Languages & Notations for Systems Biology

Luca Cardelli
Microsoft Research
Cambridge UK
2004-11-12 Sheffield

www.luca.demon.co.uk
Aims

BioInformatics: Mining -omics experimental data.
Molecular Biology: Understanding the components of living matter.
Systems Biology: Understanding the connectivity of those components.

Modeling biological systems
not as continuous systems (traditional)
but as reactive systems (information-processing)

Because they have some similar features:

Deep layering of abstractions.
Complex composition of simpler components.
Discrete (non-linear) transitions.
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 description languages, for various purposes.
- Studying their kinetics (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
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. “How do components interact?”
2. “Gather high-throughput data.”

Glycan Machine
Sugars
Surface and Extracellular Features

Protein Machine
Aminoacids
Metabolism, Propulsion
Signal Processing
Molecular Transport

Membrane Machine
Phospholipids
Confinement
Storage
Bulk Transport

Gene Machine
Nucleotides
Regulation
Diverse:
- chemical toolkits
- instruction sets
- programming models
- notations

Makes proteins: where, when, how much
Directs membrane construction and protein embedding
Holds genome(s), confines regulators
Holds receptors, actuators, hosts reactions
Model Integration
Different time and space scales
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.

![Diagram of protein interactions]

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/
Domain architecture and autoinhibitory interactions in modular switch proteins. (a) Src family kinases contain N-terminal SH3 and SH2 domains, and a kinase domain flanked by intramolecular SH3-binding and SH2-binding sites (when the C-terminal motif tyrosine is phosphorylated by Csk). The crystal structures of several family members show that both intramolecular domain interactions function in concert to lock the kinase in an inactive conformation. Activating stimuli (red) include external SH2 or SH3 ligands. After initial activation, the kinase is maintained in an active state by autophosphorylation of its activation loop. (b) SHP-2 phosphatase contains two SH2 domains and a phosphatase domain. The crystal structure of the phosphatase shows that the N-terminal SH2 domain participates in an autoinhibitory interaction that directly blocks the phosphatase active site. Binding of external SH2 ligands activates by disrupting the autoinhibitory interaction. (c) N-WASP contains an Ena/VASP homology 1 (EVH1) domain, a B motif, a GBD, a proline-rich segment (pro) and an output region (VCA) that alone binds the Arp2/3 complex and stimulates its actin nucleation activity. The B and GBD motifs are required to repress activity and, by current models, are thought to participate in intramolecular interactions (only the structure of the GBD intramolecular complex for WASP is known). GTP-bound Cdc42 and FIP2 synergistically activate N-WASP.

Some Allosteric Switches

<table>
<thead>
<tr>
<th>Domain architecture</th>
<th>Repressed state</th>
<th>Activated state</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) Input SH3-SH2 → P</td>
<td>SH3-SH2→ P</td>
<td>On</td>
</tr>
<tr>
<td>(b) Input SH2-SH2 → P</td>
<td>SH2-SH2→ P</td>
<td>On</td>
</tr>
<tr>
<td>(c) Input EVH1 → GBD → pro VCA</td>
<td>Arp2/3</td>
<td>On</td>
</tr>
</tbody>
</table>

- 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.

Covariant 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.
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. Kohl
The Protein Machine “Instruction Set”

**On/Off switches**

Inaccessible

Protein

Binding Sites

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

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.

**cf.** BioCalculus [Kitano & Nagasaki], \(\kappa\)-calculus [Danos & Laneve]
Notations for the Protein Machine

• Stochastic \( \pi \)-Calculus
  - Priami (following Hillston’s PEPA) formalizes a stochastic version of \( \pi \)-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 \( \pi \)-restriction.

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

• k-calculus
  - Danos and Laneve (following Kitano’s BioCalculus) 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
  - Chabrier-Rivier-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

DNA → messenger RNA → PROTEIN → SYSTEMS

- 4-letter digital code
- 4-letter digital code
- 20-letter digital code
- 50,000(?) shapes

Lactose Operon

Metabolic space
- Meabolite 1 → Metabolite 2
- Protein 1 → Protein 2 → Complex 34

Protein space
- Gene 1 → Gene 2 → Gene 3 → Gene 4

Gene space

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

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
Repessed
Expressing
Gene Regulatory Networks

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

NetBuilder
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 [Francois & Hakim].

