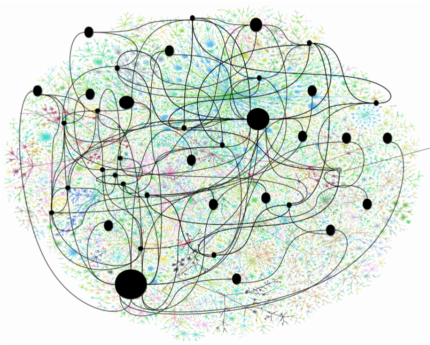
π calculus: toward global computing

Christian Williams

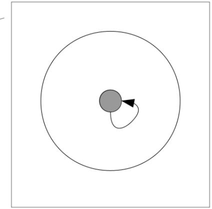
Applied Category Theory @ UC Riverside

February 12, 2019



What is the internet?

What is a computer?



π calculus   language of interaction

```
N:=  a, b, c, …                         Names

P:=  0             null                 Processes
     + P|P         parallel
     + a(x).P      input
     + āy          output

     āy|a(x).P => P{y/x}                Communication
```

ϙ̄k | ϙ(x).acs[x] => acs[k]

"… by understanding communicating systems, there is a sense of **completion**; for we can now begin to describe what goes on **outside** a computer in the same terms as what goes on **inside** ... we may say that we inhabit a **global computer**, an informatic world which demands to be understood just as fundamentally as physicists understand the material world."

— Robin Milner, creator

RChain: Secure. Sustainable. Scalable.

(transition)

relaxed intro.

designing communities: magic.

let's interact.

syntax

$N ::= \{a, b, c, \ldots\}$

$P ::=$	0	nul	— do nothing
$\mid$	$\bar{a}b.P$	out	— send b on a, then P
$\mid$	$a(x).P$	inp	— receive (x) on a, then P
$\mid$	$P \mid P$	par	— do in parallel
$\mid$	$\nu x\, P$	new	— create variable with scope P
$\mid$	$!P$	rep	— replicate P
$\mid$	$P + P$	sum	— do first xor second

syntax (in math)

$$N = \{a, b, c, \ldots\}$$

$$P \ni 0$$

$$\overline{} : N^2 \times P \longrightarrow P$$

$$(\,) : N^2 \times P \longrightarrow P$$

$$| : P^2 \longrightarrow P$$

$$\nu : N \times P \longrightarrow P$$

$$! : P \longrightarrow P$$

$$+ : P^2 \longrightarrow P$$

names: free and bound

$\mathrm{fn}(P) := $ free names of P

$$\mathrm{fn}(0) = \emptyset$$

$$\mathrm{fn}(\bar{a}b.P) = \{a,b\} \cup \mathrm{fn}(P)$$

$*$ $$\mathrm{fn}(a(x).P) = \{a\} \cup \mathrm{fn}(P) \setminus \{x\}$$

binding

(conventions)

$$\mathrm{fn}(P \mid Q) = \mathrm{fn}(P) \cup \mathrm{fn}(Q)$$

$*$ $$\mathrm{fn}(\nu x\, P) = \mathrm{fn}(P) \setminus \{x\}$$

$$\mathrm{fn}(!P) = \mathrm{fn}(P)$$

$$\mathrm{fn}(P+Q) = \mathrm{fn}(P) \cup \mathrm{fn}(Q)$$

communication

COMM: $\overline{a}z.P \mid a(x).Q$

$\Rightarrow \quad P \mid Q\{z/x\}$

example: nondeterminism

$$\begin{tikzcd} & \bar{a}y \mid a(x).\bar{x}v \mid \bar{a}z \arrow[dl] \arrow[dr] \arrow[d, phantom, "\text{nondeterminism}"] & \\ \bar{y}v \mid \bar{a}z & & \bar{a}y \mid \bar{z}v \end{tikzcd}$$

$$\bar{a}y.\bar{y}m \mid a(x).x(p).\bar{x}e \mid \bar{a}z.\bar{z}n$$

names are resources

example: restriction

$$\nu s(\overline{x}s.\overline{s}a.\overline{s}b \mid x(u).u(y).u(z).\overline{y}z)$$

