A process algebra for quantum processes

Laura Bussi

14/06/2019

UNIVERSITÀ DI PISA

Outline

Quantum information

Quantum information is the information of the state of a quantum system. As classical information is represented by using bits, quantum information is epitomized by qubits.
Bits and qubits fundamentally differ:

- a classical bit can be only either 0 or 1;
- a qubit can be in a state 0 or 1, or in a *coherent superposition* of both;
- coherence holds until the qubit is not measured: measurment of a qubit cause the immediate collapse of the qubit in one between the two states with a certain probability.

The Dirac notation

Standard representation of qubits relies on the Dirac notation, also known as bra-ket notation:

- as we can choose an orthonormal basis of the linear vector space, we can write a coherent superposition as $\alpha|0\rangle + \beta|1\rangle$.
- $|0\rangle$ and $|1\rangle$ are, respectevly, the two orthonormal vectors of the basis:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

- α and β can be in general complex numbers. Given the above state, we have that a measurment of that state will result in observing 0 with probability $|\alpha^2|$ and 1 with probability $|\beta^2|$. Obviously, it must hold $|\alpha^2| + |\beta^2| = 1$.

Pure and mixed states

As said, a qubit in a *pure state* is fully represented by a linear combination $\alpha|0\rangle + \beta|1\rangle$, i.e. a single ket vector in a Hilbert space.
However, due to its nature, quantum states are difficult to isolate and might be entangled with the enviroment. Then we have to consider the so called *mixed state*, i.e. a collection of pure states $|\psi_i\rangle$ each associated with a probability p_i.
Again, it must hold $0 \leq p_i \leq 1$ and $\sum_i p_i = 1$.
In order to handle quantum states which include mixed states, we introduce the *density operator* (or *density matrix*, denoted as ρ) representation:

$$\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$$

Operations on qubits and measurements

Quantum logic gates are used to implement operations on qubits. Operations are performed over qubits in a pure state and include creation and removal of superposition (*Hadamard operator*) and negation of a qubit w.r.t. the state of another qubit (*Controlled-NOT*). The latter is a primary component in building a quantum computer.

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Measurment of a qubit is an irreversible operation. In particular, a measurment of a qubit whose state is entangled with another system will transform the pure state of the qubit in a mixed state.

What about security?

Quantum computing is both a possibility and a threat:

- fundamental theorems of quantum information states that qubits cannot be copied nor broadcasted;
- the BB84 key distribution protocol, that we're going to se later on, exploits the fact that qubits cannot be measured without changing their state.
- On the other hand, the larger amount of information held by a single qubit compared to a classical bit causes a sensible speedup in solving traditionally "hard" problems, on which the security of many cryptographic protocols relies on.
- Thus, dissemination of quantum computers may result in having no longer secure protocols.

Towards a process algebra for quantum systems

As they involve quantum information, designing quantum protocols is surely error prone, even more than designing classical protocols. Having a formal language that can be used to design and verify quantum protocols plays a fundamental role in quantum security.
Process algebras are expressive models for concurrent computation in classical computing. In the following we'll see an extension of CCS for handling quantum processes, called qCCS.

Probabilistic Labelled Transition Systems

As seen, quantum measurments give rise to probability distributions, thus it is quite natural to model the evolution of a quantum process by using pLTSs.
We consider then a probabilty distribution over a set S:

$$\Delta : S \to [0,1] \qquad\qquad s.t. \sum_{s} \Delta(s) = 1$$

The set $\lceil \Delta \rceil = \{s \mid \Delta(s) > 0\}$ is called the support of Δ. We call $\overline{s}$ a *point distribution*, i.e. a distribution s.t. $\lceil \overline{s} \rceil = \{s\}$.
We denote as $Dist(S)$ the set of finite support distributions over S.
Then, a pLTS is a triple $\langle S, Act_\tau, \to \rangle$ where S is a set of states, Act_τ is a set of labels $Act \cup \{\tau\}$ and $\to \subseteq S \times Act_\tau \times Dist(S)$.

