Quantum Hypercomputation—Hype or Computation?

Amit Hagar[*], Alex Korolev[†]

February 19, 2007

Abstract

A recent attempt to compute a (recursion–theoretic) non–computable function using the quantum adiabatic algorithm is criticized and found wanting. Quantum algorithms may outperform classical algorithms in some cases, but so far they retain the classical (recursion–theoretic) notion of computability. A speculation is then offered as to where the putative power of quantum computers may come from.

[*] HPS Department, Indiana University, Bloomington, IN, 47405. hagara@indiana.edu

[†] Philosophy Department, University of BC, Vancouver, BC. korolev@interchange.ubc.ca

1 Introduction

Combining physics, mathematics and computer science, quantum computing has developed in the past two decades from a visionary idea (Feynman 1982) to one of the most exciting areas of quantum mechanics (Nielsen and Chuang 2000). The recent excitement in this lively and fashionable domain of research was triggered by Peter Shor (1994) who showed how a quantum computer could exponentially speed–up classical computation and factor numbers much more rapidly (at least in terms of the number of computational steps involved) than any *known* classical algorithm. Shor's algorithm was soon followed by several other algorithms that aimed to solve combinatorial and algebraic problems, and in the last few years theoretical study of quantum systems serving as computational devices has achieved tremendous progress. According to one authority in the field (Aharonov 1998, Abstract),

> we now have strong theoretical evidence that quantum computers, if built, might be used as powerful computational tool, capable of performing tasks which seem intractable for classical computers.

The common view is that quantum algorithms such as Shor's may help to re-describe the complexity space of computational problems, but recently it has been suggested that they may even re–write the abstract notion of computability itself by "computing the (recursion–theoretic) non computable". Here we address one such suggestion that rests on the quantum adiabatic algorithm, pointing out its failure to perform the purported hypercomputation,[1] and use this negative result to speculate further on the so–called 'superiority' of quantum computers.

That *recursive* may not be a natural physical property and that physical processes may not necessarily preserve it has been argued in the past (see, e.g., Pour-el and Richards 1981, Moore 1990, Pitowsky 1990, Hogarth 1994), but this note intends to add nothing new to the general debate on the physical possibility of hypercomputers (Copeland 2002, Shagrir and Pitowsky 2003, Davis 2003). Agnostic as we are with respect to the question whether the club of physical hypercomputers is empty or not, we direct our criticism here at the claim that the quantum adiabatic algorithm is another member of this distinguished, hypothetical or not, club.

The paper is structured as follows. In section (2) we describe the quantum adiabatic "hypercomputer". In sections (3) we flesh out the mistake upon which

[1] The term *Hypercomputer* was coined by Copeland (1998) to denote a machine that can compute non–recursive functions by performing infinite number computational steps in a finite time.

the proposed hypercomputer rests. Section (4) concludes with a speculation on where might the putative power of quantum computers come from.

2 The Quantum Adiabatic "Hypercomputer"

At the 1900 International Congress of Mathematicians, held that year in Paris, the German mathematician David Hilbert put forth a list of 23 unsolved problems that he saw as being the greatest challenges for twentieth–century mathematics. Hilbert's Tenth problem, to find a method (what we now call an algorithm) for deciding whether an arbitrary Diophantine equation has an integral solution, was shown by Yuri Matiyasevich in 1970 to be Turing undecidable. If (and only if) one could solve it, one could also solve the Halting problem.

For the last 4 years, in 10 papers posted to the quant–ph section of the Los Alamos preprints archive (some of which were also published in printed journals such as Kieu 2002; 2003; 2004; 2005) Tien D. Kieu, has been claiming to have a scheme according to which, in principle, physical quantum systems could be used to solve this prototypical computationally undecidable problem in finite time.

In order to appreciate Kieu's scheme, let us first set the stage. A well-known theorem of quantum mechanics, namely, the adiabatic theorem (See, e.g., Messiah 1961, pp. 739–746), was recently harnessed by a group of physicists from MIT (Farhi *et al.* 2000) to develop a novel quantum algorithm. Their aim was to solve in polynomial time certain decision problems that are believed to be NP–hard, and in so doing to vindicate the putative exponential superiority of quantum algorithms over their classical counterparts.[2]

The crux of the quantum adiabatic algorithm lies in the possibility of encoding a specific instance of a given decision problem in a certain Hamiltonian.[3] One then starts the system in a certain quantum state—the lowest energy state of the system, known as the ground state—of another Hamiltonian which is easy to construct, and slowly evolves the system in time towards the interpolated and desired Hamiltonian. According to the quantum adiabatic theorem and given certain

[2] Irrespective of the criticism presented here, one should bear in mind that it is still an open question whether the proposed quantum adiabatic algorithm does indeed yield an exponential speed–up. See. e.g., Van Dam, Mosca and Vazirani (2002) and Reichardt (2004). We return to this point in section 4.

