

# Adiabatic Quantum Computers and Unsolvable Problems

Bill Davis  
Johns Hopkins University

December 9th 2008

## Abstract

Quantum Computers exploit the peculiarities of the quantum world to out-perform classical electronic computers. This paper explores some of the wilder claims about adiabatic quantum computation and tries to understand if it can solve unsolvable problems.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Classical Computability</b>	<b>3</b>
<b>3</b>	<b>Hypercomputation</b>	<b>4</b>
<b>4</b>	<b>Adiabatic Quantum Computation</b>	<b>6</b>
4.1	Constructability of AQC Machines . . . . .	7
4.2	The Spectral Gap . . . . .	8
4.3	Kieu and Noncomputability . . . . .	9
<b>5</b>	<b>Objections to AQC Hypercomputability</b>	<b>10</b>
5.1	Probabilistic Nature . . . . .	11
5.2	AQC is equivalent to the Standard Model of Quantum Computation	11
5.3	Infinite Precision . . . . .	12
5.4	The existence of decoy ground states . . . . .	13
5.5	Infinite Search In Finite Time . . . . .	14
<b>6</b>	<b>Conclusion</b>	<b>14</b>

# 1 Introduction

What makes Quantum Computers exciting? Certainly increased computational power, outside of the bounds of megahertz and transistor counts, is interesting. The development of quantum computers capable of manipulating as few as 100 qubits would be capable of cracking encryption codes that would take a classical computer longer than the lifetime of the universe.

But there are limits. While it may appear that quantum computation can offer exponential speed-up, this doesn't apply universally. Scott Aaronson puts it best:

Q: But couldn't quantum computers try all possible solutions in parallel, and thereby solve NP-complete problems in a heartbeat?

A: Yes, if the heart in question was beating exponentially slowly.  
[Aar07]

These limits certainly are not obvious, and to those who haven't studied in depth the quantum computational model, which includes the author up until several months ago, it seems that the speed-up achieved by quantum computers could be unbounded. Why can't quantum computers do computations in an infinite number of alternate universes and return the result to us moments later? I think the standard answer would be that, yes, quantum computers can do an exponential number of computations in a single clock cycle. This is easy to see immediately when considering the superposition created by a Hadamard gate. With  $n$  qubits and  $n$  Hadamard gates we can create a superposition with  $2^n$  terms. But, we lose information contained in that superposition once a measurement is made, and in fact we lose everything except a single term. This is the crucial obstacle when confronting the obtainable speedup when developing quantum algorithms. In short quantum algorithms cannot solve NP-Complete problems in polynomial time. But what about unsolvable problems, can quantum computing help us there?

This question reduces to a more fundamental one. Are there computational limits which are as insurmountable as the speed of light? Most computer scientists would probably say that yes, there are limits to what we can compute. And in fact this limit may be somehow encoded as physical reality the same way mass and energy exist. However there are current theories that this is not the case. Kieu, in a number of papers, has put forth an algorithm which he claims can solve a non-solvable problem. There are a number of issues at play when trying to probe the limits of computability that require multi-disiplinary research. There are ties to computer science, electrical engineering, physics, quantum physics and astrophysics. Several of these aspects will be examined here, and framed within the context of quantum computers.

First, what is the relationship between physical reality and computability? Does physical reality impose strict limits on what is computable? Can we even develop a theory in which the uncomputable things are computable?

Second, does the formalism associated with quantum computation make a difference when it comes to computability? In particular, do Adiabatic Quan-

tum Computers (AQC) offer additional speed-up over conventional quantum computing. What about claims that AQC can not only solve NP-Complete problems quickly, but can theoretically decide undecidable questions in a finite amount of time.

## 2 Classical Computability

In the 1930's Turing introduced the imaginary Turing machine and he along with Church eventually postulated that any function which we would consider to be computable, can be computed by a turning machine. For our purposes the chief limitation imposed by the notion of computability is that a function is computable only if the computation can happen in a finite amount of time. I think it's also interesting to tie Turing machines to reality.

It is widely acknowledged that every physical system corresponds to a computational process, and that every computational process, if applicable, has to be physically and operationally feasible in some concrete realization. [LSS07]

This is important because in considering quantum computation we are harnessing a concrete realization of a computational process. If we can prove, one way or the other, the relationship between NP-Complete problems and quantum computers, and further that  $P \neq NP$ , we have then learned something about the fundamental nature of the universe.

