Quantum Strategies in Protein Folding Simulations

Krishna Balaji S[1,*] and Manoj NH[2,†]
[1] *Department of Physics, IIT Madras,Chennai, India*
[2] *Department of EE, IIT Madras,Chennai, India*

Protein folding remains a computationally challenging problem due to the size of the conformational space and the ruggedness of the energy landscape. Coarse-grained lattice models and modern quantum approaches offer promising strategies. In this report, we present a concise overview of some quantum treatments of protein folding using simplified models, while also building upon existing methods to come up with alternative approaches to undestand the process of protein folding.

I. THE PROTEIN FOLDING PROBLEM

Proteins are the molecular workhorses of life, driving biochemical reactions, providing structural support, transporting molecules, and regulating nearly every biological process. As essential components of life, proteins are integral to the Central Dogma of biology—where DNA encodes genetic instructions, RNA copies those instructions for translation, and proteins that emerge as the final product of the translation process execute these instructions.

Each protein is synthesized in cells as a linear sequence of amino acids, dictated by DNA's genetic code. However, this one-dimensional chain must fold into a specific three-dimensional structure to become biologically active. This transformation, known as **protein folding**, is the key step in creating functional proteins and even slight errors can result in dysfunction or mis-folding diseases, making the protein folding problem central to both fundamental biology and medical research. It is also much easier to determine the sequence of an unknown amino acid chain as compared to its native structure.

The **protein folding problem** refers to the challenge of predicting a protein's native three-dimensional structure from its amino acid sequence. This folding is governed by a complex interplay of intramolecular forces— such as hydrogen bonds, hydrophobic interactions, and electrostatic attractions— and occurs spontaneously in a matter of milliseconds to seconds. The exact pathway a protein follows through its 'energy landscape' of vast conformations that are possible due to the two freely rotatable Carbon-Carbon bonds per amino acid to reach its native folded structure is deeply intricate and still not fully understood.

Current approaches to solving this problem can be broadly split into two categories: **Simulators** and **predictors**. Current approaches (particularly those based on machine learning such as AlphaFold) have demonstrated impressive accuracy on *predicting* well-characterized proteins. These tools rely heavily on pattern recognition from massive biological datasets, making them highly effective at predicting structures that resemble known proteins. Consequently, they struggle with novel sequences that differ significantly from training data, cannot robustly model intermediate states or folding pathways that are of great therapeutic interest, and are blind to subtle misfoldings that may still yield deceptively similar final structures. This limits their utility in studying protein dynamics, folding kinetics, and conditions caused by misfolded proteins such as Alzheimer's and Parkinson's.

However, simulating the folding process classically based on molecular dynamics is expensive in time and resources; ANTON-2, a specialized supercomputer made specifically for this task, can simulate only 59.4 μs of folding time per day (proteins fold over the scale of milli-seconds). Decentralized attempts such as Folding@Home, which crowd sourced compute for the project, are not scalable to the point of widespread use in therapeutics and custom protein design either.

This is where quantum computing has the potential to offer an advantage. Protein folding is fundamentally a quantum mechanical process, governed by interactions such as hydrogen bonding, van der Waals forces, and electrostatics—all of which are inherently quantum in nature. Classical simulations of these interactions scale poorly due to the combinatorial explosion of possible conformations. Quantum computers, however, can represent and explore complex superpositions of folding states more naturally and efficiently by making use of quantum parallelism and superposition. By integrating quantum approaches into the modeling of folding pathways—rather than just final structures—we aim to bridge the gap between predictive accuracy and mechanistic understanding, opening the door to novel insights in structural biology and therapeutic design.

In this project, as a proof of concept, we simulate the protein folding mechanics onto proteins fit on a **lattice models** and then also explore a new **'contact map based' encoding scheme** that allows us to simulate the folding pathway without involving molecular

[*] ep21b021@smail.iitm.ac.in
[†] ee23b044@smail.iitm.ac.in

dynamics simulations. Both these approaches can be run on present day NISQ quantum device and involve discretization of the problem in some way to make it tractable to computers (both classical and quantum).

