On the construction of model Hamiltonians for adiabatic quantum computing 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*

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 quantum system of coupled quantum bits. General strategies for constructing Hamiltonians to be used to solve optimization problems of physical/chemical/biological interest via quantum computation by adiabatic evolution are given. As an example we implemented the Hamiltonian corresponding of 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

Classical or quantum devices could be constructed in such a way that their energy spectrum is analogous to the spectrum of a problem of physical importance. These devices could be annealed to their local or global energy minima, which might encode the solution to the computational problem. Many examples of possible systems on which to apply this technique have been studied using methods of statistical mechanics, such as simulated annealing [1] and quantum annealing [2–5]. For example, problems related to polymer physics, lattice models and protein folding fall into this category [6–8]. The physical annealing process of these devices, either by a classical or a quantum process, may be competitive with more traditional computational approaches, such as digital computers. One of the goals of this study is to describe in detail the construction of an useful Hamiltonian related to protein folding for its use in this context.

Finding the ensemble of low-energy conformations of a peptide given its primary sequence is a fundamental problem of computational biology and is commonly known as the "protein-folding problem" [9–15]. It is commonly assumed that the native fold of a protein is the conformation corresponding to the lowest minimum free energy (the so-called thermodynamic hypothesis [16]), although some exceptions have been proposed [17, 18]; therefore, 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 [19] and the calculation of molecular energies [20]. In this report, we investigate the approach of using an adiabatic quantum computer for folding a highly simplified protein model.

The protein-folding problem comprises three parts: 1) the model used to represent the protein; 2) the energy function; and 3) the search algorithm. The models range from full atomistic detail to simplified models where each amino acid residue is treated as a single point [12]. However, even for the simplest of models, the classical time complexity (the time it takes to solve the problem using a classical computer) of the optimization scales exponentially with the size of the system [21].

The HP (H: hydrophobic, P: polar) lattice model [21] is one of the simplest protein models that still retains some of the folding behavior of real proteins. As such, it has been a useful benchmark for testing optimization algorithms such as simulated annealing [22], genetic algorithms [23–27], and ant colony optimization [28]. Other heuristic methods such as hydrophobic core threading [29], chain growth [30, 31], contact interactions [32], and hydrophobic zippers [33] have also been considered. The HP model has also been useful for a qualitative investigation of the nature of the folding process and the interaction between proteins. The HP model, as 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 points on a grid, which is typically a square grid for the 2D model and a cubic grid for the 3D model; and 3) The only interaction amongst amino acids is the favorable contact between two H residues that are not neighbors in the sequence; this interaction is defined to have an energy of -1 in arbitrary units representing a hydrophobic effect which tends to fold the protein in such a way that the H residues end up buried in a predominantly hydrophobic core, while the P or hydrophilic residues remain 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 [21], where N is the length of the polypeptide chain. The proofs of the NP-completeness of both the 2D and 3D HP models can be found in the literature [34, 35]. Due to this exponential growth, exhaustive enumeration (the only certain way of finding the global minimum using a classical computer) becomes intractable when N

reaches 30 to 40 residues, depending on the hardware resources. For longer sequences, heuristics and stochastic algorithms have been employed for N up to 136 for the 3D HP model [32]; the global minimum for the $N = 136$ test sequence is not known with certainty. Another notable property of the HP model in two dimensions is the high degeneracy of the global minima for most sequences. For $N = 20$, only about 2% of the possible sequences have a single global minimum (not counting the mirror image).

Sec. II presents the general quantum algorithm, the terms of the Hamiltonian necessary to obtain the folded structure of the protein, and a description of all techniques used to map the problem to arrays of coupled quantum bits [36, 37]. Sec. III explains the construction of the core piece of the algorithm, i.e. the Hamiltonian that encodes the lowest-energy conformation of the protein. Finally, in Sec. IV-VI we solve in detail the four amino acid sequence HPPH in a two-dimensional grid.

II. IMPLEMENTATION OF AN ADIABATIC QUANTUM ALGORITHM TO SOLVE THE HP MODEL

We begin this section by describing the mapping of the N amino acid sequence into binary variables, which in turn will be mapped to spin variables in the quantum mechanical version of the algorithm. Following, we discuss strategies for finding the global minima that involve quantum mechanics, such as quantum adiabatic evolution. This could be realized using quantum computers or quantum devices, when and if they become available. Recent progress in architectures such as superconducting Josephson junctions is promising [36–39]. An alternative to this approach is to simulate the quantum evolution either fully or approximately using a classical computer. This technique is known in the literature as quantum annealing [2–5].

A. Mapping of the amino acids onto the 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 (as shown in Fig. 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. Let q denote a particular configuration of the protein in the grid, written in the form

$$q = \underbrace{q_{16}q_{15}}_{y_4}\ \underbrace{q_{14}q_{13}}_{x_4}\ \underbrace{q_{12}q_{11}}_{y_3}\ \underbrace{q_{10}q_{9}}_{x_3}\ \underbrace{q_{8}q_{7}}_{y_2}\ \underbrace{q_{6}q_{5}}_{x_2}\ \underbrace{q_{4}q_{3}}_{y_1}\ \underbrace{q_{2}q_{1}}_{x_1}, \tag{1}$$

where x_i and y_i are the x and y coordinate of the ith amino acid. Notice that for a D dimensional lattice and a number N of amino acids, the number of Boolean variables q_i forming the bit string q is equal to $DN \log_2 N$. For example, for the particular case of $N = 4$, $D = 2$, the length of the bit string q is 16 and therefore the number of configurations that can be explored is 2^{16}. In the quantum version of the problem, we consider these configurations to span a Hilbert space of dimension 2^{16}, and the state vectors can be written as

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

For the purposes of implementing the Hamiltonian which encodes the ground state of the protein on a spin-1/2 quantum computer [40], or in particular onto an Ising-like Hamiltonian with a transverse magnetic field [41] (see Sec. II B), the 16-qubit Hilbert space can be realized 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 $|\sigma_i^z = -1\rangle$. In other words, the quantum version of the configuration states are related to spin variables through the transformation

$$q_i = \frac{1}{2}(1 - \sigma_i^z). \tag{3}$$

Fig. 2 shows an example of the coordinate mapping given a specific sequence of residues or amino acids. The first step is to fix the two middlemost amino acids in a central position. In the case of four amino acids, fixing these two reduces translational symmetry. This leads to a reduction of the number of binary variables from sixteen to eight, i.e., 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, the two middle ones, have a constant fixed value throughout the optimization process. A general way to do this is to assign the $(N/2)^{th}$ amino acid to the $(N/2)^{th}$ grid point in all D dimensions and then fix the $(N/2+1)^{th}$ amino acid 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 hardware proposed for obtaining the lowest-energy conformation of the protein is composed of superconducting flux qubits arranged in a square lattice, with couplers connecting nearest neighbors and next-nearest neighbors [36]. The quantum computational model upon which the hardware is based is that of adiabatic quantum computation [42–44]. In adiabatic quantum computing, a time-evolving quantum state $|\psi(t)\rangle$ represents the possible solutions to the problem of interest as a superposition of states in a given computational basis. The goal of the algorithm is to transform the initial state into a final state which encodes the answer to the problem. The specific

constraints of the problem are encoded in the final Hamiltonian.

The state $|\psi(t)\rangle$ evolves in time according to the Schrödinger equation,

$$i\hbar\frac{d}{dt}|\psi(t)\rangle = H(t)|\psi(t)\rangle \tag{4}$$

where $H(t)$ is the time-dependent Hamiltonian operator. The design of the quantum computer algorithm takes advantage of the quantum adiabatic theorem [45], which is satisfied whenever $H(t)$ is slowly varying throughout the time of propagation $t \in [0,\tau]$. Let $|\psi_g(t)\rangle$ be the instantaneous ground state of $H(t)$. If we construct $H(t)$ such that the ground state of $H(0)$, 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 $t \in [0,\tau]$. The form of $H(0)$ is usually constructed in such a way that $|\psi_g(0)\rangle$ is a uniform superposition of all possible configurations of the system, i.e.

$$|\psi_g(0)\rangle = \frac{1}{\sqrt{2^{16}}}\sum_{q_i\in\{0,1\}}|q_{16}\rangle|q_{15}\rangle\cdots|q_2\rangle|q_1\rangle \tag{5}$$

where the sum is over all 2^{16} basis vectors. The initial Hamiltonian, $H(0)$, is achieved by applying a magnetic field in the $x-$direction to each quantum spin. More details can be found in the work of Farhi *et al.* [42].

