# Optimization via Quantum Adiabatic Computing

Fengwei Sun    Huiting Zhu

## 1 Optimization and Combinatorial Optimization

Mathematical optimization refers to a class of problems that require us to find the most optimal solution (in terms of a specific objective function) among a set of feasible solutions. These problems are of high importance in a variety of fields including physics, engineering and economics, and there has been a great amount of effort in deriving efficient algorithms to solve various types of optimization problems.

### 1.1 Optimization Problems

**Definition 1** *Let $S$ be the domain of $x$. A* ***mathematical optimization*** *problem has the following form:*

$$\textit{minimize } f(x) \quad \textit{subject to } g_i(x) \leq 0, i = 1, \ldots, m$$

*where $x \in S$ is the optimization variable of the problem, the function $f : S \to \mathbb{R}$ is the objective function and the functions $g_i : S \to \mathbb{R}, i = 1, \ldots, m$ are the constraint functions.*

Depending on the domain of the variable and the formulation of the objective/constraint functions, there is a great number of branches in the studies of optimization problems, including the following:

- Convex optimization
- Integer programming
- Combinatorial optimization

In the rest of the paper, we will focus on the discussion of combinatorial optimization and the algorithms to solve these problems.

### 1.2 Combinatorial Optimization

**Definition 2** ***Combinatorial optimization*** *is a process of searching for the maxima/minima of an objective function $f : S \to \mathbb{R}$ where the domain of the function $S$ is a discrete space.*

**Example (MAX-3-SAT):** Suppose we have a set of variables $X = \{x_i\}, i = 1, \ldots, n$ where $x_i \in \{0, 1\}$. Let a $3 - SAT$ formula be $\varphi(x_1, \ldots, x_n) = C_1 \cap C_2 \cap \cdots \cap C_k$ where $C_j = x_{j1} \cup x_{j2} \cup x_{j3}$ is a 3-literal clause of the variables $x_{j1}, x_{j2}, x_{j3}$, which are either variables in $X$ or their negations. For simplicity, let

$$C_j(x_{j1}, x_{j2}, x_{j3}) = \begin{cases} 1 & if\ C_j\ is\ satisfied\ by\ x_{j1}, x_{j2}, x_{j3} \\ 0 & otherwise \end{cases}$$

Let $f(x) = \sum_{j=1}^{k} C_j(x)$ be the objective function, then the optimization problem is defined as

$$\max_x f(x) \quad \textit{where } x \in \{0, 1\}^n$$

In other words, this problems try to find the maximum number of clauses that can be satisfied by any binary variable assignment over $x_i's$.

The decision version of this optimization problem is the following: is $max_x f(x) = k$, i.e. does there exist a variable assignment $x$ such that $f(x) = k$? The solution is "yes" if and only if all the clauses in the $3 - SAT$ formula are satisfied by some variable assignment. This problem is known as the **3-SAT** problem, which is NP-complete. This implies that MAX-3-SAT is also NP-hard, since solving MAX-3-SAT will give a solution to 3-SAT directly.

A number of combinatorial optimization problems have been proven to be NP-complete, including Knapsack problem, travelling salesman problem, integer programming and etc. While many of them have wide range of real-life applications, there's no efficient algorithm that solves any of the above problems in polynomial running time.

# 2 Quantum Speedups of Optimization Problems

## 2.1 Exponential Speedup of Solving Linear Systems

Inspired by the advantage of quantum algorithms against traditional algorithms in the problems including Simon's problem and Fourier transform, researchers are actively seeking for quantum speedups for optimization problems.

For linear optimization, one of the fundamental problems is to solve the equation $Ax = b$ in which $A \in \mathbb{R}^{n \times n}, b, x \in \mathbb{R}^n$. This is equivalent to solving the matrix inverse $A^{-1}$, which takes running time of at least $O(N^{2.3})$ on classical machines. With some assumptions on the input format and small matrix condition number, an algorithm was proven to solve the linear system in the running time of $O(\log(N))$ [15], which is an exponential speedup to the classical algorithms. More specifically, it gave an approximate solution in the quantum form, which might not be convenient for all applications. Even though there are multiple limitations of this algorithm to solve matrix inverse problems in general, it is believed that quantum algorithms do have an advantage in a wider range of optimization problems.

