Languages & Notations for Systems Biology

Luca Cardelli

Microsoft Research
Cambridge UK

2004-09-15 Unconventional Programming Paradigms '04
Le Mont St Michel

www.luca.demon.co.uk
Aims

Modeling biological systems.
   Helping out in Systems Biology.

By adapting paradigms and techniques developed for modeling information-processing systems.

Because they have some similar features:

   Deep layering of abstractions.
   Complex composition of simpler components.
   Discrete (non-linear) evolution.
   Digital coding of information.
   Reactive information-driven behavior.
   Very high degree of concurrency.
   "Emergent behavior" (not obvious from part list).
**Methods**

*Model Construction (writing things down precisely)*

Studying the notations used in systems biology.
Formulating process calculi, for various purposes.
Studying their dynamics (semantics).

*Model Validation (using models for postdiction and prediction)*

**Stochastic Simulation**

Stochastic = Quantitative concurrent semantics.
Based on compositional descriptions.

**“Program” Analysis**

Control flow analysis
Causality analysis

**Modelchecking**

Standard, Quantitative, Probabilistic
Storing Processes

• Today we represent, store, and analyze:
  - Gene sequence data
  - Protein structure data
  - Metabolic network data
  - Compartmentalized reaction data (SBML)
  - ...

• How can we represent, store, and analyze biological processes?
  - Scalable, precise, dynamic, highly structured, maintainable representations for systems biology.
  - Not just huge lists of chemical reactions or differential equations.
Structural Architecture

Eukaryotic Cell
(10~100 trillion in human body)

Membranes everywhere

- Nuclear membrane
- Mitochondria
- Golgi
- Vesicles
- E.R.
- Plasma membrane (<10% of all membranes)
Functional Architecture

The Abstract Machines of Biochemistry

- Biochemical Networks - The Protein Machine
- Gene Regulatory Networks - The Gene Machine
- Transport Networks - The Membrane Machine

Systems Biology

1. “We (kind of) understand the components; but how does the system work?”
2. “Use high-throughput experiments to gather system data.”

- Different chemical toolkits
- Different instruction sets
- Different programming models
- Different notations

Gene Machine

Nucleotides

Regulation

Holds genome(s), confines regulators

Model Integration
Different time and space scales

Protein Machine

Amino acids

Metabolism, Propulsion
Signal Processing
Molecular Transport

Membrane Machine

Phospholipids

Confinement
Storage
Bulk Transport

Signals conditions and events

Makes proteins, where/when/how much

Directs membrane construction
and protein embedding

Holds receptors, actuators
hosts reactions

Implements fusion, fission
1. The Protein Machine

• Complex folded-up shapes that:
  – Fit together, dock, undock.
  – Excite/unexcite, warp each other.
  – Bring together, catalyze, transform materials.
  – Form complex aggregates and networks.

• Mapping out such networks:
  – In principle, it’s “just” a very large set of chemical equations.
  – Notations have been developed to summarize and abstract.

An actual molecular interaction network.
(Nodes are distinct protein kinds, arcs mean that two kinds of proteins interact.)
Protein Structure

Primary

The 20 Aminoacids

Secondary

Alpha Helix, Beta Sheet

Tertiary

Green Fluorescent Protein

Quaternary

Triose Phosphate Isomerase

http://www.cmbi.kun.nl/gvteach/bioinformatica1/
Some Allosteric Switches

Allosteric ("other shape") reactions modify accessibility.

Kinase
= donates phosphate P
= phosphorilates other proteins

Phosphatase
= accepts phosphate P
= dephosphorilates other proteins

Logical AND
at equal concentrations of the individual input stimuli, activation is much higher if both stimuli are present

"Phosphatase Kinase Kinase" = a kinase that activates a kinase that activates a phosphatase that deactivates a protein.

Humans have the same number of modular protein domains (building blocks) as worms, but twice the number of multi-domain proteins.
**MIM: Molecular Interaction Maps (Kohn)**

The double-arrowed line indicates that proteins A and B can bind to each other. The "node" placed on the line represents the A:B complex.

Asymmetric binding where protein A donates a peptide that binds to a receptor site or pocket on protein B.

