Construction of Model Hamiltonians for Adiabatic Quantum Computation and its Application to Finding Low-Energy Conformations of Lattice Protein Models

On the construction of model Hamiltonians for adiabatic quantum computation and its application to finding low energy conformations of lattice protein models

Alejandro Perdomo,[1] Colin Truncik,[2] Ivan Tubert-Brohman,[1] Geordie Rose,[2] and Alán Aspuru-Guzik[1]

[1]*Department of Chemistry and Chemical Biology, Harvard University, 12 Oxford Street, 02138, Cambridge, MA*

[2]*D-Wave Systems, Inc., 4401 Still Creek Drive, Suite 100, Burnaby, BC V5C 6G9, Canada*

Abstract

In this report, we explore the use of a quantum optimization algorithm for obtaining low energy conformations of protein models. We discuss mappings between protein models and optimization variables, which are in turn mapped to a system of coupled quantum bits. General strategies are given for constructing Hamiltonians to be used to solve optimization problems of physical/chemical/biological interest via quantum computation by adiabatic evolution. As an example, we implement the Hamiltonian corresponding to the Hydrophobic-Polar (HP) model for protein folding. Furthermore, we present an approach to reduce the resulting Hamiltonian to two-body terms gearing towards an experimental realization.

PACS numbers: 87.15.Cc, 03.67.Ac, 05.50.+q, 75.10.Nr

I. INTRODUCTION

Finding the ensemble of low-energy conformations of a peptide given its primary sequence is a fundamental problem of computational biology, commonly known as the protein folding problem [1, 2, 3, 4, 5, 6, 7]. The native fold conformation is usually assumed to correspond to the global minimum of the protein's free energy (according to the so-called thermodynamic hypothesis [8]), although some exceptions have been proposed [9, 10]. Thus, the protein folding problem can be described as a global optimization problem. Algorithms for quantum computers have been developed for many applications such as factoring [11] and the calculation of molecular energies [12]. In this report, we investigate the approach of using an adiabatic quantum computer for folding a highly simplified protein model.

The HP (H: hydrophobic, P: polar) lattice model [13] is one of the simplest protein models implemented. Still its accuracy in predicting some of the folding behaviour of real proteins has made it a useful benchmark for testing optimization algorithms such as simulated annealing [14], genetic algorithms [15, 16, 17, 18, 19], and ant colony optimization [20]. Other heuristic methods such as hydrophobic core threading [21], chain growth [22, 23], contact interactions [24], and hydrophobic zippers [25] have also been considered. The HP model has also been useful for a qualitative investigation of the nature of the folding process and the interactions between proteins. The HP model depicted in Fig. 1 is defined by three assumptions: 1) There are only two kinds of amino acids or residues, hydrophobic (H) and polar (P); 2) residues are placed on a grid (typically a square grid for the 2D model and a cubic grid for the 3D model); 3) the only interaction among amino acids is the favorable contact between two H residues that are not adjacent in the sequence. The energy of this interaction is defined as -1 in arbitrary units, representing a hydrophobic effect which tends to fold the protein in a way that aggregates the H residues in a predominantly hydrophobic core, and leaves the P residues at the surface of the protein. The search for the native conformation of the protein is represented by a self-avoiding walk on the grid.

An important property of the model is that the number of possible conformations is roughly proportional to 2.7^N [13], where N is the length of the polypeptide chain. Proofs of the NP-completeness of both the 2D and 3D HP models have been given [26, 27]. Due to this exponential growth, global optimality proofs become impractical when N reaches approximately 50 residues. For longer sequences, heuristics and stochastic algorithms have

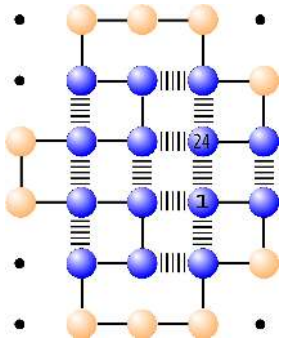

FIG. 1: (Color online) The lattice protein hydrophobic-polar (HP) model, showing the global energy minimum conformation for a sequence of 24 amino acids, HHPHPPPHHHHPPHHHHPPPHPHH ($E = -12$). Blue (dark grey) beads represent hydrophobic residues (H) and orange (light grey) beads represent polar residues (P). The model consists of a self-avoiding chain with favorable ($E = -1$) energetic interactions among hydrophobic residues in contact. Contact between nearest neighbors in the primary sequence are unavoidable, and their contribution is not added to the calculated energy. Black dots represent lattice sites. Dotted lines represent favorable energetic interactions, solid lines represent the self-avoiding chain.

been employed for N up to 136 for the 3D HP model [24].

This report is structured as follows. Sec. II presents the general quantum algorithm and the terms of the Hamiltonian necessary to obtain the folded structure of the protein, and describes how to map the problem to arrays of coupled quantum bits [28, 29]. Sec. III explains the construction of the core component of the algorithm, the Hamiltonian that encodes the lowest energy conformation of the protein. In Sec. IV we solve in detail the four amino acid sequence HPPH in a two-dimensional grid. In Sections V and VI we discuss the resources necessary to carry out the reduction from a general k-body Hamiltonian to a two-body Hamiltonian, as a function of the size of the protein.

II. AN ADIABATIC QUANTUM ALGORITHM FOR THE HP MODEL

We begin this section by describing the mapping of a sequence of N amino acids into binary variables, which will in turn be mapped to spin variables in the quantum mechanical version of the algorithm.

A. Mapping amino acids onto a lattice

The mapping of the coordinates of a sequence of N amino acids to a given grid of size $N \times N$ is developed as follows. We assume, without loss of generality, that the number of amino acids is a power of 2. A binary representation for the labels of the grid requires $\log_2 N$ binary variables to specify the position of an amino acid in each dimension, as shown in Fig. 2. The position of each of N amino acids in a D-dimensional lattice may thus be encoded by a bit string q composed of exactly $DN \log_2 N$ binary variables q_i. For example, for $N = 4$, $D = 2$, the length of the bit string q is 16 and therefore the number of configurations that can be explored is 2^{16}. Let q denote a particular configuration of the protein in the grid, written in the form

$$q = \underbrace{q_{16}q_{15}}_{y_4}\ \underbrace{q_{14}q_{13}}_{x_4}\ \underbrace{q_{12}q_{11}}_{y_3}\ \underbrace{q_{10}q_9}_{x_3}\ \underbrace{q_8q_7}_{y_2}\ \underbrace{q_6q_5}_{x_2}\ \underbrace{q_4q_3}_{y_1}\ \underbrace{q_2q_1}_{x_1}, \tag{1}$$

where x_i and y_i are the x and y coordinate of the i-th amino acid. Fig. 2 shows an example of the coordinate mapping given a specific sequence of residues or amino acids.

In the quantum version of the problem, these configurations span a Hilbert space of dimension 2^{16}. The state vectors can be written as

$$|q\rangle \equiv |q_{16}\rangle\, |q_{15}\rangle \cdots |q_2\rangle\, |q_1\rangle\,. \tag{2}$$

We wish to implement a Hamiltonian which encodes the ground state of the protein on a spin-1/2 quantum computer [30], or, in particular onto an Ising-like Hamiltonian with a transverse magnetic field [31] (see Sec. II B). To do so, we realize the 16-qubit Hilbert space as a system of 16 spin-1/2 particles, with $|q_i = 0\rangle$ mapped to the spin state $|\sigma_i^z = +1\rangle$ and $|q_i = 1\rangle$ mapped to $|\sigma_i^z = -1\rangle$, with these spin states as the computational basis. In other words, the quantum version of the configuration states is related to spin variables through the transformation

$$\hat{q}_i \equiv \frac{1}{2}(I - \hat{\sigma}_i^z), \tag{3}$$

with $I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ and $\sigma^z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$, the identity operator and the σ^z Pauli matrix represented in the computational basis, respectively.

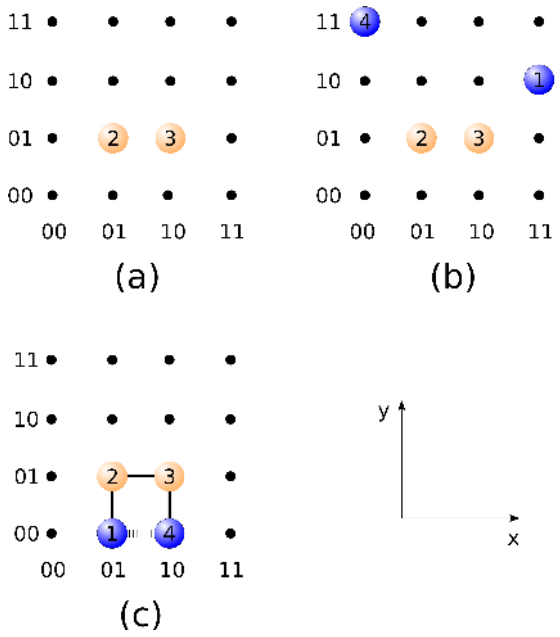

FIG. 2: (Color online) Grid-labeling conventions for a sequence of 4 amino acids, HPPH. **(a)** Amino acids 2 and 3 are fixed in the center of the grid to eliminate translational degeneracy. **(b)** One of the possible invalid configurations that might arise in the search and that would need to be discarded by the optimization algorithm. **(c)** Lowest-energy conformation for this example. The dotted line between amino acids 1 and 4 represents the hydrophobic interaction favored by the HP model. The configurations to optimize assume the form $q = q_{16}q_{15}q_{14}q_{13}$ 0110 0101 $q_4q_3q_2q_1$, where the set of variables $q_{16}q_{15}q_{14}q_{13}$ and $q_4q_3q_2q_1$ determine the position of amino acids 4 and 1, respectively. For the particular case in (b), $q =$ 1100 0110 0101 1011.

In Sec. III we will derive an energy function in terms of the $ND\log_2 N$ binary variables used to describe all of the possible configurations for the N amino acids in a D-dimensional lattice. This energy function is constructed so that its minimum will yield the lowest-energy conformations of the protein. Eq. 3 provides the rule for the mapping of this energy function to a quantum Hamiltonian. Each q_i in the energy function will be replaced by an operator $\hat{q}_i$. The operator $\hat{q}_i$ is to be understood as a short hand notation for a quantum operator acting on the i-th qubit of the $ND\log_2 N$ multipartite Hilbert space, $\mathcal{H}_{NDlog_2N} \otimes \mathcal{H}_{NDlog_2N-1} \otimes \cdots \otimes \mathcal{H}_i \otimes \cdots \otimes \mathcal{H}_1$. The explicit form of $\hat{q}_i$ is given by $I \otimes I \otimes \cdots \otimes \hat{q}_i \otimes \cdots \otimes I$. Notice that the operator $\hat{q}_i$ as defined in Eq. 3 has been placed in the i-th position, and the identity operator acts on the rest of the Hilbert space. Products of the form q_iq_j will be replaced by

a quantum operator $\hat{q}_i\hat{q}_j$, which is a shorthand notation for the operators $\hat{q}_i$ and $\hat{q}_j$ acting on the i-th and the j-th qubits, respectively. As an illustrative example, consider an energy function dependent on four binary variables,

$$E(q_1, q_2, q_3, q_4) = 1 - q_1q_2 + q_1q_3 + q_2q_3q_4,$$

which will be mapped to a Hamiltonian acting on a four qubit Hilbert space, $\mathcal{H}_4 \otimes \mathcal{H}_3 \otimes \mathcal{H}_2 \otimes \mathcal{H}_1$. In the instance of this particular energy function the Hamiltonian will assume the form

$$\begin{aligned}\hat{H} &= I \otimes I \otimes I \otimes I - I \otimes I \otimes \hat{q} \otimes \hat{q} + I \otimes \hat{q} \otimes I \otimes \hat{q} + \hat{q} \otimes \hat{q} \otimes \hat{q} \otimes I \\ &\equiv I - \hat{q}_1\hat{q}_2 + \hat{q}_1\hat{q}_3 + \hat{q}_2\hat{q}_3\hat{q}_4. \end{aligned} \tag{4}$$

Following this mapping, transformation of any energy function to the quantum Hamiltonian is straightforward.

