A Programmer's Survey of the Quantum Computing Paradigm

Philippe Jorrand

***Abstract*—** Research in quantum computation is looking for the consequences of having information encoding, processing and communication exploit the laws of quantum physics, i.e. the laws which govern the ultimate knowledge that we have, today, of the foreign world of elementary particles, as described by quantum mechanics. This paper starts with a short survey of the principles which underlie quantum computing, and of some of the major breakthroughs brought by the first ten to fifteen years of research in this domain; quantum algorithms and quantum teleportation are very biefly presented. The next sections are devoted to one among the many directions of current research in the quantum computation paradigm, namely quantum programming languages and their semantics. A few other hot topics and open problems in quantum information processing and communication are mentionned in few words in the concluding remarks, the most difficult of them being the physical implementation of a quantum computer. The interested reader will find a list of useful references at the end of the paper.

***Keywords*—** Quantum information processing, quantum algorithms, quantum programming languages.

I. Introduction

INFORMATION is physical: the laws which govern its encoding, processing and communication are bound by those of its unavoidably physical incarnation. In today's informatics, information obeys the laws of Newton's and Maxwell's classical physics: this statement holds all the way from commercial computers down to (up to?) their most abstracted models like Turing machines and lambda-calculus. Today's computation is classical.

Quantum information processing and communication was born some twenty years ago, as a child of two major scientific achievements of the 20th century, namely quantum physics and information sciences. The driving force of research in quantum computation is that of looking for the consequences of having information encoding, processing and communication based upon the laws of quantum physics, i.e. the ultimate knowledge that we have, today, of the foreign world of elementary particles, as described by quantum mechanics. The principles and the major results of quantum information processing are very briefly introduced in this paper. For a more detailed, but still concise and gentle introduction, see [24]. A pedagogical and rather thorough textbook on quantum computing is [21]. For a dense and theoretically profound presentation, the reader is referred to [16].

Manuscript received September 15, 2001. This work was supported in part by Centre National de la Recherche Scientifique, France, and by Université Joseph Fourier, Grenoble, France.

Dr. Philippe Jorrand is with Leibniz Laboratory, 46 avenue Felix Viallet, 38000 Grenoble, France (phone: +33 4 76 57 46 47; fax: +33 4 76 57 46 02; e-mail: Philippe.Jorrand@imag.fr).

II. From Physics to Computing

Quantum mechanics, which is the mathematical formulation of the laws of quantum physics, relies on four postulates: (*i*) the state of a quantum system (i.e. a particle, or a collection of particles) is a unit element of a Hilbert space, that is a vector of norm 1 in a *d*-dimensional complex vector space; (*ii*) the evolution of the state of a closed quantum system (i.e. not interacting with its -classical- environment) is deterministic, linear, reversible and characterized by a unitary operator, that is by a dxd unitary matrix applied to the state vector; (*iii*) the measurement of a quantum system (i.e. the observation of a quantum system by its -classical- environment) irreversibly modifies the state of the system by performing a projection of the state vector onto a probabilistically chosen subspace of the Hilbert space, with renormalization of the resulting vector, and returns a value (e.g. an integer) to the classical world, which just tells which subspace was chosen; and (*iv*) the state space of a quantum system composed of several quantum subsystems is the tensor product of the state spaces of its components (given two vector spaces P and Q of dimensions p and q respectively, their tensor product is a vector space of dimension pxq).

The question is then: how to take advantage of these –rather strange- postulates to the benefits of computation? The most widely developed approach to quantum computation exploits all four postulates in a straightforward manner. The elementary physical carrier of information is a qubit (quantum bit), i.e. a quantum system (electron, photon, ion, ...) with a 2-dimensional state space (postulate *i*); the state of a n-qubit register lives in a 2^n-dimensional Hilbert space, the tensor product of n 2-dimensional Hilbert spaces (postulate *iv*). Then, by imitating in the quantum world the most traditional organization of classical computation, quantum computations are considered as comprising three steps in sequence: first, preparation of the initial state of a quantum register (postulate *iii* can be used for that, possibly with postulate *ii*); second, computation, by means of deterministic unitary transformations of the register state (postulate *ii*); and third, output of a result by probabilistic measurement of all or part of the register (postulate *iii*).

III. Quantum Ingredients for Information Processing

These postulates and their consequences can be interpreted from a more informational and computational point of view, thus providing the elementary quantum ingredients which are at the basis of quantum algorithm design.

