Adiabatic Quantum Computation

Alice Balest[1]

[1] *Physics Department, ETH Zurich*[*]

(Dated: 30 May 2022)

Adiabatic Quantum Computation (AQC) is a computer architecture based on the results of the adiabatic theorem. The theorem states quantitative conditions to apply AQC. These conditions further influence the costs of AQC. Analyzing the costs one can find that in some cases AQC leads to a quantum speedup over classical calculations. AQC can simulate algorithms implemented in the circuit model up to a polynomial overhead. Universality of AQC follows from the equivalence of a unitary operation (implemented using a set of gates in the circuit model) and the Hamiltonian involved in the adiabatic calculation. Although AQC is universal, its application is limited by the size of the energy gap of the Hamiltonian involved in the computation and by the lack of information on the ground state for certain models. An alternative model to AQC, which combines adiabatic evolution with the circuit model, can be realized with adiabatic gate teleportation. In this manuscript we discuss the adiabatic theorem and the equivalence of AQC and circuit model. We then review examples of adiabatic algorithm such as the adiabatic Grover, adiabatic Deutsch-Jozsa, adiabatic Bernstein-Vazirani and the glued trees problem algorithm. Finally, we discuss adiabatic gate teleportation.

Contents

[*]Electronic address: balesta@student.ethz.ch

I. INTRODUCTION

The rough idea of adiabatic quantum computation is to solve a computational problem by incorporating its solution to an eigenstate of a Hamiltonian, which is in general hard to solve, and use adiabatic evolution to find the solution. To this end, one introduces a time-dependent Hamiltonian, with an initial form that is easy to solve and a final form that is identical to the Hamiltonian of interest. The crucial point is to slowly change the time-dependent Hamiltonian such that its eigenstate evolves towards the instantaneous state of the Hamiltonian describing the problem, where the solution to the computational problem is incorporated.

Why does the evolution of the time-dependent Hamiltonian need to be slow? How slow is the time-dependent Hamiltonian evolution for certain adiabatic algorithms? How is this Hamiltonian defined? Where is the final solution to the computational problem? These are some of the key points of theoretical understanding of AQC (applied to closed systems) discussed in this document.

An intuitive experiment to observe adiabatic evolution would be changing the length of an harmonic oscillating pendulum's string at different speed. Adjusting the length of the string almost instantaneously would change the frequency of the pendulum to the point of losing the oscillatory motion of the pendulum. Adjusting the length of the string gradually, would allow the pendulum to keep oscillating with a new frequency; the pendulum gets enough time to adapt while oscillating.

What allows to solve a computational problem through evolution method is the choice to evolve the time-dependent Hamiltonian in an adiabatic way. According to the adiabatic theorem, it is possible to evolve a system in a certain eigenstate of a time-dependent Hamiltonian $H(t)$ (solution to the time dependent Schrödinger equation) keeping the system in the same eigenstate as prior to the evolution. Intuitively, it means that it is possible to find an unknown instantaneous state (which is the solution of a certain computational problem), identifying this "solution" state thanks to the fact that we know where the solution is incorporated. It is enough to know in which energy level of the final Hamiltonian is incorporated.

In quantum physics, the instantaneous state $|\psi_f^0\rangle$ containing the solution is defined to be the solution to the Hamiltonian problem at time T ($H_f = H(t = T)$):

$$H_f|\psi_f^0\rangle = E_f^0|\psi_f^0\rangle \tag{1.1}$$

in the same energy level (for example the ground state) where the initial Hamiltonian ($H_i = H(t = 0)$) is initialized:

$$H_i|\psi_i^0\rangle = E_i^0|\psi_i^0\rangle. \tag{1.2}$$

The initial Hamiltonian is an arbitrary Hamiltonian, which is easier to prepare and to solve compared to the final Hamiltonian.

The initial and final Hamiltonian are different interpolated time solution of the time-dependent Hamiltonian $H(s)$:

$$H(s) = A(s)H_i + B(s)H_f \tag{1.3}$$

where $A(s)$ and $B(s)$ are time schedule. For example, linear interpolation would require $A(s) = s, B(s) = 1 - s$. An alternative is unitary interpolation, see Section 4.2 for an example of computation with unitary interpolation. To simplify, choosing $s = t/T$ allows to evolve the Hamiltonian over the interval $s \in [0, 1]$.

The condition on the speed of the evolution of the Hamiltonian to fulfill adiabatic evolution is stated by the adiabatic theorem, which exists in many variants. Some versions of the theorem statements are approximated conditions and require less restrictions on the Hamitonians involved in the calculus. More rigorous versions of the theorem have restrictions on the choice of the Hamiltonian and specified for each restriction how much the evolution needs to slow down to keep the system in the same eigenstate, for example the ground state. The costs of applying AQC can be directly connected to the conditions stated by the adiabatic theorem. Then it is possible to calculate, if it is worth to carry out the computation with AQC.

The idea of using adiabatic evolution has been generalized to non-optimization problem, which is the case of adiabatic evolution using non-stoquastic Hamiltonians. The new generalized definition of AQC with non-stoquastic Hamiltonians allowed to show that AQC is universal. Intuitively, it can be shown that a circuit model can replicate the Hamiltonian of the adiabatic computation. Vice versa, also the set of gates used in a circuit model to implement a unitary operation can be approximated by a Hamiltonian. Although the system is universal, its application to calculus is limited. In fact, it is difficult to find the ground state for some Hamiltonians. Moreover, the energy gap between ground state and first excited state of the Hamiltonian can be too small and reduce the efficiency of the quantum speedup (costs and energy gap are inversely proportional).

The review of AQC in this document is based on the results reported in Adiabatic Quantum Computation by Albash and Lidar [1]. In Section II, we present some quantitative conditions stated by the

adiabatic theorem and how these conditions impact the cost of applying AQC. In Section III, one can find the reciprocal simulations that lead to universality of AQC. Next to universality to solve computational problem, we briefly review decision problems such as the k-local Hamiltonian problem, which belongs to a different class of problems, and can be solved with AQC too. In Section IV, we discuss some successful applications of AQC architecture that lead to the same speedup obtained by the circuit model. We report the adiabatic Grover's algorithm, the adiabatic Deutsch-Jozsa algorithm, the adiabatic Bernstein-Vazirani algorithm, the glued trees problem and Google PageRank algorithm. In Section V, we discuss stoquastic AQC and explain how it is different from AQC. In Section VI, we present a description of adiabatic gate teleportation, which combines adiabatic evolution with circuit model's gates. This part of this review on AQC is based on Adiabatic Gate Teleportation by Bacon and Flammia [2]. Concluding remarks are stated in Section VII.

II. CONDITIONS AND COSTS OF AQC

A. The adiabatic theorem

According to [1], to realize the Hamiltonian evolution sufficiently slow for the system to stay in the same initial eigenstate j, the state $|\varepsilon_j(t)\rangle$ must fulfil one of the following approximated or rigorous condition or other versions of the theorem.

The more popular and probably oldest approximated criterion is:

$$\max_{t\in[0,t_f]} \frac{|\langle\varepsilon_i|\partial_t\varepsilon_j\rangle|}{|\varepsilon_i-\varepsilon_j|} = \max_{t\in[0,t_f]} \frac{|\langle\varepsilon_i|\partial_t H|\varepsilon_j\rangle|}{|\varepsilon_i-\varepsilon_j|^2} \ll 1\,, \qquad \forall i\neq j\,, \tag{2.1}$$

where t_f is the final time. When the time-dependent Hamiltonian is initialized in the ground state of the first instantaneous Hamiltonian, the indexes i, j are $1, 0$, respectively. This means that for a system initialized in the ground state, the evolution has to last at least t_f, to be with high probability close to the instantaneous ground state of the Hamiltonian calculated at time t_f.