In order to eliminate redundancy due to translational symmetry, we fixed the two middle amino acids in a central position (see Fig. 2). This reduces the number of binary variables in the bit string from sixteen to eight. The variables corresponding to amino acids 1 and 4: $q_4q_3q_2q_1$ and $q_{16}q_{15}q_{14}q_{13}$, respectively, become the variables of interest, and the variables $q_8q_7q_6q_5$ and $q_{12}q_{11}q_{10}q_9$ corresponding to amino acids 2 and 3, become constant throughout the optimization process. In general, the $(N/2)^{th}$ amino acid is assigned to the $(N/2)^{th}$ grid point in all D dimensions. The $(N/2+1)^{th}$ amino acid is fixed to the $(N/2+1)^{th}$ grid point in the x direction and to the $(N/2)^{th}$ grid point in all other $D-1$ dimensions. As shown in Fig. 2, the final configuration we will try to optimize for the case of four amino acids takes the form $|q\rangle = |q_{16}q_{15}q_{14}q_{13}\rangle\,|0110\rangle\,|0101\rangle\,|q_4q_3q_2q_1\rangle$.

B. Adiabatic Quantum Computation

The goal of an adiabatic quantum algorithm is to transform an initial state into a final state which encodes the answer to the problem. A quantum state $|\psi(t)\rangle$ in the 2^n-dimensional Hilbert space for n qubits, evolves in time according to the Schrödinger equation

$$i\hbar\frac{d}{dt}\,|\psi(t)\rangle = \hat{H}(t)\,|\psi(t)\rangle\,, \tag{5}$$

where $\hat{H}(t)$ is the time-dependent Hamiltonian operator. The design of the algorithm takes advantage of the quantum adiabatic theorem [32], which is satisfied whenever $\hat{H}(t)$ varies

slowly throughout the time of propagation $t \in [0, \tau]$. Let $|\psi_g(t)\rangle$ be the instantaneous ground state of $\hat{H}(t)$. If we construct $\hat{H}(t)$ such that the ground state of $\hat{H}(0)$, denoted as $|\psi_g(0)\rangle$, is easy to prepare, the adiabatic theorem states that the time propagation of the quantum state will remain very close to $|\psi_g(t)\rangle$ for all $t \in [0, \tau]$. One way to choose $\hat{H}(0)$ is to construct it in such a way that $|\psi_g(0)\rangle$ is a uniform superposition of all possible configurations of the system, i.e.

$$|\psi_g(0)\rangle = \frac{1}{\sqrt{2^n}} \sum_{q_i \in \{0,1\}} |q_n\rangle |q_{n-1}\rangle \cdots |q_2\rangle |q_1\rangle \tag{6}$$

summing over all 2^n vectors of the computational basis. Notice that an initial Hamiltonian of the form

$$\hat{H}(0) = \sum_{i=1}^{n} \hat{q}_x^i = \sum_{i=1}^{n} \frac{1}{2}(I - \hat{\sigma}_i^x) \tag{7}$$

would have as a non-degenerate ground state the vector $|\psi_g(0)\rangle$ defined in Eq. 6.

Similarly to the operator $\hat{q}$ from Eq. 3, we define

$$\hat{q}_x^i \equiv \frac{1}{2}(I - \hat{\sigma}_i^x), \tag{8}$$

with $I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ and $\sigma^x = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$, the identity operator and the σ^x-Pauli matrix represented in the computational basis, respectively.

For example, for the case of four qubits, $n = 4$, $\hat{H}(0)$ is given by,

$$\hat{H}(0) = \sum_{i=1}^{4} \hat{q}_x^i = \hat{q}_x^1 + \hat{q}_x^2 + \hat{q}_x^3 + \hat{q}_x^4 \tag{9}$$

$$= I \otimes I \otimes I \otimes \hat{q}_x + I \otimes I \otimes \hat{q}_x \otimes I + I \otimes \hat{q}_x \otimes I \otimes I + \hat{q}_x \otimes I \otimes I \otimes I. \tag{10}$$

To find the lowest energy conformation of the protein, one defines a Hamiltonian, $\hat{H}_{protein}$, whose ground state encodes the solution. Adiabatic evolution begins with $\hat{H}(0)$ and $|\psi_g(0)\rangle$, and ends in $\hat{H}_{protein} = \hat{H}(\tau)$. If the adiabatic evolution is slow enough, the state obtained at time $t = \tau$ is $|\psi_g(\tau)\rangle$, the ground state of $\hat{H}(\tau) = \hat{H}_{protein}$. The details about the construction of $\hat{H}_{protein}$ will be provided in Sec. III. A possible adiabatic evolution path can be constructed by the linear sweep of a parameter $t \in [0, \tau]$,

$$\hat{H}(t) = (1 - t/\tau)\hat{H}(0) + (t/\tau)\hat{H}_{protein}. \tag{11}$$

Even though Eq. 11 connects $\hat{H}(0)$ and $\hat{H}_{protein}$, determining the optimum value of τ is an important and non-trivial problem in itself. In principle, the adiabatic theorem states that

over sufficient adiabatic time τ, the state $|\psi(\tau)\rangle$ will converge to the solution to the problem $|\psi_g(\tau)\rangle$. The magnitude of τ dictates the ultimate usefulness of the quantum algorithm proposed in this work. Farhi *et al.* [33, 34] showed promising numerical results for random instances of the Exact Cover computational problem.

Notice that the parameter τ determines the rate at which $\hat{H}(t)$ varies. Following the notation from Farhi *et al* [33], consider $\hat{H}(t) = \tilde{H}(t/\tau) = \tilde{H}(s)$, with instantaneous values of $\tilde{H}(s)$ defined by

$$\tilde{H}(s)\,|l;s\rangle = E_l(s)\,|l;s\rangle \tag{12}$$

with

$$E_0(s) \leq E_1(s) \leq \cdots \leq E_{N-1}(s) \tag{13}$$

where N is the dimension of the Hilbert space. According to the adiabatic theorem, if the gap between the two lowest levels, $E_1(s) - E_0(s)$, is greater than zero for all $0 \leq s \leq 1$, and taking

$$\tau \gg \frac{\varepsilon}{g_{min}^2} \tag{14}$$

with the minimum gap, g_{min}^2, defined by

$$g_{min} = \min_{0\leq s\leq 1}(E_1(s) - E_0(s)), \tag{15}$$

and ε given by

$$\varepsilon = \max_{0\leq s\leq 1} |\langle l=1;s|\frac{d\tilde{H}}{ds}|l=0;s\rangle\ |, \tag{16}$$

then we can make

$$|\langle l=0;s=1|\psi(\tau)\rangle| \tag{17}$$

arbitrarily close to 1. In other words, the existence of a nonzero gap guarantees that $|\psi(t)\rangle$ remains very close to the ground state of $\hat{H}(t)$ for all $0 \leq t \leq \tau$, if τ is sufficiently large.

In the following sections, we derive the expression for an energy function which is mapped to $\hat{H}_{protein}$ using the procedure explained in Sec II A. The final expression for $\hat{H}_{protein}$ corresponds to an array of coupled qubits. We use H to denote both the Hamiltonians and the energy functions given that the mapping is straightforward as explained at the end of Sec. II A.

III. CONSTRUCTION OF THE LATTICE PROTEIN HAMILTONIAN FOR ADIABATIC QUANTUM COMPUTATION

Our goal in this section is to find an algebraic expression for an energy function in which the ground state represents the lowest energy conformation of a protein. Ideally, this energy function should contain the least possible number of terms. In order to optimize the computational resources, we desire terms with low locality, defined as the number of products of q_i's that appear in a certain term (e.g., a term of the form $h = q_1 q_3 q_4 q_6$ is 4-local).

A. Small computer science digression

Encoding positions of the amino acids in the grid in terms of Boolean variables makes it very convenient to use tools from computer science and basic Boolean algebra [35]. In this section, we will review these tools before using them to contruct arbitrary Hamiltonians that encode the spectrum of statistical mechanical models. We begin with some simple relations that are useful in the derivation of the Hamiltonian terms.

Consider two Boolean variables x and y. Expressions for the operations AND, OR, NOT can be written algebraically as:

$$f_{\text{AND}}(x,y) = xy \qquad \text{AND operation } (x \wedge y)$$
$$f_{\text{OR}}(x,y) = x + y - xy \qquad \text{OR operation } (x \vee y)$$
$$f_{\text{NOT}}(x) = 1 - x \qquad \text{NOT operation } (\neg x)$$

An additional useful Boolean operator for the construction of Hamiltonian terms is XNOR. The output of the XNOR function is 0 unless all its arguments have the same value. The two-input version XNOR operation is also known as *logical equality*, here denoted as EQ,

$$f_{\text{EQ}}(x,y) = 1 - x - y + 2xy \qquad \text{XNOR operation}(x \text{ EQ } y)$$

The XNOR operator can be used to construct a very useful term for statistical mechanics Hamiltonians, an on-site repulsion penalty (described in Sec. III B and illustrated in Fig. 3).

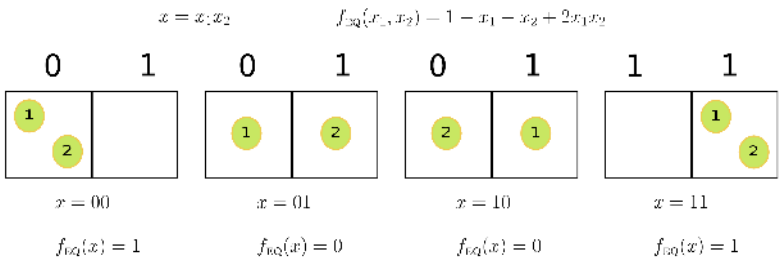

FIG. 3: (Color online) Illustrative example of one of the uses of the XNOR Boolean function in our scheme for the construction of Hamiltonians. Consider two particles 1 and 2 that are restricted to occupy either position 0 or 1 in the dimension shown, and let x_1 and x_2 encode the position particle 1 and particle 2 respectively. The Boolean function f_{EQ} can be interpreted as an onsite repulsion Hamiltonian which penalizes configurations where $x_1 = x_2$. The possible configurations are encoded in the bit string $x = x_1x_2$.

