QC protein folding - report

Johannes Schmidt, schmidt@ins.uni-bonn.de

October 25, 2022

1 Introduction

Research on quantum computers (QC), specifically algorithms, suggested a possibly huge impact in applications (Shors algorithm, Grover algorithm, phase estimation)[Sho94][Gro96][Kit95]. Since the hardware is not there yet (so called NISQ era) researchers still explore the possibilities of current hardware, although, these approaches seem to be less promising[LB20][MK18]. The Variational Quantum Eigensolver (VQE) is one of such approaches[ea] and it's used in a variety of applications. For example the simulation of molecules[PMS+14][KMT+17] or solving protein folding[RBWT21][1].

The aim of this report is to inspect the proposed *resource-efficient algorithm for protein folding*[RBWT21] and determine if one can expect its use in the industry. In other words, when we compare this quantum solution with it's classical competitor, is there some advantage?
Strong emphasis is put onto the following leading questions:

1 What are the classical parameters one feeds into the (quantum-)calculation of the expectation value. How are they expressed in the Hamiltonian?

2 Usually the Hamiltonian implemented on the QC is a relaxed version of the actual problem. How do the authors argue that their approximation is indeed related to the problem they try to solve?

In order to answer these questions, a short recap of the methods used there is presented, directly with focus on the leading questions (Section 2). After that the Hamiltonian is investigated in more detail (Section 3) and an attempt to compare its performance with a classical algorithm is done (Section 4). Finally the leading questions will be answered (Section 5).

2 Quick recap of [RBWT21]

Lets divide the paper into four parts:

- introduction
- qubits and Hamiltonian
- folding algorithm and scaling
- applications and conclusion

2.1 introduction

Here they present a rough overview over the work done. Among other things they claim the NP-hardness of the problem even when reduced to it's simplest model (Hydrophobic-Polar)[BGMP13]. Since there is no known quantum algorithm solving this problem in polynomial time there is **no evidence of an exponential speedup** using quantum computers, especially for NISQ-computers/VQE. What they present is a Hamiltonian with $\mathcal{O}(N^4)$ scaling with N being the monomers. This should not be confused with the run time of the algorithm. This encodes the length of the circuit, so captures the

[1] Code can be seen here: https://qiskit.org/documentation/nature/tutorials/09_Protein_Folding.html

time it takes to evaluate the energy of the system. Therefore one needs to do that step multiple times and adjust parameters in between. This leads to the question:

What would be the scaling of a classical energy computation?

This question however is not easily answered. See Section 4 for more investigation.

They also state their use of the so called *coarse-grained* model, which is a model commonly used in solving protein folding[KGK+16].

2.2 qubits and Hamiltonian

Qubits capture the 'turn' t_i, which is defined as the discrete angle between bead i and i+1. Because of the tetrahedral lattice structure there are only 4 possible angles in two different alternating states $\mathcal{A}$ and $\mathcal{B}$. Therefore the sequence can be encoded by with $N_{cf} = 4(N-3)$ qubits. Note that this assumes a one-hot (they call it *sparse*) encoding of the angles (so 1000 for t_i being the first angle out of the 4 possible, 0100 for t_i being the second, ...). In the paper there is also a *denser* encoding presented. That further optimizes this scaling by a factor of 2.

The scaling of the qubits encoding the interaction between the beads is $\mathcal{O}(N^2)$, because it's entirely determined by the main and side chains of the polymer. Here we choose l as a hyper parameter that desices for the number of neathrest neighbors considered for the energy calculation.

The Hamiltonian consists of 3 terms:

$$H(\mathbf{q}) = H_{\text{gc}}(\mathbf{q}_{\text{cf}}) + H_{\text{ch}}(\mathbf{q}_{\text{cf}}) + H_{\text{in}}(\mathbf{q}) \tag{1}$$

- H_{gf} dealing with *geometrical constraints*, so the contribution of the primary sequence itself.
- H_{ch} encoding the *chirality constraints*, so the contribution of the side chains.
- H_{in} for *interaction*

Furthermore we have $\mathbf{q} = \{\mathbf{q}_{\text{cf}}, \mathbf{q}_{\text{in}}\}$

2.3 folding algorithm and scaling

The solution of the folding problem is the ground state of the Hamiltonian $H(\mathbf{q})$. This is done by the CVaR-VQE (modicication of VQE)[BNR+20] and a Differential Evolution optimizer. The parameters optimized are denoted by $\boldsymbol{\theta} = (\boldsymbol{\theta}^{\text{cf}}, \boldsymbol{\theta}^{\text{in}})$ and define the values in the configurational and interaction registers/angles.

The scaling of the algorithm is defined by the number of pauli strings in the Hamiltonian. Its estimation was already stated in the introduction. For the arguments see the paper.