Representation of multimolecular complexes: $x$ is $A:B$; $y$ is $(A:B):C$. This notation is extensible to any number of components in a complex.

Covaent modification of protein A. The single-arrowed line indicates that A can exist in a phosphorylated state. The node represents the phosphorylated species.

Cleavage of a covalent bond: dephosphorylation of A by a phosphatase.

Proteolytic cleavage at a specific site within a protein.

Stoichiometric conversion of A into B.

Transport of A from cytosol to nucleus. The node represents A after it has been transported into the nucleus.

Formation of a homodimer. Filled circle on the right represents another copy of A. The node on the line represents the homodimer A:A.

$z$ is the combination of states defined by $x$ and $y$.

Enzymatic stimulation of a reaction.

General symbol for stimulation. A bar behind the arrowhead signifies necessity.

General symbol for inhibition.

Shorthand symbol for transcriptional activation.

Shorthand symbol for transcriptional inhibition.

Degradation products
Molecular Interaction Maps

The p53-Mdm2 and DNA Repair Regulatory Network

Figure 6B: The p53-Mdm2 and DNA repair regulatory network (version 2) - May 19, 1999

Taken from Kurt W. Kohn
The Protein Machine “Instruction Set”

On/Off switches

Protein

Binding Sites

Inaccessible

Inaccessible

Each protein has a structure of binary switches and binding sites. But not all may be always accessible.

cf. BioCalculus [Kitano&Nagasaki], κ-calculus [Danos&Laneve]

Switching of accessible switches.
- May cause other switches and binding sites to become (in)accessible.
- May be triggered or inhibited by nearby specific proteins in specific states.

Binding on accessible sites.
- May cause other switches and binding sites to become (in)accessible.
- May be triggered or inhibited by nearby specific proteins in specific states.
Notations for the Protein Machine

- **Stochastic π-Calculus**
  - Priami (following Hillston's PEPA) formalizes a stochastic version of π-calculus where channels have communication rates.

- **BioSPI**
  - Regev-Shapiro-Silverman propose modeling chemical interactions (exchange of electrons and small molecules) as “communication”.
  - Standard stochastic simulation algorithms (Gillespie) can be used to run in-silico experiments.
  - Complex formation is encoded via π-restriction.

- **PEPA**
  - Calder Gilmore and Hillston model the ERK pathway.

- **k-calculus**
  - Danos and Laneve (following Kitano’s BioCalculation) define a calculus where complex formation is primitive.

- **(Stochastic) Petri Nets**
  - S.Reddy’94 modeling pathways.
  - Srivastava Perterson and Bentley analyze and simulate E.coli stress response circuit.

- **Bio State Charts**
  - Harel uses State Charts to model biological interactions via a semi-graphical FSM notation.

- **Pathway Logic**
  - Talcott-Eker-Knapp-Lincoln use term-rewriting.

- **BioCham**
  - ChabrierRivier-Fages-Soliman use term-rewriting and CLT modelchecking.

- **Kohn Diagrams, Kitano Diagrams**

- **SBML (Systems Biology Markup Language)**
  - XML dialect for MIM’s:
    - Compartments (statically nested)
    - Reagents with concentrations
    - Reactions with various rate laws
  - Read and written by many tools via the Systems Biology Workbench protocol
    - Graph editors
    - Simulators (including simulation web services)
    - Databases
2. The Gene Machine

The “Central Dogma” of Molecular Biology

Pretty far from the atoms.
The Gene Machine “Instruction Set”

cf. Hybrid Petri Nets [Matsuno, Doi, Nagasaki, Miyano]

Regulation of a gene (positive and negative) influences transcription. The regulatory region has precise DNA sequences, but not meant for coding proteins: meant for binding regulators.

Transcription produces molecules (RNA or, through RNA, proteins) that bind to regulatory region of other genes (or that are end-products).