[3] This can be done by capitalizing on the well-known fact that any decision problem can be derived from an optimiztion problem by incorporating into it a numerical bound as an additional parameter.

conditions (See Appendix), the result of such physical process is another energy ground state that encodes the solution to the desired decision problem.

More precisely, one encodes the solution to a decision problem in the ground state of a Hamiltonian $H_{\rm P}$ which is diagonal in the computational basis and is a member of a one-parameter family of Hamiltonians $\tilde{H}(s)$ varying smoothly for $0 \leq s \leq 1$, and then sets $H(t) = \tilde{H}(t/T)$ so that T (the run–time of the algorithm) governs how slowly H varies. One now defines the instantaneous eigenstates and eigenvalues of $\tilde{H}(s)$ by

$$H(s)|j;s\rangle = E_j(s)|j;s\rangle \ , \tag{1}$$

with

$$E_0(s) \leq E_1(s) \leq \cdots \leq E_{N-1}(s) \ , \tag{2}$$

where N is the dimension of the Hilbert space. Suppose $|\psi(0)\rangle$ is the ground state of $\tilde{H}(0)$, that is,

$$|\psi(0)\rangle \ = \ |j=0; s=0\rangle \ . \tag{3}$$

According to the adiabatic theorem, if the gap $g_{\rm min}$ between the two lowest levels, $E_1(s) - E_0(s)$, is strictly greater than zero for all $0 \leq s \leq 1$, then

$$\lim_{T\to\infty} \Big|\langle j=0; s=1|\psi(T)\rangle\Big| = 1 \ . \tag{4}$$

where $|\psi(t)\rangle$ is the state at time t.

One next chooses an Hamiltonian $H_{\rm I}$ which is easy to construct and which does not commute with $H_{\rm P}$ (otherwise one would have to encode the solution already in the initial Hamiltonian!) and defines the interpolating Hamiltonian to be

$$\tilde{H}(s) = (1-s)H_{\rm I} + sH_{\rm P} \ , \tag{5}$$

so that

$$H(t) = \left(1 - \frac{t}{T}\right)H_{\rm I} + \frac{t}{T}H_{\rm P} \ . \tag{6}$$

Under these conditions, and in the adiabatic limit, i.e., for T long enough, the evolution from $t = 0$ to $t = T$ (starting in the ground state of $H_{\rm I}$) will lead to the ground state of $H_{\rm P}$ and to the solution of the problem.

Kieu's insight was to harness the novel quantum adiabatic algorithm to solve another decision problem, namely, Hilbert's Tenth. His idea was that one can capitalize on the infinite dimensionality of the Hilbert space that 'accompanies' every quantum system in order to perform *in parallel* infinite computational steps in a finite time—a task that a hypercomputer, whether classical or quantum, is supposed to be capable of performing.

Kieu designed the target (interpolated) Hamiltonian as to mimic the form of the left–hand–side squared of the original Diophantine equation. This, in turn, guaranteed the existence of a global minimum: The Diophantine equation has at least one integer solution if and only if the final ground state of the target Hamiltonian is zero, and has no integer solution otherwise. Next, Kieu claimed to have *proven* an ingenious probabilistic criterion that allows one, by measuring $H_{\rm P}$, to identify whether the quantum system has indeed reached its ground state, no matter what T is.[4] If not, according to Kieu, one needs only to enlarge the evolution time T and iterate the algorithm, until the ground state (which is ensured to exist through the boundedness of $H_{\rm P}$) is achieved.

Let us consider a particular example, say, the equation

$$D = (x+1)^3 + (y+1)^3 - (z+1)^3 + cxyz = 0, \qquad c \in Z \ , \tag{7}$$

with unknowns x, y, and z. To find out whether this equation has any non-negative integer solution by quantum algorithms, it requires the realisation of a Fock space. Upon this Hilbert space, we construct the Hamiltonian corresponding to (7)

$$H_{\rm P} \;=\; \left((a_x^\dagger a_x+1)^3 + (a_y^\dagger a_y+1)^3 - (a_z^\dagger a_z+1)^3 + c(a_x^\dagger a_x)(a_y^\dagger a_y)(a_z^\dagger a_z)\right)^2 ,$$

which has a spectrum bounded from below—semidefinite, in fact.[5] Note that the operators $N_j = a_j^\dagger a_j$ have only non-negative integer eigenvalues n_j, and that $[N_j, H_{\rm P}] = 0 = [N_i, N_j]$ so these observables are simultaneously measurable. For some (n_x, n_y, n_z) the ground state $|g\rangle$ of the Hamiltonian so constructed has the properties

$$\begin{aligned} N_j|g\rangle &= n_j|g\rangle, \\ H_{\rm P}|g\rangle &= \left((n_x+1)^3 + (n_y+1)^3 - (n_z+1)^3 + cn_xn_yn_z\right)^2|g\rangle \equiv E_g|g\rangle, \end{aligned}$$

[4] According to Kieu (e.g., in his 2005, 178), this criterion amounts to excluding any state other than the ground state from occupying the energy spectrum of $H_{\rm P}$ with probability $> 1/2$ for any $T > 0$. It is noteworthy that in all of his papers Kieu offered no analytic proof for this criterion, only a simple example in which such criterion is indeed satisfied.