The approximated condition 2.1 on the states can be generalized to another approximated and more recent criterion 2.2 which works for Hamiltonians, which have eigenstates with oscillating population (for example when the Hamiltonian have a driving term):

$$\frac{1}{t_f}\max_{s\in[0,1]} \frac{|\langle\varepsilon_i(s)|\partial_s H(s)|\varepsilon_j(s)\rangle|}{|\varepsilon_i(s)-\varepsilon_j(s)|^2} \ll 1\,, \qquad \forall i\neq j\,, \tag{2.2}$$

where $s = t/t_f$.

A rigorous version of the theorem states a possible upper bound on the time t_f, which for a generic Hamiltonian has at worst inverse cubic spectral gap dependence:

$$t_f \gg \max\left\{\max_{s\in[0,1]}\frac{\left\|H^{(2)}(s)\right\|}{\Delta^2(s)}, \max_{s\in[0,1]}\frac{\left\|H^{(1)}(s)\right\|}{\Delta^2(s)}, \max_{s\in[0,1]}\frac{\left\|H^{(1)}(s)\right\|^2}{\Delta^3(s)}\right\}, \tag{2.3}$$

where $\Delta \equiv \min_{s\in[0,1]}\Delta(s) = \min_{s\in[0,1]}\varepsilon_1(s)-\varepsilon_0(s)$ is the spectral gap, which for simplicity has been chosen between the first excited state and the ground state. The norm is the operator norm.

There exists a rigorous version of the theorem, that "sacrifices" the generic Hamiltonian to get inverse square energy gap dependence as obtained in the approximated version of the theorem (Equations 2.1 and 2.2):

$$t_f \geq \frac{K}{\Delta^2}|\ln(\Delta/h)|^{6\alpha}, \tag{2.4}$$

where $K > 0$, $\alpha > 1$ and $\Delta \ll h \equiv \|H(0)\| = \|H(1)\|$. In this case the Hamiltonian must be bounded, infinitely differentiable and in the Gevrey class G^α.

To get an exponentially small error in the bound of t_f, the Hamiltonian must satisfy again a Gevrey condition and the bound gets inverse cubic gap dependent ($t_f \gg \frac{C^2}{\Delta^3}$). An alternative version to obtain similar results to the approximated criterion (Equations 2.1 and 2.2) is estimated expressing t_f as a function of the path $L = \int_0^1 \||\partial_s\varepsilon(s)\rangle\|$, $t_f > O(L/\Delta)$. L is the distance covered by an eigenstate $|\varepsilon\rangle$ over the schedule $s(t)$ used to parameterize the Hamiltonian $H[s(t)]$. From $L \geq \max_s \|\partial_s H(s)\|/\Delta$ follows:

$$t_f \sim O(\max_s \|\partial_s H(s)\|/\Delta^2). \tag{2.5}$$

These results show the inverse dependence between spectral energy gap and run time. When the spectral energy gap is large, the upper bound on the run time is low. In this case, the evolution can be faster. In fact, since the spectral energy gap is large, there is also a lower probability for the system to jump to higher eigenstates.

B. Costs of AQC

The costs of using adiabatic quantum computing depends on the upper bound of the run time t_f (predicted by the adiabatic theorem) and on the energy scale of the time-dependent Hamiltonian:

$$cost = t_f \max_s \|H(s)\| . \tag{2.6}$$

Given the energy gap dependence of the upper bound of t_f, the cost of running AQC depends on the energy gap, which is for certain Hamiltonian difficult to estimate. One way to decrease the cost is to speedup the evolution interval (lower run time) amplifying the energy gap (Section III B).

III. "UNIVERSALITY" OF AQC AND HAMILTONIAN COMPLEXITY THEORY

A. Equivalence to circuit model

Definition 1 (Universal Adiabatic Quantum Computation)**.** *"A time-dependent Hamiltonian $H(t)$, $t \in [0, t_f]$, is universal for AQC if, given an arbitrary quantum circuit U operating on an arbitrary initial state $|\psi\rangle$ on n p-state particles and having depth L, the ground state of $H(t_f)$ is equal to $U|\psi\rangle$ with probability grater than $\epsilon > 0$, the number of particles on which $H(t)$ operates on is poly(n) $\forall t$, and $t_f = poly(n, L)$" [1].*

There exist various method to define a Hamiltonian, which implements the circuit model. At the same time, AQC can be implemented within the circuit model. The equivalence between AQC and circuit model (which is universal) makes AQC universal too. The two models simulate each other up to a polynomial overhead between depth (CM) and run time (AQC).

1. *From CM to AQC*

The circuit model can simulate AQC, because it can approximate the unitary operator $U(t_f, 0) = Texp[-i \int_0^{t_f} dt H(t)]$, which describes the evolution of the system where $H(t) = (1-t/t_f)H_0 + t/t_f H_1$ is the time-dependent Hamiltonian. H_0 and H_1 can be written with few-qubits unitaries choosing $H_0 = -\sum_i \sigma_i^x$ and H_1 to be 2-local in order to simplify further approximations on the unitary operator $U(t_f, 0)$. To approximate the unitary operator, which the circuit model can implement, one first needs 2 approximation.

The first approximation consists into writing $U(t_f, 0)$ as the product of M unitaries $U'(t_f, 0) = \Pi_{m=1}^M e^{-i\Delta t H'_m}$ without time-independent Hamiltonian ($H_m = H(m\Delta t)$). The error of this approximation over the Hamiltonians, bounded by $\frac{1}{M}\|H_1 - H_0\| \in \mathcal{O}(poly(n)/M)$, leads to the unitary operator approximation error $\mathcal{O}(\sqrt{t_f poly(n)/M})$.

The second approximation consists into applying Baker-Campbell-Hausdorff formula and neglecting the commutator term: $\Pi_{m=1}^M e^{-i\Delta t[(1-\frac{m\Delta t}{t_f})H_0 + \frac{m\Delta t}{t_f}H_1]} \mapsto \Pi_{m=1}^M e^{-i\Delta t(1-\frac{m\Delta t}{t_f})H_0} e^{-i\Delta t \frac{m\Delta t}{t_f}H_1}$. The error of this approximation is $\mathcal{O}(\frac{t_f^2}{M^2}\|H_0 H_1\|)$.

Combining the 2 approximations:

$$U(t_f, 0) \mapsto \Pi_{m=1}^M e^{-i\Delta t H'_m} \mapsto \Pi_{m=1}^M e^{-i\Delta t(1-\frac{m\Delta t}{t_f})H_0} e^{-i\Delta t \frac{m\Delta t}{t_f}H_1}, \tag{3.1}$$

leads to the total error $\mathcal{O}(poly(n)t_f^2/M)$, where n is the system size, t_f the run time needed for AQC and M the number of unitaries. Therefore, the circuit model can simulate AQC with up to polynomial overhead.

2. *From AQC to CM*

To realize adiabatic simulation of a n-qubit quantum circuit the Hamiltonian of AQC must encode information about the initial state $|0...0\rangle$ and the unitary gates $U_1, ...U_L$ of the circuit. At the same time, the circuit has to be described with states $|\alpha(l)\rangle$ obtained after the application of the l-th gate. Let $|\alpha(0)\rangle = |0...0\rangle$. The l-th gate has a position within the circuit. Therefore, the state $|\alpha(l)\rangle$ is not enough to describe the circuit. One way to define a new state which encode both the state $|\alpha(l)\rangle$ and the position of the l-th gate is to use Feynman clock register:

$$|\gamma(l)\rangle \equiv |\alpha(l)\rangle \otimes |1^l 0^{L-l}\rangle_c, \tag{3.2}$$

where $|1^l 0^{L-l}\rangle_c$ is the state of the Feynman clock, which in this state is set after the l-th gate. The initial Hamiltonian is chosen to have $|\gamma(0)\rangle = |\alpha(0)\rangle \otimes |0^L\rangle$ as its ground state with energy 0.