## 2.2 Development of Quantum Speedup for Combinatorial Optimization

For general combinatorial optimization problems, there has been no exponential speedup algorithm found using a quantum computer. Below is a brief illustration of the difficulties.

Consider the following problem:

**Unstructured search problem:** Suppose we are given a set of elements $X = \{x_1, \ldots, x_N\}$ and a function $f : X \to \{0, 1\}$. Our goal is to find an element $x^* \in X$ such that $f(x^*) = 1$.

This problem is the decision version of the combinatorial optimization problem

$$\max_{x \in X} f(x) = 1$$

and thus an exponential speedup of the unstructured search problem would lead to an efficient algorithm to solve combinatorial optimization problems.

It has been proven that, even though Grover's algorithm has a quadratic speedup from $O(N)$ to $O(\sqrt{N})$, this algorithm is optimal on a quantum computer. This indicate that no quantum algorithm would produce an exponential speedup for a combinatorial optimization problem given a black-box function $f(x)$. Although there's no room of development left for universal objective functions, researchers focused on the studies of optimization problems with structured objective functions and came up with several approaches, including quantum approximate optimization algorithm (QAOA), quantum annealing and adiabatic quantum optimization (AQO). In this paper, we will focus on the analysis of the current work in the area of adiabatic quantum optimization algorithms and their effectiveness.

# 3 Adiabatic Quantum Optimization

## 3.1 Adiabatic Evolution

The idea of encoding the solution of a computational problem in the ground state of a quantum Hamiltonian first appeared in 1989 [5], in the context of solving classical combinatorial optimization problems. It was later renamed quantum annealing (QA) and reinvented several times, and served as an impetus to reconsider a device to solve optimization problems by exploiting quantum evolution. The idea of adiabatic quantum optimization (AQO) was thus raised [13], wherein a quantum computer solves a combinatorial optimization problem by evolving adiabatically in its ground state. It turned out that adiabatic quantum computing is not limited to optimization problems and that the computational power of the circuit model and the adiabatic model of quantum computing are equivalent up to polynomial overhead. Both are universal for quantum computing.[1]

While QA and AQO both solve computational problems via quantum evolution towards the ground state of final Hamiltonians that encodes classical optimization problems, the evolution in QA is not necessarily adiabatic or universal and is dependent on hardware. [19] In contrast, quantum adiabatic evolution, according to Adiabatic Theorem (AT), guarantees that the state of the system will always track the instantaneous ground state, provided the Hamiltonian varies "sufficiently slowly".

In quantum adiabatic computing, the computation proceeds from an initial Hamiltonian $H_B$ whose ground state is easy to prepare, to a final Hamiltonian $H_P$ whose ground state encodes the solution to the computational problem. The evolution of the quantum state is governed by a time-dependent Hamiltonian

$$H(s) = (1-s)H_B + sH_P$$

according to the Schrödinger equation

$$\frac{1}{T}\frac{d}{ds}|\varphi_T(s)\rangle = -iH(s)|\varphi_T(s)\rangle,$$

where $s : [0,T] \mapsto [0,1]$ is the evolution schedule. Here we fix $s$ by the linear interpolation function $s(t) = t/T \in [0,1]$ so that $T$ controls the rate at which $H(s(t))$ varies. Notice that $s$ is $T$-independent. A formal definition of quantum adiabatic computing [2] is given as follows:

**Definition 3 (Adiabatic Quantum Computation)** *A k-local adiabatic quantum computation is specified by two k-local Hamiltonians, $H_0$ and $H_1$, acting on n p-state particles, $p \geq 2$. The ground state of $H_0$ is a product state. The output is a state that is $\varepsilon$-close in $l_2$-norm to the ground state of $H_1$. Let $s(t) : [0,T] \mapsto [0.1]$ (the "schedule") and let $T$ be the smallest time such that the final state of an adiabatic evolution generated by $H(s) = (1-s)H_0 + sH_1$ for time $T$ is $\varepsilon$-close in $l_2$-norm to the ground state of $H_1$.*

