponentially smaller number of qubits to encode the position of each amino acid at the trade-off with a higher level of non-locality. We apply our scheme to a simple example—a chain with four amino acids on the 2D lattice— and numerically obtain the energy spectrum during the annealing process. We confirm that the ground state indeed corresponds to the native configuration, and illegal configurations are appropriately suppressed by large energy penalty.

## I. INTRODUCTION

The protein folding problem has been one of the most important topics in biology. A protein consists of a chain of amino acid residues, which are connected to each other by the covalent bonds. Many of the proteins form three-dimensional structures, and vast of investigations indicate that the structure of a protein, called the native configuration, is encoded into the sequence of amino acids of the chain. It is the configuration that minimizes the free energy, which takes into account the energy of the system, which is dominated by non-covalent interactions [1], and the entropy, which is the contribution from thermal fluctuation [2–8]. Functionality of the protein is closely related to the three-dimensional structure, so clearly folding process plays an essential role in biological systems. For instance, misfolding can lead to a reduced functionality of the protein [9], which causes some diseases including the cancer [10–12].

There are a couple of approaches to investigate the protein folding problem. One of the important aspects to understand is the dynamics of the folding process. The question is how proteins realize their native configurations in such a short amount of time (on the order of microseconds [13]) among an astronomical number of possible configurations [14]. Another important problem is the protein structure prediction, where one is to identify the native configuration given some sequence of amino acids. It bridges a microscopic description of an amino-acid sequence to a complex structure of the protein and gives insights into the functionality of the protein. We will focus on the protein structure problem in this report.

Simulation has been a powerful tool to investigate the protein folding, and various models have been employed. For instance, simulation with the all-atom model has been an active field of research [13, 15, 16]. The idea is to simulate the molecular dynamics of the present atoms driven by the empirically obtained force field in the system (open libraries for the force field include AMBER [17] and CHARMM [18]). However, it is computationally demanding to fully reconstruct the whole dynamics of the protein folding until it reaches the native structure.

The lattice protein model is something that sits at the opposite edge to the all-atom model. It is a simplified model where all the amino acids are to be situated on lattice points. Energy function is determined by the interaction due to the non-covalent bonds that are exerted between the acids that are positioned next to each other. Protein structure prediction is turned into a computational optimization problem where the energy function is to be minimized. Despite the simplicity of the model, it has proven to be helpful to extract a statistical description of the folding [7].

Even inside the lattice model, one still needs to decide the model for interaction and the dimensionality of the lattice. The simplest interaction model is the hydrophobic-polar (HP) model where amino acids are only labeled as either hydrophobic (H) or polar (P), and we assume negative energy only between two H type acids [19]. It turns out that even in this simplest model, it is hard to find the global minimum of the energy function; it has been found to be NP-complete [20, 21]. Due to this intrinsic hardness, tractable size of the system is somewhat limited; what has been reported is 36 acids for 3D with an exact algorithm [22] and 136 acids with a heuristic algorithm [23]. Also, it has turned out that HP model is not good enough to capture some of the key properties of a real problem, such as cooperativity [24, 25]. To remedy this, more realistic interaction model, such as Miyazawa-Jernigan (MJ) model [26], needs to be considered, but it puts additional complexity to the problem. The challenge is ultimately reduced to the hardness of the computational optimization problem.

Perdomo *et al.* tackled this difficulty by having a quantum device solve the optimization problem by quantum annealing [27, 28]. Quantum annealing [29, 30] is an optimization protocol inspired by the simulated annealing [31] where it uses the quantum fluctuation instead of the thermal fluctuation. It has been argued that the quantum annealing may show a speedup over classical algorithms because of the quantum effect such as quantum tunneling, which may allow the temporary solution stuck at a local minimum to tunnel through the energy gap to reach the global minimum [32]. It has been still unclear whether such a speedup is possible, and it has been under active investigation. Although it is unlikely that it realizes an exponential speedup on a NP-complete problem, there is still a hope that it could give a polynomial or even a constant factor speedup, which makes a practical difference.

