Power of Adiabatic Quantum Computation

Itay Hen
Information Sciences Institute, USC

Physics Colloquium, USC
February 10, 2014

arXiv preprints:
1207.1712, 1301.4956, 1307.6538, 1401.5172
Motivation

in what ways quantum computers are more efficient than classical computers?

what problems could be solved more efficiently on a quantum computer?

these are two of the most basic and important questions in the field of quantum information/computation.
Outline

- introduction:
  - what is a quantum computer?
  - adiabatic quantum computing (AQC)
  - performance (so far) of AQC

- adiabatic quantum counting

- Bernstein-Vazirani and Simon’s problems in AQC.

- (time permitting:) solving the graph isomorphism problem with a quantum annealer

- conclusions and challenges
Introduction
Motivation

in what ways quantum computers are more efficient than classical computers?

what problems could be solved more efficiently on a quantum computer?

what is a quantum computer?
Motivation

in what ways quantum computers are more efficient than classical computers?
what problems could be solved more efficiently on a quantum computer?

what is a quantum computer?
a quantum computer is a machine that utilizes aspects of quantum physics to perform calculations.
What is a quantum computer?

- Both classical and quantum computers may be viewed as machines that perform computations on given inputs and produce outputs. Both inputs and outputs are strings of bits (0’s and 1’s).

- Classical computers are based on manipulations of bits. Bits are equivalent to Ising spins (−1’s and 1’s).
What is a quantum computer?

- In classical computers, computation is usually done by applying local gates to the state of the system.
- These gates (AND, OR, NOT, XOR, …) advance the state of the system until the result of the desired computation is achieved.
- At any given time, the system is in a unique classical configuration (i.e., in a state that is a string of 0’s and 1’s).
What is a quantum computer?

- quantum computers on the other hand manipulate quantum bits (or qubits for short) – equivalent to spin-1/2 particles.

- qubits can be in a superposition of up and down for example.

- the range of intermediate states that a quantum computer can be in and the range of operations it can perform are much larger than those of a classical computer (superposition, entanglement, etc.).
What is a quantum computer?

- much like classical gates, there are also quantum gates (unitary operators) with which one can construct quantum circuits.

- the Hadamard gate gives us superpositions. other gates entangle.

- **advantage**: these unique properties can be utilized towards faster computations.

- **disadvantage**: quantum computers don’t really exist. many technological difficulties. noise/interaction with environment.
What is a quantum computer?

- much like classical gates, there are also quantum gates (unitary operators) with which one can construct quantum circuits.
- the Hadamard gate gives us superpositions. other gates entangle.
- advantage: these unique properties can be utilized towards faster computations.
- disadvantage: quantum computers don’t really exist. many technological difficulties. noise/interaction with environment
Introduction

turns out that at least in theory quantum computers are faster than classical computers

- best-known example is
  Shor’s algorithm for integer factorization:
  - solves the problem in polynomial time: **exponential speedup**.
  - of practical importance (RSA code breaking).
  - current quantum computers can factor integers up to 21.
  - there may be a classical algorithm that is just as fast (or even faster) but that has not yet been found.

- other examples are mostly orcaular, e.g., Grover’s search algorithm which gives a quadratic speedup.
quantum computation does not have to be done via gates.

there are other ways to quantum-compute, that may be more practical.

one such way is Adiabatic Quantum Computing (AQC).

AQC is a general approach to solve a broad range of hard optimization problems using “quantum annealing” [Farhi et al., 2001]

USC’s ISI hosts one such machine “D-Wave Two” (only two available).
The adiabatic theorem of QM

- The adiabatic theorem of QM tells us that a physical system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough and if there is a gap between the eigenvalue and the rest of the Hamiltonian's spectrum.
The adiabatic theorem of QM

- The adiabatic theorem of QM tells us that a physical system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough and if there is a gap between the eigenvalue and the rest of the Hamiltonian's spectrum.

- Example: change the strength of a harmonic potential of a system in the ground state:

- An abrupt change (a diabatic process):
The adiabatic theorem of QM

- The adiabatic theorem of QM tells us that a physical system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough and if there is a gap between the eigenvalue and the rest of the Hamiltonian's spectrum.

- Example: change the strength of a harmonic potential of a system in the ground state:

- A gradual slow change (an **adiabatic** process): wave function can “keep up” with the change.
The quantum adiabatic algorithm (QAA)

- the mechanism proposed by Farhi et al., the QAA:

1. take a difficult (classical) optimization problem, generically: find minimizing configuration of the cost function $E = f(x_1, x_2, \ldots, x_n)$ where $x_i$ are binary variables ($0,1$) and $f(\cdots)$ is given.