And we can extend this concept to include not just the relationship between  $P$  and  $NP$  but also the relationship between computable and non-computable functions. Very early on in the study of the field Turing was able to show that there existed problems which are unsolvable from the standpoint of a Turing machine. This along with Godel's Incompleteness Theorem effectively demonstrated the limits of mathematical thought. So, given that we know problems exist which we cannot solve, not even in principal, what is the advantage of further discussion. Do computers exist more powerful then Turing machines, so called hypercomputers. Even if they did what could we say about them?

For the purposes of this paper I will be considering a few instances of provably NP-Complete and non-decidable problems.

**3SAT** Satisfiability is the problem of determining whether a given Boolean sentence contains variable assignments such that the sentence evaluates to true. There is no known algorithm for this problem which is better than brute-force search through all the possible variable assignments. 3SAT is a special instance of SAT where each clause of the Boolean sentence contains 3 Boolean literals. An example sentence of this form is

$$(A \vee B \vee C) \wedge (A \vee E \vee F)$$

The general case of satisfiability can be reduced to 3SAT. This problem will come into play with the description of Adiabatic Quantum Computing because of its easy formulation as a particular Hamiltonian.

### **Hilbert's Tenth Problem**

Given a Diophantine equation with any number of unknown quantities and with rational integral numerical coefficients: To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable in rational integers.

In short is there a general algorithm which takes as input a Diophantine equation and produces as output either a set of integer assignments which solves the problem or outputs cannot be solved. A Diophantine equation is a polynomial with solutions restricted to the integers. This problem is of importance because in the 1970's it was proven that a general algorithm for solving Diophantine equations would require a search through an infinite set. As there is no way of determining whether this infinite search will find an answer or not, in finite time, this problem is undecidable.

This problem is described because Kieu uses it due to its equivalency to the Halting problem.

## **3 Hypercomputation**

To better understand the relationship between classical computation and hypercomputation, it is useful to consider the relationship between Euclidean and non-Euclidean geometry. [Ord06]

In order to understand if Adiabatic Quantum Computers can solve Unsolv-able problem it might help to discuss what it even means to solvable an unsolvable problem. It certainly seems to defy logic to even consider the issue as non-Euclidean geometry might have seemed to early geometers. First it might help to clarify terminology. Kieu used the term noncomputable, and Hodges uses classically insolvable. Both of them are referring to perhaps a better term undecidable. But really they are referring to problems which cannot be decided using Turing Machines. The leap from undecidable over Turing machines to unsolvable in general follows from the Turing-Church hypothesis that in order for a problem to be solved, in a manner we might all agree is in fact correct, it must be computable by a Turing Machine. Then it follows that problems which cannot be solved using a Turing Machine cannot in fact be solved by any machine that can actually exist. In order to lay the groundwork for this discussion, consider the following scenario, which might help to identify the relevant issues.

Is it physically impossible to solve the Halting Problem? There have recently been investigations into this question. One of the more interesting cases involved relativistic Turing machines. Take a Turing machine  $M$  and a program

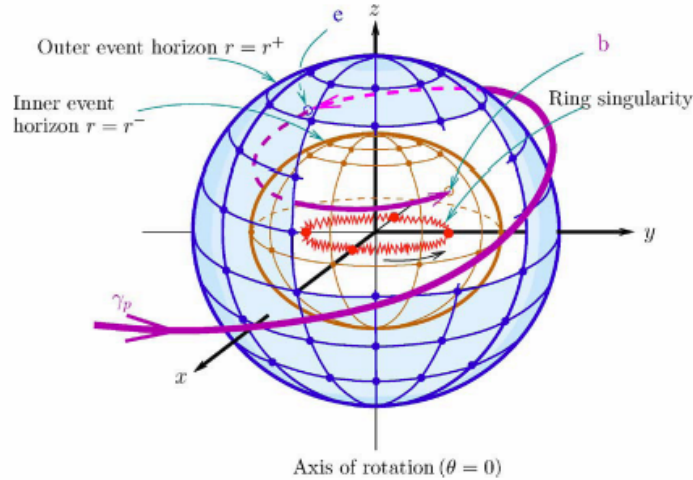


Figure 1: This illustration from [AB] shows how a Kerr rotating black hole consisting of two event horizons and a singularity can support a relativistic Turing machine.