The crucial part of this approach is to come up with an encoding scheme from the lattice protein folding problem to a Hamiltonian that can be implemented on the currently-available quantum device [33, 34]. Several schemes have been proposed [27, 28, 35], each of which has its own advantage and disadvantage. For instance, the "turn encoding" used in [28] is efficient in required number of qubits, but it comes with high level of complication in the Hamiltonian. The "diamond encoding" proposed in [35], on the other hand, allows for a simple construction of the Hamiltonian, but it is inefficient in the required number of qubits.

In this report, we provide a new encoding scheme that sits in the middle of these two. Our scheme allows for a straightforward construction of the Hamiltonian while greatly reducing the required number of qubits compared to the diamond encoding. It is inspired by the diamond encoding but uses exponentially less number of qubits to encode the location of amino acids. Because of this property, we call our encoding scheme *log-diamond encoding.* We first briefly review the quantum annealing and provide a detailed explanation on how the encoding works. We then apply it to a simple example and numerically check that it gives the right configuration as its solution.

## II. LATTICE PROTEIN FOLDING ON A QUANTUM ANNEALER

Here, we briefly review the quantum annealing protocol from the perspective of the computational optimization problem.

We use $X_i$ and $Z_i$ to denote Pauli $X$ and $Z$ operators acting on the $i$th qubit, and we call $\{|0\rangle, |1\rangle\}$ computational basis where they are the eigenstates of $Z$, i.e. $Z|0\rangle = |0\rangle, Z|1\rangle = -|1\rangle$. The space of $n$-qubit states is spanned by the tensor product of the single-qubit computational basis states. There are $2^n$ of those basis states. Suppose we would like to obtain the ground state of the Hamiltonian $H_p$ that only consists of Pauli $Z$ operators, such as Ising model $H_p = \sum_{ij} J_{ij} Z_i Z_j$. Clearly, the eigenstates of $H_p$ are computational basis states, but finding the ground state is hard in general. Now, let us consider the following time-dependent Hamiltonian

$$H(s) = sH_p + (1-s)H_i, \ 0 \le s \le 1 \tag{1}$$

where $H_i$ is some initial Hamiltonian, which does not commute with $H_p$. We choose $H_i$ such that the ground state is easy to prepare. For instance, $H_i$ can be chosen as $H_i = -\sum_i X_i$, whose ground state is clearly $|+\rangle^{\otimes n}$ where $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. The protocol works as follows. We prepare the ground state of $H(0) = H_i$. We then gradually increase $s$ from 0 to 1. According to the quantum adiabatic theorem [36], if the change in the Hamiltonian is slow enough, the state always stays in the ground state of the Hamiltonian. Thus, at the end of the protocol for $s = 1$, the state should be the ground state of $H_p$, which is the answer we wanted. How slow it should be depends on the minimum gap of the Hamiltonian. Specifically, let $T$ be the running time of the algorithm. Then, to ensure the final state remains the ground state, it is sufficient to take $T \gg \max_s ||\dot{H}(s)||/\Delta^2$ where $\Delta$ is the minimum gap of $H(s)$ [36]. It means that if the gap is only polynomially small, the algorithm runs in polynomial time whereas it takes exponential time for the Hamiltonian with an exponentially small gap.

## III. LOG DIAMOND ENCODING

Our construction is inspired by the diamond encoding method proposed in [35]. The idea of the diamond encoding is to assign qubits for $i$th acid only to the lattice points that are possibly reached by the $i$th acid. The diamond encoding allocates one qubit to each possible lattice point, so it requires $i^2$ qubits when $i$ is even and $i^2 - 1$ qubits when $i$ is odd for encoding the location of the $i$th acid. Our encoding solves this inefficiency by encoding it into exponentially smaller numbers of qubits in the trade-off for locality of the interaction. Although we explain for the case of 2D, extension to higher dimensions is straightforward. Imagine the 2D lattice specified by the $x, y$ coordinate $(x, y)$, and suppose that the first acid is positioned at $(0, 0)$. The second acid may be positioned either at $(1, 0), (0, 1), (-1, 0), (0, -1)$. In general, the $i + 1$th acid can be only positioned at the point whose distance from the point at which the $i$th acid is 1. To see the structure of the reachable lattice points, it is convenient to introduce the 'layer', which is a set of lattice points positioned on the edges of a diamond. Let $L_k = \{(x, y) | \, x, y \in \mathbb{Z}, |x+y| = k-1 \vee |x-y| = k-1\}$, then the $i$th acid can be only positioned at the point in

$$\begin{cases} L_i \cup L_{i-2} \cup \ldots L_4 \cup L_2 & (i : \text{even}) \\ L_i \cup L_{i-2} \cup \ldots L_3 \cup L_1 & (i : \text{odd}) \end{cases} \tag{2}$$