2.4 applications and conclusion

They did two folding experiments: 10 amino acid protein Angiotensin on a simulator and a 7 amino acid peptide on a IBM Q processor machine. They state that with the use of entanglement it could lead to a quantum advantage. So when comparing to the classical version, the improvements done by the CVaR in terms of the number of shots needed and possible entanglement should be considered. However, note that increasing the degree of entanglement leads to an increased length of the circuit[2]. So it depends on the precise application at hand and is therefore out of the scope of this report. Nevertheless it will only improve by a factor (see Section 4).

3 investigation of the Hamiltonian encoding

In this section formulas are avoided as much as possible to reduce redundancy with the source paper. Instead their high level ideas are presented.
As already mentioned there are two encodings presented, sparse and dense. For the formulas they decided for a generalized representation. So instead of the actual qubits they write it as functions $f_0(i)$,

[2]This can also be verified here

..., $f_3(i)$ which can be checked in (SI-3) to (SI-10) of the supporting information section of [RBWT21].

Before talking about the individual terms, here are the key points for the objective questions:

- The Hamiltonian basically consists of many constraints that make sure certain conditions are fulfilled a the value $H(\mathbf{q}) = 0$. For example for the sparse encoding a term that makes sure that only one of the qubits in the one hot encoding is one.
- As classical (hyper-) parameters there are some engineered numbers, denoted by λ_i with $i \in \{\text{back}, \text{chiral}, 1, 2, 3, 5\}$, used for deciding of the degree of penalty when such conditions are not fulfilled. These are usually large numbers with additional (sequence position dependent) constraints to make sure that the Hamiltonian is always non-negative and has the desired properties for zero.
- The energy contributions of the interaction are given by the $\epsilon_{i,j}$ parameters. They are also determined at the beginning through established tables.

Let's investigate on each individual term of the Hamiltonian separately now.

3.1 $H_{\mathbf{gc}}$: geometrical/growth constraints

There are two geometrical main constraints one needs to be aware of. First to make sure that no knots occur, so only one bead per lattice point. Second avoids occurrence of bifurcation, i.e. there as only one sequence defining the main chain and at most one additional bead representing the side chain.

3.2 $H_{\mathbf{ch}}$: chirality

This part enforces the correct chirality for the protein. How this is done in detail will not be reviewed in detail here, because it will not lead to much insight.

3.3 $H_{\mathbf{in}}$: interaction

In the section *The interaction qubits* of the paper they claim that the parameters contributing to the energy landscape can be chosen arbitrarily, or from previous related research. These parameters make sure that the ground state is actually related to the physical protein in terms of the relation between the beads.

They present an approximate realization of the delta distribution to successively decode the interactions between the beads. First direct neighbors, than second next and so on.

$$H_{\text{in}} = H_{\text{in}}^{(1)} + H_{\text{in}}^{(2)} + H_{\text{in}}^{(3)} + \cdots$$

This approach allows for using the tetrahedral lattice structure to further reduce the number of degrees of freedom in the Hamiltonian. Each $H_{\text{in}}^{(i)}$ is further reduced to the possible constellations and therefore distances between the beads.

4 classic vs. quantum

Lets first look at why there actually might be an improvement using a quantum algorithm for that problem at all:

The VQE algorithm aims to find the lowest eigenvalue of a Hamiltonian by iteratively generating the corresponding eigenstate. With use of entanglement, one can reduce the space of search and potentially speed up the process[WBW+20][DVPGR21]. However, than one needs to make sure that the solution lies within that space. Furthermore, if the eigenstate corresponding to the lowest eigenvalue is not an entangled one, there will be no advantage. This is because than it's necessary to check all 2^n states, which would lead to the same complexity as for the classical case[KAJ].

For example, when using VQE for combinatorial optimization there is no hope for improvement, because in this case the Hamiltonian is diagonal. Therefore, the eigenstates are not entangled[Nan19].

In our protein folding case, there is no diagonal Hamiltonian. This however, does not imply that the eigenstate of the lowest eigenvalue has to be an entangled state. In the source paper there is also no investigation on that. So there is no clear statement possible concerning this aspect.

Also note that this is not a comparison of the state-of-the-art method of quantum vs. the state-of-the-art method of classical computers. There are more sophisticated classical methods for solving this task, like heuristic simulations[LLPDS11] and deep learning approaches[JEP+21].

Lets now look a bit more into detail with what to expect from the CVaR-VQE compared to VQE. Much research has shown the advantage of CVaR-VQE[DVPGR21][BNR+20]. Note however that this only reduces the shots needed to yield the solution of the expectation values. This also leads only to a speed up of a constant factor. In the case of [RBWT21] it's roughly a factor of 10.

When one tries to find a classical version of the VQE there is no literature about it. This is probably because it's an artificial way to simulate a quantum algorithm with no evidence of improvement or anything useful. So the only ways to 'simulate' it classically would be to simulate what the quantum computer would do or to compute the expectation value classically. Both of them don't give valuable insights, because the first does the same thing and the second is infeasible. So this would not be a useful way of comparing quantum with classical algorithms.