[5] The creation operators $a_x^\dagger$, $a_y^\dagger$ and $a_z^\dagger$ are similar to those of the 3D simple harmonic oscillator.

$$\begin{aligned} [a_j, a_j^\dagger] &= 1 \qquad \text{for } j = x, y, z, \\ [a_k, a_j] = [a_k, a_j^\dagger] &= 0 \qquad \text{for } j \neq k \ . \end{aligned} \tag{8}$$

Thus, after enough iterations, a projective measurement of the energy E_g of the ground state $|g\rangle$ will yield the answer for the decision problem: the Diophantine equation has at least one integer solution if and only if $E_g = 0$, and has no solutions otherwise. (If $c = 0$ in our example, we know that $E_g > 0$ from Fermat's last theorem.)

If there is one unique solution then the projective measurements of the observables corresponding to the operators N_j will reveal the values of various unknowns. If there are many solutions, finitely or infinitely as in the case of the Pythagoras theorem, $x^2 + y^2 - z^2 = 0$, the ground state $|g\rangle$ will be a linear superposition of states of the form $|n_x\rangle|n_y\rangle|n_z\rangle$, where (n_x, n_y, n_z) are the solutions. In such a situation, the measurement may not yield all the solutions. However, finding all the solutions is not the aim of a decision procedure for this kind of problem.

Notwithstanding this, measurements of N_j of the ground state would always yield some values (n_x, n_y, n_z) and a straightforward substitution would confirm if the equation has a solution or not. Thus the measurement on the ground state either of the energy or of the number operators will be sufficient to give the result for the decision problem.

Since the final Hamiltonian (designed as to mimic the form of the left–hand–side squared of the original Diophantine equation) has an integer spectrum and is bounded from below (i.e., there exists, by construction, a global minimum for H_{P}), the evolution time of Kieu's algorithm is finite. Thus, it appears that, at least in theory, Kieu's hypercomputer does indeed work: Given that the algorithm purports to find a global energy minimum, "all" one needs to do in order to compute the (recursion–theoretic) non–computable is to let the system evolve *slowly enough*, measure its energy, and iterate this procedure until a ground state is achieved with probability $> 1/2$ and an answer to the decision problem is found.

A major breakthrough in computer science? A vindication of the superiority of quantum computers over their classical counterparts? Unfortunately, neither is true. The next section explains why.

3 How Slow is *Slowly Enough*?

We now proceed to show that the proposed quantum adiabatic algorithm cannot compute a (recursion–theoretic) non–computable function.

3.1 Mind the Gap

A crucial ingredient in the adiabatic algorithm is the energy gap between the ground state E_0 and the next excited state E_1:

$$g_{min} = \min_{0 \leq t \leq T} (E_1(t) - E_0(t)). \tag{9}$$

This gap controls the evolution time of the algorithm, in the exact following way (see Appendix):

$$T \gg 1/g^2_{min}. \tag{10}$$

The problem is that in the absence of a detailed spectral analysis, *in general* nobody knows what g is, how it behaves, or how to compute it!

Now some of the fanfare in Kieu's papers is built around the idea that there *always* exists such a gap and that the computation *halts* in any case (since the final Hamiltonian H_{P}, by construction, has an integer spectrum and is bounded from below). We set aside the issue of the feasibility of the manufacturing of such a Hamiltonian, which appears to require infinite precision (Hodges 2005. See also below), but even if we grant such (possibly unfeasible) manufacturing capacities, their merit is still questionable: Classically too there may always exist a halting time, only that it is not computable. This is easiest to appreciate in the case of Turing's halting problem: Consider all Turing machines with k states; throw away all those that fail to stop on the input 1; among the others take the one that runs longest; call the number of steps of that machine $T(k)$. Now we "know" that in order to decide whether a machine with k states stops on the input 1, we have to wait $T(k)$ steps. But of course we don't really know, because $T(k)$ is not computable, growing faster than any recursive function.

What Kieu is doing is to define an adiabatic process whose time is of that order (and whose gap g is therefore incomputably small). The fact that there is some T which will do the job is not a big deal (nor is, therefore, the fact that we can use finite but unbounded dimensional Hilbert spaces for each instance). Indeed, if someone told us what $T(k)$ is, we would not have needed infinitely many steps to complete the job classically.

