Quantum Computing - Algorithmic Efficiency and Optimization

by

Samay Kothari

HONORS PROJECT MONSOON-21

Submitted to
Center for Computational Natural Sciences and Bioinformatics

Supervisor:
Prof. Harjinder Singh

2021

ABSTRACT

Quantum Simulation is an important application of future quantum computers with applications in quantum chemistry, condensed matter and beyond. Quantum simulation of fermionic systems presents a specific challenge. The problem of mapping of fermions and fermionic operator to qubits and qubit operator, since there are differences in their properties, major one being that qubits in a quantum computer are distinguishable but fermions in a fermionic system are not. There are various transformations that can be done to overcome this problem, the jordan-wigner transformations that allows for representation of a fermionic operator by $\mathcal{O}(n)$ qubit operations and the Bravyi-Kitaev transformation that takes the simulation cost of $\mathcal{O}(\log n)$ qubit operations for one fermionic operations.

One of the most pragmatic approach applications of Variational Quantum Algorithms is estimating low-lying eigen states and corresponding eigenvalues of a give hamiltonian. Hence, the first proposed VQA, the Variational Quantum Eigensolver (VQE), was developed to provide a near-term solution to this task. Generalisation of Variational Quantum Eigensolver is a Variational Quantum Thermalizer. The goal of VQT is to prepare a thermal state of a given Hamitonian $\hat{H}$ at temperature T. The VQE gives us the ground state at of the hamiltonian at zero temperature, VQT can give is the thermal state for the given value of temperature, not only the ground state.

Chapter 1

Fermionic Mappings

1.1 Introduction

In his seminal article that anticipated the field of quantum information, Feynman argued that simulating quantum systems on classical computers takes an amount of time that scales exponentially with the size of the system, while the cost of quantum simulations can scale in polynomial time with system size. [1] A quantum simulation algorithm for quantum chemical Hamiltonians enables the efficient calculation of properties such as energy spectra, reaction rates, correlation functions, and molecular properties for molecules larger that those that are currently through classical calculations.
Quantum simulation of electronic structure requires a representation of fermions by systems of qubits. In 1997, Abrams and Lloyd proposed a simulation scheme for fermions hopping on a lattice [2]. In 2002, Somma et al. used the Jordan-Wigner to generalize the simulation scheme to proposed by Abrams and Lloyd. [3] The Jordan-Wigner transformation has since been used to outline a scalable quantum algorithm for the simulation of molecular electron dynamics.
Further refinements to these methods were made my Bravyi and Kitaev [4]. From the point of view of fundamental physics, such constructions show that quantum computation of electronic structure does not suffer from an analog of the sign problem; that is, fermion antisymmetry represents no significant obstacle to efficient algorithms.
A simulation scheme can be broken into two pieces: first, to map occupation number basis vectors to states of qubits; and second, to represent the fermionic creation and anhilation operators in terms of operations on qubits on a way that preserces the fermionic anti-commutation relations.
The Jordan-Wigner transformation is then used to write the electronic Hamiltonian as a sum over products of Pauli spin operators acting on the qubits of the quantum computer. Subsequently the hamiltonian terms h_k, where $\hat{H} \sum_k h_k$, are converted into the unitary gates that are the corresponding time evolution operators. Even though the h_k do not necessarily commute, their sequential execution on a quantum computer can be made to approximate the unitary propagator $e^{-i\hat{H}}t$ through a Trotter decomposition. Finally, the iterative phase estimation algorithm (IPEA) is used to approximate the eigenvalue of an

