Integration of quantum processes in cyber-physical systems

Vitor Fernandes

High-Assurance Software Laboratory/INESC TEC
Universidade do Minho

Q Days
QuantaLab Workshop in Quantum Computation
April 13, 2019

Concurrency

- Emerged in the 19th century
  - railroads
  - telegraphic communications
- Popularized within computer science in the 60s by Dijkstra
- Nowadays is ubiquitous
  - web sites
  - mobile apps
- Concurrency:
  - is used to speed up processes
  - leads to notions of communication

Concurrency

- Emerged in the 19th century
  - railroads
  - telegraphic communications

- Popularized within computer science in the 60s by Dijkstra

- Nowadays is ubiquitous
  - web sites
  - mobile apps

- Concurrency:
  - is used to speed up processes
  - leads to notions of communication

Concurrency

- Emerged in the 19th century
  - railroads
  - telegraphic communications
- Popularized within computer science in the 60s by Dijkstra
- Nowadays is ubiquitous
  - web sites
  - mobile apps
- Concurrency:
  - is used to speed up processes
  - leads to notions of communication

Concurrency

- Emerged in the 19th century
  - railroads
  - telegraphic communications
- Popularized within computer science in the 60s by Dijkstra
- Nowadays is ubiquitous
  - web sites
  - mobile apps
- Concurrency:
  - is used to speed up processes
  - leads to notions of communication

Concurrency

Usual Approach

Division of a task into independently small ones, which communicate with each other and are performed in an interleaved way

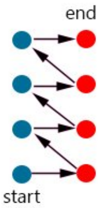


Figure: Example of a concurrent execution

Concurrency: Problem

Memory access is a famous problem, due to:

- mutual exclusion
- deadlock
- starvation

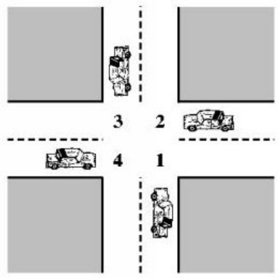


Figure: Cars on a crossing

Calculus of Communicating Systems

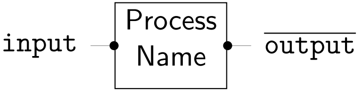
- One of the first algebras for reasoning about concurrency
- Developed by Robin Milner in 1989
- The central concept is that of a process, intuitively a black box

Calculus of Communicating Systems

- One of the first algebras for reasoning about concurrency
- Developed by Robin Milner in 1989
- The central concept is that of a process, intuitively a black box

Calculus of Communicating Systems

- One of the first algebras for reasoning about concurrency
- Developed by Robin Milner in 1989
- The central concept is that of a process, intuitively a black box



CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \stackrel{\mathit{def}}{=} \overline{\mathtt{coin}}.(\mathtt{tea}.\mathtt{P} + \mathtt{coffee}.\mathtt{P})$$

CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \overset{def}{=} \overline{\mathtt{coin}}.(\mathtt{tea.P}+\mathtt{coffee.P})$$

CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \overset{def}{=} \overline{\mathtt{coin}}.(\mathtt{tea.P} + \mathtt{coffee.P})$$

CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \stackrel{def}{=} \overline{\mathtt{coin}}.(\mathtt{tea}.\mathtt{P}+\mathtt{coffee}.\mathtt{P})$$

CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \stackrel{def}{=} \overline{\mathtt{coin}}.(\mathtt{tea}.\mathtt{P}+\mathtt{coffee}.\mathtt{P})$$

Syntax fragment

$\mathtt{P},\ \mathtt{Q} ::= \mathtt{K} \mid \alpha.\mathtt{P} \mid \sum_{i \in I} \mathtt{P}_i \mid \mathtt{P}|\mathtt{Q}$

CCS: Syntax

CCS main constructors:

- . (action prefixing)
- $+$ (choice operator; non-deterministic)
- $|$ (parallel composition)

Example of a CCS process:

$$\mathtt{P} \stackrel{def}{=} \overline{\mathtt{coin}}.(\mathtt{tea}.\mathtt{P}+\mathtt{coffee}.\mathtt{P})$$