II. LATTICE MODELS AND DO THEY REALLY WORK

Lattice models are simplified models for protein structure that represent them by placing amino acid residues on the vertices of a lattice. This coarse-grained approach simplifies the complex three-dimensional folding problem by reducing the conformational space to a set of self-avoiding walks on the lattice. Each lattice vertex can represent one or more residues, and the protein backbone is modeled as a chain passing through consecutive lattice points.

A critical question is whether such simplified lattice models can capture the essential features of native protein folds. In the seminal work by Hinds and Levitt [1], they've demonstrated that despite its simplicity, lattice models can effectively discriminate native-like structures from non-native ones. This is highlighted by the *alignment-optimization* strategy.

In this approach, the amino acid sequence is aligned onto a given lattice path by assigning short contiguous segments of residues to each lattice vertex. The contact energy between two lattice vertices is computed as the average of the pairwise residue-residue contact energies between all residues assigned to those vertices:

$$C_{ij} = \frac{1}{4}\left(e_{r_{m_i} r_{m_j}} + e_{r_{m_i} r_{m_j+1}} + e_{r_{m_i+1} r_{m_j}} + e_{r_{m_i+1} r_{m_j+1}}\right), \tag{1}$$

where r_m indicates the residue type at position m in the sequence. By optimizing this alignment, the lattice model is allowed to explore a reduced conformational space while still preserving the overall low-energy nature of its structure, thereby enabling it to identify native-like folds.

The unreasonable effectiveness of such an analysis motivated the pursuit of modelling proteins on a lattice by considering just two types of molecules on each site, hydrophobic (H) and polar (P) called the HP model.

III. THE HP MODEL

Given a sequence of amino acids, we can define a coarse-grained version of the chain by dividing sections of its chain into Hydrophobic (H) and Polar (P) depending on nature of the amino acids present. This is thus presented as a "HP sequence". The HP model is a simplistic model for protein on a lattice, where the protein is represented as a self-avoiding chain of hydrophobic (H) and polar (P) sites residing on a 2D lattice. Although there exist many ways to implement the same, we discuss a model that can be easily formulated as a QUBO here. The interaction energy is given by,

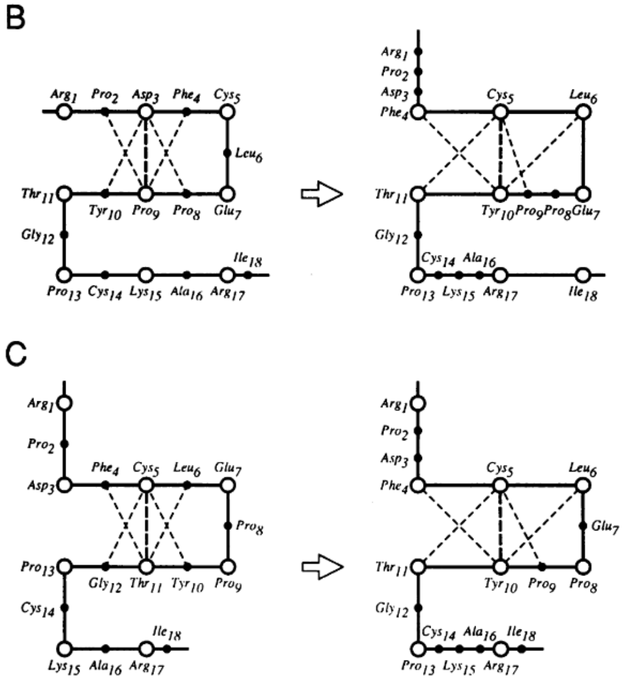

FIG. 1. Example of the alignment-optimization procedure on a simple amino acid chain. The Alignment procedure assigns a set of amino acids to lattice points and uses these sites to compute contact energies. Starting at two different initial conditions on a lattice, by allowing only shifts of amino acids along lattice points, it was found that a set of these converge to a mapping which favors location of hydrophobic AAs on lattice sites due to very strong nature of its interaction.

$$E = \sum_{f<f'} U(h_f, h_{f'}) C_{ff'} \tag{2}$$

where h_f denotes the type of bead (H or P) at position f in the sequence.

