Programming-based systems biology

Corrado Priami
outline
Biology needs a theory able to highlight causality and abstract data into knowledge to elucidate the architecture of biological complexity.

*Sidney Brenner*
The grand challenge for all scientific and engineering disciplines in the 21st century is complexity.

Lee Hood
IMPACT CORE COMPUTER SCIENCE AND SYSTEMS BIOLOGY

Quantitative operational descriptions of the mechanistic behavior of biological systems

Computational thinking

Not ambiguous specifications for simulation and analysis tools

Coping with combinatorial explosion of systems description

MOVING BEYOND MATH MODELING: ADDRESS CONCURRENCY AND COMPLEXITY
BIOINFORMATICS IS DIFFERENT

Comparison of strings
Storage of experiment results
Visualization of complex data
Search and analysis of data sets...

Mainly Structural/Static descriptions

MATHEMATICAL AND COMPUTATIONAL BIOLOGY ARE DIFFERENT

Static abstract relationships
Computer assisted solutions
Global pictures of dynamics
OUR DRIVING CHALLENGES

- Interaction
- Emergence
- Partial knowledge
- Ambiguous observations
- Multi-level, multi-scale in space and time
- Causal relationships and context-awareness

LOW-LEVEL LOCAL MECHANISMS AFFECT HIGH-LEVEL GLOBAL BEHAVIOR
A MODELING FORMALISM

Encode info manipulation by bio-systems

A formal framework to reason about bio-systems

Predict biological behavior and identify new hypotheses

Unambiguous description to share knowledge
the development of the appropriate languages to describe information processing in biological systems and the generation of more effective methods to translate biochemical descriptions into the functioning of the logic circuits that underpin biological phenomena

Paul Nurse
THE METAPHOR

- biological entities
- interaction capabilities
- complex/decomplexe dynamics
- processes
- interfaces
- binding/unbinding
- state change


# PI-CALCULUS

<table>
<thead>
<tr>
<th>Processes</th>
<th>Domains, Molecules/Proteins, Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primitives</td>
<td>Join/Split, Conditional Events</td>
</tr>
<tr>
<td></td>
<td>Restriction for Compartments, Membranes, Complexes</td>
</tr>
<tr>
<td>Complex/</td>
<td>Not explicitly modeled</td>
</tr>
<tr>
<td>Decomplex</td>
<td>Interplay between private names and scope size</td>
</tr>
<tr>
<td>Low-level</td>
<td>Reversibility of interactions</td>
</tr>
<tr>
<td>Programming</td>
<td></td>
</tr>
<tr>
<td>Modeling</td>
<td>Emergent behavior must be programmed</td>
</tr>
<tr>
<td>Interaction</td>
<td>Key-lock mechanism</td>
</tr>
<tr>
<td>Implementation</td>
<td>General structural congruence</td>
</tr>
</tbody>
</table>
# BETA-BINDERS

<table>
<thead>
<tr>
<th>Processes</th>
<th>Domains, Molecules/Proteins, Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primitives</td>
<td>Join/Split, Conditional Events</td>
</tr>
<tr>
<td></td>
<td>Restriction for Compartments, Membranes, Complexes</td>
</tr>
<tr>
<td>Complex/Decomplex</td>
<td>Not explicitly modeled</td>
</tr>
<tr>
<td></td>
<td>Interplay between private names and scope size</td>
</tr>
<tr>
<td>Low-level programming</td>
<td>Reversibility of interactions</td>
</tr>
<tr>
<td>Modeling</td>
<td>Emergent behavior must be programmed</td>
</tr>
<tr>
<td>Interaction</td>
<td>Key-lock mechanism</td>
</tr>
<tr>
<td>Implementation</td>
<td>General structural congruence</td>
</tr>
</tbody>
</table>
BLENX PRINCIPLES

- Set of interacting boxes with internal behavior
- Non key-lock interaction mechanism
- Complex description and their Dynamic generation
- Typed and dynamic varying interfaces
- 1-1 correspondence with Biological elements
- Events

// CDP
let CDP_ : bproc = #(y,CDP) [ rep start?().Prog_CDP | Prog_CDP ];
let dCDP_ : bproc = #(y,dCDP) [ rep start?().Prog_CDP | Prog_CDP ];
let dCTP_ : bproc = #(y,dCTP) [ rep start?().Prog_CDP | Prog_CDP ];