Lifting the transition relation

Since we want to allow distributions over states to perform an action, we have to lift the relations $\xrightarrow{\alpha}$.
Let $\mathcal{R} \subseteq S \times \mathit{Dist}(S)$ be a relation from states to distributions in a pLTS. Then $\mathcal{R}^\circ \subseteq \mathit{Dist}(S) \times \mathit{Dist}(S)$ is the smallest relation satisfying the following rules:

- $s \; \mathcal{R} \; \Delta \implies \overline{s} \; \mathcal{R}^\circ \; \Delta$
- $\forall i \in I.\ \Theta_i \; \mathcal{R} \; \Delta_i \implies (\sum_{i \in I} p_i \Theta_i) \; \mathcal{R}^\circ \; (\sum_{i \in I} p_i \Delta_i)$, for $p_i \in [0,1]$, $\sum_{i \in I} p_i = 1$, where I is a countable index set.

For a weak transition $\stackrel{\overline{\alpha}}{\Rightarrow}$ we write $\Delta \stackrel{\overline{\alpha}}{\Rightarrow} \Theta$ whenever $\Delta \stackrel{\overline{\tau}}{\Rightarrow} \xrightarrow{\alpha} \stackrel{\overline{\tau}}{\Rightarrow} \Theta$.

Syntax of qCCS

Three types of data are considered:

- *bool* for booleans;
- *real* for real numbers;
- *Qbt* for qubits.

As a consequence, we have two infinite sets of variables:

- *cVar* for classic variables;
- *qVar* for quantum variables.

We also assume a set *Exp* of expressions over real numbers and a set *BExp* of expressions over booleans.

Syntax of qCCS

Of course, we need two types of channels:

- *cChan* for classical channels, denoted as c, d...
- *qChan* for quantum channels, denoted as $\underline{c}$, $\underline{d}$...

A relabelling function f is a map on $cChan \cup qChan$ such that $f(cChan) \subseteq cChan$ and $f(qChan) \subseteq qChan$.
We write $\widetilde{q}$ for a sequence of distinct variables $q_1, ..., q_n$. The terms in qCCS are generated by the following grammar:

$$P, Q ::= nil \mid \tau.P \mid c?x.P \mid c!x.P \mid \underline{c}?x.P \mid \underline{c}!x.P \mid \mathcal{E}[\widetilde{q}].P$$

$$\mid M[\widetilde{q}; x].P \mid P + Q \mid P||Q \mid P[f] \mid P \setminus L \mid \mathit{if}\ \mathbf{b}\ \mathit{then}\ P \mid A(\widetilde{q}; \widetilde{x})$$

Free quantum variables

Free quantum variables are defined inductively:

$$qv(nil) = \emptyset \qquad qv(\tau.P) = qv(P)$$
$$qv(c?x.P) = qv(P) \qquad qv(c!e.P) = qv(P)$$
$$qv(\underline{c}?q.P) = qv(P) \setminus q \qquad qv(\underline{c}!q.P) = qv(P) \cup q$$
$$qv(\mathcal{E}[\widetilde{q}].P) = qv(P) \cup \widetilde{q} \qquad qv(M[\widetilde{q}; x].P) = qv(P) \cup \widetilde{q}$$
$$qv(P + Q) = qv(P) \cup qv(Q) \qquad qv(P||Q) = qv(P) \cup qv(Q)$$
$$qv(P[f]) = qv(P) \qquad qv(P \setminus L) = qv(P)$$
$$qv(\mathit{if}\ \mathbf{b}\ \mathit{then}\ P) = qv(P) \qquad qv(A(\widetilde{q}; \widetilde{x})) = \widetilde{q}$$

For a process to be legal we require that:

- $q \notin qv(P)$ in $\underline{c}!q.P$
- $qv(P) \cap qv(Q) = \emptyset$ in $(P||Q)$
- Each constant $A(\widetilde{p}; \widetilde{x})$ has a defining equation $A(\widetilde{p}; \widetilde{x}) = P$, where P is a term with $qv(P) \subseteq \widetilde{p} = \emptyset$ and $fv(P) \subseteq \widetilde{x}$

Informal semantics of qCCS

Since qCCS is an extension of value-passing CCS, classical operators behaves as expected.
As for the process $\underline{c}?q.P$, it receives a quantum datum over the quantum channel $\underline{c}$ and evolves into P, while $\underline{c}!q.P$ sends a quantum datum over the quantum channel $\underline{c}$ and evolves into P.
$\mathcal{E}$ represents a trace-preserving super-operator, i.e. a local quantum operation acting on a mixed state, applied to $\widetilde{q}$. The set of all the trace-preserving super-operators is denoted as TSO_H, for H being a (finite dimensional) Hilbert space.
The process $M[\overline{q}; x].P$ measures the states of the sequence $\overline{q}$ and stores the result into the classical variable x.