2. encode the problem in a “problem” Hamiltonian, $\hat{H}^{(p)}$, such that its ground state encodes the solution.

3. prepare the system in the ground state of another, easily solvable, “beginning (driver) Hamiltonian” $\hat{H}^{(b)}$.

4. vary the Hamiltonian *slowly* and smoothly from $\hat{H}^{(b)}$ to $\hat{H}^{(p)}$ until ground state of $\hat{H}^{(p)}$ is reached.
The quantum adiabatic algorithm (QAA)

- The interpolating Hamiltonian is this:

\[ \hat{H}(t) = [1 - s(t)]\hat{H}^{(b)} + s(t)\hat{H}^{(p)} \]

- \( \hat{H}^{(b)} \) is an easily solvable beginning Hamiltonian, which does not commute with \( \hat{H}^{(p)} \).
- \( \hat{H}^{(p)} \) is the problem Hamiltonian whose ground state encodes the solution of the optimization problem.

- The parameter \( s \) obeys \( 0 \leq s(t) \leq 1 \), with \( s(0) = 0 \) and \( s(\mathcal{T}) = 1 \). Also: \( \hat{H}(0) = \hat{H}^{(b)} \) and \( \hat{H}(\mathcal{T}) = \hat{H}^{(p)} \).

- Here, \( t \) stands for time and \( \mathcal{T} \) is the running time, or complexity, of the algorithm.
The quantum adiabatic algorithm (QAA)

- the interpolating Hamiltonian is this:

\[
\hat{H}(t) = [1 - s(t)]\hat{H}^{(b)} + s(t)\hat{H}^{(p)}
\]

- the adiabatic theorem ensures that if the change in \( s(t) \) is slow enough, the system will stay close to the ground state of the instantaneous Hamiltonian throughout the evolution.

- a measurement at the end of the evolution, will give the solution of the original problem with high probability.

process should be slow, but how fast can it still be?
Landau-Zener transition

- **bottleneck is likely to be** where the gap to the first excited state is smallest. There, the **probability to “get off track” is maximal.**

- **generic case:** Landau-Zener theory gives the prob. to stay in ground state as:

\[ P \approx 1 - e^{-\frac{\pi \Delta^2}{2\hbar \nu}} \]

- **runtime should obey:**

\[ T \gg \frac{\text{max}_s |\langle 0|d\hat{H}/ds|1\rangle|}{\text{min}_s [\text{gap}^2]} \]
Landau-Zener transition

- bottleneck is likely to be where the gap to the first excited state is smallest. Here, the probability to “get off track” is maximal.

- generic case: Landau-Zener theory gives the probability to stay in the ground state as:

\[ P \approx 1 - e^{-\frac{\pi \Delta^2}{2 \hbar v}} \]

- runtime should obey:

\[ T \gg \frac{\max_s |\langle 0 | d\hat{H} / ds |1 \rangle|}{\min_s [\text{gap}^2]} \]
Performance of the QAA

how powerful is AQC?

- Shor’s algorithm is formulated in terms of gates. Can it be formulated as an adiabatic algorithm? Unclear.

- It is known that AQC is **polynomially-equivalent** to the gate model [Aharonov et al., 2004]. However, equivalence is suboptimal and has an artificial flavor.

- We do have an **adiabatic equivalent** to Grover’s algorithm [Roland and Cerf, 2002]. Generalizable to Quantum Counting and other problems [IH, 2013], but only quadratic speedup.
Performance of the QAA

- QAA fails for many optimization problems in NP (and in P).

- figure shows behavior of gap as a function of problem size on a log-linear scale for three SAT problems (one in P).

- gap closes exponentially fast (IH and A.P. Young, 2011).

- one reason is: it’s hard to incorporate our knowledge of the structure of the problem into the algorithm (e.g., hard to choose a clever beginning Hamiltonian) to speed up computation.
Quantum Adiabatic Counting
Motivation

Grover’s gate-model algorithm (1997) for unstructured search: Quadratic speedup

- given an unstructured database of size $N$, with $M$ marked items, how long (how many queries) will it take us to find a marked item?
- classically, all one can do is randomly query the database.
- on average, $\sim N/M$ queries are needed.

Roland and Cerf (2002): An adiabatic quantum analogue that is just as efficient
Roland and Cerf [2002] proposed an algorithm based on Local Adiabatic Evolution (LAE). Formulated a local condition for adiabaticity, which however requires more knowledge about the underlying problem.

