

Re: Quantum computer using using artificial atoms.

Source: <http://www.derkeiler.com/Newsgroups/sci.crypt/2005-02/5053.html>

From: Décio Luiz Gazzoni Filho (*decio_at_decpp.removethis.net*)

Date: 02/28/05

Date: Mon, 28 Feb 2005 11:36:17 -0300

jstevh@msn.com wrote:

> *Seth wrote:*

>> *In article <1109528598.941998.92320@o13g2000cwo.googlegroups.com>,*

>> *<jstevh@msn.com> wrote:*

>>

>>> *Beth wrote:*

>>

>>>> *Please, please, please read a book or take a course in*

> *computational*

>>>> *complexity theory. Quantum computable and (deterministic-)*

> *Turing*

>>>> *computable are not the same thing and can not be made to be the*

> *same*

>>>> *thing – not with any known theory of computation.*

>>

>>> *Why? Because you say so?*

>>

>>> *I presume from your statement that I should read a book on*

>>> *computational complexity theory that you have actually read one, so*

> *my*

>>> *next question should be easy for you.*

>>

>>> *Can you cite any known text that supports your claim?*

>>

>> *I gave you one reference which has a brief discussion of complexity*

>> *theory already – Quantum Computation and Quantum Information. You*

> *can*

>> *try the article "A Survey of Quantum Complexity Theory" by Umesh V.*

>> *Vazirani found in the book "Quantum Computation: A Grand Mathematical*

>> *Challenge for the Twenty-First Century and the Millenium", Samuel J.*

>> *Lomonaco, Jr., Editor, American Mathematical Society, ISBN*

>> *0-8218-2084-2. This article describes a quantum Turing machine,*

> *which*

>> *is different from a Turing machine – as you can see from the*

>> *mathematical definitions provided.*

sci.crypt: Re: Quantum computer using using artificial atoms.

>>
>
> <deleted>
>
> Sounds interesting and I read the rest of your post but wanted to focus
> on the end where I think you are getting into the area where I want to
> make an important point.
>
>>
>> As I've said before, the issue is efficiency. You can simulate a
>> quantum computer on a conventional computer, but you can't
> necessarily
>> get a quantum polynomial time algorithm to run in polynomial time on
> a
>> conventional computer. Of course, you could prove me wrong by
> proving
>> $P=NP$.
>
> Factoring has never been proven to be a hard problem, but for anecdotal
> reasons is believed to be a hard problem, while my surrogate factoring
> algorithms are in P .

No. If a single j choice factored even a fraction of *all* integers (not just the small) ones, then it might be argued that it is probabilistic polynomial time, with a high chance of failure. As it stands, it's still exponential, and much worse than trial division — I believe it's $O(M)$ instead of $O(\sqrt{M})$ which is the running time of trial division.

> Now they don't work all the time, which is, of course, the major point
> that people seize upon, but they do often work.

Correction: they *sometimes* work with *small* numbers. I'm still waiting for a factorization of an RSA challenge.

> Like I've said, I can demonstrate a solution with bigger numbers than
> 15, which is the best that mechanical implementations have managed to
> do with quantum.

There's nothing surprising about this. Every important factoring algorithm (and there are many of them) can do much better than yours. What would be surprising is if your algorithm did worse than Shor's, which is at its infancy in terms of implementation.

> So, I have this method, which puts factoring in P ,

No it doesn't.

> which already factors well enough to show that there could be something to
> the concept,

No it doesn't, if you compare it to any other established algorithm.

Re: Quantum computer using using artificial atoms.

sci.crypt: Re: Quantum computer using using artificial atoms.

- > *and I have a different view of Turing machines and their generality than*
- > *others who are trying to push the idea that quantum Turing is different*
- > *from Turing.*

Of course you have a view of the world that goes against common sense and established knowledge; that's practically the definition of a crank.

- > *Time will tell,*

It has already spoken.

Décio