Human (and mammalian) Genome Size
3Gbp (Giga base pairs) 750MB @ 4bp/Byte (CD)
Non-repetitive: 1Gbp 250MB
In genes: 320Mbp 80MB
Coding: 160Mbp 40MB
Protein-coding genes: 30,000-40,000

M.Genitalium (smallest true organism)
580,073bp 145KB (eBook)
E.Coli (bacteria): 4Mbp 1MB (floppy)
Yeast (eukarya): 12Mbp 3MB (MP3 song)
Wheat 17Gbp 4.25GB (DVD)
**Gene Composition**

Is a shorthand for:

Under the assumptions [Kim & Tidor]
1) The solution is well-stirred
   (no spatial dependence on concentrations or rates).
2) There is no regulation cross-talk.
3) Control of expression is at transcription level only
   (no RNA-RNA or RNA-protein effects)
4) Transcriptions and translation rates monotonically
   affect mRNA and protein concentrations resp.

---

**Ex: Bistable Switch**

---

**Ex: Oscillator**

Expressed
Repressed
Expressing
Gene Regulatory Networks

http://strc.herts.ac.uk/bio/maria/NetBuilder/

Begin coding region
Indirect Gene Effects

No combination of standard high-throughput experiments can reconstruct an a-priori known gene/protein network [Wagner].

One of many bistable switches that cannot be described by pure gene regulatory networks [François & Hakim].
Structure of the Coding Region

The Central Dogma

DNA → transcription → mRNA → translation → Protein

RNA is not just an intermediary; it can:
- Fold-up like a protein
- Act like an enzyme
- Regulate other transcribed RNA
- Direct protein editing
- ...

Challenging the Dogma (in higher organisms)

97-98% of the transcriptional output of the human genome is non-protein-coding RNA.
30-40,000 "protein genes" (1.5% of genome)
60-100,000 "transcription units" (>30% of genome is transcribed)
Structure of a Regulatory Region

Fig. 1. Endo16 cis-regulatory system and interactive roles of module A. (A) Diversity of protein binding sites and organization into modular subregions [modified from (7)]. Specific DNA binding sites are indicated as red blocks; modular subregions are denoted by letters G to A (Bp, basal promoter). Proteins binding at the target sites considered in this work are indicated: Otx, SpOtx-1 (12); SpGCF1 (14); the proteins CG, Z, and P, which are not yet cloned; and protein C (CREB family protein (18)) in subregion F. Proteins for which sites occur in multiple regions of the DNA sequence (indicated by the black line) are shown beneath. (B) Sequence of module A and location of protein binding sites. Sites are indicated in the same colors as in (A). A fragment containing CG2 and CG3 sites as well as Bp has no endoderm-specific activity and serves other upstream cis-regulatory systems promiscuously; similarly, the Endo16 cis-regulatory system functions specifically with heterologous promoters substituted for Bp (5, 8, 19). Boxed sequences indicate conserved core elements of the target sites (7, 12, 14), not the complete target site sequences. (C) Integrative and interactive functions of module A (5, 8). Module A communicates the output of all upstream modules to the basal transcription apparatus. It also initiates endoderm expression, increases the output of modules B and S, and is required for functions of the upstream modules F, E, and DC. These functions are repression of expression in nonendodermal domains and enhancement of expression in response to LiO1.

2300bp! > average protein

Taken from Eric H Davidson
Function of a Regulatory Region

- DNA
- Begin coding region
- And
- Or
- Sum
- Amplify
- Gate
- time varying influence
- scalar factor
- inhibitory switch

B
- If \( F = 1 \) or \( E = 1 \) or \( CD = 1 \) and \( Z = 1 \)
- \( \alpha = 1 \)
- else \( \alpha = 0 \)
- if \( P = 1 \) and \( CG_1 = 1 \)
- \( \beta = 2 \)
- else \( \beta = 0 \)
- if \( CG_1 = 1 \) and \( CG_2 = 1 \) and \( CG_3 = 1 \)
- \( \gamma = 2 \)
- else \( \gamma = 1 \)
- \( z(t) = B(t) + G(t) \)
- \( z(t) = \beta \cdot \gamma(t) \)
- if \( z(t) = 0 \)
- \( \zeta(t) = Otz(t) \)
- else \( \zeta(t) = r(t) \)
- if \( \alpha = 1 \)
- \( \eta(t) = 0 \)
- else \( \eta(t) = \xi(t) \)
- \( \xi(t) = \gamma \cdot \eta(t) \)
- Final output communicated to BTA

