Dialogues on Quantum Computing

Cristian S. Calude
University of Auckland

What sort of machines do useful computation in a universe described by classical mechanics? The answer was provided in 1936 by the British mathematician Alan Turing, and it's known today as the *Turing machine*. But even in 1936 classical mechanics was known to be false, and so one could have asked the question: What sort of machines do useful computation in a universe described by quantum mechanics?

In a trivial sense, everything is a quantum computer. A pebble is a quantum computer for calculating the constant-position function; current computers exploit quantum effects (like electrons tunneling through barriers) to control computation and to be able to run fast. But quantum computing is much more than that.

Q: What has computing to do with physics?

A: Information, essential for any form of computing, is not a pure abstract entity. In fact, measuring, communicating and computing are *all* about exchanging information. Information is inevitably tied to a physical embodiment or representation; it can be engraved on stone tablets, represented by holes punched in a card, or by a present/absent charge or by a spin up or down.

R. Landauer: "The computer has made us aware that information is a physical entity".

D. Deutsch: "The reason why we find it possible to construct, say, electronic calculators, and indeed why we can perform mental arithmetic . . . *is that the laws of physics "happen" to permit the existence of physical models for the operations of arithmetic* such as addition, subtraction and multiplication.

Q: What is quantum computing?

A: Quantum computing is the quest to understand what sort of machines do useful computation in a universe described by quantum mechanics. Today the subject is mostly theoretical, but tentatively, slowly and hesitantly groping towards some practical applications.

Q: What is quantum mechanics?

A: Quantum mechanics tries to describe the behaviour of very small objects, the size of atoms or smaller, in contrast with relativity theory which describes the laws of larger everyday objects. Interestingly, particles do not behave in the same way as larger everyday objects, such as billiard balls. If we strike a billiard ball in a very precise way and we know its exact initial position, then we can predict with (theoretical) certainty where it will go. The same is not true for particles.

Q: If classical mechanics is wrong, why do we still use it?

A: Classical mechanics is flawed *only* when dealing with the very small (atomic size) or the very fast (near the speed of light). For everyday things, classical physics does an excellent job.

Q: What are the main features of quantum mechanics?

A: Here are five:

- *Quantisation*: observable quantities do not vary continuously, but come in discrete chunks called quanta.
- *Randomness*: physical reality is irreducibly random.
- *Interference*: the outcome of a quantum process depends on all the possible histories of that process.
- *Superposition*: the ability of carrying out computations with "blends" of states, superpositions.
- *Entanglement*: two spatially separated and non-interacting quantum systems, that have interacted in the past could have some locally inaccessible information in common – information which cannot be accessed in any experiment performed on either of them alone.

Q: Are these features useful for quantum computing?

A: Quantisation makes quantum computing possible at all. Randomness, superposition and interference make quantum computers more powerful than classical ones. Entanglement is useful in quantum cryptography.

Q: Computers are constantly becoming smaller, faster, cheaper and more potent. Why should we be concerned with quantum computing?

A: The information handling capacity of PCs has grown at a rate ten million times faster than the information handling capacity of our nervous systems during the 4 billion years since life began on Earth. Yet,

- this exponential race will not guarantee solutions to the many intractable/undecidable problems,
- the conventional technology will hit the wall in less than 10 to 15 years because information is carried by discrete quanta.

Q: Can you give a simple example of a problem and a quantum-like solution?

A: A famous example is the Merchant's Problem: "A merchant learns than one of his five stacks of $\Gamma = 1$ gram coins contains only false coins, $\gamma = 0.001$ grams heavier than normal ones. Can he find the odd stack by a single "weighing"?

The well-known solution of this problem is the following: we take one coin from the first stack, two coins from the second stack, ..., five coins from the last stack. Then by weighing the combination of coins described above we obtain the number $Q = 15 + \gamma \times n$ grams, which tells us that the n-th stack contains false coins.

Q: In contrast with classical computers which use bits, quantum computers process quantum bits. What's the difference?

A: We have four concepts:

- The mathematical bit (0 or 1).
- A classical system representing a bit, called Cbit.
- The mathematical quantum bit.
- A quantum system (event) in which we have two possible mutually exclusive outcomes realizing a quantum bit, called Qbit.

Examples of Qbits are: an atom, a nuclear spin or a polarised photon. For example, the state of a spin-$\frac{1}{2}$ particle, when measured, is always found to be in one of two possible states, represented as

$$|+\frac{1}{2}\rangle \text{ (spin-up) or } |-\frac{1}{2}\rangle \text{ (spin-down).}$$

All knowledge of the quantum system is based upon acts of observation. The information derived from an elementary act of observation is no more than a single bit, but *there is more to it than that.*

Before measurement, the system can be in any intermediate quantum state, that is in a superposition of 0 and 1, in a (sort of) mixture of 0 and 1 containing both classical (contradicting) states at once; after observation, we get either 0 or 1 with some probability. So, *an observation is simultaneously like a coin-toss and not like a coin-toss.*

Q: Are Qbits responsible for the famous "exponential explosion"?

A: Yes. Any classical register composed of three Cbits can store in a given moment of time only one out of eight different numbers because the register can be in only *one* out of eight possible configurations:

$$000, 001, 010, 011, 100, 101, 111.$$

A quantum register composed of three Qbits can store in a given moment of time *all* eight numbers in a quantum superposition. If we increase the number of Qbits to the register, then we increase its storage capacity exponentially: three Qbits can store eight different numbers at once, four Qbits can store sixteen different numbers at once, in general n Qbits can store 2^n numbers at once.