B. Hamiltonian terms for protein folding: the HP model

Most of the configurations represented by the bit strings q of Eq. 1 are invalid protein states. We seek a Hamiltonian that energetically favors valid configurations of the HP model by eliminating configurations in which more than one amino acid occupy the same grid point, and discarding configurations that violate the primary sequence of amino acids. This Hamiltonian can be written as

$$H_{protein} = H_{onsite} + H_{psc} + H_{pairwise}, \tag{18}$$

where H_{onsite} is an onsite repulsion term for amino acids occupying the same grid point, H_{psc} is a primary sequence constraint term, and $H_{pairwise}$ is a pairwise interaction term that represents favorable hydrophobic interactions between adjacent hydrophobic amino acids.

Each protein configuration can be described by a string of $ND\log_2 N$ bits, where D is the number of dimensions and N is the number of amino acids. Without loss of generality, N is here contrained to be a power of two. Below, we describe each term in Eq. 18.

1. *Onsite term, H_{onsite}*

The first term in Eq. 18, H_{onsite}, prevents two or more amino acids from occupying the same grid point. For a given protein, at least one position variable must differ between each pair of amino acids for H_{onsite} to evaluate to zero. As an illustrative example, a simple one-dimensional two-site Hamiltonian is shown in Fig. 3 using the XNOR operation described in Sec. III A.

The general term for D dimensions and N amino acids is

$$H_{onsite}(N,D) = \lambda_0 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} H^{ij}_{onsite}(N,D) \tag{19}$$

with

$$H^{ij}_{onsite}(N,D) = \prod_{k=1}^{D} \prod_{r=1}^{\log_2 N} \Big(1 - q_{f(i,k)+r} - q_{f(j,k)+r} + 2\, q_{f(i,k)+r}\, q_{f(j,k)+r}\Big) \tag{20}$$

and

$$f(i,k) = D(i-1)\log_2 N + (k-1)\log_2 N. \tag{21}$$

The terms enclosed by the parentheses in Eq. 20 are XNOR functions. The double product of these terms tests that all of these conditions are considered simultaneously by using AND relations. If all the binary variables describing the coordinates of the i-th and j-th amino acids are equal, then the series of products of XNOR functions is evaluated to $+1$. In this case, the energy penalty λ_0 with $\lambda_0 > 0$ is enforced. There will be no energy penalty, however, if even one of the binary variables for the i-th and j-th amino acids is different.

The function $f(i,k)$ is a pointer to the bit substring describing the coordinates of a particular amino acid. The index i points to the i-th amino acid and the index k points to the first bit variable of the k-th spatial coordinate. Here, $k = 1$ corresponds to the x coordinate, $k = 2$ to the y coordinate, and $k = 3$ to the z coordinate. For example, consider the case with $N = 4$ and $D = 2$. If we are interested in referring to the first binary variable describing the y coordinate ($k = 2$), for the third amino acid ($i = 3$), a direct substitution in Eq. 21 would yield $f(3,2) = 10$, which is indeed the variable we are interested in according to the convention established in Eq. 1.

2. Primary structure constraint, H_{psc}

The term H_{psc} in Eq. 18 evaluates to zero when two amino acids P and Q that are consecutive sequence-wise must be nearest neighbors on the lattice. Nearest-neighbors are defined as those points with a rectilinear (L_1) distance of $d_{PQ} = 1$ between them. We define a distance function that gives the base 10 distance squared between any two amino acids P and Q on the lattice,

$$d^2_{PQ}(N, D) = \sum_{k=1}^{D} \Big(\sum_{r=1}^{\log_2 N} 2^{r-1} (q_{f(P,k)+r} - q_{f(Q,k)+r}) \Big)^2 \tag{22}$$

with $f(i, k)$ defined as in Eq. 21.

A simple way of defining H_{psc} is

$$H'_{psc}(N, D) = \lambda_1 \sum_{m=1}^{N-1} (1 - d^2_{m,m+1})^2 \tag{23}$$

Or, preferably,

$$H_{psc}(N, D) = \lambda_1 \Big[-(N-1) + \sum_{m=1}^{N-1} d^2_{m,m+1} \Big]. \tag{24}$$

Unlike Eq. 23, the improved Hamiltonian in Eq. 24 is always 2-local regardless of the number of amino acids or the dimensionality of the problem, since $d^2_{PQ}(N, D)$ is always 2-local.

First, notice that for valid configurations, all $(N-1)$ terms in the sum will equal one, and $H_{psc}(N, D)$ evaluates to zero. If any of the $d^2_{m,m+1}$ terms is zero, meaning that two amino acids occupy the same location, then H_{onsite} will be drastically raised by the energy penalty λ_0. This can be achieved by setting $\lambda_0 > \lambda_1$, and $\lambda_1 = N$. After excluding configurations in which any $d^2_{m,m+1}$ are zero, only configurations with values of $d^2_{m,m+1} > 1$ are left. In these instances, $H_{psc}(N, D) > 0$ and λ_1 will play the role of an energy penalty since $\lambda_1 > 0$. Choosing $\lambda_1 = N$ and $\lambda_0 = N + 1 > \lambda_1$ constrains unwanted or penalized configurations to eigenstates of $H_{protein}$ with energies greater than zero, while plausible configurations of the protein correspond to energies less than or equal to zero. Note that the minimum energy of the HP model, in the case of all hydrophobic sequences with the maximum number of favorable contacts, is always greater than $-N$. This is satisfied in general for N amino acids in either two or three dimensions.

3. Pairwise hydrophobic interaction term, $H_{pairwise}$

The HP model favors hydrophobic interactions by lowering the energy by 1 whenever non-nearest neighboring hydrophobic amino acids are a rectilinear distance of 1 away.

This kind of interaction is represented by the following general expression:

$$H_{pairwise}(N, D) = -\sum_{i=1}^{N}\sum_{j=1}^{N} G_{ij} H^{ij}_{pairwise} \tag{25}$$

Here G is an $N \times N$ symmetric matrix with entries G_{ij} equal to $+1$ when amino acids i and j are hydrophobic and non-nearest neighbors, and 0 otherwise. Note that G_{ij} is set to zero for amino acids that are neighbors in the protein sequence. Notice also that alternate definitions of G_{ij} could allow us to define lattice protein models that are more complex than the HP model. One of these models is the more realistic Miyazawa-Jernigan model [36] which includes interactions between 20 types of amino acids.

The form of $H^{ij}_{pairwise}$ depends on the spatial dimensionality of the problem. In two dimensions, we have

$$\begin{aligned} H^{ij}_{pairwise} \;=\; & H^{ij,2D}_{pairwise}(N) = x^{ij,2D}_{+}(N) + x^{ij,2D}_{-}(N) \\ & + y^{ij,2D}_{+}(N) + y^{ij,2D}_{-}(N) \end{aligned} \tag{26}$$

and in three dimensions,

$$\begin{aligned} & H^{ij}_{pairwise} = H^{ij,3D}_{pairwise}(N) = x^{ij,3D}_{+}(N) + x^{ij,3D}_{-}(N) \\ & + y^{ij,3D}_{+}(N) + y^{ij,3D}_{-}(N) + z^{ij,3D}_{+}(N) + z^{ij,3D}_{-}(N) \end{aligned} \tag{27}$$

The terms on the right hand side of Eq. 27 are independent; each one serves to query whether the j-th amino acid is located, with respect with the i-th amino acid, to the right, left, above, below, in front, or behind as represented by $x^{ij,3D}_{+}$, $x^{ij,3D}_{-}$, $y^{ij,3D}_{+}$, $y^{ij,3D}_{-}$, $z^{ij,3D}_{+}$, and $z^{ij,3D}_{-}$ terms, respectively. If the j-th amino acid is located at a distance of exactly one in any direction, $H^{ij}_{pairwise}$ is set to $+1$; otherwise it is set to zero. There is a subtle but important condition embedded in these terms: they all vanish if the rightmost binary variable describing the i-th residue's coordinate of interest (say x for $x^{ij,3D}_{+}$ and $x^{ij,3D}_{-}$ or y for $y^{ij,3D}_{+}$ and $y^{ij,3D}_{-}$ or z for $z^{ij,3D}_{+}$ and $z^{ij,3D}_{-}$) does not end in 0, i.e., the coordinate has to correspond to an even number. This is why we intentionally double count each pair of amino acids in Eq. 25 by allowing both indexes i and j iterate from 1 to N. No special treatment

is provided for the case where $i = j$, since the diagonal terms of G_{ij} are all zero due to the lack of amino acid self interaction. Finally, because we want the interaction to be attractive rather than repulsive, we use the minus sign in Eq. 25.

The case of N amino acids in a two dimensional grid for $N = 2^M$ and $M \geq 3$: The terms listed below correspond to the pairwise interaction Hamiltonian terms described above. The expressions below were constructed for $M \geq 3$. The four amino acid case ($M = 2$) is much simpler and will be discussed in Sec. IV. The expression for $x_+^{ij,2D}(N)$ is

$$\begin{aligned} x_+^{ij,2D}(N) = (1 - q_{f(i,1)+1})q_{f(j,1)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,1)+s} \\ -q_{f(i,1)+s} + 2\, q_{f(j,1)+s}\, q_{f(i,1)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,2)+r} \\ -q_{f(j,2)+r} + 2q_{f(i,2)+r}q_{f(j,2)+r}) \end{aligned} \tag{28}$$

The first two factors of $x_+^{ij,2D}(N)$ (Eq. 28) treat the rightmost binary digit of the x position of the i-th and j-th amino acid. The first factor guarantees that the i-th residue is in an even position on the x-axis. For an interaction to be considered, the position of the j-th residue on the x-axis must be odd, as required by the second factor $q_{f(j,1)+1}$. The remaining factors of $x_+^{ij,2D}$ are XNOR functions that ensure that the rest of the binary digits that encode the x position are equal for the i-th and j-th amino acids. Finally, all the digits encoding the y position have to be equal, so that the i-th and j-th amino acids are nearest neighbors displaced only in the x-directionforcing the two residues to be in the same row. If all these conditions are satisfied, $x_+^{ij,2D}$ evaluates to $+1$; otherwise it evaluates to 0. These conditions rely on the fact that adding 1 to an even number only changes the rightmost binary digit from 0 to 1.