$$\Rightarrow \nu s(\overline{s}a.\overline{s}b \mid s(y).s(z).\overline{y}z)$$

$$\Rightarrow \nu s(\overline{s}b \mid s(z).\overline{a}z)$$

$$\Rightarrow \nu s(\overline{a}b)$$

(passing a private channel)

congruence

SC-SUM-ASSOC	$M_1 + (M_2 + M_3)$	$\equiv$	$(M_1 + M_2) + M_3$
SC-SUM-COMM	$M_1 + M_2$	$\equiv$	$M_2 + M_1$
SC-SUM-INACT	$M + \mathbf{0}$	$\equiv$	M
SC-COMP-ASSOC	$P_1 \mid (P_2 \mid P_3)$	$\equiv$	$(P_1 \mid P_2) \mid P_3$
SC-COMP-COMM	$P_1 \mid P_2$	$\equiv$	$P_2 \mid P_1$
SC-COMP-INACT	$P \mid \mathbf{0}$	$\equiv$	P
SC-RES	$\nu z\, \nu w\, P$	$\equiv$	$\nu w\, \nu z\, P$
SC-RES-INACT	$\nu z\, \mathbf{0}$	$\equiv$	$\mathbf{0}$
SC-RES-COMP	$\nu z\, (P_1 \mid P_2)$	$\equiv$	$P_1 \mid \nu z\, P_2$, if $z \notin \mathsf{fn}(P_1)$
SC-REP	$!P$	$\equiv$	$P \mid !P$

Table 1.1. *The axioms of structural congruence*

REFL	$P = P$
SYMM	$P = Q$ implies $Q = P$
TRANS	$P = Q$ and $Q = R$ implies $P = R$
CONG	$P = Q$ implies $C[P] = C[Q]$

Table 1.2. *The rules of equational reasoning*

reduction

$$\text{R-INTER} \quad \frac{}{(\overline{x}y.\, P_1 + M_1) \mid (x(z).\, P_2 + M_2) \longrightarrow P_1 \mid P_2\{y/z\}}$$

$$\text{R-TAU} \quad \frac{}{\tau.\, P + M \longrightarrow P}$$

$$\text{R-PAR} \quad \frac{P_1 \longrightarrow P_1'}{P_1 \mid P_2 \longrightarrow P_1' \mid P_2} \qquad \text{R-RES} \quad \frac{P \longrightarrow P'}{\nu z\, P \longrightarrow \nu z\, P'}$$

$$\text{R-STRUCT} \quad \frac{P_1 \equiv P_2 \longrightarrow P_2' \equiv P_1'}{P_1 \longrightarrow P_1'}$$

Table 1.3. *The reduction rules*

example: replication

$$\nu s \quad s(v).!\bar{s}v \qquad ?$$

$$!\,\nu p \quad \bar{n}p.p(z).!\bar{p}z \qquad ?$$

(! ~ permanent, ν ~ local)

what about tuples?

$$x(y,z) \overset{\text{def}}{\underset{?}{=}} x(y).x(z)$$

$$\overline{x}\langle a,b\rangle \overset{\text{def}}{=} \overline{x}a.\overline{x}b$$

$$¿\quad x(y,z).\overline{y}z \mid \overline{x}\langle a,b\rangle \mid x(y,z).\overline{z}y \quad ?$$

polyadicity

$$\overline{a}\langle b_1, b_2 \cdots b_n \rangle \stackrel{\text{def}}{=} \nu w\ \overline{a}w.\overline{w}b_1.\cdots\overline{w}b_n$$

$$a(x_1, x_2 \cdots x_n) \stackrel{\text{def}}{=} a(w).w(x_1).\cdots w(x_n)$$

polyadic: send + receive lists of names

recursion

recursion

$A[x] \stackrel{def}{=} P$ $\quad fn(P) \subseteq \{x\}$, P contains A's

$\longrightarrow A[x] = !a(x).P\{\overline{a}y/A[y]\}$

$A[z] \in S$

$\longrightarrow \nu a(\overline{a}z \mid !a(x).P\{''\}) \in S$

(calling yourself)