Fig. 1. The importance of specifying gene activity when reconstructing genetic networks. (a) A hypothetical biochemical pathway involving two transcription factors, a protein kinase, and a protein phosphatase, as well as the genes encoding them. See text for details. (b) Shown is a list of perturbation effects for each of the five genes in (a), when perturbing individual genes by deleting them, and when using mRNA expression level as an indicator of gene activity. The left-most symbol in each line stands for the perturbed gene. To the right of each colon is a list of genes whose activity is affected by the perturbation. (c) Analogous to (b) but for a different notion of gene activity (phosphorylation state).
The Central Dogma

DNA \rightarrow \text{transcription} \rightarrow \text{mRNA} \rightarrow \text{translation} \rightarrow \text{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

C Module A functions:

Vegetal plate expression in early development:

Synergism with modules B and G enhancing endoderm expression in later development:

Repression in ectoderm (modules E and F) and skeletogenic mesenchyme (module DC):

Modules E, F and DC with LICI treatment:

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 [a 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 CG3 and CG4 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 G, 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 LICI.

2300bp! > average protein

Taken from Eric H. Davidson
Function of a Regulatory Region

If \( F = 1 \) or \( E = 1 \) or \( CD = 1 \) and \( Z = 1 \), repression functions of modules \( F, E, \) and \( DC \) mediated by \( Z \) site.

\( \alpha = 1 \), if \( (P = 1 \text{ and } CG_1 = 1) \)

\( \beta = 2 \), if \( (CG_1 = 1 \text{ and } CG_2 = 1 \text{ and } CG_3 = 1) \)

\( \gamma = 2 \), final step up of system output.

\( \beta(t) = B(t) + G(t) \)

Positive input from modules \( B \text{ and } G \)

\( \xi(t) = \beta \cdot \xi(t) \)

Synergistic amplification of module \( B \) output by \( CG_1-P \) subsystem.

If \( (\xi(t) = 0) \),

\( \zeta(t) = \text{Otx}(t) \)

Switch determining whether \( \text{Otx} \) site in module \( A \), or upstream modules (i.e., mainly module \( B \)), will control level of activity.

If \( (\alpha = 1) \),

\( \eta(t) = 0 \)

Repression function inoperative in endoderm but blocks activity elsewhere.

Else \( \eta(t) = \xi(t) \)

Final output communicated to BTA.
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

Molecular transport and transformation through dynamic compartment fusion and fission.

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

Aggressive fusion (virus)

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

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

• 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 ➔ (Fission)
Mate ➔ (Fusion)
Endo ➔ (Fission)
Exo ➔ (Fusion)

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

Infection  |  Replication  |  Progeny

(MBC p.279)

[Diagram of viral reproduction process]

- Virus
- RNA (Capsid, Membrane, Envelope protein)
- Nucleocapsid
- Cytosol
- Translation
- Disassembly
- Assembly
- Budding
- Endosome
- Mate
- Exo
- Nucleus
- Endoplasmic Reticulum
- Vesicle
- Drip

[Annotated diagram showing the stages of viral reproduction from infection to progeny]
Abstract Machines of Biochemistry

Gene Machine
Nucleotides

Regulation

Protein Machine
Aminoacids
Metabolism, Propulsion
Signal Processing
Molecular Transport

Membrane Machine
Phospholipids
Confinement
Storage
Bulk Transport

Model Integration
Different time
and space scales

Holds genome(s),
confines regulators

Directs membrane construction
and protein embedding

Makes proteins,
where/when/howmuch

Signals conditions and events

Holds receptors, actuators
hosts reactions

Implements fusion, fission
Stochastic Process Calculi
Chemistry vs. $\pi$-calculus

A process calculus (chemistry, or SBML)

\[
\begin{align*}
Na + Cl & \rightarrow_{k_1} Na^+ + Cl^- \\
Na^+ + Cl^- & \rightarrow_{k_2} Na + Cl
\end{align*}
\]

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

A compositional graphical representation, and the corresponding calculus.