Essentially, the interaction energy is considered when two beads on the chain that are not nearest neighbors reside in adjacent positions on the lattice (given by the function $C_{ff'}$, which equals 1 when the beads are nearest neighbors on the lattice but not on the chain, i.e., $|f - f'| > 1$, and is 0 otherwise).

The function $U(h_f, h_{f'})$ determines the interaction energy, which is set to -1 if h_f and $h_{f'}$ are both H (modelling the attractive interaction between Hydrophobic molecules), and 0 otherwise. Hence, the energy of the protein directly depends on the number of nearest-neighbor (NN) HH contacts.

IV. FORMULATING THE HP QUBO

Let the HP sequence be given as $(h_1, \ldots, h_N)$, where $h_i \in \{\mathrm{P}, \mathrm{H}\}$. To define a QUBO, we first construct a set of binary field variables by dividing the $L \times L$ lattice into a checkerboard pattern alternating between $+1$ and -1.

We make the observation that, from parity, if we start with the first HP bead on a lattice site that is assigned $+1$, then every odd-numbered bead is assigned to lattice sites with $+1$, and every even-numbered bead belongs to a site with -1. For the purpose of modelling, the sites are flattened into a 1D array, so that the parity of the array index matches the parity of the corresponding HP chain index.

This allows us to define two binary variables σ_s^f and $\sigma_{s'}^{f'}$, where s, f run over even indices (i.e., $0, 2, \ldots$) and s', f' run over odd indices (i.e., $1, 3, \ldots$). We set $\sigma_s^f = 1$ if bead f is located at site s, and 0 otherwise. Similarly, $\sigma_{s'}^{f'} = 1$ if bead f' is at site s', and 0 otherwise.

Hence, the total energy E_{HP} can be written as:

$$E_{\mathrm{HP}} = -\sum_{|f-f'|>1} C(h_f, h_{f'}) \sum_{\langle s,s' \rangle} \sigma_s^f \, \sigma_{s'}^{f'}, \tag{3}$$

To put it in simple terms, the function $C(h_f, h_{f'}) = 1$ if and only if both locations f and f' on the chain are occupied by hydrophobic (H) beads. For every such non-NN pair on the chain, we sum over the product $\sigma_s^f \, \sigma_{s'}^{f'}$ over all NN pairs of lattice sites $\langle s, s' \rangle$.

As discussed earlier, this specialized function ensures that only contributions from NN H-H contacts on the lattice (that are not adjacent on the chain) are included in the energy computation.

We now define a set of constraints that the chain must follow. Each of the constraint is enforced by an energy term $E_i, i = 1, 2, 3$ which is added as a penalty to our interaction energy E_{HP} term.

a. Each bead must belong to only one site:

$$E_1 = \sum_f \left(\sum_s \sigma_s^f - 1 \right)^2 + \{\text{same for odd parity}\}, \tag{3}$$

b. Each site must only contain one bead (self-avoiding condition):

$$E_2 = \frac{1}{2} \sum_{f_1 \neq f_2} \sum_s \sigma_s^{f_1} \sigma_s^{f_2} + \{\text{same for odd parity}\}, \tag{4}$$

This term penalizes multiple beads occupying the same site.

c. Connectivity of the chain must be preserved: This is enforced by the energy term E_3, which introduces a penalty for every time two NN beads on the HP chain do not belong to NN sites on the lattice:

$$E_3 = \sum_{1 \leq f < N} \left(\sum_s \sigma_s^f \sum_{|s'-s|>1} \sigma_{s'}^{f+1} \right) + \sum_{1 \leq f' < N} \left(\sum_{s'} \sigma_{s'}^{f'} \sum_{|s-s'|>1} \sigma_s^{f'+1} \right), \tag{5}$$

The total energy is thus given by

$$E = E_{\mathrm{HP}} + \sum_{i=1}^{3} \lambda_i E_i, \tag{4}$$

where each of the above Energy terms E_{HP}, E_i are quadratic in the binary field variables σ_s^f and $\sigma_{s'}^{f'}$, which makes the total energy function E a quadratic function of the binary variables σ_s^f and $\sigma_{s'}^{f'}$ and can hence be formulated as a QUBO in these variables.

