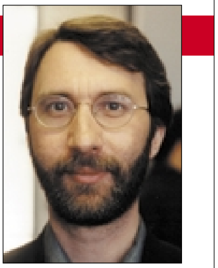
TRENDS & CONTROVERSIES

A quantum leap for AI



By Haym Hirsh
Rutgers University
Hirsh@cs.rutgers.edu

November 1994 saw the near-simultaneous publication of two papers that threw the notion of computing on its head. On November 11, 1994, a paper by Leonard Adleman appeared in *Science* demonstrating that a vial of DNA fragments can serve as a computer for solving instances of the Hamiltonian path problem. Less than two weeks later, Peter Shor presented a paper in Santa Fe, New Mexico, at the 35th Annual Symposium on Foundations of Computer Science, demonstrating how a quantum computer could be used to factor large numbers in a tractable fashion. Both these publications showed how nontraditional models of computation had the potential to effectively solve problems previously believed to be intractable under traditional models of computation. However, the latter work, using a quantum model of computation proposed by Richard Feynmann and others in the early 1980s, resonated well with AI researchers who had been coming to terms with Roger Penrose's 1989 book *The Emperor's New Mind*. In this book (and its sequel, *Shadows of the Mind: A Search for the Missing Science of Consciousness*, which appeared in paperback form only a month before these papers), Penrose challenges the possibility of achieving AI via traditional "Turing-equivalent" computation devices, conjecturing that the roots of intelligence can be traced to macroscopic quantum effects in the brain. These two quantum strands form the motivation for a small community of researchers exploring the topic of this issue's "Trends and Controversies" feature—the potential uses of quantum computing for AI.

Subhash Kak's leadoff essay provides an excellent overview of the foundations of this area, explaining, for example, how a quantum computer makes it possible to manipulate an exponential number of states in a search problem in a single clock step. He also discusses the philosophical motivations that have led people to explore the use of quantum computing in AI.

To the many with the widely held belief that tractable search is a core question in achieving AI, Tad Hogg's essay explains how quantum computing can potentially form the basis for tractable search on what have previously been considered intractable problems. One difficulty is that the "common" quantum computing approach of amplitude amplification can yield at most a square-root improvement in an algorithm's runtime. Hogg proposes mapping techniques commonly used in AI to the quantum computing world, specifically by using heuristics that embody knowledge about the structure of a problem (such as the number of conjuncts satisfied by a truth assignment for a given satisfiability problem) within the search process.

Finally, Dan Ventura looks at the mutual benefits received by advances in the use of quantum computing for AI, for both AI researchers as well as those studying quantum computation. Attempting to apply quantum computing to AI problems might provide the right fodder for researchers in quantum computing. Similarly, the use of quantum computing might enable new advances previously thought impossible within the AI community. Ventura also discusses some of the difficulties that may lie ahead for those hoping to achieve AI through quantum computing. Particularly important—as anyone who has attempted interdisciplinary research knows—is the (understandable) gap in motivations, background, and vocabularies of those working in these two fields.

We are still quite far from having quantum computers sitting on our desktops running Unix or Windows. However, as advances in quantum computing continue to be made, it is nice to know that researchers such as the authors in this issue's "Trends and Controversies" might already be coming to an understanding of how to effectively use the results of these advances. Indeed, their work might itself become responsible for these important advances in quantum computing as well.

As a final note, this issue marks the passing of the cartoonist's pen from Kevin Knight, who has diligently served in this role for the last four years, to Sally Lee, who is responsible for the cartoon appearing in this installment. Although we are sorry to see Kevin's retirement from this post, we are happy to have such an able successor. Thank you both Kevin and Sally.

—***Haym Hirsh***

Quantum computing and AI

Subhash Kak, Louisiana State University

Every few years, we hear of a new technology that will revolutionize AI. After careful reflection, we find that the advance is within the framework of the Turing machine model and equivalent, in many cases, to existing statistical techniques. But this time, in quantum computing, we seem to be on the threshold of a real revolution—a "quantum" leap—because it is a true frontier beyond classical computing. But will these possibilities be realized any time soon?

Classical computers work on classical logic and can be viewed as an embodiment of classical physics. Quantum computers, on the other hand, are based on the superpositional logic of quantum mechanics, which is an entirely different paradigm. Conventional explanation sees consciousness arising as an emergent property of the classical computations taking place in the circuits of the brain, but this does not address the question of how thoughts and feelings arise. If brains perform quantum processing, this might be the secret behind consciousness. Furthermore, it might explain several puzzling features of animal and human intelligence and provide a new direction to develop AI machines. In this brief survey, I present the rationale for the convergence between quantum computing and AI and discuss prospects for realizing the technology.

The weirdness of quantum mechanics

Let me begin with the quantum framework. It is a theory that provides a means of obtaining information about a system in the microworld associated with various attributes (component states). A quantum state is a linear superposition of its component states. Suppose the two component states are represented by |0> and |1>, which could be the two spin states of an elementary particle (up or down), or polarization states of a photon (horizontal or vertical), and so on. Then, the general form of the superposition state, $|S>$, will be

$$|S> = a\,|0> + b\,|1>.$$

The weights, a and b, are called *probability amplitudes* and are, in general, complex numbers, subject to the condition that $|a|^2 + |b|^2 = 1$. The mod squares of the probability amplitudes, $|a|^2$ and $|b|^2$, are the probabilities of obtaining either of the two component states upon observation.