A. Superposition

At any given moment, the state of quantum register of n qubits is a vector in a 2^n-dimensional complex vector space, i.e. a vector with at most 2^n non zero complex components, one for each of the 2^n different values on n bits: the basis of this vector space comprises the 2^n vectors $|i\rangle$, for i in $\{0,1\}^n$ ($|i\rangle$ is Dirac's notation for vectors denoting quantum states). This fact is exploited computationally by considering that this register can actually contain a superposition of all the 2^n different values on n bits, whereas a classical register of n bits may contain only one of these values at any given moment.

B. Quantum Parallelism and Deterministic Computation

Let f be a function from $\{0,1\}^n$ to $\{0,1\}^m$ and x be a quantum register of n qubits initialized in a superposition of all values in $\{0,1\}^n$ (this initialization can be done in one step by a very simple operation). Then, computing $f(x)$ is achieved by a deterministic, linear and unitary operation applied to the state of x: by linearity, a single application of this operation distributes over all 2^n dimensions and produces all 2^n values of f in a single computation step. Performing this operation for any, possibly non linear f while obeying the linearity and unitarity laws of the quantum world, requires a register of $n+m$ qubits formed of the register x, augmented with a register y of m qubits. Initialy, y is in any arbitray state $|s\rangle$ on m qubits: before the computation of f, the larger register of $n+m$ qubits contains a superposition of all pairs $|i,s\rangle$ for i in $\{0,1\}^n$. After the computation of f, it contains a superposition of all pairs $|i, s\oplus f(i)\rangle$ for i in $\{0,1\}^n$, where $\oplus$ is bitwise addition modulo 2. It is easy to verify that, for any f, this operation on a register of $n+m$ qubits is unitary (it is in fact its own inverse). In many cases, it will be applied with $s=0$, which results in a superposition of all simpler pairs $|i, f(i)\rangle$ for i in $\{0,1\}^n$.

C. Probabilistic Measurement and Output of a Result

After f has been computed, all its values $f(i)$, for i in $\{0,1\}^n$, are superposed in the y part (m qubits) of the register of $n+m$ qubits, each of these values facing (in the pair $|i,f(i)\rangle$) their corresponding i which is still stored in the unchanged superposition contained in the x part (n qubits) of that register. Observing the contents of y will return only one value, j, among the possible values of f. This value is chosen with a probability which depends on f since, e.g. if $f(i)=j$ for more than one values of i, the probability of obtaining j as a result will be higher than that of obtaining k if $f(i)=k$ for only one value of i (and the probability of obtaining l if there is no i such that $f(i)=l$ will of course be 0). This measurement also causes the superposition in y to be projected onto the 1-dimensional subspace corresponding to the basis state $|j\rangle$, i.e. the state of the y part collapses to $|j\rangle$, which implies that all other values of f which were previously superposed in y are irreversibly lost.

D. Interference

Using appropriate unitary operations, the results of the 2^n parallel computations of f over its domain of definition can be made to interfere with each other. Substractive interference will lower the probability of observing some of these value in y, whereas additive interference will increase the probability of observing other values and bring it closer to 1. Because of probabilistic measurement, a major aim of the organization and principles of quantum algorithms will be to assemble the unitary operations for a given computation in such a way that, when a final measurement is applied, a relevant result has a high probability to be obtained. The whole game of quantum algorithmics is precisely to assemble a minimal number of well chosen unitary operations (the quantum algorithm) so that a measurement of the final state will have probablity as close to 1 as possible to produce a correct result.

E. Entangled States

Measuring y after the computation of f is in fact measuring only m qubits (the y part) among the $n+m$ qubits of a register. The state of this larger register is a superposition of all pairs $|i,f(i)\rangle$ for i in $\{0,1\}^n$ (e.g., in this superposition, there is no pair like $|2,f(3)\rangle$): this superposition is not a free cross-product of the domain $\{0,1\}^n$ of f by its image in $\{0,1\}^m$, i.e. there is a strong correlation between the contents of the x and y parts of the register. As a consequence, if measuring the y part returns a value j, with the state of that part collapsing to the basis state $|j\rangle$, the state of the larger register will itself collapse to a superposition of all remaining pairs $|i,j\rangle$ such that $f(i)=j$. This means that, in addition to producing a value j, the measurement of the y part also causes the state of the x part to collapse to a superposition of all elements of the $f^{-1}(j)$ subset of the domain of f. This correlation between the x and y parts of the register is called entanglement: in quantum physics, the state of a system composed of r sub-systems is not, in general, simply reducible to an r-tuple of the states of the components of that system. Entanglement has no equivalent in classical physics and it constitutes the most powerful resource for quantum information processing and communication.

