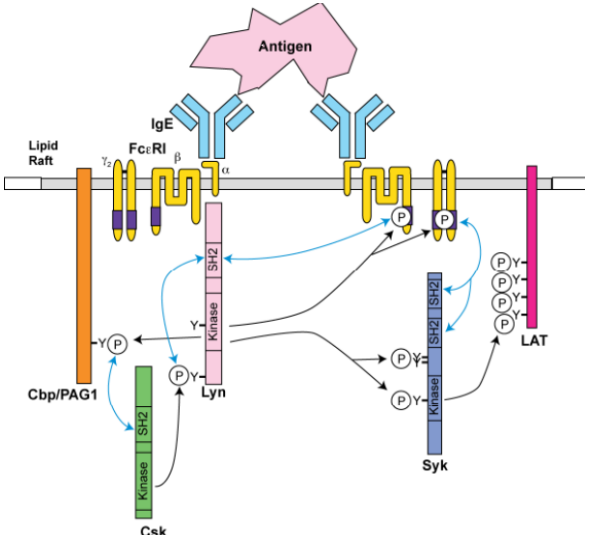
Introduction to rule-based modeling with BioNetGen and RuleBender

Jim Faeder
Department of Computational and Systems Biology
University of Pittsburgh School of Medicine



Presenters:

Cihan Kaya
Jose Juan Tapia
John Sekar
Sanjana Gupta

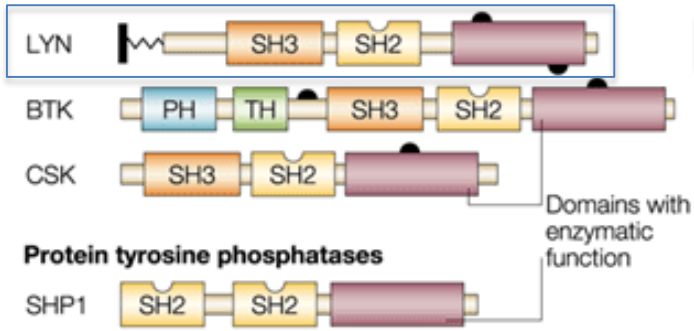
For additional information and references see
http://bionetgen.org/index.php/Tutorials

Agenda for Day 2

8:30-9:30 - Intro to Rule-Based Modeling (**Jim**)
9:30-9:45 - Break
9:45-11:00 - Basic models in RuleBender (**Jim**):
Lotka-Volterra, Polarization, Michaelis-Menten, CaMKII
11:00-11:30 - Modeling DA neurons (**Cihan**)
11:30-12:00 - Atomizer demo and models (**Jose**)
12:00-1:00 - Lunch
1:00-1:30 - Visualization demo (**John**)
1:30-2:30 - Clustering and confinement models (**Sanjana**)
2:30-5:00 - Projects and open demos and discussions

What is Rule-based Modeling (RBM)?

Molecules are modeled as *structured objects*

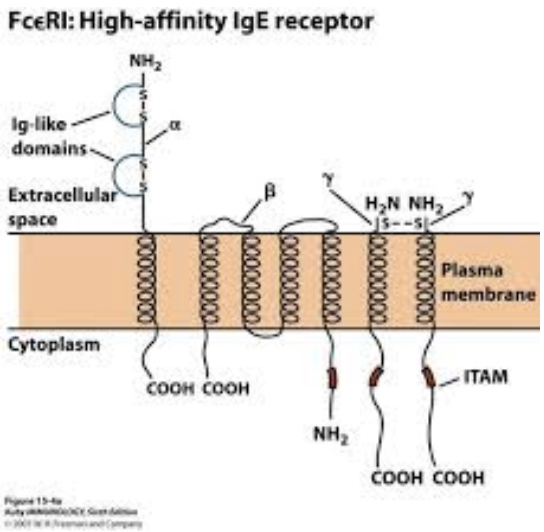



```
LYN(SH3,SH2,kin,Y397~0~P,Y508~0~P)
```

Molecule

Components

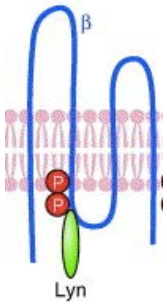
States




```
FceRI(a_Ig,b_Y218~0~P,g_ITAM~0~P)
```

What is Rule-based Modeling (RBM)?

Rules define the interactions of molecules



"Lyn SH2 domain binds to phosphorylated Tyr 218 on the β subunit of FcεRI "

Reactants **Products** **Rate Law**

```
LYN(SH2) + Rec(bY_218~P) <-> LYN(SH2!1).FceRI(bY_218~P!1) kpL, kmL
```

bond

Center – elements modified by the action of the rule

Context – elements required for reaction to occur but not modified

"Don't write don't care" – elements not mentioned may be in any state

➔ *One rule can generate reactions involving many different species*

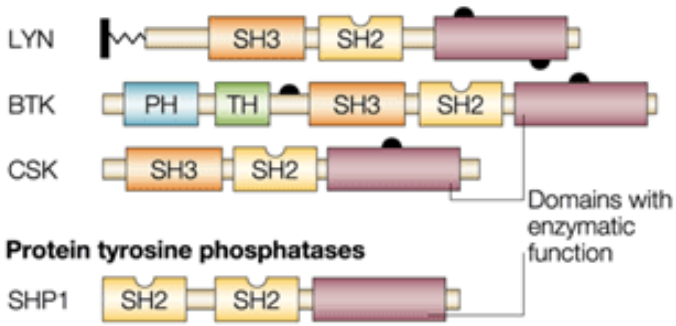
Reaction rate determined by **Mass Action kinetics**

rate forward = kpL*[Lyn(SH2)]*[Rec(bY_218~P)]

↑ rate per reactant set (instance rate)

Why was RBM developed?

