Using the Energy Distribution as a Benchmarking Among Models for Adiabatic Quantum Processing in the Protein Folding Problem

A Anaya and F Delgado
Tecnologico de Monterrey, School of Engineering and Sciences, México.
E-mail: fdelgado@tec.mx

Abstract. Determining the tertiary complexity of proteins using ab-initio algorithms is a hard problem. Adiabatic Quantum Computing allows to construct simple Ising-like Hamiltonians in order to elucidate the aminoacid interactions minimizing the conformation energy of the protein. Exploring the energy distribution of the Hamiltonian allows to compare the efficiency of different models there proposed. This article compare the efficiency of two algorithms through the conformation energy of a protein stating a benchmark through the energy distributions of the conformation states.

1. Introduction
Proteins are polymers of alternating sequences of 20 possible aminoacids encoded in the DNA. They carry out most of the enzymatic activity, performing metabolic regulation and structural stabilization of cells. Cyrus Levinthal was one of the first who study the theoretical implications of protein folding (PF) [1], that is, the mechanism by which protein folds to its native structure. According with his studies, a protein having 100 aminoacids will have $10^{48}$ possible conformations. If interconversion between two different conformations takes around $10^{-11}$ seconds, then it would take $10^{29}$ years to explore all of them. Therefore, proteins have evolved to optimize the timescales of folding [2].

Computational models have been developed in order to understand this problem [3]. Classical lattice algorithms have proved to find the minima of the energy corresponding to the native structure of a protein, however, the computational complexity and time scales make it unfeasible for large molecules. Adiabatic quantum computing (AQC) is a quantum computation approach, based on the quantum adiabatic theorem [4], where a system initially prepared on a non-degenerate ground state (for a Hamiltonian $H_0$) will remain on the ground state, up to a global phase, while the Hamiltonian evolves properly slow into $H_p$ (which codifies the solution of a problem) as $H(t) = (1-t)H_0 + tH_p$.

In this paper we discuss two algorithms found in literature to construct such Hamiltonians by 1) mapping the coordinates of the aminoacids on qubits [5] and 2) mapping the turns of the aminoacid sequence [6, 7]. We provide a comprehensive notation of the Hamiltonians $H_p$, then a discussion of the energy distribution under interaction potentials like the Hydrophobic-Polar (HP) and the Miyazawa-Jeringan (MJ) allows us to compare the efficiencies of both models [3,8]. We do not take care of $H_0$ here because this aspect is irrelevant for the discussion of efficiency.
2. Alternative adiabatic quantum algorithms for protein folding

We compare two models proposed in the literature for protein folding under AQC: a) site codifying approach (SC) [5], and b) turns codifying approach (TC) [6, 7]. In the following sections we briefly describe the details of each one without extensive details, which can be reviewed directly on their sources [5,9]. Despite that, in this paper we provide a comprehensive and compact structure for such models.

2.1. Codifying the aminoacid position in a cubic lattice on qubits

SC model encodes in qubits the discrete positions the N aminoacids in the protein on a $D = 2, 3$ dimensional lattice of $2^m \geq N$ nodes in each spatial direction. There, $m$ is assumed independent of $N$ in general: $(Q^0_j \in \{0, \ldots, 2^m - 1\}$, as a string of $m$ binary digits $q_{sm} \ldots q_{sn}$) for each spatial direction. Then, if the aminoacid $r$ lies in the $k$ direction on the position $Q^0_j$, provided that the $r$th binary digit is $q_r$, the state of $n = NDm$ qubits codifying the protein’s arrangement is:

$$|Q^{10}_S\rangle \equiv |q_{1_1}q_{m_1}q_{m+1}q_{m+1}q_{m+1}q_{m+2} \ldots q_{m+n-1}q_{n}| = \bigotimes_{j=1}^{N} \bigotimes_{k=1}^{D} \bigotimes_{r=1}^{m} |q_{f_{j,k,r}}\rangle \equiv \bigotimes_{j=1}^{N} |Q^{10}_j\rangle,$$

