

Process algebras and network motifs 2

- Introduction
 - Goals
 - Methods
 - Motivations
- Review of π-calculus
 - Syntax
 - Structural Equivalence
 - Semantics
 - Stochastics

- Review of Kinetic Proofreading
 - Origins
 - Dynamics
 - Examples
 - Modeling in π-calculus
- Introduction of reflective calculi
 - Syntax
 - Structural Equivalence
 - Semantics
 - A New Approach to Stochastics
 - Modeling in a reflective calculus

May 04, 2005

Trento Seminar

π -calculus review – syntax

 $N \qquad ::= \qquad \sum_{i} \pi_{i} A_{i}^{\bullet} \qquad Normal \ processes$ $P,Q \qquad ::= N \mid P \mid Q \mid (\text{rec } K.F) \langle a \rangle^{\bullet} \mid K \langle a \rangle \mid (\upsilon x) P \quad Processes$

 $F ::= P | (\lambda x)F | (vx)F$ Abstractions

 $C ::= P \mid [x]C \mid (vx)C \qquad Concretions$

 $F \mid C$ Agents

 $\therefore = x \mid -x$ Synchronizers

a, b, c, x, y, z range over N, the set of names

a, ..., x range over vectors of names, i.e. over N*

Denote by N, P, C, 7, A the set of terms formed by the corresponding production

• We denote by 0 the zero-ary summation

 $\bullet |F| = |a|$ (the arity of F equals the length of a)

May 04, 2005

Trento Seminar

π -calculus review - free names

$$FN(\sum_{i} \pi_{i}.A_{i}) = \bigcup_{i} \{x_{i}\} \cup FN(A_{i}), \, \pi_{i} \in \{x_{i}, -x_{i}\}$$

$$FN(P|Q) = FN(P) \cup FN(Q)$$

$$FN((\text{rec } K.F)\langle a \rangle) = FN(F) \cup \{a\}$$

$$FN((\upsilon x)F) = FN(F)/\{x\} \quad FN((\upsilon x)C) = FN(C)/\{x\}$$

$$FN((\lambda x)F) = FN(F)/\{x\}$$

$$FN([x]C) = FN(C) \cup \{x\}$$

 $\{x\}$ denotes the set of names supporting the vector x

May 04, 2005

Trento Seminar

π -calculus review - structural equivalence

- **Defn.** Two agents, A, B, are α -equivalent, written $A \equiv_{\alpha} B$, iff they differ by the change of a bound name.
- Defn. Structural congruence, ≡, is the smallest congruence containing α-equivalence satisfying the following conditions:
 - § (N/=,0,+) is a commutative monoid
 - § $(\mathcal{P}/=,0,|)$ is a commutative monoid
 - § $(\upsilon x)0 = 0$, $(\upsilon x)(\upsilon y)A = (\upsilon y)(\upsilon x)A$
 - § $((\upsilon x)P)|Q = (\upsilon x)(P|Q)$, provided $x \notin FN(Q)$
 - § $(\operatorname{rec} K. (\lambda x)P)\langle a \rangle = P\{a/x\}\{(\operatorname{rec} K. (\lambda x)P)\langle b \rangle / K\langle b \rangle\}$
 - § $(\lambda x)(\upsilon y)F = (\upsilon y)(\lambda x)F$, provided $x \neq y$
 - § $[x](\upsilon y)C = (\upsilon y)[x]C$, provided $x \neq y$

May 04, 2005

Trento Seminar

π -calculus review - operational semantics

Notation: let
$$F = (\lambda x)P$$
, $C = (\upsilon z)[y]Q$, $|x| = |y|$, $x \cap z = \emptyset$ $F \cdot C \otimes (\upsilon z)(P|Q)\{y/x\}$

comm:
$$(...+x.F)|(-x.C+...) \rightarrow F \cdot C$$

$$par: \frac{P \to P'}{P|Q \to P'|Q}$$

$$res: \frac{P \to P'}{(\upsilon z)P \to (\upsilon z)P'}$$

equiv:
$$\frac{P7P', P' \rightarrow Q', Q'7Q}{P \rightarrow Q}$$

May 04, 2005

Trento Seminar

- 1. Is this theory closed in the same way arithmetic is closed?
- 2. What constitutes a theory of names?
- 3. What roles do names play in this theory?
- 4. Are there computations involved in fulfilling these roles?
- 5. Does the theory account for these computations?
- 6. Could these computations be replaced by others?
- 7. Is name-equality the only basis for synchronization?
- 8. How does this relate to physical interpretations of π -calculus?

