Quantum Adiabatic Circuits

Itay Hen\textsuperscript{1} \textsuperscript{1}
\textsuperscript{1}Information Sciences Institute, University of Southern California, Marina del Rey, California 90292, USA
(Dated: January 22, 2014)

We introduce a class of quantum adiabatic algorithms that we claim may be treated as the equivalents of the unitary gates of the quantum gate model. We argue that these gates form a universal set and may therefore be used as building blocks in the construction of arbitrary ‘adiabatic circuits’, analogously to the manner in which the gates are used in the circuit model. One implication of the above constructions is that arbitrary classical boolean circuits as well as gate model circuits may be directly translated to adiabatic algorithms with no additional resources or complexities.

PACS numbers: 03.67.Ac,03.67.Lx
Keywords: Adiabatic quantum computing, Quantum adiabatic algorithm

Introduction.

Theoretical research on quantum computing is motivated by the exciting possibility that quantum computers are inherently more efficient than classical computers due to the advantages that the laws of quantum mechanics provide, such as parallelism, tunneling and entanglement. The implications of having at our disposal reliable quantum computing devices with which diverse problems ranging from code breaking \[1\] through protein folding \[2\] to database searching \[3\], are solved much faster than with the classical computers or supercomputers of today, are of course tremendous.

The actual implementation of quantum computing devices is however hindered by many challenging difficulties, the most prominent of which being the control or removal of quantum decoherence \[4\]. Recent promising experimental research findings \[5–7\] in the field of Adiabatic Quantum Computing (AQC) suggest that a leading candidate to be the first device to solve practical classically-hard problems using quantum principles is the so-called ‘quantum annealer’, which implements the simple yet potentially-powerful quantum-adiabatic algorithmic approach proposed by Farhi et al. \[8\] about a decade ago.

The aforementioned experimental studies, as well as other theoretical work such as the theorem of polynomial equivalence between AQC and the predominant gate model (GM) paradigm of quantum computing \[9, 10\], provide ample motivation for determining the computational capabilities of AQC and its precise relations with other quantum computing paradigms, specifically GM. Demonstrating that algorithms such as Shor’s integer factorization \[1\] are implementable as efficiently on a quantum adiabatic computer would undoubtedly have many practical as well as theoretical consequences that would resonate well beyond Quantum Computing.

Recent studies \[11–14\] examining the performance of certain AQC algorithms, such as Unstructured Database Search \[11\], Quantum Counting \[12\] and Simon’s problem \[13\], against their GM counterparts, suggest that the equivalence between AQC and GM is stronger than the one implied by the principles of polynomial equivalence prescribed in the seminal study of Aharonov et al. \[3, 15\]. However, to date, no such strict equivalence or explicit construction to show that this is indeed the case, has been demonstrated.

Here, we argue that a step forward in that direction may be made by considering the construction of a class of quantum adiabatic algorithms, or subroutines, that we claim may be treated as the equivalents of the unitary gates of the quantum gate model. These ‘adiabatic gates’ form a universal set and may therefore be used to construct general ‘adiabatic circuits’, analogously to the manner in which gates are used in the circuit model. One implication of these constructions is that classical boolean circuits as well as gate model circuits may be directly translated to adiabatic algorithms with no additional resources or complexities.

In our construction of adiabatic gates, we shall be using quantum adiabatic evolution somewhat unconventionally. In our approach, we shall consider the adiabatic evolution of several systems in parallel, followed by a single measurement at the end of the evolutions, performed on the entire system. The main principles of AQC as well as the new approach are presented next.

Controlled adiabatic evolution.

In AQC, one normally (albeit not exclusively) seeks the minimum value and corresponding input configuration of a given cost function, that is encoded as the final (or ‘problem’) Hamiltonian, $\hat{H}(f)$, such that the ground state of the final Hamiltonian and its energy are the solution to the original problem \[15\]. To find the solution, the system is prepared in the ground state of another ‘beginning’ (or ‘driver’) Hamiltonian $\hat{H}(b)$ that must not commute with $\hat{H}(f)$ and has a ground state that is fairly easy to prepare. The Hamiltonian of the system is then slowly interpolated between $\hat{H}(b)$ and $\hat{H}(f)$, normally via $\hat{H}(t) = f_1(t)\hat{H}(b) + f_2(t)\hat{H}(f)$ where $f_1(t)$ [$f_2(t)$] is a smoothly-varying function of time that is positive (zero) at $t = 0$ and zero (positive) at $t = T$. Here, $T$ stands for the runtime of the algorithm. If this process is done slowly enough, the system will stay close to the ground state of the instantaneous Hamiltonian throughout the evolution \[17, 18\], so that one finally obtains a state close to the ground state of $\hat{H}(f)$. At this point, measuring the...
state will yield the solution of the original problem with high probability.