With this gap in mind, we can now think of the following problem: For each given running-time of the algorithm T we have to come up with a process whose rate of change is $\approx T^{-1}$. Question: How do we know that we are implementing the correct rate of change *while $H(t)$ is evolving*? Apparently, by being able to measure differences of order T^{-1}, that is, having a sensitive "speedometer". When the going gets rough we approach very slow speeds of the order of $\sim T^{-1}(k)$, which begs the question, since we can then compute $T(k)$ using our "speedometer"; no fancy quantum computer is needed. If we don't have a "speedometer", then even if we decided to increase the running-time from T to, say, $T + 7$, we will have no clue that the machine is indeed doing $(T + 7)^{-1}$ km/h and not T^{-1} km/h. In this case, clearly, Kieu's algorithm cannot be implemented since we will never know how slowly we should evolve our physical system. But then we will also fail to fulfill the adiabatic condition which ensures us that once we have reached the desired final Hamiltonian, its ground state encodes the solution.

Kieu may argue in response that his (allegedly proven) ingenious probabilistic criterion (along with the iteration of the algorithm) allows him to detect whether the ground state was achieved, that is, whether the algorithm has indeed evolved adiabatically in order to ensure a meaningful result when reading of the energy eigenstate of H_{P}. His idea seems to be the following: *In general*, when one performs such an adiabatic cooling, one doesn't meet this probabilistic criterion (that ensures that it is only the ground state which will appear with probability $>$ $1/2$ upon measuring H_{P}) for *any* state—applying the number operator gives lots of different answers when one repeats the experiment, and none of them comes up more than half of the time. In this case one simply doubles the running time and tries again, and so on. Kieu's claim, call it the HALF CLAIM, is that because there exists (by construction) a global minimum, and some respective correct value of T that makes the evolution slow enough, one is bound to have success eventually.

Now if the HALF CLAIM were true, it would have been a remarkable achievement. To see this, recall that the adiabatic theorem is a *necessary* condition for tracking the ground state. In other words, it ensures that the system's evolution will track the ground state only when certain conditions are met, and only in the adiabatic limit, i.e., when $T \to \infty$. By claiming that, *no matter what T is*, no single state other than the ground state will occupy the energy spectrum with probability $> 1/2$, Kieu is in fact claiming to have proven a theorem much stronger than the adiabatic theorem, which, all by itself, says nothing about non–adiabatic evolutions.

Intuitively, then, it would not be at all surprising if the HALF CLAIM turned out to be false. And unfortunately, as it turns out, the HALF CLAIM *is* false! Although it is true in the adiabatic limit (when $T \to \infty$) and, for a finite T, even in very special (and very simple) cases of two– and three–dimensional Hamiltonians (which happen to be those picked up by Kieu in his so called 'numerical simulations' that accompanied the HALF CLAIM), in general it appears that for a finite T, 'decoy' excited states will occupy the energy spectrum with much higher probability (sometimes even greater than 97%!) than the desired ground state of

H_{P} in dimensions higher than three.[6] But if the HALF CLAIM is false, then the dream of the quantum adiabatic hypercomputer evaporates.

3.2 The Same Old Story (Told Quantum Mechanically)

In order to see what is left of the quantum adiabatic hypercomputer, stripped as it is from the HALF CLAIM, let us first remind ourselves what undecidability means in a typical classical setting.

Suppose we have a black box implementing some function (unknown to us) with a global minimum; it takes natural numbers as input and produces natural numbers as output according to some rule hidden inside the box. Assuming that all we can do is to call this function (use the black box) as many times as we wish (plus some thinking), is it possible to find the function's global minimum? The answer is clearly no, but it is important to see exactly why.

In trying to locate a global minimum we can proceed either systematically, by going over each consecutive natural number starting from 0, feeding it into the box and recording the corresponding output, or in some more complicated deterministic or probabilistic manner. At each step, out of all arguments we have checked so far we keep those that minimize the function and discard the others; the former are the global minimum candidates. Note that, if we proceed systematically, sooner or later, after a finite number of steps (number of function's callings), we will always reach a function's global minimum (as we know a global minimum exists). This knowledge (of the fact that we will eventually stumble upon a global minimum), however, adds next to nothing to solving our task. The problem, obviously, is that, even if we have just reached an actual (non–zero) global minimum, there is no way for us to identify it *as such*. Given the resources we have, we can never be sure whether the function does not take a yet smaller value on the next step.

Thus the fact that we will always reach a global minimum in a finite number of steps is of no help to us. The problem is undecidable only due to our principal inability to identify a global minimum *as such*. Logically, the reason for this undecidability is that defining the property of being a *global* minimum involves quantification over an infinite domain: we say that the function f reaches its global minimum at a point $n_0 \in \mathbb{N}$ iff $\forall n \in \mathbb{N} : \; f(n_0) \leq f(n)$. Trying to

[6] Smith (2005) constructs 3 counterexamples to the HALF CLAIM, thus proving its falsity. We thank Andrew Hodges for pointing out Smith's result to us when writing the final version of this paper.

identify a global minimum as such by brute–force search would require checking the inequality infinitely many times, thus undecidability.