$P$ , which may or may not halt given some input. Take the machine, programmed with  $P$  and set it running on a computer located within a spaceship. Take the spaceship and fire it toward a singularity, e.g. a black hole, with the computer programmed to fire a rocket if the program ever halts. Since the spaceship will experience an infinite amount of time relative to an outside observer, if the program ever completes, it will in fact fire the rocket in finite time relative to an outside observer. We will be able to see the rocket, or otherwise detect some result, in some finite amount of time [Sta08]. However since the spaceship is inside a black hole the event will be undetectable outside the influence of the black hole. This requires that, in order to use this relativistic computer or obtain the results of the calculation, any outside programmer will also have to be inside the event horizon of the black hole in order to detect the outcome of the computation. This is ok, however, because by finding a rotating Kerr black hole, any scientists who want to determine the consistency of the Zermelo-Frankel axioms will be able to lead full lives after they launch their computer, provided they never try to leave the confines of the black hole which they entered [AB].

The study of machines which can perform computations above and beyond classical Turing machines is called Hypercomputing and there have been a number of very recent papers which have attempted to formalize the terms and definitions of this newly emerging field. There have been a number of extensions proposed to classical Turing Machines including [Ord06];

1. Oracles, or an unspecified device which can determine set membership.

2. Random Turing machines, where the Turing machine has access to a random bitstrings.
3. Infinite Time machines, Zeno Turing machines, where each computational steps takes half the time as the one before it.
4. Turing machines with an infinite input or infinite number of states.

Some of these extensions allow the machine to compute more then what a standard Turing machine is capable of. The physical manifestation of one of these machines while certainly unlikely is hopefully not impossible.

This shows that extensions to Turing machines are a valid field of study. Developments in this inter-disciplinary field will drive understanding of the physical limitations imposed upon computational power.

## 4 Adiabatic Quantum Computation

Adiabatic Quantum Computation (AQC) is an attempt to create another model of computation that incorporates quantum physics but does not use the standard model of quantum gates implemented by unitary operators. It instead based upon the adiabatic theorem of quantum physics which states formally that if there exists a quantum system with Hamiltonian  $H_1$  with ground energy state  $\Psi_1$  and we transform the system into  $H_2$ , using the transformation

$$H \frac{t}{T} = (1 - \frac{t}{T})H_1 + \frac{t}{T}H_2$$

then the ground energy state  $\Psi_1$  is transformed into  $\Psi_2$  where  $\Psi_2$  is the ground energy state of the system  $H_2$  [Kie03a, AR04]. This process takes place over time  $T$ . An immediate question one might have is what is the value of  $T$ . Since the total time  $T$  directly relates to the runtime of the process, how can we know that  $T$  can be controlled when solving computational problems? While full discussion of this appears in a later section here we can say that there really isn't enough empirical evidence at this time to fully describe how  $T$  can be controlled. It will even be different from one problem to another and even differ based on the parameters used when solving a problem.

AQC uses this trick of quantum mechanics as a model of information in the following way. A system is purposefully put into known ground state with Hamiltonian  $H_1$ . The answer to some problem is encoded into a Hamiltonian  $H_2$ , for which the ground state may not be able to be calculated. The system is then transformed, according to the adiabatic theorem, into a system with Hamiltonian  $H_2$ . The adiabatic theorem then guarantees that the system will then be in state  $\Psi_2$ , which is the answer to our problem. This is true even when  $H_2$  has any number of possible energy states/eigenvalues. By measuring the system and obtaining the value for  $\Psi_2$  we have answered the question we set out to solve.

What makes this a useful model for computation was put forward in [FGGS00]

Many computationally interesting problems can be recast into an equivalent problem of finding a variable assignment that minimizes an "energy" function.