which is illustrated for $D = 3$ in the Figure 1a. This model combines notations in base-2 or base-10 for simplicity, so that $|Q^{10}_S\rangle, Q^{10}_S \in \{0, \ldots, 2^m - 1\}$ is a short base-10 notation from the corresponding binary representation. In addition, $f_{j,k,r} = (j - 1)Dm + (k - 1)m + r$ reports each qubit position on the last state structure. Taking $N = 2^m$ in the following and restricting the two central aminoacids translationally and rotationally (TRR criteria) on the lattice center $(\frac{N}{2}, \frac{N}{2}, \frac{N}{2}), (\frac{N}{2} + 1, \frac{N}{2}, \frac{N}{2})$ as in [5], thus reducing the valid states by considering $|Q^0_S\rangle = |110...11011011010...10\rangle$ and $|Q^0_S^{10}\rangle = |000...001110111011010...10\rangle$ fix (for $D = 3$, as instance). Taking $\hat{1}_s, \hat{X}_s, \hat{Y}_s, \hat{Z}_s$ as the extended Pauli operators for the qubit $s$, we define the position operators for each qubit in terms of $\hat{q}^{X}_s = \frac{1}{2}(\hat{1}_s - \alpha_s), \alpha \in \{X, Y, Z\}$ (particularly, only the use of $\hat{q}^{X}_s$ is necessary). Because we work always on the basis of $Z_s$, we can express the complete SC model of PF for AQC in terms of their eigenvalues $q^a_s$ [5], thus leaving out the use of operators:

$$H_{onsite} = \lambda_0 \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sum_{k=1}^{D} \sum_{r=1}^{m} \text{XNOR}(q^{Z}_{f_{i,k,r}}, q^{Z}_{f_{j,k,r}}),$$

$$H_{psc} = \lambda_1 (-N - 1) + \sum_{m=1}^{N-1} \sum_{k=1}^{D} \sum_{r=1}^{m} 2^{r-1} (q^{Z}_{f_{m,k,r}} - q^{Z}_{f_{m+1,k,r}})^2,$$

$$H_{pairwise} = - \sum_{i,j=1}^{N} G_{ij} H_{ij:D}^{pairwise} = - \sum_{i,j=1}^{N} G_{ij} \sum_{\sigma=\pm 1}^{D} C_{\sigma:i,j:D},$$

$$C_{\sigma:i,j:D} = (1 - q^{Z}_{f_{i,k,1}})q^{Z}_{f_{j,1,1}}(1 - \prod_{l=1}^{m-1}(1 - q^{Z}_{f_{i,l,1}}))^{\frac{1}{2}(1-\sigma)} \prod_{c \in C} \prod_{r=2}^{m} \text{XNOR}(q^{Z}_{f_{i,k,1},s}, q^{Z}_{f_{j,1,1},s}),$$

HP model [5] states $H_0 = H_{onsite} + H_{psc} + H_{pairwise}$. Positive Hamiltonian terms impose a penalty in the overall model while those negative impose an award. There, $\text{XNOR}(q^Z_s, q^Z_{s'}) = 1 - q^Z_s - q^Z_{s'} + 2q^Z_s \otimes q^Z_{s'}$ is the binary XNOR function, being equal to one only if all qubits are
equal between any pair of aminoacids, thus imposing a penalty. Thus, \( H_{\text{onsite}} \) is responsible of the single node occupation for each aminoacid by avoiding the same value on each corresponding qubit between any pair of aminoacids. \( H_{\text{pac}} \) is responsible for the consecutive unitary distance in the pair sequence. There, the sums represent the addition of the quadratic distances between the overall pairs of consecutive aminoacids in the protein balanced to zero by the term \(-(N-1)\) only if such distances are one (note that \( H_{\text{onsite}} \) prevents such quadratic distances become zero). \( H_{\text{pairwise}} \) is responsible of the hydrophobic H-H interactions boosting unitary distances among such aminoacids. Here, as illustration purposes, only the two central aminoacids are \( P \). \( G_{ij} \) is a symmetric matrix settling the pairs with H-H interactions. Then: 
\[
G_{ij} = \theta_{ij}(1 - \delta_{ij}) \prod_{k=1,j}(1 - \delta_{k \frac{x}{2}})(1 - \delta_{k \frac{x}{2} + 1}) \quad (\theta_{ij} = 1 \text{ has been selected in this model}).
\]
If \( k, l, m \) (\( k, l \)) is a permutation of \( 1, 2, 3 \) (\( 1, 2 \)), for \( D = 3 \) (\( 2 \)), then \( C = \{ l, m \} \) (\( \{ l \} \)). \( \delta \gamma(\{ x_s \}) \) is the function for the \( v \)th base-2 digit in \( z = x + \frac{1}{2}(1 - \sigma) \) where \( x = x_m x_{m-1} ... x_1 \) \([5]\), in order to both positions become consecutive with a unitary distance in the two binary alternatives. If \( N = \lambda_1 < \lambda_0 = N + 1 \), \( H_{\text{onsite}} + H_{\text{pac}} \) is a positive penalty and zero only for valid PF solutions, while \( H_{\text{pairwise}} \) is negative in such cases, otherwise is positive or zero, with a minimum of \(-N\) for \( H \) chains \([5]\).

