A Framework for Automated Verification of Quantum Protocols

Nick Papanikolaou

nikos@dcs.warwick.ac.uk
http://www.warwick.ac.uk/go/nikos

*Joint work with Rajagopal Nagarajan (Warwick) and Simon Gay (Glasgow).
Intensive research over past 10-20 years on quantum computation and quantum information

- Chance of solving problems hitherto considered impossible

Recent upsurge of interest

- Implementation of practical quantum communication systems esp. quantum cryptography

Increasing need for **design, simulation, analysis** tools

Two levels of analysis:

- **High-level**: properties of systems with both quantum & classical components
- **Low-level**: properties of quantum subsystems, esp. quantum protocols and quantum algorithms
To develop a verification tool enabling analysis of quantum protocols at both levels.

We wish to facilitate automated reasoning about:

- **Quantum state**
- **Time**
- **Knowledge** of agents

**Approach:** *model-checking*
Quantum bits (qubits): superpositions of basis vectors, e.g.:

\[ |\psi\rangle = \alpha |0\rangle + \beta |1\rangle \]

Quantum state: a vector belonging to complex vector space (Hilbert space)

Continuous state space; countably infinite

\( n \)-qubit systems:

- dimension of state space grows exponentially: \( 2^n \) basis vectors.
- states are either:
  - decomposable (products of individual states)
  - or entangled (cannot be decomposed)
Transformations or operations on quantum states are *linear* and *reversible*.

\[ A |\psi\rangle = |\psi'\rangle \quad \text{where } A^{-1}A = I \text{ and } A = A^\dagger \]

Quantum operators or quantum gates are described by matrices.

Common gates:

- Controlled NOT (on 2 qubits) \( \text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \)
- Hadamard \( H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \)
- Pauli gates \( \sigma_0, \ldots, \sigma_3 \) (identity, bit flip, phase flip, bit and phase flip)
- Phase gate \( \Phi_{\theta} \) (rotation by \( \theta \))
The current state of any $n$-qubit system is unknown until it is measured.

Measurement is **destructive** and **probabilistic**.

- It collapses the current state to one of the $n$ basis vectors at random.

Measurement is the only way to extract a classical result from a quantum computation.

**Example:** Measuring a qubit

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

- with respect to basis \{\ket{0}, \ket{1}\} gives \ket{0} or \ket{1} at random.
- with respect to other basis \{\ket{a}, \ket{b}\} gives \ket{a} or \ket{b} at random.
Initial state of entangled pair shared by Alice and Bob:

$$|\psi\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$

1. To transmit integer $n$ (0 ≤ $n$ ≤ 3), Alice applies Pauli transformation $\sigma_n$ to her qubit $x$.
2. She physically transfers qubit $x$ to Bob.
3. Bob applies CNOT to qubits $x$ and $y$.
4. Bob applies Hadamard to $x$.
5. Bob measures $x$ and $y$. The result uniquely determines $n$. 
Simon Gay (Glasgow) and Rajagopal Nagarajan (Warwick) have developed a quantum process algebra, CQP, for modelling such protocols.

CQP has a **formal semantics** and a **type system**.

**Example:** modelling the dense coding protocol in CQP:

\[ Alice(x : Qbit, q : \hat{[}Qbit], n : 0..3) = \{ x \ast= \sigma_n \} \cdot q![x] \cdot 0 \]

\[ Bob(y : Qbit, q : \hat{[}Qbit]) = q?[x : Qbit] \cdot \{ x, y \ast= CNot \} \cdot \{ x \ast= H \} \cdot Use(\text{measure } x, y) \]

\[ System(x : Qbit, y : Qbit, n : 0..3) = (\text{new } q : \hat{[}Qbit])(Alice(x, q, n) | Bob(y, q)) \]
■ **PRISM: Probabilistic Model Checker (Kwiatkowska, Norman, Parker)**
  
  □ [http://www.cs.bham.ac.uk/~dxp/prism](http://www.cs.bham.ac.uk/~dxp/prism)
  □ Suitable for verifying properties of concurrent systems exhibiting probabilism

■ Quantum behaviour is inherently probabilistic

■ We have used PRISM to analyse some simple properties of a quantum cryptographic protocol, as well as dense coding, teleportation, and more
  
  □ not nearly as powerful as a general security proof
  □ can only model a handful of qubits and steps
Need to develop a general approach to:
- identify finite set of quantum states in a Hilbert space of dimension n, which is closed under the operations that arise in a protocol
- we did this manually for 2-3 qubits

It turns out that we can represent states of interest by Pauli operators; a closed group of operations (the Clifford group) transforms any one Pauli operator into another Pauli operator (viz. stabilizer formalism)
- The Clifford group operations are the ones which mostly arise in quantum protocols
- So we only need to represent a handful of operators and their effects on one another in order to simulate a whole class of quantum protocols.
Research on the foundations of quantum theory led to the development of quantum logic, which differs from classical propositional logic.

Some authors have developed quantum logics for reasoning about finite sets of qubits.

A tool for checking whether a protocol model satisfies a given formula would be highly desirable.

As is the case for classical security protocols, a tool which allows us to reason about:

- knowledge of agents in a quantum protocol
- quantum state at various times during the computation

is extremely valuable, and is likely to assist protocol designers and implementors.
Overall we have reported on work-in-progress, namely the design and implementation of a verification tool for quantum protocols.

- We covered the basics of QCQI.
- We looked at a simple quantum protocol.
- We reviewed CQP and the use of PRISM.
- We discussed some of the considerations entering into the design of a practical verification tool.