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

$$H(t) = (1 - t/\tau)H(0) + (t/\tau)H_{protein} \tag{6}$$

Even though Eq. 6 connects smoothly $H(0)$ and $H_{protein}$, determining the optimum value of τ is by itself a very important and non-trivial question. In principle, the adiabatic theorem states that for a long enough adiabatic time τ, the state $\psi(\tau)$ will be the right solution to the problem $|\psi_g(\tau)\rangle$. To determine how long is long enough is what dictates the ultimate usefulness of the quantum algorithm proposed in this work. Farhi *et al* [42, 43] showed that it is possible to obtain quantum speed-up for some hard instances of some NP-complete problems.

Notice that the parameter τ determines the rate at which $H(t)$ varies. Following the notation from Farhi *et al* [42], consider $H(t) = \tilde{H}(t/\tau) = \tilde{H}(s)$, and the instantaneous values of $\tilde{H}(s)$ are defined by

$$\tilde{H}(s)|l;s\rangle = E_l(s)|l;s\rangle \tag{7}$$

with

$$E_0(s) \le E_1(s) \le \cdots \le E_{N-1}(s) \tag{8}$$

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 strictly greater than zero for all $0 \le s \le 1$, and taking

$$\tau \gg \frac{\varepsilon}{g_{min}^2} \tag{9}$$

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

$$g_{min} = \min_{0\le s\le 1}(E_1(s) - E_0(s)), \tag{10}$$

and ε given by

$$\varepsilon = \max_{0\le s\le 1}|\langle l=1;s|\frac{d\tilde{H}}{ds}|l=0;s\rangle|, \tag{11}$$

then we can make

$$|\langle l=0;s=1|\psi(\tau)\rangle| \tag{12}$$

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 $H(t)$ for all $0 \le t \le \tau$, if τ is big enough.

In the following section we will derive the expression for $H_{protein}$ for an array of coupled 2-level quantum systems.

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

Our goal in this section is to find an algebraic expression for a Hamiltonian whose ground state is the lowest-energy conformation of the protein. Ideally, the Hamiltonian should have the least possible number of terms. In order to optimize the computational resources, we also want to have terms with low locality. Locality is defined as the number of products of q_i's that appear in a certain term, e.g., a term of the form $h = q_1q_3q_4q_6$ is 4-local.

A. Small computer science digression

Since the position of the amino acids in the grid are given in terms of Boolean variables, it is very convenient to use tools from computer science and basic Boolean algebra [46]. In this section, we will review these tools for the purpose of developing a tool for constructing arbitrary Hamiltonians for 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 AND, OR, and NOT operations can be written algebraically as:

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

An additional useful expression for the construction of Hamiltonian terms is the XNOR operation. The XNOR Boolean function has the true logic value 1 as an output, if all its arguments have the same value. The two-input version XNOR operation is also known as *logical equality*, here denoted as EQ,

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

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

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. Therefore, we need to find a Hamiltonian that energetically favors valid configurations of the HP model, by eliminating configurations where two amino acids are on the same grid point, and discarding the configurations that violate the primary sequence of amino acids. This Hamiltonian can be written as

$$H_{protein} = H_{onsite} + H_{psc} + H_{pairwise}, \qquad (13)$$

where H_{onsite} is an onsite repulsion term, H_{psc} is a primary structure constraint term, and $H_{pairwise}$ is pairwise interaction term that represents favorable hydrophobic interactions.

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, which without loss of generality can be made a multiple of two. Following, we describe each one of the terms in Eq. 13.

1. Onsite term, H_{onsite}

The first term in Eq. 13, H_{onsite}, prevents two or more amino acids from occupying the same grid point. If one position variable is different between two amino acids, then H_{onsite} must evaluate to zero. As an illustrative example, a simple one-dimensional two-site Hamiltonian is shown in Fig. 3 and described in Sec. III A.

The general term for D dimensions and N amino acids is

$$H_{onsite}(N,D) = \lambda_0 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} H^{ij}_{onsite}(N,D) \qquad (14)$$

with

$$H^{ij}_{onsite}(N,D) = \prod_{k=1}^{D} \prod_{r=1}^{\log_2 N} \Big(1 - q_{f(i,k)+r} - q_{f(j,k)+r} + 2\, q_{f(i,k)+r}\, q_{f(j,k)+r}\Big) \qquad (15)$$

and

$$f(i,k) = D(i-1)\log_2 N + (k-1)\log_2 N \qquad (16)$$

The terms enclosed in the parentheses of Eq. 15 are XNOR functions. The double product of these terms tests that all of these conditions are considered simultaneously by using AND-type relations. In other words, if all the binary variables describing the coordinates of the ith and jth 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 ith and jth amino acids is different.

The function $f(i,k)$ is a pointer to the bit string describing the coordinates of a particular amino acid. The index i points to the ith amino acid and the index k points to the first bit variable of the kth spatial coordinate. Here, $k = 1$ corresponds to the x coordinate, $k = 2$ to the y coordinate, and $k = 3$ to the z coordinate.

2. Primary structure constraint, H_{psc}

The term H_{psc} in Eq. 13 enforces the primary structure of the protein. The primary structure constraint evaluates to zero when two amino acids P and Q that are consecutive sequence-wise are restricted to be nearest neighbors. Nearest-neighbors must have a rectilinear (L_1) distance of $d_{PQ} = 1$ on the grid. For the derivation of this term, it is very useful to construct a function for the base 10 distance squared between any two amino acids P and Q on the grid. The distance function gives the base 10 distance squared between any two points P and Q on the grid.

$$d^2_{PQ}(N,D) = \sum_{k=1}^{D} \Big(\sum_{r=1}^{\log_2 N} 2^{r-1} (q_{f(P,k)+r} - q_{f(Q,k)+r}) \Big)^2 \qquad (17)$$

with $f(i,k)$ defined as in Eq. 16.

A simple way of defining H_{psc} is

$$H'_{psc}(N,D) = \lambda_1 \sum_{m=1}^{N-1} (1 - d^2_{m,m+1})^2 \qquad (18)$$

but an improved expression which achieves the same goal is

$$H_{psc}(N,D) = \lambda_1 \Big[-(N-1) + \sum_{m=1}^{N-1} d^2_{m,m+1} \Big] \qquad (19)$$

First notice that for valid configurations, all $(N-1)$ terms in the sum will equal one and then $H_{psc}(N,D)$ would evaluate to zero. If any of the $d^2_{m,m+1}$ terms is zero, then H_{onsite} will raise the energy abruptly through the penalty λ_0. This can be achieved setting $\lambda_1 < \lambda_0$. Now that we have discarded the possibility of any $d^2_{m,m+1}$ being zero,

what is left are configurations with values of $d^2_{m,m+1} > 1$. In these instances, $H_{psc}(N, D) > 0$ and λ_1 will play the role of an energy penalty ($\lambda_1 > 0$). Choosing $\lambda_0 = N$ and $\lambda_1 < \lambda_0$ will guarantee that the eigenstates of $H_{protein}$ associated with energies greater than zero correspond to unwanted or penalized configurations, while conformations associated with energies less than or equal to zero will correspond to plausible configurations of the protein.

Because $d^2_{PQ}(N, D)$ is always 2-local, this Hamiltonian term is always 2-local regardless of the number of amino acids or the dimensionality of the problem. Note that that Eq. 19 is superior to Eq. 18, which is 4-local.

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

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

The following general expression represents this kind of interaction:

$$H_{pairwise}(N, D) = -\sum_{i=1}^{N}\sum_{j=1}^{N} G_{ij} H^{ij}_{pairwise}, \qquad (20)$$

where G is an $N \times N$ symmetric matrix where the entries G_{ij} are $+1$ when two amino acids interact hydrophobically, and 0 otherwise. Note that G_{ij} is set to zero for amino acids that are neighbors in the protein sequence. Notice also that defining G_{ij} in a different way would allow us to specify lattice protein models that are more complex than the HP model. One of these models is the more realistic Miyazawa-Jernigan model [47] which includes interactions between 20 types of amino acids [47].

The form of $H^{ij}_{pairwise}$ depends on the number of spatial dimensions of the problem. For the two dimensional case, we have,

$$\begin{aligned} H^{ij}_{pairwise} &= H^{ij,2D}_{pairwise}(N) = x^{ij,2D}_{+}(N) + x^{ij,2D}_{-}(N) \\ &\quad + y^{ij,2D}_{+}(N) + y^{ij,2D}_{-}(N) \end{aligned} \qquad (21)$$

and in three dimensions,

$$\begin{aligned} H^{ij}_{pairwise} = H^{ij,3D}_{pairwise}(N) = x^{ij,3D}_{+}(N) + x^{ij,3D}_{-}(N) \\ + y^{ij,3D}_{+}(N) + y^{ij,3D}_{-}(N) + z^{ij,3D}_{+}(N) + z^{ij,3D}_{-}(N) \end{aligned} \qquad (22)$$

Each term in the right hand side of Eq. 22 acts independently and its purpose is to query whether the jth amino acid is to the right of the ith through $x^{ij,3D}_{+}$, or to the left through $x^{ij,3D}_{-}$, above through $y^{ij,3D}_{+}$, below through $y^{ij,3D}_{-}$, in front through $z^{ij,3D}_{+}$ or behind through $z^{ij,3D}_{-}$. If the jth amino acid is in any of these positions, i.e., a distance of one in any direction, $H^{ij}_{pairwise}$ will give $+1$; otherwise it will be zero. There is a subtle but important condition encoded in these terms: in order for any of these operators to act, the rightmost binary variable describing the ith 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 $z^{ij,3D}_{-}$) has to end in 0, i.e., the coordinate has to correspond to an even number. If this condition is not satisfied, then the term will vanish. This constraint explains the intentional double counting in Eq. 20 given by the freedom in indexes i and j running from 1 to N. Additionally, there is no special treatment 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. 20.

