Complexity of Adiabatic Quantum Computing on Selected NP-Complete Problems

Liu Jiacheng

1   Introduction

The adiabatic theorem states that if the Hamiltonian evolves slowly enough, a state starting from the ground state will track the ground state. The adiabatic theorem provides foundation for AQC. In AQC we start from the ground state of a "beginning Hamiltonian" that is easy to prepare, and encode the solution in the ground state of a "problem Hamiltonian". The time complexity of this AQC algorithm is related to the time required to achieve a high success probability.

2   Algorithm

We take a linear algebra approach to quantum mechanics. Let basis state $|z\rangle = |b_{n-1} \dots b_0\rangle$ represent a subset that includes vertex i if $b_i = 1$, and excludes vertex i if $b_i = 0$. Generally there will be 2^n basis states, but since we are only interested in subsets of size k, we may consider the subspace spanned by basis states $|z\rangle$ with $h(z) = k$, and represent an arbitrary state as a $\binom{n}{k}$-dimensional vector.

For example, if $n = 4, k = 2$, the set of basis states is

$$\{|0011\rangle, |0101\rangle, |0110\rangle, |1001\rangle, |1010\rangle, |1100\rangle\}$$

and an arbitrary state

$$
\begin{aligned}
\psi &= \begin{bmatrix} c_0 & c_1 & c_2 & c_3 & c_4 & c_5 \end{bmatrix}^T \\
&= c_0 |0011\rangle + c_1 |0101\rangle + c_2 |0110\rangle + c_3 |1001\rangle + c_4 |1010\rangle + c_5 |1100\rangle \\
1 = |\psi|^2 &= \sum_{i=0}^{5} c_i^2
\end{aligned}
$$

We would like to use a uniform superposition of Hamming-k states as the beginning state vector. We would also like the Hamiltonian to be relatively local - the Hamiltonian matrix is sparse. Thus the beginning Hamiltonian is

$$H_B = -\sum_{i>j}^{n} S^{ij} \tag{1}$$

where S^{ij} swaps the ith and jth qubits. The non-degenerate ground state of H_B

$$|\psi(0)\rangle = \binom{n}{k}^{-1/2} \sum_{h(z)=k} |z\rangle \tag{2}$$

is exactly as desired.

H_B has diagonal values of $-\binom{n}{2} - \binom{n-k}{2}$, and off-diagonal values of -1. There are $k(n-k)$ non-zero off-diagonal elements in each column, where the row and column indices correspond to Hamming-k states that differ by a 0-1 swap.

For example, if $n = 4, k = 2$, then

$$H_B = P \begin{bmatrix} 1 \\ 4 \\ 5 \\ 2 \\ 3 \\ 6 \end{bmatrix} + P \begin{bmatrix} 4 \\ 2 \\ 6 \\ 1 \\ 5 \\ 3 \end{bmatrix} + P \begin{bmatrix} 5 \\ 6 \\ 3 \\ 4 \\ 1 \\ 2 \end{bmatrix} + P \begin{bmatrix} 2 \\ 1 \\ 3 \\ 4 \\ 6 \\ 5 \end{bmatrix} + P \begin{bmatrix} 3 \\ 2 \\ 1 \\ 6 \\ 5 \\ 4 \end{bmatrix} + P \begin{bmatrix} 1 \\ 3 \\ 2 \\ 5 \\ 4 \\ 6 \end{bmatrix} \tag{3}$$

$$= \begin{bmatrix} 2 & 1 & 1 & 1 & 1 & 0 \\ 1 & 2 & 1 & 1 & 0 & 1 \\ 1 & 1 & 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 2 & 1 & 1 \\ 1 & 0 & 1 & 1 & 2 & 1 \\ 0 & 1 & 1 & 1 & 1 & 2 \end{bmatrix} \tag{4}$$

The problem Hamiltonian is

$$H_P |z\rangle = \sum_{i>j} (1 - G_{ij}) z_i z_j |z\rangle \tag{5}$$

To verify if a given computation time guarantees significantly large probability of producing the correct answer, we find the final state by numerically simulating the evolution as dictated by the Schrödinger equation (the reduced Plank constant $\hbar$ is absorbed into the Hamiltonian)

$$i\frac{\partial |\psi(t)\rangle}{\partial t} = H(t) |\psi(t)\rangle \tag{6}$$

We use Euler method to compute the final state $|\psi(T)\rangle$: choose a small time step size Δt, and in each step we update the state vector by

$$|\psi(t + \Delta t)\rangle = |\psi_A(t + \Delta t)\rangle = |\psi(t)\rangle - i\Delta t H(t) |\psi(t)\rangle \tag{7}$$

However, this method is not scalable. In each step the time complexity is $O\left(\binom{n}{k}^2\right)$ if using dense-matrix representation of H, or $O(k(n-k)\binom{n}{k})$ if using sparse-matrix representation of H. Therefore, we borrow the idea of projector Monte Carlo.