It is clear from the above description, that the analog, continuous, nature of AQC is inherently very different from the discrete nature of GM algorithms that are normally constructed by carrying out local unitary operations that act sequentially to advance the state of the system. For this reason it has been hard so far to draw meaningful analogies between AQC and GM. In what follows, we shall use a slightly unconventional ‘protocol’ for adiabatic evolution, one which somewhat generalizes the above adiabatic procedure, and which, as we shall show, will allow us to perform more complicated calculations than those allowed by the usual scheme.

Consider the following adiabatic-evolution Hamiltonian, defined over a bipartite system:

\[ \hat{H} = f_1(t) \cdot 1 \otimes \hat{H}^{(b)} + f_2(t) \sum_j \hat{P}_j \otimes \hat{H}_j^{(f)} \]

\[ = \sum_j \hat{P}_j \otimes \left[ f_1(t) \hat{H}^{(b)} + f_2(t) \hat{H}_j^{(f)} \right], \] (1)

where the operators \( \{ \hat{P}_j \} \) form a complete set of orthogonal projections on the first subsystem (i.e., \( \hat{P}_i \hat{P}_j = \delta_{ij} \hat{P}_i \) and \( \sum_j \hat{P}_j = 1 \)). The above Hamiltonian may be interpreted as one that executes a ‘controlled’ adiabatic evolution of the second (target) subsystem, interpolating between the beginning Hamiltonian \( \hat{H}^{(b)} \) and one of possibly several final Hamiltonians \( \hat{H}_j^{(f)} \), the latter being determined by the state of the first (control) subsystem, via the projection operators.

For the initial state of the entire system \( |\psi_{\text{init}}\rangle \) to be in the ground state of the total beginning Hamiltonian \( 1 \otimes \hat{H}^{(b)} \), it suffices that \( |\psi_{\text{init}}\rangle \) be in a product state \( |\psi_{\text{init}}\rangle = |\psi\rangle \otimes |\text{g.s.}^{(b)}\rangle \) where the state of the first subsystem \(|\psi\rangle\) could be chosen arbitrarily, and the state of the second subsystem \(|\text{g.s.}^{(b)}\rangle\) is the (non-degenerate) ground state of \( \hat{H}^{(b)} \).

The choice of which of the \( \hat{H}_j^{(f)} \) will serve as the final Hamiltonian, is determined by the state of the first subsystem. If the state of the first subsystem \(|\psi\rangle\) lies in the subspace projected by \( \hat{P}_k \) for some \( k \) (i.e., \( \hat{P}_j |\psi\rangle = \delta_{jk} |\psi\rangle \) for all \( j \)), then the final Hamiltonian will be \( \hat{H}_k^{(f)} \). Of course, the state of the first system will in general have non-vanishing overlap on all subspaces projected by \( \{ \hat{P}_j \} \). From simple linearity considerations, it is easy to see that in this general case, each of these components will evolve according to their respective final Hamiltonians, which would in turn mean that the total Hamiltonian will drive a number of independent adiabatic processes in parallel, each corresponding to its own subspace in the Hilbert space of the first subsystem. Generally, controlled adiabatic evolution may be described as the process:

\[ |\psi_{\text{init}}\rangle = |\psi\rangle |\text{g.s.}^{(b)}\rangle \rightarrow |\psi_{\text{final}}\rangle = \sum_j \hat{P}_j |\psi\rangle |\text{g.s.}_j^{(f)}\rangle, \] (2)

where \(|\text{g.s.}_j^{(f)}\rangle\) is the ground state of \( \hat{H}_j^{(f)} \). Note that in the trivial case where \( \{ \hat{P}_j \} = \{ 1 \} \), the entire process is reduced to the usual adiabatic scheme.

Utilizing the above form of adiabatic evolution, we will next demonstrate how one can use the principles of AQC to construct a class of quantum adiabatic algorithms, which could be viewed as the equivalents of the gates of the gate model. Using these gates, as a direct consequence, adiabatic ‘circuits’ may be constructed. The class of adiabatic gates that we consider below is the general single-qubit rotation and its slightly more complicated generalization of controlled rotation.