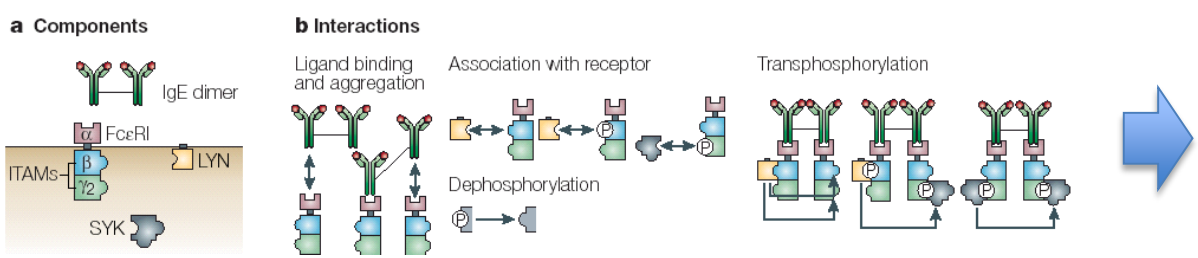
- Proteins are multi-functional

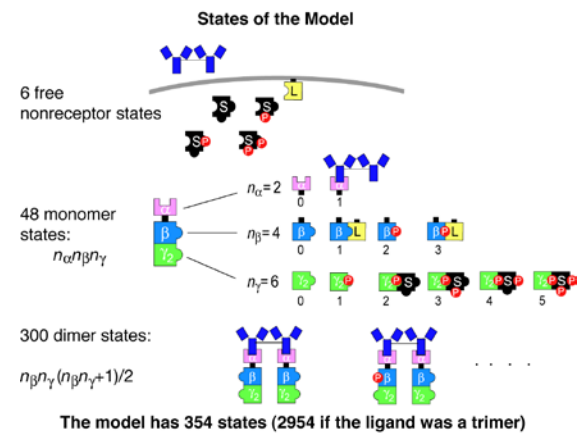


multiple sites of binding

multiple sites of posttranslational modification

- Representing their known interactions requires handling of *combinatorial complexity*



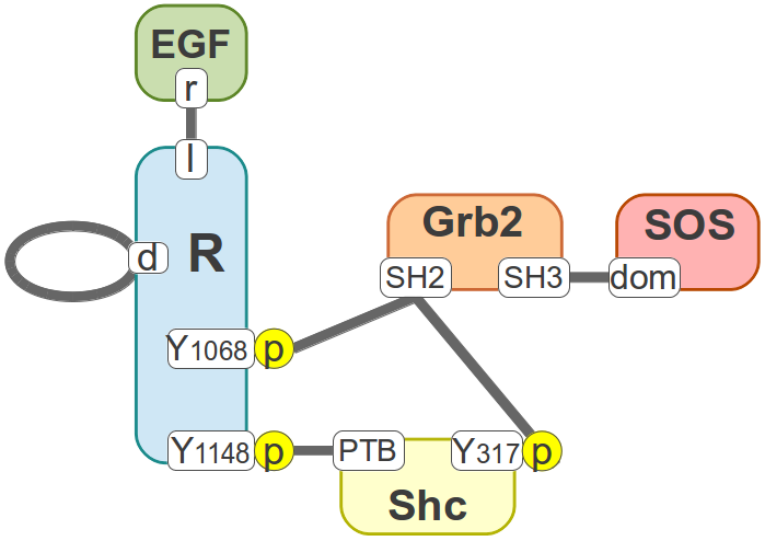


Small number of rules

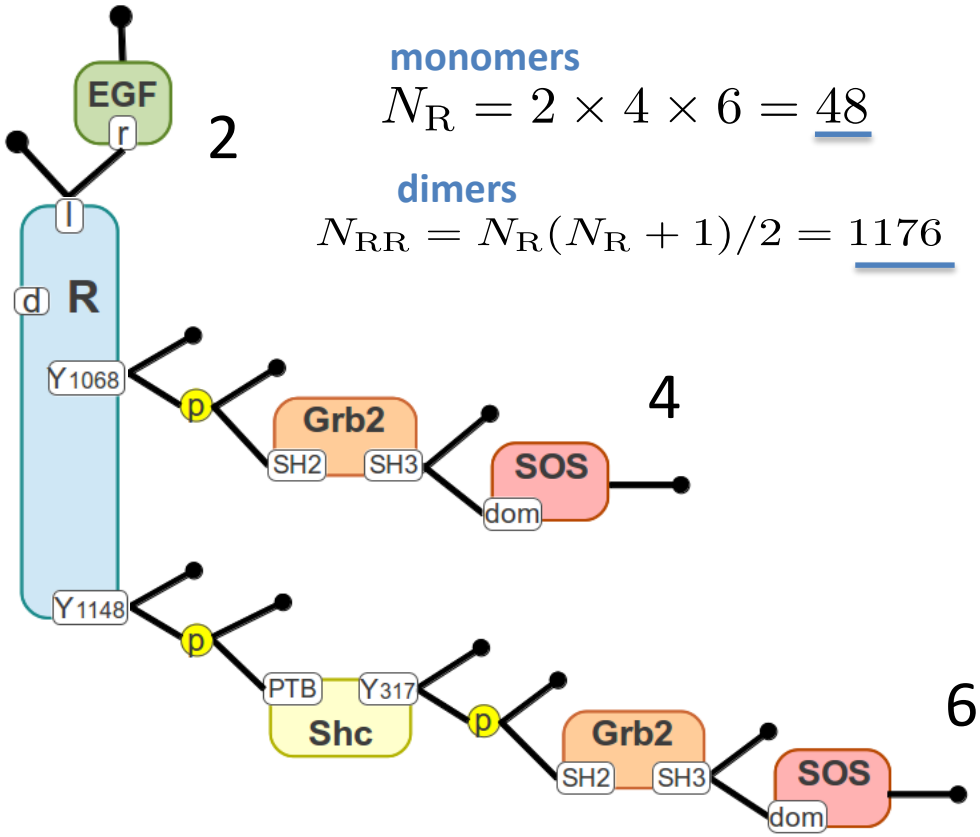
Small number of components and interactions ➔huge number of possible species and reactions

Combinatorial Complexity in Biochemical Interactions

Contact Map for molecules involved in EGFR signaling



Enumeration of receptor-containing species



monomers

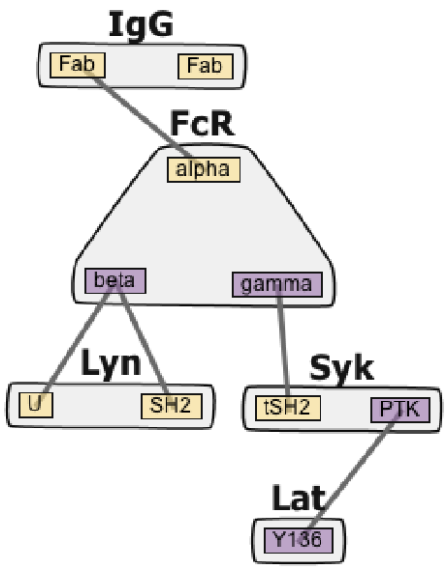
$$N_{\mathrm{R}} = 2 \times 4 \times 6 = \underline{48}$$

dimers

$$N_{\mathrm{RR}} = N_{\mathrm{R}}(N_{\mathrm{R}} + 1)/2 = \underline{1176}$$

Why use RBM?