May 04, 2005

Trento Seminar

 $x := x \mid -x$ Synchronizers

a, b, c, x, y, z range over N, the set of names

a, ..., x range over vectors of names, i.e. over N*

Denote by N, P, C, 7, A the set of terms formed by the corresponding production

• We denote by 0 the zero-ary summation

 $|\bullet|F| = |a|$ (the arity of F equals the length of a)

May 04, 2005

Trento Seminar

 $N ::= \sum_{i} \pi_{i} A_{i}^{\bullet}$ Normal processes

P,Q ::= $N \mid P \mid Q \mid (\text{rec } K.F) \langle a \rangle \circ \mid K \langle a \rangle \mid (vx)P$ Processes

 $F \qquad ::= \qquad P \mid (\lambda x)F \mid (\upsilon x)F \qquad Abstractions$

 $::= P \mid [x]C \mid (vx)C \qquad Concretions$

 $A ::= F \mid C$

N is left unspecified $x \mid -x$

a, b, c, x, y, z range over N, the set of names

a, ..., x range over vectors of names, i.e. over N*

Denote by N, P, C, 7, A the set of terms formed by the corresponding production

• We denote by 0 the zero-ary summation

 $|\bullet|F| = |a|$ (the arity of F equals the length of a)

May 04, 2005

Trento Seminar

$$FN(\sum_{i} \pi_{i}.A_{i}) = \bigcup_{i} \{x_{i}\} \cup FN(A_{i}), \, \pi_{i} \in \{x_{i}, -x_{i}\}$$

$$FN(P|Q) = FN(P) \cup FN(Q)$$

$$FN((\operatorname{rec} K.F)\langle a \rangle) = FN(F) \cup \{a\}$$

$$FN((\upsilon x)F) = FN(F)/\{x\} \quad FN((\upsilon x)C) = FN(C)/\{x\}$$

$$FN((\lambda x)F) = FN(F)/\{x\}$$

$$FN([x]C) = FN(C) \cup \{x\}$$

 $\{x\}$ denotes the set of names supporting the vector x

May 04, 2005

Trento Seminar

$$FN(\sum_{i} \pi_{i}.A_{i}) = \bigcup_{i} \{x_{i}\} \cup FN(A_{i}), \ \pi_{i} \in \{x_{i}, -x_{i}\}$$

$$FN(P|Q) = FN(P) \cup FN(Q)$$

$$FN((\text{rec } K.F)\langle a \rangle) = FN(F) \cup \{a\}$$

$$FN((\upsilon x)F) = FN(F)/\{x\} \quad FN((\upsilon x)C) = FN(C)/\{x\}$$

$$FN((\lambda x)F) = FN(F)/\{x\}$$

$$FN([x]C) = FN(C) \cup \{x\}$$

 $\{x\}$ denotes the set of names sup

Calculating free/bound names requires 'knowing' name equality

May 04, 2005

Trento Seminar

Notation: let F 7
$$(\lambda x)P$$
, C 7 $(\upsilon z)[y]Q$, $|x| = |y|, x \cap z = \emptyset$

$$F \cdot C \otimes (\upsilon z)(P|Q)\{y/x\}$$

comm:
$$(...+x.F)|(-x.C+...) \rightarrow F \cdot C$$

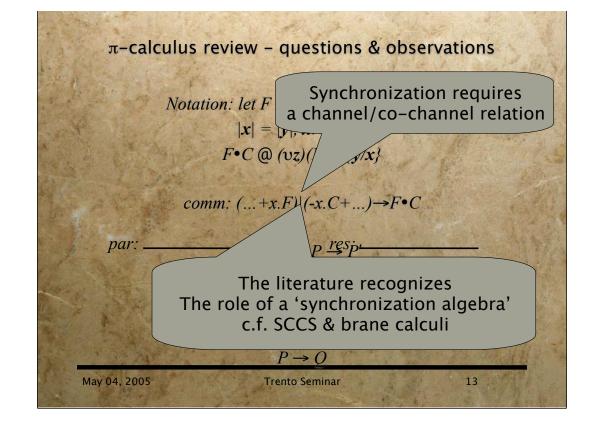
par:
$$P \rightarrow P' \quad P \stackrel{res}{\longrightarrow} P'$$

$$P|Q \rightarrow P' |Q \quad (\upsilon z)P \rightarrow (\upsilon z)P'$$

equiv:
$$\frac{P 7 P', P' \rightarrow Q', Q' 7 Q}{P \rightarrow Q}$$

May 04, 2005

Trento Seminar



π -calculus review - questions & observations Completeness Compositionality Concurrency Cost TM V × × V λ-calculus × × Petri Nets V V × CCS/CSP V V × Mobile process algebras $\overline{\mathbf{V}}$ $\overline{\mathbf{V}}$ $\overline{\mathbf{V}}$ $\overline{\mathbf{V}}$ May 04, 2005 Trento Seminar 14

- . The theory accomplishes this by (partially) elucidating the roles of names in computation
- 2. Anything that can conceivably and consistently play those roles can be used as the cornerstone for describing processes over them -- to some approximation -- by the π -calculus

	Name	Process
1.	Electrons	Small molecules
2.	Small molecules Proteins	
3.	Proteins	Cells
4.	Cells	Tissues
5.	Tcp/ip ports	Network protocols
6.	Urls	Web applications
7.	Mail addresses	Human e-communication
8.	Objids	Object-based applications

9.