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 two amino acid case ($M = 2$) is much simpler and will be discussed in Sec. IV. The expression for $x^{ij,2D}_{+}(N)$ is,

$$\begin{aligned} x^{ij,2D}_{+}(N) = (1 - q_{f(i,1)+1}) q_{f(j,1)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,1)+s} \\ - q_{f(i,1)+s} + 2\, q_{f(j,1)+s}\, q_{f(i,1)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,2)+r} \\ - q_{f(j,2)+r} + 2 q_{f(i,2)+r} q_{f(j,2)+r}) \end{aligned} \qquad (23)$$

.

The first two factors of $x^{ij,2D}_{+}(N)$ (Eq. 23) treat the rightmost binary digit of the x position of the ith and jth amino acid. The first factor guarantees that the ith residue is in an even x position and tests if the jth residue is next to the ith. If this condition is satisfied, then the x position of the ith residue has to be odd. The case where the position of the ith residue is odd is dealt with the second factor $q_{f(j,1)+1}$. The remaining factors of $x^{ij,2D}_{+}$ can be interpreted as XNOR functions, equivalent to those used in the construction of the onsite Hamiltonian described in Sec. III B 1. These factors ensure that the rest of the binary digits that encode the x position are equal for the ith and jth residues. Finally, all the digits encoding the y position have to be equal, forcing the two residues to be in the same row. If all these conditions are satisfied, $x^{ij,2D}_{+}$ evaluates to $+1$ and otherwise 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 $y^{ij,3D}_{+}$ follows the same procedure as that of $x^{ij,3D}_{+}$, namely,

$$\begin{aligned} y^{ij,2D}_{+}(N) = (1 - q_{f(i,2)+1}) q_{f(j,2)+1} \prod_{s=2}^{\log_2 N} (1 - q_{f(j,2)+s} \\ - q_{f(i,2)+s} + 2\, q_{f(j,2)+s}\, q_{f(i,2)+s}) \prod_{r=1}^{\log_2 N} (1 - q_{f(i,1)+r} \\ - q_{f(j,1)+r} + 2 q_{f(i,1)+r} q_{f(j,1)+r}) \end{aligned} \qquad (24)$$

$$x_{-}^{ij,2D}(N) = (1-q_{f(i,1)+1})q_{f(j,1)+1}\Big[1-\prod_{k=1}^{\log_2 N}(1-q_{f(i,1)+k})\Big](q_{f(j,1)+2}+q_{f(i,1)+2}-2\,q_{f(j,1)+2}\,q_{f(i,1)+2})$$
$$\prod_{r=3}^{\log_2 N}\Big[1-(q_{f(j,1)+r}+\prod_{u=2}^{r-1}q_{f(j,1)+u}-2\prod_{u=2}^{r}q_{f(j,1)+u})$$
$$-q_{f(i,1)+r}+2q_{f(i,1)+r}(q_{f(j,1)+r}+\prod_{u=2}^{r-1}q_{f(j,1)+u}-2\prod_{u=2}^{r}q_{f(j,1)+u})\Big]$$
$$\prod_{s=1}^{\log_2 N}(1-q_{f(i,2)+s}-q_{f(j,2)+s}+2q_{f(i,2)+s}q_{f(j,2)+s}) \quad (25)$$

The construction of $x_{-}^{ij,2D}$ involves the following considerations. We have to consider the cases where the ith amino acid is in an even x position by means of the factor $(1-q_{f(i,1)+1})$. Since we are interested in querying whether the jth amino acid is to the left of the ith, we need to use a different procedure than that of Eq. 23. We add $00\cdots01$ to the x coordinate of the jth residue and then check by means of XNOR functions if the result matches the x coordinate of the ith amino acid. As in the previous case of $x_{+}^{ij,2D}$, we need to impose conditions that guarantee that the y coordinate is the same for both amino acids to make sure they are in the same row. What makes the problem harder than the case of $x_{+}^{ij,2D}$ is the fact that having picked i to be in an even spot, j is supposed to be in an odd spot and the addition of $00\cdots01$ to an odd binary number in general will not just change the last digit due to the carry bits. After using the circuit presented in Fig. 4, we obtained 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,

$$\begin{aligned} z_1 &= 0 \\ z_2 &= 1 - x_2 \\ z_i &= x_i + \prod_{u=2}^{i-1} x_u - 2\prod_{u=2}^{i} x_u \quad \text{for} \quad 3 \le i \le n \\ z_{n+1} &= \prod_{u=2}^{n} x_u \end{aligned}$$

This algebraic result for the z_i digit was constructed by exploiting the circuit for addition of two binary numbers shown in Fig. 4 and the Boolean algebra introduced in Sec. III A. This result is fundamental for the derivation of $x_{-}^{ij,2D}$ in Eq. 25. The ith digit of z is taken as the result of adding $00\cdots01$ to the x coordinate of the jth residue and compared to the corresponding coordinate of the ith amino acid through XNOR operations.

The remaining factor in Eq. 25, $[1-\prod_{k=1}^{\log_2 N}(1-q_{f(i,1)+k})]$, makes $x_{-}^{ij,2D}$ zero if the x coordinate of the ith amino acid is $00\cdots00$. We need to exclude this possibility since there is nothing to the left of $00\cdots00$. At the same time this also takes care of the possibility whenever the jth amino acid has an x coordinate equal to $11\cdots11$ that would match the coordinates $00\cdots00$ after the addition of 1. Therefore, the z_{n+1} case does not need to be considered because the expression would be different from zero only in the case of j having $11\cdots11$.

The construction of $y_{-}^{ij,3D}$ follows the same procedure as that of $x_{-}^{ij,3D}$, namely,

$$y_{-}^{ij,2D}(N) = (1-q_{f(i,2)+1})q_{f(j,2)+1}\Big[1-\prod_{k=1}^{\log_2 N}(1-q_{f(i,2)+k})\Big](q_{f(j,2)+2}+q_{f(i,2)+2}-2\,q_{f(j,2)+2}\,q_{f(i,2)+2})$$
$$\prod_{r=3}^{\log_2 N}\Big[1-(q_{f(j,2)+r}+\prod_{u=2}^{r-1}q_{f(j,2)+u}-2\prod_{u=2}^{r}q_{f(j,2)+u})$$
$$-q_{f(i,2)+r}+2q_{f(i,2)+r}(q_{f(j,2)+r}+\prod_{u=2}^{r-1}q_{f(j,2)+u}-2\prod_{u=2}^{r}q_{f(j,2)+u})\Big]$$
$$\prod_{s=1}^{\log_2 N}(1-q_{f(i,1)+s}-q_{f(j,1)+s}+2q_{f(i,1)+s}q_{f(j,1)+s}) \quad (26)$$

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 $H_{protein}$, as well as procedures required to reduce the locality of the terms to 2-local. These estimates provide an assessment of the size of a quantum device necessary for potential experimental realizations of the algorithm. The reduction of the locality of the terms involves ancillary qubits.

Since in our algorithm we are fixing two amino acids, the number of qubits needed to encode the coordinates of the $(N-2)$ remaining amino acids is $(N-2)D\log_2 N$. Each amino acid requires $D\log_2 N$ qubits. From the expressions 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. The onsite and pairwise terms involve comparison of the coordinates of two amino acids, and the expansion of all possible terms leads to a single term involving all possible q variables for the i-th and j-th amino acid. As described in Sec. III B 2, 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$(N-2)D\log_2 N$

term and 2-local $\binom{(N-2)D\log_2 N}{2}$ terms appear in the expansion, but not all possible 3-local or higher locality terms will be present. For example, the term $q_i q_j q_k$ where the indexes i, j and k are associated with three different amino acids would not be included in the expansion, since every term regardless of its locality should involve only products of qubits coming from two amino acids. Table I summarizes the number of k-local terms required for constructing the protein Hamiltonian, $H_{protein} = H_{onsite} + H_{psc} + H_{pairwise}$. Notice that Eq. 14 contains all terms in Table I (and therefore in $H_{protein}$) and the expansion of Eq. 14 gives a total number of terms that scales polynomially as N^{2D+2}. The alternative count from the combinatorial expressions of Table I scales exactly as a N^6 for $D = 2$ and as N^8 for $D = 3$ in agreement with Eq. 14. Counting directly from Eq. 14 is an upper bound to the number of terms since some of the terms in the expansion will be duplicated. Table I provides the exact term count.