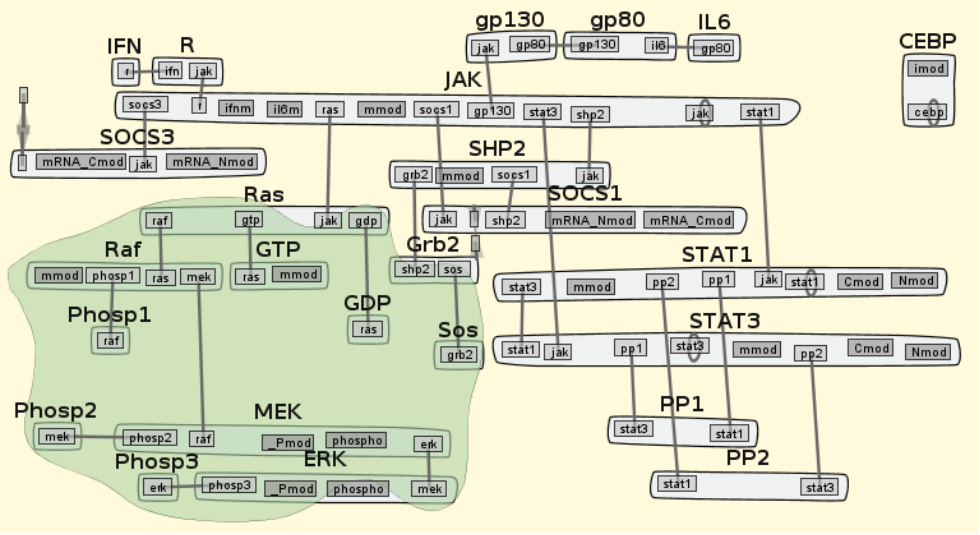
- *Concise and precise* representation of biochemical knowledge
  - rules are simple (less context) when interactions are modular
- *Flexible* with respect to simulation method
  - Deterministic / stochastic
  - Well-mixed / compartmental / spatial
- Structures and rules are *reusable*
  - Rule libraries
- Compact visual representation
  - Contact map and beyond



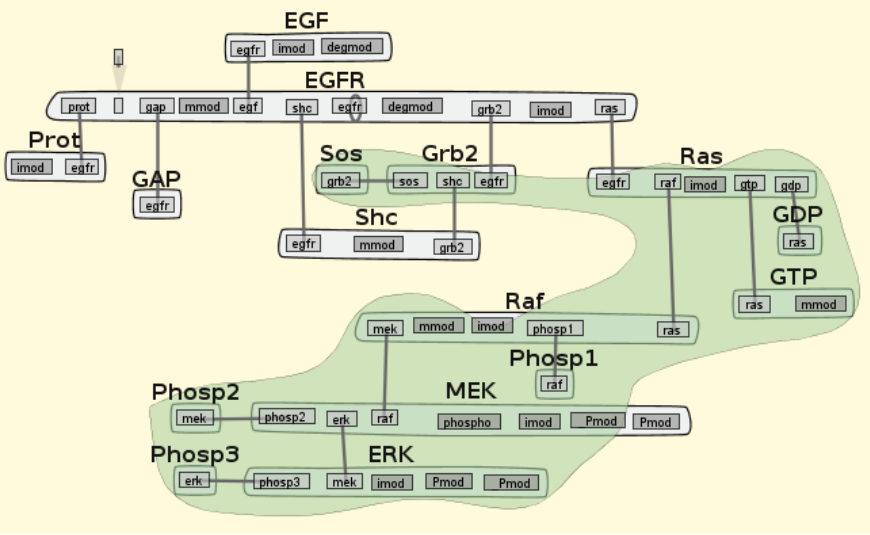
Contact map

Model Comparison

Generated by MOSBIE, a model exploration system within RuleBender



BioModels 543



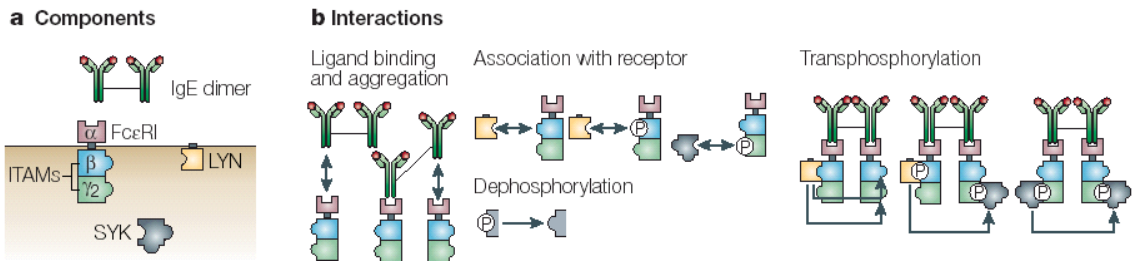
BioModels 19

Some Rule-Based Modeling Tools

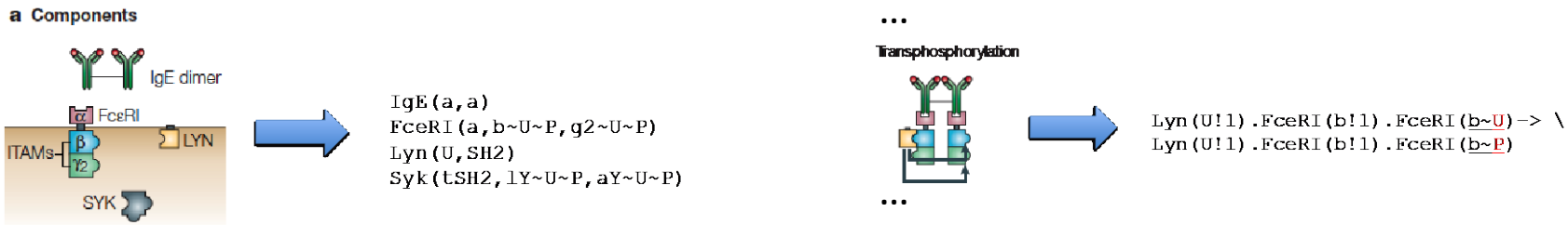
- BioNetGen & relatives
  - RuleBender
  - NFsim
  - BioNetGen@VirtualCell
- Kappa & relatives
- ML-rules
- Simmune
- pySB
- SBML Level 3 Multi