## 3.2 Adiabatic Theorem

We consider the computation to be successful if, as $s$ smoothly varies from 0 to 1, the evolution could converge to the target ground state of $H(s)$ with arbitrarily small error in polynomial time. Realizing a quantum speedup over classical computation requires a careful analysis and choice of the adiabatic schedule $s(t)$ and the structure (e.g., smoothness) of the Hamiltonian. Here we are going to have a closer look at the adiabatic theorem that essentially form the backbone of the adiabatic quantum computing, as it provides a sufficient condition for the success of the computation and quantifies the nature of the slow variation. There are many variants of the adiabatic theorem, with different assumptions and performance guarantees. All rigorous versions of AT have the following essential assumption: the eigenvalues $\varepsilon_0(s)$ are always separated from the rest of the Hamiltonian's spectrum by a non-vanishing gap. Here the gap refers to the ground state gap, i.e., the minimum eigenvalue gap between the ground state and the first excited state of $H(s)$

$$\Delta = \min_{s\in[0,1]} \Delta(s) = \min_{s\in[0,1]} \varepsilon_1(s) - \varepsilon_0(s) > 0,$$

where $\varepsilon_j(s)$ denote the eigenvalue of the instantaneous eigenstate $|\varepsilon_j(s)\rangle$ of $H(s)$ and $\varepsilon_0(s) \leq \varepsilon_1(s) \leq \cdots$. We summarize a few of the significant results and refer the audience to the original literature for the details and proofs.

- **Gap dependence of total evolution time** The best gap dependence to date was given in [9]. Assuming that $H(s)$ is bounded and infinitely differentiable and that $H(s)$ belongs to the Gevrey class $G^{\alpha}$, i.e., the Hamiltonian is varied sufficiently smoothly, an error bound of $o(1)$ can be ensured if the total evolution time $T$ scales as the square of the inverse gap.

- **Arbitrarily small error** It was proved in [14] that, with the assumptions of vanishing boundary derivatives at $s = 0, 1$ and the Gevrey condition, the adiabatic error can be made exponentially small in $T$ at $s = 1$, if the total evolution time $T \gg \frac{C^2}{\Delta^3}$ for some constant $C$. Notice that in this version of AT, the inverse gap dependence is cubic.

# 4 Failures of Adiabatic Quantum Optimization

We are interested in whether adiabatic quantum optimization can provide quantum speedups to solve NP-complete problems over classical algorithms. In AQO, the eigenvalue of final Hamiltonian (hopefully the ground state and thus the solution to the NP-complete problem) is determined by measuring the system state, so the quantum algorithms give the answer probabilistically. We hope that the system converges to the desired ground state with probability close to unity in polynomial time.

## 4.1 Minimum Gap and Quantum Speedups

According to the adiabatic theorem, the efficiency of a quantum adiabatic algorithm is limited by the spectral gap $\Delta$ between the two lowest-energy instantaneous eigenstates of the system Hamiltonian. Specifically, if the instantaneous eigenvalue gap becomes exponentially small at any point in the evolution, the computation time would grow exponentially. Bounding the minimum gap $\Delta$ is one of the most fundamental problems in adiabatic quantum computing, which is non-trivial and remains an open challenge.

Numerical simulations have raised examples of polynomially decaying gaps for some instances of NP-complete problems [11, 20] as well as examples in which AQC takes exponential time and thus fails to outperform classical algorithms [21, 8]. However, the scale of numerical simulations is constrained by the problem size, since the computation grows (and the eigenvalue gap $\Delta$ decreases) exponentially with the system size, while classical algorithms do not suffer such a slowdown. Also, they provide little insight to design efficient adiabatic quantum algorithms. It is more important to unveil the quantum evolution black-box and to obtain insights to explain the slowdowns in the performance of adiabatic algorithms.