- **Global adiabaticity:**
  \[
  \frac{1}{\mathcal{F}} \leq \varepsilon \frac{\min_s [\text{gap}^2]}{\max_s |\langle 0 | d\hat{H} / ds | 1 \rangle|}
  \]

- **Local adiabaticity:**
  \[
  \frac{ds}{dt} \leq \varepsilon \frac{\text{gap}(s)^2}{|\langle 0 | d\hat{H} / ds | 1 \rangle|}
  \]
The adiabatic Grover’s algorithm

- In AQC terminology, we are given a scrambled (diagonal) problem Hamiltonian with $M$ zeroes and $(N - M)$ ones:

$$\hat{H}^{(p)} = \text{diag} (\pi [0, \ldots, 0, 1, \ldots 1])$$

- Here, $\pi []$, is an arbitrary permutation.

- We are then asked to find a basis state that corresponds to a zero entry (a “solution” state / a ground state).

- Key idea: wisely choose beginning Hamiltonian:

$$\hat{H}^{(b)} = -|+\rangle\langle+|$$ with

$$|+\rangle = N^{-1/2} \sum_{w=1 \ldots N} |w\rangle$$

- Ground state is $|+\rangle$ with energy $-1$ (the rest of the spectrum is 0’s). symmetric with respect to all computational-basis states.
The adiabatic Grover’s algorithm

- The interpolating Hamiltonian is

\[ \hat{H}(t) = [1 - s(t)]\hat{H}^{(b)} + s(t)\hat{H}^{(p)} \]

- Symmetry of Hamiltonian reduces the problem to a two-level system as the wave function, at all times, has the form:

\[ |\psi\rangle = a(t) \sum |solution\_states\rangle + b(t) \sum |non\_solution\_states\rangle \]

- The equations can therefore be solved analytically to give the gap and matrix element for each value of \( s \).
The adiabatic Grover’s algorithm

- gap is simply given by:

\[ g(s) = \sqrt{1 - 4s(1 - s)(1 - M/N)} \]

- the matrix element can also be easily computed (also, bounded by 1).

- use the LAE formula to establish a path \( s(t) \):

\[ \frac{ds}{dt} = \varepsilon g(s)^2 \]

\[ T \sim \frac{1}{\varepsilon} \sqrt{N/M} \]

- when integrated, in the large \( N \) limit, we get:

- i.e., a quadratic speedup over the classical algorithm.
  also, same speedup as the original Grover algorithm.
Quantum counting

what if the number of solutions $M$ is not known?

- this could be disastrous for the gate-model Grover’s algorithm. Underestimating or overestimating $M$ could lead to substantial errors in the result of the algorithm.

- in the gate model, this is avoided by first applying an algorithm to estimate (i.e., to count) $M$.

- this procedure is known as quantum counting. Requires $\sim \sqrt{N}$ operations.
Quantum counting

- remarkably, in the adiabatic case, counting is actually not strictly needed, because of the structure of the gap:

$$g_M(s) = \sqrt{1 - 4s(1 - s)(1 - \frac{M}{N})}$$
Quantum counting

- the gap

\[ g_M(s) = \sqrt{1 - 4s(1-s)(1 - M/N)} \]

- has the property:

\[ M_1 \leq M_2 \implies g_{M_1}(s) \leq g_{M_2}(s) \]

- which implies the following for the adiabatic path.

\[ \frac{ds}{dt} \leq \varepsilon \frac{g_{M_1}(s)^2}{|\langle 0 | d\hat{H} / ds | 1 \rangle|} \leq \varepsilon \frac{g_{M_2}(s)^2}{|\langle 0 | d\hat{H} / ds | 1 \rangle|} \]

- i.e., it is okay to underestimate \( M \) (e.g., take \( M = 1 \))
An adiabatic QC algorithm

- This interesting property may be used to formulate a quantum adiabatic counting algorithm.

- Given a Grover-type problem Hamiltonian with \( M \) solutions (zero entries), where \( M \) is unknown, how can AQC be used to estimate \( M \)?

- Answer in short: run the QAA repeatedly using an adiabatic path \( s(t) \) that is optimal for another choice, \( M_* \):

\[
\frac{ds}{dt} = \varepsilon g_{M_*}(s)^2
\]

- Key question: how many QAA runs do we need?
Probability of finding a solution

- for $M, M_* \ll N$, i.e., number of solutions is small (this is the interesting case),

- we can solve the two-level system that really has $M$ solutions but with a path that is optimal for some $M_*$. 

- we can calculate the probability to end up in the superposition of solution-states. For large $N$, we get:

$$P_{solution}^{(QAA)} = |\langle \psi | \psi_{sol} \rangle|^2 \approx 1 - e^{-\frac{\pi M}{4\varepsilon M_*}} \approx \frac{\pi M}{4\varepsilon M_*}$$

(true if we use $M_*$ to overestimate $M$, so exp. is small).