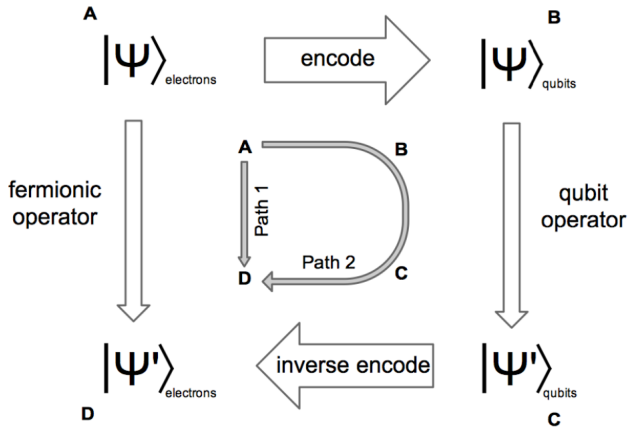

Figure 1.1: A simulation scheme first encodes fermionic states in qubits, then acts with the qubit operator representing the fermionic operator (obtained by the associated transformation), then inverts the encoding to obtain the resultant fermionic state. The criterion for a successful simulation scheme is that this procedure reproduces the action of the fermionic operator, in other words this diagram commutes.

input eigenstate.
The idea of Bravyi-Kitaev basis and transformation, named after the authors, provide a more effecient mapping between electronic and qubit hamiltonians. While the occupation number basis and the Jordan-Wigner transformation allow for the representation of a single fermionic creation or annihilation operator by $\mathcal{O}(n)$ qubit operation, the Bravyi-Kitaev transformations and basis require only $\mathcal{O}(\log n)$ qubit operations to represent one fermionic operator.

1.2 Jordan-Wigner Transformation

Fermionic systems go under formalism of second quantization, in which n single-particle states can be either empty or occupied by a spinless fermionic particle. In the context of quantum chemistry these n states represent spin orbitals, ideally one-electron energy eigenfunctions and often molecular orbitals found by the Hartree-Fock method. The form of electronic occupation number basis vectors suggests the following identification between electronic basis states on the left and states of our quantum computer:

$$|f_{n-1}\dots f_1 f_0\rangle \to |q_{n-1}\rangle \dots \bigotimes |q_1\rangle \bigotimes |q_0\rangle\,, \quad f_j = q_z \in 0,1$$

That is, we let the state of each qubit $|q_j\rangle$ store f_j, the occupation number of orbital j. We refer to this method of encoding fermionic states as the occupation number basis

for qubits. The next step is to map fermionic creation and annihilation operators onto operators. Expressing the qubit creation and annihilation operators in terms of pauli matricses suggest a way forward:

$$\hat{Q^+} = |1\rangle\langle 0| = \frac{1}{2}(\sigma^x - i\sigma^y), \quad \hat{Q^-} = |0\rangle\langle 1| = \frac{1}{2}(\sigma^x + i\sigma^y)$$

The mutual anti-commutation of the three Pauli matrices allows us to recognize that $\hat{Q}^{\pm}$ anti-commutes with σ^z. Thus if we represent the action of $a_j^\dagger$ or a_j by acting with $\hat{Q}_j^{\pm}$ and with σ^z on all qubits with index less that j, our qubit operators will obey the fermionic anti-commutation relations. Put differently, the states of our quantum compute rwill acquire the same phases under the action of our qubit operator as do the electronic basis states under the action of the corresponding creation or annihilation operator. Th effect of the string of σ^z gates is to introduce the required phase change of -1 if the parity of the set of qubits with index less thatn j is 1(odd), and to do nothing if the parity is 0(ever), where the parity of a set of qubits is just the sum(mod 2) of the numbers that represent the states they are in. We can completely represent the fermionic creation and annhilation operators in terms of basic qubit gates:

$$a_j^\dagger = Q_j^+ \bigotimes Z_{j-1}^{\rightarrow} = \frac{1}{2}(X_j \bigotimes Z_{j-1}^{\rightarrow} - iY_j \bigotimes Z_{j-1}^{\rightarrow})$$

$$a_j = Q_j^- \bigotimes Z_{j-1}^{\rightarrow} = \frac{1}{2}(X_j \bigotimes Z_{j-1}^{\rightarrow} + iY_j \bigotimes Z_{j-1}^{\rightarrow})$$