// RR
let RR_ : bproc = #(r,RR) [ rep start?().Prog_RR | Prog_RR ];

// CELL
// f1
let Cell : bproc = #(cell,Alive) [ nil ];
when ( Cell : : cell_death_function ) delete(1);
run 100 CDP_ || 100 dFdC_ || 1000 CDA_ || 1000 RR_ || 100 Cell || 100 DNA_healthy
// Tree.prog
[ steps = 10, delta = 10 ]
<< BASERATE:inf, HIDE:inf, UNHIDE:inf, CHANGE:inf >>

// Initiator Definition
let Initiator : bproc = #(!out,I) [ !out!().out!(root).nil ];

// Node Definition
let p1 : pproc = !out1?().out2?().out1!(node).out2!(node).nil ;
let p2 : pproc = !out1?().out2?().inp!().inp?(m).out1!(m).out!(m).nil ;
let nodeP : pproc =
  inp!().inp?(t).( t!() | (node?().unhide(out1).unhide(out2).p2 + root?().unhide(out1).unhide(out2).p1 ) );
let Node : bproc = #(inp,IN),#h(out1,ONE),#h(out2,TWO)
  [ nodeP ];

// Init
run 2 Initiator || 10 Node

// Tree.types
{ I, IN, ONE, TWO }
%%
{ (I,IN,100,0,inf), (ONE,IN,1,0,inf), (TWO,IN,1,0,inf) }

a static list of the entities of the initial configuration
+
their amount and specificity for interaction.

ENABLING LIBRARY-BASED MODELING

Further examples and downloads: http://www.cosbi.eu/index.php/research/prototypes/beta-wb
**BLENX: MONOMOLECULAR ACTIONS**

```
```

```
P₁ | P₂{z/x} | P₃ | P
```

```
unhide(y).P₁ | P₂{z/x} | hide(y).P₃ | delay(2).P
```

```
P₁ | P₂{z/x} | P₃ | delay(2).P
```

```
unhide(y).P₁ | P₂{z/x} | P₃ | delay(2).P
```

```
P₁ | P₂{z/x} | P₃ | delay(2).P
```

---

```
effect(x,T) | effect(y,U) | effect(z,V)
```

```
effect(x,T) | effect(y,U) | effect(z,V)
```

```
effect(x,T) | effect(y,U) | effect(z,V)
```

```
effect(x,T) | effect(y,U) | effect(z,V)
```

```
effect(x,T) | effect(y,U) | effect(z,V)
```

---

```
```

```
P₁ | P₂{z/x} | P₃ | P
```

```
unhide(y).P₁ | P₂{z/x} | hide(y).P₃ | delay(2).P
```

```
P₁ | P₂{z/x} | P₃ | delay(2).P
```

```
unhide(y).P₁ | P₂{z/x} | P₃ | delay(2).P
```

```
P₁ | P₂{z/x} | P₃ | delay(2).P
```
BLENX: MONOMOLECULAR ACTIONS
BLENX: MONOMOLECULAR ACTIONS

\[ \text{die}(1). P \mid \text{ch}(x, D). P + \text{delay}(2). P \]

\[ \text{die}(1). P \mid P \]
BLENX: BIMOLECULAR ACTIONS

\[
x?w.P_1 | y?k.P_2 | \text{hide}(y).P_3
\]

\[
z!v.S_1 | y?k.S_2 | y!w.S_3
\]

\{ T, U, V, ... \}
\%
\{
  ...
  (U,V,r),
  (T,V,r1),
  ...
\}
BLENX: BIMOLECULAR ACTIONS

\[
\begin{align*}
x & : T \\
y & : U \\
x & ? w . P_1 | \ y & ? k . P_2 | \text{hide}(y) . P_3 \\
z & : V \\
z & ! v . S_1 | \ y & ? k . S_2 | \ y & ! w . S_3 \\
\end{align*}
\]

\[
\{ T, U, V, \ldots \} \\
\% \%
\{
\ldots
(U,V,r),
(T,V,r1),
\ldots
\}
\]
BLENX: BIMOLECULAR ACTIONS