V. BUILDING UPON THE HP MODEL: INTRODUCING AMINO ACID INTERACTIONS

In our work, we build upon the HP model introduced in [2] by extending the input to our folding model to represent a general amino acid chain on a lattice. In doing so, we introduce interaction terms whose strengths are not assumed to be uniform (as in the HP model), but instead vary between different amino acid pairs.

To model these interaction energies, we've adopted the well-known Miyazawa–Jernigan (MJ) model for nearest-neighbor (1-NN) interactions [3]. The interaction energies are input as a matrix sourced from [4], which compiled these values directly from the original MJ interaction energy data [3].

The key distinction when using the MJ matrix over the HP model lies in the dependence of the interaction energies on nature of Amino acid at the bead. Unlike the binary classification in the HP model (hydrophobic or polar), the MJ matrix assigns a unique interaction energy to each amino acid pair, thereby allowing for a more realistic and detailed modeling of protein folding behavior.

VI. PARAMETERS FOR QA ON A LATTICE MODEL

For modelling the folding of a chain of length N on a 2D lattice, we expect the use of NL^2 spins (or $NL^2/2$ in our above formulation that exploits parity), where $L \times L$ is the size of the 2D square lattice. The longest possible value of N that will be needed is proportional to L (when the chain is completely stretched out).

However, we can reduce the conformational space by restricting the value of L. The intuition for this can be obtained from a rigorous study of polymer configurations on a 2D lattice, which is modelled similarly to an ideal chain performing a self-avoiding walk. Analysis of these configurations using statistical mechanics [5] reveals that for the set of physical configurations of the polymer, the radius of gyration of the polymer chain (averaged over all configurations) scales with the chain length as $N^{0.5}$. This suggests that a chain of non-interacting beads on a lattice occupies an area of the order of the chain length.

We can then propose that for a similar self-avoiding walk with only attractive interactions, the scaling would be better than $N^{0.5}$. This provides motivation for choosing the lattice size L in our simulation of an N-bead HP sequence as

$$L \sim k\sqrt{N}$$

as opposed to $L \sim N$, which would account for all possible configurations. k can whence be treated as a hyperparameter, and in this work we have used existing knowledge of some HP configurations as per [2] to fix a suitable value of the grid directly.

VII. QA ON MJ-LATTICE MODELS

A. Folding Angiotensin-II

Angiotensin-II is an octapeptide (one of the many Angiotensin peptides) with the primary sequence: Asp–Arg–Val–Tyr–Ile–His–Pro–Phe (DRVYIHPF). Due to its small size and variety of inter-amino-acid interactions (due to presence of polar, aromatic, and hydrophobic residues), we decided this would be the perfect candidate to test our lattice-based Folding model on.

We attempt to fold the Angiotensin-II peptide using our MJ-based lattice QUBO formulation on a 4 × 4 2D grid. The folding is performed using D-Wave's ocean quantum annealing (QA) solver, with penalty strengths $\lambda_i = 15, i = 1, 2, 3$ introduced via manual trial and error considering the order of magnitude of the MJ interaction. The QUBO matrix thus encodes these penalty violations as well as the total MJ interaction energies of all possible chains.

The annealer returned a configuration with a total MJ interaction energy of **-15.89**, corresponding to a physically favorable conformation. As shown in Figure 1, the folded state shows multiple favorable contacts, of which the notable ones are a hydrophobic interaction between tyrosine (Y) and valine (V), while there is also presence of polar interaction from Arginine (R) with the same neighbour valine (V). The folding also allows Isoleucine (I) and Phenyl alanine (F) to interact through their strong hydrophobic interactions. We notice that, although there are many low-energy configurations due to different kinds of interactions in this large molecule, specific configurations of folding can only be physically predicted with more than a binary information at every lattice (that of H/P) into account, with the above fold being a good example of the same.