F. No-Cloning

A direct consequence of the linearity of all operations that can be applied to quantum states (a two line trivial proof shows it) is that the state of a qubit a (this state is in general an arbitrary superposition, i.e. a vector made of a linear combination of the two basis state vectors $|0\rangle$ and $|1\rangle$), cannot be duplicated and made the state of another qubit b, unless the state of a is simply either $|0\rangle$ or $|1\rangle$ (i.e. not an arbitrary superposition). This is true of the state of all quantum systems, including of course registers of qubits used during a quantum computation. In programming terms, this means that the "value" (the state) of a quantum variable cannot be copied into another quantum variable.

These basic quantum ingredients and their peculiarities will of course have far reaching consequences, as soon as

algorithm, programming languages and semantic frameworks incorporate and make use of quantum resources.

IV. Quantum Algorithms

Richard Feynman launched in 1982 [10] the idea that computation based upon quantum physics would be exponentially more efficient than based upon classical physics. Then, after the pioneering insight of David Deutsch in the mid eighties [8], who showed, by means of a quantum Turing machine, that quantum computing could indeed not, in general, be simulated in polynomial time by classical computing, it was ten years before the potential power of quantum computing was demonstrated on actual computational problems.

The first major breakthrough was by Peter Shor [27]: in 1994, he published a quantum algorithm operating in polynomial time ($O(\log^3 N)$) for factoring an integer N, whereas the best classical algorithm is exponential. Shor's algorithm relies on a known reduction of the problem of factoring to that of finding the order of a group, or the period of a function: then, since order finding can be achieved by a Fourier Transform, the key of Shor's algorithm is a Quantum Fourier Transform, which is indeed exponentially more efficient than classical FFT (Fast Fourier Transform), thanks to quantum parallelism, entanglement and tensor product. The exponential drop of complexity brought by Shor's algorithm has dramatic consequences in classical cryptography, e.g. RSA, where the security precisely relies upon the difficulty of factoring large integers. Once a running quantum computer is available, most currently used systems for secure communications are breakable in a few seconds.

Two years later, in 1996, Lov Grover [13] published a quantum algorithm for searching an unordered database of size N, which achieves a quadratic acceleration (it operates in $O(N^{1/2})$) when compared with classical algorithms for the same problem (in $O(N)$). Grover's algorithm relies upon a very subtle use of interference, now known as amplitude amplification, which performs a stepwise increase of the probability of measuring a relevant item in the database, and which brings this probability very close to 1 after $N^{1/2}$ steps. Although less impressive than the exponential drop in complexity of Shor's algorithm, the quadratic drop of complexity of Grover's algorithm has a much wider range of applications, namely in information retrievial.

Since then, these results have been generalized and extended to related classes of problems. Shor's algorithm solves an instance of the hidden subgroup problem [19] for abelian groups and a few extensions to non-abelian cases have been designed. In addition to Quantum Fourier Transform, order finding and amplitude amplification, other candidates to the status of higher level building blocks for quantum algorithmics have emerged, such as quantum random walks on graphs [15]. Principles for distributed quantum computing have also been studied and successfully applied to a few classes of problems with, in some cases, an exponential drop in communication complexity. Very recently, on the basis of amplitude amplification, quadratic and other quantum speedups have been found for several problems on graphs, such as connectivity, minimum spanning tree and single source shortest paths [9].

V. Teleportation

Another major result, by Charles Bennet and others in 1993 [3], was the design of theoretical principles leading to a quantum teleportation protocol, which takes advantage of entanglement and of probabilistic measurement: the state of a quantum system a (e.g. a qubit) localized at A's place can be assigned, after having been measured, thus destroyed, to another quantum system b (e.g. another qubit), localized at B's place, without the state of a being known neither by A nor by B, and without neither a nor any other quantum system carrying the state of a being moved along a trajectory between A and B. It is important to notice that this is not in contradiction with no-cloning: there is still only one instance of the teleported state, whereas cloning would mean that there coexist one original and one copy.