IV. CASE STUDY: FOUR AMINO ACIDS IN TWO DIMENSIONS, HPPH.

For the purposes of designing an experiment for early quantum devices, we concentrate on the simplest possible instance of the HP-model: a four amino acid loop that contains a favorable interaction and therefore that "folds". This instance is probably not large enough to discern between a quantum adiabatic approach and a classical approach, but would be a landmark experiment for the development of early adiabatic quantum devices.

In Sec. IV A we present the protein Hamiltonian, we follow with the partitioning of the N-local Hamiltonian terms to 2-local. Finally, we present the numerical simulations to find the local minimum through the use of the algorithm proposed.

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

The onsite Hamiltonian for this particular example takes the form,

1. Onsite term, H_{onsite}

$$H_{onsite}(N=4,D=2) = \lambda_0(H^{12}_{onsite} + H^{13}_{onsite} + H^{14}_{onsite} + H^{24}_{onsite} + H^{34}_{onsite}) \quad (27)$$

with

$$H^{ij}_{onsite}(N=4,D=2) = \prod_{k=1}^{2}\prod_{r=1}^{2}\Big(1 - q_{f(i,k)+r} - q_{f(j,k)+r} + 2\,q_{f(i,k)+r}\,q_{f(j,k)+r}\Big) \quad (28)$$

and

$$f(i,k) = 4(i-1) + 2(k-1) \quad (29)$$

Note that H^{23}_{onsite} does not appear in Eq. 27 since, as described in Sec. II A, we fixed the two middle-most amino acids. This pair would not contribute any energy penalties in the onsite term *a priori* .

2. Primary structure constraint term, H_{psc}

The pairwise term,

$$d^2_{PQ}(N=4,D=2) = \sum_{k=1}^{2}\Big(\sum_{r=1}^{2} 2^{r-1}(q_{f(P,k)+r} - q_{f(Q,k)+r})\Big)^2 \quad (30)$$

with

$$\begin{aligned} H_{psc}(N=4,D=2) &= \lambda_1\big(-3 + d^2_{12} + d^2_{23} + d^2_{34}\big) \\ &= \lambda_1\big(-2 + d^2_{12} + d^2_{34}\big) \end{aligned} \quad (31)$$

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 which are in adjacent sites in the lattice.

$$H^{2D}_{pairwise}(N=4,D=2) = -(H^{14,2D}_{pairwise} + H^{41,2D}_{pairwise}) \quad (32)$$

For this particular case of interest

$$H^{ij,2D}_{pairwise}(N=4) = x^{ij,2D}_{+}(N=4) + x^{ij,2D}_{-}(N=4) + y^{ij,2D}_{+}(N=4) + y^{ij,2D}_{-}(N=4) \quad (33)$$

The explicit forms of these functions are:

$$x^{ij,2D}_{+}(N=4) = (1 - q_{f(i,1)+1})q_{f(j,1)+1}(1 - q_{f(j,1)+2} - q_{f(i,1)+2} + 2\,q_{f(j,1)+2}q_{f(i,1)+2}) \prod_{s=1}^{2}(1 - q_{f(i,2)+s} - q_{f(j,2)+s} + 2q_{f(i,2)+s}q_{f(j,2)+s}) \quad (34)$$

$$y^{ij,2D}_{+}(N=4) = (1 - q_{f(i,2)+1})q_{f(j,2)+1}(1 - q_{f(j,2)+2} - q_{f(i,2)+2} + 2\,q_{f(j,2)+2}\,q_{f(i,2)+s}) \prod_{s=1}^{2}(1 - q_{f(i,1)+s} - q_{f(j,1)+s} + 2q_{f(i,1)+s}q_{f(j,1)+s}) \quad (35)$$

$$x_{-}^{ij,2D}(N=4) = (1-q_{f(i,1)+1})q_{f(j,1)+1}q_{f(i,1)+2}$$
$$(q_{f(j,1)+2}+q_{f(i,1)+2}-2q_{f(j,1)+2}q_{f(i,1)+2})\prod_{s=1}^{2}(1-$$
$$q_{f(i,2)+s}-q_{f(j,2)+s}+2\,q_{f(i,2)+s}q_{f(j,2)+s}) \quad (36)$$

$$y_{-}^{ij,2D}(N=4) = (1-q_{f(i,2)+1})q_{f(j,2)+1}q_{f(i,2)+2}$$
$$(q_{f(j,2)+2}+q_{f(i,2)+2}-2\,q_{f(j,2)+2}q_{f(i,2)+2})\prod_{s=1}^{2}(1-$$
$$q_{f(i,1)+s}-q_{f(j,1)+s}+2\,q_{f(i,1)+s}q_{f(j,1)+s}) \quad (37)$$

After expanding all the terms for H_{onsite}, H_{psc} and $H_{pairwise}$, the final version of $H_{protein}$ depends on the 8 binary variables encoding the coordinates of amino acids 1 and 4. The spin $\frac{1}{2}$ representation can be obtained after substituting the q_i for operators $\hat{q}_i = \frac{1}{2}(1-\sigma_z)$, and the final result is a 2^8 x 2^8 Hamiltonian matrix. 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

$$H_i = H(t=0) = \sum_{i=1}^{8}\hat{q}_x = \sum_{i=1}^{8}\frac{1}{2}(1-\sigma_x) \quad (38)$$

with

$$|\psi_g(t=0)\rangle = \frac{1}{\sqrt{2^8}}\sum_{q_i\in\{0,1\}}|q_{16}q_{15}q_{14}q_{13}q_4q_3q_2q_1\rangle \quad (39)$$

Finally, we can construct a time dependent Hamiltonian given by Eq. 6,

$$H(t) = (1-t/\tau)H_i + (t/\tau)H_{protein} \quad (40)$$

This time dependent Hamiltonian is also a 2^8 x 2^8 matrix and 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 $0 \le t/\tau \le 1$. The spectrum of the corresponding $H(t)$ for this four amino acid peptide HPPH is given in Fig. 5

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 picked. 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, in this section we explain how to reduce a k-local Hamiltonian to a 2-local Hamiltonian, while conserving the low-lying spectrum of the Hamiltonian. We will use one of the gadgets developed by Biamonte [48]. 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. Let's assume a simpler Hamiltonian of the form,

$$H = 1 - q_1q_2q_3 \quad (41)$$

. This Hamiltonian has a non-degenerate ground state given by $q = q_3q_2q_1 = 111$. The energy associated with this configuration is 0 in arbitrary units. The configurations for all other possible values of the binary variables q_1, q_2 and q_3 have an energy equal to 1 in arbitrary units. Our goal is to obtain an energy function H such that it still preserves these energies along with their associated bit strings, but we want H to have terms involving just 1-local and 2-local terms. That is,

$$H(q_1,q_2,q_3) = 1 - q_1q_2q_3$$
$$H' = c_0 + \sum_{i=1}^{M}c_i\tilde{q}_i + \sum_{i=1}^{M-1}\sum_{j=i+1}^{M}d_{ij}\tilde{q}_i\tilde{q}_j \quad (42)$$

In Eq. 42 the new binary variables $\tilde{q}$ include the original variables q_i and the additional ancillary variables required for achieving the reduction. The total number of variables M is larger than the original set of variables due to the required ancilla qubits. For example, consider the substitution $a_1 = q_1q_2$. Since the new form of H becomes $H = 1 - a_1q_3$, this Hamiltonian will not provide enough information to recover the values of q_1 and q_2 due to the irreversibility of the AND function. For example, if $a_1 = 0$, then we do not know which of the three bit strings $q_1q_2 = 01$, $q_1q_2 = 10$ and $q_1q_2 = 00$ generated $a_1 = 0$. Therefore, one cannot simply reduce the number of variables in the Hamiltonian without losing information. We need to use a_1 as an extra ancilla bit and try to add a function which is still 2-local and involves q_1, q_2, q_3 and a_1. As a consequence we have now a larger Hilbert space spanned by configurations of the form $|a_1q_3q_2q_1\rangle$. The added function needs to penalize cases where the truth table for the AND function is violated, i.e., it should penalize cases where $a_1 \neq q_1q_2$. According to Biamonte [48], the following term can be added to our reduced Hamiltonian,

$$H_\wedge(q_1,q_2,a_1) = \delta(3a_1 + q_1q_2 - 2q_1a_1 - 2q_2a_1) \quad (43)$$

and our original Hamiltonian in Eq. 41 assumes the reduced form,

$$H'(q_1,q_2,q_3,a_1) = 1 - a_1q_3 + H_\wedge(q_1,q_2,a_1)$$
$$= 1 - a_1q_3 + \delta(3a_1 + q_1q_2 - 2q_1a_1 - 2q_2a_1) \quad (44)$$

To summarize the reduction procedure, Table II lists the truth table for both Hamiltonians, the original and the 2-local Hamiltonian H'. Notice that by setting δ above the energy values of all plausible conformations of the

protein, the two Hamiltonians H and H' will have the same truth table for the first eight rows of Table II. Any other configurations will be penalized. The first eight rows correspond to the configurations where the relation $a_1 = q_1 \wedge q_2$ is satisfied.