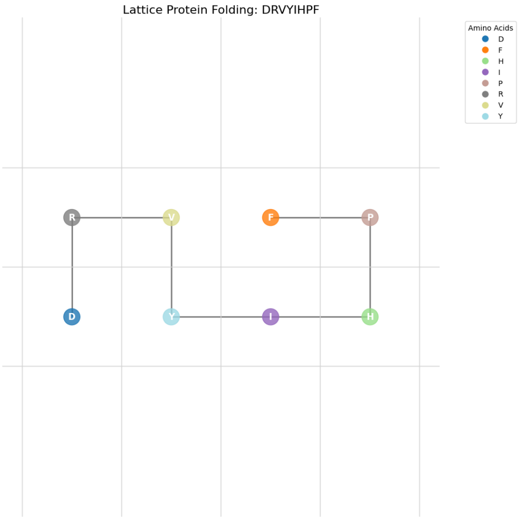

FIG. 2. Best folded conformation result Angiotensin-II on a 4 × 4 grid

Another observation with this result is that it allows specific interactions (eg: IF to dominate over other configurations (eg, IR) due to the stronger hydrophobic interactions between Isoleucine and Phenylalanine). This kind of nuance helps us filter out a large number of configurations and helps realize a more realistic contact matrix when using actual MJ interactions with varying strengths, over a simple binary HP model.

VIII. CHALLENGES IN QA FOR LATTICE APPROACHES

A major bottleneck in quantum annealing for HP models is that practical proteins lie in the range of amino acid chains of length 50 to 3000; this problem becomes intractable not only with classical computing (due to exponential increase in simulation time) but also with the present state of quantum annealers (exponential increase in number of noisy qubits). These lattice models typically scale as

$$\text{Number of qubits} \sim ND \log N$$

where N is the number of amino acids and D is the dimensionality of the lattice. For $N = 50$ and $D = 3$, this yields a requirement over 1000 logical qubits for fully neat computation, which is beyond the capacity of present annealers. Although alternative encoding approaches exist, noise in physical qubits still play a role as we notice from the different solutions given by the machine throughout the course of our simulation.

IX. CONCLUSION

Therefore, while lattice-based QUBO formulations provide a compact and structured approach to modeling protein folding, their practical implementation on near-term quantum annealers remains a significant challenge.

Thus, in the NISQ era of computing, work in this domain may require hybrid quantum-classical strategies that overcome much of the errors by incorporating a classical feedback system that can account for errors such as decoherence in physical qubits. More feasible near-term approaches involve quantum simulations using NISQ-friendly algorithms such as the Variational Quantum Eigensolver (VQE), which we will explore in subsequent work.

X. THE PATHWAY FINDING PROBLEM

While predicting the final folded structure of a protein is a significant challenge in itself, understanding the 'path' it takes is an even deeper and largely unsolved problem. Despite the large number of possible conformations, most proteins fold reliably into their native structure within milliseconds to seconds, implying the existence of efficient, biologically evolved pathways through the conformational energy landscape. A folding pathway is a sequence of intermediate conformations that a protein adopts as it moves from its initial unfolded state to its functional native structure. These intermediates are stabilized by partial interactions and follow energetically favorable transitions. Mapping such a pathway is crucial for understanding folding kinetics, detecting misfolding or aggregation-prone states (as seen in diseases like Alzheimer's), and designing synthetic proteins or drugs that target folding intermediates. The 'predictive' methods described in the introduction section can only arrive at the final structure with no insight at the possible stages misfolding could occur; hence I have included this problem in the project to highlight the advantages Simulation based approaches have.

From a computational perspective, pathway finding is difficult. Classical methods, like molecular dynamics (MD), attempt to simulate this directly by solving equations of motion for atoms over time. However, MD simulations are limited by the size of the time steps (on the femtosecond scale) and the massive number of steps needed to reach biologically relevant folding timescales, making the process computationally infeasible for anything but small, fast-folding proteins. An approach called **'Graph Based Sampling'** was proposed by Ziad Fakhoury et. al. in 2023 [6] that sidesteps MD simulations to find the folding pathway.