[FGGS00] apply AQC to 3SAT in the following manner. For a boolean statement with any number of clauses  $C$ , each with 3 variables  $i_C, j_C, k_C$ , define an energy function

$$h_C(i_C, j_C, k_C) = \begin{cases} 0 & \text{if the clause is satisfied by } (i_C, j_C, k_C) \\ 1 & \text{Otherwise} \end{cases}$$

The total energy of the system is then the sum of the energies of each clause in the statement.

$$h = \sum_C h_C$$

By minimizing the energy of this equation  $h$  we can find if any Boolean statement has a satisfying assignment. This can be found by locating the ground energy state of a system with Hamiltonian  $H_p$  where

$$H_p = \sum_c H_{P,C}$$

Each  $H_{P,C}$  represents an encoding of a Boolean clause in the Hilbert space spanned by  $N = 2^N$  basis vector where  $N$  is the number of qubits used in solving the problem. This feature of the algorithm is similar to the standard model of quantum computation. That is, a Boolean statement with  $N$  variables can be encoded into a Hamiltonian with  $N$  qubits, with a space spanned by  $2^N$  basis vectors. If we could calculate the ground state of  $H_p$ , then we would be able to determine if the Boolean clause is satisfiable, or not.

The chief difference, then, between this AQC model and the standard quantum computational model is the area of measurement. In the standard model the exponential state space is usually in a superposition, and any measurement instantaneously forces the system into an eigenvector based upon a probabilistic assignment. There is no way to determine from the measurement what the state of the system was the moment the measurement was made. With AQC, a measurement is made when the system is in a state described by the Hamiltonian  $H_2$ , and because the system was in the ground state  $\Psi_1$  when the process was started the system is guaranteed to be in the ground state  $\Psi_2$  when the process is over. Therefore the measurement should always be happening on a system which is *always* in an eigenstate, and not in a superposition of states.

#### 4.1 Constructability of AQC Machines

There have been a number of claims related to the benefits of AQC over the standard quantum computing model. Without more research it will be difficult to tell the hype from the reality for some time.

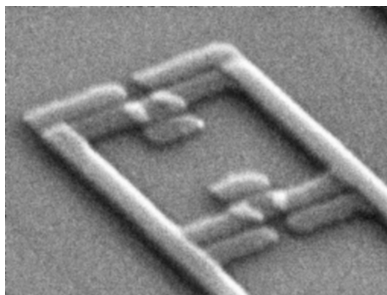


Figure 2: An Electron Microscope image of a flux qubit. These qubits are generated using a voltage source and a resonator tank-circuit for readout of results

One of the potential advantages of AQC is the ease with which quantum computers based on this model can be built. It appears that quantum circuits of the nature can be built using today's technology on something resembling silicon circuits. In 2007 [vIG<sup>+</sup>07] were able to create a three qubit system using flux qubits.

The physical operation of these quantum systems is well beyond the scope of this paper. But in principal a AQC based computer can operate by electrical principals. "By applying a dc bias current through the tank and currents through the lines we can separately change the magnetic flux in the qubits." [vIG<sup>+</sup>07]. This flux manipulation allows the computer to manipulate the Hamiltonian of the system, effecting the adiabatic change.

There is even a commercial company, D-Wave, which claims to have constructed a multi-qubit system based on this principal. They have issued press releases like "World's First Commercial Quantum Computer Demonstrated" and "World's First 28-qubit Quantum Computer Demonstrated Online at Supercomputing 2007 Conference". These claims seem more marketing speak then actual progress, no one has independently verified that they are producing entangled qubits but it is likely they are close to demonstrating a working quantum computer with multiple qubits. One of the principal problems faced with when constructing working machines is that in order to obtain the initial ground state, the quantum computer, or at least the pieces one would like to operate adiabatically, must be cooled almost to absolute zero degrees. To solve this problem D-Wave "cools circuits made of the rare metal niobium to five-thousandths of a degree above absolute zero." [And07]

But even if AQC machines are actually constructed, what kind of performance advantage will they be able to obtain? Current evidence in this regards is somewhat mixed, due to the uncertain nature of the AQC process and what reasonable expectations are for their performance.



## 4.2 The Spectral Gap

One of the interesting results developed as a consequence of AQC is the relationship between the spectral gap and computational complexity. The spectral gap is directly related to the amount of time  $T$  required to a process to be performed adiabatically. This time was directly calculated in [vMV02] as

$$\tau(s) \gg \frac{\frac{dH(s)}{ds}}{g(s)^2}$$

where  $g(s)$  denotes the gap between the smallest eigenvalues of  $H(s)$ , in other words the energy gap between the ground state and the first energy state. It follows that the larger this gap is, the faster the adiabatic process can be. So what determines the size of this gap?