The ground state $|\eta\rangle$ of the final Hamiltonian, which is supposed to encode the circuit, is defined as a normalized superposition of the states $|\gamma(l)\rangle$:

$$|\eta\rangle = \frac{1}{\sqrt{L+1}} \sum_{l=0}^{L} |\gamma(l)\rangle. \tag{3.3}$$

The final Hamiltonian is defined so that $|\eta\rangle$ is its ground state with energy 0.

The time-independent Hamiltonian of AQC is

$$H(s) = (1-s)H_{init} + sH_{final} = H_{input} + H_c + (1-s)H_{c-init} + \frac{s}{2}H_{circuit}, \tag{3.4}$$

where

$$H_{init} = H_{c-init} + H_{input} + H_c = |1_1\rangle_c\langle 1_1| + \sum_{i=1}^{n} |1_i\rangle\langle 1_i| \otimes |0_1\rangle_c\langle 0_1| + \sum_{l=1}^{L-1} |0_l 1_{l+1}\rangle_c\langle 0_l 1_{l+1}|, \tag{3.5}$$

$$H_{final} = \frac{1}{2}H_{circuit} + H_{input} + H_c = \sum_{l=1}^{L-1} H_l + \sum_{i=1}^{n} |1_i\rangle\langle 1_i| \otimes |0_1\rangle_c\langle 0_1| + \sum_{l=1}^{L-1} |0_l 1_{l+1}\rangle_c\langle 0_l 1_{l+1}|. \tag{3.6}$$

The term $H_c = \sum_{l=1}^{L-1} |0_l 1_{l+1}\rangle_c\langle 0_l 1_{l+1}|$ is used to shift illegal Feynman clock states to energy equal 1, instead of energy equal 0 like for legal clock state. Illegal clock state are those states which contains the sequence 01. The term $H_{c-init} = |1_1\rangle_c\langle 1_1|$ forces the first initial Feynman clock to be $|0^L\rangle_c$. Applying the term $H_{input} = \sum_{i=1}^{n} |1_i\rangle\langle 1_i| \otimes |0_1\rangle_c\langle 0_1|$ to the state $|0^L\rangle_c$ leads to the state $|0^n\rangle$. The last term H_l is chosen to mimic the unitary operation U_l generated by the l-th gate:

$$\begin{aligned} H_1 &= \mathbb{1} \otimes |0_1 0_2\rangle_c\langle 0_1 0_2| - U_1|1_1 0_2\rangle_c\langle 0_1 0_2| \\ &\quad - U_1^\dagger|0_1 0_2\rangle_c\langle 1_1 0_2| + \mathbb{1} \otimes |1_1 0_2\rangle_c\langle 1_1 0_2|, \end{aligned} \tag{3.7}$$

$$\begin{aligned} H_{2\le\ell\le L-1} &= \mathbb{1} \otimes |1_{\ell-1} 0_\ell 0_{\ell+1}\rangle_c\langle 1_{\ell-1} 0_\ell 0_{\ell+1}| \\ &\quad - U_\ell|1_{\ell-1} 1_\ell 0_{\ell+1}\rangle_c\langle 1_{\ell-1} 0_\ell 0_{\ell+1}| \\ &\quad - U_\ell^\dagger|1_{\ell-1} 0_\ell 0_{\ell+1}\rangle_c\langle 1_{\ell-1} 1_\ell 0_{\ell+1}| \\ &\quad + \mathbb{1} \otimes |1_{\ell-1} 1_\ell 0_{\ell+1}\rangle_c\langle 1_{\ell-1} 1_\ell 0_{\ell+1}|, \end{aligned} \tag{3.8}$$

$$\begin{aligned} H_L &= \mathbb{1} \otimes |1_{L-1} 0_L\rangle_c\langle 1_{L-1} 0_L| \\ &\quad - U_L|1_{L-1} 1_L\rangle_c\langle 1_{L-1} 0_L| \\ &\quad - U_1^\dagger|1_{L-1} 0_L\rangle\langle 1_{L-1} 1_L| \\ &\quad + \mathbb{1} \otimes |1_{L-1} 1_L\rangle_c\langle 1_{L-1} 1_L|. \end{aligned} \tag{3.9}$$

The efficiency of using this construction to produce a Hamiltonian, which realizes a circuit, is measured finding t_f. One can find that t_f scales as a polynomial in L. However, this method to simulate a circuit in AQC does not necessarily lead to the same quantum speedup, which is obtained in the circuit model. One example is the adiabatic Grover algorithm discussed in Section IV A. In fact, the choice of the schedule can still slow down the evolution and increase the cost of AQC.

B. Universality limits

As mentioned in Section II B the cost of running AQC depends on the energy gap between ground state and first excited state. For some application it is hard to estimate this gap and evaluate whether it worth to carry out AQC. In certain cases the gap is too large. As a result, the time needed for AQC and its cost would be too large. Most of the adiabatic algorithms implemented with AQC do not lead to speedup over classical results. One way to improve this situations is making the energy gap larger to obtain a lower upper bound on the time. One way is adiabatic gap amplification, which leads to a quadratic energy gap amplification. Such amplification requires frustration free Hamiltonians,"a sum over positive semi-definite operators $H = \sum_{k=1}^{L} a_k \Pi_k$, with $a_k \in [0,1]$ and $L = \text{polylog}(N)$" [1]. In the context of quantum circuit with Q gates unitary operators, L is polynomial in Q. The projector Π_k describe the interaction between spins of nearest particles. This model makes the adiabatic evolution

time polynomial in Q. To observe the quadratic square amplification, the Hamiltonian $H(s)$ must be modified to the Hamiltonian $\bar{H}(s)$ as

$$H(s) = \sum_{k=1}^{L} a_k \Pi_k(s) \qquad \bar{H}(s) = \sum_{k=1}^{L} \sqrt{a_k} \Pi_k(s) \otimes (|k\rangle\langle 0| + |0\rangle\langle k|) \tag{3.10}$$

with $s \in [0,1]$ and ancillas $|0\rangle, |k\rangle$. As a consequence, the eigenvalues $\{\lambda_j\}$ for $j > 1$ and the ground state $|\psi(s)\rangle$ of $H(s)$ lead to the eigenvalues $\{\pm\sqrt{\lambda_j}\}$ for $j > 1$ and the degenerate ground state $|\psi(s)\rangle|10\ldots0\rangle$ of $\bar{H}(s)$, while the ground state eigenvalue remains 0.

Note that the Hamiltonian $\bar{H}(s)$ can be further simplified to a single particle subspace (writing the ancilla $|k\rangle \mapsto |0\ldots010\ldots0\rangle$): $\bar{H}(s) = \sum_{k=1}^{L} \sqrt{a_k}\Pi_k(s) \otimes (\sigma_k^+\sigma_0^- + \sigma_k^-\sigma_0^+)$ with Pauli operators $\sigma^{\pm} = (\sigma_x \pm i\sigma_y)/2$. Furthermore, adjusting the Hamiltonian adding penalty terms $Z_0 = \frac{1}{4}\sqrt{\Delta}(\mathbb{1} + \sigma_0^z)$, $Z = (L-2)\mathbb{1} - \sum_{k=0}^{L}\sigma_k^z$ allows to remove degeneracy of the ground state

$$H'(s) = \frac{1}{L^{1/d}}\left[\sum_{k=1}^{L}\sqrt{a_k}\Pi_k(s) \otimes (|k\rangle\langle 0| + |0\rangle\langle k|) + \frac{1}{4}\sqrt{\Delta}(\mathbb{1} + \sigma_0^z)\right] + Z.$$

C. Hamiltonian complexity theory

AQC's approach to find a solution can be used to solve some decision problems too. For example the k-local Hamiltonian problem and the k-SAT problem.