Note, however, that one has to distinguish between *actual* infinity and *potential* infinity (i.e., unboundedness) . In the context of measurement accuracy, for example, the former calls for a measurement of all the binary digits of a real number "at once", as it were. The latter, by contrast, calls for an increasing unbounded degree of accuracy with *different* measurement instances. Kieu's algorithm involves *both* notions: First, it purports to perform a brute–force search on the entire infinite domain in a finite time.[7] Second, it involves a decision procedure that iterates unboundedly in order to decide whether the brute–force search was done correctly. For those who incline to reject the physical possibility of hypercomputers, e.g., Davis (2003), the first feature is sufficient to rule out the algorithm. But since we have declared agnostics with respect to the question of the physical existence of hypercomputers, the issue at stake is, in our view, not actual infinity, but rather potential infinity, or unboundedness.

Put differently, to our mind the reason behind the failure of Kieu's algorithm lies not just in its requirement for infinite precision.[8] Moreover, the fact that the global minimum for the 'computed' function exists by construction (which ensures a non–zero energy gap and hence a finite evolution time) is of no consequence. Rather, it is the fact that this finite time is *unbounded* which kills the algorithm. And since Kieu, while guaranteeing that the brute–force search will eventually halt, fails to supply a criterion that would allow one to identify whether or not the algorithm has halted on the global minimum, the whole construction, despite his aspirations, lacks the ability to identify a global minimum *as such*. The problem is thus no different than any other corresponding classical case of undecidability, and quantum mechanics adds nothing to its solution.

Put another way, the gist behind the adiabatic algorithm is that after a sufficiently long evolution time, one is certain to have retrieved the correct result of the decision problem just by performing a measurement on the ground state. However, when the evolution time is unknown, a non–zero energy reading upon a measurement of a final state can be interpreted in two very different ways. On one hand, it may be said to be an eigenvalue of an excited state. In such case, clearly,

[7] As an anonymous referee has correctly remarked, if the evolution of the algorithm took place in a Hilbert space with a fixed *finite* number of dimensions, then g_{min} would in principle be computed to any desired accuracy.

[8] For a criticism based on this objection see Hodges (2005). Although we agree with Hodges that Kieu's hypercomputer may require infinite precision to be *actually manufactured*, we prefer to focus here on what is or is not possible *in theory*.

the evolution was non–adiabatic, hence one must iterate the algorithm with another, longer, evolution time. On the other hand, it may be said to be an eigenvalue of the ground state. In such case, clearly, the algorithm has performed correctly and one has a (negative) answer to the decision problem. But since one cannot check a negative answer to a classically undecidable problem, how can one tell, without knowing T in advance, that this negative 'answer' is indeed correct, that is, that no iterations are needed anymore? Without a criterion for distinguishing a ground state from all other excited states which is *independent* of the knowledge of the adiabatic evolution time T, one simply can't.

4 What Is *Quantum* In Quantum Computing?

Summarizing, Kieu's quantum adiabatic "hypercomputer" fails for a simple reason: As one should intuitively expect from an algorithm that relies on the adiabatic theorem alone, and as shown by Smith (2005), even if the adiabatic conditions are satisfied, then for a *finite* running time $t < T$, there is *in general* no guarantee that the final energy state will be the ground state. Consequently, there is no way to distinguish a 'decoy' excited state from a non–zero ground state, i.e., there is no way to identify a global minimum *as such*. Repairing this failure requires knowing in advance the exact adiabatic running time T (or, equivalently, the precise behavior of the energy gap throughout the time–evolution of the algorithm), which in Kieu's case is just another undecidable problem.

It is crucial to note, however, that since we have distinguished here between actual infinity and potential infinity, or unboundedness, our criticism is directed not at the possibility of physical hypercomputation *per se*, but rather at the claim that Kieu's quantum adiabatic algorithm can be harnessed to compute the (recursion–theoretic) non–computable. Contrary to other cases of physical hypercomputation (e.g., Hogarth 1994), arriving at the solution *here* requires solving the same type of undecidable problem Kieu has originally set forth to solve with his algorithm.

We would like to conclude with a speculation about the so called 'superiority' of quantum computers over their classical counterparts. As mentioned in the introduction, quantum computing has become an industry and one of the most fascinating domains of quantum theory. Notwithstanding this excitement, and apart from the almost insurmountable problem of practically realizing and implementing a large scale quantum computer (Unruh 1995, Haroche and Raimond 1996), a crucial theoretical question remains open, namely—what physical resources are responsible for the putative power of quantum computing? Put another way, what are the essential features of quantum mechanics that allow one to solve problems or simulate certain systems far more efficiently than on a classical computer? It is also remarkable that the relevance of features commonly thought essential to the superiority of quantum computers, e.g., entanglement and interference (Josza 1997), is recently being questioned (Linden and Popescu 1999, Biham 2004). Moreover, even if these features *do* play an essential role in the putative quantum speed–up, it is still unclear *how* they do so (Fortnow 2003).[9]

The question 'what is *quantum* in quantum computing?', theoretical as it may seem, has an enormous practical consequence. One of the embarrassments of quantum computing is the fact that, so far, only one algorithm has been discovered, namely Shor's quantum Fourier transform, for which a quantum computer is significantly faster than any *known* classical one.[10] It is almost certain that one of the reasons for this scarcity of quantum algorithms is related to the lack of our understanding of what makes a quantum computer quantum.