Repression functions of modules F, E, and DC mediated by Z site
Both P and CG1 needed for synergistic link with module B
Final step up of system output
Positive input from modules B and G
Synergistic amplification of module B output by CG-F subsystem
Switch determining whether Otz site in module A, or upstream modules (i.e., mainly module B), will control level of activity
Repression function inoperative in endoderm but blocks activity elsewhere

Taken from Eric H Davidson
Gene Machine Programs

• All that goes to show that:
  - The faithful description of even a simple genetic network is probably going to require writing a fairly substantial “program”/model.
The Programming Model

Strange facts about genetic networks:

**Not an operator algebra.** The output of each gate is fixed and pre-determined; it is never a function of the input!

**Not term-rewriting, nor Petri nets.** Inhibition is widespread.

**Not Communicating Sequential Processes.** Feedback is widespread: asynchronous communication needed to avoid immediate self-deadlocks. Even the simplest gates cannot be modeled as a single synchronous process.

**Not Message-Passing between genes.** Messages themselves have behavior (e.g., they stochastically decay and combine), hence messages are processes as well.

**Not Data-Flow.** Any attempt to use data-flow-style modeling seems doomed because of widespread loops that lead to deadlocks.

**Stochastic broadcasting.** The apparently crude idea of broadcasting a whole bunch of asynchronous decaying messages to activate a future gate, means there are never any “pipeline full” deadlocks, even in presence of abundant feedback loops.

**Stochastic degradation.** Degradation is fundamental for system stability, and at the same time can lead to sudden instability and detection of concentration levels.
Notations for the Gene Machine

• Many of the same techniques as for the Protein Machine apply.
  - Process Calculi, Petri Nets, Term-Rewriting Systems...

• But the “programming model” is different.
  - Asynchronous stochastic control.
  - Biologically poorly understood.
  - Network “motifs” are being analyzed.

• Specific techniques:
  - Hybrid Petri Nets
    • [Matsuno, Doi, Nagasaki, Miyano] Gene Regulation
    • Genomic Object Net www.genomicobject.net

• Gene Regulation Diagrams
• Mixed Gene-Protein Diagrams
3. The Membrane Machine

Very far from the atoms.

Molecular transport and transformation through dynamic compartment fusion and fission.

The Instruction Set

Well, what is all that for? “Given the complicated pathways that have evolved to synthesize them, it seems likely that these [modified proteins] have important functions, but for the most part these functions are not known” [MBP p.609]
Membrane Fusion

1. Cell membrane
2. Virus membrane

Aggressive fusion (virus)

3. By unknown mechanisms, the exoplasmic leaflets of the two membranes fuse” [MCB p745]

Proposed sequence of events in pH sensitive hemagglutinin membrane fusion

4. 5. 6. Taken from Tamm Laboratory

Cooperative fusion (vesicle)

“Fusion of the two membranes immediately follows prefusion, but precisely how this occurs is not known” [MCB p742]
Membrane Fission

Assembly and disassembly of the clathrin coat

Vesicle Formation

“Nonetheless, the actual process whereby a segment of phospholipid bilayer is ‘pinched off’ to form a pit and eventually a new vesicle is still not understood” [MCB p.746]

Cytokinesis (Mitosis)
Notations for the Membrane Machine

- “Snapshot” diagrams
  - In biology literature.
- P-Systems
  
  http://psystems.disco.unimib.it/.

- BioAmbients
  - An extension of BioSPI along Ambient Calculus lines (with more bio-relevant mobility primitives) to model dynamic compartments.

- Brane Calculi
  - Computation on the membrane...
The Membrane Machine “Instruction Set”

Arbitrary subsystem

Endo → Exo

Zero case

Pino

One case

Phago

Arbitrary subsystem

Mito → Mate

Zero case

Drip

One case

Bud
Locally Implementable!