The k-local Hamiltonian problem decides whether the smallest eigenvalue of $H = \sum_{i=1}^{r} H_i$ (with $r = \text{poly}(n)$, $\|H_i\| = \text{poly}(n)$) is smaller than the threshold a, or all the eigenvalues are larger than the threshold b, where inputs a and b are chosen so that $b - a > \frac{1}{\text{poly}(n)}$.

This problem can be hard to solve depending on the Hamiltonian, sometimes even for quantum computers.

AQC can be used to solve the k-local Hamiltonian problem, defining the time-dependent Hamiltonian of AQC with the k-local Hamiltonian as final Hamiltonian. Once the ground state is found, one can compute the energy eigenvalue of the ground state of the k-local Hamiltonian of the problem and observe whether is smaller or bigger than a and b.

The k-SAT problem consists into finding whether there exists literals $X = (x_1, \ldots, x_n)$ such that $\Phi(X) = 1$ where $\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_r$ is a Boolean formula with clauses $C_i = x_{i_1} \vee x_{i_2} \vee \cdots \vee x_{i_k}$ and $x_i \in \{0,1\}$. Solving the 3-local Hamiltonian problem with AQC leads to the solution of the 3-SAT problem, since it can be mapped to the 3-local Hamiltonian problem. Applying the 3-local Hamiltonian to the state $|X\rangle = |x_{i_1}, x_{i_2}, x_{i_3}\rangle$, each 3-local projectors H_i is chosen to give eigenvalue 0 and eigenvalue 1 for satisfying and unsatisfying assignments, respectively. Therefore, when the 3-local Hamiltonian applied to the state $|X\rangle$ leads to an eigenvalue larger than 0 the 3-SAT problem is unsatisfiable.

1. *Decision problems classes*

As reported in [1] a decision problem Q is in the class of NP problem iff there exists a verifier V so that for all inputs η:

- "if $Q(\eta) = 1$, then there exists a witness X such that $V(\eta, X) = 1$"[1],
- "if $Q(\eta) = 0$, then for all witnesses X we have $V(\eta, X) = 0$"[1].

The problem Q is NP-complete when it is in NP and every problem problem in NP can be solved using an algorithm that solves for the problem Q. When this reduction is in polynomial time, the problem Q is NP-hard.

The class MA (Merlin Arthur) is the probabilistic version of NP. The verifier V must fullfil probabilstic conditions $Pr(V(\eta, X) = 1) \leq \frac{2}{3}$ and $Pr(V(\eta, X) = 1) \geq \frac{1}{3}$ instead of $V(\eta, X) = 1$ and $V(\eta, X) = 0$, where the probabilities do not have to be necessarily $(2/3, 1/3)$.

The class QMA is the quantum version of MA, that works with quantum computers. In quantum computers the MA witness X becomes a state $|X\rangle \in (\mathbb{C}^2)^{\otimes \text{poly}(|\eta|)}$. A decision problem Q is in the class QMA iff there exists a verifier algorithm (efficient: polynomial time) V so that for all inputs η:

- "if $Q(\eta) = 1$, then there exists a witness $|X\rangle$ such that $\Pr(V(\eta, |X\rangle) = 1) \geq \frac{2}{3}$ (completeness)"[1],
- if $Q(\eta) = 0$, then for all witnesses $|X\rangle$ we have $\Pr(V(\eta, |X\rangle) = 1) \leq \frac{1}{3}$ (soundness)"[1].

According to [1], AQC can be used to solve some decision problems, working as the verifier algorithm. At the same time, the study of Hamiltonian complexity theory, especially QMA-completeness helps to understand universality of AQC.

IV. THEORY OF AQC APPLYED TO ALGORITHMS

A. Grover algorithm

Unstructure search consists into finding a specific item x_0 in the unsorted set $\{1,..,N\}$ such that $f(x_0)=1$ only for x_0 where $f:\{1,...,N\}\to\{0,1\}$. The problem of unstructure search can be solved classically by applying f to every element of the set $\{1,..,N\}$. This is realized by querying the research to an oracle, which can identify x_0, until the oracle finds x_0. The solution provided by Grover algorithm's for quantum computers allowed to save time and reach up to a quadratic speedup over the classical computer result. The approach with circuit model consists also into querying the initially prepared state through a combination of phase oracles and Hadamard gates, but only up to $\sqrt{N}$ times. It shows the quantum speedup $\mathcal{O}(\sqrt{N})$ of the circuit model.

The adiabatic Grover's algorithm can reach the same speedup of the circuit model version. To simplify the problem the Hamiltonian is defined using $s=t/t_f$ so that $s\in[0,1]$. The time-dependent Hamiltonian $H(s)=(1-A(s))H_0+A(s)H_1$ and its schedule $A(s)$ are chosen to encode the problem at the beginning with H_0 and to find the solution at the end with H_1. The choice of $A(s)$ affects the speedup of the algorithm. The first attempt is to set $A(s)=s$. As a consequence, H_0 is the initial Hamiltonian and H_1 is the final Hamiltonian.

The initial Hamiltonian $H_0=\mathbb{1}-|\phi\rangle\langle\phi|$ is defined using the state $|\phi\rangle=\frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}|i\rangle=|+\rangle^{\otimes n}$ with $\langle\phi|\phi\rangle=1$. Because of this choice, those states $|\psi_j\rangle$ orthogonal to the state $|\phi\rangle$ have eigenvalue $\lambda_j=1$

Eigenstates of H_0 are $|\psi_0\rangle=|\phi\rangle$ and $|\psi_j\rangle$ with eigenvalues $\lambda_0=0$ and $\lambda_j=1$, respectively.

The final Hamiltonian $H_1=\mathbb{1}-|m\rangle\langle m|$ is defined using the state $|m\rangle$ of the binary representation m of the specific item x_0. Eigenstates of H_1 are $|\psi_0\rangle=|m\rangle$ and $|\psi_j\rangle$ with eigenvalues $\lambda_0=0$ and $\lambda_j=1$, respectively. The state $|\phi\rangle$ can be written as a sum of the $|m\rangle$ state and those states orthogonal to $|m\rangle$:

$$|\phi\rangle=\frac{\sqrt{N-1}}{\sqrt{N}}\frac{1}{\sqrt{N-1}}\sum_{i=0,i\neq m}^{N-1}|i\rangle+\frac{1}{\sqrt{N}}|m\rangle=\frac{\sqrt{N-1}}{\sqrt{N}}|m^\perp\rangle+\frac{1}{\sqrt{N}}|m\rangle, \tag{4.1}$$

where $|m^\perp\rangle=\frac{1}{\sqrt{N-1}}\sum_{i=0,i\neq m}^{N-1}|i\rangle$.

Since $|\phi\rangle$ can be written with $|m\rangle$ and $|m^\perp\rangle$, also the time-dependent Hamiltonian $H(s)$ can be written in the two-dimensional subspace span$\{|m\rangle,|m^\perp\rangle\}$:

$$[H(s)]_{|m\rangle,|m^\perp\rangle}=\frac{1}{2}\mathbb{1}_{2\times2}-\frac{\Delta(s)}{2}\begin{pmatrix}\cos\theta(s) & \sin\theta(s)\\ \sin\theta(s) & -\cos\theta(s)\end{pmatrix}, \tag{4.2}$$

where $\Delta(s)=\sqrt{(1-2s)^2+\frac{4}{N}s(1-s)}$, $\cos\theta(s)=\frac{1}{\Delta(s)}[1-2(1-s)(1-\frac{1}{N})]$, $\sin\theta(s)=\frac{2}{\Delta(s)}(1-s)\frac{1}{\sqrt{N}}\sqrt{1-\frac{1}{N}}$. Equation 4.2 shows that $H(s)$ have only 2 eigenvalues $\epsilon_0=\frac{1}{2}(1-\Delta(s))$ and $\epsilon_1(s)=\frac{1}{2}(1+\Delta(s))$ different from 1. The other N-2 eigenstates of the system have eigenvalue $\lambda=1$.