example: update

$$S[v] = !(\bar{s}v + s(w).S[w])$$

$$= !s(v).\,!(\bar{s}v + s(w).\bar{s}w)$$

updateable storage

$$(P \mid S[i])$$

$$= (P \mid \nu s(\bar{s}i \mid !s(v).\,!(\bar{s}v + s(w).\bar{s}w)))$$

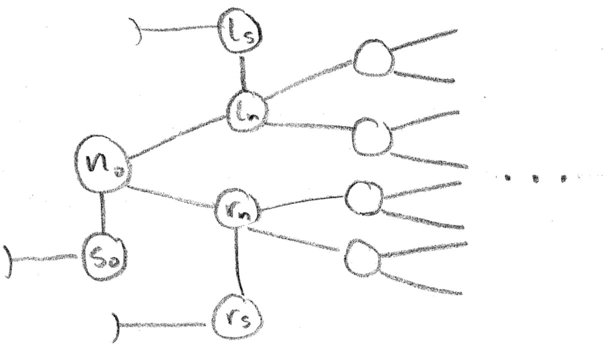
example: binary tree

$$
\begin{aligned}
&\mathrm{tree}[\mathit{node}, \mathit{store}] := \\
&\quad \mathbf{new}\ \mathit{left}_n, \mathit{left}_s, \mathit{right}_n, \mathit{right}_s \\
&\quad (\,!\,\overline{\mathit{node}}\langle \mathit{left}_n, \mathit{left}_s, \mathit{right}_n, \mathit{right}_s \rangle \\
&\quad \mid \mathit{store}(\mathit{value}).\ !\,\overline{\mathit{store}}\langle \mathit{value} \rangle \\
&\quad \mid \mathrm{Tree}[\mathit{left}_n, \mathit{left}_s] \mid \mathrm{Tree}[\mathit{right}_n, \mathit{right}_s]\,)
\end{aligned}
$$

(evolving system of cells)

internal

example: binary tree



$$T \equiv \overline{t}\langle n_0, s_0\rangle \mid !\,t(n,s).\,(\nu\, l_n, l_s, r_n, r_s)$$
$$!\,\overline{n}\langle l_n, l_s, r_n, r_s\rangle \mid s(v).\,!\,\overline{s}v \mid \overline{t}\langle l_n, l_s\rangle \mid \overline{t}\langle r_n, r_s\rangle)$$

example: HIGUL

a pair of processes cooperate
to build two unbounded chains of storage cells

$$P \overset{\text{def}}{=} (\nu n,s)(H \mid I \mid G \mid U \mid L)$$

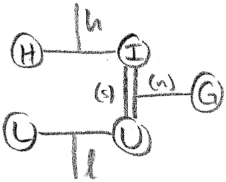
$$H \overset{\text{def}}{=} h(z).\,!\bar{h}z$$

$$L \overset{\text{def}}{=} \ell(z).\,!\bar{\ell}z$$

$$G \overset{\text{def}}{=} !\nu p\ \bar{n}p.p(z).\,!\bar{p}z$$

$$I \overset{\text{def}}{=} \nu i\,(\bar{i}h \mid !i(z).n(a).\bar{z}a.\bar{s}a.s(c).\bar{i}c)$$

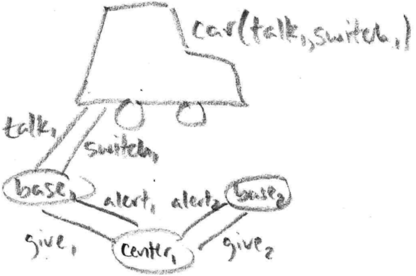
$$U \overset{\text{def}}{=} \nu u\,(\bar{u}\ell \mid !u(z).n(b).\bar{z}b.\ n(c).\bar{z}c.s(a).\bar{s}c.\bar{u}a)$$



dynamic topology

<u>mobility</u>

- $car[talk, switch]$
  $\stackrel{def}{=} talk.car[talk, switch]$
  $+\ switch(t', s').car[t', s']$
- $base[t, s, g, a]$
  $\stackrel{def}{=} t.base[t, s, g, a]$
  $+\ g(t', s').\bar{s}\langle t', s'\rangle.idlebase(t, s, g, a)$
- $idlebase[t, s, g, a] \stackrel{def}{=} a.base(t, s, g, a)$
- $center_1 \stackrel{def}{=} \overline{give_1}\langle talk_2, switch_2\rangle.alert_2.center_2$
  $center_2 \stackrel{def}{=} \overline{give_2}\langle talk_1, switch_1\rangle.alert_1.center_1$