Rule-Based Modeling protocol

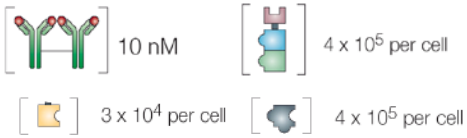
1. Identify components and interactions



2. Translate into objects (molecules) and rules



3. Determine **concentrations** and **rate constants**



4. Simulate and analyze the model

ODE's

SSA

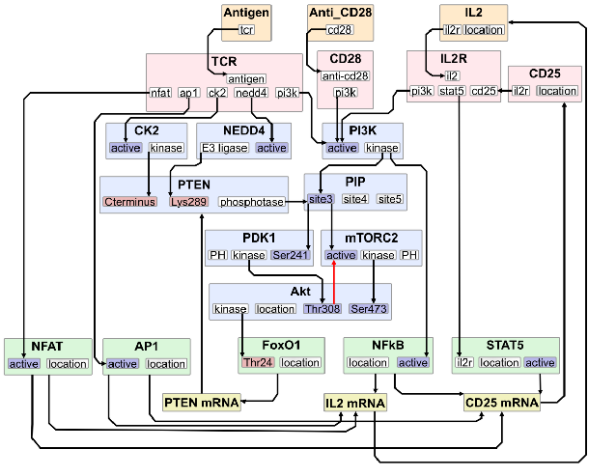
PDE's

BD

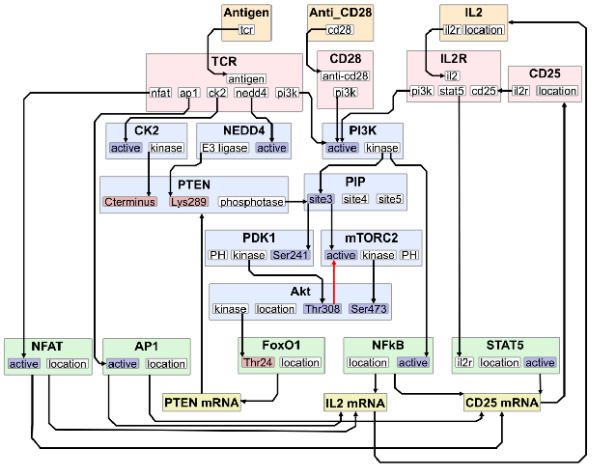
Systems Modeling Workflow

1. Construct model based on **known** and **hypothesized** interactions.

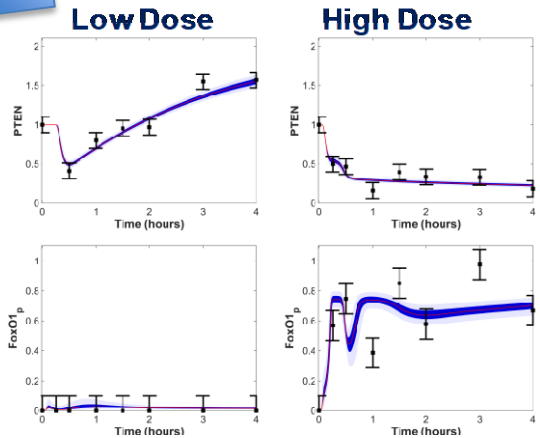
RuleBender
BioNetGen



Systems Modeling Workflow

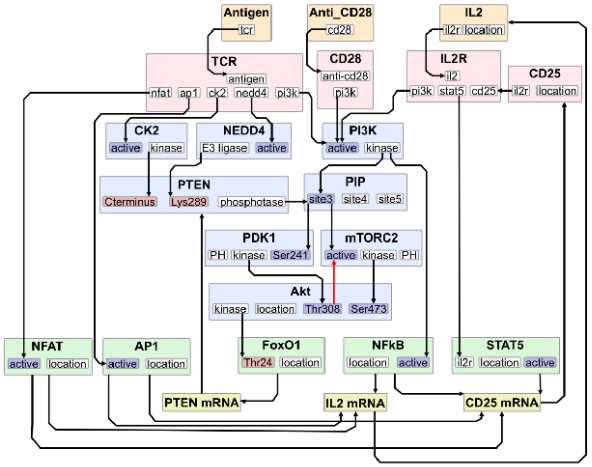


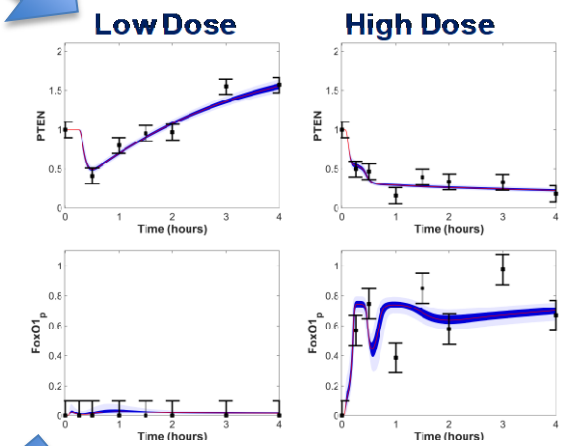
2. Calibrate model parameters to available data.