The highest lower bound on the time t_f is measured at time $s=1/2$ where the distance $\Delta_{min}=\epsilon_0-\epsilon_1=\frac{1}{\sqrt{N}}=2^{-n/2}$ becomes minimal. The adiabatic condition does not lead to a speedup over the classical computation. To obtain the same speedup of the circuit model, one must slow down the speed of the schedule $A(s)$ only where the energy gap is minimal ($s=1/2$), not necessarily over the entire interval. In fact, where the gap is smaller, it is more probable that the actual state of the time-dependent Hamiltonian gets promoted to the higher state and evolve in the wrong energy level (level which at the end of the evolution is not close to the level containing the solution of the problem). One way to find the quadratic speedup is to choose the Ansatz

$$\partial_s A=c\Delta^p[A(s)] \qquad c=\int_0^1\Delta^{-p}[A(s)]\partial_s A ds=\int_{A(0)}^{A(1)}\Delta^{-p}(u)du, p, c>0, \tag{4.3}$$

so that $A(0)=0$, $A(1)=1$. As a consequence, the rigorous adiabatic condition for generic Hamiltonians leads to the condition $t_f\gg2\pi\sqrt{N}[1+\log(2N)]$. Carrying out he computation with $p=2$ leads to the quadratic speedup $t_f\to\frac{\pi}{2c'}\sqrt{N}$.

B. Deutsch-Josza algorithm

The Deutsch-Josza algorithm determines whether the function $f:\{0,1\}^n\mapsto\{0,1\}$ is either constant or balanced. While classically the Deutsch-Josza algorithm's job takes up to $2^{n-1}+1$ queries, the same

process in the circuit model needs only 1 query. The adiabatic Deutsch-Josza algorithm reaches the same quantum speedup of the circuit model over the classical algorithm.

The time-dependent Hamiltonian which interpolates between initial and final Hamiltonian can be defined either using linear interpolation (as for the adiabatic Grover algorithm) or using unitary interpolation.

In case of unitary interpolation, the evolution of the time-dependent Hamiltonian is

$$H(s) = \tilde{U}(s)H(0)\tilde{U}(s), \tag{4.4}$$

with $\tilde{U}(s) = \exp\left(i\frac{\pi}{2}sU\right)$, $\tilde{U}(1) = iU$. U is a unitary matrix. The initial Hamiltonian $H(0)$ is defined again as the projections on the state $\phi\rangle$, written $H(0) = \omega\sum_{i=1}^{n}|-\rangle_i\langle -|$ where ω is the scale. The unitary transformation U is the diagonal matrix $U = \mathrm{diag}[(-1)^{f(0)}, \cdots, (-1)^{f(2^n-1)}]$ which solves the Deutsch-Jozsa problem $U|x\rangle = (-1)^{f(x)}|x\rangle$ with $x \in \{0,1\}^n$. The final Hamiltonian $H(1)$ is defined to encode the final solution in its ground state $|\psi(1)\rangle = U|\psi(0)\rangle$, which is the case when $H(1) = UH(0)U^\dagger$.

Applying the adiabatic theorem condition leads to the time interval $t_f \gg 1/\omega$. Since, the bound on the time t_f is independent of n, the run time of the algorithm is $\mathcal{O}(1)$.

C. Bernstein-Vazirani algorithm

The Bernstein-Vazirani algorithm tells whether the number of 1 bits in common between $w \in \{0,1\}^n$ and the unknown string $a \in \{0,1\}^n$ is even ($f_a(w) = w \odot a = 0$) or odd ($f_a(w) = w \odot a = 1$). In the basis $\{|+\rangle, |-\rangle\}$, the state of the function can be written as $|f_a(w)\rangle = \frac{1}{\sqrt{2}}\left(|+\rangle + (-1)^{f_a(w)}|-\rangle\right)$. Classically, solving this problem would require to check every bit of the string, which would take n queries in total. The quantum solution of both circuit model (depth) and AQC (time) shows polynomial speedup $\mathcal{O}(1)$ over the classical approach. The adiabatic Bernstein-Vazirani algorithm's time dependent Hamiltonian is different from the adiabatic Grover Hamiltonian and the adiabatic Deutsch-Josza Hamiltonian since it acts on two separated subsystems A (n qubit) and B (1 qubit). To solve the problem, the goal is to find a state (in the subsystem A), which encodes all the bits of a. The time-dependent Hamiltonian:

$$H(s) = (1-s)H_0 + sH_1 = \sum_{w\in\{0,1\}^n}|w\rangle_A\langle w| \otimes H_w(s), \tag{4.5}$$

with $H_w(s) = \frac{1-s}{2}\left(\mathbb{1}_B - \sigma_B^x\right) - \frac{s}{2}\left(\mathbb{1}_B + (-1)^{f_a(w)}\sigma_B^z\right)$ interpolates linearly between the initial Hamiltonian H_0 and the final Hamiltonian H_1:

$$H_0 = \frac{1}{2}\left(\mathbb{1}_A \otimes (\mathbb{1}_B - \sigma_B^x)\right) = \frac{1}{2}\sum_{w\in\{0,1\}^n}|w\rangle_A\langle w| \otimes \left(\mathbb{1}_B - \sigma_B^x\right),$$

$$H_1 = \sum_{w\in\{0,1\}^n} h_w = \sum_{w\in\{0,1\}^n} -\frac{1}{2}|w\rangle_A\langle w| \otimes \left(\mathbb{1}_B + (-1)^{f_a(w)}\sigma_B^z\right). \tag{4.6}$$

The ground state of the initial Hamiltonian $|\psi(0)\rangle = \sum_{w\in\{0,1\}^n} c_w|w\rangle_A \otimes |+\rangle_B$ with $c_w = 2^{-n/2}\forall w$ has eigenvalue 0. The adiabatic evolved state is:

$$|\Psi(t_f)\rangle = \frac{1}{2^{n/2}}\sum_{w\in\{0,1\}^n}|w\rangle_A \otimes \exp\left[-it_f\int_0^1 \varepsilon_{0,w}(s)ds\right]|f_a(w)\rangle_B, \tag{4.7}$$

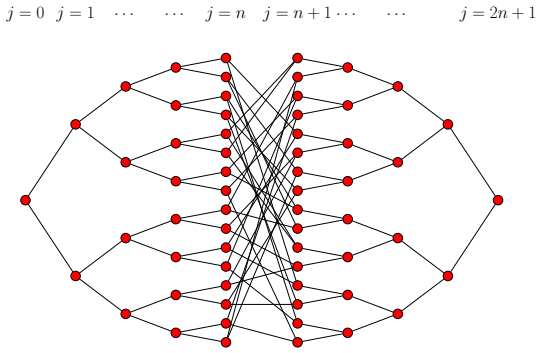
with $\varepsilon_{0,w}(s)$ the instantaneous ground state energy of $H_w(s)$. The final state is obtained evolving only the subsystem B, which contains only one qubit. As a result, the evolution is $\mathcal{O}(1)$. Writing the state $|f_a(w)\rangle$ in the $\{|+\rangle, |-\rangle\}$ basis, splits the final state$|\Psi(t_f)\rangle$ into two states $|\Psi_+\rangle, |\Psi_-\rangle$ with the same probability and eigenvalues $1, -1$:

$$|\Psi_+\rangle = \frac{1}{2^{n/2}}\sum_{w\in\{0,1\}^n}|w\rangle_A \otimes |+\rangle_B = |\Psi(0)\rangle, \tag{4.8}$$