Q: What can you do with superpositions?

A: We can perform operations on them. During such an operation each number in the superposition is affected and as the result we obtain a massive parallel computation albeit in just one piece of quantum hardware. As in the solution of the Merchant's Problem, we can act at once on all stacks of coins. A quantum computer offers an enormous gain in time and memory.

Q: Where is the catch?

A: Qbits suffer from a major limitation which doesn't affect Cbits: given a superposition of Qbits in some state, there is nothing one can do to the Qbits to be able to extract what that state is in.

Q: Is this the only limitation?

A: No. There are also limited possibilities to extract the information contained in a Qbit. Learning the value of a combination of Cbits is so easy (you print it out) that it is not even explicitly regarded as a part of the computation. More importantly, Cbits are not altered by "reading" them. Not anymore with Qbits: we can extract the information from a Qbit *only* by measurement, a process which:

(a) is probabilistic (recall the intrinsic randomness of quantum mechanics), and

(b) affects the state of the Qbit; simple operations, like copying a Cbit into another Cbit, are not available in quantum computing.

Q: So, what are Qbits good for?

A: The art is to produce a superposition in which the useful information has a high probability of being indicated by measurement and the unimportant information can be expected to appear with probability close to zero. To make the result safe, one has to be able to easily *confirm* the result of the computation ...

Q: Can you give an example?

A: Peter Shor has shown in 1994 that quantum factoring integers is dramatically faster than any *known* classical algorithm. The obvious method of factoring a number N represented by n bits requires about $2^{n/2}$ trials.

A much smarter algorithm (based on sophisticated mathematical results) does the job in approximately $2^{c\sqrt[3]{n}}$ steps, where c is a constant; still, factoring a number of a million of bits would require a time larger than the age of the Universe.

Shor has observed that the factoring problem can be rephrased in terms of a search for how often some "period" of a finite sequence is repeating itself within the sequence. For example, the sequence

123412341234

has 1234 as period which repeats itself three times. Periods may be seen as waves, undulating streams.

The quantum algorithm is polynomial-time in the number of bits necessary to represent the number to be factored. Confirming the result is easy.

Q: What about Grover's quantum algorithm?

A: Start with an example. Searching a telephone directory containing n names in alphabetic ordering requires about $\log_2 n$ steps. Searching the name in the telephone directory, when the telephone number is known, is much more difficult because the list is unsorted with respect to telephone numbers. We need about $n/2$ steps on average and n steps in the worst case.

Looking up a name given a number is exponentially more difficult than looking up a number given a name.

Grover's quantum algorithm searches an unsorted list very fast; his procedure needs roughly $\pi/4\sqrt{n}$ quantum steps.

Q: What will quantum computers be good at?

A: These are the most important applications currently known:

- *Cryptography*: RSA code breaking, perfectly secure communication.
- *Searching*: fast searching (Grover's algorithm).
- *Simulating*: efficient simulation of quantum-mechanical systems.

Q: Can I learn quantum computing without understanding quantum mechanics?

A: Yes, you can. Recently, L. Fortnow has published a nice paper titled

> "One complexity theorist's view of quantum computing"

in which he shows that a large part of quantum computing can be understood without any knowledge of quantum mechanics.

Arguably, the amount of quantum mechanics required for the mainstream quantum computing is limited : this parallels the situation of classical computing, where computer scientists need not know much about transistors and the way they work.

Q: How soon a quantum computer might be built?

A: Lab experiments show that the basic principles of quantum computing are sound. To realistically compete with classical computing, quantum computing must be carried out on significantly larger scales ... It is unreasonable to make predictions; however, it is reasonable to expect that small milestones will continue to appear.

Q: Sure, quantum computing is not the only unconventional type of computing. What are other approaches?

A: Quantum computing is just one unconventional paradigm. DNA computing is another one; membrane computing is close but still different.

Q: Do you recommend any papers or books on quantum computing?

A: Here are some titles:

- J. Gruska. *Quantum Computing,* McGraw-Hill, London, 1999.
- M. Hirvensalo. *Quantum Computing*, Springer-Verlag, Berlin, 2001.
- A. Yu. Kitaev, A. H. Shen, M. N. Vylalyi. *Classical and Quantum Computation*, American Mathematical Society, Providence, Rhode-Island, 2002.
- M. A. Nielsen, I. L. Chuang. *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge, 2000.
- C. P. Williams, S. H. Clearwater. *Explorations in Quantum Computing,* Springer-Verlag, New York, 1997.
- C. P. Williams, S. H. Clearwater. *Ultimate Zero and One: Computing at the Quantum Frontier,* Springer-Verlag, Heidelberg, 2000.

Q: What about web references?

A:There are many. Here are some important sites:

- Quantum Computing at IBM Research Yorktown, `http://www.research.ibm. com/quantuminfo/`.
- Oxford Centre for Quantum Computation, `http://www.qubit.org/`.
- John Preskill Course (Physics of Computation), `http://www.theory.caltech.edu/people/preskill/ph229/`.
- The Home of the Home Pages Page (in Quantum Computing), `http://www.cs.berkeley.edu/~vandam/homes.html`.