As it turns out, the gap size may be related to the computational difficulty of the problem. In [Mit08], simulated experiments were run using AQC solve different instantiations of 3SAT. 3SAT problems can be described by the relative difficulty as function of the ratio between clauses and variables. [Mit08] developed a

correlation between quantum spectral complexity and 'classical'  
computation complexity and the formulation of a quantitative cor-  
respondence between the two systems.

The implication of this is that like heuristic techniques, there may be cases where an ACQ algorithm performance will differ drastically as a function of the input. Increased computational complexity could lead to increased spectral complexity which in turn increases the time needed for a the adiabatic process.

## 4.3 Kieu and Noncomputability

The chief proponent of claims that AQC can solve NP-Complete and Turing non-decidable problems is Tien Kieu from Swinburne University of Technology in Australia. Kieu presents a general algorithm which he claims is capable of deciding whether an arbitrary Diophantine equation has a solution or not. What follows is a brief summarization of his algorithm. For more details consult [Kie06b, Kie03a, Kie05, Kie03b, Kie06a]. The first step is to develop a Hamiltonian representation  $H_p$  of the Diophantine equation under consideration and then eventually use AQC to evolve a known ground state  $H$  into  $H_p$ . This translation is canonical, for a given Diophantine equation  $D(x_1, x_2 \dots x_K) = 0$  with  $K$  variables,

$$H_p = (D(a_1^\dagger a_1 \dots a_K^\dagger a_K))^2$$

Where  $a$  and  $a^\dagger$  are the creation and annihilation operators for the Fock space. And the operator  $a_1^\dagger a_1$  is the number operator which gives the particular

number of particles in a specific state. Specifically

$$a = \begin{pmatrix} 0 & \sqrt{1} & 0 & 0 & \dots \\ 0 & 0 & \sqrt{2} & 0 & \dots \\ 0 & 0 & 0 & \sqrt{3} & \dots \\ 0 & 0 & 0 & 0 & \dots \end{pmatrix}$$

This is one important difference between Kieu's treatment of AQC and the standard model of quantum computation. Kieu requires that the infinite dimensional Hamiltonians exist in a Fock space, which is a kind of Hilbert space with a variable number of basis vectors. In normal quantum computation, qubits exist in spaces with  $2^N$  basis vectors when operating on  $N$  qubits. To realize Kieu's process requires implementing a physical system having an infinite number of energy levels.

The other important variable defined is the initial Hamiltonian for the system,

$$H_I = \sum_{i=1}^K (a_i^\dagger - \alpha_i^*)(a_i - \alpha_i)$$

for some complex number  $\alpha$

The last variable which needs to be assigned in order for the AQC to perform its magic is the value  $T$ , which is the length of time over which the algorithm needs to operate. There is a problem here, however. In this case we do not "in advance know in general how long is sufficiently long" for the transformation to occur, so we do not know what to set the value  $T$  too. Tieu fixes this up by showing that we can probabilistically identify the ground state.

$$|n_0\rangle \text{ is the ground state of } H_p \Leftrightarrow |\langle \psi | T | n_0 \rangle|^2 > \frac{1}{2}$$

That is, after evolving the system for some time  $T$ , we perform a number of measurements, and we obtain one state with probability more than 0.5, we know that state is the ground state and we're done. Otherwise we continue to increase  $T$ , until we obtain our result. Kieu asserts that he has shown that "we can confidently know that for each Diophantine equation and each set of  $\alpha$ 's, there is a *finite* evolution time after which the adiabaticity condition is satisfied."

So here is the algorithm.

1. Put the system into the ground state with Hamiltonian  $H_i$ , which is a "well known coherent state in quantum optics" [Kie05].
2. Evolve the system for some time  $T$
3. Make  $K$  measurements of the system obtaining states  $|k_i\rangle$ .
4. If any  $|k_i\rangle$  happens when frequency greater than  $\frac{K}{2}$  we are ready to return a solution. If not, return to step 2.

## 5 Objections to AQC Hypercomputability

As might be expected with such a controversial claim, there have been numerous objections leveled against Kieu's ideas. To be fair, he has subsequently attempted to address every objection brought to bear on AQC hypercomputability. The objections can be divided into two categories, technical and theoretical. The technical objections are hard to decipher for an outsider to quantum theory. They involve the specifics of the Hamiltonian used to encode the Diophantine equation as well as the mechanics of the time evolution process as it interacts with the spectral gaps between energy states. The theoretical objections revolve around the dynamics of information theory and the possibility of infinite search in finite time or the equivalency between AQC and the standard model of quantum computation.

