Process Algebras & Network Motifs

section 2

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

π-calculus review - syntax

N	$::=$	$\sum_i \pi_i.A_i$ ❶	*Normal processes*
P,Q	$::=$	$N \mid P\|Q \mid (\text{rec } K.F)\langle \mathbf{a} \rangle$ ❷ $\mid K\langle \mathbf{a} \rangle \mid (\upsilon \mathbf{x})P$	*Processes*
F	$::=$	$P \mid (\lambda \mathbf{x})F \mid (\upsilon \mathbf{x})F$	*Abstractions*
C	$::=$	$P \mid [\mathbf{x}]C \mid (\upsilon \mathbf{x})C$	*Concretions*
A	$::=$	$F \mid C$	*Agents*
π	$::=$	$x \mid -x$	*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 $\mathcal{N}$, $\mathcal{P}$, $\mathcal{C}$, $\mathcal{F}$, $\mathcal{A}$ the set of terms formed by the corresponding production

❶ *We denote by 0 the zero-ary summation*

❷ $|F| = |\mathbf{a}|$ *(the arity of F equals the length of* $\mathbf{a}$*)*

π-calculus review – free names

$$FN(\sum_i \pi_i.A_i) = \bigcup_i \{x_i\} \cup FN(A_i),\ \pi_i \in \{x_i, \text{-}x_i\}$$

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

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

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

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

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

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

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**, $\equiv$, *is the smallest congruence containing α-equivalence satisfying the following conditions:*

- § $(\mathcal{N}/\equiv, 0, +)$ *is a commutative monoid*
- § $(\mathcal{P}/\equiv, 0, |)$ *is a commutative monoid*
- § $(\upsilon x)0 \equiv 0$, $(\upsilon x)(\upsilon y)A \equiv (\upsilon y)(\upsilon x)A$
- § $((\upsilon x)P)|Q \equiv (\upsilon x)(P|Q)$, *provided* $x \notin FN(Q)$
- § $(\mathrm{rec}\ K.\ (\lambda \mathbf{x})P)\langle \mathbf{a} \rangle \equiv P\{\mathbf{a}/\mathbf{x}\}\{(\mathrm{rec}\ K.\ (\lambda \mathbf{x})P)\langle \mathbf{b} \rangle / K\langle \mathbf{b} \rangle\}$
- § $(\lambda x)(\upsilon y)F \equiv (\upsilon y)(\lambda x)F$, *provided* $x \neq y$
- § $[x](\upsilon y)C \equiv (\upsilon y)[x]C$, *provided* $x \neq y$

π–calculus review – operational semantics

$$\textit{Notation: let } F \equiv (\lambda \mathbf{x})P,\ C \equiv (\upsilon \mathbf{z})[\mathbf{y}]Q,$$

$$|\mathbf{x}| = |\mathbf{y}|,\ \mathbf{x} \cap \mathbf{z} = \varnothing$$

$$F \bullet C \ @\ (\upsilon \mathbf{z})(P|Q)\{\mathbf{y}/\mathbf{x}\}$$

$$\textit{comm: } (\ldots + x.F)|(\text{-}x.C + \ldots) \rightarrow F \bullet C$$

$$\textit{par: } \frac{P \rightarrow P'}{P|Q \rightarrow P'|Q} \qquad \textit{res: } \frac{P \rightarrow P'}{(\upsilon \mathbf{z})P \rightarrow (\upsilon \mathbf{z})P'}$$

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

π-calculus review - questions & observations

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?

π-calculus review - questions & observations

N	$::=$	$\sum_i \pi_i.A_i$ ❶	*Normal processes*
P,Q	$::= N \mid P\|Q \mid (\mathrm{rec}\ K.F)\langle \boldsymbol{a} \rangle$ ❷ $\mid K\langle \boldsymbol{a} \rangle \mid (\upsilon \boldsymbol{x})P$		*Processes*
F	$::=$	$P \mid (\lambda \boldsymbol{x})F \mid (\upsilon \boldsymbol{x})F$	*Abstractions*
C	$::=$	$P \mid [\boldsymbol{x}]C \mid (\upsilon \boldsymbol{x})C$	*Concretions*
A	$::=$	$F \mid C$	*Agents*
π	$::=$	$x \mid -x$	*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 $\mathcal{N}$, $\mathcal{P}$, $\mathcal{C}$, $\mathcal{F}$, $\mathcal{A}$ the set of terms formed by the corresponding production

