

QUANTI

A Multidisciplinary Knowledge-Based System for Quantum Information Processing

Esma AÏMEUR Emmanuel BLANCHARD Gilles BRASSARD Sébastien GAMBS

Université de Montréal, Département IRO
C.P. 6128, Succursale Centre-Ville, Montréal (Québec), H3C 3J7 Canada
E-mail: {aimeur,blanchae,brassard,gambssseb}@iro.umontreal.ca

ABSTRACT

Quantum Information Processing (QIP) is the new and burgeoning field that studies the implication of quantum mechanics for information processing purposes. It can be used in practice to provide for unbreakable cryptography and it offers in principle a new computing paradigm that sometimes allows for exponential speedups. In particular, quantum computers, should they ever be built, have the potential to bring to its knees most of contemporary cryptography as currently used over the Internet to protect sensitive information such as credit card numbers. Despite its recent popularity, the experts are still in short supply and in high demand, whence the need for a knowledge-based system. What makes this goal both interesting and challenging is the fact that QIP is at the crossroads of computer science, mathematics, physics and chemistry. The multidisciplinary nature of this field makes it necessary to develop a system that will be easy to use by experts as well as students coming from each of the underlying disciplines. Among the tasks that we address is the design of a robust knowledge elicitation process. For this purpose, a novel elicitation algorithm is developed to capture and structure the knowledge of experts across disciplines. This process is generic and could be used in a variety of multidisciplinary contexts other than QIP. Our knowledge base can serve as a curriculum in the field of Intelligent Tutoring Systems.

KEYWORDS: Authoring tools, Knowledge acquisition, Computer-aided education, Quantum computing.

INTRODUCTION

Our goal is to develop *QUANTI*, a multidisciplinary knowledge base for Quantum Information Processing (QIP), which is a new and burgeoning discipline at the crossroads of computer science, mathematics, physics and chemistry [5,9]. The multidisciplinary nature of this field makes it all the more challenging and interesting for the development of a knowledge base because it is crucial to be able to cater to users coming from all four underlying disciplines.

In view of the growing interest for quantum information processing on the international scene, and of the current short supply of experts in this young discipline, the need for this knowledge base is undeniable. In particular, it can serve as a repository to store valuable expertise in the domain and it can be at the core of an Intelligent Tutoring System [1]. The potential impact of this knowledge base goes far beyond teaching. It can be used by experts in some disciplines to access the expertise of colleagues from another or from their own discipline. It could also play a role in discovering new concepts and proving new theorems.

The knowledge base is organized in a hierarchical manner. It consists of several semantic networks of concepts. Each concept in turn is a graph whose nodes (called *entities*) are entries in the repertory. We allow experts with diverse expertise to cooperate on the creation of the knowledge base by use of a well-defined protocol. Our approach is characterized by a novel algorithm for the knowledge base elicitation process, which is a hybrid between the classical depth-first and breadth-first search algorithms. This algorithm is somewhat reminiscent of the hill-climbing heuristic sometimes used for searching purposes.

There are several knowledge modelling techniques in the literature such as the *generic task*, proposed by Chandrasekaran [8]—also present with some variants in the KADS model [14] and in the model of *components of expertise* [13]—and the *role limiting method* of McDermott [11]. According to these techniques, there are two organizational principles: the *task-oriented principle* and the *domain-oriented principle*. Our approach belongs mostly to the latter. Like AQUINAS [4], DESIRE [6] and Common-Kads [10], we handle multiexpertise, but our shared model is domain-oriented.

After this Introduction, an overview of the field of quantum information processing is given. Our knowledge base elicitation process is described and illustrated in the following section. The architecture of the system and the methodology applied are then explained in detail, followed by a discussion of our future plans for an Intelligent Tutoring System. Finally, we conclude and give directions for the future in the last section.