## 4.2 Mechanisms of Performance Slowdowns

We consider the formation of the minimum gap as is relevant to the strucutre of the problem and hope to gain insights for designing efficient adiabatic algorithms through the study of the slowdown mechanisms for AQC. Recall that an adiabatic algorithm is described using three variable components of the time-dependent system Hamiltonian: (1) initial Hamiltonian $H_B$; (2) problem Hamiltonian $H_P$; (3) evolution schedule $s : [0, T] \mapsto [0, 1]$. A different component can give a different adiabatic optimization algorithm for the same problem. Poor choices for the initial Hamiltonians and final Hamiltonians could destroy the structure contained in the cost function and result in algorithmic failure. A time-optimal evolution path for adiabatic computing was suggested in [18, 17] and proven to have the power to improve the performance of an adiabatic algorithm by increasing the dimension of the control parameter space. Furthermore, modification on the initial and/or problem Hamiltonians help overcome the occurring of the exponentially small minimum gap during the evolution due to anti-crossings as a result of a first-order quantum phase transition. [10, 6, 7] Explanations and examples for each case have been summarized below:

- **Poor choices for the initial Hamiltonian**

  Suppose that $h(z)$ is a classical cost function to be minimized and is used to define a problem Hamiltonian diagonal in the $z$ basis with a ground state subspace of dimension $k$: $H_P = \sum_z h(z)|z\rangle$.

If one chooses the initial Hamiltonian

$$H_B = E(\mathbb{I} - |s\rangle\langle s|)$$

to be a one-dimensional projector onto the uniform superposition $|s\rangle$ over all basis states $|z\rangle$, then the computational time

$$T \geq \frac{b}{E}\sqrt{\frac{N}{k}} - \frac{2\sqrt{b}}{E}$$

where $b = \langle\varphi(T)|P|\varphi(T)\rangle > 0$ is the success probability and $P$ is the spectral projector onto the ground subspace of $H_P$. Regardless of the form of $h(z)$, $T/\sqrt{N} \to 0$ as $N \to \infty$. The adiabatic algorithm fails because all $|z\rangle$ states are treated identically by $H_B$, and any structure contained in the cost function is lost. [12]

- **Poor choices for the final Hamiltonian**

  Suppose that $h(z)$ is a cost function to be minimized and $\pi$ is a permutaion over the $N$ computational basis states such that $h^{\pi}(z) = h(\pi^{-1}(z))$. Consider a problem Hamiltonian

$$H_{P,\pi} = \sum_{z=0}^{N-1} h^{\pi}(z)|z\rangle\langle z| = \sum_{z=0}^{N-1} h(z)|\pi(z)\rangle\langle\pi(z)|$$

  and the final Hamiltonian

$$H_{\pi}(t) = H_D(t) + c(t)H_{P,\pi}$$

  for an arbitrary $\pi$-independent $H_D(t)$ with $|c(t)| \leq 1$ for all $t$.

  If the algorithm succeeds with probability at least $b$ for a set of $\epsilon N!$ permutations, then

$$T \geq \frac{\epsilon^2 b}{16h^*}\sqrt{N-1} - \frac{\epsilon\sqrt{\epsilon/2}}{4h^*}$$

  where $h^* = \sqrt{\sum_z h(z)^2/(N-1)}$. It is impossible for the algorithm to find the minimum of $h^{\pi}$ in time less than order $\sqrt{N}$ even for a fraction of a typical permutaion $\pi$. Even though $h^{\pi}(z)$ and $h(z)$ have the same values, the relationship between the input and output is scrambled by the permutation, which destroys any structure in $h(z)$ and results in algorithmic failure. [12]

- **Non-optimal evolution schedule**

  An adaptive adiabatic schedule that slows down as the gap decays, is an essential approach for a quadratic speedup with adiabatic Grove algorithm [18]. The ideas of adaptive or locally optimized adiabatic schedule rose from a variational time-optimal strategy for determining the interpolation of Hamiltonian. It is optimal in the sense that it raises the shortest total evolution time $T$ while guaranteeing that the final evolved state is close to the target final ground state. Consider the Hamiltonian with a set of control parameters $\vec{x}(t)$ such that $H(t) = H[\vec{x}(t)]$, and a parameterization of $\vec{x}(t)$ in terms of $s(t)$ with $s(0) = 0$ and $s(T) = 1$. $v = ds/dt$ characterizes the speed of motion along the control trajectory $\vec{x}[s(t)]$. The total evolution time is thus given by

$$T = \int_0^1 \frac{ds}{v(s)}.$$

  Define the Lagrangian

$$\mathcal{L}[\vec{x}(s), \dot{\vec{x}}(s)] \equiv \frac{\|\partial_s H(s)\|^2_{HS}}{\Delta^p(s)} \quad (p > 0)$$

  based on the form of the adiabatic condition, and the adiabatic-time functional

$$\mathcal{T}[\vec{x}(s)] = \int_0^1 ds\mathcal{L}[\vec{x}(s), \dot{\vec{x}}(s)].$$