In the case of a k-local Hamiltonian term, e.g., $h = q_1 q_2 \cdots q_k$, the reduction is carried out iteratively, adding the penalty function $H_\wedge(q_i, q_j, a_n)$ for every substitution of the form $a_n = q_i q_j$. For a k-local term, $(k-2)$ substitutions are required for the reduction to 2-local, and therefore require $(k-2)$ ancilla bits.

For the particular case of the protein Hamiltonian the reduction procedure needs to be repeated $(N-2)(N^D - D\log_2 N - 1)$ times . All the terms in the HP Hamiltonian include at most interaction between 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 variables coming from amino acid i and l variables from amino acid j. Since all the terms are of this form, to obtain a 2-local Hamiltonian, all the products corresponding to each cluster have to be converted to 1-local terms. We concentrate in reducing terms for the variables describing each amino acid, for a total of $D\log_2 N$ variables. All possible combination of two variables from the $D\log_2 N$ variables for an amino acid are substituted. The number of ancilla 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] 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. By employing an additional ancilla bit per term and using the proposed reduction method, all these terms collapse to 1-local with respect to the i aminoacid, i.e., these terms will assume the form $[1][l]$. Repeating the procedure for terms with higher locality until we exhaust the $D\log_2 N$ variables for a specific amino acid will give us the number of substitutions or ancilla bits needed per amino acid to reduce products a particular cluster $[k]$ to [1] or 1-local. The total number of substitions per aminoacid corresponds to corresponds to $\sum_{k=2}^{D\log_2 N}\binom{D\log_2 N}{k}$. The proedure needs to be carried out for all $(N-2)$ amino acids the number of ancilla qubits required is therefore $(N-2)\sum_{k=2}^{D\log_2 N}\binom{D\log_2 N}{k}$. Adding the ancillary qubits to the original $(N-2)D\log_2 N$ quantum bits, then number of qubits needed to represent a 2-local Hamiltonian version of the protein Hamiltonian is given by,

$$T_2 = (N-2)\sum_{k=2}^{D\log_2 N}\binom{D\log_2 N}{k} + (N-2)D\log_2 N = (N-2)(N^D - 1) \quad (45)$$

Eq. 45 provides the number of qubits needed to find the lowest-energy conformations for a protein with N amino acids in D dimensions. In particular, for the case of the example 4 amino acid peptide HPPH in 2-dimensions consider in Sec. IV, 30 qubits will be required. For a 32 amino acid protein in 3-dimensions corresponding to the limits for an exhaustive search in a modern classical computers, the number of qubits required for carrying out this experiment is 983010 qubits.

The resource counting of this section is very relevant to experimental implementations. On one hand, a 30-qubit system with arbitrary 2-local couplings could be employed to fold the smallest instance of the HP model. On the other hand, about a million quantum bits would be required to compete with current classical computers.

In Sec. VI, we present an illustrative example of the reduction of a 4-local Hamiltonian to 2-local and the direct comparison of both energy spectra to show the conservation of the low-lying energy states of the original k-local Hamiltonian.

VI. LOCALITY-REDUCED HAMILTONIAN EXAMPLE

To examine the validity of the reduction to 2-local, we constructed a toy 4-local Hamiltonian.

$$H_{toy} = 1 - q_1 q_2 q_3 q_4 \quad (46)$$

Using the technique described in Sec. V, the corresponding 2-local Hamiltonian is

$$H_{toy,reduced} = 1 - a_1 a_2 + \delta(3a_1 + q_1 q_2 - 2a_1 q_1 - 2a_1 q_2) + \delta\,(3a_2 + q_3 q_4 - 2a_2 q_3 - 2a_2 q_4) \quad (47)$$

where the reduction added two ancillary bits a_1 and a_2. Both of these Hamiltonians have the same minimum value of $q_i = 1$. This is accomplished by plotting the energy levels of Eq. 6 vs. t/τ, where $H_{protein}$ is replaced either by H_{toy} or $H_{toy,reduced}$ (see Fig. 7). From the spectra of Fig. 7, it can be seen that the ground state is preserved in the transformation. It is clear that the ground state is conserved but unfortunately due to degeneracy issues and overlap of lines in the spectra it is hard to visualize in the figure that indeed both spectra have the same number of states in the first excited state of H_{toy} as well. There is a one-to-one correspondence between the states of the 4-local toy Hamiltonian and the ones of the reduced one. The states with energy greater than 1 in the reduced Hamiltonian spectra correspond to states which violate the AND condition introduced by the reduction (see Sec. V).

VII. CONCLUSIONS

We constructed the essential elements of an adiabatic quantum algorithm for finding the low energy conformations of a protein in a lattice model. The number of resources (qubits) needed to map N amino acids to an

$N \times N$ lattice is $(N-2)D\log_2 N$ and the maximum locality of the final Hamiltonian is determined by the interaction between pairs of amino acids is $2D\log_2 N$.

General strategies to construct energy functions to map into further 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 of Hamiltonians associated physical systems where onsite energies and/or pairwise potentials are present.

We also demonstrated an application of the Biamonte scheme for converting the proposed $2D\log_2 N$ local Hamiltonian into a 2-local Hamiltonian, aiming toward an experimental implementation in quantum devices. The resulting couplings although 2-local are not necessarily couplings amongst nearest neighbor quantum bits in a two-dimensional geometry. The experimental implementation of these couplings might prove challenging, or require the introduction of a large number of additional ancilla quantum bits.

The scaling of the qubit resources required for this problem is polynomial, nevertheless a million quantum bits with two-local couplings would be required to compete with a modern classical computer. The advantage of adiabatic quantum computers is that no active ultrafast quantum control to perform quantum gates is required, but in this case, the downside is the large number of qubits required to compete with a classical computer. It is instructive to compare this results to the case of quantum chemical molecular energies using the gate model [20]. In this case, the threshold where a quantum computer would outperfom the best modern classical computers is about 100 quantum bits and millions of gates or elementary quantum operations.

As previously mentioned, the construction of the final Hamiltonian encoding the solution to the problem is an essential piece for the evolution of the system according to the time-dependent Schrödinger equation. The most important question remaining to be explored in future work is the scaling of the running time τ with respect to the number of amino acids N. The running time τ is dependent on the particular instance of the problem. In our case, the instances correspond to different protein sequences. It has been proposed that proteins have evolved towards a many-dimensional funnel-like potential energy surface [15]. The sequences that show a funnel-like structure might be easier to study using adiabatic quantum computation. The funnel structure might facilitate the annealing of the quantum wave function toward low-energy conformations. The efficiency of the algorithm for protein sequences of biological importance of more realistic models might be better than that for the average instance of the problem if this hypothesis is correct.

Further studies need to be done to determine the running time, τ. Its scaling with the number of amino acids will eventually determine the efficiency of the algorithm. Our hope is to find a speed-up for proteins or instances that are intractable with classical computers. Recently, Schutzhold *et al.* [44] proposed that adiabatic quantum algorithms that are constructed to present second-order phase transitions could be more efficient, and in particular showed how a second-order adiabatic algorithm was more efficient numerically for random instances of the 3-satisfability problem. Detailed numerical simulations to explore the potential speedup of the algorithm over classical methods will be reported in a future publication.

Acknowledgments

We thank Jacob Biamonte and Ivan Kassal for helpful discussions. Partial funding for this project was provided by a D-Wave Systems research contract.

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

This extension is 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,

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

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

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

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

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

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

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

[1] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, *Science* **220**, 671 (1983), 10.1126/science.220.4598.671.

[2] A. Das and B. Chakrabarti, editors, *Quantum annealing and other optimization methods*, Springer-Verlag, New York, 2005.

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

[4] G. Santoro, R. Martonak, E. Tosatti, and R. Car,

Science **295**, 2427 (2002), 10.1126/science.1068774.

[5] J. Brooke, D. Bitko, T. Rosenbaum, and G. Aeppli, *Science* **284**, 779 (1999), 10.1126/science.284.5415.779.