QUANTUM INFORMATION PROCESSING

Throughout the second half of the twentieth century, *information theory* has grown into one of the most beautiful and fruitful branch of mathematics. It has become indispensable in many aspects of our data-driven society whenever information has to be stored, transmitted, processed or protected. Nevertheless, despite its undeniable successes, it is firmly rooted in classical physics, which is at best an approximation of the quantum world in which we live. This has prevented us from tapping the full potential of physical reality for information processing purposes. *Quantum mechanics*, which is perhaps the most successful scientific theory of all time, explains the behaviour of elementary particles such as photons, electrons and atoms. According to quantum mechanics, things do not behave at that scale the way we are used to in our macroscopic experience. For example, a particle can be simultaneously in two distinct places. Amply corroborated by experiments, the predictions of quantum mechanics have never been observed to be wrong, no matter how strange.

Quantum Information Processing is the new and exciting field that studies the implication of quantum mechanics for information processing purposes. In the past few years, it has grown tremendously in worldwide interest and activity, especially since Peter Shor's momentous discovery in 1994 that quantum computers, if they could be built, would bring to its knees the cryptographic schemes that are currently used on the Internet to protect the transmission of sensitive information such as credit card numbers [12]. Classical and quantum information can be used together to accomplish tasks that neither could achieve alone, such as quantum cryptography [2], quantum computing [5,9], quantum teleportation [3] and the computation of distributed tasks with vastly reduced communication cost [7]. Some of these concepts are still theoretical, but others have been implemented in the laboratory.

The key to understanding the foundations of QIP is the dichotomy that exists between the everyday notion of classical information and its less intuitive quantum counterpart. *Classical information* can be read, copied and transcribed into any medium; it can be transmitted and broadcast, but its transmission is limited by the speed of light. In contrast, *quantum information* cannot be read without disturbance, it cannot be copied or broadcast at all, but sometimes it *appears* to propagate instantaneously and it can exist in a superposition of classical states. More precisely, a quantum bit—known as a *qubit*—can be in a superposition of the two classical binary states (0 and 1) and an n -bit quantum register can be in an arbitrary superposition of all 2^n different classical states on strings of n bits.

Quantum cryptography exploits the impossibility of measuring quantum information reliably and the unavoidable disturbance caused by any attempt at such measurements. This phenomenon can be exploited to implement a cryptographic system that allows two people to communicate in absolute secrecy, even under the nose of an eavesdropper having access to unlimited computing

power. A prototype has been built in Switzerland, in which the cities of Nyon and Geneva, which are 23 kilometres apart, are linked by a quantum cryptographic channel that runs underneath Lake Geneva. The possibility of transmitting quantum states through satellites for cryptographic purposes is currently under investigation at the Los Alamos National Laboratory. Several other prototypes have been built round the world.

Quantum computers exploit the fact that a quantum system can be in a superposition of each of its possible classical states to perform exponentially many computations *simultaneously* in a single piece of hardware. In principle, more work can be done in a short quantum computation involving a few thousand quantum bits than would be possible for a classical computer the size of the Universe. Although this idea is still theoretical, several laboratories are actively working at implementing small-scale versions of quantum algorithms. This research has the potential to bring about a revolution in computer science that could be as spectacular and far-reaching as that created decades ago by the invention of the transistor.

Quantum teleportation conjures up images of *Star Trek*, but it is not science fiction. It uses some of the most counterintuitive aspects of quantum mechanics to allow for the transmission of quantum information through a classical channel, as illustrated in Figure 1. It is considered as one of the most fundamental corner stones of quantum information processing. The prestigious journal *Science* has declared its first experimental realization to be one of the ten most important scientific breakthroughs in 1998.