One can find the time-optimal curve $\vec{x}_{QAB}(s)$ (the quantum adiabatic brachistochrone) by solving the geodesic equation $\delta\mathcal{T}[\vec{x}(s)]/\delta\vec{x}(s) = 0$ [17]. The optimal path is a geodesic in the control manifold, so as the gap becomes smaller, the curvature of the control manifold is higher, and the evolution schedule slows down. It was shown that this reformulation using the time-optimal strategy can be recast in a natural differential-geometric framework and improve the performance of adiabatic quantum algorithms for which the gap (or estimate thereof) is known.

- **First-order phase transition and perturbative crossings**

  During adiabatic evolution, an adiabatic quantum computer may go through a quantum phase transition (QPT). It was shown in [4] that a very small gap $\Delta$, extremely sensitive to the Hamiltonian parameters, may arise if the anti-crossing between the energy levels corresponding to the local and global minimum is a result of a first-order QPT. A first-order QPT happens if the final Hamiltonian has some low energy local minima with small Hamming distance between each other compared to the global minimum. Perturbation theory indicates that the size of the minimum gap depends exponentially on the Hamming distance between the two minima involved in the phase transition. On the other hand, a first-order QPT and thus an exponentially small minimum gap may not occur if none of the local minima has the above properties [4].

  [3] demonstrated the problem of perturbative crossings close to the end of the adiabatic evolution and thus the occurring of an exponentially small minimum gap, for random instances of the NP-complete exact cover problem. They related the exponentially small minimum gap to Anderson localization of the eigenfunctions of $H(s)$ in the ground state space, and proved that the Hamming weight between the two low-energy eigenstates can be $\Theta(n)$, which is problematic for the adiabatic algorithm. It was also claimed that adiabatic quantum optimization failed with high probability close to unity for large random instances of the problem and the failure did not rely on specific form of the problem Hamiltonian for Exact Cover. However, the generality of the second claim was questioned and it was shown later that the problem only occurs for one particular implementation of the adiabatic algorithm.

  Farhi et al. [10] suggested that it is possible to overcome the exponentially small minimum gap due to a first-order QPT by randomly modifying the adiabatic path, by keeping the same problem Hamiltonian $H_P$ but choosing a random initial Hamiltonian $H_B$. Repeating the procedure a number of times polynomial in the problem dimension $k$ (if the problem instance has $k$ clauses) removes small gaps near $s = 1$ with substantial probability. Changing the parameters in the problem Hamiltonian (without modifying the problem instances) will also prevent the first-order QPT from occurring. [6, 7] demonstrated that polynomial reduction of NP-complete problems might only preserve the solution and alter the energy levels of the problem Hamiltonian. The minimum gap can be increased drastically when the excited energy levels are changed.

# 5 A Short-Path Algorithm for Combinatorial Optimization

Despite the concerns in the previous section, we believe that a well-conditioned problem could be efficiently solved by the adiabatic quantum algorithms. In this section, we focus on a specific combinatorial optimization problem MAX-E$D$-LIN2 and present an algorithm [16] that solves it with better efficiency than the Grover's algorithm.

## 5.1 Problem Description

The problem is to find the ground state of a problem Hamiltonian $H_Z$, assuming that the ground energy $E_0$ is known. For this specific problem, the following conditions of $H_Z$ are imposed for simplicity and the validity of the algorithm:

- $H_Z$ is a weighted sum of products of Pauli Z operators, where each product has $D$ operators on distinct qubits. For instance, if $D = 2$, this Hamiltonian is an instance of the Ising model.

- Each product has an integer-valued weight, and the sum of the absolute values of all the weights are limited to a polynomial size of $N$, where $N$ is the number of variables in the problem.
- If $D$ is odd, we assume $H_Z$ has a unique ground state; otherwise, we assume that $H_Z$ has a doubly degenerate ground state.

With the above assumptions about the problem Hamiltonian $H_Z$, we can provide the following results which would lead to an efficient algorithm to solve the ground state, either exactly, or with a good approximation.