The construction of $y_+^{ij,2D}$ follows the same procedure as that of $x_+^{ij,2D}$, namely,

$$\begin{aligned} y_+^{ij,2D}(N) = (1 - q_{f(i,2)+1})q_{f(j,2)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,2)+s} \\ -q_{f(i,2)+s} + 2\, q_{f(j,2)+s}\, q_{f(i,2)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,1)+r} \\ -q_{f(j,1)+r} + 2q_{f(i,1)+r}q_{f(j,1)+r}) \end{aligned} \tag{29}$$

The construction of $x_{-}^{ij,2D}$,

$$
\begin{aligned}
x_{-}^{ij,2D}(N) = (1 - q_{f(i,1)+1}) q_{f(j,1)+1} \Big[1 - \prod_{k=1}^{\log_2 N} (1 - \\
q_{f(i,1)+k}) \Big] (q_{f(j,1)+2} + q_{f(i,1)+2} - 2\, q_{f(j,1)+2}\, q_{f(i,1)+2}) \\
\prod_{r=3}^{\log_2 N} \Big[1 - (q_{f(j,1)+r} + \prod_{u=2}^{r-1} q_{f(j,1)+u} - 2 \prod_{u=2}^{r} q_{f(j,1)+u}) \\
- q_{f(i,1)+r} + 2 q_{f(i,1)+r} (q_{f(j,1)+r} + \\
\prod_{u=2}^{r-1} q_{f(j,1)+u} - 2 \prod_{u=2}^{r} q_{f(j,1)+u}) \Big] \\
\prod_{s=1}^{\log_2 N} (1 - q_{f(i,2)+s} - q_{f(j,2)+s} + 2 q_{f(i,2)+s} q_{f(j,2)+s})
\end{aligned}
\tag{30}
$$

involves several considerations. As in the expression for $x_{+}^{ij,2D}$, the first factor $(1 - q_{f(i,1)+1})$ tests if the i-th amino acid is in an even position along the x-axis. Here, we are interested in querying whether the j-th amino acid is directly to the left of the i-th, and apply a different procedure than that of Eq. 28. We add $00\cdots01$ to the x coordinate of the j-th residue, thus moving "right" by one unit, and use the XNOR function to check if the result matches the x coordinate of the i-th amino acid. The problem is not as trivial as the case of $x_{+}^{ij,2D}$. Setting i at an even coordinate value along the axis of interest forces j to be in an odd coordinate. However, adding $00\cdots01$ to an odd binary number in general will change more digits than just the last digit due to carry bits. We used the circuit presented in Fig. 4 and the Boolean algebra introduced in Sec. III A to obtain the general expression for the addition of $00\cdots01$ to an n-bit number. If we take $x = x_n x_{n-1} \cdots x_2 x_1$ and $y = 00\cdots01$, then the result $z = z_{n+1} z_n z_{n-1} \cdots z_2 z_1$ for the addition $z = x + y$ is the recursive algebraic expression,

$$
\begin{aligned}
z_1 &= 0 \\
z_2 &= 1 - x_2 \\
z_k &= x_k + \prod_{u=2}^{k-1} x_u - 2 \prod_{u=2}^{k} x_u \quad \text{for} \quad 3 \leq k \leq n \\
z_{n+1} &= \prod_{u=2}^{n} x_u
\end{aligned}
$$

As in the case of $x_{+}^{ij,2D}$, we impose conditions that guarantee that the y coordinate is the

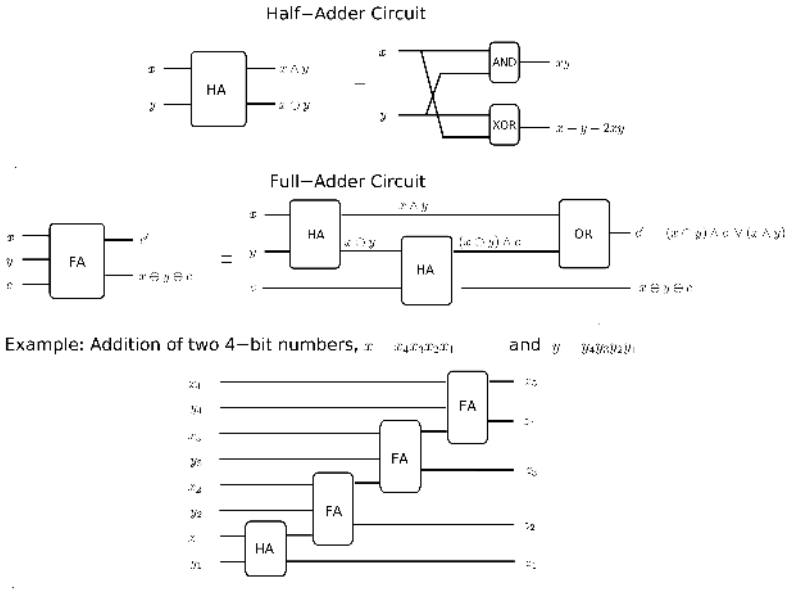

FIG. 4: Half-adder and full-adder components for the addition circuit implemented in the pairwise interaction Hamiltonian. We show the implementation of these two components for the addition of two 4-bit numbers yielding $z = z_5z_4z_3z_2z_1$. The addition of n-bit numbers can be generalized trivially.

same for both amino acids (that they are in the same row).

A special case arises when the j-th amino acid is at the rightmost position in the grid, with an x coordinate value of $11\cdots11$. When $00\cdots01$ is added to this coordinate, z_{n+1} evaluates to 1 and the n bits z_1 to z_n evaluate to 0. Since only the first n bits are used to compare coordinates, this z would be an undesirable match with an i-th amino acid positioned at $x = 00\cdots00$. Notice that a value of $x = 00\cdots00$ positions the i-th amino acid positioned at the minimal/leftmost position in the grid, for which $x_-^{ij,2D}$ should not even be considered. The factor $[1 - \prod_{k=1}^{\log_2 N}(1 - q_{f(i,1)+k})]$ in Eq. 30 sets the term $x_-^{ij,2D}$ to 0 if the x coordinate of the i-th amino acid is $00\cdots00$, taking care of both of these concerns.

The construction of $y_{-}^{ij,2D}$ follows the same procedure as that of $x_{-}^{ij,2D}$, namely,

$$
\begin{aligned}
y_{-}^{ij,2D}(N) = (1 - q_{f(i,2)+1}) q_{f(j,2)+1} \Big[1 - \prod_{k=1}^{\log_2 N} (1 - \\
q_{f(i,2)+k}) \Big] (q_{f(j,2)+2} + q_{f(i,2)+2} - 2\, q_{f(j,2)+2}\, q_{f(i,2)+2}) \\
\prod_{r=3}^{\log_2 N} \Big[1 - (q_{f(j,2)+r} + \prod_{u=2}^{r-1} q_{f(j,2)+u} - 2 \prod_{u=2}^{r} q_{f(j,2)+u}) \\
- q_{f(i,2)+r} + 2 q_{f(i,2)+r} (q_{f(j,2)+r} + \\
\prod_{u=2}^{r-1} q_{f(j,2)+u} - 2 \prod_{u=2}^{r} q_{f(j,2)+u}) \Big] \\
\prod_{s=1}^{\log_2 N} (1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2 q_{f(i,1)+s} q_{f(j,1)+s})
\end{aligned}
\tag{31}
$$

The three-dimensional extension of these equations is presented in the Appendix.

C. Maximum locality and scaling of the number of terms in $H_{protein}$

In this section, we estimate the number of terms included in the total Hamiltonian $H_{protein}$ and present procedures required to reduce the locality of the terms to 2-local. These estimates assess the size of a quantum device necessary for eventual experimental realizations of the algorithm. The reduction of the locality of the terms involves ancillary qubits.

Each amino acid requires $D \log_2 N$ qubits to specify its position in the lattice. Since our algorithm fixes the position of two amino acids, the number of qubits needed to encode the coordinates of the $(N-2)$ remaining amino acids is $(N-2)D \log_2 N$. From the expressions given for H_{onsite}, H_{psc} and $H_{pairwise}$, one can deduce that the maximum locality is determined by $2D \log_2 N$ — the number of qubits corresponding to two amino acids. As described in Sec. III B 2, the H_{psc} term is always 2-local in nature regardless of the number of amino acids. For scaling arguments, it is crucial to point out that all possible 1-local and 2-local terms, that account for $(N-2)D \log_2 N$ and $\binom{(N-2)D \log_2 N}{2}$ total terms, repectively, appear in the expansion, but that not all possible 3-local or higher locality terms will be present. For example, the terms $q_i q_j q_k$, where the indexes i, j and k are associated with three different amino acids, are not part of the expansion, since every term should only involve products of qubits describing two amino acids, regardless of its locality. Table I summarizes the number of k-local terms required to construct the protein Hamiltonian,

$H_{protein} = H_{onsite} + H_{psc} + H_{pairwise}$. The alternative count from the combinatorial expressions of Table I scales as N^6 for $D = 2$ and as N^8 for $D = 3$. Table I provides the exact term count.

TABLE I: The number of k-local terms obtained in the final expression for $H_{protein}$ as a function of the number of amino acids N, $N = 2^M$, and dimensions (D) of the lattice.

locality	Number of terms, T_k
$k = 0$	1
$k = 1$	$(N-2)D\log_2 N$
$2 \leq k \leq D\log_2 N$	$\binom{N-2}{2}\sum_{i=1}^{k-1}\binom{D\log_2 N}{i}\binom{D\log_2 N}{k-i} + (N-2)\binom{D\log_2 N}{k}$
$D\log_2 N < k \leq 2D\log_2 N$	$\binom{N-2}{2}\sum_{i=k-D\log_2 N}^{D\log_2 N}\binom{D\log_2 N}{i}\binom{D\log_2 N}{k-i}$
Total number of terms	$\sum_{k=0}^{2Dlog_2 N} T_k \sim N^{2D+2}$

IV. CASE STUDY: HPPH

With the goal of designing an experiment for adiabatic quantum computers with small numbers of qubits, we concentrate on the simplest possible instance of the HP-model – a four amino acid loop that contains a favorable interaction and therefore "folds".

In Sec. IV A we present the protein Hamiltonian, followed by the partitioning of the N-local Hamiltonian terms to 2-local. Finally, we present numerical simulations which confirm the local minimum through the use of the proposed algorithm.

A. Hamiltonian terms for the case of four amino acids in 2D

The onsite Hamiltonian for this example takes the form

1. *Onsite term, H_{onsite}*

$$H_{onsite}(N = 4, D = 2) = \lambda_0(H^{12}_{onsite} + H^{13}_{onsite} + H^{14}_{onsite} + H^{24}_{onsite} + H^{34}_{onsite}) \quad (32)$$

with

$$H^{ij}_{onsite}(N=4,D=2) = \prod_{k=1}^{2}\prod_{r=1}^{2}\Big(1 - q_{f(i,k)+r} - q_{f(j,k)+r} + 2\,q_{f(i,k)+r}\,q_{f(j,k)+r}\Big) \tag{33}$$

and

$$f(i,k) = 4(i-1) + 2(k-1). \tag{34}$$

Note that H^{23}_{onsite} does not appear in Eq. 32 since, as described in Sec. II A, the two central amino acids are fixed in position and guaranteed not to occupy overlapping gridpoints that would contribute an energy penalty to the onsite term *a priori* . On the other hand, other terms involving amino acids 2 and 3 cannot be discarded, since these amino acids will affect their other neighbors through H_{psc} and they can participate in hydrophobic interactions through $H_{pairwise}$.

2. Primary structure constraint term, H_{psc}

The pairwise term

$$d^2_{PQ}(N=4,D=2) = \sum_{k=1}^{2}\Big(\sum_{r=1}^{2} 2^{r-1}(q_{f(P,k)+r} - q_{f(Q,k)+r})\Big)^2 \tag{35}$$

with

$$\begin{aligned} H_{psc}(N=4,D=2) &= \lambda_1\big(-3 + d^2_{12} + d^2_{23} + d^2_{34}\big) \\ &= \lambda_1\big(-2 + d^2_{12} + d^2_{34}\big) \end{aligned} \tag{36}$$

takes advantage of the fact that $d^2_{23} = 1$ by construction.