\{ T, U, V, \ldots \}
\%
\{
\ldots
(U,V,r1),
(T,V,r2),
\ldots
\}

x:T  y:U
A

z:V
B

x?w.P_1 \mid y?k.P_2 \mid \text{hide}(y).P_3

z!v.S_1 \mid y?k.S_2 \mid y!w.S_3
BLENX: EVENTS

\[
\text{when } (Q :: 10) \text{ new } (1);
\]

\[
\text{when } (R, Q :: f) \text{ join } (P);
\]

\[
\text{let } f = |S|*\sqrt{|Q|}/k
\]
Example
Predator - Prey
let Predator : bproc = #((x,Hunt)) nil; 
when (PredatorRep :: inf) split (Predator,Predator);

let PreyRep : bproc = #((x,Hunt),(y,Life)) nil; 
when (PreyRep :: inf) split (Prey,Prey);

run 1000 Predator || 1000 Prey || 1 Nature
When (PredatorRep::inf) split (Predator,Predator)

\{ Hunt, Life \}
\%
\{
  (Hunt,Hunt,10.0),
  (Life,Life,0.01)
\}

\[ x:\text{Hunt} \]
\[ y:\text{Life} \]
\[ \text{die}() + x()(.\text{nil}) \]
\[ x?().\text{die}(\text{inf}) + y()(.\text{nil}) \]
PREDATOR - PREY

[Graphs showing predator-prey dynamics]
END Example
Predator - Prey
BLENX: COMPLEXES

{ T, U, V, Z, ... }

{ (U, Z, 3.5),
  (U, V, 1.5) }

Q R S

Q R S

x:T   y:U   z:V   y:Z

%%
BLENX: COMPLEXES

{x:T, y:U, z:V, y:Z}

{ T, U, V, Z, ... }

{ (U, Z, 3.5),
  (U, V, 1.5, 2.5, 10) }
BLENX: STRUCTURES

DYNAMICALLY CREATING STRUCTURES

\{ A, B \}
\%
\%
\{ (A,B,1,1,0) \}
DYNAMICALLY CREATING STRUCTURES

{ A, B }
%%
{ (A,B,1,1,0) }

X:A
Y:B

X:A
Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B

X:A

Y:B
DYNAMICALLY CREATING STRUCTURES

\{ A, B \}
\%
\{ (A, B, 1, 1, 0) \}
Example
SELF assembly of trees
SELF ASSEMBLY OF TREES

// Tree.prog
[ steps = 10, delta = 10 ]
<< BASE RATE:inf, HIDE:inf, UNHIDE:inf >>

// Initiator Definition
let Initiator : bproc = #(out,l) [ out?().out!(root).nil ];

// Node Definition
let p1 : pproc = !(out1?().out2?().out1!(node).out2!(node).nil) ;
let p2 : pproc = !(out1?().out2?().inp!().inp?(m).out1!(m).out2!(m).nil) ;
let nodeP : pproc =
  inp!().inp?(t).( t!() | (node?().unhide(out1).unhide(out2).p2 +
  root?().unhide(out1).unhide(out2).p1 ) );
let Node : bproc = #(inp,IN),#h(out1,ONE),#h(out2,TWO) [
  nodeP ];

// Init
run 2 Initiator || 10 Node

// Tree.types
{ I, IN, ONE, TWO }
%
{
(I,I,100,0,inf),
(ONE,I,1,0,inf),
(TWO,I,1,0,inf)
}
SELF ASSEMBLY OF TREES
SELF ASSEMBLY OF TREES

```
    Initiator1
      x1
      x1IN
        Node1
          y:ONE
          y:TWO
          x:IN
          Node
            y:ONE
            y:TWO
            x:IN
            Root2
              y:ONE
              y:TWO
              x:IN
              Initiator1
                x1
                x1IN
                  Root2
                    y:ONE
                    y:TWO
                    x:IN
                    Initiator1
                      x1
                      x1IN
                        Root
                          y:ONE
                          y:TWO
                          x:IN
                          Child
                            y:ONE
                            y:TWO
                            x:IN
                            Child
```

END Example
SELF assembly of trees
BLENX
RECAP MAIN CONCEPTS
RECAP BLENX

• Simulation and analysis (causality, logical properties)

• Scalability, modularity, compositionality

• Different levels of abstraction and refinement

• Easy models, easy libraries due to combinatorial effects ruled out at modeling level

• Executable vs. solvable specifications, modules vs. variables, dynamic relations vs. static relations
Usability of the framework
COSBI WORKING FLOW

DEFINE THE QUESTION

GATHER AND ORGANIZE RELEVANT DATA

EXPERIMENTAL DATA

NETWORKS DATABASE

INFERENCE

FORM HYPOTHESIS

MODELLING

COSBILAB Model

PERFORM EXPERIMENT, COLLECT DATA

SIMULATION

Simulator (BlenX)