The fact that the amplitudes are complex numbers implies that a quantum system cannot be effectively simulated by the Monte Carlo method using random numbers. You cannot run a physical process if its probability amplitude is negative or complex!

Apart from this, the counterintuitive nature of quantum mechanics arises from the fact that, upon interaction with a measurement apparatus, the linear superposition quantum state reduces to one of its component states with the appropriate probability. This aspect of quantum mechanics renders the framework nonlinear—and irreversible if the time variable is changed in sign.

For decades, philosophers of science have agonized over the many bizarre implications of quantum mechanics, such as that an organism can be both dead and alive before it is observed (Schrödinger's cat paradox), the present can influence the past (Wheeler's delayed-choice scenario), effects can propagate instantaneously in apparent violation of the ceiling of the speed of light (EPR paradox), and so on.[1]

Quantum mechanics' strange effects arise because it is so contrary to rules of classical logic. Nevertheless, we must live by quantum mechanics because it is the most successful theory available and because it lets us understand the microworld—including chemistry and biology—and devise electronics and computers.

The power of quantum computing

The dynamics of an isolated quantum system are governed by the Schrödinger equation, which can be cast in a form where the system's future states are obtained by multiplication by a unitary matrix. The algorithm designer must first find the unitary matrix for the given computing problem and then map the matrix into a sequential product of smaller matrix operations that can be implemented relatively easily. The fact that a quantum computation is nothing more than matrix multiplication of a certain kind should give comfort to computer scientists—in operational terms, it is not weird at all!

A quantum computer exploits the inherent parallelism that is provided by the superposition of the quantum state. A quantum register with n binary cells is able to store 2^n sequences simultaneously, in contrast to a classical register, which can store only 1 of the 2^n sequences at a time. By its ability to simultaneously process very many problems, the quantum computer makes it possible to devise new kinds of algorithms that provide substantial speedup over classical methods—that speedup, in principle, could be exponential.[2]

A basic issue in quantum computing is to separate the good solution from the many other data sequences that are simultaneously present on the quantum register, and this must be done without looking, because interaction with the contents of the register will cause the superposition state to collapse to one of its components. We achieve this separation by strengthening the amplitude of the desired (or marked) state by changing the difference in the phase angles of the marked and unmarked states.

Small implementations, at the level of proof of concept, of quantum computers have been made based on different technologies, such as NMR, trapped ions, quantum dots, and cavity quantum electrodynamics. Current problems with quantum computer technology include initialization, decoherence, and error correction.

The problem of initialization arises from a fundamental uncertainty in the phase of the state. This uncertainty can render the techniques for strengthening of the desired state useless. Decoherence is the inability to completely shield the quantum system from unpredictable interaction with the environment, causing the state function to lose its superposition; decoherence times range from a fraction of a second to a few hundred seconds. Techniques for error correction of quantum bits have been proposed, but these work under very artificial and unrealistic assumptions.[3]

Schrödinger's computer. —Sally O. Lee

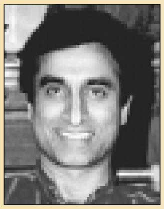
Quantum computing at the basis of biological information processing

The case that quantum computing is at the basis of biological information processing and, consequently, the explanation for the power of animal intelligence, relies on the following elements:

- *Philosophical*. The argument is that, at the deepest level of description nature, is quantum-mechanical. The world of mathematics, as a product of the human mind, sits on top of the sequence physical -> chemical -> mental -> mathematical. If our ideas, with their concomitant mathematics, can describe the quantum-mechanical physical reality, it should only be possible because the brain's information processing has a quantum-mechanical basis. Another version of this argument is that quantum mechanics as a universal theory should also apply to information and organization, so the brain's information processing cannot be understood but in quantum-mechanical terms. Starting on this problem in the mid-1970s, I have, over the years, developed a framework for quantum neural computing.[4,5]
- *Neurophysiological*. The interior of living cells is organized around the cytoskeleton, which is a web of protein

polymers. The major components of the cytoskeleton are the microtubules, which are hollow tubes 25 nm in diameter, consisting of 13 columns of tubulin dimers arranged in a skewed hexagonal lattice. Some researchers have argued that the microtubules support coherent, macroscopic quantum states. They see brain processing as a hybrid quantum and classical computation.[6]
- *Behavioral science*. Human and nonhuman animal intelligence appears to have features that lie beyond the capacity of the most powerful machines. Conceptualization is not unique to humans and ability to use language is not a precondition to cognition or abstract processing. This processing appears to run according to a noncomputable program.

Tad Hogg is a member of the Internet Ecologies group at the Xerox Palo Alto Research Center. His research interests include mechanisms to enhance privacy for Web-based communities, market-like algorithms for multiagent systems, distributed controls for collections of micromachines (smart matter), and phase-transition behaviors in combinatorial search and their use for designing quantum computer algorithms. He received a BS from Caltech and a PhD from Stanford, both in physics. Contact him at Xerox PARC, 3333 Coyote Hill Road, Palo Alto, CA 94304; hogg@parc.xerox.com; http://www.parc.xerox.com/hogg.