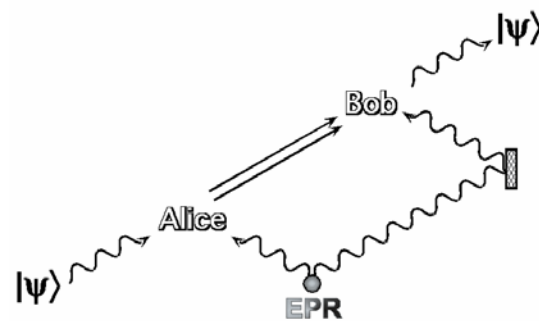


Figure 1: Diagram for quantum teleportation

KNOWLEDGE BASE ELICITATION PROCESS

The knowledge base is a network of concepts. Each concept is represented by a graph whose nodes are entries in the repertory. For instance, Figure 2 shows the graph that corresponds to the concept of *qubit*. The node corresponding to the concept itself serves as *root* in the data structure. Among other things, it specifies the identity of the author of the concept. The links between nodes can be of several types: *prerequisite*, *composition*, *property*, *association*, *example*, *non-example*, *exception*, *near-miss*, *analogy*, *exclusion*, etc. A node can be pointed at from different types of links: for instance, a given node can serve as an example for some concept and non-example for another. Each node, except root nodes, also points to its parent.

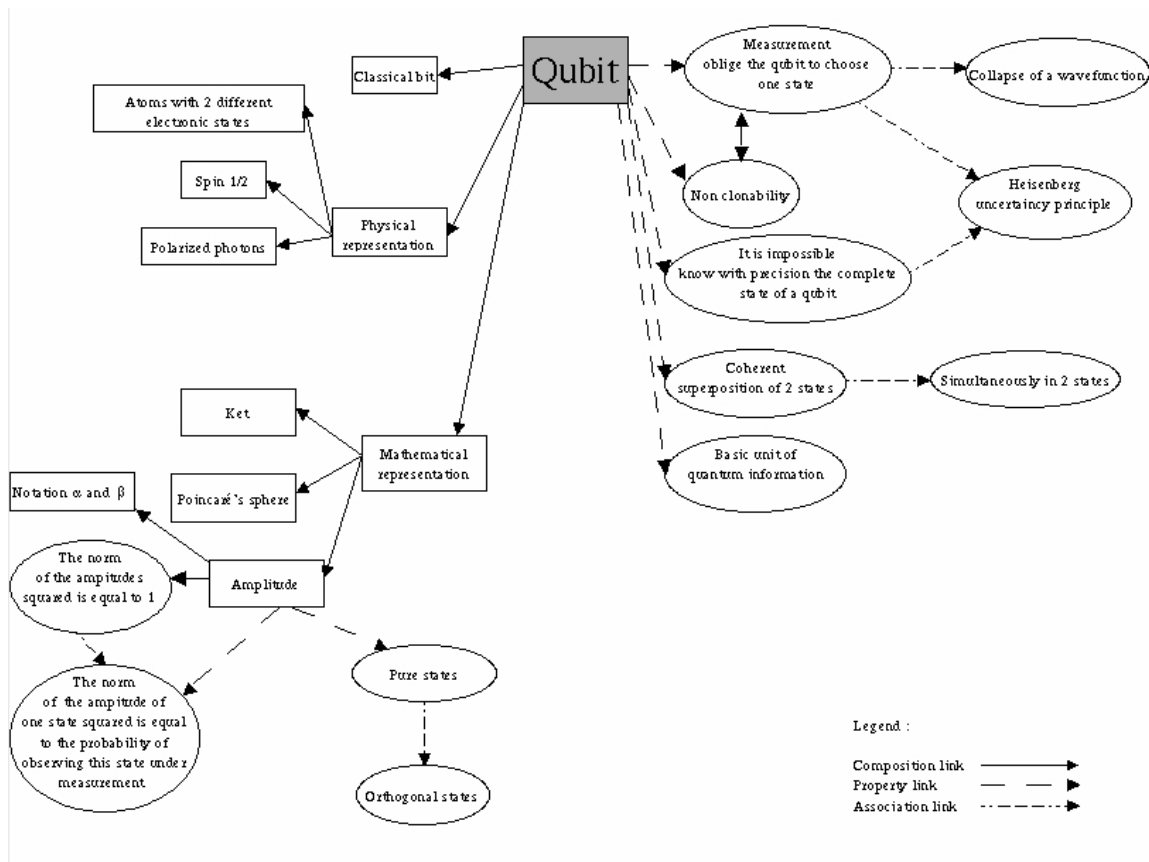


Figure 2: Graph for the concept of *qubit*.