Operational Semantics

Set of rules that determines the behavior of a process

Going quantum

The usual applications of process algebra are also important in quantum:

- quantum communication protocols
- concurrent quantum systems

Quantum Calculus of Communicating Systems

- Developed by Mingsheng Ying *et al* in 2008
- No classical information involved
- Semantics in terms of labeled transition systems
- Extends CCS with super-operators

Quantum Calculus of Communicating Systems

- Developed by Mingsheng Ying *et al* in 2008
- No classical information involved
- Semantics in terms of labeled transition systems
- Extends CCS with super-operators

Quantum Calculus of Communicating Systems

- Developed by Mingsheng Ying *et al* in 2008
- No classical information involved
- Semantics in terms of labeled transition systems
- Extends CCS with super-operators

Quantum Calculus of Communicating Systems

- Developed by Mingsheng Ying *et al* in 2008
- No classical information involved
- Semantics in terms of labeled transition systems
- Extends CCS with super-operators

qCCS: Underlying challenges

Restrictions from quantum mechanics:

- No cloning theorem
  - Solved by syntactic restrictions
- Measurement destroys superpositions

qCCS: Underlying challenges

Restrictions from quantum mechanics:

- No cloning theorem
  - Solved by syntactic restrictions
- Measurement destroys superpositions

qCCS: Underlying challenges

Restrictions from quantum mechanics:

- No cloning theorem
  - Solved by syntactic restrictions
- Measurement destroys superpositions

qCCS: Syntax

Syntax fragment

$\texttt{P,Q} ::= \texttt{A}(\tilde{x}) \mid \texttt{nil} \mid \varepsilon[X].\texttt{P} \mid c?x.\texttt{P} \mid c!x.\texttt{P} \mid \texttt{P}+\texttt{Q} \mid \texttt{P}||\texttt{Q}$

- $\texttt{A}(\tilde{x})$: constant processes
- $\varepsilon[X]$: action of a super-operator on a quantum state
- $c?x$: receive a quantum variable
- $c!x$: send a quantum variable

The operational semantics is based on a configuration.

Configuration

Pair formed by a process, P, and a state, ρ:

$$\langle P, \rho \rangle$$

qCCS: Syntax

Syntax fragment

$\texttt{P,Q} ::= \texttt{A}(\tilde{x}) \mid \texttt{nil} \mid \varepsilon[X].\texttt{P} \mid c?x.\texttt{P} \mid c!x.\texttt{P} \mid \texttt{P+Q} \mid \texttt{P}||\texttt{Q}$

- $\texttt{A}(\tilde{x})$: constant processes
- $\varepsilon[X]$: action of a super-operator on a quantum state
- $c?x$: receive a quantum variable
- $c!x$: send a quantum variable

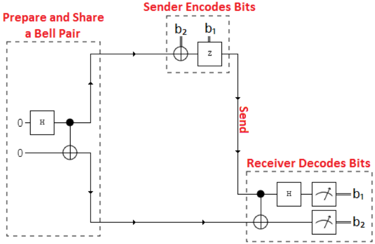
The operational semantics is based on a configuration.

Configuration

Pair formed by a process, P, and a state, ρ:

$$\langle P, \rho \rangle$$

qCCS: Superdense Coding

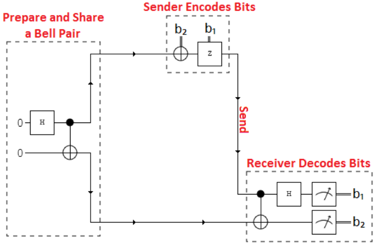


$$\langle SDC, \rho \rangle = \langle (A||B||C), \rho \rangle$$

$$\rho = |00\rangle$$

$$A = d?x_1.(I[x_1].c!x_1.nil + Z[x_1].c!x_1.nil + X[x_1].c!x_1.nil + Z[x_1].X[x_1].c!x_1.nil)$$
$$B = d?y_1.c?x_2.CNOT[x_2, y_1].H[x_2].M_{0,1}[x_2, y_1].nil$$
$$C = c?x_0.c?y_0.H[x_0].CNOT[x_0, y_0].d!x_0.d!y_0.nil$$