In this new approach, an N-amino acid protein structure is expressed as a 'Contact Map' $\mathbf{G}$, where each element $\mathbf{G}_{ij} \in \{0,1\}$; $\mathbf{G}_{ij} = 1$ if amino acid residues i and j are within a threshold in euclidean space. The path to find now is the sequence of states $\{\mathbf{G}_0, \mathbf{G}_1, ..., \mathbf{G}_T\}$ where $\mathbf{G}_T$ is the final structure's contact map. Adding constraints over how the states can evolve, the search space to find the optimal sequence is vastly narrowed down. An important point to note is that this approach is **independent of the specific geometric representation or lattice model** we use on our protein.

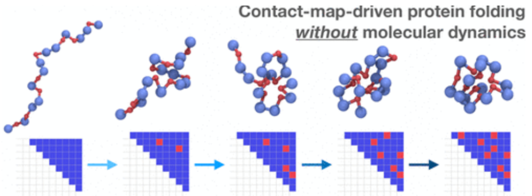
Contact-map-driven protein folding
without molecular dynamics

FIG. 3. Depiction of a Contact map sequence and their corresponding structure

Note $\mathbf{G}_{ij} = \mathbf{G}_{ji}$ and $\mathbf{G}_{i,i+1} = 1$ trivially by geometry. This implies $\mathbf{G}$ is a symmetric matrix, with the elements adjacent to the diagonal always being 1. Also, residue-i cannot connect to residue-$i+2$ (this is a steric constraint, such a connection would impose a sharp angle at residue-i). This implies $\mathbf{G}_{i,i+2} = 0$. Now the only free variables remaining are $\{\mathbf{G}_{ij} | \quad 1 \leq i \leq N-2; i+2 \leq j \leq N\}$. Taking this over '$T-1$' intermediate timesteps (because $\mathbf{G}_0$ and $\mathbf{G}_T$ are fixed) , we get $(T-1)(N-3)(N-2)/2$ free variables for the pathway finding problem.

XI. QUBO FORMULATION

The referenced paper used Simulated Annealing over the free variables to find the optimal path. I realized the same can implemented with a QUBO formulation expressed below:

1. **Steric Constraints:** Each amino acid residue can only interact with a maximum of two other residues apart from the ones it is adjacent to on the chain. This implies the sum of the free variables in each column and row should not exceed 2. The corresponding cost function is:

$$C_{steric} = \sum_{t=1}^{T} \left(\sum_{i=1}^{N-3} (d_{i,t} - 1)^2 + \sum_{j=1}^{N-3} (d_{j,t} - 1)^2 \right)$$

$$d_{i,t} = \sum_{j=i+3}^{N} \mathbf{G}_{t,ij}, \quad d_{j,t} = \sum_{i=j+3}^{N} \mathbf{G}_{t,ij}$$

this penalizes structures with more than 2 connections per residue and slightly favors 1 connection per residue.

2. **Smoothness and Continuity:** As we are solving for the entire path $\{\mathbf{G}_0, \mathbf{G}_1, ..., \mathbf{G}_T\}$ in one go, we need a constraint to ensure any two adjacent states differ only slightly to represent valid protein structure evolutions.

$$C_{smooth} = \sum_{t=1}^{T} \sum_{i=1}^{N-3} \sum_{j=i+3}^{N} (\mathbf{G}_{t,ij} - \mathbf{G}_{t-1,ij})^2$$

3. **Amino Acid interactions:** To account for the stabilizing (attraction) and destabilizing (repulsing) effect of interactions between amino acids:

$$C_{inter} = \sum_{t=1}^{T} \sum_{i=1}^{N-3} \sum_{j=i+3}^{N} \mathbf{G}_{t,ij} * h_{ij}$$

where h_{ij} is positive for destabilizing interactions and negative for stabilizing ones; the magnitude is determined by the strength of the interaction and can be taken from the Miyazawa–Jernigan table. If a toy H/P model is being considered, h_{ij} is a negative value iff residues i and j are both hydrophobic and $h_{ij} = 0$ otherwise.