## 5.2 Main Results

Denote $J_{tot}$ as the sum of the absolute values of the weights of all the products in $H_Z$, and $W(E)$ as the number of computational basis states with expectation value $E$ for $H_Z$.

**Theorem 4** *Assume that $H_Z$ follows the assumption above, and let $B = -bE_0$ where $b \in (0,1), K = C\log(N)$ for some constant $C$. Then at least one of the following holds:*

1. *The algorithm finds the ground state in expected time*
$$O(2^{N/2} \exp{[-\frac{b}{2CD}\frac{N}{\log(N)}]})$$
2. *There is some probability distribution $p(u)$ on the computational basis states with entropy at least*
$$S^{comp} \geq N(1-O(1)/C)$$
*and with expected value of $H_Z$ at most $(1-b)E_0 + O(1)\frac{J_{tot}}{N^2}C^2D^2\log(N)^2$. Moreover, for any $\eta > 0$ and energy $E$ such that*
$$E \leq E_0 + (1+\eta)(b|E_0| + O(1)\frac{J_{tot}}{N^2}C^2D^2\log(N)^2)$$
*we have*
$$\log(W(E)) \geq N(1 - O(1)\frac{1+\eta}{\eta}\frac{1}{C}) - \frac{1+\eta}{\eta}O(\log(N))$$

With the above theorem, we are able to derive an algorithm that satisfies the following corollary.

**Corollary 5** *Given $H_Z$ and $E_0$, there's an algorithm that outputs either of the following:*

1. ***Approximate:** it returns a state with energy at most $E' = E_0 + 1.01(b|E_0| + O(1)\frac{J_{tot}}{N^2}C^2D^2\log(N)^2)$ with running time at most $O(2^{O(1)N/C})$;*
2. ***Exact:** it returns the exact ground state of $H_Z$ with expected running time $O(2^{N/2}\exp{[-\frac{b}{2CD}\frac{N}{\log(N)}]})$.*

*Proof:* Let $\eta = 0.01$, and run the algorithm below.

**Step 1:** Try repeated sampling of states to find a state with energy at most $E'$ by taking $O(2^{O(1)N/C})$ samples. If any sample succeeds, terminate the algorithm and return "Approximate" and the corresponding state. Assuming the correctness of part 2 in the above theorem, each sample would succeed with probability $2^{-O(1)N/C}/poly(N)$, and thus the probability that there is one success in all the samples is at least $1-2^{-N}$.

**Step 2:** If no sample succeeds in the previous step, run the short-path algorithm (described in the next section) in parallel with a brute force search until one of them finds the exact ground state, and return "Exact" and the corresponding state. Assuming the correctness of part 1 in the above theorem, the running time of the short-path algorithm is $O(2^{N/2}\exp{[-\frac{b}{2CD}\frac{N}{\log(N)}]})$. Considering the low failure rate in step 1, we match the expected running time of the algorithm in the corollary.

## 5.3 Algorithm Description and Key Ideas

**The Short-Path Adiabatic Algorithm**

**Step 1:** Prepare the qubits in the state $\psi_+ = |+\rangle^{\otimes N}$.

**Step 2:** Use the measurement algorithm to evolve the Hamiltonian $H_s$ from $s = 1$ to $s = 0$ where

$$H_s = H_Z - sB(X/N)^K, X = \sum_i X_i$$

**Step 3:** Measure the state in the computational basis and compute the value of $H_Z$ after the measurement; if it is equal to $E_0$, claim success and return the outcome of the measurement.

The measurement algorithm in the step 2 is where the adiabatic evolution takes place. Denote $E_{0,s}$ as the ground energy of the Hamiltonian $H_s$, and $\delta = \Omega(1/poly(N))$ as the minimum gap of the ground states of $H_s$ and the rest of the spectrum.

**The Measurement Algorithm**

**Step 1:** Let $\psi$ be the input state. Phase estimate $\psi$ using $H_1$. If the estimated energy is larger than $E_{0,1} + \delta/2$, then terminate the algorithm and return failure; otherwise continue.