3. Pairwise term, $H_{pairwise}$

Finally, a pairwise interaction term is required to impose an energy stabilization for non-nearest neighbor hydrophobic amino acids that occupy adjacent sites in the lattice.

For the sequence HPPH,

$$G = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \tag{37}$$

and therefore,

$$H^{2D}_{pairwise}(N=4,D=2) = -(H^{14,2D}_{pairwise} + H^{41,2D}_{pairwise}). \tag{38}$$

For this particular case of interest

$$H^{ij,2D}_{pairwise}(N=4) = x^{ij,2D}_{+}(N=4) + x^{ij,2D}_{-}(N=4) + y^{ij,2D}_{+}(N=4) + y^{ij,2D}_{-}(N=4). \tag{39}$$

The explicit forms of these functions are:

$$x^{ij,2D}_{+}(N=4) = (1-q_{f(i,1)+1})q_{f(j,1)+1}(1-q_{f(j,1)+2} - q_{f(i,1)+2} + 2\,q_{f(j,1)+2}q_{f(i,1)+2}) \prod_{s=1}^{2}(1-q_{f(i,2)+s} - q_{f(j,2)+s} + 2q_{f(i,2)+s}q_{f(j,2)+s}), \tag{40}$$

$$y^{ij,2D}_{+}(N=4) = (1-q_{f(i,2)+1})q_{f(j,2)+1}(1-q_{f(j,2)+2} - q_{f(i,2)+2} + 2\,q_{f(j,2)+2}\,q_{f(i,2)+s}) \prod_{s=1}^{2}(1-q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s}), \tag{41}$$

$$x^{ij,2D}_{-}(N=4) = (1-q_{f(i,1)+1})q_{f(j,1)+1}q_{f(i,1)+2} (q_{f(j,1)+2} + q_{f(i,1)+2} - 2q_{f(j,1)+2}q_{f(i,1)+2}) \prod_{s=1}^{2}(1 - q_{f(i,2)+s} - q_{f(j,2)+s} + 2\,q_{f(i,2)+s}q_{f(j,2)+s}), \tag{42}$$

$$y^{ij,2D}_{-}(N=4) = (1-q_{f(i,2)+1})q_{f(j,2)+1}q_{f(i,2)+2} (q_{f(j,2)+2} + q_{f(i,2)+2} - 2\,q_{f(j,2)+2}q_{f(i,2)+2}) \prod_{s=1}^{2}(1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2\,q_{f(i,1)+s}q_{f(j,1)+s}). \tag{43}$$

After expanding all of the terms in H_{onsite}, H_{psc} and $H_{pairwise}$, we fix amino acids 2 and 3 as described in Sec. II A, substituting the variables $q_{12}q_{11}q_{10}q_9\ q_8q_7q_6q_5$ by the constant values 0110 0101 as shown in Fig. 2. The final expression for $H_{protein}$ now depends on the 8 binary variables encoding the coordinates of amino acids 1 and 4, $q_4q_3q_2q_1$ and $q_{16}q_{15}q_{14}q_{13}$, respectively. For convenience in notation, we relabel the coordinates of amino acid 4 from

$q_{16}q_{15}q_{14}q_{13}$ to $q_8q_7q_6q_5$. After these substitutions, the final expression for the energy function $H_{protein}$ will be dependent on products involving the variables q_1 through q_8. Following the mapping explained at the end of Sec. II A, the quantum expression for $\hat{H}_{protein}$ is a $2^8 \times 2^8$ matrix. This Hamiltonian matrix defines the final Hamiltonian $\hat{H}(t=\tau)$ of the adiabatic evolution. The initial Hamiltonian representing the transverse field whose ground state is a linear superposition of all 2^8 states in the computational basis can be written as

$$\hat{H}_0 \equiv \hat{H}(t=0) = \sum_{i=1}^{8} \hat{q}_x^i = \sum_{i=1}^{8} \frac{1}{2}(I - \hat{\sigma}_i^x) \tag{44}$$

with

$$|\psi_g(t=0)\rangle = \frac{1}{\sqrt{2^8}} \sum_{q_i \in \{0,1\}} |q_8q_7q_6q_5q_4q_3q_2q_1\rangle \tag{45}$$

Finally, we can construct a time dependent Hamiltonian as shown in Eq. 11,

$$\hat{H}(t) = (1 - t/\tau)\hat{H}_0 + (t/\tau)\hat{H}_{protein} \tag{46}$$

This time dependent Hamiltonian is also a $2^8 \times 2^8$ matrix as well. The instantaneous spectrum can be obtained by diagonalizing at every t/τ without need to specify τ. Since τ is the running time, we are interested in $0 \leq t/\tau \leq 1$. The spectrum of the corresponding $\hat{H}(t)$ for this four amino acid peptide HPPH is given in Fig. 5.

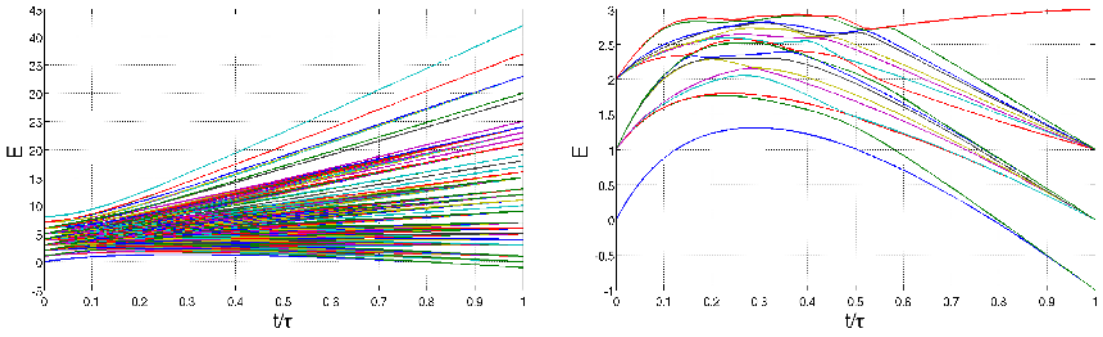

FIG. 5: (Color online) Spectrum of the instantaneous energy eigenvalues for the 8-local time dependent Hamiltonian used in the algorithm for the peptide HPPH (left). The plot to the right examines the lowest 15 states of the 256 states from the left.

Snapshots of the instantaneous ground state are shown in Fig. 6. Even though these snapshots do not correspond to explicit propagation of the Schrödinger equation, they indicate that the final $H_{protein}$ is correct and that it provides the correct answer if a sufficiently

long time τ is allowed. Notice that at $t/\tau = 0$, the amplitude for all 256 states is equal, indicating a uniform superposition of all states; at $t/\tau = 1$, the readout corresponds to the two degenerate solutions of HPPH.

V. CONVERTING AN N-LOCAL HAMILTONIAN TO A 2-LOCAL HAMILTONIAN

Motivated by the possibility of an experimental implementation, we explain how to reduce the locality of a Hamiltonian from k-local to 2-local while conserving its low-lying spectrum. We use Boolean reduction techniques [37, 38] for Hamiltonians contructed from energy functions with structure similar to $H_{protein}$, where all of terms are sums of tensor products of σ_z^i operators. By reducing the locality of the interactions, we introduce new ancilla qubits to represent higher order interactions with sums of at most 2-local terms. Here, we present an illutrative example with a relative simple energy function but the methodology can be immediately extended to higher locality energy functions such as the one resulting in $H_{protein}$.

Consider a 4-local energy function of the form

$$H_{toy}(q) = 1 + q_1 - q_2 + q_3 + q_4 - q_1q_2q_3 + q_1q_2q_3q_4. \tag{47}$$

As shown in Table II, this energy function has a unique minimum energy given by $q = q_4q_3q_2q_1 = 0010$. The energy associated with this configuration is 0 in arbitrary units and all other possible values of the binary variables q_1, q_2, q_3 and q_4 have energies ranging from 0 to 4.

The goal is to obtain an energy function H' that preserves these energies along with their associated bit strings, but defines H' using only 1-local and 2-local terms. That is, the goal is to obtain a substitution for H_{toy} with the following form,

$$H'(\tilde{q}_1, \cdots, \tilde{q}_M) = c_0 + \sum_{i=1}^{M} c_i\tilde{q}_i + \sum_{i=1}^{M-1}\sum_{j=i+1}^{M} d_{ij}\tilde{q}_i\tilde{q}_j. \tag{48}$$

In Eq. 48 the new set of binary variables $\tilde{q}$ includes the original variables q_i as well as ancillary variables required to reduce locality. The extra ancillary bits raise the total number of variables to M.