Depth-first and breadth-first search are well-established techniques for *exploring* an explicit graph that already exists in the computer or an implicit graph for which the rules that define nodes and edges are well defined. However, neither of these techniques is well adapted for the human-driven *creation* of graphs in knowledge bases. If a human were asked to create such a graph in a depth-first manner, he would be required to complete the entire subgraph that stems from the first child of the root before anything about the other children is requested. By the time he is returned to the root, he may well have lost track of the list of children he might have had in mind at the start of the thought process. For instance, in the elicitation of the concept of *qubit*, the expert would have to provide all details on both the physical and mathematical representations of a qubit before he is allowed to mention the first-level subconcepts shown on the right of the root in Figure 2. On the other hand, in a breadth-first approach, the human would be asked at the outset to list all the children of the root—which is good—but the level-by-level scan inherent to breadth-first search would not allow him to concentrate on a given concept before his attention is requested on something else. Taking the *qubit* example of Figure 2 again, the expert would be asked to fill in “collapse of the wavefunction” at the far right before the “mathematical representation” has been fully elicited.

To circumvent these shortcomings, we propose a novel graph elicitation process, which is a hybrid between depth-first and breadth-first search. Our process, which is described formally in Figure 3, is reminiscent of the hill-climbing search heuristic, but it is used for creating a graph rather than for searching it. It uses both a stack—implicit in the recursive calls—and a queue. In essence, the expert is asked to start from the concept that the system wants to elicit from him, which serves as root. He is then invited to choose from the repertory all its immediate children in the graph and to specify the type of those children and of the links from the root to them. This elicitation process is then launched recursively from the first child; upon completion, it is launched again from the second child, etc. Even though the main process is guided, fine control remains in the hands of the expert, which allows him to add new nodes and new links at any time whenever the thought occurs to him: in this way, the knowledge base itself is incremental.

The process by which the graph given in Figure 2 was elicited is as follows. First, the system asks the expert to choose all the immediate successors for the concept of *qubit*. This choice is made from a repertory, in which experts can add new terms at any time. Here, the expert chooses *components* “classical bit”, “physical representation” and “mathematical representation”, and *properties* “Measurement obliges...”, “Non-clonability”, “It is impossible...”,

“Coherent superposition...” and “Basic unit of quantum information”. It is only after this list has been completed that these nodes are recursively elicited one by one from the expert. Our knowledge elicitation algorithm is currently under evaluation in comparison to more classical depth-first and breadth-first approaches.

We have described the process by which the graph corresponding to each concept is elicited from the experts. The knowledge base itself is elicited much in the same way. However, recall that the knowledge base consists of several networks of graphs of concepts. In the elicitation process for the knowledge base, concepts are considered to be atomic nodes. Consequently, in the depth-first part of the network elicitation process, concepts are *not* elicited when they are first encountered. Rather, the concept elicitation process described above is postponed until one or several networks have been elicited. However, experts can request concept elicitation at any time during the high-level knowledge base elicitation process.

The networks that compose the knowledge base can have different purposes. An important network is the *curriculum*—as described in the following sections—which links the concepts for intelligent tutoring purposes according to prerequisite constraints. Other networks can represent *procedures* to accomplish various tasks, such as the *implementation* of quantum cryptography, or *algorithmic tests* to determine if one bipartite state can be transformed into another by means of local operations and classical communication. Since the same concepts can be (logical) nodes in several distinct networks, the networks are in fact graphs of *pointers* to graphs of concepts.

ARCHITECTURE OF THE SYSTEM

The most important resource in any knowledge base are the experts who contribute to creating it. They have to be selected carefully and logged into the Authors Profile Database (see Figure 4). Even though the entire system is automatic once an expert belongs to this database, a human intervention is required to allow access to new experts. We do not describe here the criteria that could be used to make this kind of decision, because they are necessarily subjective. This is somewhat similar to the way most research journals are run: Associate Editors have full power to decide which papers to accept in their areas of expertise, but only the Editor-in-Chief is entitled to select the Editorial Board.