where:

$$Z_i^{\rightarrow} = \sigma_i^z \bigotimes \sigma_{i-1}^z \bigotimes \ldots \sigma_1^z \bigotimes \sigma_0^z$$

and where it is assumed that any qubit not explicitly operated on is acted on by the identity. The operator $Z_i^{\rightarrow}$ is a "parity operator" with eigenvalues ± 1, corresponding to eigenstates for which the subset of bits with index less than or equal to i has even or odd parity, respectively.
The above correspondence, a mapping of interacting fermions to spins, is the Jordan-Wigner transformation.

1.3 Bravyi-Kitaev Basis and Transformation

1.3.1 Bravyi-Kitaev Basis

Two kinds of information are required to simulate fermionic operators with qubits: the occupation of the target orbital, and the parity of the set of orbitals with index less than the target orbital. The previous two approaches are dual in the way that they store this information. With the occupation number basis and its associated Jordan-Wigner transfor- mation, the occupation information is stored locally but the parity information

is non-local, whereas in the parity basis method and its corresponding operator transformation, the parity information is stored locally but the occupation information is non-local.

The Bravyi-Kitaev basis is a middle ground. That is, it balances the locality of occupation and parity information for improved simulation efficiency. The general form of such a scheme must be to use qubits $|b_j\rangle$ to store partial sums $\sum_{s=k}^{l} f_s$ of occupation numbers according to some algorithm. In other words we can say, in the exposition that follows, when we write that a qubit "stores a set of orbitals", what is meant is that the qubit stores the parity of the set of occupation numbers corresponding to that set of orbitals.

Bravyi and Kitaev's encoding has an elegant binary grouping structure. A qubit of index j always stores orbital j. For even values of j, this is the only orbital that it stores, but for odd values of j, it also stores a certain set of adjacent orbitals with index less than j. Just as with the parity basis transformation, this encoding can be symbolized in a matrix β_n that acts on bit string vectors corresponding to occupation number basis vectors of length n to transform them to the corresponding Bravyi-Kitaev-encoded bit strings.

Given this encoding, we need to determine, for index j, which qubits in the Bravyi-Kitaev basis store the parity of all orbitals with index less than j, which qubits store a partial sum including orbital j, and which qubits determine whether qubit j has the same parity or inverted parity with respect to orbital j.

1.3.2 Sets of Qubits

These are the parity set (the qubits in the Bravyi-Kitaev basis that store the parity of all orbitals with index less than j), the update set (the qubits that store a partial sum including orbital j), and the flip set (the qubits that determine whether qubit j has the same parity as orbital j).

1.3.3 Bravyi-Kitaev Transformation

Operating in this basis requires that we find the analogues to the qubit creation and anni- hilation operators, as well as the parity operator, $Z_i^{\rightarrow}$, and the update operator, $X_i^{\leftarrow}$, in the Bravyi-Kitaev basis.

Even j

In the case that j in even, we should act with $Q^{\pm}$ on qubit j, just as for the Jordan-Wigner transformation, because the Bravyi-Kitaev encoding stores with $j = 0(mod2)$ in the qubit with same index. There are then two additional tasks that dictate how to

represent the fermionic operators in the Bravyi-Kitaev basis: determining the parity of occupied orbitals with index less than j, and updating qubits with index greater than j that store a partial sum that includes occupation number j.
For parity set of the qubits in P(j) is is equal to that of the set of orbitals with index less than j. By analogy with the Jordan-wigner method, we act with σ^z on all qubits with indices in $P(j)$. Secondly, by the analogy of the parity basis method, we also act with σ^z on all qubits in the appropriate $U(j)$; that is, we apply the operator $X_{U(j)}$. This has the effect of updating all the qubits that store a set of orbital j. The size of $U(j)$ also scales like $\mathcal{O}(\log n)$. To summarize: to represent $a_j^\dagger$ or a_j in the Bravyi-Kitaev basis, for j even, we act with σ^z on all qubits in $P(j)$, $\hat{Q}^\pm$ on qubit j, and with σ^x on all qubits in $U(j)$.