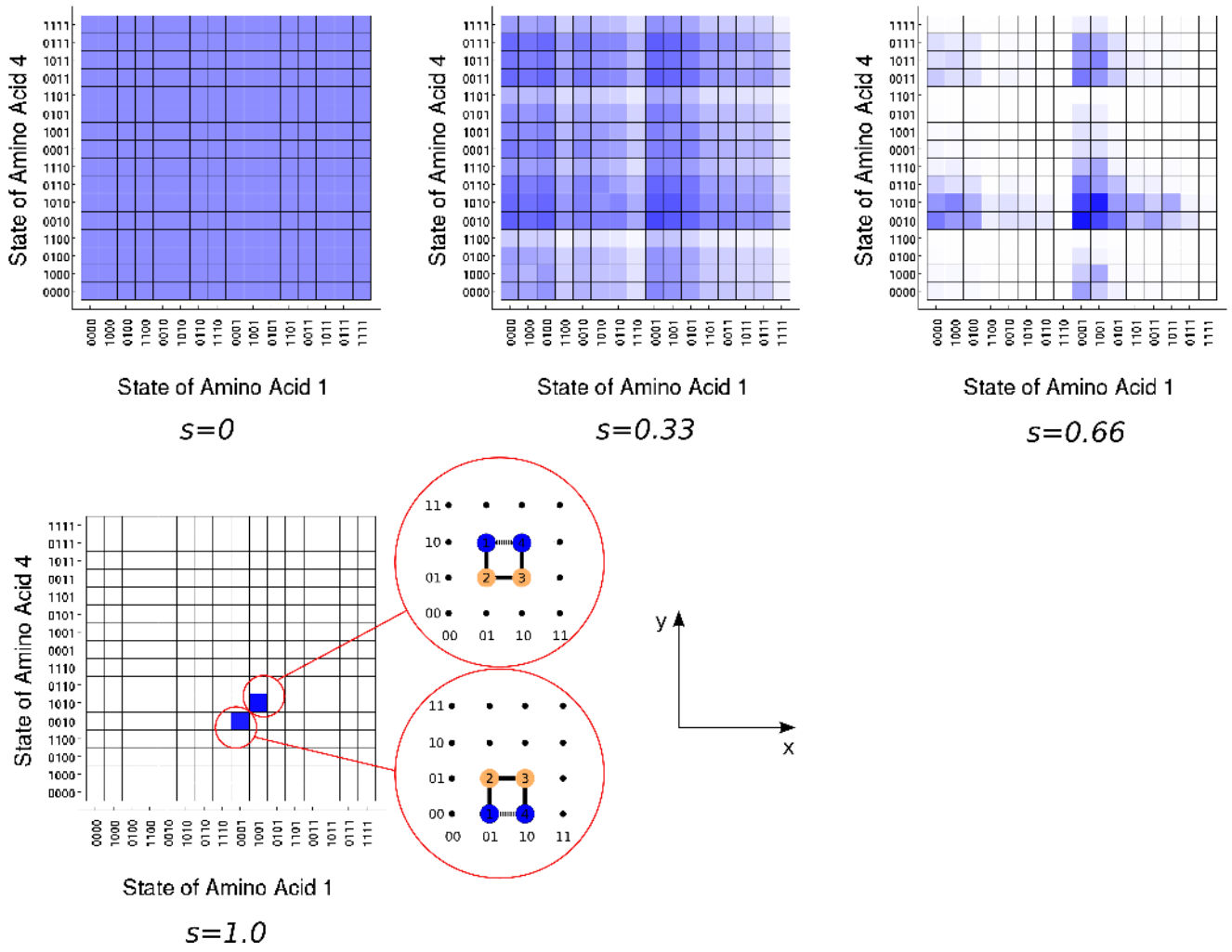

FIG. 6: (Color online) Snapshots of the instantaneous ground state for $H(t)$. The brightness of the box is proportional to $|c_n|^2$. Axis labels and state vectors for each particular box correspond to $|\psi\rangle = \sum_{n=0}^{255} c_n |n\rangle$ with $|n\rangle$ the n-th state vector out of the 256 possibilities given by $|q_{16}\rangle |q_{15}\rangle |q_{14}\rangle |q_{13}\rangle |q_4\rangle |q_3\rangle |q_2\rangle |q_1\rangle$. Notice that the x axis is given by $|q_4\rangle |q_3\rangle |q_2\rangle |q_1\rangle$ and the y axis given by $|q_{16}\rangle |q_{15}\rangle |q_{14}\rangle |q_{13}\rangle$. The final state corresponds to the two degenerate minima shown at the end

TABLE II: Truth table for the energy function $H_{toy}(q) = 1 + q_1 - q_2 + q_3 + q_4 - q_1q_2q_3 + q_1q_2q_3q_4$.

q_4	q_3	q_2	q_1	$H(q_1, q_2, q_3, q_4)$
0	0	1	0	0
0	0	0	0	1
0	0	1	1	1
0	1	1	0	1
0	1	1	1	1
1	0	1	0	1
0	0	0	1	2
0	1	0	0	2
1	0	0	0	2
1	0	1	1	2
1	1	1	0	2
0	1	0	1	3
1	0	0	1	3
1	1	0	0	3
1	1	1	1	3
1	1	0	1	4

Since the information contained within the problem and the solution we are seeking both rely on the original set of q variables (in the case of protein folding, for example, the string q encodes the positions of the amino acids in the lattice), we must be able to identify values corresponding to the original q, regardless of the substitutions made to convert a k-local function to a 2-local. The new energy function H' needs to have the energy values of the original function in its energy spectrum. In addition, the values of the bit string $\tilde{q}$ for these energies must match the same values of q in the original function. For the particular example of Eq. 47, consider the substitutions, $q_1q_2 \rightarrow \tilde{q}_5$ and $q_3q_4 \rightarrow \tilde{q}_6$. These two subtitutions introduce two new independent binary variables, $\tilde{q}_5$ and $\tilde{q}_6$ and regardless of the values of q_1, q_2, q_3 and q_4, they can take any value in $\{0, 1\}$. Since we want to preserve both the physical meaning of the original energy function, as well as its energy spectrum,

we need to perform an action on the cases where the conditions $\tilde{q}_5 = q_1 \wedge q_2$ and $\tilde{q}_6 = q_3 \wedge q_4$ are not satisfied and lack any meaning in the context of the original energy function. One way to address this problem while keeping the original spectrum intact is to add a penalty function which enforces the conditions $\tilde{q}_5 = q_1 \wedge q_2$ and $\tilde{q}_6 = q_3 \wedge q_4$. For every substitution of the form $q_i q_j \to \tilde{q}_n$, consider a function of the form [37]

$$H_\wedge(q_i, q_j, \tilde{q}_n) = \delta(3\tilde{q}_n + q_i q_j - 2q_i\tilde{q}_n - 2q_j\tilde{q}_n). \tag{49}$$

As shown in Table III, for $\delta > 0$, the function $H_\wedge(q_i, q_j, \tilde{q}_n)$ is greater than zero whenever $\tilde{q}_n \neq q_i \wedge q_j$ and it evaluates to zero whenever $\tilde{q}_n = q_i \wedge q_j$.

TABLE III: Truth table for the function $H_\wedge(q_i, q_j, \tilde{q}_n) = \delta(3\tilde{q}_n + q_i q_j - 2q_i\tilde{q}_n - 2q_j\tilde{q}_n)$ used for the locality reduction procedure described in Sec. V.

$\tilde{q}_n$	q_i	q_j	$H_\wedge(q_i, q_j, \tilde{q}_n)$
0	0	0	0
0	0	1	0
0	1	0	0
1	1	1	0
1	0	0	3δ
1	0	1	δ
1	1	0	δ
0	1	1	δ

A two-local expression of the form presented in Eq. 48 can be obtained by adding one $H_\wedge(q_i, q_j, \tilde{q}_n)$ function for each substitution $q_1 q_2 \to \tilde{q}_5$ and $q_3 q_4 \to \tilde{q}_6$ and by making the additional trivial substitutions $q_1 \to \tilde{q}_1$, $q_2 \to \tilde{q}_2$, $q_3 \to \tilde{q}_3$, and $q_4 \to \tilde{q}_4$, to conveniently change in notation to the set of binary variables $\tilde{q}$. For the case of the energy function of Eq. 47, the locality reduced version is

$$\begin{aligned} H_{toy,reduced}(\tilde{q}) &= 1 + \tilde{q}_1 - \tilde{q}_2 + \tilde{q}_3 + \tilde{q}_4 - \tilde{q}_5\tilde{q}_3 + \tilde{q}_5\tilde{q}_6 + H_\wedge(q_1, q_2, \tilde{q}_5) + H_\wedge(q_3, q_4, \tilde{q}_6) \\ &= 1 + \tilde{q}_1 - \tilde{q}_2 + \tilde{q}_3 + \tilde{q}_4 - \tilde{q}_5\tilde{q}_3 + \tilde{q}_5\tilde{q}_6 + \delta(3\tilde{q}_5 + \tilde{q}_1\tilde{q}_2 - 2\tilde{q}_1\tilde{q}_5 - 2\tilde{q}_2\tilde{q}_5) \\ &+ \delta(3\tilde{q}_6 + \tilde{q}_3\tilde{q}_4 - 2\tilde{q}_3\tilde{q}_6 - 2\tilde{q}_4\tilde{q}_6). \end{aligned} \tag{50}$$

Recall that the additional functions $H_\wedge(\tilde{q}_1, \tilde{q}_2, \tilde{q}_5)$ and $H_\wedge(\tilde{q}_3, \tilde{q}_4, \tilde{q}_6)$ increase the energy of $H_{toy,reduced}$ by at least δ whenever the conditions $\tilde{q}_5 = \tilde{q}_1 \wedge \tilde{q}_2$ and $\tilde{q}_6 = \tilde{q}_3 \wedge q_4$ are not satisfied. Table IV shows the one-to-one mapping between the energies of non-penalized configurations of $H_{toy,reduced}(\tilde{q})$ and configurations presented in Table II associated with $H_{toy}(q)$. Even though there is a unique configuration $\{\tilde{q}_6 = \tilde{q}_3 \wedge \tilde{q}_4, \tilde{q}_5 = \tilde{q}_1 \wedge \tilde{q}_2, \tilde{q}_4, \tilde{q}_3, \tilde{q}_2, \tilde{q}_1\}$ associated with every $\{q_1, q_2, q_3, q_4\}$ with the same energy, it does not necessarily hold that the lowest 2^4 out of the 2^6 energies of $H_{toy,reduced}$ consist of the 2^4 energies of H_{toy}. For example, if we pick a small penalty δ in Table IV, say $0 \leq \delta \leq 4$, then some of the states penalized by either $H_\wedge(\tilde{q}_1, \tilde{q}_2, \tilde{q}_5)$ or $H_\wedge(\tilde{q}_3, \tilde{q}_4, \tilde{q}_6)$ can still have an energy within the energy values of H_{toy}. To avoid this situation, we can choose $\delta > \max(H_{toy})$ which will be sufficient to remove the energies of the penalized states from the region corresponding to energies of H_{toy}, therefore conserving the low-lying spectra of the original H_{toy}. Using the mapping explained at the end of Sec. II A, the quantum version of the 4-local energy function from Eq. 47 is:

$$\hat{H}_{toy} = I + \hat{\tilde{q}}_1 - \hat{\tilde{q}}_2 + \hat{\tilde{q}}_3 + \hat{\tilde{q}}_4 - \hat{\tilde{q}}_1\hat{\tilde{q}}_2\hat{\tilde{q}}_3 + \hat{\tilde{q}}_1\hat{\tilde{q}}_2\hat{\tilde{q}}_3\hat{\tilde{q}}_4. \tag{51}$$

The quantum version of the 2-local reduced form presented in Eq. 50 is,

$$\begin{aligned}\hat{H}_{toy,reduced} = I + \hat{\tilde{q}}_1 - \hat{\tilde{q}}_2 + \hat{\tilde{q}}_3 + \hat{\tilde{q}}_4 - \hat{\tilde{q}}_5\hat{\tilde{q}}_3 + \hat{\tilde{q}}_5\hat{\tilde{q}}_6 + \delta(3\hat{\tilde{q}}_5 + \hat{\tilde{q}}_1\hat{\tilde{q}}_2 - 2\hat{\tilde{q}}_1\hat{\tilde{q}}_5 - 2\hat{\tilde{q}}_2\hat{\tilde{q}}_5) \\ + \delta(3\hat{\tilde{q}}_6 + \hat{\tilde{q}}_3\hat{\tilde{q}}_4 - 2\hat{\tilde{q}}_3\hat{\tilde{q}}_6 - 2\hat{\tilde{q}}_4\hat{\tilde{q}}_6)\end{aligned} \tag{52}$$

Notice that $\hat{H}_{toy}$ acts on a 2^4 dimensional Hilbert space, $\text{span}\{|\tilde{q}_4\rangle \otimes |\tilde{q}_3\rangle \otimes |\tilde{q}_2\rangle \otimes |\tilde{q}_1\rangle\}$, while $\hat{H}_{toy,reduced}$ acts on a 2^6 dimensional Hilbert space, $\text{span}\{|\tilde{q}_6\rangle \otimes |\tilde{q}_5\rangle \otimes |\tilde{q}_4\rangle \otimes |\tilde{q}_3\rangle \otimes |\tilde{q}_2\rangle \otimes |\tilde{q}_1\rangle\}$.