Teleportation also has been generalized. The measurement used in its original formulation was such that the state eventually obtained for b was the same as the state initially held by a (up to a correcting operation which still had to be applied, depending on the probabilistic outcome of that measurement). By changing the way the measurement is done (in fact, by appropriately rotating the basis upon which the measurement of a will project the state of a), it has been found that the state teleported to b could be not the state initially held by a, but that state to which a rotation, i.e. a unitary operation has been applied. In other words, entanglement and measurement, i.e. the resources needed by teleportation, can be used to simulate computations by unitary tranformations. This has given rise to a whole new direction of research in quantum computation, namely measurement-based quantum computation [14,18,23].

VI. Quantum And Classical

There is an implicit, but obvious and ever present invariant in all these different ways of organizing quantum computations and quantum algorithms. Quantum computations operate in the quantum world, which is a foreign and unknowable world. No one in the classical world will ever know what the superposition state of an arbitrary qubit is, the only information one can get is 0 or 1, through measurement, i.e. the classical outcome of a probabilistic projection of the qubit state vector onto basis vectors |0> or |1>: if one gets 0, the only actual information which is provided about the state before measurement is that it was not |1>, because |0> and |1> are orthogonal vectors. Then, for the results of quantum computations to be useful in any way, there is an intrinsic necessity of cooperation and communication controlled by the classical world. All quantum algorithms, either based upon unitary transformations or upon measurements, if they are of any relevance, eventually end up in a final quantum state which hides, among its superposed basic states, a desired result. Such a result is asked for upon request by the classical world, which decides at that point to

perform a measurement on part or all of the quantum register used by the computation. But measurement is probabilistic: its outcome may be a desired result, but it may well be something else. For example, Grover's algorithm ends up in a state where desired results have a probability very lose to 1 to be obtained, but other, unwanted results may also come out from the final measurement, although with a much lower probability.

The whole game of quantum algorithmics is thus to massage the state of the quantum register so that, in the end, desired results have a high probability to be obtained, while doing that at the minimum possible cost, i.e. minimal number of operations applied (time) and of qubits used (space). This is achieved through interferences (by means of appropriate unitary operations), through the establishment of entangled states and through measurements in appropriate bases. But this is not the end: once a measurement outcome is obtained by the classical world, it must be checked, by the classical world, for its validity. If the result satisfies the required conditions to be correct, termination is decided by the classical world. If it does not, the classical world decides to start the quantum part of the computation all over. For example, in the case of Grover's algorithm, if the element of the database produced by the measurement is not correct, the whole quantum search by amplitude amplification is started again by the classical world.

In general, algorithms will not contain one, but several quantum parts embedded within classical control structures like conditions, iterations, recursions. Measurement is not the only channel through which the classical and quantum worlds interact, there is also the initialization of quantum registers to a state chosen by the classical world (notice that such initializations can only be to one among the basis states, since they are the only quantum states which correspond, one to one, to values expressible by the classical world). A quantum part of an algorithm may also, under the control of the classical world, send one of its qubits to another quantum part. Notice that the physical carrier of the qubit must be sent, not its state, because of no-cloning. This quantum to quantum communication is especially useful for quantum cryptographic communication protocols, a family of distributed quantum algorithms of high relevance, in the very near future, among the commercial applications of quantum information processing.

This means that not only the peculiarities of the basic quantum ingredients for computing have to be taken into account in the design of languages for the formal description of quantum algorithms and quantum protocols, but also the necessity of embedding quantum computations within classical computations, of having both worlds communicate and cooperate, of having classical and quantum parts be arbitrarily intermixed, under the control of the classical side, within the same program.

VII. Quantum Programming

While quantum computing is in its infancy, quantum programming is still in embryonic state. Quantum computing is on its way to becoming an established discipline within computer science, much like, in a symmetric and very promising manner, quantum information theory is becoming a part of quantum physics. Since the recent birth of quantum computing, the most important efforts have been invested in the search for new quantum algorithms that would show evidence of significant drops in complexity compared with classical algorithms. Obtaining new and convincing results in this area is clearly a crucial issue for making progress in quantum computing. This research has been, as could be expected, largely focusing on complexity related questions, and relying on approaches and techniques provided by complexity theory.