Subhash Kak is a professor in the Department of Electrical and Computer Engineering at Louisiana State University in Baton Rouge. His research interests include quantum computing, neural networks, information technology, and the history of science. He obtained his PhD from the Indian Institute of Technology, Delhi. He was one of the first to work on bringing together quantum mechanics and information. Most recently, he has worked on the theory and applications of instantaneously trained artificial neural networks. Contact him at the Dept. of Electrical and Computer Eng., LSU, Baton Rouge, LA 70803-5901; kak@ee.lsu.edu; http://www.ee.lsu.edu/kak.

Dan Ventura is a research scientist with the *fonix* Corporation, working on the development of state-of-the-art technology for large-vocabulary continuous speech recognition. His main areas of interest include neural networks, machine learning, quantum computation, and their eclectic combination. He received his PhD in computer science from Brigham Young University. Contact him at the *fonix* Corp., 180 W. Electron Rd., Drapier, UT 84040; dventura@fonix.com or dan@axon.cs.byu.edu; http://axon.cs.byu.edu/Dan.

Another motivation to study quantum phenomena in biological information processing is the phenomenon of consciousness. Some physicists have argued that consciousness is a part of the quantum framework because it is this that causes the reduction of the state function. More directly, if there are no grounds for assuming that consciousness arises just from the complexity of the neural mechanisms, and if classical processes cannot explain it, then it might very well emerge from a nonclassical—quantum—process. It is plausible that the notion of "self," which provides a unity to experience, is a result of quantum processes.[7]

Self-organization

We can view animal intelligent behavior as a continuing self-organization of the animal to the changing environment. Each animal is sufficiently intelligent because it survives in its ecological environment. Likewise, brains continually go through self-organization, which is what provides the animal the ability to respond to novel situations. Because quantum mechanics is a framework where the environment naturally comes into the picture, it defines an appropriate basis for the consideration of self-organization. So, we expect that quantum computing would help us find a basis for the development of programs and machines that have self-organizational ability.

What if quantum computers were harnessed for AI?

Only toy versions of quantum computers have been built to date. But it is reasonable to assume that if they existed they would, by solving many currently intractable problems efficiently, bring about a revolution in AI.

Take, for example, the protein-folding problem, which is important in bioinformatics. Proteins are sequences of a large number of amino acids. Once a sequence is established, the protein folds up rapidly into a highly specific 3D structure that determines its function in the organism. Likewise, a drug's 3D structure defines its effectiveness. If we could study 3D structures on a computer, it would save a great deal of the expense of test-tube experiments.

It has been estimated that a fast computer applying plausible rules for protein folding would need 10^{127} years to find the final folded form for even a very short sequence of just 100 amino acids. Such a mathematical formulation of the protein-folding problem shows that it is NP-complete.[8] Yet Nature solves this problem in a few seconds. Assuming that the basis of this solution is quantum-mechanical, a quantum computer should be able to solve such a problem relatively easily.

Extremely fast quantum algorithms have already been proposed for some optimization problems. Their potential impact on AI is enormous. But quantum computing technology is yet to overcome fundamental technological hurdles. We cannot yet say whether it will be a couple of years, or decades, before the dream of building quantum computers is fulfilled. It is only then that we will be able to answer questions such as whether quantum computers ultimately lead to conscious machines.

References

1. M.P. Silverman, *More Than One Mystery*, Springer-Verlag, New York, 1995.
2. S. Kak, "Quantum Information in a Distributed Apparatus," *Foundations of Physics*, Vol. 28, 1998, pp. 1005–1012.
3. S. Kak, "The Initialization Problem in Quantum Computing," *Foundations of Physics*, Vol. 29, 1999, pp. 267–279.
4. S. Kak, "Quantum Neural Computing," *Advances in Imaging and Electron Physics*, Vol. 94, 1995, pp. 259–313.
5. S. Kak, "The Three Languages of the Brain: Quantum, Reorganizational, and Associative," *Learning as Self-Organization*, K. Pribram and J. King, eds., Lawrence Erlbaum Associates, Mahwah, N.J., 1996, pp. 185–219.
6. M. Jibu et al., "Quantum Optical Coherence in Cytoskeletal Microtubules: Implications for Brain Function," *BioSystems*, Vol. 32, 1994, pp. 195–209.
7. R. Penrose, *Shadows of the Mind*, Oxford Univ. Press, New York, 1994.
8. A.S. Fraenkel, "Protein Folding, Spin Glass and Computational Complexity," *Proc. Third Ann. DIMACS Workshop on DNA Based Computers*, Univ. of Pennsylvania, Philadelphia, 1997.