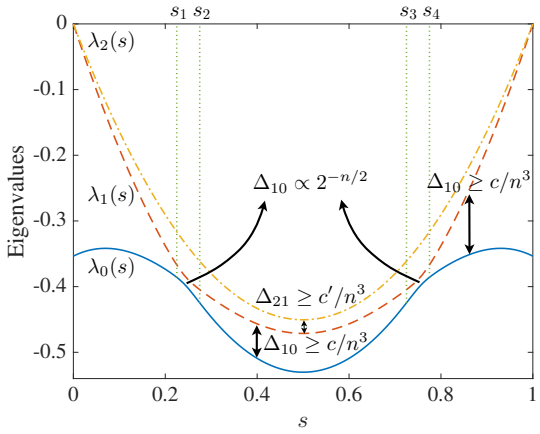
$$|\Psi_-\rangle = \frac{1}{2^{n/2}}\sum_{w\in\{0,1\}^n}|w\rangle_A \otimes (-1)^{f_a(w)}|-\rangle_B = \frac{1}{2^{n/2}}\bigotimes_{k=0}^{n-1}\left(|0\rangle_k + (-1)^{a_k}|1\rangle_k\right)_A \otimes |-\rangle_B, \tag{4.9}$$

using $\sum_{w\in\{0,1\}^n}(-1)^{f_a(w)}|w\rangle_A = \bigotimes_{k=0}^{n-1}\left(|0\rangle_k + (-1)^{a_k}|1\rangle_k\right)_A$. Therefore, when the measured eigenvalue is -1, the final state encode the bits of a, because k-th qubits $|+\rangle_k \Rightarrow a_k = 0$ and $|-\rangle_k \Rightarrow a_k = 1$. When the measured eigenvalue is $+1$ the measurement has to be repeated. This result highlights that exists a probability of failing the measurement. After m tries, the probability to succeed is 2^{-m}.

D. Glued trees problem



(a) Two binary trees with the last symmetric columns of leaves randomly connected with two vertices of the opposite column. From [1].



(b) Evolution of the first three states' eigenvalues and their gap to each other with time s. From [1].

Figure 1: The glued tree problem sketch and eigenvalues evolution.

The glued tree problem's goal consists into finding the name of one origin vertex following the shortest path which connects the origin vertices of two binary trees connected at random. Figure 1 shows an example where the depth n of the binary trees is $n = 4$. The red dots consists into the vertices (in total $N = 2^{n+2} - 2$), which are labelled by randomly chosen 2n-bit string. The two columns of leaves between the trees are connected randomly with two vertices of the opposite column. The difficulty of the problem consists into finding the path knowing only the adjacent vertices of the vertex which is being considered. Using an oracle, accessing this oracle would give the output of the next vertex. Classically, to solve the glued tree problem would require a subexponential in n number of oracle calls. The adiabatic quantum solution to the problem can be find in polynomial time ($t_f \approx n^6$). Let the initial and final Hamiltonian:

$$H_0 = -|a_0\rangle\langle a_0| \qquad H_1 = -|a_{N-1}\rangle\langle a_{N-1}|, \tag{4.10}$$

where a_0 and a_{N-1} are the bit strings of the origin vertices of the trees. The states to the Hamiltonian are $|c_j\rangle = \frac{1}{\sqrt{N_j}} \sum_{i \in j-\text{th column}} |a_i\rangle$ with $N_j = 2^j$ for $0 \leq j \leq n$ and $N_j = 2^{2n+1-j}$ for $n+1 \leq j \leq 2n+1$. The time-dependent Hamiltonian is:

$$H(s) = (1-s)\alpha H_0 - s(1-s)A + s\alpha H_1, \tag{4.11}$$

where A is another Hamiltonian defined as $\langle c_j|A|c_{j+1}\rangle = \sqrt{2}$ for $j = n$ otherwise equals 1, $\alpha \in (0, 1/2)$ is a constant and $s(t)$ the schedule, which is chosen to evolve adiabatic if the energy gap scales as $1/n^3$. The Hamiltonian keeps the evolution of the state in the subspace $\{c_j\}$. Figure 1b shows the energy of the ground state and the first excited state. Around the values $s_x = \alpha/\sqrt{2}$ and $1 - s_x$ the energy gap closes exponentially. Around these points there are two areas ($s \in [s_1, s_2)$, $s \in [s_3, s_4)$) where the energy gap scales as $1/e^n$, that makes the evolution non-adiabatic since the states will transit to the first excited state with high probability. In case $s \in [0, s_1) \cup [s_2, s_3) \cup [s_4, 1)$, the evolution stays in the same state, since the evolution is sufficiently adiabatic, because the spectral energy gap is larger.

E. Adiabatic PageRank algorithm

Adiabatic PageRank algorithm's goal is to prepare a state replicating the PageRank vector $\vec{p}$ through adiabatic quantum computation. The algorithm combined with other quantum protocols can reach a regular quantum speedup over the classical version, but without this combination the adiabatic algorithm would not be more efficient than the classical version. The PageRank vector $\vec{p}$ (eigenvector of the Google matrix with eigenvalue 1) can be computed with power method which converges, since the Google matrix G is upper bounded:

$$\vec{p} = \lim_{k \mapsto \infty} G^k \vec{p}_0 \qquad G := \alpha P_2^T + (1-\alpha)E. \tag{4.12}$$

The vector $\vec{p}_0$ could be any probability vector. The variable $E \equiv |\vec{v}\rangle\langle\vec{e}|$ is computed for example with vector $\vec{v} = \vec{e}/n$ where n is the size of the web-graph (number of nodes). The probability parameter

$\alpha \in (0,1)$ describes the movement of the walker on the web-graph, whether it follows the nodes structures or it hopes. The matrix P_2 is a stochastic matrix, defined as a modified version of the transition matrix P_1:

$$P_1(i,j) = \begin{cases} 1/d(i) & \text{if } (i,j) \text{ is an edge of the directed adjacent matrix;} \\ 0 & \text{else,} \end{cases} \tag{4.13}$$

with $d(i)$ is the out-degree of the ith node. The P_1 rows having zero entry elements gets replaced in P_2 by the vector $\vec{e}/n$, whose entries are $1/n$. The initial h_0 and final Hamiltonians h_1 to realize the adiabatic algorithm are defined as non-local Hamiltonians (notation h):

$$h_1 = h(G) \equiv (\mathbb{1} - G)^\dagger(\mathbb{1} - G) \qquad\qquad h_0 = h(G_c) \equiv (\mathbb{1} - G_c)^\dagger(\mathbb{1} - G_c), \tag{4.14}$$

where G is a generic Google matrix and G_c is the Google matrix associated with the complete graph. The ground state of the final Hamiltonian is $|\pi\rangle \equiv \vec{p}/\|\vec{p}\|$, because 1 is the maximal eigenvalue of G and its eigenvector is the PageRank vector $\vec{p}$. The ground state of the initial Hamiltonian h_0 is $|\psi(0)\rangle = \sum_{j=1}^{n} |j\rangle/\sqrt{n}$. The time-dependent Hamiltonian is

$$h(s) = (1-s)h_0 + sh_1. \tag{4.15}$$

This adiabtic evoulution on random web-graph takes the run time $t_f \sim (\log\log n)^{b-1}(\log n)^b$ where $b > 0$.