However, the much longer experience from classical computer science tells that the study of complexity issues is not the only source of inspiration toward the creation, design and analysis of new algorithms. There are other roads, which run across the lands of language design and semantics. A few projects in this area have recently started, following these roads. Three quantum programming language styles are under study: imperative, parallel and distributed, and functional. This naturally opens new and challenging research issues in the domain of semantic frameworks (operational, denotational, axiomatic), where the peculiarities of quantum resources have to be dealt with in a formal, mathematical and consistent fashion. This research, in turn, is expected to provide fresh insights into the properties of the quantum world itself.

The sequential and imperative programming paradigm, upon which all major quantum algorithmic breakthroughs have relied, is still widely accepted as "the" way in which quantum + classical computations are organized and should be designed. However, before any language following that style was designed, and even today, the quantum parts of algorithms are mostly described by drawing pictures of quantum gate networks, which are to quantum computing what logical gate circuits are to classical computing. This is of course very cumbersome and far from useful for understanding and proving properties of programs. This is why some imperative languages for quantum + classical programming have been design first.

The most representative quantum imperative programming language is QCL (Quantum Computing Language), a C flavoured language designed by B. Ömer at the University of Vienna [22]. Another one, qGCL (Quantum Guarded Command Language) was due to P. Zuliani at Oxford University [30], with the interesting design guideline of allowing the construction by refinement of proved correct programs.

Functional programming offers a higher level of abstraction than most other classical programming paradigms, especially than the imperative paradigm. Furthermore, it is certainly one of the most fruitful means of expression for inventing and studying algorithms, which is of prime importance in the case of quantum computing. A natural way to try and understand precisely how this programming style can be transposed to quantum computing is to study quantum versions of lambda-calculus.

This is being done, among others, by A. Van Tonder at Brown University [28]. His approach puts forward the fact that there is a need for new semantic bases in order to accommodate disturbing peculiarities of the quantum world. A striking example are the consequences of no-cloning. In quantum programs, there are quantum variables, i.e. variables storing quantum states. However, since it is impossible to duplicate the state of a qubit, it is impossible to copy the value of a quantum variable. This has far reaching consequences, e.g., in lambda-calculus, an impossibility to stay with classical beta-reduction for representing function application. Van Tonder [29] and J.Y. Girard [12] are suggesting that linear logic may be the way out of this specifically quantum issue.

On the functional side, there is also QPL (a Quantum Programming Language), designed by P. Selinger at the University of Ottawa [26]. QPL is a simple quantum programming language with high-level features such as loops, recursive procedures, and structured data types. The language is functional in nature, statically typed, and it has an interesting denotational semantics in terms of complete partial orders of superoperators (superoperators are a generalization of quantum operations). All of these authors are still fighting toward a satisfactory consistent integration of all quantum peculiarities, i.e. not only no-cloning, which naturally comes as their first major concern, but also probabilistic measurement, the necessary presence of both quantum and classical data and operations, etc.

The third style, process calculi, are an abstraction of communicating and cooperating computations which take place during the execution of parallel and distributed programs. They form a natural basis for rigorous and high level expression of several key aspects of quantum information processing: measurement, cooperation between quantum and classical parts of a computation, multi-party quantum computation, description and use of teleportation and of its generalizations, description and analysis of quantum communication and cryptographic protocols. Representatives of this approach are CQP (Communicating Quantum Processes) , which is being designed by S. Gay and R. Nagarayan at the Universities of Warwick and Glasgow [11], and QPAlg (Quantum Process Algebra), designed by M. Lalire and Ph. Jorrand at the University of Grenoble [17]. Both CQP and QPAlg have formally defined operational semantics, in the Plotkin's inference rules style, which include a treatment of probabilistic transitions due to the measurement postulate of quantum mechanics. All of this, of course, is still ongoing research

VIII. Issues in Semantics

All the language designs for quantum programming are still at the stage of promising work in progress. The core issues clearly remain at the semantics level, because of the many non-classical properties of the quantum world. No-cloning, entanglement, probabilistic measurement, mixed states (a more abstract view of quantum states, for representing probabilistic distributions over pure states), together with the necessary presence of both worlds, classical and quantum, within a same program, call for further in depth studies toward new bases for adequate semantic frameworks.