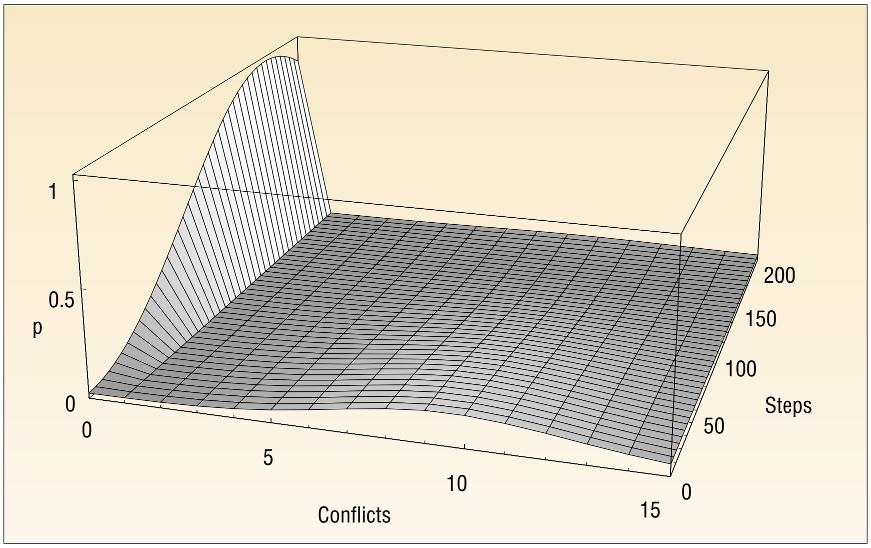


Figure 1. An example of amplitude amplification.

Quantum search heuristics

Tad Hogg, Xerox Palo Alto Research Center

In 1994, Peter Shor's polynomial-time factoring algorithm[1] showed that quantum computers[2] could rapidly solve an important problem thought to require exponential time on our current, "classical" machines. This algorithm inspired considerable interest in quantum computing, leading to additional algorithms and, so far, implementations with a few bits.[3]

Particularly relevant for artificial intelligence is how well quantum computers perform combinatorial searches, such as arise in scheduling, planning, and theorem proving. Can quantum computers solve all NP search problems in polynomial time? As with classical computers, the answer to this question isn't known, although it appears to be no.

Lacking a definitive answer, a practical fallback is to ask how well quantum computers perform for *typical* searches encountered in practice. Heuristics often solve them much faster than worst-case analyses suggest, motivating the study of heuristic quantum algorithms.

Searching with quantum computers

Instead of a single value for each bit at a time, quantum computers operate simultaneously on both values. With this quantum parallelism, a machine with just n bits manipulates 2^n states in parallel, just as if it were 2^n copies of an n-bit classical machine, each running the same program on different data.

However, like the proverbial watched pot that never boils, quantum parallelism operates only while the computer isn't observed. Observation gives each bit a specific value, 0 or 1. The probability an observation produces a particular state—that is, specific values for the n bits—is determined by a complex number, called an amplitude, associated with the state.

At first sight, quantum parallelism seems ideally suited for NP search problems, which have rapid tests of whether a given state is a solution. Quantum parallelism can test *all* states with about as many computational operations as a classical machine uses to test just one. Unfortunately, performing the test doesn't change the amplitudes: an observation made afterwards has no better chance of producing a solution than before. Instead, algorithms must not only test the states but also change amplitudes based on the results.

Amplitudes change through interference,[4] conceptually the same process as interfering light or sound waves. For quantum search, interference arises because the final amplitude for a given state is the sum of contributions from all search paths leading to that state. With the ingredients of parallelism, observation, and interference, the algorithm designer must arrange changes in amplitudes along each path so those leading to solutions combine mostly in-phase, giving large amplitude, while others combine with different phases, leading to significant cancellation.

Quantum search algorithms typically have the following form:

1. Initialize the computer—for example, give all states the same amplitude.
2. Without observing the computer, repeat for a specified number of steps: (a) compute properties of all search states in parallel (for example, whether the state is a solution); (b) use these properties to change amplitudes through interference.
3. Observe the computer, producing a single final state.

If the final state isn't a solution, the entire algorithm is repeated. Performance is commonly measured by the number of steps, or the times the state properties are evaluated, including those due to any repetitions, before a solution is found. This measure corresponds to the number of states examined or nodes expanded in a search tree by classical methods. The actual time for each step depends on implementation details of future quantum computers, such as their clock speeds.

One goal of these algorithms is rapid search. However, heuristics are also useful if they simplify hardware implementations. Maintaining quantum parallelism over many steps is difficult: environmental disturbances eventually destroy the parallelism just as when the machine is observed. Novel error correction methods can help,[5] but a heuristic that significantly increases solution amplitudes with only a few steps of parallelism, even if at the expense of more repetitions of the algorithm, will be easier to implement than one using many more parallel steps but fewer repetitions.

Amplifying amplitude in solutions

Lov Grover introduced the amplitude amplification quantum search technique,[6] which was subsequently generalized.[7] It uses only one property of search states, whether they are solutions, to improve probabilistic classical methods, such as heuristic repair.[8] Typically, such methods perform a series of trials until a solution is found. Each trial starts from a randomly selected initial state and then changes the state until it either finds a solution or reaches a prespecified limit on the number of changes. In the latter case, another trial is performed from a new initial state. Suppose a single trial succeeds with probability P. Classically, the search requires $1/P$ trials, on average, to find a solution. Performing the trials on a quantum computer, amplitude amplification needs only about $1/\sqrt{P}$ trials. Thus, for heuristics whose search cost is due mainly to the many repetitions, quantum machines give a square-