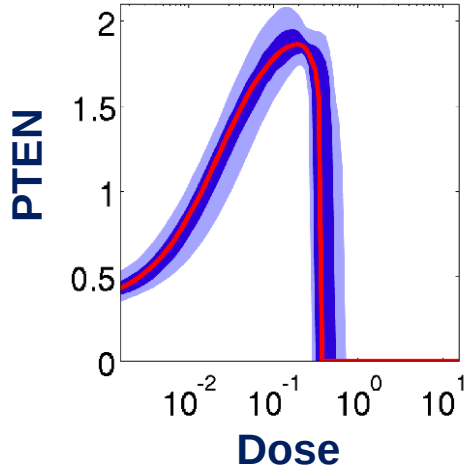


ptempest
BioNetFit

Systems Modeling Workflow



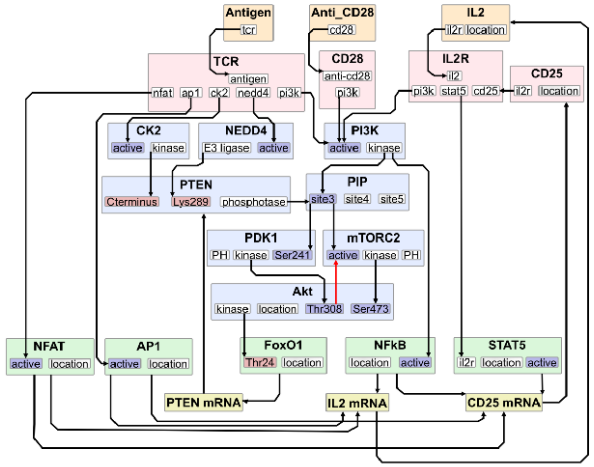


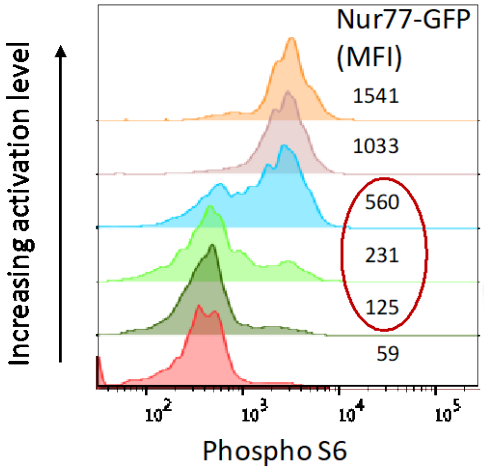


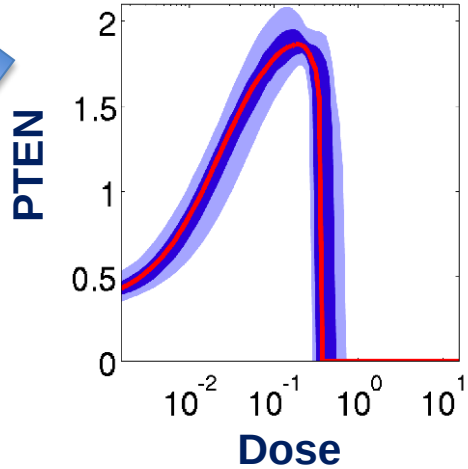
ptempest
RuleBender

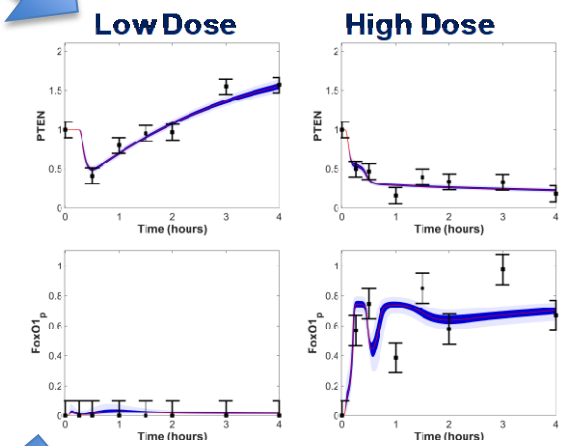
3. Analyze model to find novel behaviors.

Systems Modeling Workflow





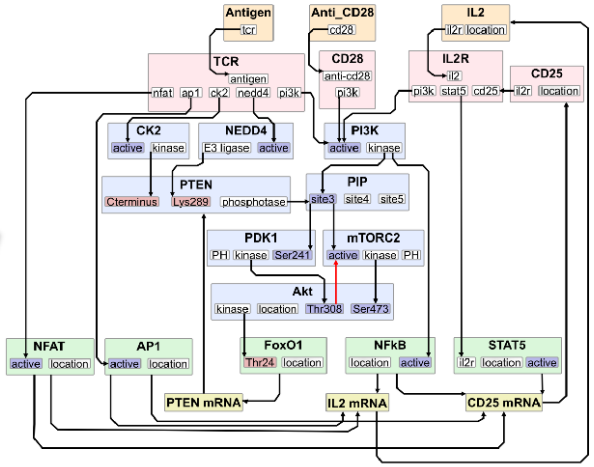




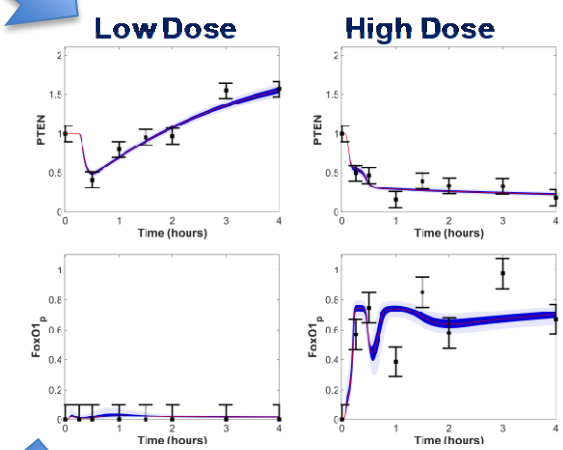
4. Perform experiments to test predictions.

Systems Modeling Workflow

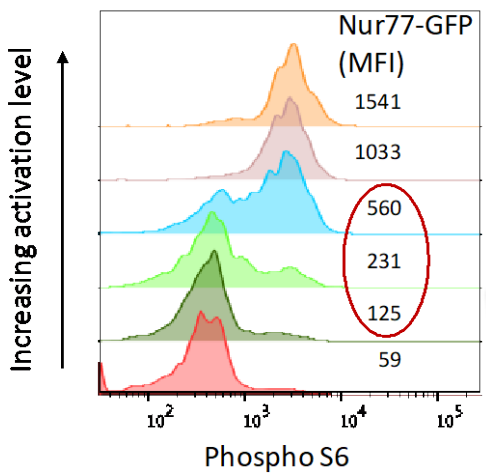
1. Construct model based on **known** and **hypothesized** interactions.