Operational semantics (i.e. a formal description of how a quantum + classical program operates) is the easiest part, although probabilities, no-cloning and entanglement already require a definitely quantumized treatment. For example, leaving the scope of a quantum variable is not as easy as leaving the scope of a classical variable, since the state of the former may be entangled with the state of more global variables. Several of the languages mentionned previously have their semantics defined in the operational style. But, even in this rather naïve approach to semantics, much remains to be done, like, in the process calculi approach, the definition of an equivalence among processes. This would not only provide a more satisfying and abstract semantics, but also allow a rigorous and formal approach to a number challenging questions in quantum computing. For example, it is known that quantum computations described by unitary transformations can be simulated by using measurements only [14,18,23], and that quantum computation by measurements is a way to get around decoherence, which is the major obstacle on the way to the physical implementation of a quantum computer (see the comments in the concluding remarks). Then, it would be very useful to make sure, upon well founded formal bases, that a computation specified by means of unitary transformations is indeed correctly implemented by means of measurements.

Axiomatic semantics (what does a program do? How to reason about it? How to analyze its properties, its behaviour?) is a very tricky part. Defining quantum versions of Hoare's logic or Dijkstra's weakest precondition would indeed provide logical means for reasoning on quantum + classical programs and protocols and constitute formal bases for developing and analyzing such systems. Some attempts toward a dynamic quantum logic, based on the logical study of quantum mechanics initiated in the thirties by Birkhoff and von Neumann [4] have already been made, for example by Brunet and Jorrand [5], but such approaches rely upon the use of orthomodular logic, which is extremely uneasy to manipulate. Of much relevance, and in the same direction, is the recent work of D'Hondt and Panangaden on quantum weakest preconditions [7], which establishes a semantic universe where programs written in QPL [26] can be interpreted in a very elegant manner.

Another long-term goal is the definition of a compositional denotational semantics which would accommodate quantum as well as classical data and operations, and provide an answer to the question: what is a quantum + classical program, which mathematical object does it stand for? Working toward this objective has been attempted by P. Selinger with QPL, although there are still major difficulties with second order functions. Recent results on categorical semantics for quantum information processing by Abramsky and Coecke [1,2], and other different approaches like the the work of van Tonder [29] and the interesting manuscript of J. Y. Girard [12] on the relations between quantum computing and linear logic, are

also worth considering for further research in those directions.

In fact, there are still a great number of wide open issues in the domain of languages for quantum programming and of their semantics. For a compilation of recent results and an overview of significant ongoing research on all these topics, the interested reader is referred to [25].

IX. Concluding Remarks

The preceeding sections provide a very partial and biaised survey of the current status of research in the quantum information processing and communication paradigm. There are many other hot topics: quantum algorithms and quantum complexity, of course, but also distributed quantum computation and quantum communication complexity, quantum cryptography and quantum secret sharing, quantum information theory and quantum communication channels, understanding and characterizing entangled quantum states, measurement based quantum computation and other non standard principles for quantum information processing (adiabatic quantum computation, topological quantum computation), formal models, abstract machines, languages and machine architectures for quantum computing, and, last but not least, physical implementation of a quantum computing device.

This is currently viewed as the most difficult issue. Physicists are still looking for a way to inscribe the qubit in some suitable material substrate. Several avenues are being explored, among them nuclear magnetic resonance (NMR), trapped ions, trapped neutral atoms, optics (photon=qubit), electronic spins, Josephson's junctions, and others. Criterias for a suitable qubit implementation have been agreed upon: qubits must be initialisable in some standard state (e.g. $|0\rangle$), a set of basic unitary operations and measurements (quantum instruction set) must be applicable and provide universality, the technology must be scalable (i.e. allow a significant number of qubits to co-exist and be usable within the same architecture) and, probably the most crucial obstacle opposed by the quantum world, qubits must stay in a coherent state (i.e. not entangled with their surrounding physical environment) during a sufficiently long time, so that an operation, possibly followed by error recovery, can be applied correctly. This time is currently estimated at 10^4 times the time needed for an elementary unitary operation. A recent study [31] seems to indicate that some technologies are doomed to fail (e.g. NMR is not scalable), whereas others are rather promising (e.g. trapped ions and Josephson's junctions). In any case, the most optimistic physicists expect a quantum computer of reasonable size not before 15 to 20 years from now.

