QCB

QUANTUM COMPUTERS

Certainty from uncertainty

Charles H. Bennett

QUANTUM mechanics is best known for its uncertainty principle, but now physicists and computer scientists have found that quantum mechanics can also serve as a source of certainty. Quantum effects within a computer can be used, in principle, to solve quickly and with complete certainty some kinds of problem that a classical computer could only solve very slowly, or else with a small chance of error[1-3].

The extra power of quantum computers arises from the ability of a quantum system, when it is not being observed, to be in many places at the same time. A single photon making its way through a snowball, for example, simultaneously follows all possible optical paths through the snowball, and emerges in a way that is determined by the constructive and destructive interference among all the paths.

Interference

A single quantum computer could similarly, in principle, follow many distinct computation paths all at the same time and produce a final output depending on the interference of all of them. Although a number of obstacles[4] appear to stand in the way of achieving controllable interference among computation paths in a practical quantum computer, there has been much recent progress in understanding how, in principle, the possibility of such interference makes the theory of quantum computation differ from its somewhat better understood classical counterpart.

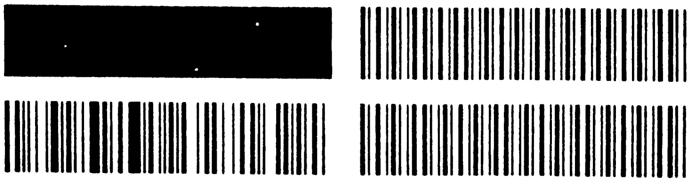
An early but probably vain hope was that quantum parallelism might provide a fast way of solving problems such as factoring or the travelling-salesman problem, which appear to be hard in the same way as finding a needle in a haystack is hard, that is because they involve a search for a successful solution among exponentially many candidates. A computer able to test all the candidates in parallel, and signal unambiguously if it found one that worked, would solve these problems exponentially faster than known methods

It is easy to program a quantum computer to branch out into exponentially many computation paths; the difficult part is getting the paths to interfere in a useful way at the end, so that the answer comes out with a non-negligible probability. This difficulty is illustrated by the factoring problem above. Suppose the quantum computer is programmed to factor a 100-digit number by trying in parallel to divide it by all numbers of fifty digits or fewer. If any of these approximately 10^{50} computations yields a zero remainder, it will in a sense have solved the problem. But if there is only one successful path, the interference pattern among all the paths, which determines the behaviour of the computer as a whole, will scarcely be affected. Quantum computers cannot amplify an answer found on a single computation to a detectable level because interference is an additive process, to which each path contributes only as much weight as it started out with.

Therefore, a quantum computer's chance of quickly solving a problem of this type is no greater than that of a classical stochastic computer, which can choose a single computation path randomly. For interference to achieve anything useful, it must be applied to problems whose solution depends on many computation paths.

The recent spate of progress in quantum computation stems from a paper by David Deutsch of Oxford University and Richard Jozsa, now at Université de Montréal[1]. The authors use a quantum computer to determine global properties of a boolean function f of an n-bit argument, by having the computer branch into 2^n computation paths, one for each value of the argument x. They show that the computer can then cause the paths to interfere in such a way as to determine, quickly and with no chance of error, whether the function f is 'unanimous' ($f(x)=0$ or $f(x)=1$ for all x) or 'balanced' ($f(x)=0$ for exactly half the x values and $f(x)=1$ for the other half).

A classical computer, by contrast, limited to testing arguments one at a time, might find unanimity among all the $f(x)$ it had tested so far, but would still not be able absolutely to rule out the 'balanced' possibility until it had tested more than half the paths, a job requiring an exponentially increasing amount of time. If we are unwilling to tolerate any chance of error, the problem of distinguishing balanced from unanimous functions can be solved quickly on a quantum computer but not on a classical one. On the other hand, if even a tiny chance of error is allowed, a classical stochastic computer could also solve the problem quickly, by testing a few dozen x values at random, then declaring the function to be unanimous if they all agreed.