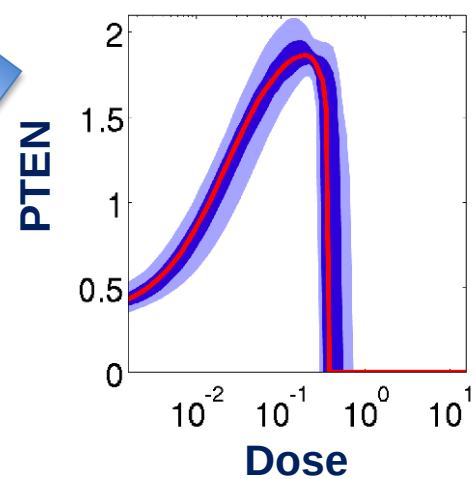


2. Calibrate model parameters to available data.



Repeat as needed





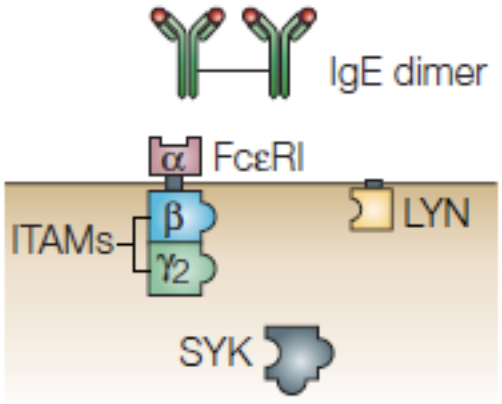
3. Analyze model to find novel behaviors.

4. Perform experiments to test predictions.

SPECIFYING A RULE-BASED MODEL

Defining Molecules

Molecules are the basic objects in a BNG model



BIONETGEN Language

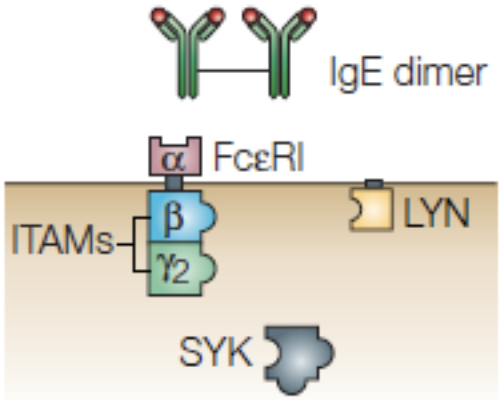
```
IgE(a,a)
FceRI(a,b~U~P,g2~U~P)
Lyn(U,SH2)
Syk(tSH2,lY~U~P,aY~U~P)
```

Components represent molecule elements

- Domains
- Motifs
- Properties

Defining Molecules

Molecules are the basic objects in a BNG model



<u>BioNetGen Language</u>

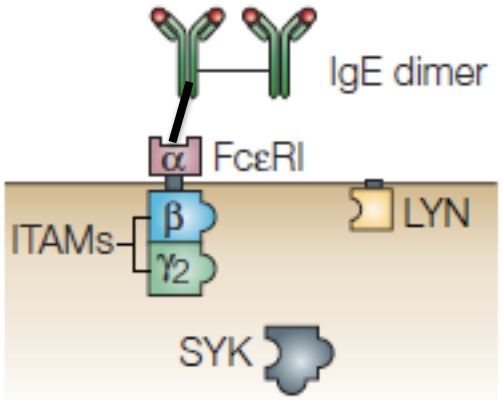
```
IgE(a,a)
FceRI(a,b~U~P,g2~U~P)
Lyn(U,SH2)
Syk(tSH2,lY~U~P,aY~U~P)
```

Components may have different states representing

- posttranslational modifications
- conformational state
- …

Binding

Molecules bind other molecules through components



BioNetGen Language

```
IgE(a,a!1).FceRI(a!1,b~U,g2~U)
```

Bonds are formed by linking two components. The '.' indicates a set of molecules forming a complex.

```
FceRI(a,b~U!1,g2~U).Lyn(U!1)
```

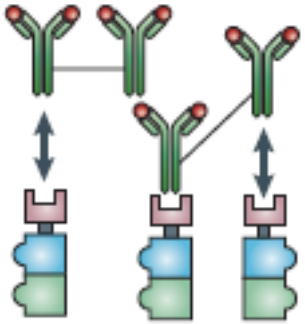
Components may have both states and bonds.

```
Lyn(SH2!1,Cterm~P!1)
```

Bonds may occur within a molecule.

Defining Interaction Rules

Ligand binding and aggregation

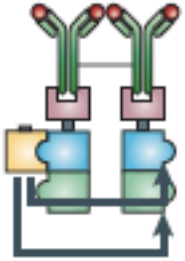


BioNetGen Language

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1)
…
```

binding and dissociation

Transphosphorylation



```
Lyn(U!1).FceRI(b!1).FceRI(b~U)-> \
Lyn(U!1).FceRI(b!1).FceRI(b~P)
```

component state change

Parts of a rule

Reactants | Arrow | Products | Rate Law

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

Parts of a rule

Reactants | Arrow | Products | Rate Law

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

```
IgE(a,a)  FceRI(a,b~U,g~U)
```

Reactant patterns select properties of each reactant molecule.

Parts of a rule

Reactants | Arrow | Products | Rate Law

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

```
IgE(a,a)   FceRI(a,b~U,g~U)

           FceRI(a,b~P,g~U)
```

States of components not specified don't affect the match.
"Don't write, don't care."

Reactant patterns select properties of each reactant molecule.

Parts of a rule

Reactants | Arrow | Products | Rate Law

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

```
IgE(a,a)+ FceRI(a,b~U,g~U)<-> IgE(a,a!1).FceRI(a!1,b~U,g~U) kp1, km1

IgE(a,a)+ FceRI(a,b~P,g~U)<-> IgE(a,a!1).FceRI(a!1,b~P,g~U) kp1, km1
```

Reactant patterns select properties of each reactant molecule.

Because patterns can match many different species, each rule can generate many reactions.

Center and context

The **center** of a rule is the part that the rule changes.

bond is formed

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

The **context** is the part that is necessary for the rule to happen but is unchanged.

Center and context

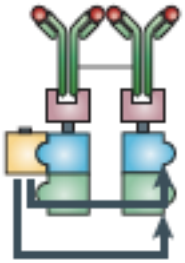
The **center** of a rule is the part that the rule changes.

bond is formed