It is often said that it is 'quantum parallelism' which is behind the potential speed–up of quantum computing. But things are not that simple. The elusive character of the physical resource responsible for the quantum speed–up can be nicely demonstrated with the following example. Consider a solution of a decision problem, say SATISFIABILITY, with a quantum algorithm based on a Fourier transform. What we are given here as input is a proposition in the propositional calculus and we have to decide whether it has a satisfying truth assignment. As Pitowsky (2002) shows, the quantum algorithm appears to solve this problem by testing all 2^n assignments "at once", yet this quantum 'miracle' helps us very little since any measurement performed on the output state collapses it, and if there is one possible truth assignment that solves this decision problem, the probability of retrieving it is 2^{-n}, just as in the case of a probabilistic classical Turing machine which guesses the solution and then checks it.

Pitowsky's conclusion is that in order to enhance computation with quantum

[9] Ironically, what puzzles philosophers and computer scientists may be regarded as quite simple from the physicist's point of view: the quantum adiabatic model of computation has translated the question of where the power of quantum computers lies into the language of spectral gaps. Physicists, as it happens, have been analyzing these gaps for almost a century, but not for the purpose of deciding whether these gaps decrease polynomially or exponentially as the number of particles increases to infinity. . .

[10] Recall, however, that the classical complexity of FACTORING is unknown, and it will not be totally surprising if a classical polynomial time algorithm for this task is found. From a foundational perspective the quadratic speed–up in Grover's search algorithm (1996) is better than Shor's since it is demonstrable. Classical search does require (worst case) linear time in the size of the data.

mechanics we must construct 'clever' superpositions the measurement of which increases the probability of successfully *retrieving* the result far more than that of a pure guess. Shor's algorithm is a (unique?) example of both *a construction* of such a 'clever' superposition and of *a retrieval* of the solution that can be done in polynomial time.[11] The quantum adiabatic algorithm (Farhi *et al.* 2000) may give us similar results, contingent upon the existence of an energy gap that decreases polynomially with the input.[12]

Generalizing Pitowsky's example, we would like to conjecture that the putative power of quantum computers, at least in the quantum circuit model, lies in the quantum state, rather than in its dynamical evolution.[13] In other words, since the whole of the quantum computation can be simply represented as a *single* unitary transformation from the input state to the output state, the quantum speed–up is unlikely to result from the Schrödinger's equation. Rather, the quantum speed–up might result from a very specific quantum state (the 'clever' superposition—to use Pitowsky's term) that can store *retrievable* information in its phase relations much more efficiently than any classical state.

Another argument in favor of this conjecture (Unruh, personal communication) is that the Hilbert subspace 'visited' during a quantum computational process is, at any moment, a linear space spanned by all of the vectors in the total Hilbert space which have been created by the computational process up to that moment. But this Hilbert subspace is thus a subspace spanned by a polynomial number of vectors and is thus at most a polynomial subspace of the total Hilbert space. A classical simulation of the quantum evolution on a Hilbert space with a polynomial number of dimensions (that is, a Hilbert space spanned by a number of basis vectors which is polynomial in the number of qubits involved in the computation), however, can be carried out *prima facie* in a polynomial number of classical computations. Were quantum *dynamics* important to the efficiency of quantum computing, the latter could be mimicked in a polynomial number of steps with a classical computer.

[11] Note that Shor's algorithm involves three major steps: First, one creates the 'clever' entangled state with a set of unitary transformations. The result of the computation—a global property of a function—is now 'hidden' in this state; Second, in order to retrieve it, one projects it on a subspace of the Hilbert space, and finally one performs another set of unitary transformations in order to make this result measurable in the original computational basis. *All* these steps count as *computational* steps as far as the efficiency of the algorithm is concerned. We thank Jeff Bub for discussion on this point.

[12] See Hagar and Tamir (in preparation).

[13] Here we use the Schrödinger representation in which the dynamical variable is the quantum state that evolves in time according to Schrödinger's equation.

Of course, anybody familiar with the standard quantum computational model would reasonably interpret this statement as saying that polynomial–time quantum computation is no more powerful than classical polynomial–time computation. We do not claim this, but instead would like to stress the following point: One does not end a quantum computation with an *arbitrary* superposition, but aims for a very special, 'clever' state. Quantum computations cannot *always* be mimicked with a classical computer because the characterization of the computational subspace of certain quantum states is difficult, and it seems that certain special quantum states cannot be *classically* represented as vectors derivable via a quantum computation in an optimal basis, or at least that one cannot do so in such a way that would allow one to calculate the outcome of the final measurement made on those states.