This leaves the question, how to compare quantum algorithms against classical equivalents?

This was done through abstract investigation of complexity classes relative to oracles[3][BBBV97]. However, this is only useful for studying algorithms like Simons, Grover and Shors algorithm. In the NISQ-era there are no such theoretical results. So one has to investigate into more practical comparisons, like scaling of both state-of-the-art methods. This is clearly out of the scope of this report.

5 Conclusion

Let's answer now the two questions from the beginning:

What are the classical parameters one feeds into the (quantum-)calculation of the expectation value and how are they expressed in the Hamiltonian?

- As hyperparameters there are penalty parameters λ_i and the energy parameters $\epsilon_{i,j}$. The penalty parameters appear in every term of the Hamiltonian to enforce the necessary constraints. The energy parameters only appear in the interaction Hamiltonian.
- The parameters which are optimized are the angles $\boldsymbol{\theta} = (\boldsymbol{\theta}^{\text{cf}}, \boldsymbol{\theta}^{\text{in}})$. They also appear all the parts of the Hamiltonian.

Usually the Hamiltonian implemented on the QC is a relaxed version of the actual problem. How do the authors argue that their approximation is indeed related to the problem they try to solve?
They use a coarse-grained approximation on a tetrahedral lattice. This captures the important folding angles α-helix and β-sheet. The amount of the interaction (encoded in the $\epsilon_{i,j}$ parameters) is taken from tables of previous related work.

In terms of estimating the industrial relevance of this method there is no clear conclusion. Neither in the source paper, nor in this report. Only pointers of what would be important in further investigation were given here. So for answering this questions more insight into relevant applications and classical state-of-the-art methods is necessary.

[3] This blog post from Prof. Scott Aaronson also explains it well.

References

[BBBV97] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, oct 1997.

[BGMP13] Jacques M. Bahi, Christophe Guyeux, Kamel Mazouzi, and Laurent Philippe. Computational investigations of folded self-avoiding walks related to protein folding. *Computational biology and chemistry*, 47:246–56, 2013.

[BNR+20] Panagiotis Kl. Barkoutsos, Giacomo Nannicini, Anton Robert, Ivano Tavernelli, and Stefan Woerner. Improving variational quantum optimization using CVaR. *Quantum*, 4:256, apr 2020.

[DVPGR21] Pablo Díez-Valle, Diego Porras, and Juan José García-Ripoll. Quantum variational optimization: The role of entanglement and problem hardness. *Phys. Rev. A*, 104:062426, Dec 2021.

[ea] Nikolaj Mollet et al. Quantum optimization using variational algorithms on near-term quantum devices. DOI = 10.1088/2058-9565/aab822.

[Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search, 1996.

[JEP+21] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, and Clemens Me. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, August 2021.

[KAJ] KAJ226. Role of entanglement in a vqe ansatz for combinatorial problems. https://quantumcomputing.stackexchange.com/questions/17585/role-of-entanglement-in-a-vqe-ansatz-for-combinatorial-problems, 30.03.2022.

[KGK+16] Sebastian Kmiecik, Dominik Gront, Michal Kolinski, Łukasz Wieteska, Aleksandra Badaczewska-Dawid, and Andrzej Kolinski. Coarse-grained protein models and their applications. *Chemical Reviews*, 116, 06 2016.

[Kit95] A. Yu. Kitaev. Quantum measurements and the abelian stabilizer problem, 1995.

[KMT+17] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, Sep 2017.

[LB20] Frank Leymann and Johanna Barzen. The bitter truth about gate-based quantum algorithms in the nisq era. *Quantum Science and Technology*, 5(4):044007, Sep 2020.

[LLPDS11] Kresten Lindorff-Larsen, Stefano Piana, Ron O. Dror, and David E. Shaw. How fast-folding proteins fold. *Science*, 334(6055):517–520, 2011.

[MK18] Salvatore Mandrà and Helmut G Katzgraber. A deceptive step towards quantum speedup detection. *Quantum Science and Technology*, 3(4):04LT01, jul 2018.

[Nan19] Giacomo Nannicini. Performance of hybrid quantum-classical variational heuristics for combinatorial optimization. *Phys. Rev. E*, 99:013304, Jan 2019.

[PMS+14] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1), 7 2014.

[RBWT21] Anton Robert, Panagiotis Kl. Barkoutsos, Stefan Woerner, and Ivano Tavernelli. Resource-efficient quantum algorithm for protein folding. *npj Quantum Information*, 7(1), Feb 2021.

[Sho94] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[WBW+20] Andreas J. C. Woitzik, Panagiotis Kl. Barkoutsos, Filip Wudarski, Andreas Buchleitner, and Ivano Tavernelli. Entanglement production and convergence properties of the variational quantum eigensolver. *Physical Review A*, 102(4), oct 2020.