DRAW CONCLUSIONS, TEST HYPOTHESIS

ANALYZE DATA

VISUALISATION

COSBILAB Graph

COSBILAB Plot

GRAPH

PLOT
MODELING WORKFLOW

- Components represent the biological entities acting in the scenario;

- Interaction rules are specified in an **intelligible, narrative language**;

- Kinetic parameters are summarized in a single page from where users can perform multiple **in silico experiments**;

- An integrated, stochastic simulation engine (based on the Gillespie algorithm) is included;

- Models are translated to BlenX, a lower level modeling language for biological systems.
Programming without programming
COSBILAB MODEL

High-level, tabular interface for BlenX

Hiding programming details from BlenX users

Strongly inspired to a narrative language


COSBI MODELING PHILOSOPHY

Intelligible models:

• Understandable to non-expert modelers
• Easy to write, modify and reuse
• Manage combinatorial complexity of dynamics
INTELLIGIBILITY

< Obscure syntax
< No bio keywords

Narrative Language

< Large size
< Complex mathematics

Minimize required modeling information
EGFR MOLECULE
$2^9$ configuration of EGFR that is $512$ different states
EGFR HOMODIMERS

$2^{18}$ configuration of EGFR Homodimers that is more than 250,000 different states
EGFR MOLECULE

9 sites for phosphorylation

EGF

site for ligand binding

site for dimerization

EGFR
INTERACTION BETWEEN EGFR AND EGF

EGF

EGFR

P

P

P

P
INTERACTION BETWEEN EGFR AND EGF
INTERACTION BETWEEN EGFR AND EGFR
SITE’S DYNAMIC FOR STATE CHANGE

Site’s Dynamic for state change
SITE’S DYNAMIC FOR STATE CHANGE

EGFR

EGF

P-
P-
P-
P-
P-
P
SITE’S DYNAMIC FOR STATE CHANGE

Site’s Dynamic for state change

EGFR

EGF
DISCRETE STOCHASTIC APPROACH
DISCRETE STOCHASTIC APPROACH

• Molecular interaction is discrete
• Receive more detailed information
• Able to produce more accurate predictions
DISCRETE STOCHASTIC APPROACH EGFR EXAMPLE

Average EGFR Phosphorylation

(time)
DISCRETE STOCHASTIC APPROACH EGFR EXAMPLE

Distribution at time t=1

- Below Average: 0.439
- Above Average: 0.563
COSBILAB Model

The gemcitabine example

P. Lecca, O. Kahramanogullari, D. Morpurgo, C. Priami, R. Soo. Modelling the tumor shrinkage pharmacodynamics with BlenX, 1st IEEE International Conference on Computational Advances in Bio and medical Sciences (ICCABS), 2011

An algorithmic pharmacodynamics model for the time course of tumor shrinkage by gemcitabine in non-small cell lung cancer patients with the following events:

- part of the injected gemcitabine degrades;

- the gemcitabine interacts with the tumor without any effect;

- The gemcitabine interacts with the tumor, is consumed and the tumor cell is killed;

- tumor grows.

- new doses of drug are regularly injected
Gemcitabine (2’-2’-difluorodeoxycytidine, dFdC) is a nucleoside analog used in oncology to block DNA replication in tumor cells;

- Gemcitabine is transported from plasma into the cell and then it is subjected to deamination and/or multiple phosphorylation leading to its active triphosphate (dFdCTP and dFdUTP) metabolites;

- As an example, a simplified version of gemcitabine metabolic network will be modeled.
COMPONENTS

Gemcitabine (dFdC) has sites for deamination, phosphorylation, and inhibition of dCMPD enzyme.
COMPONENTS

Gemcitabine (dFdC) has sites for deamination, phosphorylation, and inhibition of dCMPD enzyme.