**Adiabatic single-qubit rotation gates.**

Consider a single qubit in an arbitrary unknown state \(|\psi\rangle\). Let us now attach to it an auxiliary qubit, initialized to the computational \(|0\rangle\) state (which we shall identify as pointing in the positive \(z\)-direction):

\[ |\psi_{\text{init}}\rangle = |\psi\rangle \otimes |0\rangle. \] (3)

This will be the initial state of an adiabatic algorithm whose evolution will be governed by the Hamiltonian:

\[ \hat{H}(t) = |\vec{n}\rangle \langle \vec{n}| \otimes \hat{H}_0(t) + |\vec{n}_\perp\rangle \langle \vec{n}_\perp| \otimes \hat{H}_\phi(t), \] (4)

where \( \hat{H}_0(t) \) and \( \hat{H}_\phi(t) \) are adiabatic-evolution Hamiltonians, conditioned to act only within the respective subspaces projected by the orthogonal projection operators \([|\vec{n}\rangle \langle \vec{n}| = 1/2 (1 + \vec{n} \cdot \vec{\sigma}) \) and \([|\vec{n}_\perp\rangle \langle \vec{n}_\perp| = 1/2 (1 - \vec{n} \cdot \vec{\sigma}) \) where \( \vec{\sigma} = (\sigma_x, \sigma_y, \sigma_z) \) defined on the Hilbert space of the first qubit. Here, \(|\vec{n}\rangle\) and \(|\vec{n}_\perp\rangle\) form a basis that corresponds to a predetermined unit vector \(\vec{n}\) on the Bloch sphere of the first qubit. The above Hamiltonian should be interpreted as driving two independently-acting, parallel, adiabatic processes [defined by \( \hat{H}_0(t) \) and \( \hat{H}_\phi(t) \)], each acting within their own respective subspace.

The adiabatic-evolution Hamiltonians are chosen to be

\[ \hat{H}_\phi(t) = -\cos \theta(t) \sigma_z - \sin \theta(t) (\cos \phi \sigma_x + \sin \phi \sigma_y), \] (5)

and \( \hat{H}_0(t) = \hat{H}_{\phi = 0}(t) = -\cos \theta(t) \sigma_z - \sin \theta(t) \sigma_x \).

Similarly to \(\vec{n}\), the angle \(\phi\) is also a free parameter of the Hamiltonian. The time-dependence of the Hamiltonian is given here by the angle \(\theta(t)\) such that \(\theta(t = 0) = 0\), and \(\theta(t = T) = \pi/2\). For simplicity, we shall henceforth assume the dependence \(\theta(t) = \pi t / 2T\).

Note that the total Hamiltonian, Eq. (4), is two-local and is of the general form introduced in Eq. (1), with \( \hat{H}^{(b)} = -\sigma_z \) and \( \{ \hat{P}_1, \hat{P}_2 \} = \{ |\vec{n}\rangle \langle \vec{n}|, |\vec{n}_\perp\rangle \langle \vec{n}_\perp| \} \).

The two final Hamiltonians are \( \hat{H}_1^{(f)} = -\sigma_z \) and \( \hat{H}_2^{(f)} = -\sin \phi \sigma_x - \cos \phi \sigma_y \).

Defining \(|+\phi\rangle \equiv \frac{1}{\sqrt{2}} (|0\rangle + e^{i\phi} |1\rangle)\), the Hamiltonian, Eq. (4), will act differently and in parallel on the two complementary subspaces, evolving the auxiliary qubit, initially at \(|0\rangle\), to \(|+\phi\rangle\) [the latter being the ground state of \( \hat{H}_0(T) \)] in the subspace projected by \(|\vec{n}\rangle \langle \vec{n}|\) and to \(|+\phi\rangle\) [the ground state of \( \hat{H}_\phi(T) \)] in the subspace projected by \(|\vec{n}_\perp\rangle \langle \vec{n}_\perp|\). The two evolutions are sketched in Fig. 1.
In the latter case, the resultant qubit will have been rotated by an angle $\phi$ around the $\hat{n}$-axis. In the former case, where the initial state is obtained, one would have to simply repeat the adiabatic evolution one or more times in order to obtain the desired rotated state. The probability of failure (to rotate) is therefore exponentially small: $P_{\text{failure}} = 2^{-k}$ where $k$ is the number of adiabatic executions of the algorithm. Therefore, on average, each adiabatic gate would have to be carried out twice. One could also imagine the gate as including a feedback loop that repeats the adiabatic evolution each time the state of the auxiliary qubit after the measurement is found to be in its initial setting, and halts once the qubit is found to have been flipped. We therefore now have an adiabatic machinery to rotate a qubit by an arbitrary angle $\phi$ around an arbitrarily chosen axis $\hat{n}$, i.e., we have constructed an adiabatic, general, single-qubit 'gate'.