dynamic topology

$$\mathit{system}_1 \equiv (\nu \vec{c})(\mathit{car}(\mathit{talk}_1, \mathit{switch}_1) \mid \mathit{base}_1 \mid \mathit{idle}_2 \mid \mathit{center}_1)$$

$$\Rightarrow (\nu \vec{c})(\mathit{car}(\mathit{talk}_1, \mathit{switch}_1) \mid \overline{\mathit{switch}_1}\langle \mathit{talk}_2, \mathit{switch}_2 \rangle.\mathit{idle}_1 \mid \mathit{idle}_2 \mid \mathit{alert}_2.\mathit{center}_2)$$

$$\Rightarrow (\nu \vec{c})(\mathit{car}(\mathit{talk}_2, \mathit{switch}_2) \mid \mathit{idle}_1 \mid \mathit{idle}_2 \mid \mathit{alert}_2.\mathit{center}_2)$$

$$\Rightarrow (\nu \vec{c})(\mathit{car}(\mathit{talk}_2, \mathit{switch}_2) \mid \mathit{idle}_1 \mid \mathit{base}_2 \mid \mathit{center}_2) \equiv \mathit{system}_2$$

categories

$$\bar{a}b.P \mid a(x).Q \Rightarrow P \mid Q\{b/x\}$$

$$\begin{tikzcd}
 & (N^2P)^2 \arrow[r] \arrow[dr, phantom, "\Downarrow \mathrm{comm}"] & P^2 \arrow[dr] & \\
N^3P^2 \arrow[ur] \arrow[drr] & & & P \\
 & & P^2 \arrow[ur] &
\end{tikzcd}$$

?

enriched algebraic theories

— with binding

reflection

reflection

code $\Longleftrightarrow$ data

$$p(n) = \overset{\text{nul}}{1} + \overset{\text{out}}{n^2 p(n)} + \overset{\text{inp}}{n^2 p(n)} + \overset{\text{par}}{p(n)^2}$$

$$\underset{\text{new}}{n\,p(n)} + \underset{\text{rep}}{p(n)} + \underset{\text{sum}}{p(n)^2}$$

$p(r) = r$ (least fixed point — "Y combinator")

ρ calculus

<u>rho</u>

$P ::= 0 \mid a?(\vec{x}).P \mid a!(P) \mid P|P \mid {}^{*}a$

evaluate

$N ::= @P$ reference

COMM: $a!(P) \mid a?(x).Q \Rightarrow Q\{@P/x\}$

application: blockchain

application: biology [1]

Biology	Process calculi
Entity	Process
Interaction capability	Channel
Interaction	Communication
Modification/evolution	State change

Table 1: Process calculi abstraction for systems biology

application: biology [2]

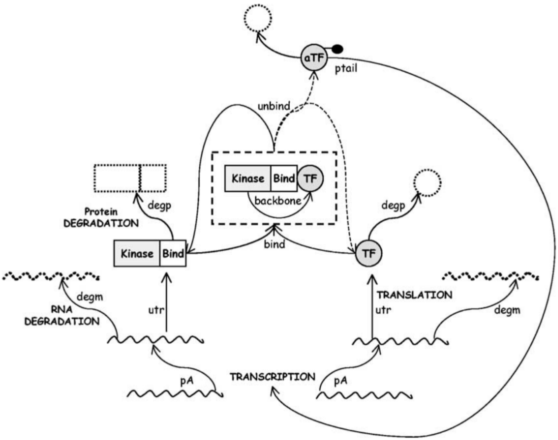


Fig. 1. A simple biomolecular process: Transcriptional regulation by positive feedback.

application: biology [2]