When an expert decides (or is asked) to contribute to the knowledge base, he must first identify himself to the system by means of a *password*. This is required in order to prevent unauthorized modifications to the databases. The elicitation process described in the previous section can then begin. Anything from the vocabulary (repertory) to graphs of concepts to semantic networks can then be elicited. But at all times, the system must be vigilant to verify that the overall *coherence* of the database is maintained. Problems that will necessarily occur in a multiexpert knowledge base are even more likely to come up in our multidisciplinary context, regardless of the professionalism of the experts. Because conflicts between experts are unavoidable, it is crucial to have a well-defined protocol to detect and resolve them, as described in the next two subsections. Moreover, a *repository* must be maintained to keep a trace of every action made by each of the experts.

```

Process initialization(root)
for each word in Dictionary do
  mark word has being not visited
  ask expert to choose the type of node root
  mark root has having been visited
  call elicitation(root)

Process elicitation(node)
initialize queue to an empty queue
{ breadth-first selection of all the immediate successors for node }
repeat
  ask author to select from Dictionary an entry succ for node to point at
  if succ has not been visited then
    mark succ has having been visited
    ask author to choose the type of node succ
    add succ to queue
  ask author to choose the type of link from node to succ
until author does not wish to select another successor for node
{ depth-first elicitation of those successors }
while queue is not empty do
  let succ be the first element of queue
  remove succ from queue
  call recursively elicitation(succ)