Global Views

Mito \rightarrow \text{(Fission)}

Mate \rightarrow \text{(Fusion)}

Endo \rightarrow \text{(Fission)}

Exo \rightarrow \text{(Fusion)}

Same Local View!
Mito/Mate by 3 Endo/Exo
Ex: Viral Reproduction

Virus

RNA
Capsid
Membrane
Envelope protein

Nucleocapsid

Cytosol

Phago

Endosome

Mate

Exo

Disassembly

Translation

RNA

Assembly

RNA Replication

Budding

Nucleus

Endoplasmic Reticulum

Drip

Vesicle

Infection

Replication

Progeny

[MBC p.279]
Abstract Machines of Biochemistry

Gene Machine
Nucleotides

Protein Machine
Aminoacids

Membrane Machine
Phospholipids

Regulation

Makes proteins, where/how/howmuch

Signals conditions and events

Directs membrane construction and protein embedding

Holds genome(s), confines regulators

Holds receptors, actuates and hosts reactions

Model Integration
Different time and space scales

Implements fusion, fission

Metabolism, Propulsion
Signal Processing
Molecular Transport

Confinement
Storage
Bulk Transport
Process Calculi
Chemistry vs. $\pi$-calculus

A process calculus (chemistry, or SBML)

\[ \text{Na} + \text{Cl} \rightarrow_{k_1} \text{Na}^+ + \text{Cl}^- \]
\[ \text{Na}^+ + \text{Cl}^- \rightarrow_{k_2} \text{Na} + \text{Cl} \]

This graphical representation degenerates into spaghetti diagrams: precise and dynamic, but not scalable, structured, or maintainable.

A different process calculus

\[ \text{Na} = e_{k_1}!. e_{k_2}? . \text{Na} \]
\[ \text{Cl} = e_{k_1}? . e_{k_2}!. \text{Cl} \]
Stochastic $\pi$-calculus Executive Summary

- A process calculus:
  - The modular representation of concurrent (and stochastic) processes of all kinds.
  - Cuts down to CTMCs in the finite case (not always), then standard tools are applicable.
  - Can be given friendly automata-like scalable graphical syntax (work in progress).
  - Is directly executable (e.g. via Gillespie).
  - Is analyzable (large body of literature, at least in the non-stochastic case).

Figure 2. Regulating Gene Expression by Positive Feedback [8]

Figure 3. Protein $A$ molecules v.s. time in presence (left) and absence (right) of $TF$
Regev-Shapiro: “Molecules as Computation”

<table>
<thead>
<tr>
<th>Molecule</th>
<th>Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interaction capability</td>
<td>Channel</td>
</tr>
<tr>
<td>Interaction</td>
<td>Communication</td>
</tr>
<tr>
<td>Modification</td>
<td>State change</td>
</tr>
</tbody>
</table>

Cellular Abstractions: Cells as Computation
Regev&Shapiro NATURE vol 419, 2002-09-26, 343

This mapping works well both for the “protein machine” (synchronous communication) and the “gene machine” (asynchronous communication). But is not enough for the “membrane machine”. 
\( \pi \text{-calculus} \)

**Syntax**

\[
\pi ::= \ x(y) \quad \text{receive } y \text{ along } x \\
\overline{x}(y) \quad \text{send } y \text{ along } x
\]

\[
P ::= 0 \mid \sum_{i \in I} \pi_i.P_i \mid [x = y]P \mid P_1|P_2 \mid (\text{new } x)P \mid 1P
\]

**Structural congruence**

**Renaming of bound variables**

\[
x(y).P = x(z).\{z/y\}P \quad \text{if } z \notin FN(P)
\]

\[
\text{(new } y).P = \text{(new } z).\{z/y\}P \quad \text{if } z \notin FN(P)
\]

**Structural congruence laws**

\[
P|Q \equiv Q|P \quad \text{commutativity of parallel composition}
\]

\[
(P|Q)|R \equiv P|(Q|R) \quad \text{associativity of parallel composition}
\]

\[
P + Q \equiv Q + P \quad \text{commutativity of summation}
\]

\[
(P + Q) + R \equiv P + (Q + R) \quad \text{associativity of summation}
\]

\[
\text{(new } x)0 \equiv 0 \quad \text{restriction of inert processes}
\]

\[
\text{(new } x)(\text{new } y)P \equiv \text{(new } y)(\text{new } x)P \quad \text{polyadic restriction}
\]

\[
((\text{new } x)P|Q) \equiv \text{(new } x)(P|Q) \quad \text{if } x \notin FN(Q) \quad \text{scope extrusion}
\]

\[
\text{!}P \equiv P\,!P \quad \text{replication}
\]