### 5.1 Probabilistic Nature

Since the value of total time  $T$  for adiabatic transformation is unknown when the algorithm is started, it is necessary to probabilistically determine when the algorithm is terminated. After  $T$  time has elapsed and 100 measurements are made, a non-ground state measurement may be obtained 51 times. Since this triggers the termination criteria, the algorithm will have returned an incorrect result. Kieu addresses this by appealing to the Weak Law of Large Numbers, by noting that the error is inversely proportional to the number of samples taken. Therefore error can always be made as small as possible by taking a sufficient number of samples.

### 5.2 AQC is equivalent to the Standard Model of Quantum Computation

It was shown in [AvK<sup>+</sup>07] that the adiabatic computation model and the standard model of quantum computation involving unitary gates are polynomial equivalent. What is the effect upon Hypercomputation using AQC. At first it would appear to severely limit the possibilities. We know, for example, that we can simulate quantum computation using a regular digital computer. It follows that a digital computer and a quantum machine can calculate the same things. If AQC and the standard model are polynomial equivalent, then whatever it is calculable by AQC is also calculable by a run of the mill digital computer.

Kieu responds to this by invoking the infiniteness of the Fock space in which the calculations take place as opposed to the finite Hamiltonians used in the standard formalisms. Kieu's approach

... is from the start based on infinite-dimension Hamiltonians acting on some Fock space and also based on the special properties and unique status of their ground states. The unitary operations are then followed as the Schrodinger time evolutions. Even at the Hamiltonian level higher orders of the operators  $a$  and  $a^\dagger$ , i.e. not just

two-body but many-body interactions in a sense, are already present. This proliferation, which is even more pronounced at the level of the time-evolution operators, together with the infinite dimensionality and the unique energetic status of the vacuum could be the reasons behind the ability to compute, in a finite number of steps, what the dimensionally finite unitary operators of the standard quantum Turing computation cannot do in a finite number of steps [Kie03a].

Is this claim convincing? It seems that Kieu has simply replaced an infinite time calculation with an infinite Fock space. The question then seems to become can this Fock space, in practice, be realized in the physical world. This definitely limits the applicability of the algorithm because even Kieu admits,

we cannot have available an unbounded number of levels for the quantum algorithm [Kie03a].

Is this limitation similar in nature to the finiteness of memory used in a Turing machine. After all there are physical limitations on how much memory can be in use by a Turing machine, despite in the infinite ideal. This question likely deserves more research and it seems to be tied to the amount of information that can be physically realized in a single location.<sup>1</sup>

### 5.3 Infinite Precision

There are a number of issues that come into play when considering the effect that precision will have on Kieu's algorithm. First, what will happen when a discrete set, the integers, is translated into a Hamiltonian which has members that take on real values. As Hodges notes [Hod05], if we try translating the value 2 and end up with 2.0000... any error would "wreck" the results of the calculation. Kieu addresses this issue directly, describing the distinction between infinite precision and unbounded precision. While Hodges and other critics claim that infinite precision is required in order to realize the algorithm, Kieu has the opinion

that it may not be a foregone conclusion, as some may have thought, that the physically available precision is not adequate for an implementation of the proposed algorithm.

Here unbounded refers to as much precision as the implementer has the time or desire to achieve. This is certainly not an extremely convincing argument. Kieu also tries to resolve this difficulty with multiple samples, achieving a statistical average inversely proportional to  $\sqrt{N}$  for  $N$  trials. Increasing  $N$  until "we could confidently confirm the (integer-valued) central value."

Along with precision comes the introduction of noise. Since these quantum systems must be kept in the ground states for the Adiabatic transition to occur,

---

<sup>1</sup>As Kieu has a point to every counter-point, he notes that a "finitely prepared qubit can have infinite algorithmic informational theoretic complexity" [Kie05]. This can be achieved by alternating non-compatible observables to generate a (truly) random sequence of bits.

it seems that they can be extremely susceptible to environmental noise. Kieu claims that even "though these fluctuations in the energy levels are ever present and cannot be reduced to zero, there is no physical principle, and hence no physical reason, why their size cannot be reduced to a size smaller than some required scale." What is left to be understood is how the effect of an infinite number of energy states can be managed so that the spectral gap is large enough to allow the algorithm to succeed. And additional work by [Mit08] seems to indicate the computational complexity and spectral gaps are inextricably linked.