V. STOQUASTIC AQC

Stoquastic adiabatic quantum computation (StoqAQC) is AQC with k-local stoquastic Hamiltonians, which are Hamiltonians characterized by real nonpositive off-diagonal matrix elements (with respect to a certain basis). In case of the computational basis, the Hamiltonian H would be stochastic iff $\langle x|H|x'\rangle \leq 0 \quad \forall x, x' \in \{0,1\}^n \quad x \neq x'$. In order to restrict stoquasticity within this section to the computational basis, the basis is chosen so that the initial Hamiltonian always carries a minus sign. Furthermore, it is possible to generalize the definition of classes such as MA to the stoquastic version StoqMA, where the verifier is stoquastic and allowed to do the final measurement also in the basis $\{|+\rangle, |-\rangle\}$. Like in the case of non-stoquastic AQC with the problem of the k-local Hamiltonian, also StoqAQC gives rise to a set of problems, the StoqAQCEval characterized by:

- "a continuous family of ($k \geq 2$)-local (k fixed) stoquastic Hamiltonians $H(s) = \sum_{i=1}^{r} H_i(s)$ with $r = \text{poly}(n)$ and parameterized by $s \in [0,1]$. For all i and all s, the non-zero entries of $H_i(s)$ are specified by poly(n) bits of precision, and $\|H_i(s)\| = \text{poly}(n)$. The ground state energy gap $\Delta[H(s)]$ satisfies $\Delta[H(s)] \geq 1/\text{poly}(n)$ for all s"[1],
- "two real numbers a and b specified with poly(n) bits of precision, and $b - a > 1/\text{poly}(n)$"[1].

Since the k-local stoquastic Hamiltonian for $k \geq 2$ is StoqMA-complete, StoqAQCEval is in StoqMA. Those problem which are reducible in polynomial time to StoqAQCEval (BStoqP) are in StoqMA.

A. Simulation of stoquastic Hamiltonians

There are evidences, which show that stoqAQC might not be as powerful as universal quantum computation. At the same time there is not a theorem, which rules out a potential quantum speedup of StoqAQC. Two classical competitors are Path Integral Quantum Monte Carlo (PIQMC) and diffusion Quantum Monte Carlo (D-QMC).

The first issue dealing with universality starts from the fact that the ground state of a stoquastic Hamiltonian can be expressed as a linear combination with nonnegative amplitudes. As a consequence, the ground state can be described with a classical probability distribution, which suggests the limits of finding a quantum speedup. Moreover, stoquastic k-local Hamiltonian is not QMA-complete. Furthermore, gapped StoqAQC can be simulated in PostBPP (can post select on some subset). "It suffices to call an oracle for problems in PostBPP a polynomial number of times to efficiently sample from the ground state of a gapped stoquastic Hamiltonian"[1]. As a consequence, assuming that StoqAQC could realize universal quantum computation, would mean to be able to classically simulate quantum algorithms in polynomial time, which leads to the impossible conclusion PostBPP=PostBQP.

There are examples where PI-QMC can not find the solution as fast as StoqAQC or it can not find the solution at all due to topological obstructions. The first example is a sombrero-like potential for a single particle with minimal radius r. The trajectories in imaginary time of the particle lies around the circle of radius r. They lies with low probability away from the circle, because of the steep potential. As a result,

it takes several winding (exponential time) to shift winding number sector and PI-QMC won't be able to equilibrate. On the other hand, the adiabatic evolution requires polynomial time. In the "bouquet of circles" example PI-QMC do not necessarily converge and finding the ground state with this method is less probable.

Using certain non-topological obstructions, for example the discrepancy between L_1 and L_2-normalization, it is possible to simulate efficiently stoquastic adiabatic processes (polynomial time) while the D-QMC simulations fail. D-QMC succeeds only allowing for exponential costs. An example consists in the Hamiltonian $H(s) = \frac{1}{n}[L + b(s)W] - c(s)P$ with W the Hamming wight operator, L the graph Laplacian and $P = |0\cdots 0\rangle\langle 0\cdots 0|$ which is equivalent to:

$$H(s) = -\frac{1}{n}\sum_{j=1}^{n}\left(X_j + \frac{1}{2}b(s)Z_j\right) - c(s)P. \tag{5.1}$$

The schedules are:

$$b(s) = \begin{cases} 2sb \\ b \end{cases} \quad c(s) = \begin{cases} 0 & s \in [0, 1/2) \\ (2s-1)c & s \in [1/2, 1] \end{cases} . \tag{5.2}$$

It follows that StoqAQC evolution converges to the ground state $|0\cdots 0\rangle$ in polynomial time, since the overall minimum gap at $s = 0$ is $2/n$.

Dropping the assumption that the stoquastic adiabatic evolution has to evolve in the ground state, it is possible to show that StoqAQC is universal. Universality of StoqAQC can be verified for excited state evolution. In the case of 3-local Hamiltonian reported in this section it is also QMA-complete. Consider the Hamiltonian:

$$H_{ZZXX} = \sum_i d_i X_i + h_i Z_i + \sum_{i\leq j} J^x_{ij} X_i X_j + J^z_{ij} Z_i Z_j, \tag{5.3}$$

with real coefficients d_i, h_i, J^x_{ij} and J^z_{ij}. Writing $H_{ZZXX} = -\sum_k \alpha_k T_k$ allows to eliminate the negative matrix entries.

The matrix operators $T_k \in \{\pm X_i, \pm Z_i, \pm X_i X_j, \pm Z_i Z_j\} \in 2^n \times 2^n$ ($\alpha_k > 0$) are symmetric and takes values $+1, -1, 0$. These matrices are generalized to the 2-local or 3-local operators (permutation matrices) $\tilde{T}_k \in 2^{n+1} \times 2^{n+1}$ acting on $n+1$ qubits by changing the values as

$$1 \to \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \; -1 \to \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \; 0 \to \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{5.4}$$

The new Hamiltonian acting on $n+1$ qubits:

$$\tilde{H}_{ZZXX} = -\sum_k \alpha_k \tilde{T}_k \tag{5.5}$$

is a (3-local) stoquastic Hamiltonian which is equivalent to $\tilde{H}_{ZZXX} = H_{ZZXX} \otimes |-\rangle\langle -| + \bar{H}_{ZZXX} \otimes |+\rangle\langle +|$ where $\bar{H}_{ZZXX} = -\sum_k \alpha_k |T_k|$.

The spectrum of $\tilde{H}_{ZZXX}$ separates into two sectors generated by the eigenstates $|\varepsilon\rangle$ of H_{ZZXX} ($\mathcal{L}_- = |\varepsilon_j\rangle \otimes |-\rangle$) and by the eigenstates $|\bar{\varepsilon}\rangle$ of $\bar{H}_{ZZXX}$ ($\mathcal{L}_+ = |\bar{\varepsilon}_j\rangle \otimes |+\rangle$).

Because there are no interactions changing the ancilla qubit from $|\pm\rangle$ to $|\mp\rangle$), the evolution in $\mathcal{L}_-$ remains there. The ground state in $\mathcal{L}_-$ is not necessarily the ground state of the full Hamiltonian.

VI. ADIABATIC GATE TELEPORTATION

Adiabatic computation can be implemented also combined with other models such as the circuit model and the holonomic model. For example, such a model can be a piecewise adiabatic evolution replicating a quantum circuit step by step. Using this model it could be possible to realize a universal quantum computer. To realize this type of Hamiltonians is necessary to understand quantum teleportation and adiabatic teleportation. Quantum teleportation consist into transferring a qubit state "between distant parties via an initial shared entangled state and two bits of classical communication"[2]. Adiabatic teleportation realizes the swap via adiabatic evolution using a control scheme. This process can be modelled by three qubits (first qubit to be swapped to the third qubit, second qubit necessary as control qubit). The idea is to adiabatically evolve a Hamiltonian with Bell state in the qubit 1 and 2 as ground state to a Hamiltonian which has the same bell state as ground state of qubit 2 and 3. The adiabatic evolution can be described by the time dependent Hamiltonian:

$$H(s) = (1-s)H_i + sH_f, \tag{6.1}$$

where s is the schedule and H_i,H_f are the initial and final Hamiltonians:

$$H_i = -\omega(X_2X_3 + Z_2Z_3) \qquad H_f = -\omega(X_1X_2 + Z_1Z_2) \tag{6.2}$$

with X, Z single-qubit Pauli matrices, ω the energy scale. The ground state are spanned by $|0\rangle \otimes |\phi\rangle$,$|1\rangle \otimes |\phi\rangle$ at the beginning and by $|\phi\rangle \otimes |0\rangle$, $|\phi\rangle \otimes |1\rangle$ at the end.