Operational semantics of qCCS

(Tau)
$$\langle \tau.P, \rho \rangle \xrightarrow{\tau} \langle P, \rho \rangle$$

(C-Outp)
$$\frac{v = [[e]]}{\langle c!e.P, \rho \rangle \xrightarrow{c!v} \langle P, \rho \rangle}$$

(Q-inp)
$$\frac{r \notin qv(\underline{c}?q.P)}{\langle \underline{c}?q.P, \rho \rangle \xrightarrow{\underline{c}?r} \langle P[r/q], \rho \rangle}$$

(Q-Com)
$$\frac{\langle P_1, \rho \rangle \xrightarrow{\underline{c}?r} \langle P_1', \rho \rangle \qquad \langle P_2, \rho \rangle \xrightarrow{\underline{c}!r} \langle P_2', \rho \rangle}{\langle P_1 \parallel P_2, \rho \rangle \xrightarrow{\tau} \langle P_1' \parallel P_2', \rho \rangle}$$

(Meas)
$$\frac{M = \sum_{i \in I} \lambda_i E^i \qquad p_i = tr(E^i_{\tilde{q}} \rho)}{\langle M[\tilde{q}; x].P, \rho \rangle \xrightarrow{\tau} \sum_{i \in I} p_i \langle P[\lambda_i/x], E^i_{\tilde{q}} \rho E^{i\dagger}_{\tilde{q}} / p_i \rangle}$$

(Int)
$$\frac{\langle P_1, \rho \rangle \xrightarrow{\alpha} \Delta \qquad qbv(\alpha) \cap qv(P_2) = \emptyset}{\langle P_1 \parallel P_2, \rho \rangle \xrightarrow{\alpha} \Delta \parallel P_2}$$

(Rel)
$$\frac{\langle P, \rho \rangle \xrightarrow{\alpha} \Delta}{\langle P[f], \rho \rangle \xrightarrow{f(\alpha)} \Delta[f]}$$

(Cho)
$$\frac{\langle P, \rho \rangle \xrightarrow{\alpha} \Delta \qquad [[b]] = \mathtt{true}}{\langle \mathbf{if}\ b\ \mathbf{then}\ P, \rho \rangle \xrightarrow{\alpha} \Delta}$$

(C-Inp)
$$\frac{v \in \mathtt{Real}}{\langle c?x.P, \rho \rangle \xrightarrow{c?v} \langle P[v/x], \rho \rangle}$$

(C-Com)
$$\frac{\langle P_1, \rho \rangle \xrightarrow{c?v} \langle P_1', \rho \rangle \qquad \langle P_2, \rho \rangle \xrightarrow{c!v} \langle P_2', \rho \rangle}{\langle P_1 \parallel P_2, \rho \rangle \xrightarrow{\tau} \langle P_1' \parallel P_2', \rho \rangle}$$

(Q-Outp)
$$\langle \underline{c}!q.P, \rho \rangle \xrightarrow{\underline{c}!q} \langle P, \rho \rangle$$

(Oper)
$$\langle \mathcal{E}[\tilde{q}].P, \rho \rangle \xrightarrow{\tau} \langle P, \mathcal{E}_{\tilde{q}}(\rho) \rangle$$

(Sum)
$$\frac{\langle P_1, \rho \rangle \xrightarrow{\alpha} \Delta}{\langle P_1 + P_2, \rho \rangle \xrightarrow{\alpha} \Delta}$$

(Res)
$$\frac{\langle P, \rho \rangle \xrightarrow{\alpha} \Delta \qquad cn(\alpha) \cap L = \emptyset}{\langle P \backslash L, \rho \rangle \xrightarrow{\alpha} \Delta \backslash L}$$

(Cons)
$$\frac{\langle P[\widetilde{v}/\widetilde{x}, \widetilde{r}/\widetilde{q}], \rho \rangle \xrightarrow{\alpha} \Delta \qquad A(\widetilde{x}, \widetilde{q}) := P}{\langle A(\widetilde{v}, \widetilde{r}), \rho \rangle \xrightarrow{\alpha} \Delta}$$

Equivalence for quantum processes

Given a process P and a density operator ρ, a configuration is a pair (P, ρ). In order to define equivalence between two processes, we define three criteria:

- *Barb-preservation* holds when two processes have the same probability to send out values on classical channels.
- *Reduction-closure* ensures that non-deterministic choices are in some sense preserved.
- *Compositionality* holds if $C \mathcal{R} D \implies (C||R) \mathcal{R} (D||R)$, with $qv(R) \cap (qv(C) \cup qv(D)) = \emptyset$ and $\mathcal{R}$ closed under trace-preserving operators.