#### 5.4 The existence of decoy ground states

Smith in [Smi05] produced a strong critique of Kieu's algorithm, attempting to show that there were situations in which the system undergoing the Adiabatic change failed to stay in the ground state. These situations which he labeled as decoy ground states appear to occur frequently. Smith claimed in every 25 simulations, 1 produced a decoy ground state. Kieu claims in [Kie05] that no matter what the value of  $T$ , there will never be a situation where the probability of measuring a non-ground state is greater than  $\frac{1}{2}$ . This is different than the first argument about the probabilistic nature of the algorithm. There, non-ground states may appear to be measured with frequency greater than  $\frac{1}{2}$ , but this is nothing more than a statistical aberration which will likely disappear with an increasing number of samples.

In detail, Smith chose of  $H_1$  and  $H_p$  values,

$$H_1 = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -\sqrt{2} & 0 & 0 \\ 0 & -\sqrt{2} & 3 & -\sqrt{3} & 0 \\ 0 & 0 & -\sqrt{3} & 4 & -\sqrt{4} \\ 0 & 0 & 0 & -\sqrt{4} & 5 \end{pmatrix}, H_p = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Which is an application of Kieu's algorithm for a Diophantine equation with 1 variable and setting  $\alpha$  to 1. After evolving the system for  $T=13.344$  the system has eigenvalues

$$0.0144, 1.1307, 2.5406, 4.3884, 6.9288$$

Which occur with probability

$$0.0139, 0.9997, 0.0062, 0.0210, 0.0015$$

Here the second energy level 1.1307 has a .9997 percent chance of being measured, and returning a wrong value from the algorithm.

Kieu has a response to this, although it is not entirely satisfying. Since this numerical simulation is required to take place on a real machine, an infinite Fock space cannot be simulated, and Smith chose to truncate the Hamiltonian to 5 basis vectors. Kieu again appeals to the infinite Fock space claiming that,

In general, we suspect that the possibility of the violation [noted by Smith] *is an artifact of the truncation of the Fock space* since it is extremely sensitive to the truncation and associated boundary conditions [Kie06a].

## 5.5 Infinite Search In Finite Time

Grover's search algorithm gives a  $\sqrt{N}$  speedup of unstructured search problems. Isn't it the case that Kieu's algorithm beats that by searching an infinite number of integers for one which satisfies the Diophantine Equation? Kieu responds to this with the argument that finding the solution to a Diophantine equation is a finitely refutable problem. That is, if no solution exists, *within a finite domain* then there is no solution anywhere. Due to the physical nature of the encoding used for the AQC, the adiabatic process somehow "knows" what the finite domain is, based upon ground states available to it.

For example, classical minimization algorithms are unable to distinguish between a global minimum and a local minimum. Minimization algorithms based upon AQC in fact can distinguish between local and global minimum. This fact follows from that Kieu claims, as above, that there are no level crossings in the spectral flow between *any* pair of eigenstates. This allows Kieu's to ultimately reduce to finite search. This selection of an finite domain from an infinite one seems to be a bit magical. It also seems unclear why we are unable to locate this finite domain outside the bounds of quantum mechanics, that is, what exactly is encoded in the nature of the algorithm that allows for this selection, and how can we know its correctness.

## 6 Conclusion

Can AQC compute answers to Turing non-decidable questions, absolutely. Will this computation happen in a finite time, probably. Will the answer computed be correct, probably not. It seems that there are too many physical and theoretical limitations in order to realize the fantastic claims of Kieu with regards to Diophantine equations. It is still an open question of whether or not the universe is constructed in such a way as to prevent Turing non-decidable questions from ever being answered, at least without taking a rocket ship into a black hole. Kieu has addressed, and re-addressed many issues and all complaints against his algorithm but cannot make some of his fundamental assumptions disappear. There seems to be a lot of smoke but only a little fire. Particularly unconvincing is the failure to resolve the number of energy levels required for the algorithm to operate. It seems unclear if physical systems can be generated, without a infinite amount of energy, that are able to be in an infinite number of possible eigenstates. Without operating in this infinite space, the algorithm reduces to a truncated version with a finite number of basis vectors and this case was provably shown to produce decoy ground states.