\[
\begin{align*}
Na & \xrightarrow{k_1} Cl \\
Cl & \xrightarrow{k_1} Na \\
Na^+ & \xrightarrow{k_2} Cl^- \\
Cl^- & \xrightarrow{k_2} Na^+
\end{align*}
\]

The same "model"

\[
\begin{align*}
Na &= e_{k_1}. e_{k_2}?. Na \\
Cl &= e_{k_1}?. e_{k_2}!. Cl
\end{align*}
\]

A different process calculus ($\pi$)
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).
### 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”.
$\pi$-calculus

Syntax

$\pi ::= x(y) \quad$ receive $y$ along $x$
$\overline{x}(y) \quad$ send $y$ along $x$

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

Structural congruence

Renaming of bound variables

$x(y).P = x(z).\{\{z/y\} \cdot P\} \quad$ if $z \notin FN(P)$
$(\text{new } y).P = (\text{new } z).\{\{z/y\} \cdot P\} \quad$ if $z \notin FN(P)$

Structural congruence laws

$P|Q \equiv Q|P$ \quad commutativity of parallel composition
$(P|Q)|R \equiv P|(Q|R)$ \quad associativity of parallel composition
$P + Q \equiv Q + P$ \quad commutativity of summation
$(P + Q) + R \equiv P + (Q + R)$ \quad associativity of summation
$(\text{new } x)0 \equiv 0$ \quad restriction of inert processes
$(\text{new } x)(\text{new } y)P \equiv (\text{new } y)(\text{new } x)P$ \quad polyadic restriction
$((\text{new } x)P)|Q \equiv (\text{new } x)(P|Q) \quad$ if $x \notin FN(Q)$ \quad scope extrusion
$!P \equiv P!P$ \quad replication

Reaction rules

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

$P \rightarrow P'$ \quad reaction under parallel composition (PAR)
$P|Q \rightarrow P'|Q$ \quad reaction under restriction (RES)
$(\text{new } x)P \rightarrow (\text{new } x)P'$ \quad reaction under restriction (RES)
$Q \equiv P \quad P \rightarrow P' \quad P' \equiv Q'$ \quad 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.

$$a.P \text{ is replaced by } (a, r).P$$

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.

Proteins
MAPK Cascade - Huang&Ferrell


Biochemistry: Huang and Ferrell

Table 2. Predicted Hill coefficients for MAP kinase cascade components: Varying the assumed $K_m$ values

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Range of assumed $K_m$ values</th>
<th>Range of effective Hill coefficients (nM) predicted for</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKKK: 1.0, MAPKK: 1.7, MAPK: 4.9</td>
</tr>
<tr>
<td>MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.7, MAPK: 4.9</td>
</tr>
<tr>
<td>1. MAPK + MAPKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKKK: 1.0, MAPKK: 1.3-2.3, MAPK: 4.0-5.1</td>
</tr>
<tr>
<td>2. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.5-1.9, MAPK: 3.6-6.7</td>
</tr>
<tr>
<td>3. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.2-2.4, MAPK: 3.8-5.2</td>
</tr>
<tr>
<td>4. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM (300 nM$^3$)</td>
<td>MAPKK: 1.0, MAPKK: 1.7-1.8, MAPK: 4.1-6.4</td>
</tr>
<tr>
<td>5. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.7, MAPK: 3.7-6.2</td>
</tr>
<tr>
<td>6. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.7, MAPK: 4.3-5.2</td>
</tr>
<tr>
<td>7. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.7, MAPK: 3.4-6.1</td>
</tr>
<tr>
<td>8. MAPKK + MAPKKK + MAPKKK</td>
<td>60-1500 nM</td>
<td>MAPKK: 1.0, MAPKK: 1.7, MAPK: 4.7-5.1</td>
</tr>
</tbody>
</table>

The assumed $K_m$ values for each reaction were individually varied over the ranges shown, with the assumed $K_m$ values for the other nine reactions held constant. The effective Hill coefficients were calculated from the steepness of the predicted stimulus/response curves, as described in the text.

The $K_m$ value for reaction 7 has been measured to be 300 nM for the phosphorylation of a mammalian MAPK by a MAPKK (N. Ahn, personal communication). All of the other $K_m$ values were initially assumed to be 300 nM as well.