A quantum computer can quickly and with certainty distinguish a unanimous function (top, left) from a balanced function (for example, bottom, left). It can also distinguish between a large number of subtly different balanced functions, such as those on the right.

In the figure, the top left bar shows a unanimous function and the bar below it a balanced function for a toy example with $n=7$ and 128 x values. Although the global difference between these two bars is obvious to the eye, a classical computer, limited to testing points one at a time, would need to test over half the x values to be sure the upper bar was unanimous. A single quantum computer, by contrast, could quickly and surely detect the global difference by pursuing 128 computation paths in parallel and causing them to interfere.

Ethan Bernstein and Umesh Vazirani at the University of California at Berkeley have strengthened Deutsch and Jozsa's result[2] by showing that the same quantum computer that distinguishes unanimous from balanced functions can also (and still with no chance of error) distinguish among 2^n-1 kinds of balanced functions. The bars on the right of the figure show two of these functions in the toy example: a quantum computer could distinguish these functions from each other and from 125 other balanced functions, most of which would look confusingly similar to the naked eye. The class of balanced functions distinguishable by quantum computation (the so-called Walsh functions) are of independent mathematical interest, being boolean analogues of the basis functions used in Fourier analysis.

Complexity

What, then, is the relation between the powers of quantum and classical computation? As is well known, classical computational complexity theory suffers from a humiliating inability to answer some of its most basic questions, such as whether the two complexity classes — P, for deterministic polynomial time, and NP, for nondeterministic polynomial time — are equal. In consequence, the factoring and travelling-salesman problems alluded to earlier are merely thought, not known, to be hard. This frustrating situation arises because, in a real problem, alternative computation paths are not independent — for example, in factoring, if a number is divisible by 6 then it will certainly be divisible by 3. Thus one cannot be sure that there is not a much faster way of solving the problem than by blind search: either of these problems might turn out to be more like finding a needle in a department store than finding one in a haystack.

Unable to answer their biggest questions, complexity theorists have retreated to proving two lesser kinds of theorems: oracle results and completeness results. Oracle results concern the power of computers that can ask questions of a black box without being allowed to look inside it. The f functions considered above are an example of an oracle. Because Deutsch and Jozsa's proof does not allow the computer to 'open the box' and examine the instructions for calculating f, it is not an absolute proof that quantum computers are more powerful than classical ones, merely a proof that they are more powerful if certain kinds of f functions exist — functions that are balanced or unanimous, but whose instructions cannot easily be analysed to determine which. Gilles Brassard of Montréal and Andre Berthiaume[3], now at Oxford, as well as Bernstein and Vazirani[2] have proved a number of oracle results based on Deutsch and Jozsa's construction, which characterize in considerable detail the power of oracle-assisted quantum computers relative to their oracle-assisted classical analogues. The bravest of these results is a family of oracle problems which are easy for quantum computers but not for classical ones, even when the latter are allowed to make errors.

The other approach, the completeness approach, eschews oracles and is based instead on finding problems that, although not known to be hard, can be proved to be at least as hard as any other problem in a given class. Thus, in classical complexity theory, the travelling-salesman problem is called NP-complete because all other problems in the class NP can be reduced to it. It may similarly be possible to characterize the power of quantum computation by finding problems that are provably at least as hard as any problem in a given quantum complexity class. □

Charles H. Bennett is at the IBM T J Watson Research Centre, Yorktown Heights, New York 10598, USA.

[1] Deutsch, D & Jozsa, R *Proc R Soc* A439, 553–558 (1992)
[2] Bernstein, E. & Vazirani, U at *A. C M Symp. Theory of Computing*, San Diego, 1993 (in the press).
[3] Berthiaume, A & Brassard, G. at *7th IEEE Conf. Structure in Complexity Theory*, 1992 (in the press).
[4] Landauer, R. *Phys. Today* 23–29 (May 1991).