```

Figure 3: Knowledge elicitation algorithm

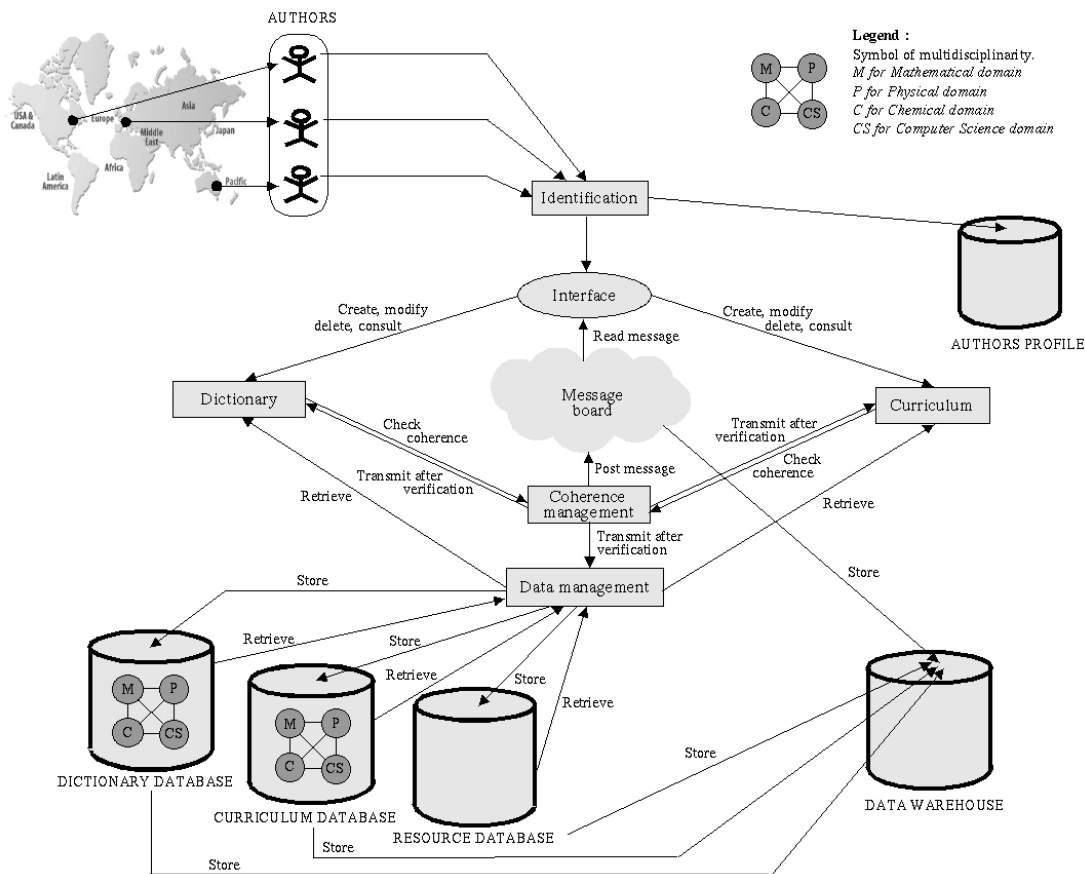


Figure 4: System architecture

Coherence Management

The Coherence Management process is used to make sure that all the databases are coherent, and that the syntax is respected. For example, when an expert wants to introduce a new entry in the repertory, the coherent management process will verify that this entry has not yet been entered. Also, if two entities have been declared synonym, the coherent management process will make sure that all occurrences of the second will be replaced by the first in the knowledge base.

Resolution of Conflicts Between Experts

There is no real conflict if two experts use different terms to refer to the same concept, but it is preferable if they can be made aware of any such occurrence and given the opportunity to make their vocabulary more uniform. A real conflict occurs when two experts use the same term to refer to two different concepts. Such conflicts, however, are not necessarily to be eliminated because they could stem from the diverse—and sometimes incompatible—background of the experts. For example, some terms can belong to more than one subdiscipline and have different definitions, such as *entropy*, which has a different meaning in mathematics

and in physics. Also, some concepts may evolve in time differently between disciplines.

Any expert is allowed to modify entities and concepts that he has created. However, if an expert wishes to modify an entry created by another expert, he must first ask permission to that expert through a *message board*. When this happens, four cases may arise:

- If both experts agree on the modification, it is implemented.
- If both experts are from the same domain of expertise but they do not agree on the modification, both opinions are kept by the system. For instance, a same entity could be given two distinct definitions, each labelled by its expert.
- If both experts are from different disciplines and the concept clearly belongs to one of them, the system gives priority to the expert from the corresponding discipline. In this case, the opinion of the other expert will not be taken into account if there is a disagreement—but it will be kept in the repository.

- If both experts are from different disciplines and the concept belongs to both of them, both opinions are kept by the system, but they are clearly labelled by the corresponding disciplines.

A positive aspect to the *existence*—and detection—of conflicts between experts is that it could make them aware that their opinion may not be universally accepted. The automatic nature of the process could even spare them from the embarrassment of finding this out in discussion with other humans! This could lead them to revise some of their views, leading to a general enrichment of the field.

TOWARD AN INTELLIGENT TUTORING SYSTEM

One of our goals is to develop a multidisciplinary Intelligent Tutoring System (ITS) for the teaching of Quantum Information Processing. Experts are currently in high demand for lecturing and teaching QIP, and they are in short supply. An ITS would play an important role in promoting today this field of the future and in forming a new generation of students who would become aware of the potential impact of quantum mechanics on computer science. Here, “student” should be taken in the broad sense because many established researchers and professors in any of the underlying fields are expected to be interested in broadening their horizons.

The ITS is composed of the following modules: a curriculum, a learner's model, a pedagogic model, didactic resources and an interface. The Curriculum Database (see Figure 4) contains all the structured information that defines the curriculum, such as the graphs that define each of the concepts to be taught, as described previously. In addition, several other databases are required.

- The *Authors Profile Database* contains information on the experts who are allowed to enter and modify information in the system. This information consists of the names and passwords of the experts, their domains of expertise (computer science, mathematics, physics or chemistry) and it keeps track of the history of their actions.
- The *Dictionary Database* contains the vocabulary specific to the discipline.
- The *Resource Database* contains all the didactic resources relevant to teaching the concepts in the curriculum, such as simulations, demonstrations, exercises, problems, etc. Those resources can be in the form of text, HTML, video, sound, etc.
- The *Data Warehouse* (or *Repository*) is kept for subsequent data mining purposes. It records a history of all the activity that takes place in the system.