- this is a simple Landau-Zener type probability. It is well-defined and is $N$-independent in the $N \to \infty$ limit.
Complexity of classical counting

- **query:**
  - pick an item at random

- **cost of one query:**
  - 1

- **$P(\text{marked item})$**
  - $\hat{p} = p = M/N$

- **error after $k$ trials:**
  - $\Delta \hat{p} = \sqrt{p(1-p)/k} \approx \sqrt{p/k}$
  - $\Delta \hat{M} \sim \sqrt{M} = \sqrt{NM/k}$

- **number of trials needed:**
  - $O(N)$

- **total cost:**
  - $O(N)$
Complexity of quantum counting

- query:
  run the QAA

- cost of one query:
  $O(\sqrt{N/M_*})$

- $P(\text{marked item})$
  $\hat{p} = 1 - e^{-\frac{\pi M}{4\varepsilon M_*}} \approx \frac{\pi M}{4\varepsilon M_*}$

- error after $k$ trials:
  $\Delta \hat{p} = \sqrt{p(1-p)/k} \approx \sqrt{p/k}$
  $\Delta \hat{M} \sim \sqrt{M} \sim \sqrt{MM*/k}$

- number of trials needed:
  $O(M_*)$

- total cost:
  $O(\sqrt{NM_*}) = O(\sqrt{NM})$
Bernstein-Vazirani and Simon's problems
controlled adiabatic evolution

- consider several quantum adiabatic evolutions running in parallel:

\[
\hat{H}_w(t) = [1 - s(t)]\hat{H}^{(b)} + s(t)\hat{H}_w^{(p)}
\]

- the set \{\ket{w}\bra{w}\}\ is a complete set of orthogonal 1D projection operators onto the \ket{w}\ subspaces.

- all beginning Hamiltonians are the same (\hat{H}^{(b)}) but final Hamiltonians (\hat{H}_w^{(p)}) are different.

- e.g.,

\[
\begin{pmatrix}
\hat{H}_1(t) & 0 & 0 \\
0 & \hat{H}_2(t) & 0 \\
0 & 0 & \hat{H}_3(t)
\end{pmatrix}
\]
Controlled adiabatic evolution

- total beginning Hamiltonian is:
  \[ \hat{H}(t = 0) = \sum_w |w\rangle\langle w| \otimes \hat{H}_w(t = 0) = 1 \otimes \hat{H}^{(b)} \]

- so ground state of beginning Hamiltonian is any:
  \[ |\psi_b\rangle = \sum_w c_w |w\rangle \otimes |gs_b\rangle \]

- the state in the first subsystem determines which \( \hat{H}_w(t) \) will act. in general:
  \[ |\psi_f\rangle = \sum_w c_w |w\rangle \otimes |gs_w\rangle \]
The Bernstein-Vazirani problem

- the BV problem was the first to show quantum (polynomial) speedup over probabilistic classical algorithms.

- the problem is: given an oracle function of the integer $w$:

\[ f(w) = \sum_{k=0}^{n-1} a_k w_k \mod 2 \]

find the integer $a$.

- here, $a_k$ and $w_k$ are the bits of the $n$-bit input $w$ and the $n$-bit unknown integer $a$. 
The Bernstein-Vazirani problem

- given the oracle function:

\[ f(w) = \sum_{k=0}^{n-1} a_k w_k \mod 2 \]

- how can we find \( a \)?

- classically: it takes \( O(n) \) queries. e.g., call \( f(\cdot) \) with \( w = 2^j \) for \( j = 0..n-1 \).

\[ f(2^j) = a_j \]

- quantum gate (circuit) model requires \( O(1) \) calls. how about AQC?
The Bernstein-Vazirani problem

- to solve with AQC, we first need to set up a quantum oracle.
- in gate model, oracle is unitary. In AQC it is hermitian:

\[
\hat{H}^{(p)} = \frac{1}{2} \sum_{w \in \{0,1\}^n} |w \rangle \langle w| \otimes [1 - (-1)^{f(w)} \sigma_z]
\]

- this is an \((n+1)\)-bit hermitian operator.
  - first subspace contains \(n\) bits. Second subspace has 1 bit.
- act with \(\hat{H}^{(p)}\) on a classical state \(|w \rangle \otimes |0\rangle\), you get energy 0 if indeed \(f(w) = 0\) and energy 1 otherwise.
- its ground states are \(|w \rangle \otimes |f(w)\rangle\) with energy 0 (half of the states are ground states).
to solve with AQC, we first need to set up a quantum oracle.

in gate model, oracle is unitary. in AQC it is hermitian:

\[ \hat{H}(p) = \frac{1}{2} \sum_{w \in \{0,1\}^n} |w\rangle\langle w| \otimes [1 - (-1)^{f(w)} \sigma_z] \]

this is an \((n + 1)\)-bit hermitian operator.