Due to the conservation of the spectrum and bit strings described above (as reflected in Tables II and IV), the solution obtained from an adiabatic quantum algorithm using either $\hat{H}_{toy}$ or $\hat{H}_{toy,reduced}$ as $\hat{H}_{final}$,

$$\hat{H}(t) = (1 - t/\tau)\hat{H}(0) + (t/\tau)\hat{H}_{final} \tag{53}$$

should be the same.

In the case of the 2-local Hamiltonian $\hat{H}_{toy,reduced}$, the solution to the optimization problem is obtained using an adiabatic algorithm after reading the qubits associated to $\tilde{q}_4, \tilde{q}_3, \tilde{q}_2, \tilde{q}_1$ at $t = \tau$ from the space $\text{span}\{|\tilde{q}_6\rangle \otimes |\tilde{q}_5\rangle \otimes |\tilde{q}_4\rangle \otimes |\tilde{q}_3\rangle \otimes |\tilde{q}_2\rangle \otimes |\tilde{q}_1\rangle\}$ at $t = \tau$. Notice that the ancillary qubits in the six qubit version do not carry any physical information, as expected, since all of the valuable information was stored in the qubits coming from the

original expression before the reduction. The cost of reducing the locality of a Hamiltonian to another which contains at most two-body interactions is the increase in the number of resources due to the additional ancillary bits.

Figure 7 shows the the eigenenergies of Eq. 53 vs. t/τ, where $\hat{H}_{final}$ is replaced by $\hat{H}_{toy}$ (see Figure 7(a)), and by $\hat{H}_{toy,reduced}$ with $\delta = 5$, (see Fig. 7(b)). As expected from Table II and IV, Fig. 7 illustrates the preservation of the subsystem corresponding to the variables $\tilde{q}_1, \tilde{q}_2, \tilde{q}_3$ and $\tilde{q}_4$ in the ground state of both the original and reduced-locality Hamiltonian. Degeneracy and overlap of lines in the spectra in Fig. 7 make it difficult to graphically convey that both spectra in Fig. 7 indeed have 16 states for $0 \leq$ eigenenergies ≤ 4. In Fig. 7(b) we plotted the first 19 eigenstates out of the 2^6 eigenstates corresponding to $\hat{H}_{toy,reduced}$. At $t/\tau = 1$, states with energy greater than 4 correspond to states which violate the AND condition introduced by the reduction process. Notice that there are two eigenstates with eigenvalue 5 in agreement with the table presented in Appendix B after substituting $\delta = 5$, and one state which corresponds to the one of the four-degenerate manifold with $E = 6$.

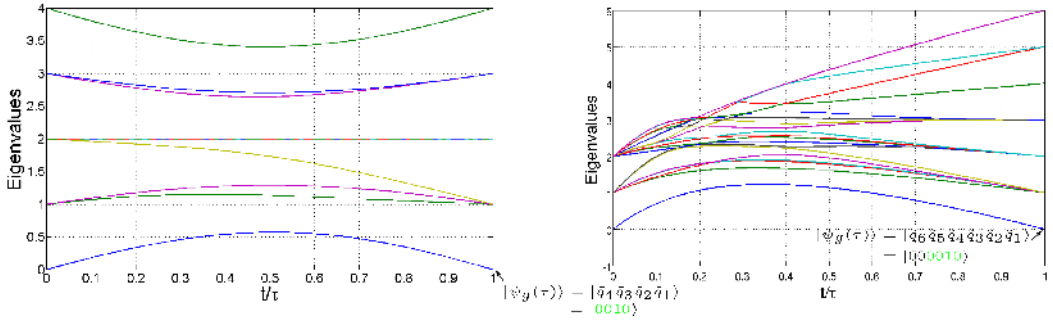

FIG. 7: (Color online) Spectrum comparison of the instantaneous energy eigenvalues for the 4-local toy Hamiltonian $\hat{H}_{toy}$ (left) and its corresponding 2-local version $\hat{H}_{toy,reduced}$(right). (left) Full spectrum of the 2^4 instantaneous eigenvalues for $\hat{H}_{toy}(\tilde{q},\tilde{q}_2,\tilde{q}_3,\tilde{q}_4)$. (right) First 19 instantaneous eigenvalues for the 2-local version of $\hat{H}_{toy}$, denoted as $\hat{H}_{toy,reduced}$ in text. The value used for δ is 5. The first 2^4 levels, $0 \leq$ eigenvalues ≤ 4, are associated to the original levels from $\hat{H}_{toy}$. The three remaining states with eigenvalues greater than 4 are penalized states which violate the conditions $\tilde{q}_n = \tilde{q}_i \wedge \tilde{q}_j$ (see Table IV for details)

VI. RESOURCES NEEDED FOR A 2-LOCAL HAMILTONIAN EXPRESSION IN PROTEIN FOLDING

For any k-local energy function, e.g., $h = q_1 q_2 \cdots q_k$, the reduction can be carried out iteratively, adding the penalty function $H_\wedge(q_i, q_j, \tilde{q}_n)$ for every substitution of the form $q_i q_j \to \tilde{q}_n$. For a k-local term, $(k-2)$ substitutions are required for the reduction to 2-local, and therefore require $(k-2)$ ancillary bits.

In the particular case of the protein Hamiltonian the reduction procedure needs to be repeated $(N-2)(N^D - D\log_2 N - 1)$ times, as described below. All the terms in the HP Hamiltonian include among at most interactions two amino acids, which results in a maximum locality of $2D\log_2 N$. In the following discussion, the cluster notation $[k][l]$ specifies the contributions of a particular $(k+l)-$local term into k variable coming from an amino acid with index i and l variables from an amino acid with index j. Since all the terms are of this form, to obtain a 2-local Hamiltonian, all products corresponding to each $[k]$ and $[l]$ of each cluster have to be converted to 1-local terms. We reduce terms for variables describing each amino acid in turn, for a total of $D\log_2 N$ variables. All possible combinations of two variables from the $D\log_2 N$ variables for an amino acid are substituted. The number of ancillary bits required for this substitution is $\binom{D\log_2 N}{2}$. These substitutions convert all terms of the form [3][0] and [2][1] to 2-local. To convert terms of the form [4][0] or [3][1] to 2-local we need to consider $\binom{D\log_2 N}{3}$ terms originally containing three variables from one amino acid. After employing an additional ancillary bit per term and applying the previous reduction step, all these terms collapse to 1-local with respect to the i-th amino acid, i.e., these terms will assume the form $[1][l]$. Iterating over the $D\log_2 N$ variables for a specific amino acid in order of increasing locality will give us the number of substitutions or ancilla bits needed per amino acid in order to reduce a particular cluster $[k]$ to $[1]$ or 1-local. The total number of substitutions per amino acid corresponds to $\sum_{k=2}^{D\log_2 N}\binom{D\log_2 N}{k} = N^D - D\log_2 N - 1$. To carry out the procedure for all $(N-2)$ amino acids the number of ancilla qubits required is $(N-2)(N^D - D\log_2 N - 1)$. The number of qubits needed to represent a 2-local Hamiltonian version of the protein Hamiltonian is given by adding the number of ancillary qubits to the number of original $(N-2)D\log_2 N$

quantum bits,

$$\begin{aligned} \#\text{ of total qubits for a 2-local expression} &= (N-2)(N^D - D\log_2 N - 1) + (N-2)D\log_2 N \\ &= (N-2)(N^D - 1) \end{aligned} \tag{54}$$

Eq. 54 provides a closed formula for the number of qubits needed to find the lowest energy conformations for a protein with N amino acids in D dimensions in our encoding. In particular, for the case of a four amino acid peptide HPPH in two dimensions considered in Sec. IV requires 30 qubits.

VII. CONCLUSIONS

We constructed the essential elements of an adiabatic quantum algorithm to find the lowest energy conformations of a protein in a lattice model. The number of binary variables needed to represent N amino acids on an $N \times N$ lattice is $(N-2)D\log_2 N$. The maximum locality of the final Hamiltonian, as determined by the interaction between pairs of amino acids using the mapping defined here, is $2D\log_2 N$.

General strategies to construct energy functions to map into other quantum mechanical Hamiltonians used for adiabatic quantum computing were presented. The strategies used in the construction of the Hamiltonian for the HP model can be used as general building blocks for Hamiltonians associated with physical systems where onsite energies and/or pairwise potentials are present.

We also demonstrated an application of the Boolean scheme for converting a k-local Hamiltonian into a 2-local Hamiltonian, aiming toward an experimental implementation in quantum devices. The resulting couplings, although 2-local, do not necessarily represent couplings among nearest neighbor quantum bits in a two-dimensional geometry. It is however known that the number of ancillary physical qubits required to embed an arbitrary N variable problem is upper-bounded by $N^2/(C-2)$, where C is the number of couplers allowed per physical qubit.

The most important question remaining to be explored in future work is the scaling of run time τ with respect to the number of amino acids N. Run time τ is dependent on the particular instance of the problem – in our case, to different protein sequences. It has been proposed that proteins have evolved towards a many-dimensional funnel-like potential

energy surface [7]. The sequences that show a funnel-like structure might be easier to study using adiabatic quantum computation, because the funnel structure may facilitate annealing of the quantum wave function toward low energy conformations.

Acknowledgments

We thank Jacob Biamonte, Sergio Boxio, Ivan Kassal, William Macready, Peter McMahon and Rolando Somma for helpful discussions. Partial funding for this project was provided by a D-Wave Systems Inc. research contract and the Institute for Quantum Science and Enginnering at Harvard University.