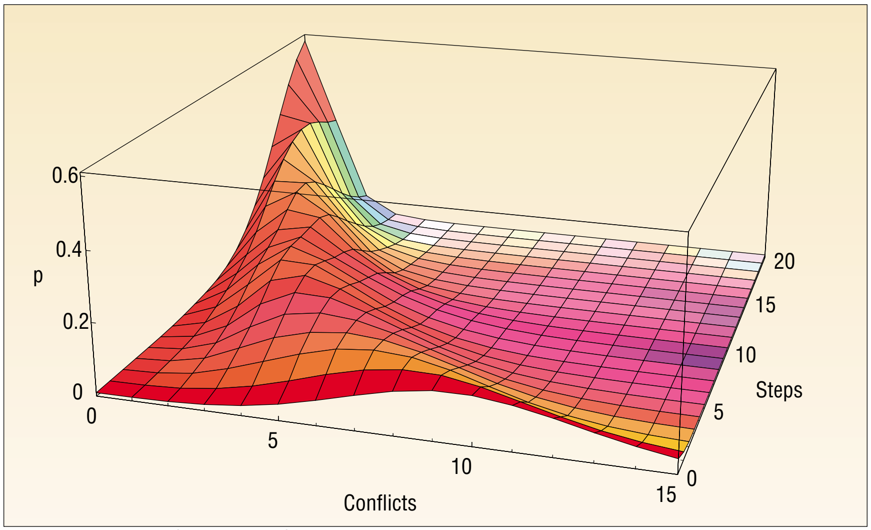


Figure 2. An example of amplitude shifting. Colors indicate the typical amplitude phases associated with each number of conflicts. Where these phases vary considerably due to incorrect choices, the colors are faded. Most of this variation occurs where amplitudes are very small, limiting their effect on the overall performance.

root improvement in performance—that is, if the classical search cost scales as e^{an}, where n is the size of the problem and a is a constant, the corresponding quantum method scales as $e^{an/2}$.

The simplest example is based on the random generate-and-test procedure where each trial is just a random guess, succeeding with probability P equal to the fraction of search states that are solutions. Figure 1 shows amplitude amplification applied to this procedure for a randomly generated 3-SAT problem with 20 variables and 80 clauses. It has 24 solutions, so $P = 24/2^{20}$. The figure shows the probability P that halting the computation at each step would give a state with the number of conflicts ranging from 0 (a solution) to 15. The values for step 0 just show the chance of getting each number of conflicts by random selection. The small chance of finding states with more than 15 conflicts isn't included in the figure.

In this example, amplitude amplification increases the likelihood of finding a solution, reaching probability $P = 1$ in 164 steps, compared to an average classical cost of $1/P = 43{,}691$ steps. Continuing the quantum algorithm beyond 164 steps reduces the likelihood of finding a solution. Thus determining when to halt the quantum computation, which can't rely on observation, is nontrivial. Instead, identifying the correct number of steps requires an estimate of the number of solutions or a few repetitions with different guesses.[9]

Amplitude amplification offers wide applicability, has a well-understood theory of its behavior, and allows quantum machines to piggyback on improvements in classical heuristics. It has already been implemented for a two-variable problem instance.[3] On the other hand, this technique does not directly apply to search methods that spend most of their time in preprocessing, such as for creating tables of inconsistent states or learning heuristic parameters, after which a solution is found rapidly. During the lengthy preprocessing, there is no chance of producing a solution and hence no opportunity for amplitude amplification. Even when it does apply, this square-root improvement is the best possible for algorithms based only on whether a state is a solution.[10] Moreover, by requiring parallelism maintained over exponentially many steps, amplitude amplification poses challenging requirements for the hardware. Further improvements, including any hope of finding polynomial-time algorithms for some types of problems, at least on average, require greater use of problem structure.

Shifting amplitude toward solutions

Potentially more powerful, but less general, quantum search methods change amplitudes using more information than just whether states are solutions. As with classical heuristics, such information consists of readily computed properties that, at least roughly, indicate how far a state is from a solution. Examples for constraint satisfaction include the number of conflicts in a state, how that number compares to those in its neighbors, and conflicts in partial assignments, as used in backtracking searches. Quantum parallelism readily evaluates such properties for all states. Using such information is effective for some cases,[11] although the search cost remains exponential for hard search problems near phase transitions identified by Peter Cheeseman, Bob Kanefsky, and William Taylor[12] and subsequently studied extensively.[13]

As an example of using more problem structure, Figure 2 shows the behavior of a heuristic I'm developing, for the same 3-SAT instance used in Figure 1. This heuristic uses exactly n parallel steps for satisfiability problems with n variables. It exploits the correlation between number of conflicts and distance to a solution for random 3-SAT problems. That is, states with relatively few conflicts tend to have more assigned values in common with a solution than states with many conflicts. This correlation is not perfect, of course, often leading classical searches to local minima or plateaus.[14] By simultaneously following all search paths, such states aren't difficulties for quantum search; instead, imperfect correlations lead to some incorrect amplitude changes. Furthermore, individual problem instances, such as the example used in the figures, differ somewhat from the average assumed by the heuristic, giving rise to additional errors. Thus contributions don't combine exactly in-phase for solutions. This means the heuristics, which initially increase the probability of finding a solution much more rapidly than amplitude amplification, are limited in how large the probability becomes. This limit, in turn, requires repeating the algorithm until a solution is found. A hybrid approach can reduce these repetitions: the quantum heuristic can be combined with amplitude amplification to gain a further square-root improvement, provided the hardware can maintain parallelism throughout the whole process.