Calculations. Eqs. 1-10 represent the reactions of the MAPK cascade, which are shown schematically in Fig. 1. We have used Goldbeter and Koshland's nomenclature for the rate constants—the letter $a$ denotes association, $d$ denotes dissociation without catalysis, and $k$ denotes product formation (11). KKKK denotes MAPKKK; KK denotes MAPKK; and K denotes MAPK.

**Fig. 1.** Schematic view of the MAPK cascade. Activation of MAPK depends upon the phosphorylation of two conserved sites [Thr-183 and Tyr-185 in rat p42 MAPK/Erk2 (4, 5)]. Full activation of MAPKK also requires phosphorylation of two sites [Ser-218 and Ser-222 in mouse Mek-1/MKK1 (6–10)]. Detailed mechanisms for the activation of various MAPKK (e.g., Raf-1, B-Raf, Mos) are not yet established; here we assume that MAPKKKs are activated and inactivated by enzymes we denote E1 and E2. MAPKK* denotes activated MAPKK. MAPK and MAPKK* denote singly and doubly phosphorylated MAPKK, respectively. MAPK-P and MAPKK-PP denote singly and doubly phosphorylated MAPKK. Pase denotes phosphatase.
As 18 Ordinary Differential Equations

The 10 reactions described above give rise to 18 rate equations:

\[
\frac{d}{dt}[KKK] = -a_1[KKK][E1] + d_1[KKK\cdot E1] + k_2[KKK^*\cdot E2]
\]

\[
\frac{d}{dt}[KKK\cdot E1] = a_1[KKK][E1] - (d_1 + k_1)[KKK\cdot E1]
\]

\[
\frac{d}{dt}[KKK^*] = -a_2[KKK^*][E2] + d_2[KKK^*\cdot E2]
\]

\[
\frac{d}{dt}[KKK\cdot E2] = a_2[KKK^*][E2] - (d_2 + k_2)[KKK^*\cdot E2]
\]

\[
\frac{d}{dt}[KK] = -a_3[KK][KKK^*] + d_3[KK-KKK^*] + k_4[KK\cdot PP\cdot P'ase]
\]

\[
\frac{d}{dt}[KK\cdot KKK^*] = a_3[KK][KKK^*] - (d_3 + k_3)[KK\cdot KKK^*]
\]

\[
\frac{d}{dt}[KK-P] = -a_4[KK-P][KK P'ase] + d_4[KK-P\cdot PP\cdot P'ase]
\]

\[
\frac{d}{dt}[K\cdot KKK-P] = a_5[KK-P][KKK^*] + d_5[KK-P\cdot KKK^*] - a_5[KK-P][KKK^*]
\]

\[
\frac{d}{dt}[KK-P\cdot PP\cdot P'ase] = a_4[KK-P][KK P'ase]
\]

\[
- (d_4 + k_4)[KK-P\cdot KK P'ase]
\]

\[
\frac{d}{dt}[KK-P\cdot KK P'ase] = a_5[KK-P][KKK^*]
\]

\[
- (d_5 + k_5)[KK-P\cdot KKK^*]
\]

\[
\frac{d}{dt}[KK-PP] = k_5[KK-P\cdot KKK^*] - a_6[KK-PP][KK P'ase]
\]

\[
+ d_6[KK-PP\cdot KK P'ase] - a_7[KK-PP][K]
\]

\[
+ (d_7 + k_7)[K\cdot KK-PP]
\]

\[
+ (d_9 + k_9)[K\cdot PP\cdot KK-PP]
\]

\[
- a_9[K\cdot P][KK-PP]
\]

\[
\frac{d}{dt}[KK-PP\cdot KK P'ase] = a_6[KK-PP][KK P'ase]
\]

\[
- (d_6 + k_6)[KK-PP\cdot KK P'ase]
\]

\[
\frac{d}{dt}[K] = -a_7[K][KK-PP] + d_7[K\cdot KK-PP]
\]

\[
+ k_8[K\cdot P\cdot K P'ase]
\]

\[
\frac{d}{dt}[K\cdot KK-PP] = a_7[K][KK-PP] - (d_7 + k_7)[K\cdot KK-PP]
\]
... Plus 7 conservation equations