Odd j

In the case where j is odd, to represent the creation and annhilation of a particle on orbital j in the Bravyi-Kitaev basis, for j even, we could simply act with $Q^\pm$ on qubit j beacuse that qubit store only the occupation number of orbital j. For j odd, qubit j stores a partial sum of occupation numbers of orbitals including, but not limited to, orbital j. Thus, in this case the state of Bravyi-Kitaev qubit is j is either equal to the occupation of orbital j (if the partiy of the other orbitals that is stores is even), or opposite to that of orbital j (if the paritay of the other orbitals that it stores is 1).

Thus, whether representing the creation on annihilation of a particle in orbital j requires that we act with Q^+ or Q^- on qubit j in Bravyi-Kitaev basis depends on t he parity of all occupation numbers other than f_j that are included in the partial sums b_j, that is the parity of the flip set of the index j. If the parity of the set of qubits with indices in $F(j)$ is even, then the creation or annihilation of a particle in orbital j requires acting with $\hat{Q}^+$ or $\hat{Q}^-$, respectively, as usual. But if the parity of this set of qubits is odd, then the creation of a particle requires acting with $\hat{Q}^-$ and the annihilation of a particle requires acting with $\hat{Q}^+$.

Chapter 2

Variational Quantum Eigensolver

2.1 Introduction

The most exemplary application of the Variation Quantum Algorithms is estimation of the low-lying eigen states and corresponding eigenvalues of a given Hamiltonian. The Variational Quantum Eigensolver(VQE) is a flagship algorithm for quantum chemistry using near-term quantum computers. It is an application of the Ritz variational principle, where quantum computer is trained to prepare the ground state of given molecule.

The inputs to the VQE algorithm are a molecular are a molecular Hamiltonian and a parametrized circuit preparing the quantum state of the molecule. Within VQE, the cost function is defined as the expectation value of the Hamiltonian computed in the trial state. The ground state of the target Hamiltonian is obtained by performing an iterative minimization of the cost function. The optimization is carried out by a classical optimized which leverages a quantum computer to evaluate which leverages a quantum computer to evaluate the cost function and calculate the gradient at each step.

2.2 Steps for VQE

2.2.1 Step 1 (Initialisation)

The first step is to specify the molecule we want to simulate. There are different ways in which we do this, one is to give a list of symbols of the constituent atoms and a one-D array of the nuclear co-ordinates in atomic units.

The molecular structure can also be imported from a file such as: PDB coordinate, CHARMM, NAMD and X-PLOR style PSF topology files and DCD trajectory files, NAMD binary restart (coordinate) files, AMBER structure (PARM) and trajectory (CRD). After this we can build the electronic Hamiltonian of the molecule from the information we have about the system. The outputs of the function are the Hamiltonian,

represented as a linear combination of Pauli operators, and the number of qubtis required for the quantum simulation.

2.2.2 Step 2 (Encoding)

Next we define the quantum circuit that prepares the trial state of the molecule with an variational parameter θ which is to be optimized in order to find the best approximation to the ground state. For example if we take the initial state of the form:

$$|\Psi(\theta)\rangle = \cos(\theta/2)\,|1100\rangle - \sin(\theta/2)\,|0011\rangle$$

Here in the Jordan-Wigner encoding the first term represents the Hartree-Fock state where the two electrons in the molecule occupy the lowest energy orbitals. The second term encodes a double excitation of the HF state where the particles are excited.
These encodings is necessary as there are differences in the properties of the fermionic particles and for the qubits, major one being the difference in the commuting property.
The implementation of the circuit is done by using the rotation and coupling gates.

