A PROGRAMMING LANGUAGE FOR QUANTUM COMMUNICATION SYSTEM DESIGN

Nikolaos K. Papanikolaou
Department of Computer Science, University of Warwick

Key words to describe the work: Quantum Computation, Quantum Information, Quantum Cryptography, Programming Language Syntax & Semantics

Key Results: A specification language, *qSpec*, is proposed for the modelling and simulation of communication protocols involving quantum-mechanical phenomena.

How does the work advance the state-of-the-art?: The application of formal methods to the field of quantum information will help prevent design flaws in systems that implement quantum cryptography and other quantum communication schemes.

Motivation (problems addressed): Several languages for "quantum programming" have been proposed before, which facilitate the description of algorithms for quantum computers. None such language has been developed specifically for protocol design and validation. Protocols for quantum cryptography give a guarantee of immunity to eavesdropping, but proofs of their unconditional security are purely theoretical. The goal of this work is to establish a framework for formal definition and validation of quantum protocols.

Introduction and Background

It comes as no great surprise to computer scientists that the ongoing miniaturization of electronic circuits will inevitably lead to the one-bit-per-atom level by the year 2020. This is just one of the reasons for the growing interest in quantum computation today. It is now understood that quantum-mechanical principles can be applied in such ways that enable the solution of problems previously considered intractable. The ability to transcribe data on entangled sets of particles allows for massive parallelism, which in turn permits efficient computation of Fourier transforms, and even the inversion of public-key cryptosystems. Therefore, using quantum-mechanical effects for computational purposes is not just a trend, but a desideratum.

What is more, it has been shown that such effects can be used to securely exchange information. The foundation of *quantum cryptography* is none other than Heisenberg's Uncertainty Principle, which states that measuring conjugate variables of a quantum system leads to an uncertain result. In other words, measuring the position of a subatomic particle prevents an observer from obtaining that particle's velocity to any reasonable degree of accuracy. More interestingly, if each bit of a cryptographic key is mapped to the polarization of a photon, and the resulting photons are transmitted over an optical fibre, an enemy attempting to recover the bits by measuring the photons can never learn the complete key with certainty. The idea of using the Uncertainty Principle in this way is attributed to S. Wiesner **[8]**, and it was subsequently developed into a full-fledged key distribution protocol **[11]**.

The existence of statistical correlations between particles is another example of a phenomenon unique to the microscopic world. Such correlations are known to physicists as *entanglement*. If particles A and B are entangled, then measuring a property of A makes known the result of the same measurement on B, even though A and B may be physically distant from each other. Entanglement can be used for quantum cryptography as well: an enemy eavesdropper is forced to measure an entangled pair if he/she needs to obtain any useful information; this measurement inevitably disturbs the state of the pair, and the disturbance can be detected by the legitimate sender and receiver. The interested reader is referred to the standard literature on the subject for details **[3,4,5]**.

The development of the new field of quantum computation and information has included several proposals for cryptographic protocols, as well as protocols for general purpose communication. The common characteristic of these protocols is that their operation relies on a particular phenomenon inherent in nature, such as those described previously. A more recent tendency in the field involves the design of "quantum programming languages" **[9, 10, 12]**. These languages are intended for the description of quantum algorithms, such as Shor's factoring algorithm, or Grover's algorithm for efficient inverse search (see **[3,4,5])**. Apparently, no such language has been proposed for describing quantum communication schemes and, in particular, quantum crypto-

graphic protocols. It is the goal of the present work to fill precisely that gap.

General Considerations

A programming language is fully defined when its syntax and formal semantics have been stated. It is computer science tradition to present programming language syntax in Backus-Naur Form (BNF), while there are several well-known approaches to defining semantics. The *denotational* approach to semantics makes a direct correspondence between constructs in a programming language and the mathematical entities that they represent. Of the "quantum programming languages" that have been proposed to date, only Peter Selinger's **QPL [9]** possesses a denotational semantics.

The language we propose, termed *qSpec*, will be especially intended for *quantum* protocol design and validation. In developing such a language, it is instructive to investigate paradigms already in use today for modelling and testing classical (non-quantum) communication protocols. The most prominent language for this purpose is PROMELA, which is used in conjunction with the SPIN automated verification tool **[2,6]**. Notably, the Association for Computing Machinery (ACM) awarded the creator of SPIN, Gerard Holzmann, with the Systems Software Award in 2001. Also, Gavin Lowe's discovery of a flaw in the Needham-Schroeder PKCS protocol using the FDR software tool **[1,7]** is heralded as one of the greatest successes of the formal approach to protocol design.

Comments on the Syntax and Semantics of qSpec

The language PROMELA shares a common syntactic feature with the quantum language **qGCL** of [10]; both are examples of *guarded-command languages*. The PRISM language developed at the University of Birmingham has been successfully used to describe the protocol BB84 **[11]**, and PRISM is also a guarded-command language. Therefore, it is likely that the *qSpec* language proposed here will inherit this feature. Another potential influence on the syntax of *qSpec* originates from process calculi, such as Robin Milner's CCS; in order to specify *who* is performing a particular action in a protocol, the syntactic form *Agent.action(parameters)* is useful. Therefore, the measurement of a quantum bit by an agent *Alice* might be expressed as

```
Alice.measure(q)
```

where `q` is a variable representing the quantum bit, and `measure` represents the action of measuring, or *observing*, a quantum state.

By giving a denotational semantics for *qSpec*, it will be straightforward to define the meaning of basic data types; the denotation of a qubit can be given as a vector in a two-dimensional Hilbert space, which is the definition used in physics texts. Difficulties are likely to arise in giving semantics for entangled qubit pairs.

Conclusions and Future Work

Once *qSpec* has been fully defined and used for the unambiguous definition of quantum protocols, it should be possible to create an automated verification tool to enable the construction of proofs of correctness and security. Another interesting line of work will be to augment *qSpec* with a type system.

References

[1] Ryan, P., Schneider, S., Goldsmith, M., Lowe, G., and Roscoe, B. *Modelling and Analysis of Security Protocols.* Pearson Education, 2001.

[2] Holzmann, G. *The Design and Validation of Computer Protocols.* Prentice-Hall, 1991.

[3] Williams, C. and Clearwater, S., *Ultimate Zero and One.* Springer-Verlag, 2000.

[4] Gruska, J., *Quantum Computing.* McGraw-Hill International, 1999.

[5] Nielsen, M.A., and Chuang, I.L., *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[6] Holzmann, G., *The SPIN Model-Checker: Primer and Reference Manual*. Pearson Education, 2003.

[7] Lowe, G., *Proceedings of the 10th IEEE Computer Security Foundations Workshop*, 1997.

[8] Wiesner, S., *Conjugate Coding*, SIGACT News 15 (1983) 78-88.

[9] Selinger, P., *Towards a Quantum Programming Language*, To appear in *Mathematical Structures in Computer Science*, 2003.

[10] Sanders, J.W., and Zuliani, P., *Quantum Programming*, TR-5-99, Oxford University.

[11] Bennett, C. H., Brassard, G., Breidbart, S. and Wiesner, S., *Quantum cryptography, or unforgeable subway tokens*, Proceedings of Crypto '82 , August 1982, Plenum Press, pp. 267—275.

[12] Ömer, B., *A procedural formalism for quantum computing.* Department of Theoretical Physics, Technical University of Vienna, 1998.