**Adiabatic controlled-rotation gates.**

The above scheme may now be easily generalized to the case where the system initially contains two input qubits, one of which is regarded as a control qubit. Here, the starting state would be the two-qubit state $|\psi\rangle = \alpha|0, \hat{n}\rangle + \beta|0, \hat{n}_\perp\rangle + \gamma|1, \hat{n}\rangle + \delta|1, \hat{n}_\perp\rangle$, where the first qubit here is used as the control qubit. An adiabatic controlled rotation is obtained by attaching, as before, an auxiliary qubit to the initial state:

$$|\psi_{\text{init}}\rangle = |\psi\rangle \otimes |0\rangle,$$

and constructing the slightly more complicated three-local Hamiltonian [13]:

$$\hat{H}(t) = (|0, \hat{n}\rangle\langle 0, \hat{n}| + |0, \hat{n}_\perp\rangle\langle 0, \hat{n}_\perp| + |1, \hat{n}\rangle\langle 1, \hat{n}|) \otimes \hat{H}_0(t) + |1, \hat{n}_\perp\rangle\langle 1, \hat{n}_\perp| \otimes \hat{H}_\phi(t),$$

where $\hat{H}_0(t)$ and $\hat{H}_\phi(t)$ are as previously defined. The end state in this case will be, similarly to the single qubit rotation gate discussed above, an equal superposition of the initial state $|\psi_{\text{init}}\rangle$ and the controlled-rotated state $|\psi_{\text{c. rot.}}\rangle = (\alpha|0, \hat{n}\rangle + \beta|0, \hat{n}_\perp\rangle) + \left(\gamma|1, \hat{n}\rangle + \delta|1, \hat{n}_\perp\rangle\right)$.

A computational-basis measurement of the auxiliary qubit would yield one of the two states with equal probability, as before.

The combination of the controlled gate above and the single-qubit general-rotation gate described earlier, suggest that we now have in our possession a universal set of gates, with which general 'adiabatic circuits' may be built, analogously to the manner in which algorithms are constructed in the gate model.

It would be advantageous though to explicitly describe a couple of specific gates. The NOT gate, for example, corresponds to a rotation by $\pi$ around the $x$-axis, i.e., it is the single-qubit rotation gate with the choice $\phi = \pi$ and...
gate, namely CNOT, may be just as easily constructed, where

\[ |\pm\rangle = \frac{1}{\sqrt{2}} (|0\rangle \pm |1\rangle). \]

The ‘controlled’ version of this gate, namely CNOT, may be just as easily constructed, using the controlled-rotation adiabatic scheme.

Another example, is the Hadamard adiabatic gate which is simply a \(\pi/2\) rotation around the y-axis:

\[ \alpha |0\rangle + \beta |1\rangle = \frac{\alpha + i\beta}{\sqrt{2}} |+y\rangle + \frac{\alpha - i\beta}{\sqrt{2}} |-y\rangle \]  
\[ \rightarrow \frac{\pi}{2}\text{rot.} \quad \frac{\alpha + i\beta}{\sqrt{2}} |+y\rangle + \frac{i(\alpha - i\beta)}{\sqrt{2}} |-y\rangle \]

\[ = \frac{\alpha + \beta}{\sqrt{2}} |0\rangle + \frac{\alpha - \beta}{\sqrt{2}} |1\rangle, \]

where \( |\pm\rangle \) are the eigenstates of the spin operator \( \hat{S}_y \).

**Quantum Adiabatic Circuits.**

It should now be clear that the above proposed class of general single-qubit and controlled rotation gates are universal, and so general adiabatic circuits may be constructed using sequences of those.

First, we note that while the above adiabatic ‘gates’ were shown to act on isolated qubits, the linearity of Quantum Mechanics ensures that the above results hold even if the target qubits are part of a larger system of qubits in a more complicated state.

A sequence of such gates in the above form may thus be used, one after the other, similarly to the manner in which circuits are constructed in the usual gate model, with the final state of the adiabatic evolution of one gate serving as the initial state of the next gate. Within this scheme, one would in principle need one auxiliary qubit for each gate in the circuit although it should be clear that gates that act at different times slices may utilize the same auxiliary qubit as their ancillary resource.

A few remarks are in order: i) Note that the final Hamiltonians of the gates presented above are not diagonal in the computational basis, unlike perhaps the ‘usual’ implementations of quantum adiabatic algorithms. ii) The gap in the adiabatic evolutions of the gates is constant throughout the evolution (and equals to 2). Therefore, the required runtime for each adiabatic gate does not scale with the total number of qubits in the system or with the number of gates in the circuit. The total runtime of a circuit of \( S \) basic gates is simply \( O(S) \). iii) Finally, note that computation in the quantum adiabatic circuit is composed of both adiabatic evolutions and local measurements, each of which having a unique role in driving the computation forward.