❶ *We denote by 0 the zero-ary summation*

❷ $|F| = |\boldsymbol{a}|$ *(the arity of F equals the length of **a**)*

π–calculus review – questions & observations

N	$::=$	$\sum_i \pi_i.A_i$❶	*Normal processes*
P,Q	$::=$	$N \mid P\vert Q \mid (\mathrm{rec}\ K.F)\langle \boldsymbol{a} \rangle$❷ $\mid K\langle \boldsymbol{a} \rangle \mid (\upsilon \boldsymbol{x})P$	*Processes*
F	$::=$	$P \mid (\lambda \boldsymbol{x})F \mid (\upsilon \boldsymbol{x})F$	*Abstractions*
C	$::=$	$P \mid [\boldsymbol{x}]C \mid (\upsilon \boldsymbol{x})C$	*Concretions*
A	$::=$	$F \mid C$	
π	$::=$	$x \mid -x$	

N is left unspecified

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

***a**, ..., **x** range over **<u>vectors</u>** of names, i.e. over* N^*

Denote by $\mathcal{N}$, $\mathcal{P}$, $\mathcal{C}$, $\mathcal{F}$, $\mathcal{A}$ the set of terms formed by the corresponding production

❶ *We denote by 0 the zero-ary summation*

❷ $|F| = |\boldsymbol{a}|$ *(the arity of F equals the length of **a**)*

π–calculus review – questions & observations

$$FN(\textstyle\sum_i \pi_i.A_i) = \bigcup_i \{x_i\} \cup FN(A_i),\ \pi_i \in \{x_i, \text{-}x_i\}$$

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

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

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

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

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

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

π–calculus review – questions & observations

$$FN(\textstyle\sum_i \pi_i.A_i) = \bigcup_i \{x_i\} \cup FN(A_i),\ \pi_i \in \{x_i, \text{-}x_i\}$$

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

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

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

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

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

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

Calculating free/bound names requires 'knowing' name equality

π-calculus review - questions & observations

$$\textit{Notation: let } F \equiv (\lambda \mathbf{x})P,\ C \equiv (\upsilon \mathbf{z})[\mathbf{y}]Q,$$

$$|\mathbf{x}| = |\mathbf{y}|,\ \mathbf{x} \cap \mathbf{z} = \emptyset$$

$$F \bullet C \ @\ (\upsilon \mathbf{z})(P|Q)\{\mathbf{y}/\mathbf{x}\}$$

$$\textit{comm: } (\ldots + x.F)|(-x.C + \ldots) \rightarrow F \bullet C$$

$$\textit{par: } \frac{P \rightarrow P'}{P|Q \rightarrow P'\,|Q} \qquad \textit{res: } \frac{P \rightarrow P'}{(\upsilon \mathbf{z})P \rightarrow (\upsilon \mathbf{z})P'}$$

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

π–calculus review – questions & observations

Notation: let F

$|\mathbf{x}| = |\mathbf{y}|, \ldots$

$F \bullet C \;@\; (\upsilon z)(\ldots \{\mathbf{y}/\mathbf{x}\}$

Synchronization requires
a channel/co-channel relation

comm: $(\ldots + x.F) \mid (\text{-}x.C + \ldots) \rightarrow F \bullet C$

par: $\qquad P \rightarrow P'$ $\quad$ *res:*

The literature recognizes
The role of a ‘synchronization algebra’
c.f. SCCS & brane calculi

$P \rightarrow Q$

π–calculus review – questions & observations

	Completeness	Compositionality	Concurrency	Cost
TM	☑	☒	☒	☑
λ-calculus	☑	☑	☒	☒
Petri Nets	☑	☒	☑	☑
CCS/CSP	☑	☑	☑	☒
Mobile process algebras	☑	☑	☑	☑