first subspace contains \(n\) bits, second subspace has 1 bit.

act with \(H\) on a classical state \(|w\rangle \otimes |0\rangle\), you get energy 0 if indeed \(f(w) = 0\) and energy 1 otherwise.

its ground states are \(|w\rangle \otimes |f(w)\rangle\) with energy 0 (half of the states are ground states).
The Bernstein-Vazirani problem

- for the beginning Hamiltonian, we choose the simple:

\[ \hat{H}^{(b)} = \hat{1} \otimes \sigma_x = \sum_{w \in \{0,1\}^n} |w\rangle \langle w| \otimes \sigma_x \]

- clearly the driver has a degenerate ground state. we will choose the initial state to be the easily-preparable

\[ |\psi_b\rangle = \sum_{w \in \{0,1\}^n} |w\rangle \otimes |+\rangle = |+\rangle \otimes |+\rangle \]

- here, \(|+\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)^\otimes\) is the state in which all spins in the subsystem are pointing in the positive \(x\)-direction.
The Bernstein-Vazirani problem

- for the beginning Hamiltonian, we choose the simple:
  \[ \hat{H}^{(b)} = \hat{1} \otimes \sigma_x = \sum_{w \in \{0,1\}^n} |w\rangle \langle w| \otimes \sigma_x \]

- clearly the driver has a degenerate ground state. we will choose the initial state to be the easily-preparable
  \[ |\psi_{\text{driver}}\rangle = |+\rangle \otimes |+\rangle \]

- here, \[|+\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \otimes \]
  is the state in which all spins in the subsystem are pointing in the positive \(x\)-direction.
The Bernstein-Vazirani problem

- the total Hamiltonian becomes:

$$\hat{H} = \sum_{w \in \{0,1\}^n} |w\rangle\langle w| \otimes [(1 - s)\sigma_x + s(1 + (-1)^{f(w)}\sigma_z)]$$

- the trick is that for each $|w\rangle$ in the first subsystem, the second subsystem tunnels adiabatically in a trivial manner from the initial $|+\rangle$ to the ground state, be it $|0\rangle$ or $|1\rangle$.

- most importantly, to reach the ground state, the runtime does not depend on the number of qubits $n$. It is $O(1)$.

- the end state is:

$$|\psi_f\rangle = \sum_{w \in \{0,1\}^n} |w\rangle \otimes |f(w)\rangle$$
The Bernstein-Vazirani problem

- The total Hamiltonian becomes:

\[
\hat{H} = \sum_{w \in \{0,1\}^n} |w\rangle\langle w| \otimes [(1 - s)\sigma_x + s(1 + (-1)^{f(w)}\sigma_z)]
\]

- The trick is that for each \( |w\rangle \) in the first subsystem, the second subsystem tunnels adiabatically in a trivial manner from the initial \(|+\rangle\) to the ground state, be it \(|0\rangle\) or \(|1\rangle\).

- Most importantly, to reach the ground state, the runtime does not depend on the number of qubits \( n \). It is \( O(1) \).

- The end state is:

\[
|\psi_f\rangle = \sum_{w \in \{0,1\}^n} |w\rangle\otimes|f(w)\rangle
\]
The Bernstein-Vazirani problem

- noting that:

\[ |f(w)\rangle = \frac{1}{\sqrt{2}} (|+\rangle + (-1)^{f(w)} |-\rangle) \]

where \(|\pm\rangle = \frac{1}{\sqrt{2}} (|0\rangle \pm |1\rangle)\) are the \(x\)-basis up and down states,

- the final state may be rewritten as:

\[ |\psi_f\rangle = \frac{1}{\sqrt{2}} \left( \sum_w |w\rangle \otimes |+\rangle + \sum_w (-1)^{f(w)} |w\rangle \otimes |-\rangle \right) \]

- i.e., the equal superposition of the initial state (left sum) and something we can use (right sum).

Itay Hen

USC Physics Colloquium

Feb 10, 2014
The Bernstein-Vazirani problem

- noting that:

\[ |f(w)\rangle = \frac{1}{\sqrt{2}} (|+\rangle + (-1)^{f(w)}|\rangle - \rangle) \]

where \(|\pm\rangle = \frac{1}{\sqrt{2}} (|0\rangle \pm |1\rangle)\) are the \(x\)-basis up and down states,

- the final state may be rewritten as:

\[ |\psi_f\rangle = \frac{1}{\sqrt{2}} \left( \sum_w |w\rangle \otimes |+\rangle + \sum_w (-1)^{f(w)} |w\rangle \otimes |\rangle - \rangle \right) \]

the initial state \(|\psi_b\rangle\)

- the equal superposition of the initial state (left sum) and something we can use (right sum) and something we can use (\(\otimes_{k=0}^{n-1} \frac{1}{\sqrt{2}} (|0\rangle + (-1)^{a_k} |1\rangle)\)

that's all we need
Simon’s problem is trickier to solve, but profit is gaining an exponential speedup over classical algorithms. Also more important as it serves as the basis for Shor’s algorithm.