2.2.3 Step 3 (Cost Function)

The next step is to define the cost function to compute the expectation value of the molecular Hamiltonian in the trial state prepared by the circuit. The cost function is defined as:

$$C(\theta) = \langle\psi(\theta)|\, H\, |\psi(\theta)\rangle$$

That is we are trying to minimize the expectation value H over a trial state $|\psi(\theta)\rangle = U(\theta)\,|\psi_0\rangle$ for some ansatz $U(\theta)$ and intial state $|\psi_0\rangle$.

2.2.4 Step 4 (Parameter Optimization)

Once an efficiently parameterized variational form(cost function) has been selected, in accordance with the variational method, its parameters must be optimized to minimize the expectation value of the target hamiltonian.
A popular optimization strategy is gradient descent where each parameter is updated in the direction yielding the largest local change in energy. Consequently, the number of evaluations performed depends on the number of optimization parameters present. This allows the algorithm to quickly find a local optimum in the search space.
However this optimization strategy often get stuck in local minima, so instead of using this for VQE, we use an appropriate optimizer for optimizing a noisy objective function is the Simultaneous Perturbation Stochastic Approximation optimizer (SPSA). SPSA approximates the gradient of the objective function with only two measurements.

2.2.5 Step 5 (Running the VQE)

After we have defined everything we then run the algorithm for a number of steps, depending upon the complexity of the structure that we have. We also define the tolerance that the optimizer should keep to run the algorithm to the next step, until the number of steps are done.
Then to compare the results we take the optimized value of θ and construct the state with it, and can verify that with the results about the ground state that we get experimentally.

Chapter 3

Variational Quantum Thermalizer(VQT)

3.1 Introduction

The variational quantum thermalization[5] task can be described as the following: given a Hamiltonian $\hat{H}$ and a target inverse temperature $\beta = 1/T$, we wish to generate an approximation to thermal state.

$$\hat{\sigma}_\beta = \frac{1}{Z_\beta} e^{-\beta \hat{H}}, Z_\beta = tr(e^{-\beta \hat{H}})$$

Here Z_β is known as the thermal partition function. Our strategy will be to phrase this quantum simulation task as a quantum-probabilistic variational learning task. VQT is a task which is well suited task for the Quantum Hamiltonian Based Models(QHBM's)

3.2 Step 1 (Pre-processing)

First step is defining the hamiltonian for the graph where we have qubits as part of a cyclic graph, the heisenberg hamiltonian is given as:

$$\hat{H} = \sum_{(i,j) \in E} X_i X_j + Y_i Y_j + Z_i Z_j$$

where we have X, Y and Z as the pauli matrices acting on the i-th qubit. E is the set of the edges, along which the qubits are interacting, of the cyclic graph. In place of a parameterized classical energy function, we have a parameterized modular Hamiltonian operator, and the variational model is a thermal state of this operator.

3.3 Step 2 (Initialisation)

The first step of initialisation is the create an initial density matrix of the state which will be stored classically, p_θ. We consider p_θ distribution to be factorized latent space model which means that we can write it as an uncorrelated tensor product of 4 one-qubit density matrix that are diagonal in computational basis. Its major motivation is that p_θ scales linearly with number of qubits rather than scaling exponentially.

$$p_\theta^i = p_i(\theta_i) \left|0\right\rangle \left\langle 0\right| + (1 - p_i(\theta_i)) \left|1\right\rangle \left\langle 1\right|$$

we can define $p_i(\theta_i)$ to be sigmoid.

$$p_i(\theta_i) = \frac{e^{\theta_i}}{e^{\theta_i} + 1}$$

3.4 Step 3 (Ansatz Circuit)

The ansatz circuit or the $U(\phi)$ which is the quantum computation part of the algorithm. It is comprised of the rotation and coupling layers. These rotations and coupling parameters are ϕ which we manipulate during the process of the optimisation. To contruct the ansatz we combine the paramerterized rotation and coupling gates. The rotation layer is simply R_x R_y R_z gates applied to each qubits. The coupling layer is coupling gates placed between the edge of the interaction graph.