Table 2
Specification of a biomolecular system

$$\mathrm{Sys} = \mathrm{Gene\_A}|\mathrm{Gene\_TF}|\mathrm{Transcr}|\mathrm{Transl}|\mathrm{RNA\_Deg}|\mathrm{Protein\_Deg}$$

$$\mathrm{Gene\_A} = (\mathrm{basal}(), 4).(\mathrm{Gene\_A}|\mathrm{RNA\_A}) + (\mathrm{pA}(), 40).(\mathrm{Gene\_A}|\mathrm{RNA\_A})$$

$$\mathrm{RNA\_A} = (\mathrm{utr}(), 1).(\mathrm{RNA\_A}|\mathrm{Protein\_A}) + (\mathrm{degm}(), 1)$$

$$\mathrm{Protein\_A} = (\nu \mathrm{bb1}, \mathrm{bb2}, \mathrm{bb3})(\mathrm{Binding\_Site}|\mathrm{Kinase})$$

$$\mathrm{Binding\_Site} = (\overline{\mathrm{bind}}\langle \mathrm{bb1}, \mathrm{bb2}, \mathrm{bb3}\rangle, 0.1).\mathrm{Bound\_Site} + (\mathrm{degp}(), 0.1).(\overline{\mathrm{bb3}}, \infty)$$

$$\mathrm{Bound\_Site} = (\overline{\mathrm{bb1}}, 10).\mathrm{Binding\_Site} + (\mathrm{degp}(), 0.1).(\overline{\mathrm{bb3}}, \infty).(\overline{\mathrm{bb3}}, \infty)$$

$$\mathrm{Kinase} = (\overline{\mathrm{bb2}}\langle \mathrm{ptail}\rangle, 10).\mathrm{Kinase} + (\mathrm{bb3}(), \infty)$$

$$\mathrm{Gene\_TF} = (\mathrm{basal}(), 4).(\mathrm{Gene\_TF}|\mathrm{RNA\_TF}) + (\mathrm{pA}(), 40).(\mathrm{Gene\_TF}|\mathrm{RNA\_TF})$$

$$\mathrm{RNA\_TF} = (\mathrm{utr}(), 1).(\mathrm{RNA\_TF}|\mathrm{Protein\_TF}) + (\mathrm{degm}(), 1)$$

$$\mathrm{Protein\_TF} = (\mathrm{bind}(\mathrm{c\_bb1}, \mathrm{c\_bb2}, \mathrm{c\_bb3}), 0.1).\mathrm{Bound\_TF} + (\mathrm{degp}(), 0.1)$$

$$\begin{aligned}\mathrm{Bound\_TF} = {} & (\mathrm{c\_bb1}(), 10).\mathrm{Protein\_TF} + (\mathrm{c\_bb3}(), \infty) + \\ & (\mathrm{c\_bb2}(\mathrm{tail}), 10).((\mathrm{c\_bb1}(), 10).\mathrm{Active\_TF}(\mathrm{tail}) + (\mathrm{c\_bb3}(), \infty))\end{aligned}$$

$$\mathrm{Active\_TF}(\mathrm{tail}) = (\overline{\mathrm{tail}}, 100).\mathrm{Active\_TF}(\mathrm{tail}) + (\mathrm{degp}(), 0.1)$$

$$\mathrm{Transcr} = (\overline{\mathrm{basal}}, 4).\mathrm{Transcr} + (\mathrm{ptail}(), 100).(\overline{\mathrm{pA}}, 40).\mathrm{Transcr}$$

$$\mathrm{Transl} = (\overline{\mathrm{utr}}, 1).\mathrm{Transl}$$

$$\mathrm{RNA\_Deg} = (\overline{\mathrm{degm}}, 1).\mathrm{RNA\_Deg}$$

$$\mathrm{Protein\_Deg} = (\overline{\mathrm{degp}}, 0.1).\mathrm{Protein\_Deg}$$

mathematics

linear logic [3]
game semantics [4]
double categories [5]
bigraphs [6]
operational semantics
categorical semantics

references

1. ''Process Calculi Abstractions for Biology''
Guerriero et. al, 2006 CCSB
2. ''Application of a stochastic name-passing calculus to representation and simulation of molecular processes''
Regev et. al, 2001 IPL
3. ''Propositions as Sessions''
Wadler, 2012 ACM
4. ''Pi Calculus, Dialogue Games and PCF''
Hyland and Ong, 1995 ACM
5. ''Cartesian Closed Double Categories, their Lambda-Notation, and the Pi-Calculus''
Bruni and Montanari, 1999 LICS
6. ''Bigraphs and Their Algebra''
Milner, 2008 ENTCS