On the other hand AQC seems like it does have a lot to offer over the

standard model of quantum computation and represents an extremely promising approach to quantum computing. Perhaps its future lies along the same path as some artificial intelligence techniques like neural nets and simulated annealing. These techniques can be extremely useful applied to certain kinds of problems with certain kinds of input, where approximation is acceptable. They currently represent the state of the art when it comes to "solving" NP-Complete problems. They may not give the correct answer 100% of the time, but at least they give *an* answer in polynomial time. If AQC can be shown to have applicability to difficult problems, and outperform other AI techniques, then it wouldn't be surprising to see special purpose AQC machines built that can act as a probabilistic oracle to regular digital computers. Their answers will have to be verified, however it may be possible to generate results with an extremely high degree of accuracy to otherwise intractable problems.

## References

- [Aar07] Scott Aaronson, *Shtetl-optimized: The blog of scott aaronson*, <http://www.scottaaronson.com/blog/>, 2007.
- [AB] Istvn Nmeti A and Gyula Dvid B, *Abstract relativistic computers and the turing barrier q*.
- [And07] Mark Anderson, *Quantum leap: The future of super-fast computing appears on the horizon.*, <http://discovermagazine.com/2007/may/quantum-leap>, 2007.
- [AR04] Andris Ambainis and Oded Regev, *An elementary proof of the quantum adiabatic theorem*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0411152>, 2004.
- [AvK<sup>+</sup>07] Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev, *Adiabatic quantum computation is equivalent to standard quantum computation*, SIAM JOURNAL OF COMPUTING **37** (2007), 166.
- [FGGS00] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser, *Quantum computation by adiabatic evolution*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0001106>, 2000.
- [Hod05] Andrew Hodges, *Can quantum computing solve classically unsolvable problems?*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0512248>, 2005.
- [Kie03a] Tien D. Kieu, *Computing the noncomputable*, Contemporary Physics **44** (2003), 51.

- [Kie03b] ———, *Numerical simulations of a quantum algorithm for hilbert's tenth problem*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0304114>, 2003, p. 89.
- [Kie05] ———, *Hypercomputability of quantum adiabatic processes: Fact versus prejudices*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0504101>, 2005.
- [Kie06a] ———, *On the identification of the ground state based on occupation probabilities: An investigation of smith's apparent counterexamples*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0602145>, 2006.
- [Kie06b] ———, *Reply to andrew hodges*, <http://www.citebase.org/abstract?id=oai:arXiv.org:quant-ph/0602214>, 2006.
- [LSS07] Alexander Leitsch, Guenter Schachner, and Karl Svozil, *How to acknowledge hypercomputation?*, <http://www.citebase.org/abstract?id=oai:arXiv.org:0712.3435>, 2007.
- [Mit08] David R. Mitchell, *Relating computational complexity and quantum spectral complexity*, <http://www.citebase.org/abstract?id=oai:arXiv.org:0811.1801>, 2008.
- [Ord06] Toby Ord, *The many forms of hypercomputation*, *Applied Mathematics and Computation* **178** (2006), 143–153.
- [Smi05] Warren D. Smith, *D.: 2006, three counterexamples refuting kieu's plan for quantum adiabatic hypercomputation and some uncomputable quantum mechanical tasks*, *J.Applied Mathematics and Computation* **187** (2005), 184–193.
- [Sta08] Mike Stannett, *The computational status of physics: A computable formulation of quantum theory*, <http://www.citebase.org/abstract?id=oai:arXiv.org:0805.1443>, 2008.
- [vIG<sup>+</sup>07] S. H. W. van der Ploeg, A. Izmalkov, M. Grajcar, U. Huebner, S. Linzen, S. Uchaikin, Th Wagner, A. Yu Smirnov, A. Maassen van den Brink, M. H. S. Amin, A. M. Zagoskin, E. Il'ichev, and H. G. Meyer, *Adiabatic quantum computation with flux qubits, first experimental results*, <http://www.citebase.org/abstract?id=oai:arXiv.org:cond-mat/0702580>, 2007.



- [vMV02] Wim van Dam, Michele Mosca, and Umesh Vazirani, *How powerful is adiabatic quantum computation?*, doi:10.1109/SFCS.2001.959902, 2002.