**Step 2:** Adiabatically evolve $\psi$ from $H_1$ to $H_0$.

**Step 3:** Phase estimate $\psi$ using $H_0$. If the estimated energy is larger than $E_{0,0} + \delta/2$, then terminate the algorithm and return failure; otherwise claim success and return $\psi$.

**Key ideas:** There are several new ideas over the typical adiabatic evolution algorithms:

- A ”short-path” is used in the evolution instead of a full evolution from the initial Hamiltonian to the problem Hamiltonian.

  If we compare the evolution Hamiltonian used in this algorithm

  $$H_s = H_Z - sB(X/N)^K$$

  with the typical adiabatic evolution Hamiltonian

  $$H'_s = sH_Z + (1 - s)X$$

  We observe that, $H_s$ in this algorithm has a fixed term $H_Z$ since the start of the evolution, rather than $H'_s$ with zero weight on $H_Z$ when $s = 0$. Note that, the evolution schedule in this algorithm is to start with $s = 1$ and end with $s = 0$, which is different from the usual schedules of adiabatic evolution.

- Instead of using a typical transverse field $X$ as the initial Hamiltonian, a variation term $-B(X/N)^K$ is involved in the adiabatic evolution to avoid small spectral gaps.

  Suppose that there’s a small spectral gap in the evolution process. This indicates that there must be a state with large expectation value of $-B(X/N)^K$ and small expectation value of $H_Z$. For a large exponent $K$ in this term, this imposes a very strong constraint on the expectation value of $X$. Given a large expectation value of $X$, it can be derived from the log-Sobolev inequality that the entropy in the computational basis states is sufficiently large, and there must be many eigenstates of $H_Z$ with low energy.

### 5.4 Proof Ideas of the Algorithm

Consider the Hamiltonian $QH_sQ$ where $Q$ projects onto the states of $H_Z$ with energy greater than $E_0$.

**Theorem 6** *Consider the Hamiltonian $QH_sQ$. Let $E^Q_{0,s}$ be the ground energy of this Hamiltonian in the subspace spanned by the range of $Q$. Assume that $E^Q_{0,1} \geq E_0 + \frac{1}{2}$, and $\langle 0|B^2(X/N)^{2K}|0\rangle \leq \frac{1}{2}$. Then*

1. *$H_s$ has gap at least $\frac{1}{2}$ between the ground and the first excited state;*
2. *Let $\psi_{0,s}$ be the ground state of $H_s$. We have*

$$\langle \psi_+ |\, \psi_{0,1}\rangle \geq \frac{1}{2}2^{-N/2}(\exp\,[\frac{BN}{(2DK+O(1/N^3))|E_0|}] - O(1))\frac{1}{poly(N)}$$

**Theorem 7** *Assume that $E^Q_{0,1} < E_0 + \frac{1}{2}, K = C\log(N)$ and $B = -bE_0$ with $b \leq 1, C = \Theta(1)$. Then there is some probability distribution $p(u)$ on computational basis states with entropy at least $S^{comp} \geq N(1-O(1)/C)$ and with expected value of $H_Z$ at most*

$$(1-b)E_0 + O(1)\frac{J_{tot}}{N^2}C^2D^2\log(N)^2$$

Theorem 6 gives the result that, under the condition $E^Q_{0,1} \geq E_0 + \frac{1}{2}$, the spectral gap in $H_Z$ is bounded, which guarantees the nontrivial speedup of the adiabatic algorithm against Grover's algorithm. On the other hand, under the other case of $E^Q_{0,1} < E_0 + \frac{1}{2}$, Theorem 7 shows the existence of probability distributions over computational basis states with high entropy and low energy, which is used to impose a lower bound for $W(E)$.

## 6 Conclusion and Future Development

Even though it has been proven that there's no quantum algorithm that outperforms the Grover's algorithm for black-box combinatorial optimization, a nontrivial speedup has been found for the NP-hard problem **MAX-E*D*-LIN2**, which either produces the exact solution or a good approximation to it.

There are several assumptions in the proof of correctness of the algorithm. For the degeneracy assumption which limits the number of ground states of $H_Z$, it can be removed by considering a slight variation of the original Hamiltonian for the adiabatic evolution, which would lead to the same algorithm.