**Reaction rules**

\[
(\cdots + \overline{x}(z).Q)(\cdots + x(y).P) \rightarrow Q|P \{z/y\} \quad \text{communication (COMM)}
\]

\[
\frac{P \rightarrow P'}{P|Q \rightarrow P'|Q} \quad \text{reaction under parallel composition (PAR)}
\]

\[
\frac{P \rightarrow P'}{(\text{new } x)P \rightarrow (\text{new } x)P'} \quad \text{reaction under restriction (RES)}
\]

\[
\frac{Q \equiv P \rightarrow P' \ P' \equiv Q'}{Q \rightarrow Q'} \quad \text{structural congruence (STRUCT)}
\]
Stochastic $\pi$-calculus

- Stochastic extension of $\pi$-calculus. [C.Priami]

  Associate a single parameter $r$ (rate) in $(0, \infty]$ of an exponential distribution to each activity $a$; it describes the stochastic behavior of the activity

  $a.P$ is replaced by $(a, r).P$

  Exponential distribution guarantees the **memoryless property**: the time at which a change of state occurs is independent of the time at which the last change of state occurred.

  **Race condition** is defined in a **probabilistic competitive** context: all the activities that are enabled in a state compete and the fastest one succeeds.

Brane Calculi

Computation “on” the membrane
Membranes are Oriented 2D Surfaces

Lipid Bilayer
Self-assembling
Largely impermeable
Asymmetrical (in real cells)
With embedded proteins
A 2D fluid inside a 3D fluid!

Lipid
- Hydrophilic head
- Hydrophobic tail

Diffusion (fast)

Flip (rare)

Extracellular Space (H₂O)
- 5nm
- ~60 atoms

Cytosol (H₂O)

Embedded membrane proteins

Channels, Pumps (selective, directional)

(Not spontaneous)
**Brane Calculi**

### Systems

$$P, Q ::= \diamond | P \circ Q | !P | \sigma(P)$$

Rests of membranes

### Branes

$$\sigma, \tau ::= 0 | \sigma|\tau | !\sigma | a.\sigma$$

Combinations of actions

### Actions

$$a ::= 1 | ...$$

(fill in as needed)

#### 1D fluids ($\sigma$) inside a 2D fluid ($P$)

TWO commutative monoids instead of
ONE of normal process calculi

$$\sigma(P)$$

$$\sigma|\tau(P)$$

$$a.\sigma|\tau = (a.\sigma)|\tau$$

N.B. Restriction ($\gamma n$) could be added to both systems and branes. It usually would originate in branes, but would extrude to whole systems.
Congruence $\equiv$ and Reaction

**System**

Fluidity
- $P \circ Q \equiv Q \circ P$
- $P \circ (Q \circ R) \equiv (P \circ Q) \circ R$
- $P \circ \circ \equiv P$

Plenitude
- $!P \equiv P \circ !P$ etc.

Units
- $0 \circ \circ \equiv \circ$ Froth/Fizz

Congruence
- $P \equiv Q \Rightarrow P \circ R \equiv Q \circ R$
- $P \equiv Q \Rightarrow !P \equiv !Q$
- $P \equiv Q \land \sigma \equiv \tau \Rightarrow \sigma(P) \equiv \tau(Q)$

**Brane**

- $\sigma|\tau \equiv \tau|\sigma$
- $\sigma|(\tau|\rho) \equiv (\sigma|\tau)|\rho$
- $\sigma|0 \equiv \sigma$

Plenitude
- $!\sigma \equiv \sigma|!\sigma$ etc.