[6] P. Sun, Y. Yin, B. Li, T. Chen, Q. Jin, D. Ding, and A.-C. Shi, *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)* **72** (2005), 10.1103/PhysRevE.72.061408.

[7] R. Goldstein, Z. Luthey-Schulten, and P. Wolynes, *Proceedings of the National Academy of Sciences* **89** (1992), 10.1073/pnas.89.11.4918.

[8] D. Ceperley and B. Alder, *Science* **231** (1986), 10.1126/science.231.4738.555.

[9] H. S. Chan and K. A. Dill, *Physics Today* **46**, p24 (19930201).

[10] S. S. Plotkin and J. N. Onuchic, *Quarterly Reviews of Biophysics* **35** (2002), 10.1017/s0033583502003761.

[11] E. Shaknovich, *Chemical Reviews* **106** (2006), 10.1021/cr040425u.

[12] L. Mirny and E. Shaknovich, *Annual Review of Biophysics and Biomolecular Structure* **30** (2001), 10.1146/annurev.biophys.30.1.361.

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

[14] M. Gruebele, *Annual Review of Physical Chemistry* **50**, 485 (1999), 10.1146/annurev.physchem.50.1.485.

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

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

[17] D. Baker and D. A. Agard, *Biochemistry* **33**, 7505 (1994), 10.1021/bi00190a002.

[18] T. Lazaridisa and M. Karplus, *Biophysical Chemistry* **100**, 367 (2003), 10.1016/s0301-4622(02)00293-4.

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

[20] A. Aspuru-Guzik, A. Dutoi, P. Love, and M. Head-Gordon, *Science* **309**, 1704 (2005), 10.1126/science.1113479.

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

[22] K. Steinhfel, A. Skaliotisa, and A. A. Albrecht, *Computer Physics Communications* **176**, 465 (2007), 10.1016/j.cpc.2006.12.002.

[23] G. A. Cox, T. V. Mortimer-Jones, R. P. Taylor, and R. L. Johnston, *Theor. Chem. Acc.* **112**, 163 (2004), 10.1007/s00214-004-0601-4.

[24] G. A. Cox and R. L. Johnston, *J. Chem. Phys.* **124**, 204714 (2006), 10.1063/1.2198537.

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

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

[27] 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), 10.1109/PDCAT.2005.31.

[28] A. Shmygelska and H. H. Hoos, *BMC Bioinformatics* **6** (2005), 10.1186/1471-2105-6-30.

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

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

[31] H. P. Hsu, V. Mehra, W. Nadler, and P. Grassberger, *Physical Review E* **68**, 021113 (2003), 10.1103/PhysRevE.68.021113.

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

[33] 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).

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

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

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

[37] 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), 10.1103/physrevlett.98.177001.

[38] A. Niskanen, K. Harrabi, F. Yoshihara, Y. Nakamura, S. Lloyd, and J. Tsai, *Science* **316**, 723 (2007), 10.1126/science.1141324.

[39] J. Plantenberg, P. de Groot, C. Harmans, and J. Mooij, *Nature* **447**, 836 (2007), 10.1038/nature05896.

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

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

[42] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, http://arxiv.org/abs/quant-ph/0001106.

[43] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *Science* **292**, 472 (2001), 10.1126/science.1057726.

[44] R. Schutzhold and G. Schaller, *Phys. Rev. A.* **74**, 060304 (2006).

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

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

[47] H. Li, C. Tang, and N. Wingreen, *Proteins: Structure, Function, and Genetics* **49**, 403 (2002), 10.1002/prot.10239.

[48] J. D. Biamonte, Unpublished.

TABLE I: Number of k-local terms that are 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
$k = 1$	$(N-2)D\log_2 N$
$2 \leq k \leq D\log_2 N$	$\binom{N-2}{2}\sum_{i=1}^{k-1}\binom{D\log_2 N}{i}\binom{D\log_2 N}{k-i} + (N-2)\binom{D\log_2 N}{k}$
$D\log_2 N < k \leq D\log_2 N$	$\binom{N-2}{2}\sum_{i=k-D\log_2 N}^{D\log_2 N}\binom{D\log_2 N}{i}\binom{D\log_2 N}{k-i}$

TABLE II: Comparison between the problem Hamiltonian, $H(q_1, q_2, q_3)$, and the locality-reduced Hamiltonian, $H'(q_1, q_2, q_3, a_1)$, defined in Eqs. 41 and 44 respectively. Values for H' and H are given in arbitrary units. The value of δ is an energy penalty applied to the cases where $a_1 \neq q_1 \wedge q_2$

q_3	q_2	q_1	a_1	$H'(q_1, q_2, q_3, a_1)$	$H(q_1, q_2, q_3)$
0	0	0	0	1	1
0	0	1	0	1	1
0	1	0	0	1	1
0	1	1	1	1	1
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	0	1	1
1	1	1	1	0	0
0	0	0	1	$1 + 3\delta$	1
0	0	1	1	$1 + \delta$	1
0	1	0	1	$1 + \delta$	1
0	1	1	0	$1 + \delta$	1
1	0	0	1	3δ	1
1	0	1	1	δ	1
1	1	0	1	δ	1
1	1	1	0	$1 + \delta$	0

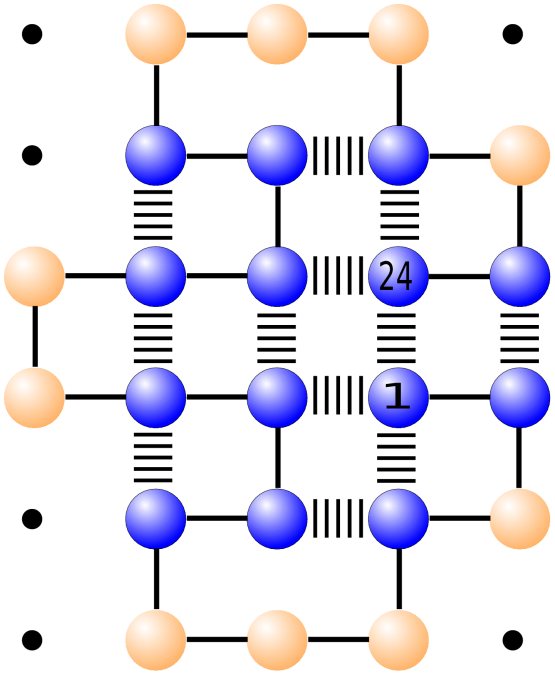

FIG. 1: The lattice protein hydrophobic-polar (HP) model, showing the global energy minimum conformation for a sequence of 24 amino acids, HHPHPPPHHHHPPH-HHHPPPHPHH ($E = -12$). The blue beads represent hydrophobic residues (H) and the orange beads represent polar residues (P). The model consists of a self-avoiding chain with favorable ($E = -1$) energetic interactions amongst hydrophobic residues in contact. The nearest-neighbor contacts due to the primary sequence do not contribute to the energy. Black dots represent lattice sites. Favorable energetic interactions are represented by the dotted lines. Solid lines represent the self-avoiding chain.

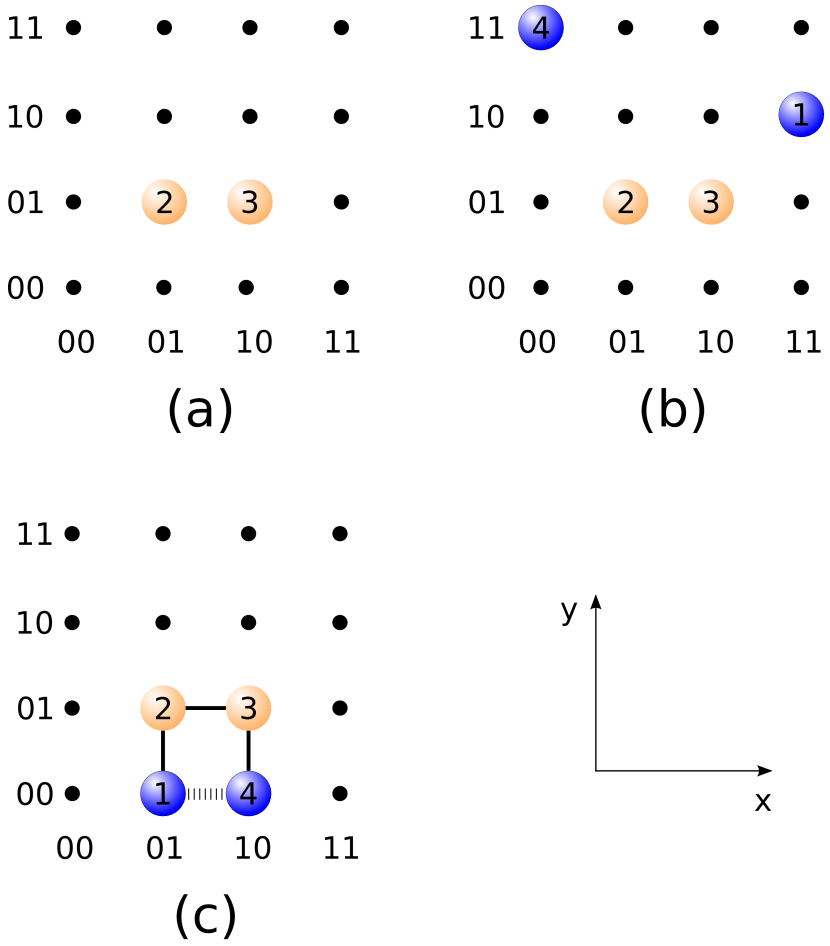

FIG. 2: Grid-labeling conventions for a sequence of 4 amino acids, HPPH. **(a)** Amino acids 2 and 3 are fixed in the center of the grid for eliminating translational degeneracy. **(b)** One of the possible invalid configurations that might arise in the search and that would need to be discarded by the optimization algorithm. **(c)** Lowest-energy conformation for this example. The dotted line between amino acids 1 and 4 represents the favored hydrophobic interaction of 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 of (b), $q =$ 1011 0110 0101 1100.