BlenX code:

let dFdCout: bproc = #(s1, c_dFdCout1), #(s2, c_dFdCout2) [ p_main | rep start_p_main?().p_main ];
COMPONENTS

- dFdC
  - nh, ph1, ph2, ph3, bin 1

Component Site Definition:
- nh: Deamination
- ph1: First phosphorylation
- ph2: Second phosphorylation
- ph3: Third phosphorylation
- bin: Inhibition of dCMPD
COMPARTMENTS

The extracellular medium, the cellular membrane, and the intracellular medium (cytosol) are considered;
### COMPARTMENTS

<table>
<thead>
<tr>
<th>Options</th>
<th>Name</th>
<th>Type</th>
<th>Enclosed in</th>
<th>Volume</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>System</td>
<td>Compartment</td>
<td>System</td>
<td>System_Volume</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Membrane</td>
<td>Membrane</td>
<td>System</td>
<td>Membrane_Volume</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cytosol</td>
<td>Compartment</td>
<td>Membrane</td>
<td>Cytosol_Volume</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Kinetic Law</th>
<th>Name</th>
<th>Value</th>
<th>Unit of Measure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume</td>
<td></td>
<td>System_Volume</td>
<td>50</td>
<td>fl (10^-9 mm^3)</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td></td>
<td>Membrane_Volume</td>
<td>0.1</td>
<td>fl (10^-9 mm^3)</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td></td>
<td>Cytosol_Volume</td>
<td>17</td>
<td>fl (10^-9 mm^3)</td>
<td></td>
</tr>
</tbody>
</table>
TRANSLocations

Gemcitabine moves from plasma into the cell passing through the cellular membrane;
Gemcitabine moves from plasma into the cell passing through the cellular membrane;

**BlenX code:**

```plaintext
let p_main : pproc =
    if (s1, c_dFdCout1) then ch(rate(r_dFdCout_in), s1, c_dFdC1).ch(s2, c_dFdC2).start_p_main!()
    endif + ...

let dFdCout: bproc = #(s1, c_dFdCout1), #(s2, c_dFdCout2) [ p_main | rep start_p_main?().p_main ];
```
### Translocations

<table>
<thead>
<tr>
<th>Components</th>
<th>Complexes</th>
<th>Global Dynamics</th>
<th>Bimolecular Dynamics</th>
<th>Binding Dynamics</th>
<th>Translocations</th>
<th>Compartments</th>
<th>Parameters</th>
<th>Initial State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action</td>
<td>(Parameter) Rate</td>
<td>(Parameter) Back Rate</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dFdC_init moves from System to Cytoplasm</td>
<td>r_dFdC_out_in</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Volumes

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Value</th>
<th>Unit of Measure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume</td>
<td>System_Volume</td>
<td>50</td>
<td>fl (10^-9 mm³)</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>Membrane_Volume</td>
<td>0.1</td>
<td>fl (10^-9 mm³)</td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>Cytosol_Volume</td>
<td>17</td>
<td>fl (10^-9 mm³)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdC_out_in</td>
<td>9.97</td>
<td>1/hours</td>
</tr>
</tbody>
</table>
BIMOLECULAR DYNAMICS

- Once inside the cytoplasm, dFdC is phosphorylated by deoxycytidine kinase (dCK);

- Built-in kinetic laws are available (Mass-Action, Michaelis-Menten, Hill) as well as User Defined;
BIMOLECULAR DYNAMICS

• Once inside the cytoplasm, dFdC is phosphorylated by deoxycytidine kinase (dCK);

• Built-in kinetic laws are available (Mass-Action, Michaelis-Menten, Hill) as well as User Defined;

**BlenX code:**

```blenx
let p_main : pproc = ... + if (s1, c_dFdC1) then s1().ch(s1, c_dFdCMP1).ch(s2, c_dFdCMP2).start_p_main!() +
  endif + ...

let dFdC : bproc = #(s1, c_dFdC1), #(s2, c_dFdC2) [ p_main | rep start_p_main?().p_main ];
let dCK : bproc = #(s, c_dCK) [ s!().startdCK!() | rep startdCK?().s!().startdCK!() ];

Affinity: (c_dCK, c_dFdC1, rate(r_dFdC_dCK))
```
BIMOLECULAR DYNAMICS

**Action**
- dCK makes dFdC Phosphorylated on ph1
- r_dFdC_dCK

**Binding Dynamics**
- (Parameter) Rate
- Active Condition
- Passive Condition

**Components**
- Type
- Kinetic Law
- Name
- Value
- Unit of Measure
- Description

**Parameters**
- Volume
- Membrane Volume
- Cytosol Volume
- Rate
- Mass Action
- r_dFdC_dCK
- 0.01
- 1/(Units*hours)
Inhibition mechanisms can be modeled in different ways, in this case we considered an inhibitory complexation between dFdCTP and dCMPD;

Binding conditions are specified to identify the target configuration.
• Inhibition mechanisms can be modeled in different ways, in this case we considered an inhibitory complexation between dFdCTP and dCMPD;

• Binding conditions are specified to identify the target configuration.