A reduction barbed congruence ($\approx_r$) is the largest relation over configurations which is barb-preserving, reduction-closed and compositional. Furthermore, we have that $C \approx_r D \implies qv(C) = qv(D)$ and $env(C) = env(D)$.

Open bisimulation

Checking for equivalence using the above definition is quite difficult.
We define a relation (called *open bisimulation*) which can be used to characterize reduction barbed congruence.
A relation $\mathcal{R} \subseteq Con \times Con$ is an open bisimulation if

- $C \mathcal{R} D \implies qv(C) = qv(D), env(C) = env(D)$ and, for any trace-preserving operator $\mathcal{E}$ we have that whenever $\mathcal{E}(C) \xrightarrow{\alpha} \Delta$ there is some Θ with $\mathcal{E}(D) \stackrel{\overline{\alpha}}{\Rightarrow} \Theta$ and $\Delta \; \mathcal{R}^{\circ} \; \Theta$.
- Both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are open bisimulation.

We denote as $\approx_o$ the largest open bisimulation. Two quantum processes P and Q are bisimilar ($P \approx_o Q$) if, for any quantum state ρ and any indexed set of classical values $\widetilde{v}$ we have $\langle P\{\widetilde{v}/\widetilde{x}\}, \rho\rangle \approx_o \langle Q\{\widetilde{v}/\widetilde{x}\}, \rho\rangle$.

Ground bisimulation

In order to simplify the proof of bisimilarity, we can separate the issues to consider *TSOs* and transitions. We define then a new relation.
A relation $\mathcal{R} \subseteq Con \times Con$ is a *ground bisimulation* if

- $C \ \mathcal{R} \ D \implies qv(C) = qv(R), env(C) = env(R)$ and whenever $C \xrightarrow{\alpha} \Delta$ there is some distrubution Θ with $D \stackrel{\overline{\alpha}}{\Rightarrow} \Theta$ and $\Delta \ \mathcal{R}^{\circ} \ \Theta$.
- Both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are ground bisimulation.

If $\mathcal{R}$ is a ground bisimulation and it is closed under super-operators, then $\mathcal{R}$ is an open bisimulation. The relation $\approx_o$ is the largest ground bisimulation which is closed under *TSOs*.
Furhtermore, $\approx_o$ is preserved by all the constructor of qCCS except for summation.

BB84: informal description

BB84 was developed by Bennet and Brassard in 1984. It provides a provable secure way to create and share a private key between two parts (Alice and Bob in the following).
The basic protocol is defined as follows:

1. Alice randomly generates two strings of bits $\widetilde{B}_a$ and $\widetilde{K}_a$, each of size n.
2. Alice prepares a string $\widetilde{q}$ of size n and sends it to Bob.
3. Bob randomly generates a string of bits $\widetilde{B}_b$ of size n.
4. Bob measures $\widetilde{q}$ according to the basis determined by the string of bits he generated. Let the result be $\widetilde{K}_b$.
5. Bob sends his measurement choice $\widetilde{B}_b$ to Alice, while Alice sends $\widetilde{B}_a$ to Bob.
6. Alice and Bob determine which bits in $\widetilde{B}_a$ and $\widetilde{B}_b$ are equal and discard the bits in $\widetilde{K}_a$ and $\widetilde{K}_b$ where the corresponding bits of $\widetilde{B}_a$ and $\widetilde{B}_b$ differ.

BB84: formalisation

Using qCCS, we can formalise BB84 as follows:

$$Alice = Ran[\widetilde{q}; \widetilde{B}_a].Ran[\widetilde{q}; \widetilde{K}_a].Set_{\widetilde{K}_a}[\widetilde{q}].H_{\widetilde{B}_a}[\widetilde{q}].A2B!\widetilde{q}.WaitA(\widetilde{B}_a, \widetilde{K}_a)$$

$$Wait(\widetilde{B}_a, \widetilde{K}_a) = a2b?\widetilde{B}_b.a2b!\widetilde{B}_a.key_a!cmp(\widetilde{K}_a, \widetilde{B}_a, \widetilde{B}_b).nil$$

$$Bob = A2B?\widetilde{q}.Ran[\widetilde{q}'; \widetilde{B}_b].M_{\widetilde{B}_b}[\widetilde{q}; \widetilde{K}_b].b2a!\widetilde{B}_b.Wait(\widetilde{B}_b, \widetilde{K}_b)$$