The location of an acid is completely determined by specifying a layer and a point in the layer, each of which we allocate qubits to. Specifically, we express the position of the $i$th acid by $\boldsymbol{s_i q_i}$ where $\boldsymbol{s_i} = s_{i1}s_{i2}\ldots s_{iT_i}$ and $\boldsymbol{q_i} = q_{i1}q_{i2}\ldots q_{iR_i}$ are the qubits labeling the layer and the position in the layer respectively. For the $i$th acid, there are $\lceil i/2 \rceil$ possible layers so $T_i = \log\lceil i/2 \rceil$. The required number of qubits to specify the point in a layer depends on the layer we are looking at since layer $L_k$ includes $4k$ lattice points. However, since we need to use the same number of qubits to specify the $i$th acid independent of the layer specified by $\boldsymbol{s_i}$, we will take $R_i = \log\lceil 4(i-1) \rceil$, the maximum possible number of qubits required, and use from the left-most qubits $q_{i1}$ to encode the position in the layer. We label the points in the layer counterclockwise from the right-most point in the layer as shown in Fig.1. There are some unused qubits in $\boldsymbol{q_i}$ for the encoding depending on the value of $\boldsymbol{s_i}$. We will see later that the values of those qubits do not matter because they do not contribute to energy function. The number of qubits required to express the position of $i$th qubit is then $T_i + R_i = \log\lceil i/2 \rceil + \log\lceil 4(i-1) \rceil$, which is exponentially smaller than the number of qubits required in the diamond encoding. For instance, for $i = 4$, we get $T_i = 1$ and $R_i = 4$. To specify the point $(1, 0)$, the qubit expression takes $\boldsymbol{s_4} = 0$ (because $(1, 0) \in L_2$ and $L_2$ is the first possible layer for $i = 4$) and $\boldsymbol{q_4} = 00 * *$ where $*$ can be either 0 or 1, each of which gives the same energy. Also, note that we do not need to allocate any qubit for $i = 1$ or $\boldsymbol{s_i}$ for $i = 2, 3$ because we assume the first acid is always positioned at $(0, 0)$, and there is only one layer to consider for $i = 2, 3$. Furthermore, we can take $\boldsymbol{q_2} = 00$ and $q_{32}q_{33} = 00$ without loss of generality because of the symmetry. The quantum state for the chain consisting of $N$ acids then takes the form

$$|00\rangle\, |q_{31}00\rangle\, |\boldsymbol{s_4 q_4}\rangle \ldots |\boldsymbol{s_N q_N}\rangle\,. \tag{3}$$

which uses $\sum_{i=4}^{N} (\log\lceil i/2 \rceil + \log\lceil 4(i-1) \rceil) + 1$ qubits.

Next, we will consider the Hamiltonian such that the configuration with minimum energy realizes the ground state of the Hamiltonian. Note that taking into account the interaction between neighboring interactions is not sufficient for constructing an appropriate Hamiltonian, because it may take the ground state that represents illegal configurations. For instance, we do not allow for configurations where more than one acids occupy the same lattice point or two acids that are connected by covalent bond are not positioned next to each other. We need to penalize these configurations by giving extra energy penalty so they will not take the ground state of the total Hamiltonian. Let us consider the Hamiltonian of the form

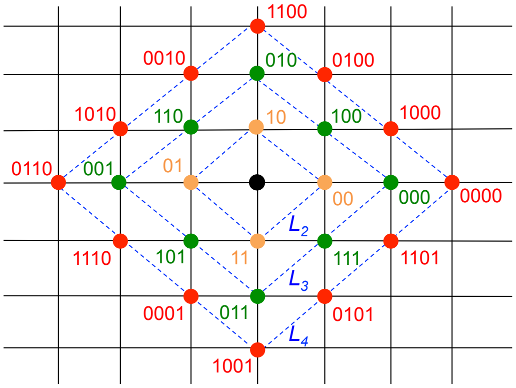


Figure 1. Schematics of the log diamond encoding. Different colors represent different layers, and points in a layer are labeled in the binary form (the left-most bit being the least significant bit) counted counter-clockwise from the right most point.

$$H_p = H_{overlap} + H_{connect} + H_{pair}. \tag{4}$$

where first two are terms penalizing the illegal configurations, and the third term takes care of the interaction between acids.