BlenX code:

```blenx
let dFdCTP : bproc = #(s1, c_dFdCTP1), #(s2, c_dFdCTP2) [ p_main | rep start_p_main?().p_main ];
let dCMPD : bproc = #(s, c_dCMPD) [ s!().startdCMPD() | rep startdCMPD?().s!().startdCMPD() ];

Affinity: (c_dFdCTP1, c_dCMPD, rate(r_dFdCTP_dCMPD_bin), rate(r_dFdCTP_dCMPD_unb), 0)
```
# BINDING DYNAMICS

<table>
<thead>
<tr>
<th>Components</th>
<th>Complexes</th>
<th>Global Dynamics</th>
<th>Bimolecular Dynamics</th>
<th>Binding Dynamics</th>
<th>Translocations</th>
<th>Compartments</th>
<th>Parameters</th>
<th>Initial State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component 1</td>
<td>Site 1</td>
<td>Component 2</td>
<td>Site 2</td>
<td>(Parameter) Binding Rate</td>
<td>(Parameter) Unbinding Rate</td>
<td>Condition</td>
<td>Compo</td>
<td></td>
</tr>
</tbody>
</table>

**Click here to add new item**

<table>
<thead>
<tr>
<th>Component 1</th>
<th>Component 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binding Condition for Component 1</td>
<td>Binding Condition for Component 2</td>
</tr>
<tr>
<td>(nh is Aminated and ph3 is Phosphorylated)</td>
<td></td>
</tr>
<tr>
<td>Unbinding Condition for Component 1</td>
<td>Unbinding Condition for Component 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Components</th>
<th>Complexes</th>
<th>Global Dynamics</th>
<th>Bimolecular Dynamics</th>
<th>Binding Dynamics</th>
<th>Translocations</th>
<th>Compartments</th>
<th>Parameters</th>
<th>Initial State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Kinetic Law</td>
<td>Name</td>
<td>Value</td>
<td>Unit of Measure</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Click here to add new item**

<table>
<thead>
<tr>
<th>Type</th>
<th>Kinetic Law</th>
<th>Name</th>
<th>Value</th>
<th>Unit of Measure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume</td>
<td>System_Volume</td>
<td>50</td>
<td>(10^9) mm(^3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>Membrane_Volume</td>
<td>0.1</td>
<td>(10^9) mm(^3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Volume</td>
<td>Cytosol_Volume</td>
<td>17</td>
<td>(10^9) mm(^3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdC_CDA</td>
<td>0.000005</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCMP_dCMPD</td>
<td>0.00005</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdU_dCK</td>
<td>0.01</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdUMP_NMPK</td>
<td>0.001</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdUDP_NDPK</td>
<td>0.002</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCTP_dCMPD_bin</td>
<td>1E-02</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCTP_dCMPD_unb</td>
<td>1E-08</td>
<td>1/(Units(\times)hours)</td>
<td></td>
</tr>
</tbody>
</table>
Gemcitabine triphosphate (dFdCTP and dFdUTP) is incorporated into DNA causing chain termination.
Gemcitabine triphosphate (dFdC and dFdU) is incorporated into DNA causing chain termination.

**BlenX code:**

```latex
let dFdU : bproc = #(s1, c_dFdU), #(s2, c_dFdU) [ p_main | rep start_p_main?().p_main ];
when (dFdU : : rate(r_dFdU_DNA)) split(DNA, Nil);
```
### GLOBAL DYNAMICS

<table>
<thead>
<tr>
<th>Action</th>
<th>(Parameter)</th>
<th>Rate</th>
<th>Condition</th>
<th>Compartment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>substitute dFdCTP with DNA</td>
<td>r_dFdCTP_DNA</td>
<td>Cytosol</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
> | substitute dFdUTP with DNA | **r_dFdUTP_DNA** | Cytosol | | |