If our conjecture holds, then in the quantum circuit model one should count the number of computational steps in the computation not by counting the number of global transformations of the state, but rather by counting the number of one– or two–qubit local transformations that are required for the *creation* of the 'clever' quantum states that ensures computation speed–up in Pitowsky's sense (i.e., states that would allow an efficient retrieval of the result in polynomial time). The trick would then be to perform these local one- or two-qubit transformations in polynomial time, and more important, to find a *physically realizable* Hamiltonian that can simulate them. It is here where the physical power of quantum computing may be found.

There is, however, an important caveat. As our criticism of the quantum adiabatic hypercomputer shows, the construction of 'clever' quantum states for the purpose of solving an intractable computational task, or the retrieval of the solution that is 'hidden' in these quantum states may sometimes require solving another, as intractable a task. Indeed, in the case of the adiabatic algorithm, Reichardt (2004) has shown that there are simple problems for which the algorithm will get stuck in a local minimum, in which there are exponentially many eigenvalues, all exponentially close to the ground state energy, and applying the adiabatic theorem, even for these simple problems, will take exponential time. Thus as much as the detection of a global minimum *as such* is an undecidable task, escaping local minima may be yet another NP–hard task, and we are back to square one.[14]

[14] Since the adiabatic model was shown to be computationally equivalent to the quantum circuit model (that is, each model can simulate the other with only polynomial, i.e., modest, overhead of resources, namely, number of qubits and computational steps. See Aharonov *et al.* 2004), it is plausible that this type of problem will recur, albeit in a different form, in the standard quantum

A The Adiabatic Theorem

Consider a quantum system described in a Hilbert space $\mathcal{H}$ by a smoothly time-dependent Hamiltonian, $H = H(t)$, for t ranging over $[t_0, t_1]$. Let $U(t)$ be the time-evolution operator from time t_0 to $t \in [t_0, t_1]$. We denote $T = t_1 - t_0$. If the following conditions are satisfied:

1. For any $t \in [t_0, t_1]$, $H(t)$ has a purely discrete spectrum with eigenvalues denoted $E^1(t), E^2(t), \ldots, E^i(t), \ldots$.

   We denote $P^1(t), P^2(t), \ldots, P^i(t), \ldots$, respectively, the projection operators on the eigenspaces.

2. The eigenvalues and the projectors are assumed to be continuous functions of t and there is no level crossing throughout the transition, i.e., the instantaneous eigenvalues remain distinct;

$$\forall t \in [t_0, t_1], \quad E^i(t) \neq E^j(t)\ if\ (i \neq j). \tag{11}$$

3. $\frac{d}{dt}P^i$, $\frac{d^2}{dt^2}P^i$ exist and are bounded and are piecewise continuous in the whole interval $[t_0, t_1]$.

Then if the system is initially in the energy state

$$|\Phi^i_a\rangle \in P^i_a(t_0)\mathcal{H}\ , \tag{12}$$

that is,

$$H(t_0)|\Phi^i_a\rangle = E^i(t_0)|\Phi^i_a\rangle\ , \tag{13}$$

then

$$\lim_{T\to\infty} U(t)|\Phi^i_a\rangle = P^i(t) \lim_{T\to\infty} U(t)|\Phi^i_a\rangle. \tag{14}$$

That is, for $T \to \infty$, if the system starts at t_0 in an eigenstate corresponding to the energy $E^i(t_0)$ it will evolve (up to a phase) to an eigenstate corresponding to $E^i(t)$ at t.

In the special case where $E^i(t_0)$ is the ground state, the adiabatic theorem ensures that in the limit $T \to \infty$ the system will remain in the ground state through

circuit model.

its time-evolution. Although in practice T is always finite, the more it satisfies a minimum ‘energy gap’ condition, the less the system will deviate from the ground state. The energy gap condition states that there must exist a non-zero gap, g, between the ground state and the first excited state in any given time, and that $T \gg 1/g^2$. What governs the efficiency of the quantum adiabatic algorithm is thus the rate in which the energy gap between the ground state and the next excited state decreases with the increasing dimension of the Hamiltonian, i.e., with the size of the input.

Acknowledgements

We thank Itamar Pitowsky, Bill Unruh, Bill Demopoulos and an anonymous referee for helpful comments. We also thank Andrew Hodges for an extensive discussion.Earlier versions of this paper were presented in the ESF Workshop on Quantum Information (Sardegna, September 2004); The Sigma Club Seminar (LSE, March 2005) and the ECAP Workshop on Quantum Information (Lisbon, August 2005) and the Foundations of Physics Seminar in UMD (Maryland, October 2005). AH is grateful for financial support from the Alexander von Humboldt Foundation, the Federal Ministry of Education and Research and the Program for the Investment in the Future (ZIP) of the German Government through a Sofja Kovalevskaja Award. AK is grateful for financial support from the UBC Li Tze Fong Memorial Fellowship.

References

[1] Aharonov, D. (1998), Quantum Computing, in *Annual Review of Computational Physics* VI, Singapore: World Scientific. See also quant-ph/9812037.