Before going into details about the construction of each term, let us introduce some functions that will be useful in the later discussions. As we saw, the lattice point for the $i$th acid is completely determined by $\boldsymbol{s_i}$ and $\boldsymbol{q_i}$, so a map from the qubit expression to the $x, y$ coordinate is well-defined. We write such a map as $\boldsymbol{r_i}(\boldsymbol{s_i}, \boldsymbol{q_i})$. For instance, $\boldsymbol{r_4}(1, 1000) = (2, 1)$ because $\boldsymbol{s_4} = 1$ specifies the layer $L_4$ (for $i = 4$, there are two possible layers, $L_2$ and $L_4$, so $\boldsymbol{s_4} = 0$ refers to $L_2$ and $\boldsymbol{s_4} = 1$ refers to $L_4$) and $\boldsymbol{q_4} = 1000$ specifies the second point in $L_4$ reached by going counter-clockwise starting from the right-most point $(3, 0)$. We also define the projector

$$P_{ik}(\tilde{x}_{ik}) = \frac{I + (-1)^{\tilde{x}_{ik}} Z^x_{ik}}{2} \tag{5}$$

where $x = s, q$ and $Z^x_{ik}$ denotes the Pauli $Z$ acting on the qubit corresponding to $x_{ik}$. $H_{overlap}$ gives energy penalty to the configurations where more than one acids occupy the same lattice point. We realize it by taking

$$H_{overlap} = \lambda_{overlap} \sum_{i=2}^{N} \sum_{j>i} \hat{f}_{ij} \tag{6}$$

where $\lambda_{overlap} > 0$ is a constant setting the amount of the energy penalty, and $\hat{f}_{ij}$ is the operator acting on the qubits corresponding to $i$th and $j$th acids that gives 1 when acting on the state where the lattice point for these two acids coincide and 0 otherwise. Explicitly, $\hat{f}_{ij}$ is written by

$$\hat{f}_{ij} = \sum{}' \left( \prod_{k,l} P_{ik}(\tilde{s}_{ik}) P_{il}(\tilde{q}_{il}) \right) \left( \prod_{k,l} P_{jk}(\tilde{s}_{jk}) P_{jl}(\tilde{q}_{jl}) \right) \tag{7}$$

where $\sum'$ means the sum over $\tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{q}}_i$ and $\tilde{\boldsymbol{s}}_j \tilde{\boldsymbol{q}}_j$ such that $\boldsymbol{r}_i(\tilde{\boldsymbol{s}}_i, \tilde{\boldsymbol{q}}_i) = \boldsymbol{r}_j(\tilde{\boldsymbol{s}}_j, \tilde{\boldsymbol{q}}_j)$.

$H_{connect}$ gives energy penalty to the configurations where two acids connected by covalent bond are not positioned next to each other on the lattice. It can be constructed as

$$H_{connect} = \lambda_{connect}(N-3) - \lambda_{connect} \sum_{i=3}^{N-1} \hat{g}_{ii+1} \tag{8}$$

where $\lambda_{connect} > 0$ is a constant, and $\hat{g}_{ii+1}$ is the operator acting on the qubits corresponding to $i$th and $i+1$th acids that gives 1 when acting on the state where the lattice point for these neighboring acids are next to each other on the lattice and 0 otherwise. The first term is added so the energy becomes zero for valid configurations. Importantly, we should set $\lambda_{overlap} \gg \lambda_{connect}$ so illegal configurations will not be allowed. $\hat{g}_{ii+1}$ is explicitly written by

$$\begin{aligned} \hat{g}_{ii+1} = \sum{}'' & \left( \prod_{k,l} P_{ik}(\tilde{s}_{ik}) P_{il}(\tilde{q}_{il}) \right) \\ \times & \left( \prod_{k,l} P_{i+1k}(\tilde{s}_{i+1k}) P_{i+1l}(\tilde{q}_{i+1l}) \right) \end{aligned} \tag{9}$$

where $\sum''$ means the sum over $\tilde{\boldsymbol{s}}_i \tilde{\boldsymbol{q}}_i$ and $\tilde{\boldsymbol{s}}_{i+1} \tilde{\boldsymbol{q}}_{i+1}$ such that $||\boldsymbol{r}_i(\tilde{\boldsymbol{s}}_i, \tilde{\boldsymbol{q}}_i) - \boldsymbol{r}_{i+1}(\tilde{\boldsymbol{s}}_{i+1}, \tilde{\boldsymbol{q}}_{i+1})|| = 1$.