3.5 Step 4 (Entropy and Cost Function)

The von neumann entropy of the state, which is determined by collection of p_{θ_i}. Since the entropy of a collection of multiple uncorrelated subsystem, we can sum the entropy for each subsystem, we can sum the entropy values of each one-qubit system in the factorized space to get the total, the Von Neumann entropy of the latent and visible states are also identical due to the invariance of Von Neumann entropy under unitary action.

We combine the ansatz and the entropy to get the cost function, we use ansatz to calculate $\left\langle x_i\right| U^\dagger(\phi) \hat{H} U(\phi) \left|x_i\right\rangle$ for each basis state $\left|x_i\right\rangle$. We then multiply these expectation values by their corresponding $(p_\theta)_i$, which is exactly the probability of sampling $\left|x_i\right\rangle$ from the distribution. Summing these gives us the expected value of the hamiltonian.

$$L_{VQT}(\theta, \phi) = \beta tr(\hat{p}_{\theta\phi} \hat{H}) - S(\hat{p}_{\theta\phi})$$

The positivity of relative entropy implies that the minimum of free energy is achieved by the thermal state.

3.6 Step 5 (Optimization)

For the optimization task we can use different types of optimizers to do the task of minimizing the cost function, in the simulation done we use the COBYLA.

3.7 Inferences and Summary

VQT is using samples from the latent distribution $x \sim p_\theta(x)$, one applies a quantum operation to prepare the state $|x\rangle\langle x|$ via unitary $\hat{V}_x$ from the initial state of the quantum device. One then applies the unitary quantum neural network $\hat{U}(\phi)$ and estimates the expectation value of the Hamiltonian $\hat{H}$ at the output via several runs of classical sampling of x and measurement.

In the casen where our state converges to the true thermal state upon convergence of the training, then value of our loss function will give us the free energy of the thermal state, which is proportional to the log of the thermal partition function:

$$L_{VQT}(\theta,\phi) \xrightarrow{\hat{p}_{\theta\phi}\to\hat{\sigma}_\beta} \beta F(\hat{\sigma}_\beta) = -\log Z_\beta$$

This is effectively a variational free energy principle, and the basis of the VQT.

As a final note, let us see how we recover the Variational Quantum Eigensolver and its variational principle in the limit of low temperature. If we divide our loss function by β, directly minimizing the free energy, $L(\theta,\phi) = \frac{1}{\beta}L_{VQT}(\theta,\phi) = F(\hat{p}_{\theta\phi})$, then in the limit of zero temperature,

$$L(\theta,\phi) \xrightarrow{\beta\to\infty} \langle\hat{H}\rangle_{\theta\phi}$$

we recover the loss function for the ground state Variational Quantum Eigensolver(VQE), and this we recover the ground state variational principle.

References

[1] R. P. Feynman, "Quantum mechanical computers," *Optics news*, vol. 11, no. 2, pp. 11–20, 1985.

[2] D. S. Abrams and S. Lloyd, "Simulation of many-body fermi systems on a universal quantum computer," *Physical Review Letters*, vol. 79, no. 13, p. 2586, 1997.

[3] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, "Simulating physical phenomena by quantum networks," *Physical Review A*, vol. 65, no. 4, p. 042 323, 2002.

[4] S. B. Bravyi and A. Y. Kitaev, "Fermionic quantum computation," *Annals of Physics*, vol. 298, no. 1, pp. 210–226, 2002.

[5] G. Verdon, J. Marks, S. Nanda, S. Leichenauer, and J. Hidary, *Quantum hamiltonian-based models and the variational quantum thermalizer algorithm*, 2019. arXiv: `1910.02071 [quant-ph]`.