π–calculus review – questions & observations

1. 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

π-calculus review - stochastics

N	::=	$\sum_i \pi_i.A_i$	*Normal processes*
P,Q	::=	$N \mid P|Q \mid (\text{rec } K.F)\langle \mathbf{a} \rangle \mid K\langle \mathbf{a} \rangle \mid (\upsilon \mathbf{x})P$	*Processes*
F	::=	$P \mid (\lambda \mathbf{x})F \mid (\upsilon \mathbf{x})F$	*Abstractions*
C	::=	$P \mid [\mathbf{x}]C \mid (\upsilon \mathbf{x})C$	*Concretions*
A	::=	$F \mid C$	*Agents*
π	::=	$(x,r) \mid (-x,r)$	***Synchronizers***

r ranges over the real numbers

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

$$comm: (\ldots+(x,r).F)|((\text{-}x,r).C+\ldots) \to^r F \bullet C$$

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

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

$$equiv: \frac{P \equiv P',\, P' \to^r Q',\, Q' \equiv Q}{P \to^r Q}$$

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?

π-calculus review – course check

logic	Programs	Games semantics	Category theory
Formulae ⟷	Types ⟷	Games ⟷	Objects
↕	↕	↕	↕
Proofs ⟷	Programs ⟷	Strategies ⟷	Morphisms

π-calculus review - course check

logic	Programs	Games semantics	Category theory
Formulae ↔ Types			
↕	↕		
Proofs ↔ Programs			

If processes play the role of programs, what plays the role of types?

Spatial logic – syntax

φ, ψ	$::=$	$true$	*verity*
		$n = n'$	*equality*
		$\neg\varphi$	*negation*
		$\varphi \wedge \psi$	*conjunction*
		0	*nullity*
		$\varphi \mid \psi$	*separation*
		$n@\varphi$	*revelation*
		$\forall n.\varphi$	*name quantification*
		$\mathsf{I}n.\varphi$	*fresh quantification*
		$\langle\alpha\rangle\varphi$	*action*
		X	*variable*
		$\mu X.\varphi$	*greatest fix point*

α	$::=$	$m\langle n\rangle$	*output*
		$m(n)$	*input*

Spatial logic – semantics

Commitment relation

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

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

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

$P \equiv P',\ P' \rightarrow^{\alpha} Q',\ Q' \equiv Q \Rightarrow P \rightarrow^{\alpha} Q$

Question: Do reductions have unique labels?

Spatial logic – semantics

$$
\begin{array}{lcl}
[\mathit{true}](v) & = & \mathcal{P} \\
[n = n'](v) & = & \mathit{if}\ n = n'\ \mathit{then}\ \mathcal{P}\ \mathit{else}\ \emptyset \\
[\neg\varphi]\ (v) & = & \mathcal{P} / [\varphi]\ (v) \\
[\varphi\wedge\psi](v) & = & [\varphi](v)\cap[\psi](v) \\
[0](v) & = & \{P : P\equiv 0\} \\
[\varphi|\psi](v) & = & \{P : \exists Q,R.P\equiv Q|R,\ Q\in[\varphi](v),\ R\in[\psi](v)\ \} \\
[n@\varphi](v) & = & \{P : \exists Q.P\equiv(\upsilon\ n)Q,\ Q\in[\varphi](v)\} \\
[\forall n.\varphi](v) & = & \bigcap_{m} [\varphi\{m/n\}](v) \\
[\mathsf{I} n.\varphi](v) & = & \bigcup_{m\in fn(\varphi,v)}([\varphi\{m/n\}](v)/\{P : m\in fn(P)\}) \\
[\langle\alpha\rangle\varphi](v) & = & \{P : \exists Q.P\rightarrow^{\alpha}Q,\ Q\in[\varphi](v)\} \\
[X](v) & = & v(X) \\
[\mu X.\varphi](v) & = & \bigcup\{S\subseteq\mathcal{P}:S\subseteq[\varphi](v[X\leftarrow S])\}
\end{array}
$$

Biologically relevant examples?

- Does *SYSTEM* reach a state where it makes no progress?

$$SYSTEM \vDash \neg\Diamond true$$

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

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

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

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

Biologically relevant examples?

It depends on how we interpret *SYSTEM* …

… *SYSTEM=SignalingPathway|DrugAgent*

$$SYSTEM \vDash \neg \Diamond true$$

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

or *SYSTEM=SignalingPathway|ModifiedGene*

$$SYSTEM \vDash \Diamond \neg <\alpha> true$$

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

or *SYSTEM=CellCycle*

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

translates roughly 'does our cell divide?'

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

Course check

Are we stuck?

Do we need to put this basic research in front of our biological investigations?

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
- 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?