Finally, $H_{pair}$ takes into account the interaction energy between acids that are not bonded by covalent bond but sit next to each other on the lattice. It is written by the form

$$H_{pair} = \sum_{i=1}^{N-3} \sum_{j>i+2} c_{ij} \hat{g}_{ij} \tag{10}$$

where $c_{ij} \leq 0$ is the matrix determined by the interaction model. For instance, in HP model, $c_{ij} = -1$ if $i$th acid and $j$th acid are hydrophobic and $c_{ij} = 0$ otherwise. In MJ model, $c_{ij}$ takes the value in the interaction table in [26]. To exclude illegal configurations, we need to set $\lambda_{connect} \gg \max_{ij} |c_{ij}|$.

Now that we construct the problem Hamiltonian $H_p$, the annealing process is ready to be run. For the initial Hamiltonian, we choose $H_i = \sum''' \left( \frac{I - X_{ik}^x}{2} \right)$ where $x = s, q$ and $X_{ik}^x$ denotes the Pauli $X$ acting on the qubit corresponding to $x_{ik}$, and $\sum'''$ denotes the summation over all the qubits except four predetermined qubits in (3). We choose this projector form so the ground state energy takes zero with the ground state being the tensor product of $|+\rangle$.

## IV. EXAMPLE

Let us apply our construction to a simple example. We consider the HP model and the chain consisting of four acids with the type H-P-P-H. (It does not matter if we consider HP model or MJ model for 4-acid chain because they have the same native configuration). We use this simple example with an obvious native configuration in order to check that our encoding properly outputs the desired native configuration as a ground state. The basis states are composed by 10 qubits $|00\rangle\,|q_{31}00\rangle\,|s_4 q_{41} q_{42} q_{43} q_{44}\rangle$ and it spans the space with dimension $2^6$.

$H_{overlap}$ only involves the second acid and the fourth acid due to the construction. The expression for the second acid is already fixed as $\boldsymbol{q_2} = 00$. The fourth acid would take the same position if and only it takes $\boldsymbol{s_4} = 0$ and $\boldsymbol{q_4} = 00**$ where $*$ denotes any value either 0 or 1. Thus, we get

$$H_{overlap} = \lambda_{overlap} \left( \frac{1 + Z_{41}^s}{2} \right) \left( \frac{1 + Z_{41}^q}{2} \right) \left( \frac{1 + Z_{42}^q}{2} \right). \tag{11}$$

where we take $\lambda_{overlap} = 20$.

$H_{connect}$ takes care of the bonds between the third and the fourth acid. Table I shows the combinations that contribute to $H_{connect}$, which composes $\hat{g}_{ii+1}$ in (8). Here, we take $\lambda_{connet} = 10$.

| $q_{31}$ | $s_{41}$ | $q_{41}$ | $q_{42}$ | $q_{43}$ | $q_{44}$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | $*$ | $*$ |
| | 1 | 0 | 0 | 0 | 0 |
| | 1 | 1 | 0 | 0 | 0 |
| | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | $*$ | $*$ |
| | 0 | 1 | 0 | $*$ | $*$ |
| | 1 | 1 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 0 | 0 |

Table I. Combinations of $q_{31}$ and $\boldsymbol{s_4 q_4}$ that contribute to $H_{connect}$.

Finally, $H_{pair}$ only involves the first and the fourth acids because it is the only pair that could have the interaction. The fourth acid can interact with the first acid if and only if $s_4 = 0$. Thus,

$$H_{pair} = c_{14} \left( \frac{1 + Z_{41}^s}{2} \right) \tag{12}$$

where we take $c_{14} = -1$.