The heuristic shown in Figure 2 also shifts the whole probability distribution toward states with fewer conflicts in each step, unlike amplitude amplification. Thus, even if a solution isn't found, the result is likely to be a state with only a few conflicts, making the heuristic useful for optimization problems.

Developing search heuristics

Quantum computers offer many new opportunities for using information available in combinatorial searches. For some problems, quantum analogs of classical methods will be useful. Other problems might allow quantum parallelism and interference to combine information diffusely scattered throughout a search space that can't be efficiently used by any classical method. This observation raises a converse problem as well: some applications, such as cryptography, rely on the ability to easily create hard instances of search problems. Such applications now have a new challenge: identify instances likely to remain hard even for quantum computers because they have only very weak correlations between easily computable properties of search states and their distances to solutions.

How much quantum heuristics can improve on the square-root speedup of amplitude amplification, especially for typical rather than worst-case problems, remains an open question. Addressing this question requires developing new heuristics and evaluating their performance. As with many classical heuristics, extensive use of problem structure precludes exact theoretical analysis. The alternative of empirical evaluation is currently limited to small problems for quantum algorithms since their simulation on classical machines requires an exponential increase in time and memory. A third approach, using regularities in classes of search problems[13] to estimate performance,[15] might address these difficulties.

We can expect continued exploration of quantum heuristics in the next few years. Though unlikely to work well in all cases, they may greatly reduce, if not eliminate, the exponential growth in search cost. AI researchers with expertise in the structure of combinatorial search and heuristics can play an important role in this work. Provided the substantial technical challenges of implementing quantum machines with many bits can be overcome, such heuristics will be key to realizing the benefit of quantum computers for AI applications. (A good source for new results in quantum computation is the Los Alamos quantum physics preprint library, available at *http://xxx.lanl.gov/archive/quant-ph*.)

References

1. P.W. Shor, "Algorithms for Quantum Computation: Discrete Logarithms and Factoring," *Proc. 35th Symp. Foundations of Computer Science*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1994, pp. 124–134.
2. D.P. DiVincenzo, "Quantum Computation," *Science*, Vol. 270, 1995, pp. 255–261.
3. I.L. Chuang, N. Gershenfeld, and M. Kubinec, "Experimental Implementation of Fast Quantum Searching," *Physical Rev. Letters*, Vol. 80, 1998, pp. 3408–3411.
4. R.P. Feynman, *QED: The Strange Theory of Light and Matter*, Princeton Univ. Press, Princeton, N.J., 1985.
5. E. Knill, R. Laflamme, and W.H. Zurek, "Resilient Quantum Computation," *Science*, Vol. 279, 1998, pp. 342–345.
6. L.K. Grover, "Quantum Mechanics Helps in Searching for a Needle in a Haystack," *Physical Rev. Letters*, Vol. 78, 1997, pp. 325–328.
7. G. Brassard, P. Hoyer, and A. Tapp, "Quantum Counting," *Proc. 25th Int'l Colloquium on Automata, Languages, and Programming (ICALP98)*, Springer-Verlag, New York, 1998, pp. 820–831.
8. S. Minton et al., "Minimizing Conflicts: A Heuristic Repair Method for Constraint Satisfaction and Scheduling Problems," *Artificial Intelligence*, Vol. 58, 1992, pp. 161–205.
9. M. Boyer et al., "Tight Bounds on Quantum Searching," *Proc. Workshop Physics and Computation (PhysComp96)*, New England Complex Systems Inst., Cambridge, Mass., 1996, pp. 36–43.
10. C.H. Bennett et al., "Strengths and Weaknesses of Quantum Computing," *SIAM J. Computing*, Vol. 26, 1997, pp. 1510–1523.
11. T. Hogg, "Solving Highly Constrained Search Problems with Quantum Computers," *J. Artificial Intelligence Research*, Vol. 10, 1999, pp. 39–66; http://www.jair.org/abstracts/hogg99a.html.
12. P. Cheeseman, R. Kanefsky, and W.M. Taylor, "Where the Really Hard Problems Are," *Proc. Int'l Joint Conf. AI,* Morgan Kaufmann, San Francisco, 1991, pp. 331–337.
13. T. Hogg, B.A. Huberman, and C.P. Williams, eds., "Frontiers in Problem Solving: Phase Transitions and Complexity," *Artificial Intelligence*, Vol. 81, 1996.
14. J. Frank, P. Cheeseman, and J. Stutz, "When Gravity Fails: Local Search Topology," *J. Artificial Intelligence Research*, Vol. 7, 1997, pp. 249–281.
15. T. Hogg, *Single-Step Quantum Search Using Problem Structure,* Los Alamos preprint quant-ph/9812049, Los Alamos Nat'l Lab, Albuquerque, N.M., 1998.
16. S. Haroche and J.-M. Raimond, "Quantum Computing: Dream or Nightmare?" *Physics Today*, Vol. 49, Aug. 1996, pp. 51–52.