- Here, the oracle is a function from $n$ bits to $(n-1)$ bits:

$$g: \{0,1\}^n \to \{0,1\}^{n-1}$$

with

$$g(x) = g(y) \text{ iff } y = x \oplus a$$

- Here too $a \neq 0$

is again an unknown $n$-bit integer.

- As with BV: find $a$!
Simon’s period-finding problem

- reminder of what $\oplus$ (XOR) is:
- classically, to find the “XOR-mask” $a$, we need an exponential number of queries.
- we must call the oracle until we get two inputs with same output. this is very unlikely (birthday paradox). problem is exponentially hard $O(2^{n/2})$ queries.
- gate model requires $O(n)$ queries. exponential speedup.
Simon’s period-finding problem

- how about AQC?
- same tricks as with BV problem. this time, the second subspace has \((n - 1)\) bits and the adiabatic oracle is:

\[
\hat{H}(p) = \sum_{w \in \{0,1\}^n} \sum_{y \in \{0,1\}^{n-1}} d(y \oplus g(w)) |w\rangle \langle w| \otimes |y\rangle \langle y|
\]

- here, the function \(d(\cdot)\) computes the Hamming distance between \(y\) and \(g(w)\).
- now the Hilbert space is that of \(n\) bits (first subsystem) plus \((n - 1)\) bits (second subsystem).
- its ground states are \(|w\rangle \otimes |g(w)\rangle\) with energy 0.
Simon’s period-finding problem

- how about AQC?
- same tricks as with BV problem. this time, the second subspace has \((n - 1)\) bits and the adiabatic oracle is:

\[
\hat{H}^{(p)} = \sum_{w \in \{0,1\}^n} \sum_{y \in \{0,1\}^{n-1}} d(y \oplus g(w)) |w\rangle\langle w| \otimes |y\rangle\langle y|
\]

Computes the Hamming distance between \(y\) and \(g(w)\).

- the Hilbert space is that of \(n\) bits (first subsystem) and \((n - 1)\) bits (second subsystem).
- its ground states are \(|w\rangle \otimes |g(w)\rangle\) with energy 0.
Simon’s period-finding problem

- for the driver, we choose the simple Hamiltonian:

\[ \hat{H}^{(b)} = \hat{1} \otimes \sum_{k=1}^{n-1} \sigma^k_x = \sum_{w \in \{0,1\}^n} |w\rangle \langle w| \otimes \sum_{k=1}^{n-1} \sigma^k_x \]

- clearly the driver is degenerate. we will choose the initial state to be the easily-preparable:

\[ |\psi_b\rangle = \sum_{w \in \{0,1\}^n} |w\rangle \otimes |+\rangle = |+\rangle \otimes |+\rangle \]

- as before, here, |+\rangle is the state where all spins in the subsystem are pointing in the positive \( x \)-direction.
Simon’s period-finding problem

- (after some algebra) the total Hamiltonian becomes:

\[ \hat{H} = \sum_{w \in \{0,1\}^n} |w\rangle \langle w| \otimes \left[ \sum_{k=1}^{n-1} [(1 - s)\sigma_x^k + s(1 + (-1)^{g_k}\sigma_z^k)] \right] \]

- as with BV, for each \( |w\rangle \) in the first subsystem, the second subsystem tunnels adiabatically in a trivial manner from the initial \( |+\rangle \) to the ground state \( |0/1\rangle \).

- most importantly, to reach the ground state, the runtime does not depend on the number of qubits \( n \), it is \( O(1) \).

- the end state is:

\[ |\psi_f\rangle = \sum_{w \in \{0,1\}^n} |w\rangle \otimes |g(w)\rangle \]
Simon’s period-finding problem

- noting that:

\[ |\psi_f\rangle = \sum_{w \in \{0,1\}^n} |w\rangle \otimes |g(w)\rangle = \sum_{w \in \{0,1\}^n / g} [|w\rangle + |w \oplus a\rangle] \otimes |g(w)\rangle \]