Fig. 2 shows the obtained ground state and the 1st excited state and corresponding configurations and energy. The ground state appropriately corresponds to

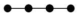
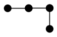
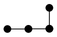
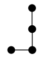
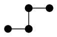
| $q_{31}$ | $s_{41}$ | $q_{41}$ | $q_{42}$ | $q_{43}$ | $q_{44}$ | configuration | energy |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 0 | * | * | | -1 |
| 0 | 1 | 0 | 0 | 0 | 0 |  | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 |  | |
| 0 | 1 | 0 | 0 | 0 | 0 |  | |
| 0 | 1 | 0 | 0 | 0 | 0 |  | |
| 0 | 1 | 0 | 0 | 0 | 0 |  | |

Figure 2. Obtained solutions for the configurations with the ground energy and the first excited energy and their corresponding configurations. $*$ denotes either 0 or 1. Four states take the energy $-1$ with the same configuration, and five states take the energy 0 with different legal configurations.

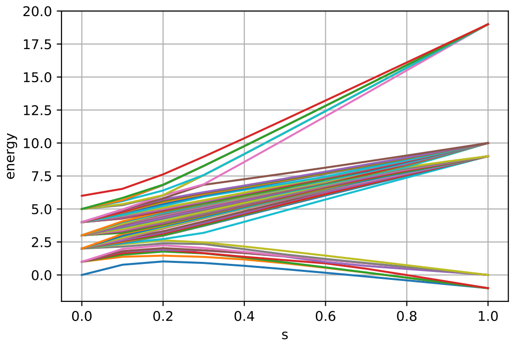


Figure 3. Computed eigenenergies during the annealing process with respect to the parameter $s$.

the expected native configuration, and it is irrelevant to the values of the last qubits which do not contribute to the energy function. The first excited states correspond to five possible legal states that do not minimize the energy. The other states with higher energies correspond to the illegal states such as the ones where either overlap or violation of the covalent bonds occurs. Fig. 3 shows the energy spectrum during the annealing process. It can be seen that some number of configurations having the same energy end up making bundles and each bundle gets separated into the energy corresponding to thier configurations at the end of the protocol. Interestingly, this degeneracy may help to reduce the error due to the thermal excitation because even if the initial ground state gets excited into an excited state at some point where the spectral gap becomes small, it will eventually go to the desired ground state as long as it jumps to the third excited state or less.

## V. CONCLUSIONS

We have introduced a new encoding scheme that maps a lattice protein folding problem to a Hamiltonian that can be implemented on a quantum annealer. It uses exponentially smaller number of qubits than the original diamond encoding method while keeping the simplicity of the construction of the Hamiltonian. The smaller use of the qubits is made possible at the cost of a higher level of non-locality comparing to the diamond encoding, which is inherently 2-local. Our construction is applicable to any interaction model from HP model to MJ model, and it can be also extended to 3D lattice in a straightforward manner. We applied our encoding scheme to a simple example and confirmed that the ground states indeed correspond to the native configuration that minimizes the energy. The degeneracy of the ground states inherent from the construction may make the process robust against the thermal noise.

Because of the space-based nature of the encoding, one could employ heuristic arguments saying that the folding tends to take place in rather a restricted region of the lattice [37–39]. It is a physically reasonable simplification but it cannot be directly applied to another qubit efficient encoding such as turn encoding [35]. It would be an interesting future work to investigate how our construction can be combined with such heuristic approaches.

It would be also interesting to try implementing our encoding on the most recent quantum device with 2048 qubits [40]. When Perdomo *et al.* performed their experiment, the quantum device back then only had 128 qubits (with 115 qubits being confirmed functional). Now that much more qubits are available, more complex proteins could be simulated, and our encoding may be suitable at such a large $N$ region because of the possible heuristic approach available that becomes more effective as the number of acids becomes large.

[1] G. D. Rose, P. J. Fleming, J. R. Banavar, and A. Maritan, Proceedings of the National Academy of Sciences **103**, 16623 (2006).

[2] C. B. Anfinsen, Science **181**, 223 (1973).

[3] E. Shakhnovich, M. Karplus, *et al.*, nature **369**, 248 (1994).

[4] G. M. Crippen, Biochemistry **30**, 4232 (1991).

[5] V. S. Pande, Physical review letters **105**, 198101 (2010).