$$\frac{d}{dt} [K-P] = k_2 [K\cdot KK\cdot PP] - a_6 [K-P][K\cdot P'ase]$$

$$+ d_6 [K-P \cdot K\cdot P'ase] - a_6 [K-P][KK\cdot PP]$$

$$+ d_5 [K-P \cdot KK\cdot PP] + k_{10} [KK\cdot PP \cdot K\cdot P'ase] \quad [24]$$

$$\frac{d}{dt} [K\cdot P\cdot K\cdot P'ase] = a_6 [K-P][K\cdot P'ase]$$

$$- (d_6 + k_6) [K\cdot P \cdot K\cdot P'ase]$$

$$\frac{d}{dt} [K\cdot P\cdot KK\cdot PP] = a_9 [K-P][KK\cdot PP]$$

$$- (d_9 + k_6) [K\cdot P \cdot KK\cdot PP] \quad [26]$$

$$\frac{d}{dt} [K\cdot PP] = -a_{10} [K\cdot PP][K\cdot P'ase]$$

$$+ d_{10} [K\cdot PP \cdot K\cdot P'ase] + k_9 [K\cdot P \cdot KK\cdot PP] \quad [27]$$

$$\frac{d}{dt} [K\cdot PP\cdot K\cdot P'ase] = a_{10} [K\cdot PP][K\cdot P'ase]$$

$$- (d_{10} + k_{10}) [K\cdot PP \cdot K\cdot P'ase] \quad [28]$$

In addition, there are seven conservation equations (Eqs. 29-35).

$$[KK_{tot}] = [KK] + [KK\cdot K] + [KK\cdot K\cdot P]$$

$$+ [KK\cdot K\cdot E2]$$

$$+ [KK\cdot K\cdot E2]$$

$$[K\cdot E1_{tot}] = [E1] + [KK\cdot E1] \quad [30]$$

$$[E2_{tot}] = [E2] + [KK\cdot E2] \quad [31]$$

$$[KK_{tot}] = [KK] + [KK\cdot P] + [KK\cdot PP] + [KK\cdot K\cdot KK\cdot K]$$

$$+ [KK\cdot P\cdot K\cdot KK\cdot K] + [KK\cdot P\cdot KK\cdot P'ase]$$

$$+ [KK\cdot PP\cdot K\cdot P'ase]$$

$$+ [KK\cdot PP \cdot K\cdot P] + [KK\cdot PP \cdot K\cdot P']$$

$$[K\cdot P'ase_{tot}] = [K\cdot P'ase] + [KK\cdot P'ase\cdot KK\cdot P]$$

$$+ [K\cdot P'ase \cdot KK\cdot PP] \quad [33]$$

$$[K_{tot}] = [K] + [K\cdot P] + [K\cdot PP] + [KK\cdot PP\cdot K]$$

$$+ [KK\cdot PP \cdot K\cdot P] + [K\cdot P\cdot K\cdot P'ase] + [KK\cdot P\cdot K\cdot P'ase] \quad [34]$$

$$[K\cdot P'ase_{tot}] = [K\cdot P'ase] + [K\cdot P\cdot K\cdot P'ase]$$

$$+ [K\cdot PP \cdot K\cdot P'ase] \quad [35]$$

These equations were solved numerically using the Runge–Kutta-based NDSolve algorithm in Mathematica (Wolfram Research, Champaign, IL). An annotated copy of the Mathematica code for the MAPK cascade rate equations can be obtained from J.E.F.
MAPK Cascade in SPiM

| !KK(); |
| new d1:r1:<> |
| (a1<d1>;(d1<>;KK<> + k1<>;KKst<>)) |

| !KKst(); |
| new d2:r2:<> |
| (a2<d2>;(d2<>;KKKst<> + k2<>;KKK<>) + a3<d3>;(d3<>;KKKst<> + k3<>;KKKst<>) + a5<d5>;(d5<>;KKKst<> + k5<>;KKKst<>)) |

| !E1(); |
| a1(d1);(d1<>;E1<> + k1<>;E1<>)) |

| !E2(); |
| a2(d2);(d2<>;E2<> + k2<>;E2<>)) |

| !KK(); |
| new d3:r3:<> |
| (a3<d3>;(d3<>;KK<> + k3<>;KK_P<>)) |

| !KK_P(); |
| new d4:r4:<> new d5:r5:<> |
| (a4<d4>;(d4<>;KK_P<> + k4<>;KK<>) + a5<d5>;(d5<>;KK_P<> + k5<>;KK_PP<>)) |