- a measurement of the second subsystem will pick out one \(|g(w_*)\rangle\) at random, leaving the first subsystem in a state:

\[
|\psi_{w_*}\rangle = \frac{1}{\sqrt{2}} (|w_*\rangle + |w_* \oplus a\rangle) \otimes |g(w_*)\rangle
\]

\[
= \frac{1}{\sqrt{2^{n+1}}} \sum_{y \in \{0,1\}^n} (-1)^{w_* \cdot y} \left[ 1 + (-1)^{f(y)} \right] |y\rangle \otimes |g(w_*)\rangle
\]

\[
= \frac{1}{\sqrt{2^{n-1}}} \sum_{f(y) = 0} (-1)^{w_* \cdot y} |y\rangle \otimes |g(w_*)\rangle,
\]
Simon’s period-finding problem

- A subsequent measurement of the first subsystem, will give us a random state \( |y \rangle \) that obeys \( f(y) = 0 \), i.e.,

\[
f(y) = \sum_{k=0}^{n-1} a_k y_k = 0 \pmod{2}
\]

- Repeating the algorithm \( n \) times (or a bit more) will generate a set of \( n \) linearly-independent equations for \( a_k \).
- This set can be solved using Gaussian Elimination (mod 2).
- The total runtime of the algorithm (without taking into account solving the set of equations) is \( O(n) \), as in the gate model.
- Is AQC equivalent to gate model? These results support the conjecture.
Solving the graph isomorphism problem with a quantum annealer

Joint work with Peter Young (UCSC)
The graph isomorphism problem

- the graph isomorphism problem: are two graphs the same upon permuting the indices?

- how could one use adiabatic quantum computation to answer this question?

- conjecture: all non-isomorphic graphs can be distinguished by:
  
  - putting a suitable “problem Hamiltonian” on the edges of the graph.
  
  - running the QAA with a transverse-field driver.
  
  - computing appropriate average quantities (not-necessarily at the end of the run), by performing repeated measurements.
The graph isomorphism problem

- if the Hamiltonian and the quantities we measure are invariant under permutation of the indices, isomorphic graphs will yield the same averages.

- we hypothesize that non-isomorphic graphs can always be distinguished.

- we have tested the hypothesis for some small graphs ($N \leq 29$) from various families of graphs that are known to be hard to distinguish (identical adjacency matrix spectra, in some cases also identical problem Hamiltonian spectra).

- so far, method seems to work if measurements are accurate enough. have not found a counterexample.
The graph isomorphism problem

- we chose a simple antiferromagnet on the graph:
  \[ \hat{H}(p)(G) = \sum_{\langle i, j \rangle \in G} \sigma_i^z \sigma_j^z \]

- main results were obtained for Strongly Regular Graphs (SRG's; families of similar but non-isomorphic graphs) but not just. we considered sizes from \( N = 15 \) to 29 vertices.

- we computed energy, \( x \)-magnetization \( (M_x) \) and the spin glass order parameter \( (Q_2) \) for different values of the adiabatic parameter \( s \):
  \[ M_x = \frac{1}{N} \sum_{i=1}^{N} \langle \sigma_i^x \rangle \]
  \[ Q_2 = \sqrt{\frac{1}{N(N-1)} \sum_{i \neq j}^{N} \langle \sigma_i^z \sigma_j^z \rangle} \]
The graph isomorphism problem

- The value of \( Q_2 \), the spin-glass order parameter, in the ground state for the two non-isomorphic SRG's on \( N = 16 \) vertices, as a function of the adiabatic parameter \( s \). The two graphs are clearly distinguished.

- Scatterplot of \( Q_2 \) against \( M_x \) in the \( s \to 1 \) limit for the 41 SRG's with \( N = 29 \). The QAA distinguishes all graphs in the family in that limit (although some of the values are close together).
Conclusions and challenges
Conclusions

- we can do many things with AQC.
- very strong evidence that AQC is as powerful as other quantum computing paradigms, specifically the gate model.
- managed to show that many AQC algorithms can be optimally formulated:
  - Grover, generalizations, quantum counting
  - deterministic Deutsch-Josza (not discussed)
  - Bernstein-Vazirani
  - Simon’s period finding problem
- hopefully in the long run, this will help say more about the “true power” of adiabatic quantum computation.
Graph isomorphism conclusions

- perhaps there are more suitable Hamiltonians. here, we have used “glassiness” to solve the graph isomorphism problem.

- perhaps there are better measurements that can be performed in order to distinguish between graphs, e.g., susceptibilities. here, we have mainly used the spin-glass order parameter.

- can be tested experimentally on existing D-Wave hardware with relatively minor modifications (ongoing work with Peter Young and Matt Wittmann at UCSC).

- it is unclear whether or not the algorithm is efficient. what is the nature of the quantum phase transition? need to investigate size-dependence of minimum gap for example.

- clearly more tests are required.
Challenges

- prove that AQC is strictly equivalent to gate-model
- come up with experimentally-practical adiabatic algorithms
- quantum adiabatic gates and quantum adiabatic circuits
- ...