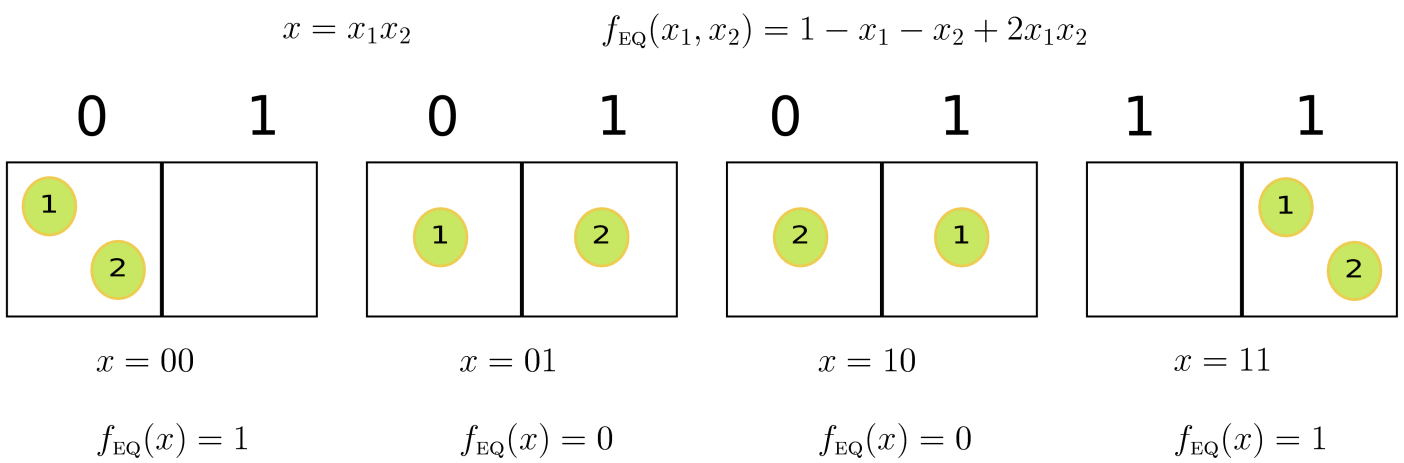

FIG. 3: Illustrative example of one of the uses of the XNOR Boolean function in our scheme for the construction of Hamiltonians. Consider 1 and 2 being two particles that are restricted to occupy either position 0 or 1, and let x_1 and x_2 encode the position of particle 1 and particle 2 respectively. 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 $x = x_1x_2$.

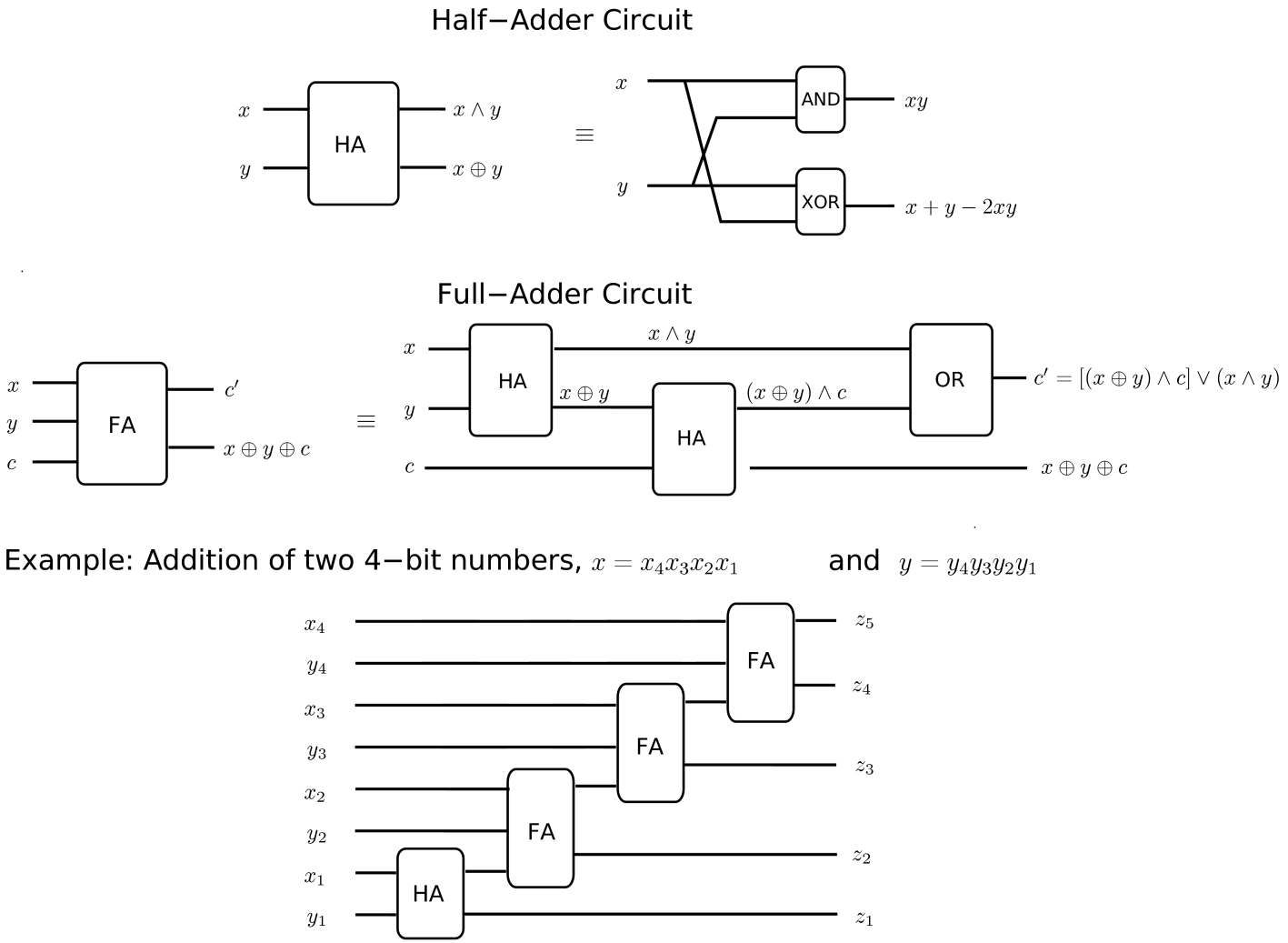

FIG. 4: Half-adder and full-adder Components for the addition circuit implemented in the pairwise interaction Hamiltonian. We show the implementation of these two components for the addition of two 4-bit numbers yielding $z = z_5z_4z_3z_2z_1$. The addition of n-bit numbers can be generalized trivially.