References

[1] Abramsky, S., Coecke, B.: Physical traces: Quantum vs. Classical Information Processing. In: Blute, R., Selinger, P. (eds.): Category Therory and Computer Science (CTCS'02). Electronic Notes in Theoretical Computer Science 69, Elsevier (2003)

[2] Abramsky, S., Coecke, B.: A Categorical Semantics of Quantum Protocols. In: Ganzinger, H. (ed.): Logic in Computer Science (LICS 2004). IEEE Proceedings 415-425 (2004)

[3] Bennet, C., Brassard, G., Crepeau, C., Jozsa, R., Peres, A., Wootters: Teleporting an Unknown Quantum State via Dual Classical and EPR Channels. Physical Review Letters, 70:1895-1899 (1993)

[4] Birkhoff, G., von Neumann, J.: Annals of Mathematics 37, 823 (1936)

[5] Brunet, O., Jorrand, P.: Dynamic Logic for Quantum Programs. International Journal of Quantum Information (IJQI). World Scientific, 2(1):45-54 (2004)

[6] Carloza, D., Cuartero, F., Valero, V., Pelayo, F.L., Pardo, J.: Algebraic Theory of Probabilistic and Nondeterministic Processes. The Journal of Logic and Algebraic Programming 55(1-2):57-103 (2003)

[7] D'Hondt, E., Panangaden, P.:Quantum Weakest Preconditions. In [25]

[8] Deutsch, D.: Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. In: Proceedings Royal Society London A, 400:97 (1985)

[9] Durr, C., Heiligman, M., Hoyer, P., Mhalla, M.: Quantum Query Complexity of some Graph Problems. In: Diaz, J. (ed): International Colloquium on Automata, Languages and Programming (ICALP'04), Lecture Notes in Computer Science, Vol. 3142, Springer-Verlag, 481-493 (2004)

[10] Feynmann, R.P.: Simulating Physics with Computers. International Journal of Theoretical Physics 21:467 (1982)

[11] Gay, S.J., Nagarajan, R.: Communicating Quantum Processes. In [25]

[12] Girard, J.Y.: Between Logic and Quantic: a Tract. Unpublished manuscript (2004)

[13] Grover, L.K.: A Fast Quantum Mechanical Algorithm for Database Search. In: Proceedings 28th ACM Symposium on Theory of Computing (STOC'96) 212-219 (1996)

[14] Jorrand, P., Perdrix, S.: Unifying Quantum Computation with Projective Measurements only and One-Way Quantum Computation. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0404125 (2004)

[15] Kempe, J.: Quantum Random Walks - An Introductory Overview. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0303081 (2003)

[16] Kitaev, A.Y., Shen, A.H., Vyalyi, M.N.: Classical and Quantum Computation. American Mathematical Society, Graduate Studies in Mathematics, 47 (2002)

[17] Lalire, M., Jorrand, P.: A Process Algebraic Approach to Concurrent and Distributed Quantum Computation: Operational Semantics. In [25]

[18] Leung, D.W.: Quantum Computation by Measurements. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0310189 (2003)

[19] Lomont, C.: The Hidden Subgroup Problem – Review and Open Problems. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0411037 (2004)

[20] Milner, R.: Communication and Concurrency. Prentice-Hall (1999)

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

[22] Ömer, B.: Quantum Programming in QCL. Master's Thesis, Institute of Information Systems, Technical University of Vienna (2000)

[23] Raussendorf, R., Browne, D.E., Briegel, H.J.: Measurement-based Quantum Computation with Cluster States. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0301052 (2003)

[24] Rieffel, E.G., Polak, W.: An Introduction to Quantum Computing for Non-Physicists. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/9809016 (1998)

[25] Selinger, P. (ed.): Proceedings of 2nd International Workshop on QuantumProgramming Languages http://quasar.mathstat.uottawa.ca/~selinger/qpl2004/proceedings.html (2004)

[26] Selinger, P.: Towards a Quantum Programming Language. Mathematical Structures in Computer Science. Cambridge University Press, 14(4):525-586 (2004)

[27] Shor, P.W.: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In: Proceedings 35th Annual Symposium on Foundations of Computer Science, IEEE Proceedings (1994)

[28] Van Tonder, A.: A Lambda Calculus for Quantum Computation. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0307150 (2003)

[29] Van Tonder, A.: Quantum Computation, Categorical Semantics and Linear Logic. Los Alamos e-print arXiv, http://arxiv.org/abs/quant-ph/0312174 (2003)

[30] Zuliani, P.: Quantum Programming. PhD Thesis, St. Cross College, Oxford University (2001)

[31] Advanced Research and Development Activity (ARDA): A Quantum Information Science and Technology Roadmap, http://qist.lanl.gov, (2004).