[6] K. A. Dill, S. B. Ozkan, M. S. Shell, and T. R. Weikl, Annu. Rev. Biophys. **37**, 289 (2008).

[7] L. Mirny and E. Shakhnovich, Annual review of biophysics and biomolecular structure **30**, 361 (2001).

[8] E. I. Shakhnovich, Physical Review Letters **72**, 3907

(1994).

[9] A. Smith, Nature **426**, 883 (2003).

[10] J. Laurén, D. A. Gimbel, H. B. Nygaard, J. W. Gilbert, and S. M. Strittmatter, Nature **457**, 1128 (2009).

[11] S. B. Prusiner, Proceedings of the National Academy of Sciences **95**, 13363 (1998).

[12] M. D. Scott and J. Frydman, in *Protein misfolding and disease* (Springer, 2003) pp. 67–76.

[13] P. L. Freddolino, C. B. Harrison, Y. Liu, and K. Schulten, Nature physics **6**, 751 (2010).

[14] C. Levinthal, Mossbauer spectroscopy in biological systems **67**, 22 (1969).

[15] D. A. Beck and V. Daggett, Methods **34**, 112 (2004).

[16] M. Karplus and J. Kuriyan, Proceedings of the National Academy of Sciences of the United States of America **102**, 6679 (2005).

[17] D. A. Case, T. A. Darden, T. r. Cheatham, C. L. Simmerling, J. Wang, R. E. Duke, R. Luo, M. Crowley, R. C. Walker, W. Zhang, *et al.*, *Amber 10*, Tech. Rep. (University of California, 2008).

[18] A. D. MacKerell Jr, D. Bashford, M. Bellott, R. L. Dunbrack Jr, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, *et al.*, The journal of physical chemistry B **102**, 3586 (1998).

[19] K. F. Lau and K. A. Dill, Macromolecules **22**, 3986 (1989).

[20] B. Berger and T. Leighton, Journal of Computational Biology **5**, 27 (1998).

[21] P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis, Journal of computational biology **5**, 423 (1998).

[22] R. D. Schram, G. T. Barkema, and R. H. Bisseling, Journal of Statistical Mechanics: Theory and Experiment **2011**, P06019 (2011).

[23] L. Toma and S. Toma, Protein Science **5**, 147 (1996).

[24] H. S. Chan, Proteins: Structure, Function, and Bioinformatics **40**, 543 (2000).

[25] H. Kaya and H. S. Chan, Proteins: Structure, Function, and Bioinformatics **40**, 637 (2000).

[26] S. Miyazawa and R. L. Jernigan, Journal of molecular biology **256**, 623 (1996).

[27] A. Perdomo, C. Truncik, I. Tubert-Brohman, G. Rose, and A. Aspuru-Guzik, Physical Review A **78**, 012320 (2008).

[28] A. Perdomo-Ortiz, N. Dickson, M. Drew-Brook, G. Rose, and A. Aspuru-Guzik, Scientific reports **2**, 571 (2012).

[29] T. Kadowaki and H. Nishimori, Physical Review E **58**, 5355 (1998).

[30] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, arXiv preprint quant-ph/0001106 (2000).

[31] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, science **220**, 671 (1983).

[32] T. Albash and D. A. Lidar, Reviews of Modern Physics **90**, 015002 (2018).

[33] M. W. Johnson, M. H. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, *et al.*, Nature **473**, 194 (2011).

[34] R. Harris, M. Johnson, T. Lanting, A. Berkley, J. Johansson, P. Bunyk, E. Tolkacheva, E. Ladizinsky, N. Ladizinsky, T. Oh, *et al.*, Physical Review B **82**, 024511 (2010).

[35] R. Babbush, A. Perdomo-Ortiz, B. O'Gorman, W. Macready, and A. Aspuru-Guzik, arXiv preprint arXiv:1211.3422 (2012).

[36] A. Messiah, *Quantum Mechanics [Vol 1-2].* (1964).

[37] D. Baker, Nature **405**, 39 (2000).

[38] M. T. Oakley, D. J. Wales, and R. L. Johnston, The Journal of Physical Chemistry B **115**, 11525 (2011).

[39] E. I. Shakhnovich, Folding and Design **1**, R50 (1996).

[40] E. Gibney, Nature News **541**, 447 (2017).