Units
- $1.\sigma \equiv \sigma$ Inaction

Congruence
- $\sigma \equiv \tau \Rightarrow \tau|\rho \equiv \tau|\sigma$
- $\sigma \equiv \tau \Rightarrow !\sigma \equiv !\tau$
- $\sigma \equiv \tau \Rightarrow a.\sigma \equiv a.\tau$

**Reaction is up to congruence**

- $P \equiv P' \land P' \rightarrow Q' \land Q' \equiv Q \Rightarrow P \rightarrow Q$

**Reactions in solution**

- $P \rightarrow Q \Rightarrow P \circ R \rightarrow Q \circ R$
- $P \rightarrow Q \Rightarrow \sigma(P) \rightarrow \sigma(Q)$

This is the whole semantics, except for the effects of individual actions.
Brane Reactions

\[ a ::= \ldots | \gamma_n | \gamma_n(\rho) | \gamma_n | \gamma_n | \circ(\rho) \]

**actions**

- phago \(\gamma\), exo \(\gamma\), pino \(\circ\)

**Coordination Tags**

Sometimes omitted

---

**Old “spontaneous” endo splits into phagocytosis (phago, often still pronounced endo) and pinocytosis (pino).**
Brane Reactions (Cartoons)

Exo

Pino

Phago
Phago: $\omega_n.\sigma|\sigma'(P) \circ \omega_n^\perp(\rho).\tau|\tau'(Q) \rightarrow \tau|\tau'(\rho|\sigma|\sigma'(P)\circ Q)$

Exo: $\omega_n^\perp.\tau|\tau'(\omega_n.\sigma|\sigma'(P)\circ Q) \rightarrow P \circ \sigma|\sigma'|\tau|\tau'(Q)$

Pino: $\otimes(\rho).\sigma|\sigma'(P) \rightarrow \sigma|\sigma'(\rho\circ P)$

N.B.: the parity of nesting of $P$ and $Q$ is preserved; this makes the reactions preserve bitonality.

N.B.: in Phago (and Pino), one could perhaps require $r$ to be, conservatively, a piece of $t$, by a non-linear rewrite:

CPhago: $\omega_n.\sigma|\sigma'(P) \circ \omega_n^\perp(\rho).\tau|\tau'|\rho'(Q) \rightarrow \tau|\tau'(\rho|\sigma|\sigma'(P)\circ Q)$
Ex: Viral Reproduction

Infection

Replication

Progeny
Ex: Viral Infection

- Virus
- Cell
- Membrane
- Endosome
- Vesicle

Phago
Mate
Exo
Assume:
\[ \text{nucap} \circ \text{cytosol} \rightarrow \rightarrow \text{nucap}^n \circ \text{envelope-vesicle}^m \circ \text{cytosol}' \]
by available cellular machinery

Then:

\[ \text{cell} \]
\[ \text{envelope-vesicle} \quad \text{nucap} \]
\[ \text{bud} \circ \text{vRNA} \circ \text{cytosol}' \]
\[ \text{Exo} \]

\[ \text{bud} \circ \text{vRNA} \circ \text{cytosol}' \]
\[ \text{cell} \quad \text{virus} \]

\[ \text{Ex: Viral Progeny} \]
“On Brane” vs. “In Brane”

Original “on brane”
Exo of Brane Calculus

“In brane” encoding
(e.g. in BioAmbients
or SMBL) goes wrong

“Ball bearing”
encoding; best we can
do “in brane”

Awkward encoding. And all kinds of things
can go wrong in the intermediate state.

• One cannot easily represent the Exo reaction in BioAmbients or any such
compartment-based calculus, nor can one easily add it as a new primitive!

• But we can add BioAmbients-like In/Out out to Brane Calculi if we want to.
**Q:** “The data are accumulating and the computers are humming, what we are lacking are the words, the grammar and the syntax of a new language…”  
D. Bray (TIBS 22(9):325-326, 1997)

**A:** “The most advanced tools for computer process description seem to be also the best tools for the description of biomolecular systems.”  
E. Shapiro (Lecture Notes)
References


Papers

BioAmbients
a stochastic calculus with compartments.

Brane Calculi
process calculi with computation “on” the membranes, not inside them.

Bitonal Systems
membrane reactions and their connections to “local” patch reactions.

www.luca.demon.co.uk/BioComputing.htm