The projector method finds the ground state of a Hamiltonian H by iteratively applying operator $P = e^{-H\Delta t}$ to an arbitrary initial state. If we make H changes slowly over time,

the state will track the ground state of H just as it does in the AQC evolution. So the new update rule is

$$|\psi(t + \Delta t)\rangle = |\psi_P(t + \Delta t)\rangle = |\psi(t)\rangle - \Delta t H(t) |\psi(t)\rangle \tag{8}$$

Theorem: if the Hamiltonian varies slowly enough over time, then the final state vector $|\psi_P(T)\rangle$ computed with the projector method is equivalent to the actual final state vector $|\psi_A(T)\rangle$ computed with adiabatic evolution, up to a phase factor and a normalization constant.

Proof: since the Hamiltonian varies slowly enough over time, the adiabatic condition is met, and the states computed with adiabatic evolution will track the ground state of the Hamiltonian in the entire process:

$$H(t) |\psi_A(t)\rangle = E_0(t) |\psi_A(t)\rangle \tag{9}$$

Suppose $|\psi_A(t)\rangle$ and $|\psi_P(t)\rangle$ are equivalent up to a complex scalar z:

$$|\psi_P(t)\rangle = z |\psi_A(t)\rangle \tag{10}$$

By (7-10),

$$|\psi_A(t + \Delta t)\rangle = |\psi_A(t)\rangle - i\Delta t H(t) |\psi_A(t)\rangle \tag{11}$$
$$= (1 - i\Delta t E_0(t)) |\psi_A(t)\rangle \tag{12}$$
$$|\psi_P(t + \Delta t)\rangle = |\psi_P(t)\rangle - \Delta t H(t) |\psi_P(t)\rangle \tag{13}$$
$$= (1 - \Delta t E_0(t)) |\psi_P(t)\rangle \tag{14}$$
$$|\psi_P(t + \Delta t)\rangle = z' |\psi_A(t + \Delta t)\rangle \text{ where } z' = \frac{1 - \Delta t E_0(t)}{1 - i\Delta t E_0(t)} z \tag{15}$$

So $|\psi_A(t + \Delta t)\rangle$ and $|\psi_P(t + \Delta t)\rangle$ are equivalent up to a complex scalar z'. QED

The states generated in this process is only a phase off from the previous process. Notably, the components of the state vector will be real and positive, so Monte Carlo can be utilized to approximate the evolution process.

We start the Monte Carlo random walk by sampling R random walkers $|z\rangle$ from the normalized distribution of $|\psi(0)\rangle$. Initially each random walker is assigned a weight $w = 1$. In each time step we randomly diffuse every random walker $|z\rangle$ by

$$(|z\rangle, w) \rightarrow (|z'\rangle, w') \tag{16}$$
$$P(z' \mid z, t) = \frac{\langle z'|I - H(t)\Delta t|z\rangle}{\sum_{h(z')=k} \langle z'|I - H(t)\Delta t|z\rangle} \tag{17}$$
$$w' = w \sum_{h(z')=k} \langle z'|I - H(t)\Delta t|z\rangle \tag{18}$$

To increase the stability and computational efficiency of random walk, a birth/death procedure is adopted: walkers with significantly large weight will be split into two walkers with the same state but downward-adjusted weight, while walkers with negligibly small weight will be removed. Finally $|\psi(T)\rangle$ is reconstructed from the random walkers at $t = T$, up to a re-normalization factor.

3 Experiment

When measuring the final state in the computational basis, there is a "success" probability of yielding a state that corresponds to a clique of size k. For practical purpose, we consider the evolution satisfactory if this success probability exceeds a fixed threshold $\frac{1}{2}$.

First, we generate a random graph of size n, and classically determine k, the size of its maximum clique, and the set $\{|z\rangle\}$ corresponding to cliques of size k. Then, we find the lower bound of computation time T that guarantees a success probability above our threshold $\frac{1}{2}$. Given a fixed time T, we simulate the adiabatic quantum evolution by projector quantum Monte Carlo. Since the success probability is a monotonically increasing function with respect to T, we can find the lower bound of T by binary search. The number of simulations to do is $\Theta(\log T)$.

For each $n \in [1, 32]$, we repeat the experiment $S = 100$ times. We use $R = 10^4$ walkers for random walk. We use time step size $\Delta t = 10^{-2}$. We use relative time accuracy $\epsilon = 10^{-2}$.

The overall complexity for one random graph is $\Theta(\log T)\Theta(T/\Delta t)\Theta(R)\Theta(1) = \Theta(R\frac{T}{\Delta t}\log T)$.

4 Result

Fig. 1 shows the distribution of minimal computation time that achieves success probability $\frac{1}{2}$.

5 Conclusion

With numerical simulation of adiabatic quantum computation and optimization with projector quantum Monte Carlo, we obtained data in a larger domain that supports the quadratic asymptotic time complexity of a quantum algorithm for finding maximum clique in random graphs. This also implies algorithms with quadratic complexity for maximum vertex independent set problem and minimum vertex cover set problem in random graphs, and polynomially fast algorithms for other NP-complete problems (since NP-complete problems are polynomially-reducible to each other) with certain restrictions. However, physical realization of these algorithms still remains challenging.