Quantum computational intelligence: answers and questions

Dan Ventura, fonix *Corp.*

In 1994, Peter Shor's discovery of an algorithm for factoring large numbers in polynomial time using a quantum computer transformed the field of quantum computation from a theoretical curiosity to a potential technology of international interest.[1] The appeal of a computational paradigm with a potentially exponential increase in capacity over classical approaches dramatically increased research in the field. Interestingly, however, discoveries of other useful quantum algorithms have come few and far between. The ramifications of a quantum factoring algorithm on cryptography not withstanding, it is beginning to appear as if quantum computation is an answer looking for a question.

The field of computational intelligence, including the subfields of machine learning, neural networks, computational-learning theory, evolutionary computation, and symbolic AI, seeks to produce algorithms for solving problems that are intractable or have no closed-form solution or are in some other way unsuitable for traditional computational methods. Many successful applications of such techniques exist; however, due to the nature of the problems to which these technologies are applied, such successes are more often the exception than the rule. In other words, we might say that computational intelligence is a question looking for an answer.

Combining the fields

Perhaps the two fields can be combined to the advantage of both. Computational intelligence seeks to extend the capabilities of classical computers. As quantum computation begins to mature, is it not natural to attempt to extend its capabilities in a similar fashion, by developing a field of quantum computational intelligence? Conversely, perhaps developing quantum computational intelligence is critical to quantum computation's continued development as a computational science. (It is still very much a developing and valuable science from a physical standpoint, even if no other algorithmic developments occur.)

Quantum computation is probabilistic in nature and is computation based upon the time evolution of a physical system. Furthermore, this physical system obeys the laws of quantum mechanics, which can be extremely counterintuitive. Two important and unusually powerful ideas unique to quantum computation (as opposed to the classical sort) are quantum parallelism and entanglement. Quantum parallelism refers to the fact that a quantum system exists as a superposition of many states at once, and therefore computation involving the system is simultaneously applied to all states represented in the quantum system. Entanglement describes the nonclassical correlations that can exist between different quantum systems through which the system can be said to communicate. These concepts embody the unique capability that quantum computation has to perform an exponential amount of information processing within a polynomial amount of space and time.

Now, consider, for what is this type of computation really suited? We are still trying to figure this out. Quantum computation is very counterintuitive, even unsettling, from a physical standpoint; from a computational standpoint, it is not so much unsettling as it is just extremely different. I do not believe anyone really has a handle yet on how to think algorithmically in this way, and so I suspect that we have yet to produce the most important developments in the field. Shor's factoring algorithm is ingenious, and it poses a very real challenge to current standards in cryptography. However, one can always come up with codes that are not based upon the difficulty of factorization. In fact, quantum cryptography—the study of cryptographic systems based on quantum principles—is a burgeoning field in its own right. Also, there is no proof that what Shor did can't be done classically (although at the moment we suspect that it cannot). So, then, what of quantum computation?

Searching quantum phone books

Search is another problem for which interesting quantum computational algorithms have been discovered. For example, Lov Grover produced an algorithm for searching an unordered list of length N in $O(\sqrt{N})$ time[2] whereas classically the same task requires $O(N)$ time. In other words, a quantum computer with a quantum phone book can find the name associated with a particular phone number significantly faster than a classical computer with a classical phone book can. As impressive an achievement as Shor's algorithm is, I will argue that Grover's quantum search is even more important because it is more purely quantum in nature and because it is provably superclassical. Furthermore, I believe it is a better indicator of the future of quantum computation.

Of course, search is an extremely common theme in traditional AI, and many approaches to computational intelligence suffer from exponential explosions in computational requirements. Quantum computation naturally processes exponential amounts of information. Computational intelligence and quantum computation are both forms of computation that can be described as fuzzy, probabilistic, inexact, and nondeterministic, for example. Despite Shor's remarkable success, quantum computation appears, in general, to be much more amenable to computational intelligence-type problems rather than to traditional problems requiring exact, deterministic solutions.

In fact, this wedding of quantum computation and computational intelligence is beginning to bear fruit. Results have been published on

- quantum associative memories with storage capacities exponentially greater than their classical counterparts,[3]
- fascinating mathematical analogies between the quantum and neural network theories,[4]
- quantum computation for evaluating decision trees,[5]
- the construction of quantum Bayesian networks,[6]
- quantum extensions to genetic algorithms,[7]
- the implementation of a neural network using quantum dots,[8]
- a quantum computational learning algorithm for learning DNF formula,[9] and
- the implementation of competitive learning in a quantum system.[10]

Research such as this hints at the enormous possibility that a study of quantum computational intelligence possesses. However, the field, as such, is still in its infancy, and really the work cited here, while on the right track, nibbles more around the edges than it does embrace the full potential of this emerging science.

And, in fact, there exists an inherent difficulty in the endeavor of communicating across fields with radically different agendas, viewpoints, strategies, and nomenclatures. For example, to someone in computational intelligence who is familiar with inductive learning, the value of an algorithm for encoding a set of examples in a quantum state is fundamentally obvious. For a physicist, on the other hand, such an algorithm might seem esoteric at best. The situation gets even more complicated when we consider the fact that computational intelligence itself is extremely interdisciplinary in nature, consisting of ideas from computer science, mathematics, psychology, biology, and statistics, to name a few. Thus, progress in such an eclectic field as quantum computational intelligence is bound to be slow, especially at first.