```
IgE(a,a)+ FceRI(a)<-> IgE(a,a!1).FceRI(a!1) kp1, km1
```

The **context** is the part that is necessary for the rule to happen but is unchanged.

Transphosphorylation

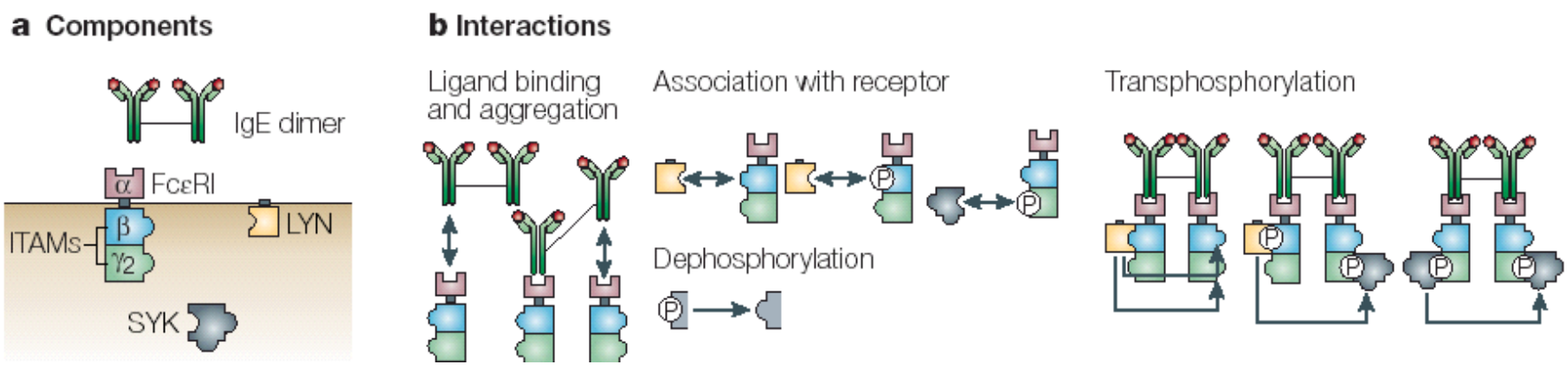


Context can represent complex biochemistry.

component state is changed

```
Lyn(U!1).FceRI(b!1).FceRI(b~U)-> \
Lyn(U!1).FceRI(b!1).FceRI(b~P)
```

Composition of a Rule-Based Model



Molecules

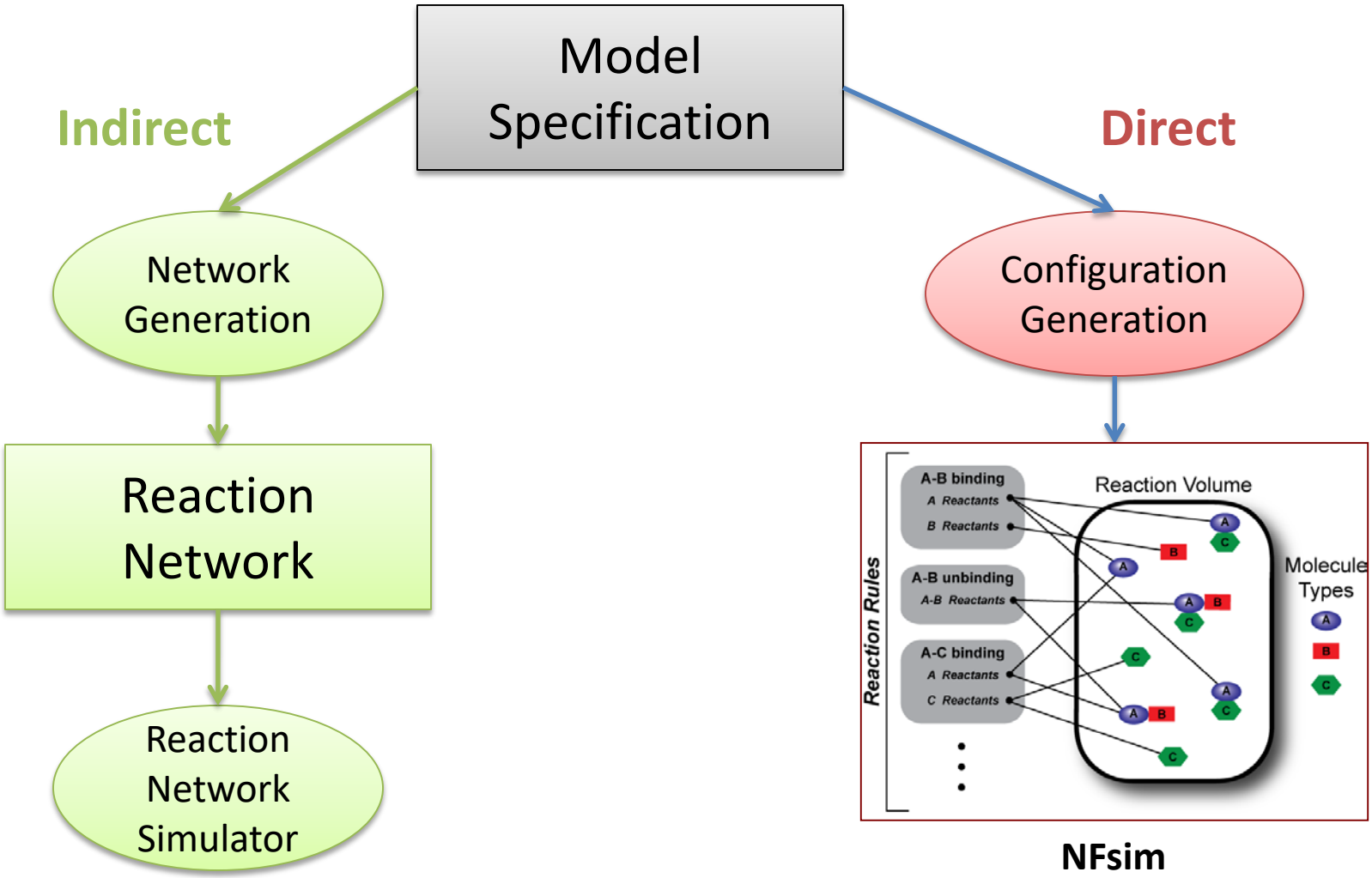
```
begin molecules
Lig(l,l)
Lyn(U,SH2)
Syk(tSH2,l~U~P,a~U~P)
Rec(a,b~U~P,g~U~P)
end molecules
```

Reaction Rules

BioNetGen language

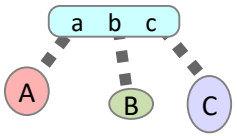
```
begin reaction_rules
# Ligand-receptor binding
1 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1
 Rec(a) + Lig(l,l) <-> Rec(a!1).Lig(l!1,l) kp1, km1