3. Nothing in the theory elucidates how these different -- but inter-related -- phenomena are processes in one description

May 04, 2005

Trento Seminar

π -calculus review - stochastics

 $(x,r) \mid (-x,r)$

Synchronizers

r ranges over the real numbers

May 04, 2005

Trento Seminar

π -calculus review - stochastics

We use the interpretation of Phillips and Cardelli: rates are specified at a more abstract level and interpreted by a machine as pertaining to scheduling

comm:
$$(...+(x,r).F)|((-x,r).C+...) \rightarrow {}^rF \cdot C$$

$$par: \frac{P \rightarrow^r P'}{P|Q \rightarrow^r P'|Q}$$

$$par: \frac{P \to^r P'}{P|Q \to^r P'|Q} \qquad res: \frac{P \to^r P'}{(vz)P \to^r (vz)P'}$$

equiv:
$$P = P', P' \rightarrow^r Q', Q' = Q$$
$$P \rightarrow^r Q$$

May 04, 2005

Trento Seminar

π -calculus review - stochastics

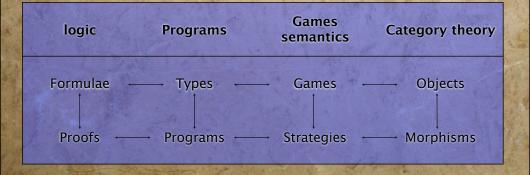
Critique the interpretation of Phillips and Cardelli:

- 1. Is this sufficient?
- 2. How could one develop a correctness criteria in which the only correct interpretation were a stochastic one?
- 3. Should rates be associated with channels or with actions? Critique the design choice that stochasticity arises only at synchronization:
- 1. Is this the only mathematically meaningful choice?
- 2. Is this a physically meaningful choice?

May 04, 2005 Trento Seminar

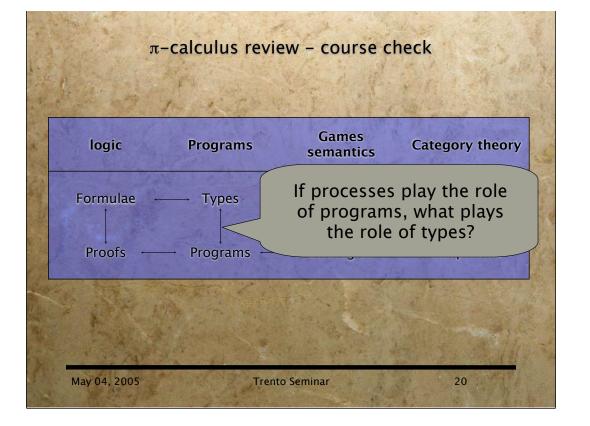
98

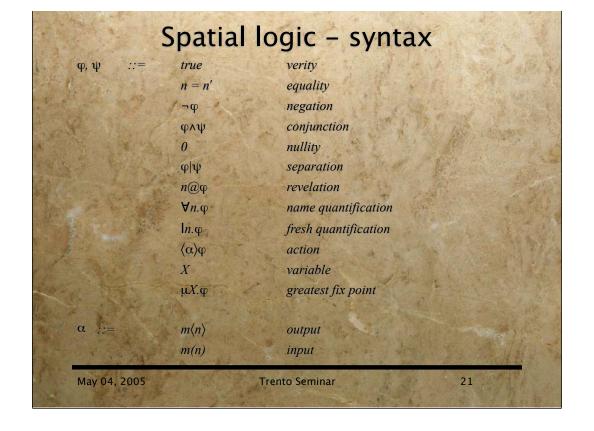
π -calculus review – course check



May 04, 2005

Trento Seminar





Spatial logic - semantics

Commitment relation

$$P \to Q \Rightarrow P \to^{\tau} Q$$

$$m, n \notin p \Rightarrow (\upsilon p)(m\langle n \rangle.Q + N|P) \to^{m\langle n \rangle}(\upsilon p)(Q|P)$$

$$m, n \notin p \Rightarrow (\upsilon p)(m(n).Q + N|P) \to^{m(n)}(\upsilon p)(Q|P)$$

$$P = P', P' \to^{\alpha} Q', Q' = Q \Rightarrow P \to^{\alpha} Q$$

Question: Do reductions have unique labels?