A difficult honeymoon

This inertial effect is as easy to understand as it is difficult to overcome—the two fields of quantum mechanics and computational intelligence, which must be reconciled to produce useful quantum computational intelligence, are disparate almost to the extreme. One can almost be characterized as rigor for the sake of rigor (although physicists will take offense at this), while the other has prospered almost exclusively on empirical success (and computational intelligence practitioners will take offense at this). Less acerbically, we might say that empirical evidence without a theoretical basis is as terrible for a physicist as is a theoretical basis without empirical usefulness for a practitioner of

Coming Next Issue

Mixed-Initiative Interaction

James Allen, University of Rochester

Curry Quinn, Duke University

Eric Horvitz, Microsoft Research

computational intelligence.

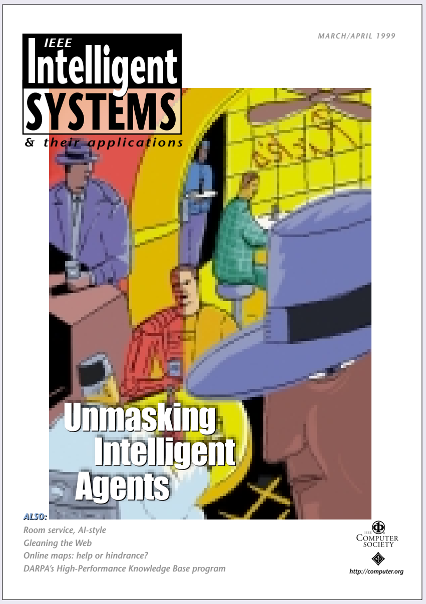
Although these generalizations are exaggerations, they do help emphasize the difficulty in producing interesting results as a union of the two sciences. Physics demands rigor and theoretical correctness. Computational intelligence demands practical application and empirical benefit. While these two approaches are hardly mutually exclusive, they are rarely considered together even within a single discipline, and the situation is exacerbated by the disparity between the two fields now attempting to unify. The difficulty is all the greater because, as is usually the case across disciplines, completely different languages are spoken and results are often needlessly reproduced for lack of sufficient communication.

We are just beginning to glimpse what can be done with quantum computation and so too with quantum computational intelligence. The kinds of problems to which computational intelligence are usually applied are often exponential in nature. Quantum computation performs an exponential amount of information processing in polynomial space and time, but most of this is usually unavailable to us. The trick is figuring out for what kinds of problems we can extract from the quantum computer something more than we could from a classical one. As we learn better how to do that, the field of quantum computational intelligence will become both the question to the answer and the answer to the question.

References

1. P. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM J. Computing*, Vol. 26, 1997, pp. 1484–1509.
2. L. Grover, "A Fast Quantum Mechanical Algorithm for Database Search," *Proc. ACM Symp. Theory of Computing*, ACM Press, New York, 1996, pp. 212–219.
3. D. Ventura and T. Martinez, "Quantum Associative Memory," to be published in *Information Sciences*, 1999.
4. M. Perus, "Neuro-Quantum Parallelism in Brain-Mind and Computers," *Informatica*, Vol. 20, 1996, pp. 173–183.
5. E. Farhi and S. Gutmann, "Quantum Computation and Decision Trees," *Physical Rev. A*, Vol. 58, No. 2, 1998, pp. 915–928.
6. R. Tucci, "Quantum Bayesian Nets," *Int'l J. Modern Physics*, Vol. B9, 1995, pp. 295–337.
7. A. Narayanan and M. Moore, "Quantum-Inspired Genetic Algorithms," *Proc. IEEE Int'l Conf. Evolutionary Computing*, IEEE Press, Piscataway, N.J., 1996, pp. 61–66. 8. E. Behrman et al., "A Quantum Dot Neural Network," *Proc. Workshop Physics of Computation*, New England Complex Sys. Inst., Cambridge, Mass., 1996, pp. 22–24.
9. N. Bshouty and J. Jackson, "Learning DNF over the Uniform Distribution Using a Quantum Example Oracle," *Proc. Eighth Ann. Conf. Computational Learning Theory*, ACM Press, New York, 1995, pp. 118–127.
10. D. Ventura, "Implementing Competitive Learning in a Quantum System," to be published in *Proc. Int'l Joint Conf. Neural Networks*, IEEE Computer Soc. Press, Los Alamitos, Calif., 1999.

July/August Issue



Special Focus

Intelligent Integration and Use of Large Test, Image, Video, and Audio

This special track will describe current research in, and potential contributions of, intelligent, integrated systems that use text, image, video, and audio (TIVA) sources. Articles will deal with two aspects of the exploitation of these sources: What intelligent systems can be built to use or reuse these sources? How can these sources help create smarter programs?

Knowledge Management and the Internet

Knowledge management involves explicit and persistent representation of an organization's knowledge, to improve the organization's activities. When this knowledge is distributed among several experts and documents located all over the world, the Internet or an intranet and World Wide Web techniques can be the means for its acquisition, modeling, and management.

IEEE Intelligent Systems

AI News You Can Use