2.2. Codifying the aminoacid turns in a cubic lattice on qubits

In the TC model we codify the turns on the protein configuration in agreement with Figure 1b, leaving codes 000 and 011 without use. With \( N \) aminoacids numbered from 0 to \( N - 1 \) confronted by turns numbered from 0 to \( N - 2 \), the qubits chain codifying the turn sequence has the following general form, with a fixed initial rotational degree of freedom:
\[
|Q_{T}^{0}\rangle \equiv |101\rangle \otimes |q_{1}q_{2}q_{3}\rangle \otimes ... \otimes |q_{3N-11}q_{3N-10}q_{3N-9}\rangle \otimes |q_{3N-8}...q_{3N-9+N_{\text{ancilla}}}\rangle \otimes ...|q_{N_{\text{ancilla}}}\rangle \otimes |q_{N+N_{\text{ancilla}}}\rangle \equiv |Q_{T}^{0}\rangle \otimes |Q_{T}^{0}\rangle_{\text{ancilla}},
\]
where the fourth qubit \( q_{0} \) states the direction of the second turn and \( q_{1}, ..., q_{3N-9} \) specifies the remaining turns three by three. The last \( N_{\text{ancilla}} \) qubits are used to codify one of the terms in the Hamiltonian conformed by fourth terms: \( H_{p} = H_{\text{back}} + H_{\text{redun}} + H_{\text{olap}} + H_{\text{pair}} \), with:
\[
H_{\text{back}} = \lambda_{\text{back}}(q_{0}d_{x}^{2} + (1 - q_{0})d_{y}^{2} + \sum_{j=2}^{N-3} \sum_{k=1}^{3} \sum_{\alpha \in \{ \pm 1 \}} \sum_{s=0}^{1} d_{\alpha(-1)^{s}k}^{j+s}),
\]
\[
H_{\text{redun}} = \lambda_{\text{redun}}(d_{000}^{j} + d_{011}^{j}),
\]
\[
H_{\text{olap}} = \sum_{i=0}^{N-4} \sum_{j=i+4}^{N-1} ((1 + i - j) \mod 2) \gamma_{jk},
\]
\[
H_{\text{pair}} = \sum_{i=0}^{N-4} \sum_{k=i+3}^{N-1} ((k - i) \mod 2) \omega_{ik} G_{ik}.
\]

\( H_{\text{back}} \) sets a penalty for the overlapping of two consecutive aminoacids (\( \lambda_{\text{back}} > 0 \)), where \( d_{sk}^{j} \) with \( k = 1, 2, 3 \) (or alternatively \( k = x, y, z \)), \( s \in \{ \pm 1 \} \) takes the value of 1 only if turn \( j \) goes into the direction \( sk: d_{x}^{j} = q_{1}\varphi q_{2}+\varphi q_{3}+\varphi, d_{y}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi, d_{z}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi, d_{x}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi, d_{y}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi, d_{z}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi. \) There, \( q_{0} = 1 - q_{0} \) and \( \varphi = 3(j - 2) \). \( H_{\text{redun}} \) states a penalty for a non-encoded turn direction (\( \lambda_{\text{redun}} > 0 \)), with:
\[
d_{000}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi, d_{011}^{j} = q_{1}+\varphi q_{2}+\varphi q_{3}+\varphi.
\]
Figure 1. a) Lattice coding using three qubits by each aminoacid spatial direction in the TTR-SC ($N = 8$, $D = 3$), and b) Turn coding using three qubits by turn in the TC.