| !KK_P(); |
| new d6:r6:<> |
| (a6<d6>;(d6<>;KK_P<> + k6<>;KK_P<>) + a7<d7>;(d7<>;KK_P<> + k7<>;KK_P<>) + a9<d9>;(d9<>;KK_P<> + k9<>;KK_P<>)) |

| !E1(); |
| a1(d1);(d1<>;E1<> + k1<>;E1<>)) |

| !E2(); |
| a2(d2);(d2<>;E2<> + k2<>;E2<>)) |

| !KK_P(); |
| new d7:r7:<> |
| (a7<d7>;(d7<>;K<> + k7<>;K_P<>)) |

| !K(); |
| new d8:r8:<> new d9:r9:<> |
| (a8<d8>;(d8<>;K_P<> + k8<>;K<>) + a9<d9>;(d9<>;K_P<> + k9<>;K_PP<>)) |

| !K_P(); |
| new d10:r10:<> |
| (a10<d10>;(d10<>;K_P<> + k10<>;K_P<>)) |

| !KPse(); |
| a8<d8>;(d8<>;KPse<> + k8<>;KPse<>) + a10<d10>;(d10<>;KPse<> + k10<>;KPse<>) |
new KKK::<> new KKKst::<> new E1::<> new E2::<>
new KK::<> new KK_P::<> new KK_PP::<> new KKPse::<>
new K::<> new K_P::<> new K_PP::<> new KPse::<>
new a1:1.0::<> new k1:1.0::<> new a2:1.0::<> new k2:1.0::<>
new a3:1.0::<> new k3:1.0::<> new a4:1.0::<> new k4:1.0::<>
new a5:1.0::<> new k5:1.0::<> new a6:1.0::<> new k6:1.0::<>
new a7:1.0::<> new k7:1.0::<> new a8:1.0::<> new k8:1.0::<>
new a9:1.0::<> new k9:1.0::<> new a10:1.0::<> new k10:1.0::<>
new spike::<>,int> (* a spike #2 high of #1 molecules *)
( !spike(a,n); if n=0 then () else (a<> | spike<a,n-1>)

| !KKK();
| new d1:1.0::<>
| (a1<d1>;(d1<>;KK<> + k1<>;KKst<>))

| !KKKst();
| new d2:1.0::<>
| (a2<d2>;(d2<>;KKKst<> + k2<>;KK<>) +
| a3<d3>;(d3<>;KKKst<> + k3<>;KKKst<>) +
| a5<d5>;(d5<>;KKKst<> + k5<>;KKKst<>))

| !E1();
| a1<d1>;d1<>;E1<> + k1<>;E1<>)

| !E2();
| a2<d2>;d2<>;E2<> + k2<>;E2<>)

| !KK();
| new d3:1.0::<>
| (a3<d3>;d3<>;KK<> + k3<>;KK_P<>))

| !KK_P();
| new d4:1.0::<>
| new d5:1.0::<>
| (a4<d4>;(d4<>;KK_P<> + k4<>;KK<>) +
| a5<d5>;(d5<>;KK_P<> + k5<>;KK_PP<>))

| !KK_PP();
| new d6:1.0::<>
| (a6<d6>;(d6<>;KK_PP<> + k6<>;KK<>) +
| a7<d7>;(d7<>;KK_PP<> + k7<>;KK_PP<>) +
| a9<d9>;(d9<>;KK_PP<> + k9<>;KK_PP<>))

| !KKPse();
| a4<d4>;d4<>;KKPse<> + k4<>;KKPse<> +
| a6<d6>;d6<>;KKPse<> + k6<>;KKPse<>)

| !K();
| new d7:1.0::<>
| (a7<d7>;(d7<>;K<> + k7<>;K_P<>))

| !K_P();
| new d8:1.0::<>
| new d9:1.0::<>
| (a8<d8>;(d8<>;K_P<> + k8<>;K<>) +
| a9<d9>;(d9<>;K_P<> + k9<>;K_PP<>))

| !K_PP();
| new d10:1.0::<>
| (a10<d10>;(d10<>;K_PP<> + k10<>;K_P<>))

| !KPse();
| a8<d8>;d8<>;KPse<> + k8<>;KPse<> +
| a10<d10>;d10<>;KPse<> + k10<>;KPse<>)

| E1<> (* input signal *) | E2<> | KKPse<> | KPse<> | spike<KKK,100> | spike<KK,100> | spike<K,100> )
MAPK Cascade Simulation

KKK
KK
K

KK-P
K-P

1xE1 injected

KKK* reacts mildly to E1
KK-PP rises quicker
K-PP flips up to 100!