<table>
<thead>
<tr>
<th>Component</th>
<th>Volume</th>
<th>Rate</th>
<th>Compartment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Membrane_Volume</td>
<td>0.1 fl (10^-9 mm^3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cytosol_Volume</td>
<td>17 fl (10^-9 mm^3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdCout_in</td>
<td>9.97 1/hours</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdC_CDA</td>
<td>0.000005 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdCMP_dCMPD</td>
<td>0.00005 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdU_dCK</td>
<td>0.01 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdUMP_NMPK</td>
<td>0.001 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdUDP_NDPK</td>
<td>0.002 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdCTP_dCMPD_bin</td>
<td>1E-02 1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r_dFdCTP_dCMPD_unb</td>
<td>1E-08 1/hours</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>r_dFdUTP_DNA</strong></td>
<td><strong>0.001 1/hours</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Diagram:**
- CDA
- dCK
- dFdC-MP
- dCMPD
- dFdU-MP
- NMPK
- dFdC-DP
- dFdC-DP
- dFdC-TP
- DNA
- Cytoplasm
- Extracellular
- Cell Membrane
INITIAL STATE
INITIAL STATE

BlenX code:
run 10000 dFdCout || 100 dCK || 100 NMPK || 100 NDPK || 100 CDA || 100 dCMPD
### INITIAL STATE

<table>
<thead>
<tr>
<th>Component/Complex Configuration</th>
<th>Type</th>
<th>(Parameter) Quantity</th>
<th>Compartment</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dFdC_init</td>
<td>Component</td>
<td>q_dFdC_out</td>
<td>System</td>
<td></td>
</tr>
<tr>
<td>dCK_init</td>
<td>Component</td>
<td>q_enzyme</td>
<td>Cytosol</td>
<td></td>
</tr>
<tr>
<td>NMPK_init</td>
<td>Component</td>
<td>q_enzyme</td>
<td>Cytosol</td>
<td></td>
</tr>
<tr>
<td>NDPK_init</td>
<td>Component</td>
<td>q_enzyme</td>
<td>Cytosol</td>
<td></td>
</tr>
<tr>
<td>CDA_init</td>
<td>Component</td>
<td>q_enzyme</td>
<td>Cytosol</td>
<td></td>
</tr>
<tr>
<td>dCMPD_init</td>
<td>Component</td>
<td>q_enzyme</td>
<td>Cytosol</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Components</th>
<th>Complexes</th>
<th>Global Dynamics</th>
<th>Bimolecular Dynamics</th>
<th>Binding Dynamics</th>
<th>Translocations</th>
<th>Compartments</th>
<th>Parameters</th>
<th>Initial State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Volume</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>System_Volume 50 fl (10^-9 mm^3)</td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCTP_dCMPD_bin</td>
<td>1E-02</td>
<td>1/(Units*hours)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCTP_dCMPD_unb</td>
<td>1E-08</td>
<td>1/Hours</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdCTP_DNA</td>
<td>0.05</td>
<td>1/Hours</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rate</td>
<td>Mass Action</td>
<td>r_dFdUTP_DNA</td>
<td>0.001</td>
<td>1/Hours</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td>q_dFdC_out</td>
<td>10000</td>
<td>Units</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td>q_enzyme</td>
<td>100</td>
<td>Units</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
COSBI WORKING FLOW

DEFINE THE QUESTION

GATHER AND ORGANIZE RELEVANT DATA

EXPERIMENTAL DATA

NETWORKS DATABASE

INFERERENCE

FORM HYPOTHESIS

MODELING COSBILAB Model

PERFORM EXPERIMENT, COLLECT DATA

SIMULATION Simulator (BlenX)

ANALYZE DATA

VISUALISATION COSBILAB Graph
COSBILAB Plot

DRAW CONCLUSIONS, TEST HYPOTHESIS

HYPOTHESIS CONSISTENT?
VISUALIZE RESULTS
SIMULATION RESULTS PLOT

Intracellular concentration versus time
SENSITIVITY ANALYSIS

Accumulation of dFdCTP w.r.t. dCTP inhibition on dCK
COSBILAB MODEL: MODELING & SIMULATING COMPLEX SYSTEMS

Create models via tabular interface. No expertise in programming nor math needed.

Easily manage experiments: knock-down genes, change rates, add a “virtual” drug.

Share models & results with colleagues, wherever they are.

Allow life scientist analyze models.

In silico science: adding knowledge, saving time.

Accelerate discovery process, add value to your work.
Spatial simulation

SPACe AND DIFFUSION

Reaction-diffusion simulator at the mesoscopic interaction scale.

Space discretised variant of the Gillespie SSA.