The final cost function to be implemented is then

$$C_{QUBO} = C_{steric} + \lambda_1 C_{smooth} + \lambda_2 C_{inter}$$

where λ_1 and λ_2 can be fine tuned based on relative impact of the costs.

XII. RESULTS

We determined the pathway for a 6 amino acid H/P chain with 2 timesteps which agreed with the simulated annealing result from the paper;
Toy Protein model used: PPHPPH, residues 3 and 6 are hydrophobic.
Final Contact Map: (1,6),(3,6) (these residues are expected to be connected in the final structure)
Contacts after Timestep1: (3,6)
Contacts after Timestep1: (1,6), (3,6)
Note that the intermediate state happened to be one that was stabilized by the favorable interaction between residues 3 and 6 as expected.

Notably, the simulated annealing procedure crashed on Google's Colab instances due to RAM shortage after using all the 12GB available RAM for $N \geq 9$; the D-Wave annealer was tested upto $N = 11$ with 4 timescales. Also, the time taken for the Simulated annealing procedure to converge increased significantly between$N = 4, 5, 6$ while the runtime on the quantum annealer did not change much for these values as expected because the annealing time on a Q. annealer is **constant**.

	SA Runtime	**QA Runtime**
$N = 4, T = 2$	2s	100ms
$N = 5, T = 2$	4s	112ms
$N = 6, T = 2$	11s	109ms

TABLE I. **Comparision of runtimes of SA and QA processes**, taken from the time taken execute the cell on Google Colab. The QA runtime includes anneal and postprocessing time. Note, for $N \geq 8$, T would have to exceed 2 to have enough intermediate structures to indicate smooth transitions; SA would perform even poorly in these cases.

XIII. IMPLICATIONS AND FURTHER WORK

The referenced paper (Fakhoury et. al., 2023) presents a complete end to end procedure to find the pathway as a sequence of contact maps and then work out the structures of the intermediates through the map sequence. We have implemented the key part of this process, finding the sequence of maps, on a quantum device. This is different from the past works on using Quantum Computing on the folding problem, which are all restricted to some lattice model.

In a follow up paper, the authors reproduced two different pathways a synthetic protein (called 'G/L') took to reach its native state and cross referenced it to the pathways predicted by fine grained MD simulations. This cannot be done on predictive approaches for the protein folding problem, providing a concrete example for the utility of simulation based approaches over predictive ones. This is something that can be tested out on the q. annealer further.

It is also suggested in the paper that the model can be modified to accommodate an undefined T parameter, i.e, we let the model evolve freely until it matches with the final structure. This can also be implemented on D-Wave's system as they allow the user to choose an initial starting wavefunction: Evolve $\mathbf{G}_0$ into $\mathbf{G}_1$, initialize the system to $\mathbf{G}_1$ and then evolve it to $\mathbf{G}_2$ and so on. This is in contrast to out current implementation, where we solved for (T-1) different sets of parameters simultaneously, and has a better resource scaling useful for larger proteins. Naturally, if the model were to be fine tuned enough, one can speculate that it can converge to the final structure **even without** the knowledge of the final structure, essentially also solving the protein folding problem.

[1] D. A. Hinds and M. Levitt, Proceedings of the National Academy of Sciences of the United States of America **89**, 2536 (1992), communicated by Aaron K. Mug, December 23, 1991.

[2] A. Irbäck, L. Knuthson, S. Mohanty, and C. Peterson, Phys. Rev. Res. **4**, 043013 (2022).

[3] S. Miyazawa and R. L. Jernigan, Journal of Molecular Biology **256**, 623 (1996), pMID: 8604144.

[4] A. Robert, P. K. Barkoutsos, S. Woerner, and I. Tavernelli, npj Quantum Information **7**, 38 (2021).

[5] P.-G. de Gennes, *Scaling Concepts in Polymer Physics* (Cornell University Press, Ithaca and London, 1979) a foundational text introducing scaling laws in polymer physics.

[6] H. S. Fakhoury Z, Sosso GC, Journal of Chemical Information and Modeling 10.1021/acs.jcim.3c00023 (2023).