APPENDIX A: EXTENSION OF THE PAIRWISE INTERACTION TO THREE DIMENSIONS AND N AMINO ACIDS , $N = 2^M$ AND $M \geq 3$

This extension follows the principles presented in Sec. III B 3 and extends the terms of the Hamiltonian to the case of a three-dimensional lattice protein. The pairwise term for the three-dimensional case is,

$$H^{3D}_{pairwise}(N) = -\sum_{i,j=1}^{N} G_{ij} H^{ij,3D}_{pairwise} \tag{A1}$$

$$\begin{aligned} x_{+}^{ij,3D}(N) = (1 - q_{f(i,1)+1}) q_{f(j,1)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,1)+s} \\ - q_{f(i,1)+s} + 2\, q_{f(j,1)+s}\, q_{f(i,1)+s}) \prod_{s=1}^{\log_2 N} (1 - q_{f(i,2)+s} \\ - q_{f(j,2)+s} + 2 q_{f(i,2)+s} q_{f(j,2)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,3)+r} \\ - q_{f(j,3)+r} + 2 q_{f(i,3)+r} q_{f(j,3)+r}), \end{aligned} \tag{A2}$$

$$y_{+}^{ij,3D}(N) = (1 - q_{f(i,2)+1})q_{f(j,2)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,2)+s} - q_{f(i,2)+s} + 2\, q_{f(j,2)+s}\, q_{f(i,2)+s}) \prod_{s=1}^{\log_2 N} (1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,3)+r} - q_{f(j,3)+r} + 2q_{f(i,3)+r}q_{f(j,3)+r}), \quad \text{(A3)}$$

$$z_{+}^{ij,3D}(N) = (1 - q_{f(i,3)+1})q_{f(j,3)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,3)+s} - q_{f(i,3)+s} + 2\, q_{f(j,3)+s}\, q_{f(i,3)+s}) \prod_{s=1}^{\log_2 N} (1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,2)+r} - q_{f(j,2)+r} + 2q_{f(i,2)+r}q_{f(j,2)+r}), \quad \text{(A4)}$$

$$x_{-}^{ij,3D}(N) = (1 - q_{f(i,1)+1})q_{f(j,1)+1}\Big[1 - \prod_{k=1}^{\log_2 N} (1 - q_{f(i,1)+k})\Big](q_{f(j,1)+2} + q_{f(i,1)+2} - 2\, q_{f(j,1)+2}\, q_{f(i,1)+2}) \prod_{r=3}^{\log_2 N} \Big[1 - (q_{f(j,1)+r} + \prod_{u=2}^{r-1} q_{f(j,1)+u} - 2\prod_{u=2}^{r} q_{f(j,1)+u}) - q_{f(i,1)+r} + 2q_{f(i,1)+r}(q_{f(j,1)+r} + \prod_{u=2}^{r-1} q_{f(j,1)+u} - 2\prod_{u=2}^{r} q_{f(j,1)+u})\Big] \prod_{s=1}^{\log_2 N} (1 - q_{f(i,2)+s} - q_{f(j,2)+s} + 2q_{f(i,2)+s}q_{f(j,2)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,3)+r} - q_{f(j,3)+r} + 2q_{f(i,3)+r}q_{f(j,3)+r}), \quad \text{(A5)}$$

$$y_{-}^{ij,3D}(N) = (1 - q_{f(i,2)+1})q_{f(j,2)+1}\Big[1 - \prod_{k=1}^{\log_2 N}(1 - q_{f(i,2)+k})\Big](q_{f(j,2)+2} + q_{f(i,2)+2} - 2\,q_{f(j,2)+2}\,q_{f(i,2)+2})$$
$$\prod_{r=3}^{\log_2 N}\Big[1 - (q_{f(j,2)+r} + \prod_{u=2}^{r-1} q_{f(j,2)+u} - 2\prod_{u=2}^{r} q_{f(j,2)+u}) - q_{f(i,2)+r} + 2q_{f(i,2)+r}(q_{f(j,2)+r} + \prod_{u=2}^{r-1} q_{f(j,2)+u} - 2\prod_{u=2}^{r} q_{f(j,2)+u})\Big]$$
$$\prod_{s=1}^{\log_2 N}(1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s})$$
$$\prod_{r=1}^{\log_2 N}(1 - q_{f(i,3)+r} - q_{f(j,3)+r} + 2q_{f(i,3)+r}q_{f(j,3)+r}), \tag{A6}$$

$$z_{-}^{ij,3D}(N) = (1 - q_{f(i,3)+1})q_{f(j,3)+1}\Big[1 - \prod_{k=1}^{\log_2 N}(1 - q_{f(i,3)+k})\Big](q_{f(j,3)+2} + q_{f(i,3)+2} - 2\,q_{f(j,3)+2}\,q_{f(i,3)+2})$$
$$\prod_{r=3}^{\log_2 N}\Big[1 - (q_{f(j,3)+r} + \prod_{u=2}^{r-1} q_{f(j,3)+u} - 2\prod_{u=2}^{r} q_{f(j,3)+u}) - q_{f(i,3)+r} + 2q_{f(i,3)+r}(q_{f(j,3)+r} + \prod_{u=2}^{r-1} q_{f(j,3)+u} - 2\prod_{u=2}^{r} q_{f(j,3)+u})\Big]$$
$$\prod_{s=1}^{\log_2 N}(1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s})$$
$$\prod_{r=1}^{\log_2 N}(1 - q_{f(i,2)+r} - q_{f(j,2)+r} + 2q_{f(i,2)+r}q_{f(j,2)+r}). \tag{A7}$$

TABLE IV: Truth table for the energy function $H_{toy,reduced}(\tilde{q}) = 1 + \tilde{q}_1 - \tilde{q}_2 + \tilde{q}_3 + \tilde{q}_4 - \tilde{q}_5\tilde{q}_3 + \tilde{q}_5\tilde{q}_6 + \delta(3\tilde{q}_5 + \tilde{q}_1\tilde{q}_2 - 2\tilde{q}_1\tilde{q}_5 - 2\tilde{q}_2\tilde{q}_5) + \delta(3\tilde{q}_6 + \tilde{q}_3\tilde{q}_4 - 2\tilde{q}_3\tilde{q}_6 - 2\tilde{q}_4\tilde{q}_6)$. The top of the table shows the 16 non-penalized states that satisfy $\tilde{q}_5 = \tilde{q}_1 \wedge \tilde{q}_2$ and $\tilde{q}_6 = \tilde{q}_3 \wedge \tilde{q}_4$. These 16 states map one to one to the states in Table II. A sample of the remaning 48 penalized states are shown after the breaking line.

$\tilde{q}_6$	$\tilde{q}_5$	$\tilde{q}_4$	$\tilde{q}_3$	$\tilde{q}_2$	$\tilde{q}_1$	$H'(\tilde{q}_1, \tilde{q}_2, \tilde{q}_3, \tilde{q}_4, \tilde{q}_5, \tilde{q}_6)$.
0	0	0	0	1	0	0
0	0	0	0	0	0	1
0	1	0	0	1	1	1
0	0	0	1	1	0	1
0	0	0	1	1	1	1
0	0	1	0	1	0	1
0	0	0	0	0	1	2
0	0	0	1	0	0	2
0	0	1	0	0	0	2
0	1	1	0	1	1	2
1	0	1	1	1	0	2
0	0	0	1	0	1	3
0	0	1	0	0	1	3
1	0	1	1	0	0	3
1	1	1	1	1	1	3
1	0	1	1	0	1	4
0	1	0	0	1	0	δ
0	1	0	1	1	0	δ
0	0	0	0	1	1	$1 + \delta$
0	1	1	0	1	0	$1 + \delta$
0	1	1	0	1	0	$1 + \delta$
1	0	0	1	1	0	$1 + \delta$
1	0	1	0	1	0	$1 + \delta$
0	0	0	1	1	1	$2 + \delta$
0	0	1	0	1	1	$2 + \delta$
⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	0	0	$3 + 3\ \delta$
1	0	0	0	1	1	$1 + 4\ \delta$
1	1	0	0	1	0	$1 + 4\ \delta$
1	1	0	1	0	0	$2 + 4\ \delta$
1	1	0	0	0	1	$3 + 4\ \delta$
1	1	1	0	0	0	$3 + 4\ \delta$
1	1	0	0	0	0	$2 + 6\ \delta$

APPENDIX B: TRUTH TABLE FOR THE FUNCTION RESULTING AFTER THE REDUCTION OF LOCALITY

---

[1] H. S. Chan and K. A. Dill, *Physics Today* **46**, 24 (1993).

[2] S. S. Plotkin and J. N. Onuchic, *Quarterly Reviews of Biophysics* **35** (2002).

[3] E. Shaknovich, *Chemical Reviews* **106** (2006).

[4] L. Mirny and E. Shaknovich, *Annual Review of Biophysics and Biomolecular Structure* **30** (2001).

[5] K. Dill, S. Bromberg, K. Yue, K. Fiebig, D. Yee, P. Thomas, and H. Chan, *Protein Science* **4**, 561 (1995).

[6] M. Gruebele, *Annual Review of Physical Chemistry* **50**, 485 (1999).

[7] T. E. Creighton, editor, *Protein folding*, New York: W. H. Freeman, 1992.

[8] C. J. Epstein, R. F. Goldberger, and C. B. Anfinsen, *Cold Spring Harbor symposia on quantitative biology* **28**, 439 (1963).

[9] D. Baker and D. A. Agard, *Biochemistry* **33**, 7505 (1994).

[10] T. Lazaridisa and M. Karplus, *Biophysical Chemistry* **100**, 367 (2003).

[11] P. W. Shor, *SIAM Journal on Computing* **26**, 1484 (1997).

[12] A. Aspuru-Guzik, A. Dutoi, P. Love, and M. Head-Gordon, *Science* **309**, 1704 (2005).

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

[14] K. Steinhfel, A. Skaliotisa, and A. A. Albrecht, *Computer Physics Communications* **176**, 465 (2007).

[15] G. A. Cox, T. V. Mortimer-Jones, R. P. Taylor, and R. L. Johnston, *Theor. Chem. Acc.* **112**, 163 (2004).

[16] G. A. Cox and R. L. Johnston, *J. Chem. Phys.* **124**, 204714 (2006).

[17] F. L. Custdio, H. J. C. Barbosa, and L. E. Dardenne, *Genetics and Molecular Biology* **27**, 611 (2004).

[18] R. Unger and J. Moult, *J. Mol. Biol.* **231**, 75 (1993).

[19] J. Song, J. Cheng, T. Zheng, and J. mao, *Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT'05)* ,

935 (2005).

[20] A. Shmygelska and H. H. Hoos, *BMC Bioinformatics* **6** (2005).

[21] R. Backofen and S. Will, *Proc. XIX Intl. Conf. on Logic Programming* , 49 (2003).

[22] T. C. Beutler and K. A. Dill, *Protein Science* **5**, 2037 (1996).

[23] H. P. Hsu, V. Mehra, W. Nadler, and P. Grassberger, *Physical Review E* **68**, 021113 (2003).

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

[25] K. Yue, K. M. Fiebig, P. D. Thomas, H. S. Chan, E. I. Shaknovich, and K. A. Dill, *Proc. Nat. Acad. Sci. USA* **92**, 325 (1995).

[26] P. Crescenzi, D. Goldman, C. Papadimitriou, A. Piccolboni, and M. Yannakakis, *Journal of Computational Biology* **5**, 423 (FAL 1998).

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

[28] W. M. Kaminsky, S. Lloyd, and T. P. Orlando, Scalable Superconducting Architecture for Adiabatic Quantum Computation, quant-ph/0403090v1.

[29] R. Harris, A. Berkley, M. Johnson, P. Bunyk, S. Govorkov, M. Thom, S. Uchaikin, A. Wilson, J. Chung, E. Holtham, J. Biamonte, A. Yu, M. Amin, and A. van den Brink, *Physical Review Letters* **98**, 177001 (2007).

[30] D. P. DiVincenzo, *Fortschritte der Physik* **48**, 771 (2000).

[31] S. Bravyi, D. P. DiVincenzo, R. Oliveira, and B. M. Terhal, quant-ph/0606140.

[32] A. Messiah, *Quantum Mechanics, Vol. II*, Wiley, New York, 1976.

[33] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, arXiv:quant-ph/0001106v1.

[34] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *Science* **292**, 472 (2001).

[35] K. Rosen, *Discrete Mathematics and its applications*, McGraw-Hill, 1999.

[36] H. Li, C. Tang, and N. Wingreen, *Proteins: Structure, Function, and Genetics* **49**, 403 (2002).

[37] J. D. Biamonte, arXiv:0801.3800v2 [quant-ph].

[38] E. Boros and P. Hammer, *Discrete Appl. Math.* **123**, 155 (2002).