The time-dependent Hamiltonian can be written using logical qubit operators ($\bar{X}_1 = XXX, \bar{X}_2 = IXX, \bar{X}_3 = XXI, \bar{Z}_1 = ZZZ, \bar{Z}_2 = ZZI, \bar{Z}_3 = IZZ$):

$$H(s) = -\omega(1-s)(\bar{X}_2 + \bar{Z}_3) - \omega s(\bar{X}_3 + \bar{Z}_2). \tag{6.3}$$

Performing the evolution adiabatically would drag the eigenstates of $\bar{X}_2$, $\bar{Z}_3$ related to the eigenvalue $+1$ to the eigenstates of $\bar{Z}_2$, $\bar{X}_3$ with eigenvalue $+1$.

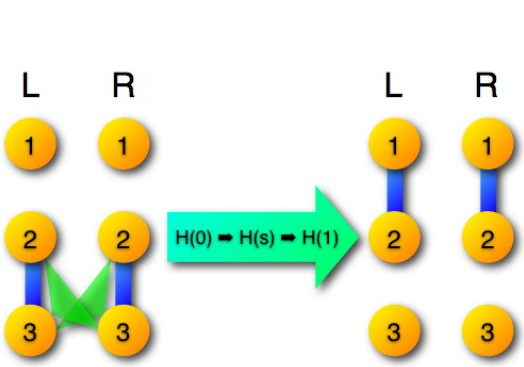
From the minimum energy gap $\sqrt{2}\omega$, it is possible to estimate that the minimal time needed by the adiabatic evolution to stay in the ground state is $T \gg O(\frac{1}{\omega})$.

In order to obtain universality the swapping performed in adiabatic quantum teleportation is combined with a single qubit gate implemented in the initial Hamiltonian to obtain Adiabatic Gate Teleportation (AGT). One way to realize single-qubit unitary is modifying the initial Hamiltonian according to $H_i' = U_3 H_i U_3$ where U_3 is a unitary operation which rotates the third qubit. For example, to realize a Hadamard gate the initial Hamiltonian is modified to $H_i' = -\omega(X_2Z_3 + Z_2X_3)$. One way to prepare these rotated Hamiltonians is Adiabatic Gate Preparation (AGP), which consists into adiabatically evolve the initial state of the form $H_i = -\omega(X_aX_b + Z_aZ_b)$ to the initial Hamiltonian $H_i' = U_b H_i U_b^\dagger$ for certain U_b. Matrices A and B that generate $SU(2)$ can be chosen as the U_b matrix to realize universal single qubit operations. Two possible choices of A and B are:

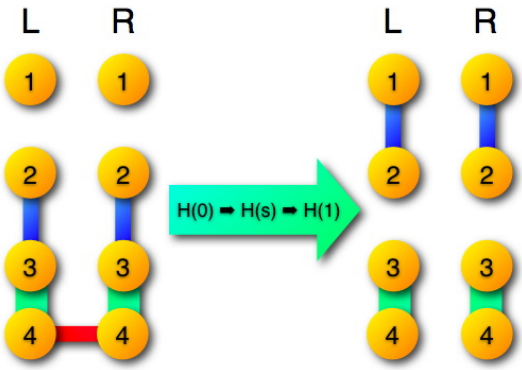
$$A = \frac{1}{2}\begin{pmatrix} 1+i\sqrt{2} & 1 \\ 1 & -1+i\sqrt{2} \end{pmatrix} \qquad B = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}. \tag{6.4}$$

The result can be generalized to two-qubit gates (the two-quibit gate is applied to two final output qubits). In case of controlled-phase (C_Z) between two logical qubits (between third and sixth physical qubit) the Hamiltonians are:

$$H_i = -\omega C_Z(X_2X_3 + Z_2Z_3 + X_5X_6 + Z_5Z_6)C_Z^\dagger \qquad H_f = -\omega(X_1X_2 + Z_1Z_2 + X_4X_5 + Z_4Z_5). \tag{6.5}$$



(a) Twice three qubits interaction (green) and two-body coupling interaction (blue). From [2].



(b) Two-body interaction (blue), strong coupling interaction in the encoded subspace (green) and coupling between logic qubits (red). From [2].

Figure 2: Sketches of the initial and final qubits involved in the adiabatic evolution process.

The challenging part of such a Hamiltonian is the interaction with 3 qubits, in fact $H_i = -\omega C_Z(X_2X_3 + Z_2Z_3 + X_5X_6 + Z_5Z_6)C_Z^\dagger = -\omega(X_2X_3Z_6 + Z_2Z_3 + Z_5Z_6 + Z_3X_5X_6)$. The Hamiltonian can be generalized to the ideal $H_{ideal} = -\lambda(X_2^L X_3^L Z_3^R + Z_2^L Z_3^L) + [L \leftrightarrow R]$ where $[L \leftrightarrow R]$ is the same with exchanged left and right qubits (See Figure 2a). To avoid three body interactions, "one replaces one of the qubits in a three qubit interaction by an encoded qubit across two qubit" [2]. To realize these interactions, it is necessary to add a fourth physical qubit for each logical qubit L and R (See Figure 2b). The new encoded subspace is realized by qubit 3 and 4 with encoded operators $\bar{X}_3 = X_3X_4$, $\bar{Z}_3 = Z_3$ or Z_4. The Hamiltonian written with the encoded operators is:

$$H_{target} = -\lambda(X_2^L \bar{X}_3^L \bar{Z}_3^R + Z_2^L \bar{Z}_3^L) - \omega(Z_3^L Z_4^L) + [L \leftrightarrow R]. \tag{6.6}$$

The term $\omega(Z_3^L Z_4^L)$ consists into strong coupling to keep the encoded term in the same subspace.

An alternative choice is to use two-body gadget Hamiltonians as in:

$$H_i = -\lambda(X_2^L X_3^L + X_4^L Z_4^R + Z_2^L Z_3^L) - \omega(Z_3^L Z_4^L) + [L \leftrightarrow R]. \tag{6.7}$$

Both H_i and H_{target} are evolved towards

$$H_f = -\lambda(X_1^L X_2^L + Z_1^L Z_2^L) - \omega(Z_3^L Z_4^L) + [L \leftrightarrow R]. \tag{6.8}$$

According to [2] the energy gap of the evolution is by bounded $\Delta E(s) \leq \omega r^2$ for $r\lambda/\omega < 0.5$.

AGT distinguishes itself from Holonomic Quantum Computing (HQC), since it also uses adiabatic evolution to produce a quantum gate but not as a cycle evolution like in HQC.

VII. SUMMERY AND CONCLUDING REMARKS

To conclude Adiabatic Quantum Computing can solve problems, which can be solved by the quantum circuit model. The AQC process speedup is polynomially close to the solution obtained by the circuit model for every algorithm. However, the limitations on certain Hamiltonians affect the speed of the evolution. Each algorithm deserves a separate analysis and discussion to improve its cost with AQC. For some of the fundamental algorithms of quantum information processing, the separate extra analysis shows that AQC reach the same speedup of the circuit model (See Section IV). In this sense, these field needs further research.

Moreover, AQC universality helps to understand QMA-completeness and vice-versa.

It is also an ongoing discussion how powerful is StoqAQC respect to AQC and if such a model can reach a speedup over classical algorithms. It seems that classical solutions reach the results of StoqAQC.

To conclude the discussion, it is possible to combine the adiabtic evolution with the circuit model to build other universal architecture systems.

---

[1] T. Albash and D. A. Lidar, Rev. Mod. Phys. **90**, 015002 (2018).
[2] D. Bacon and S. T. Flammia, Phys. Rev. Lett. **103**, 120504 (2009), arXiv:0905.0901 [quant-ph] .