**Summary and conclusions.**

We have shown how to use controlled adiabatic evolutions to construct general single-qubit and also controlled two-qubit ‘adiabatic gates’. We have argued that using the above adiabatic gates as building blocks, all gate model circuits (as well as classical boolean circuits) may be built, and at no additional complexity cost or resource overhead. For example, one could straightforwardly construct an adiabatic version of Shor’s integer factorization algorithm \[1\] using only two-local and three-local Hamiltonians. The theoretical and practical implications of an implementable Shor’s algorithm, on a many-qubit quantum annealer that will become available in the near future \[5–7\], may be tremendous, both in the field of Quantum Computing and well beyond it.

On a more technical note, it would be interesting to consider a setup in which all the gate measurements in an adiabatic circuit, are postponed to the end of the circuit and are performed in the correct order, where the ‘failed’ gates, i.e., those gates for which the auxiliary qubits are found back in their initial state, are repeated as necessary until all gates are applied successfully.

Adiabatic quantum computing, in its traditional form, has been shown to have several advantages over the gate model \[20–21\], making it more fault-tolerant and robust against decoherence and dephasing. However the circuit AQc construction proposed here differs from traditional AQC in two main features. First, the usual AQC is normally thought of as one continuous process interpolating between one beginning Hamiltonian and one final Hamiltonian, thereby eliminating the need for gates, that usually also carry around gate errors and therefore need error correction. Second, within the usual AQC scheme, the existence of a gap between the ground state and the rest of the spectrum throughout the adiabatic evolution serves to protect the system against decoherence and dephasing. Given that the method proposed above uses adiabatic gates and degenerate ground states, it would be of importance to study the fault-tolerance and robustness of the scheme presented here. It would be interesting to know whether the adiabatic gates presented here are more amenable to fault-tolerant types of error corrections than unitary evolution gates and how robust these systems are against dephasing and decoherence when compared to traditional AQC.

We thank Mohammad Amin, Dave Bacon, Amir Kalev, Federico Spedalieri and Peter Young for useful comments and discussions. A special thanks to Eleanor Rieffel for fruitful discussions and suggestions.

---

\[ \text{tayhen@isi.edu} \]

[1] P. W. Shor, in *Proc. 35th Symp. on Foundations of Computer Science*, edited by S. Goldwasser (1994) p. 124.

[2] A. Perdomo-Ortiz, N. Dickson, M. Drew-Brook, G. Rose, and A. Aspuru-Guzik, *Scientific Reports* 2, 571 (2012).

[3] L. K. Grover, *Phys. Rev. Lett.* 79, 325 (1997).

[4] M. Schlosshauer, *Rev. Mod. Phys.* 76, 1267 (2004).

[5] C. C. McGeoch and C. Wang, in *Proceedings of the 2013 ACM Conference on Computing Frontiers* (2013).

[6] A. J. Berkley et al., *Phys. Rev. B* 87, 020502(R) (2013).
[7] M. W. Johnson et al., Nature 473, 194 (2011).
[8] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, Science 292, 472 (2001), a longer version of the paper appeared in arXiv:quant-ph/0104129.
[9] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, SIAM Journal of Computing 37, 166 (2007).
[10] A. Mizel, D. A. Lidar, and M. Mitchell, Phys. Rev. Lett. 99, 070502 (2007).
[11] J. Roland and N. J. Cerf, Phys. Rev. A 65, 042308 (2002).
[12] I. Hen, “How fast can quantum annealers count?” (2013), arXiv:1301.4956.
[13] I. Hen, “Period finding with adiabatic quantum computation.” (2013), arXiv:1307.6538.
[14] I. Hen, J. Phys. A: Math. Theor. 47, 045305 (2014), arXiv:1302.7256.
[15] D. Aharonov, W. van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev (2004).
[16] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, Science 292, 472 (2001).
[17] T. Kato, J. Phys. Soc. Jap. 5, 435 (1951).
[18] A. Messiah, Quantum Mechanics (North-Holland, Amsterdam, 1962).
[19] One could imagine replacing the three-local Hamiltonian with an equivalent sequence of adiabatic two-local constructions with the same effect, in order to make the model more attractive experimentally [22].
[20] F. E. Childs, A. M. and J. Preskill, Phys. Rev. A 65, 012322 (2001).
[21] M. Amin, D. V. Averin, and J. A. Nesteroff, Phys. Rev. A 79, 022107 (2009), (arXiv:0708.0384).
[22] Dave Bacon, Private Communication.