$$Wait(\widetilde{B}_b, \widetilde{K}_b) = a2b?\widetilde{B}_a.key_b!cmp(\widetilde{K}_a, \widetilde{B}_a, \widetilde{B}_b).nil$$

$$BB84 = (Alice||Bob)\backslash\{a2b, b2a, A2B\}$$

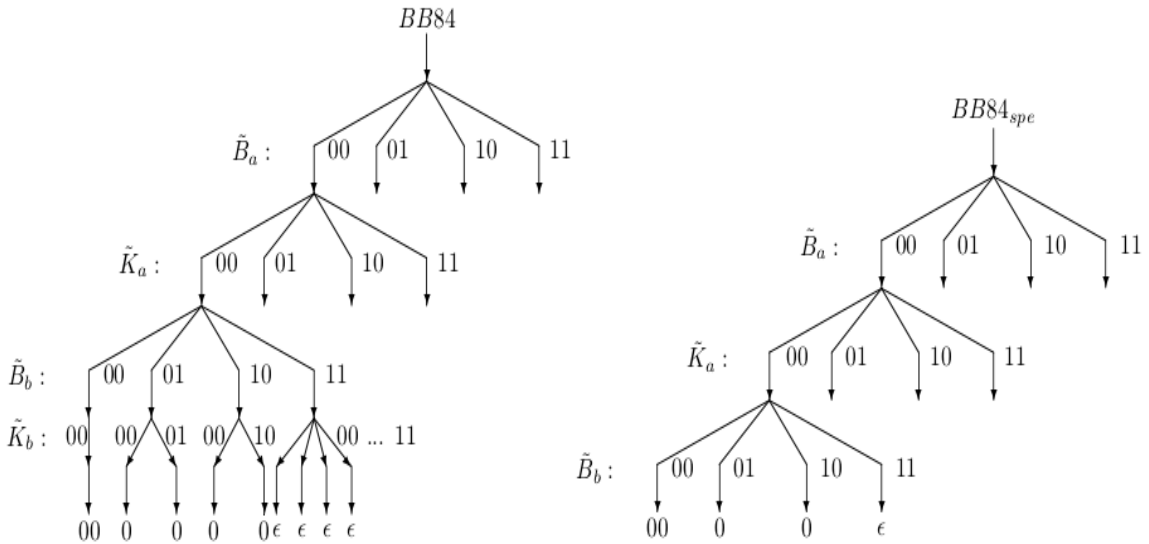
BB84: correctness

To show the correctness of BB84, we let:

$$BB84_{spe} = Ran[\widetilde{q}; \widetilde{B}_a].Ran[\widetilde{q}; \widetilde{K}_a].Ran[\widetilde{q}'; \widetilde{B}_b].$$

$$(key_a!cmp(\widetilde{K}_a, \widetilde{B}_a, \widetilde{B}_b).nil||key_b!cmp(\widetilde{K}_a, \widetilde{B}_a, \widetilde{B}_b).nil)$$

It can be checked from the pLTS that $BB84 \approx_o BB84_{spe}$, here for the case $n = 2$:



BB84 vs Eve

To test security of BB84, we can take into account an eavesdropper (named Eve) who tries a typical man-in-the-middle attack.

$$Alice' = (Alice||key_a?\widetilde{K}'_a.Pstr_{\widetilde{K}'_a}[\widetilde{q};\widetilde{x}].a2b!\widetilde{x}.a2b!SubStr(\widetilde{K}'_a,\widetilde{x}).b2a?\widetilde{K}''_b.$$

$$(if\ SubStr(\widetilde{K}'_a,\widetilde{x}) = \widetilde{K}''_b\ then\ key'_a!RemStr(\widetilde{K}'_a,\widetilde{x})\ else\ alarm_a!0.nil))\backslash\{key_a\}$$

$$Bob' = (Bob||key_b?\widetilde{K}'_b.a2b?\widetilde{x}.a2b?\widetilde{K}''_a.b2a!SubStr(\widetilde{K}'_b,\widetilde{x}).$$

$$(if\ SubStr(\widetilde{K}'_b,\widetilde{x}) = \widetilde{K}''_a\ then\ key'_b!RemStr(\widetilde{K}'_b,\widetilde{x})\ else\ alarm_b!0.nil))\backslash\{key_b\}$$

$$Eve = A2E?\widetilde{q}.Ran[\widetilde{q}'',\widetilde{B}_e].M_b[\widetilde{q};\widetilde{K}_e].Set_{\widetilde{K}_e}[\widetilde{q}].H_{\widetilde{B}_e}[\widetilde{q}].E2B!\widetilde{q}.key'_e!\widetilde{K}_e$$