The other assumption is that all terms of $H_Z$ have the same degree $D$. This assumption has been used in the proof that the adiabatic evolution has a high probability of success. If this assumption is removed, there is no guarantee that the algorithm would still produce the exact solution with high probability. Therefore, for problems without this assumption (e.g. MAX-3-SAT), it is not well understood if the algorithm works.

In the future development, the studies on the generalization of this algorithm would be the focus. It might be hard to solve general combinatorial optimization problems through this short-path algorithm, but it gives a good direction that these problems could be approached through the analysis on the distribution of the low-energy states. Some sampling methods could help us understand these distributions and thus indicate if there's an efficient algorithm to solve any specific problem.

# References

[1] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, *Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation*, SIAM J. Comput., 37(1):166–194, 2007.

[2] T. Albash, and D. A. Lidar, *Adiabatic Quantum Computing*, arXiv:1611.04471v2, 2018.

[3] B. Altshuler, H. Krovi, and J. Roland, *Anderson Localization Makes Adiabatic Quantum Optimization Fail*, Proceedings of the National Academy of Sciences, 107(28):12446, 2010.

[4] M. H. S. Amin, and V. Choi, *First-order Quantum Phase Transition in Adiabatic Quantum Computation*, Physical Review A, 80(6):062326, 2009.

[5] B. Apolloni, C. Carvalho, and D. de Falco, *Quantum Stochastic Optimization*, Stochastic Processes and their Applications, 33(2):233-244, 1989.

[6] V. Choi, *Adiabatic Quantum Algorithms for the NP-Complete Maximum-Weight Independent Set, Exact Cover and 3SAT Problems*, arXiv:1004.2226, 2010.

[7] V. Choi, *Different Adiabatic Quantum Optimization Algorithms for the NP-Complete Exact Cover Problem*, Proceedings of the National Academy of Sciences, 108(7):E19-E20, 2001.

[8] W. van Dam, M. Mosca, and U. Vazirani, *How Powerful is Adiabatic Quantum Computation?*, Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science, 279-287, 2001.

[9] A. Elgart, and G. A. Hagedorn, *A Note on the Switching Adiabatic Theorem*, Journal of Mathematical Physics 53:10, 2012.

[10] E. Farhi, J. Goldstone, D. Gosset, S. Gutmann, H. B. Meyer, and P. Shor, *Quantum Adiabatic Algorithms, Small Gaps, and Different Paths*, Quantum Info. Comput., 11(3):181, 2011.

[11] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, *A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem*, Science 292(5516):472-475, 2001.

[12] E. Farhi, J. Goldstone, S. Gutmann, and D. Nagaj, *How to Make the Quantum Adiabatic Algorithm Fail*, arXiv:quant-ph/0512159, 2005.

[13] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, *Quantum Computation by Adiabatic Evolution*, arXiv:quant-ph/0001106, 2000.

[14] Y. Ge, A. Molnár, and J. I. Cirac, *Rapid Adiabatic Preparation of Injective PEPS and Gibbs States*, arXiv:1508.00570, 2015.

[15] A. W. Harrow, A. Hassidim, and S. Lloyd, *Quantum Algorithm for Solving Linear Systems of Equations*, arXiv:0811.3171v3, 2009.

[16] B. Hastings, *A Short Path Quantum Algorithm for Exact Optimization*, arXiv:1802.10124v3, 2018.

[17] A. T. Rezakhani, W. J. Kuo, A. Hamma, D. A. Lidar, and P. Zanardi, *Quantum Adiabatic Brachistochrone*, Phys. Rev. Lett., 103(8):080502, 2009.

[18] J. Roland,, and N. J. Cerf, *Quantum Search by Local Adiabatic Evolution*, Phys. Rev. A, 65(4):042308, 2002.

[19] W. Vinci, and D. A. Lidar, *Optimally Stopped Optimization*, arXiv:1608.05764, 2016.

[20] A. P. Young, S. Knysh, and V. N. Smelyanskiy, *Size Dependence of the Minimum Excitation Gap in the Quantum Adiabatic Algorithm*, Phys. Rev. Lett., 101(17):170503, 2008.

[21] M. Žnidarič, and M. Horvat, *Exponential Complexity of an Adiabatic Algorithm for an NP-Complete Problem*, Physical Review A, 73(2):022329, 2006.