For the implementation of our curriculum, we have used Sun Microsystems' *Java Development Kit 1.2.2*. Java was chosen because it is an object-oriented language that is well-adapted for the development of interfaces. Even more importantly for the future, Java is entirely portable. Our

interfaces were created using *Simplicity Professional*, which includes a graphical editor capable of generating Java code. The advantage of *Simplicity* compared to *Java Builder* is that it requires less computing power and disk storage space. We have organized the curriculum within a Web-based distributed environment with a client-server architecture. For this purpose, we have used HTML and tools such as *Remote Method Invocation* (RMI).

This curriculum has been validated by five experts: three in quantum information processing and two in intelligent tutoring systems. The process during which the expertise was elicited from the experts lasted three months.

CONCLUSIONS AND FUTURE DIRECTIONS

We have designed a knowledge-based system for quantum information processing, which is an emerging multidisciplinary field (computer science, mathematics, physics, chemistry). For this purpose, we have designed a novel knowledge elicitation process, which is a hybrid between breadth-first and depth-first search, and we have given an explicit algorithm for it. Our elicitation process is currently under evaluation. Furthermore, we have developed a protocol for the resolution of conflicts between experts, which is necessary when experts from various disciplines wish to cooperate. Our approach is generic and could be applied *mutatis mutandis* to other multidisciplinary fields.

Quantum information processing is very rich and many concepts have not yet been included in our system, such as interaction-free measurements. We intend to enlist the help of additional experts in order to enrich our knowledge base. The curriculum validation is currently carried out for the most part by humans even though coherence aspects are verified automatically.

REFERENCES

1. Aïmeur, E., Frasson, C. and Dufort, H., “Co-operative learning strategies for intelligent tutoring systems”, *Applied Artificial Intelligence: An International Journal* **14**(5): 465–490, 2000.
2. Bennett, C.H., Brassard, G. and Ekert, A.K., “Quantum cryptography”, *Scientific American*, 164–171, October 1992.
3. Bennett, C.H., Brassard, G., Crépeau, C., Jozsa, R., Peres, A. and Wootters, W.K., “Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels”, *Physical Review Letters* **70**(13): 1895–1899, 1993.
4. Boose, J. and Bradshaw, J., “Expertise transfer and complex problems: Using Aquinas as a knowledge-acquisition workbench for knowledge-based systems”, *International Journal of Human Computers Studies* **51**(2): 453–478, 1999.
5. Brassard, G., *Quantum Information Processing for Computer Scientists*, MIT Press, to appear.

6. Brazier, F.M., Jonker, C.M. and Treur, J., "Principles of compositional multi-agent system development", in *Proceedings of the 15th IFIP World Computer Congress, WCC'98, Conference on Information Technology and Knowledge System* (J. Cuenca, editor), pages 347–360, 1998.
7. Buhrman, H., Cleve, R. and Wigderson, A., "Quantum vs. classical communication and computation", in *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 63–68, 1998.
8. Chandrasekaran, B., "Generic tasks in knowledge-based reasoning: High-level building blocks for expert systems design", *IEEE Expert*, 23–30, 1986.
9. Chuang, I.L. and Nielsen, M.A., *Quantum Computation and Quantum Information*, Cambridge University Press, 2000.
10. Hoog, R., Martil, M., Wielinga, B., Taylor, R., Bright, C. and Van de Velde, W., "The CommonKADS model set", Deliverable DM1.1x of ESPRIT Project P5248 "KASA-II", 1994.
11. McDermott, J., "Preliminary steps toward a taxonomy of problem solving", in *Automating Knowledge Acquisition for Expert Systems* (S. Marcus, editor), Boston, Kluwer Academic, 1988.
12. Shor, P.W., "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer", *SIAM Journal on Computing* **26**: 1484–1509, 1997.
13. Steels, L., "Components of expertise", *AI Magazine*, **11**(2): 29–49, 1990.
14. Wielinga, B., Shreiber, A. and Breuker, J., "KADS: A modelling approach to knowledge engineering", *Knowledge Acquisition*, 4, 1992.