May 04, 2005

Trento Seminar

```
Spatial logic - semantics
        [true](v)
        [n = n'](v) =
                                         if n = n' then \mathcal{P} else \emptyset
        [-\varphi](v) = \mathcal{P}/[\varphi](v)
                                        [\varphi](v)\cap [\psi](v)
       [\varphi \wedge \psi](v) = [\varphi](v) \cap [\psi]
[\theta](v) = \{P : P = 0\}
                              = \{P: \exists Q, R. P = Q | R, Q \in [\varphi](v), R \in [\psi]\}
         \begin{array}{c} [\varphi|\psi](v) \\ (v) \end{array} \} 
        [n@\varphi](v)
                                         \{P: \exists Q.P \equiv (v \ n)Q, \ Q \in [\varphi](v)\}
        [\forall n.\varphi](v)
                                                \bigcap_{m} \left[ \varphi\{m/n\} \right] (v)
                                                 \bigcup_{m \in fn(\varphi,\nu)} ([\varphi\{m/n\}](\nu)/\{P : m \in fn(P)\})
        [\ln, \varphi](v)
        [\langle \alpha \rangle \varphi](v)
                                                 \{P: \exists Q.P \rightarrow \alpha Q, Q \in [\varphi]\}
          (v)}
        [X](v)
                                                 v(X)
        [\mu X.\varphi](v)
                                                 \bigcup \{S \subseteq \mathcal{P}: S \subseteq [\varphi](v/X \leftarrow S)\}
May 04, 2005
                                                Trento Seminar
                                                                                                     23
```

Biologically relevant examples?

Does SYSTEM reach a state where it makes no progress?

$$SYSTEM = \neg \diamond true$$

• Is there a state where input on site α is not possible?

$$SYSTEM = \Diamond \neg \langle \alpha \rangle true$$

Does it reach a state where the process is spatially divided into two distinct agents?

$$SYSTEM = \Diamond (\neg 0 \mid \neg 0)$$

May 04, 2005

Trento Seminar

Biologically relevant examples?

It depends on how we interpret SYSTEM ...

... SYSTEM=SignalingPathway|DrugAgent

 $SYSTEM = \neg \diamond true$

translates roughly 'does our drug cause the pathway to cease to function?'

or SYSTEM=SignalingPathway | ModifiedGene

 $SYSTEM = \Diamond \neg \langle \alpha \rangle true$

translates roughly 'does our gene modification cause the pathway to cease to block a certain protein-protein interaction?'

or SYSTEM=CellCycle

 $SYSTEM = \Diamond (\neg 0 \mid \neg 0)$

translates roughly 'does our cell divide?'

May 04, 2005

Trento Seminar

Course check

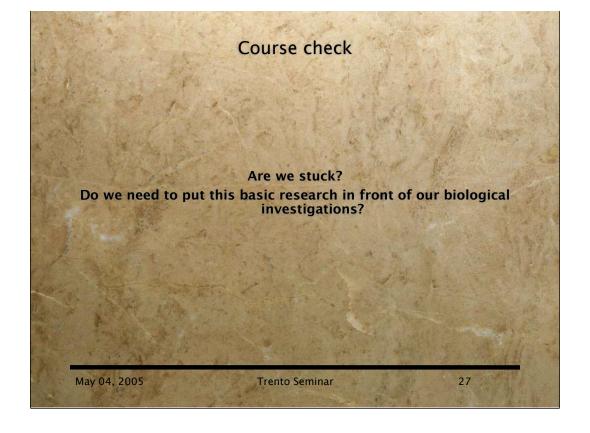
To be an instance of the proposition-as-types paradigm we need

- a proof object
 - This semantics does not offer one
 - It does however provide a model-checker and identify an interesting class of processes for which checking terminates
- A cut-elimination theorem
 - There are variants of spatial logic (and other proof systems for calculus) offering cut-elimination theorems
 - But they are with respect to the 'wrong' kind of cut, i.e. not correlated to parallel composition

To be useful in the biological setting we need a logic that is stochastic

- There are stochastic logics (and stochastic model-checkers)
 - None of them support mobility

May 04, 2005 Trento Seminar



Course check

My answer is 'no, we are not stuck'

- From the proposition—as—types point of view we still have the formula and models and the deep organizing principle of equality
 - We will need the proof-theoretic apparatus when we wish to calculate at the level of formulae

28

- The stochastic/mobility feature trade-off is more serious
 - How far can we get with only mobility?
 - How far can we get with only stochasticity?
 - How hard is it to introduce stochasticity into an HML?

May 04, 2005 Trento Seminar