qCCS: Superdense Coding

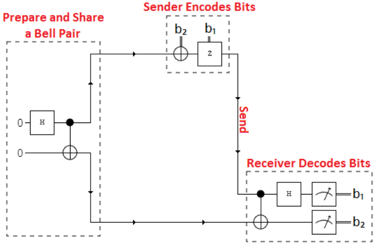


$$\langle SDC, \rho \rangle = \langle (A||B||C), \rho \rangle$$

$$\rho = |00\rangle$$

$$A = d?x_1.(I[x_1].c!x_1.nil + Z[x_1].c!x_1.nil + X[x_1].c!x_1.nil + Z[x_1].X[x_1].c!x_1.nil)$$
$$B = d?y_1.c?x_2.CNOT[x_2, y_1].H[x_2].M_{0,1}[x_2, y_1].nil$$
$$C = c?x_0.c?y_0.H[x_0].CNOT[x_0, y_0].d!x_0.d!y_0.nil$$

qCCS: Superdense Coding



$$\langle SDC, \rho \rangle = \langle (A||B||C), \rho \rangle$$

$$\rho = |00\rangle$$

$$A = d?x_1.(I[x_1].c!x_1.nil + Z[x_1].c!x_1.nil + X[x_1].c!x_1.nil + Z[x_1].X[x_1].c!x_1.nil)$$
$$B = d?y_1.c?x_2.CNOT[x_2, y_1].H[x_2].M_{0,1}[x_2, y_1].nil$$
$$C = c?x_0.c?y_0.H[x_0].CNOT[x_0, y_0].d!x_0.d!y_0.nil$$

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Importance of time

Classically, the notion of time is essential to concurrent systems.

- is the basis of several synchronization mechanisms
- allows to reason about real timed systems
- and communication protocols

All these aspects also apply to the quantum setting.
Moreover, the notion of time also plays a key role in:

- qubit coherence
- qram architecture in the cloud

Our goal is to introduce a temporal version of qCCS.

Examples

- Computer Science
  - Control the time of calculations to see if it exceeded the coherence time of a qubit.
- Physics
  - Photonic experiments, like quantum random number generator

Figure: Quantum Randon Number Generator

Examples

- Computer Science
  - Control the time of calculations to see if it exceeded the coherence time of a qubit.
- Physics
  - Photonic experiments, like quantum random number generator

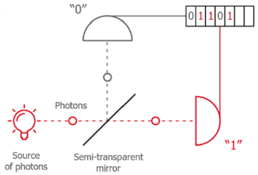


Figure: Quantum Randon Number Generator

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Choices, choices, choices...

Already in the classical setting several design choices arise:

- should actions be considered instantaneous?
- should time domain be discrete?
- should time interfere with non-determinism?
- should processes need to be synchronized to communicate?

In quantum, these questions need to be answered as well.

Ideas for timed qCCS

Some ideas are:

- temporal action that delays t units of time: $\sigma(t)$
- actions are considered to be performed instantaneously
- time domain is continuous
- time do not interfere with non-determinism
- processes need to be synchronized to communicate

Ideas for timed qCCS

Some ideas are:

- temporal action that delays t units of time: $\sigma(t)$
- actions are considered to be performed instantaneously
- time domain is continuous
- time do not interfere with non-determinism
- processes need to be synchronized to communicate

Ideas for timed qCCS

Some ideas are:

- temporal action that delays t units of time: $\sigma(t)$
- actions are considered to be performed instantaneously
- time domain is continuous
- time do not interfere with non-determinism
- processes need to be synchronized to communicate

Ideas for timed qCCS

Some ideas are:

- temporal action that delays t units of time: $\sigma(t)$
- actions are considered to be performed instantaneously
- time domain is continuous
- time do not interfere with non-determinism
- processes need to be synchronized to communicate

Ideas for timed qCCS

Some ideas are:

- temporal action that delays t units of time: $\sigma(t)$
- actions are considered to be performed instantaneously
- time domain is continuous
- time do not interfere with non-determinism
- processes need to be synchronized to communicate

To do

- Syntax of Timed qCCS
- Operational semantics
- Notions of behavioral equivalence
- Answer the question
  - Do quantum laws interfere in the integration of time?

Thank you for your attention