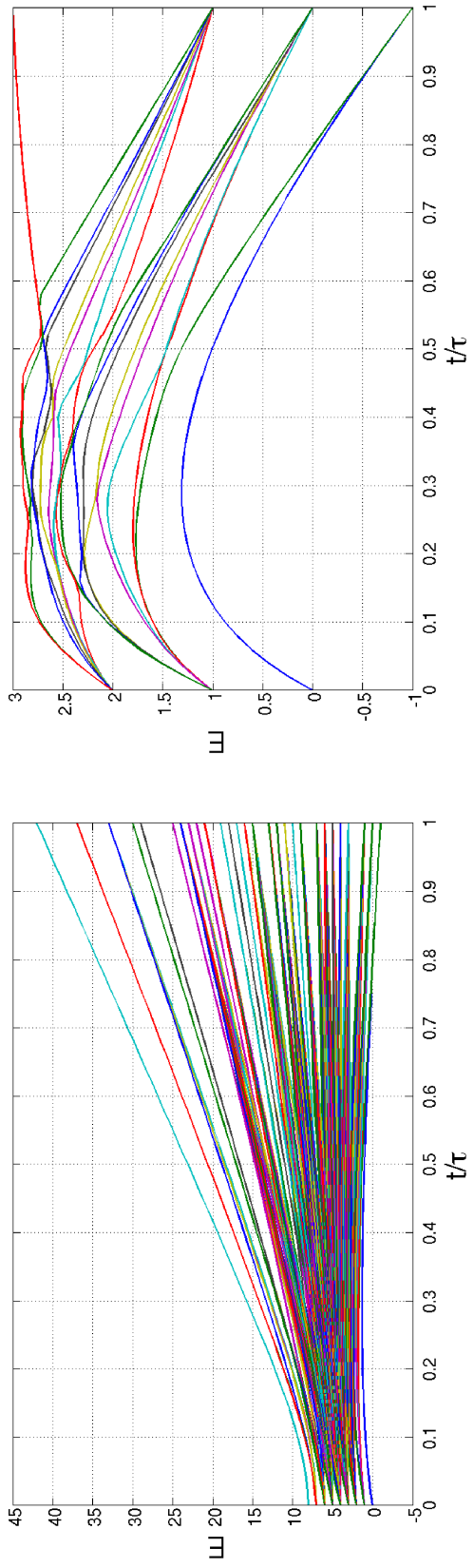

FIG. 5: Spectrum of the instantaneous energy eigenvalues for the 8-local time dependent Hamiltonian used in the algorithm for the peptide HPPH (left). The plot to the right examines the lowest 15 states of the 256 states from the left.

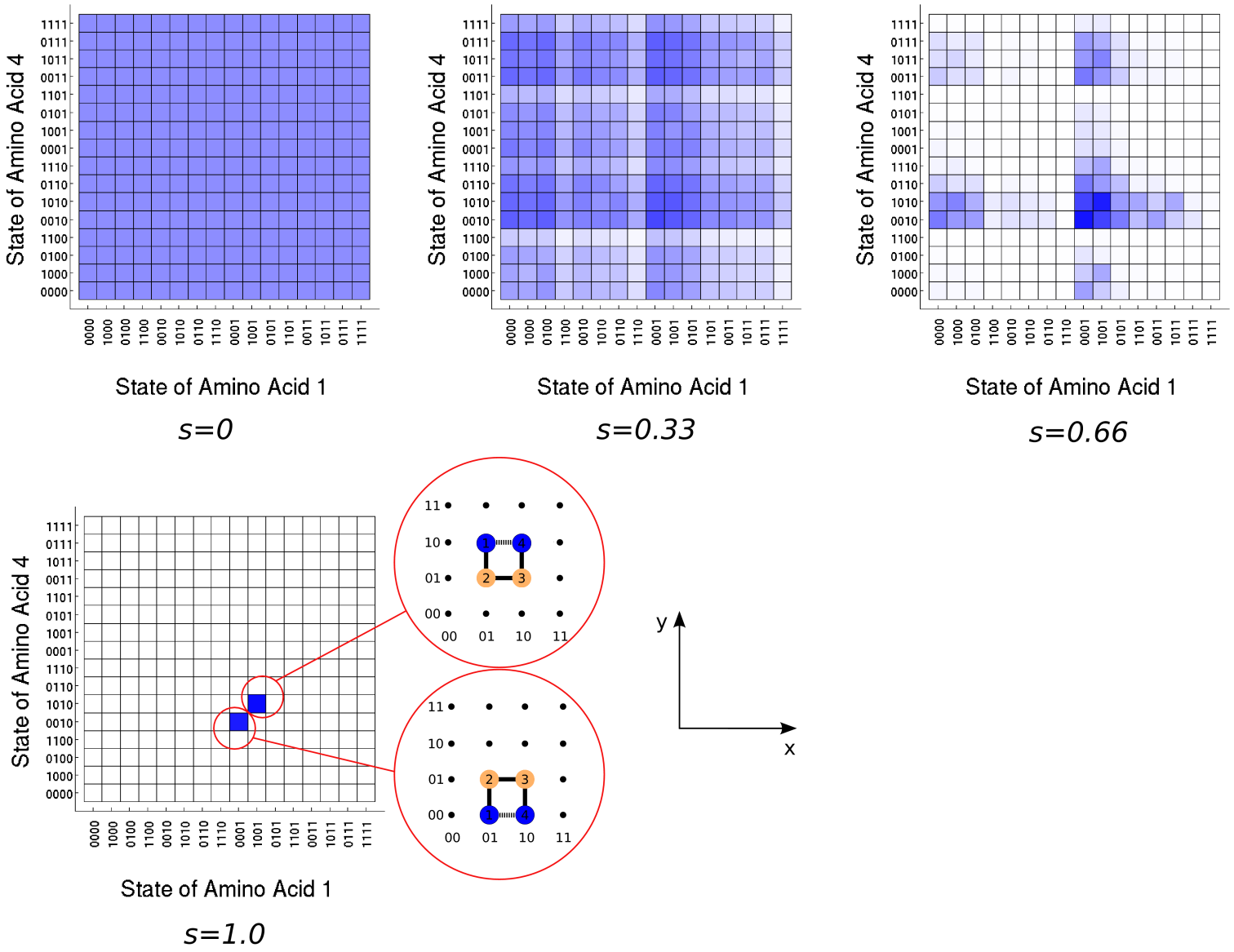

FIG. 6: Snapshots of the instantaneous ground state for $H(t)$. The brightness of the box is related to $|c_n|^2$. The labels of the axes and the way to read out the state vector for a particular box are according to $|\psi\rangle = \sum_{n=0}^{255} c_n |n\rangle$ with $|n\rangle$ the n-th state vector out of the 256 possibilities giving 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 giving by $|q_4\rangle |q_3\rangle |q_2\rangle |q_1\rangle$ and the y axis giving by $|q_{16}\rangle |q_{15}\rangle |q_{14}\rangle |q_{13}\rangle$. The final state correspond to the two degenerate minima shown at the end

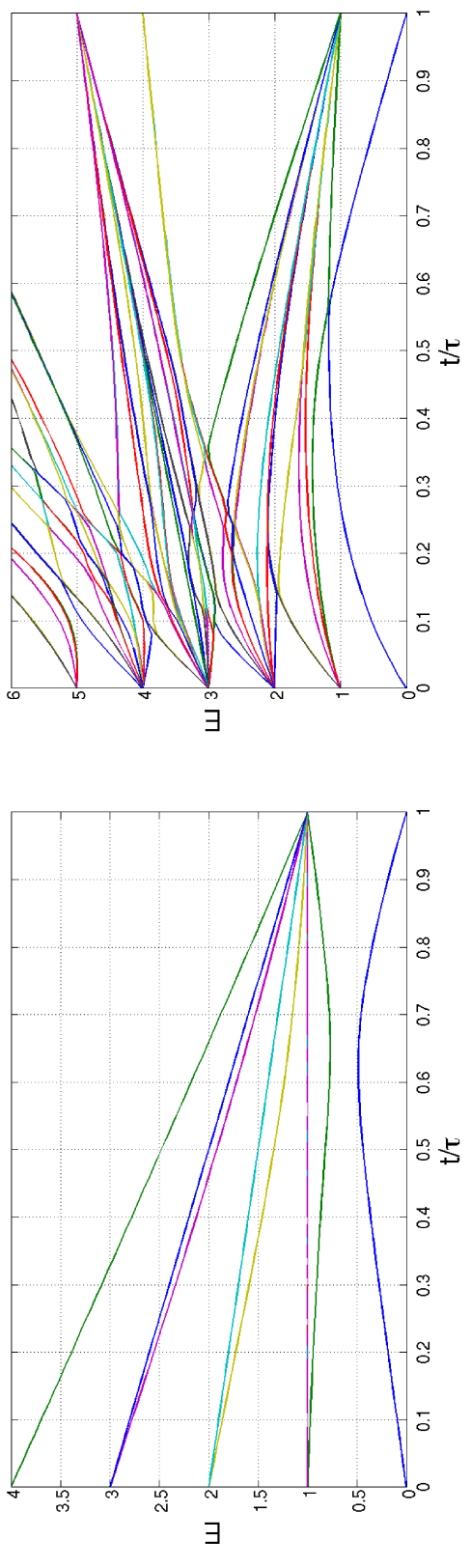

FIG. 7: Spectrum comparison of the instantaneous energy eigenvalues for the 4-local toy Hamiltonian (left) and 2-local toy Hamiltonian (right). Notice that the ground state of the 4-local toy Hamiltonian is conserved after the reduction