$H_{\text{olap}}$ sets a penalty to avoid the overlapping among non-consecutive aminoacids as follows. Distance $D_{jk} = (x_j - x_k)^2 + (y_j - y_k)^2 + (z_j - z_k)^2$ between $j < k$ aminoacids should be codified, it is defined departing from the cumulative position of aminoacid $m$ calculated from the turns code (which are $2$–local):

$$x_m = m\Theta_m^{m\in S} + (1 + q_0)\Theta_{m\geq 2} + \Theta_{m>2} \sum_{j=2}^{m-1} (d^l_jx - d^l_x),$$

$$y_m = (1 - q_0)\Theta_{m\in S} + \Theta_{m>2} \sum_{j=2}^{m-1} (d^l_jy - d^l_y),$$

$$z_m = \Theta_{m>2} \sum_{j=2}^{m-1} (d^l_jz - d^l_z),$$

where $\Theta_m^{m\in S} = 1$ if $m \in S$, otherwise $\Theta_m^{m\in S} = 0$. Then, the quadratic distance becomes $4$–local because $3$–local terms in the sums are cancelled. Due to non-consecutive aminoacids $j < k$ could overlap only if between them there are an odd number of aminoacids, the maximum number of qubits to codify their maximal mutual quadratic distance is $\mu_{jk} = \lceil 2\log_2(k - j) \rceil ((1 + k - j) \mod 2), j < k$ ($\lceil x \rceil$ is the ceiling function). Then, the total number of qubits necessary to codify the maximal mutual distances between the whole non-consecutive qubits is $N_{\text{ancilla}} = \sum_{i=0}^{N-5} \sum_{j=i+4}^{N-1} \mu_{ij}$. Note that a minimal distance of $4$ is needed for an overlapping. In addition, a pointer for each ancilla codifying the distance is given by: $p_{jk} = (3N - 8) + \sum_{s=0}^{j} \sum_{t=s+4}^{N-1} \mu_{st} - \sum_{u=k}^{N-1} \mu_{ju}$. Thus, we construct $\alpha_{jk}$ as the difference between the maximal quadratic distance (linear) codified in $\mu_{jk}$ qubits and the real quadratic distance $D_{jk}$:

$$\alpha_{jk} = \sum_{l=0}^{\mu_{jk}-1} q_{p_{jk} + 2\mu_{jk} - 1 - l} 2^l. \gamma_{jk}$$

It fulfills the property: $0 \leq \alpha_{jk} \leq 2^{\mu_{jk}} - 1$. Then, defining:

$$\gamma_{jk} = \lambda_{\text{olap}}(2^{\mu_{jk}} - D_{jk} - \alpha_{jk})^2,$$

which is zero only if $D_{jk} \geq 1$, thus if $\lambda_{\text{olap}} > 0$, it becomes a penalty term for the overlapping of non-consecutive aminoacids. Finally, $H_{\text{pair}}$ boosts the $H$ interactions as in the SC model, the $G_{ij}$ is defined as in SC model, this in order to compare both
models for the H...HPPH...H protein, but it is closely related to other cases considering HP or MJ approaches. Note that $\omega_{jk} = 1$ if $D_{jk} = 1$ and 0 otherwise, between a couple of aminoacids $j \neq k$. Here, $\lambda_{\text{pair}} < 0$.