[2] Aharonov, D., *et al.* (2004), Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation, Archive Preprint quant-ph/0405098.

[3] Biham, E., *et al.* (2004), Quantum Computing Without Entanglement, *Theoretical Computer Science*, 320, pp. 15–33.

[4] Copeland, J. (2002), Super Turing Machines, *Complexity* 4, pp. 30–32.

[5] Copeland, J. (2002), Hypercomputation, *Minds and Machines* 12, pp. 461–502.

[6] Davis, M. (2003), The Myth of Hypercomputation, in C. Teuscher (Ed.) *Alan Turing, Life and Legacy of a Great Thinker*, Springer: New York

[7] Farhi, E., *et al.* (2000), Quantum Computation by Adiabatic Evolution, Archive Preprint quant-ph/0001106.

[8] Feynman, R. (1982), Simulating Physics with Computers, *International Journal of Theoretical Physics*, 21, pp. 467–488.

[9] Fortnow, L. (2003), One Complexity Theorist's View of Quantum Computing, *Theoretical Computer Science*, 292, pp. 597–610.

[10] Grover, L. (1996), A Fast Quantum Mechanical Algorithm for Database Search, *Proc. 28th ACM Symp. Theory of Computing*, pp. 212-219.

[11] Hagar, A. and Tamir, B. (In Preparation), Simulating Quantum Computation—Some Good and Some Bad News.

[12] Haroche, S. and Raimond, J.M. (1996), Quantum Computing: Dream or Nightmare?, *Physics Today* 8, pp. 51–52.

[13] Hodges, A. (2005), Can Quantum Computing Solve Classically Unsolvable Problems?, Archive Preprint quant-ph/0512248.

[14] Hogarth, M. (1994), Non-Turing Computers and Non-Turing Computability, *PSA* 94(1), pp. 126–138.

[15] Jozsa, R. (1997), Entanglement and Quantum Computation, in S. Hugget et. al (eds.) *Geometric Issues in the Foundations of Science*, Oxford: Oxford University Press. See also quant-ph/9707034.

[16] Kieu, T.D. (2002), Quantum Hypercomputability, *Minds and Machines* 12, pp. 541–561.

[17] Kieu, T.D. (2003), Computing the Noncomputable, *Contemporary Physics*, 44, pp. 51–71.

[18] Kieu, T.D. (2004), A Reformulation of Hilbert's Tenth Problem through Quantum Mechanics, *Proceedings of the Royal Society* A 460, pp. 1535–1545.

[19] Kieu, T.D. (2005), An Anatomy of a Quantum Adiabatic Algorithm that Transcends the Turing Computability, *International Journal of Quantum Information* 3(1), 177–183.

[20] Linden, N. and Popescu, S. (1999), Good Dynamics versus Bad Kinematics: Is Entanglement Needed for Quantum Computation?, *Physics Review Letters*, 87(4), 047901. See also quant-ph/9906008.

[21] Messiah, A. (1961), *Quantum Mechanics*, Volume II, pp. 744–745. New York: Interscience Publishers.

[22] Moore, C. (1990), Unpredictability and Undecidability in Dynamical Systems, *Physical Review Letters*, 64, pp. 2354–2357.

[23] Nielsen, M.A. and Chuang I.L. (2000), *Quantum Computation and Quantum Information*, Cambridge: Cambridge University Press.

[24] Pitowsky, I. (1990), The Physical Church Thesis and Physical Computational Complexity, *Iyyun* 39, pp. 81–99.

[25] Pitowsky, I. (2002), Quantum Speed–up of Computations, *Philosophy of Science* 69, pp. S168–S177.

[26] Pitowsky, I. and Shagrir, O. (2003), Physical Hypercomputation and the Church-Turing Thesis, *Minds and Machines* 13, pp. 87–101.

[27] Pour-el, M. and Richards, I. (1981), The Wave Equation with Computable Initial Data such that Its Unique Solution is not Computable, *Advances in Mathematics*, 39, pp. 215–239.

[28] Reichardt, B.W. (2004), The Quantum Adiabatic Optimization Algorithm and Local Minima, *Proceedings of the 36th Symposium on Theory of Computing (STOC)*, pp. 502–510.

[29] Shor, P. (1994), Algorithms for Quantum Computation: Discrete Logarithms and Factoring, *Proceedings of 35th Annual Symposium on Foundations of Computer Science* (Shafi Goldwasser, ed.), IEEE Computer Society Press, 124–134. .

[30] Smith, W. (2005), Three Counterexamples Refuting Kieu's Plan for "Quantum Adiabatic Hypercomputation"; and Some Uncomputable Quantum Mechanical Tasks, Forthcoming.

[31] Van Dam, W., Mosca, M. and Vazirani, U. (2002), How Powerful Is Adiabatic Quantum Computation?, Archive Preprint quant-ph/0206003.

[32] Unruh, W.G. (1995), Maintaining Coherence in Quantum Computers, *Physical Review A* 51, pp. 992–997.