KKK*
KK-PP

1xE1 removed
1xE1 injected

All coefficients 1.0 !!!
100xKKK, 100xKK, 100xK,
1xE2, 1xKKPse, 1xKPse.

Input is 1xE1.
Output is 100xK-PP
(ultrasensitivity).
MAPK Cascade Simulation

All coefficients 1.0 !!!
100xKKK, 100xKK, 100xK, 10xE2, 10xKKPse, 10xKPse.
(so 1xE1 is no longer sufficient to produce an output)
Genes
Gene Gates and Circuits

A gene gate

\[ \text{neg}[a,b] \triangleq a_r() \cdot \tau_h \cdot \text{neg}[a,b] + \tau_e \cdot (\text{ptn}[b] \mid \text{neg}[a,b]) \]

\[ \text{ptn}[p] \triangleq p_r \cdot \text{ptn}[p] + \tau_\delta \]

A genetic circuit (engineered in E.Coli)

\[ \text{neg}[a,b] \mid \text{neg}[b,c] \mid \text{neg}[c,a] \]

The SPiM program

\begin{align*}
\text{new ptn:} & \langle \rangle \quad (\text{* Protein *}) \\
\text{new dk:} & 0.001: \langle \rangle \quad (\text{* Decay rate *}) \\
\text{new neg:} & \langle \rangle, \langle \rangle \quad (\text{* Neg Gate *}) \\
\text{new tInh:} & 0.001: \langle \rangle \quad (\text{* Inhibition rate *}) \\
\text{new tCst:} & 0.1: \langle \rangle \quad (\text{* Constitutive rate *}) \\
\text{(\text{* Protein-Gene interactions *})} \\
\text{new a:1.0:} & \langle \rangle \quad \text{new b:1.0:} \langle \rangle \quad \text{new c:1.0:} \langle \rangle \\
\end{align*}

\begin{verbatim}
( !ptn(p); (p<>;ptn<p>+dk<>;();)
 | !dk()
 | !neg(a,b);
 (a(); (tInh(); neg<a,b>) +
 tCst(); (ptn<b> | neg<a,b>))
 | !tCst<> | !tInh<>

(\text{* The circuit *})
| neg<a,b> | neg<b,c> | neg<c,a> 
\end{verbatim}

A stochastic simulation

\[ r=1.0, \epsilon=0.1, h=0.001, \delta=0.001 \]
Membranes
Brane Calculi

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

nests 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

\[ a.\sigma|\tau = (a.\sigma)|\tau \]

N.B. Restriction (\( vn \)) could be added to both systems and branes. It usually would originate in branes, but would extrude to whole systems.
Brane Reactions (Cartoons)

Exo

Pino

Phago
Phago \( \otimes_n \sigma | \sigma' (P) \circ \otimes_n (\rho). \tau | \tau' (Q) \rightarrow \tau | \tau' (\rho(\sigma | \sigma' (P)) \circ Q) \)

Exo \( \otimes_n \tau | \tau' (\otimes_n \sigma | \sigma' (P) \circ Q) \rightarrow P \circ \sigma | \sigma' | \tau | \tau' (Q) \)

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

N.B.: the parity of nesting of \( P \) and \( Q \) is preserved; this makes the reactions preserve bitonality.
Derivable Reactions (Cartoons)
Ex: Viral Reproduction

Virus → Phago → Cytosol → Endosome → Mate → Exo → Nucleus → Endoplasmic Reticulum → Drip → Vesicle → Bud

RNA → Capsid → Membrane → Envelope protein → Nucleocapsid

Infection → Replication → Progeny

[MBC p.279] annotated
Conclusions

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.

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