3. Statistical analysis of the model on a classical computer

With the models previously provided, we have constructed the energy distributions for each one by generating $10^6$ random strings as in (1) for SC and (6) for TC, then getting the respective statistics for narrow intervals on the energy scale. As in [9], but restricting us to the TTR model in the SC case (which becomes more optimal), we report the 3D case of such model in Figure 2a for 8 aminoacid in the HHHPPHHH protein model (green). Together, we report the TC case for the same 8-aminoacid model with the same interactions (red). Distributions show essential differences exhibiting the economy of TC models (before a large number of ancilla qubits should be added). The initial values of TC distribution compared with the SC one shows how the solutions of PF for TC are easier obtainable than in the SC model. Note the difference in the energy scales for both models together with the bell-shaped form lost for the TC model instead an almost uniform distribution preceded by a sudden raising near from $E = 0$. In any case, TC has the disadvantage of exhibiting more spread distributions than SC model for low values of $N$.

Figure 2. $H_p$ energy distributions $\rho(E)$ approximated from $10^6$ random configurations each one for a) TTR-SC (green) and TC (red) algorithms ($N = 8$, $D = 3$), and b) TC algorithm for $N = 6$ (green), 8 (red), 10 (cyan), 12 (orange), 14 (brown) aminoacids.

In addition, Figure 2b shows several plots related to the TC case showing the energy distributions (in a log plot) for $N = 6$ (green), 8 (red), 10 (cyan), 12 (orange), 14 (brown) aminoacids obtained from $10^6$ random configurations for each one following (6). Those distributions are numerically approximated, noting that extreme values (out of the range shown) are not obtained, but they still suggest the complete behavior. First, note the radical differences in the energy ranges which denote the increasing difficulty to find the lowest energies while $N$ increases, which gives the correct solutions for the PF problem. Also, the initial values of distributions decreases when $N$ increases, going from the sharply distribution in green for $N = 4$ (with larger values
near from $E = 0$) until the extended distribution in brown for $N = 14$, with surely very low
values near from $E = 0$. In any case, note that for extended values of the energy range, in the $10^6$
random configurations, the numeric approximations do not find negative values to approximate
exactly each distribution. The inset in Figure 2b shows the no-log version of the main plot,
exhibiting that bell-shaped distributions appear for larger values of $N$ which improves their
performance against the SC case.

4. Conclusions
A compact notation for classical simulation has been provided for the SC [9] and TC
algorithms for PF in AQC. Formulas provided explore their complexity and unfeasibility in
classical simulation but they also set a comparison between them in AQC. Random search of
configurations is a hard task even for simpler cases of eight aminoacids. However, exploring
the energy distribution of lattice models using a benchmark of a random sampling allow us to
undermine the configuration space of PF algorithms. AQC algorithms proposed to solve the
PF problems on lattice models are able to explore the configuration space to find the protein
native state. While SC in principle allow us to explore all possible configurations available for
a protein, TC has proved to be more efficient by limiting the configurations to those physically
meaningful. Therefore, developing algorithms that displace the energy distribution of models
into lower energies, it allow us to develop faster and more efficient algorithms still for AQC, but
also, to understand the physical phenomena behind PF. However, to solve PF for larger sequences
of aminoacids, still significant technological advances in quantum computers hardware should
be made.

References
[1] Levinthal C 1968 Chem. Phys. 65 44
[2] Anfinsen C 1973 Science 181 223
[3] Dill K A, Bromberg S, Yue K, Fiebig K M, Yee D P, Thomas P D and Chan H S 1995 Protein Science 4 561
[4] Farhi E, Goldstone J, Gutmann S, Lapan J, Lundgren A and Preda D 2001 Science 292 472
[5] Perdomo A, Truncik C, Tubert-Brohman I, Rose G, and Aspuru-Guzik A 2008 Phys. Rev. A 78 012320
[6] Babbush R, Perdomo A, O’Gorman B, Macready W and Aspurn-Guzik A 2014 Adv. Chem. Phys. 155 201
[7] Babej T, Ing C and Fingerhuth M 2018 Coarse-grained lattice protein folding Preprint quant-ph/1811.00713
[8] Miyazawa S and Jernigan R 1996 J. Mol. Biol. 256 623
[9] Anaya A and Delgado F 2019 J Phys: Conf. Ser. 1391 012016