Thank you!

Power of Adiabatic Quantum Computation

Itay Hen
Information Sciences Institute, USC

Physics Colloquium, USC
February 10, 2014

arXiv preprints:
1207.1712, 1301.4956, 1307.6538, 1401.5172
Grover generalization
Generalizing the Grover setup

- consider the following problem Hamiltonian:

\[ \hat{H}_p = e_0 + \text{diag}(\pi[f_0, \ldots, f_0, f_1, \ldots, f_1, \ldots, f_k, \ldots, f_k]) \]

- here, the various \( f_j \)'s (\( j = 0 \ldots K \)) are given, along with their multiplicities, \( m_j \).

- the constant \( e_0 \) is not given and \( \pi[] \) is an arbitrary permutation of the \( f_j \)'s which is also not given.

- goal: find a minimum-cost configuration.

- classically, \( O(N) \) operations are needed, as there is no structure to the problem.
Generalizing the Grover setup

- driver Hamiltonian is the symmetric one from before.

- the full Hamiltonian is invariant under the unitary transformation that “unscrambles” (i.e., orders) the problem Hamiltonian.

- leads to a compact set of \((K + 1)\) Shrödinger equations to solve:

\[
i \dot{c}_j = -(1 - s) \sum_{k=0}^{K} \eta_k c_k + sf_j c_j
\]

- there are (only) \((K + 1)\) distinct amplitudes \(c_j\). the size of the system, \(N\), does not appear (only in the ratios \(\eta_j = m_j/N\)).

- the constant \(e_0\) may be removed from the equations (affects only global phase).

- system may therefore be easily solved or at least analyzed.
An analog Deutsch-Josza algorithm

- DJ problem: find whether $\hat{F}$ is balanced:

$$\hat{F} = \text{diag}(\pi[0,0,0, \ldots, 0,1,1,1, \ldots, 1])$$

i.e., $f_0 = 0, f_1 = 1$ with $m_0 = m_1 = N/2$, or constant:

$$\hat{F} = \text{diag}(0, \ldots, 0) \text{ or } \hat{F} = \text{diag}(1, \ldots, 1)$$

- take $\hat{H}_p = \hat{F}$ and assume it is balanced.

- simplify the ($N$-independent) two-level system.

- find a (non-adiabatic) path $s(t)$ that will give (precisely) the end probability:

$$P_{\text{solution state}}(s = 1) = 1$$
The Deutsch-Josza problem

- instead of:
  - choose a path \( s(t) \)
  - find the probability profile \( p(t) \)

- we can do the opposite:
  - choose a probability profile \( p(t) \)
  - find the path \( s(t) \)

- we choose the probability profile such that at the end of the run, the system is in the superposition of solution-states with probability 1 (given that the function is balanced).
The Deutsch-Josza problem

- for example, this probability profile $p(t)$ leads to this path $s(t)$

- probability of failure (solid red) drops sharply to zero. (dashed blue line is an adiabatic path).
The Deutsch-Josza problem

- what if we execute a QAA with the path that we found for a balanced $\hat{F}$ and it turns out that $\hat{H}_p = \hat{F}$ is constant?

- initial state is the equal superposition of all basis states.

- problem Hamiltonian is proportional to the identity.

- independently of the path, the state will remain the equal superposition throughout the evolution (up to a global phase).

- measurement of the energy at the end of the evolution will produce the constant (0 or 1).
The Deutsch-Josza problem

- let us then execute the QAA twice with the path found.

\[
\hat{H}_p = \hat{F} \quad \text{QAA} \quad H_p = 1 - \hat{F}
\]

- if the function is balanced both energy readouts will be 0, because the final state will be a ground state of the problem.
- if the function is constant, energy will be that constant. once 0 and once 1.
- algorithm therefore distinguishes deterministically (all end probabilities are 1) between the two functions. it is \( N \)-independent, i.e., it is \( O(1) \).
QAA fails for many optimization problems in NP (and in P).

- Satisfiability (SAT) problems (that are NP-complete) have been studied extensively. Perfect test bed for QAA.

- In SAT problems, we ask whether there is an assignment of \( N \) bits (or Ising spins) which satisfies all of \( M \) clauses (or logical conditions).

- Satisfiability of logical expressions, e.g.,

\[
(x_1 \land \neg x_2 \land \neg x_3) \lor (x_4 \land \neg x_3 \land \neg x_5) \lor (x_2 \land x_4 \land \neg x_1).
\]

can be easily encoded as the optimization of suitably constructed problem Hamiltonians:

\[
\hat{H}_p = \sum_{a=1}^{M} \hat{H}_a(\sigma^1_z, \sigma^2_z, \ldots)
\]