Testing BB84 in a malicious environment

Now the system must include Eve (with $f_a(A2B) = A2E$ and $f_b(A2B) = E2B$):

$$BB84_E = (Alice'[f_a]||Eve||Bob'[f_b])\backslash\{a2b, b2a, A2E, E2B\}$$

The test is defined as follows:

$$TestBB84 = (BB84_E||key'_a?\widetilde{x}.key'_b?\widetilde{y}.key'_e?\widetilde{z}.$$

$$(if\ \widetilde{x} \neq \widetilde{y}\ then\ fail!0.nil\ else\ key_e!\widetilde{z}.skey!\widetilde{x}.nil))\backslash\{key'_a, key'_b, key'_e\}$$

Open bisimulation, again

Proving the security of BB84 is in general very hard.
Again, open bisimulation can help to simplify the test. Indeed we can define a simpler test:

$$TB = Ran[\widetilde{q}; \widetilde{B}_a].Ran[\widetilde{q}; \widetilde{K}_a].Ran[\widetilde{q}''; \widetilde{B}_e].Ran'_{\widetilde{B}_a,\widetilde{B}_e,\widetilde{K}_a}[\widetilde{q}; \widetilde{K}_e].Ran[\widetilde{q}'; \widetilde{B}_b].$$

$$Ran'_{\widetilde{B}_e,\widetilde{B}_b,\widetilde{K}_e}[\widetilde{q}; \widetilde{K}_b].Pstr_{\widetilde{K}_{ab}}[\widetilde{q}_a; \widetilde{x}].(\mathit{if}\ \widetilde{K}_{ab} = \widetilde{K}_{ab}\ \mathit{then}$$

$$key_e!\widetilde{K}_e.skey!RemStr(\widetilde{K}_{ab}, \widetilde{x})\ \mathit{else}$$

$$(\mathit{if}\ \widetilde{K}_{ab} \neq \widetilde{K}_{ab}\ \mathit{then}\ (alarm_a!0.nil||alarm_b!0.nil)\ \mathit{else}\ \mathit{fail}!0.nil))$$

We have that $TB \approx_o TestBB84$: proving correctness of TB suffices then to prove security of BB84 against Eve's attack.

Related works

QCCS is not the only available process algebra for quantum systems.

A first definition was provided by Marie Lalire and Philippe Jorrand in their 2004 paper, presenting QPAlg.
QPAlg, as qCCS, is inspired by CCS. It provides rules for applying unitary quantum operators, measurments and the ability to send and receive qubits. Equivalence between QPAlg processes is obtained defining *probabilistic branching bisimilarity*; however, such a bisimilarity is shown to be preserved by all operators except for parallel composition.

Another process algebra has been introduced in 2004 by Simon Gay and Rajagopal Nagarajan. Unlike QPAlg and qCCS, CQP is inspired by π-Calculus. The operational semantics is defined by using reductions, under the assumption that no external communication involving qubits happen. This limits name mobility to classical channels.

Conclusions and future works

Quantum computing is probably going to play a fundamental role in the future of security. Adapting reduction barbed congruence to quantum processes and separating super-operator applications from transitions results in having a quite simple technique to prove behavioural equivalence of processes, which can be used to prove security properties.
A simplified version of qCCS supports a semi-automated verification tool, introduced in 2016 by Feng et al.
The application of a sequential quantum programming language, as for instance QPL, to verify equations in such a tool is a possible future work.
More in general, an interesting evolution could be providing name mobility for quantum processes, taking the existing quantum process algebras as a starting point.

References

- https://en.wikipedia.org/wiki/Quantum_information
- Michael A. Nielsen, Isaac L. Chuang - Quantum Computing and Quantum Information
- Ittoop V. Puthoor - Theory and Applications of Quantum Process Calculus
- Simon Gay, Rajagopal Nagarajan - Communicating Quantum Processes
- Marie Lalire, Philippe Jorrand - A Process Algebraic Approach to Concurrent and Distributed Quantum Computation: Operational Semantics
- Yuxin Deng, Yuan Feng - Open Bisimulation for Quantum Processes