Dynamically state-dependent diffusion coefficient.
BICOID SIMULATION

Experimental

Simulated

40 min
COSBI WORKING FLOW
Define the problem you want to address

Clinical - omics experiments - Ontological

DATA MINING
Organize, integrate and analyze data

KNOWLEDGE EXTRACTION
Knowledge inference


THE SYSTEM AND THE DATA

Experimentally observed time course data of the reactants concentrations.
From these behaviors we want to estimate

\[ \theta_1, \theta_2, \theta_3, \theta_4, \theta_5 \]
COMPUTATIONAL SUPPORT: KINFER

- Loading time series of species concentrations
- Estimation of initial guesses for parameters

Example of a model and its parameters in KINFER.
Network inference

NETWORK INFERENCE

Lagged correlations

Distance clustering

Network calibration
IDENTIFICATION OF METABOLIC NETWORKS


CANCER RESEARCH 52, 533-539, February 1, 1992
MODEL PARAMETERS

The parameters are the specific speed of the following events:

• Degradation rate of gemcitabine (k1)

• Drug inefficiency rate (k2)

• Drug efficacy (i.e. constant-cell-kill rate) (k3)

• Tumor growth rate (k4)

• k5: injection is modeled as single instantaneous event having infinite rate constant
METHODS AND DATA

• We used KInfer on the time courses of tumor shrinkage and gemcitabine dosage on 56 patients to infer the parameters of the model.

• Patients have been divided with respect to sex, age and smoke history.

• The experimental curves of tumor shrinkage have been provided by R. A. Soo of the Department of Hematology-Oncology, National University Hospital, Singapore.
DRUG EFFICACY AND CLASSES OF PATIENTS

• We found that the drug efficacy (k3) is correlated to the patient sex, smoke history, and tumor stage.

• No correlation seems to exist between drug efficacy and patient age.
EFFICACY AND GENDER

Normalized between 0 and 1
EFFICACY AND SMOKE HISTORY

Drug efficac (cm/micro-M)

Current smokers
Non smokers
Smoke history
Ex-smokers

Normalized between 0 and 1
EFFICACY AND TUMOR STAGE

![Bar chart showing drug efficacy in relation to tumor stage. Stage 1 has lower efficacy compared to Stage 2.](Image)
DRUG EFFICACY VS PATIENT AGE
NETWORK Analysis and visualization


Download: http://www.cosbi.eu/index.php/research/prototypes/graph
INPUT FORMATS
ALGORITHMS FOR NETWORK ANALYSIS

**Centrality:**
- Page Rank
- Degree Centrality coefficient
- Closeness Centrality coefficient

**Distances:**
- Diameter
- Compactness Index
- Center of Gravity
- Average Distance

**Generate:**
- Watts-Strogatz Random Graph
- Erdos-Reyni Random Graph
- Barabasi-Albert Random Graph

**Shortest Path:**
- Shortest Path
- Shortest Path Matrix
- Dijkstra Shortest Path
- Bellman-Ford Shortest Path

**Vertex Coefficients:**
- Status, Contrastatus, Netstatus
- K-Index
- Degree
- Clustering Coefficients
LAYOUTS: FRUCHTERMAN REINGOLD
PROPERTY INSPECTOR
SEARCH: PAGE RANK > 1.4
SHORTEST PATH TOOL
NODE SIZE REFLECTING A NODE PROPERTY
PICTURES AS NODE ATTRIBUTES
VITAMINS AND OBESITY

Select the micronutrients

Search Networks & Pathways

Search & Analyse Protein – Protein Interaction Database

Select protein At Key Nodes

Map to QTL of Obesity/diabetes Phenotype

Select genes involved and proteins codified

CANDIDATE GENES
VITAMINS AND OBESITY

Vitamins

- Ascorbic acid
- B12
- Biotin
- Folate
- Niacin
- Pantothenic
- Pyridoxal
- Pyridoxine
- Riboflavin
- Thiamine
- Retinol
- VitD2
- VitE
- VitK
- VitD3

Metabolic & Network

- 200 proteins

Protein-Protein Interactions

- 1647 proteins along 2672 edges

Candidates @ QTLs

- GFI1
- NCOR1
- NFKB1
- PRKCA
- PPP2CA
- PPP2CB
- VDR
- RXRA
- SHBG
- SMAD3
- SMARCA4
- New Candidate
- MED1
- New Candidate
CONCLUSIONS