# Receptor-aggregation
2 Rec(a) + Lig(l,l!1) <-> Rec(a!2).Lig(l!2,l!1) kp2,km2

# Constitutive Lyn-receptor binding
3 Rec(b~Y) + Lyn(U,SH2) <-> Rec(b~Y!1).Lyn(U!1,SH2) kpL, kmL
…
```

Methods for simulating RBMs



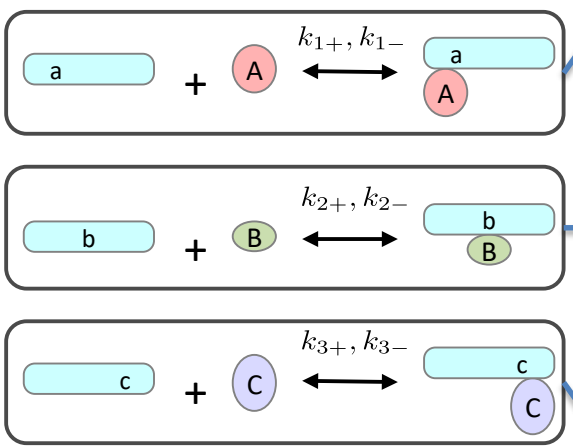
NFsim

Indirect Methods – Network Generation



contact map

reaction rules



$$\mathtt{S} + \mathtt{A} \overset{k_{1+},k_{1-}}{\rightleftharpoons} \mathtt{SA}$$
$$\mathtt{SB} + \mathtt{A} \overset{k_{1+},k_{1-}}{\rightleftharpoons} \mathtt{SAB}$$
$$\mathtt{SC} + \mathtt{A} \overset{k_{1+},k_{1-}}{\rightleftharpoons} \mathtt{SAC}$$
$$\mathtt{SBC} + \mathtt{A} \overset{k_{1+},k_{1-}}{\rightleftharpoons} \mathtt{SABC}$$

$$\mathtt{S} + \mathtt{B} \overset{k_{2+},k_{2-}}{\rightleftharpoons} \mathtt{SB}$$
$$\mathtt{SA} + \mathtt{B} \overset{k_{2+},k_{2-}}{\rightleftharpoons} \mathtt{SAB}$$
$$\mathtt{SC} + \mathtt{B} \overset{k_{2+},k_{2-}}{\rightleftharpoons} \mathtt{SBC}$$
$$\mathtt{SAC} + \mathtt{B} \overset{k_{2+},k_{2-}}{\rightleftharpoons} \mathtt{SABC}$$

$$\mathtt{S} + \mathtt{C} \overset{k_{3+},k_{3-}}{\rightleftharpoons} \mathtt{SC}$$
$$\mathtt{SA} + \mathtt{C} \overset{k_{3+},k_{3-}}{\rightleftharpoons} \mathtt{SAC}$$
$$\mathtt{SB} + \mathtt{C} \overset{k_{3+},k_{3-}}{\rightleftharpoons} \mathtt{SBC}$$
$$\mathtt{SAB} + \mathtt{C} \overset{k_{3+},k_{3-}}{\rightleftharpoons} \mathtt{SABC}$$

reactions

S, A, B, C,
SA, SB, SC,
SAB, SAC, SBC,
SABC

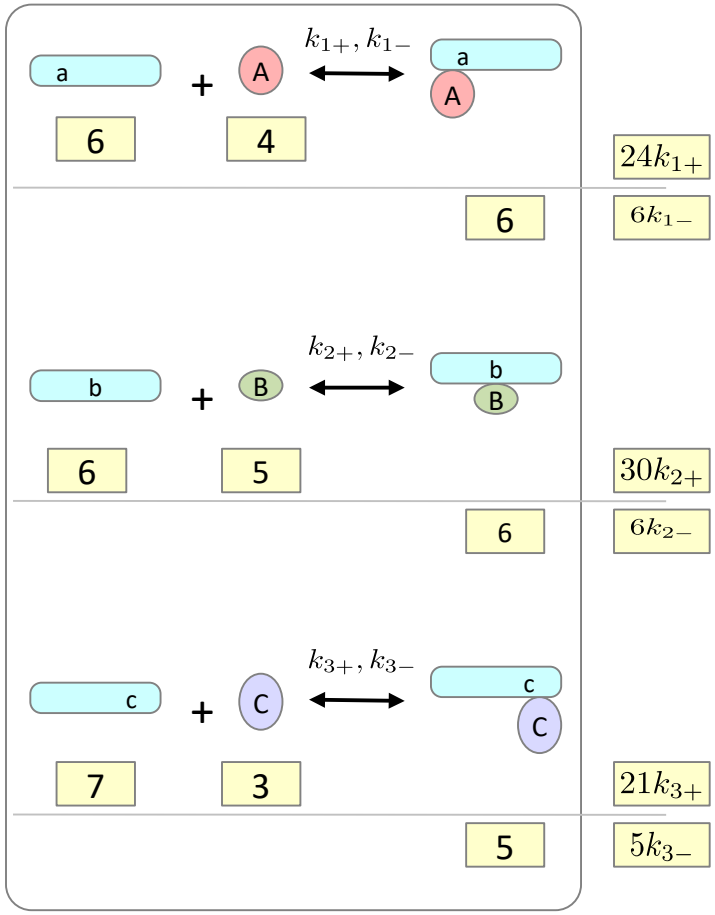
species

4 molecule types
3 rules

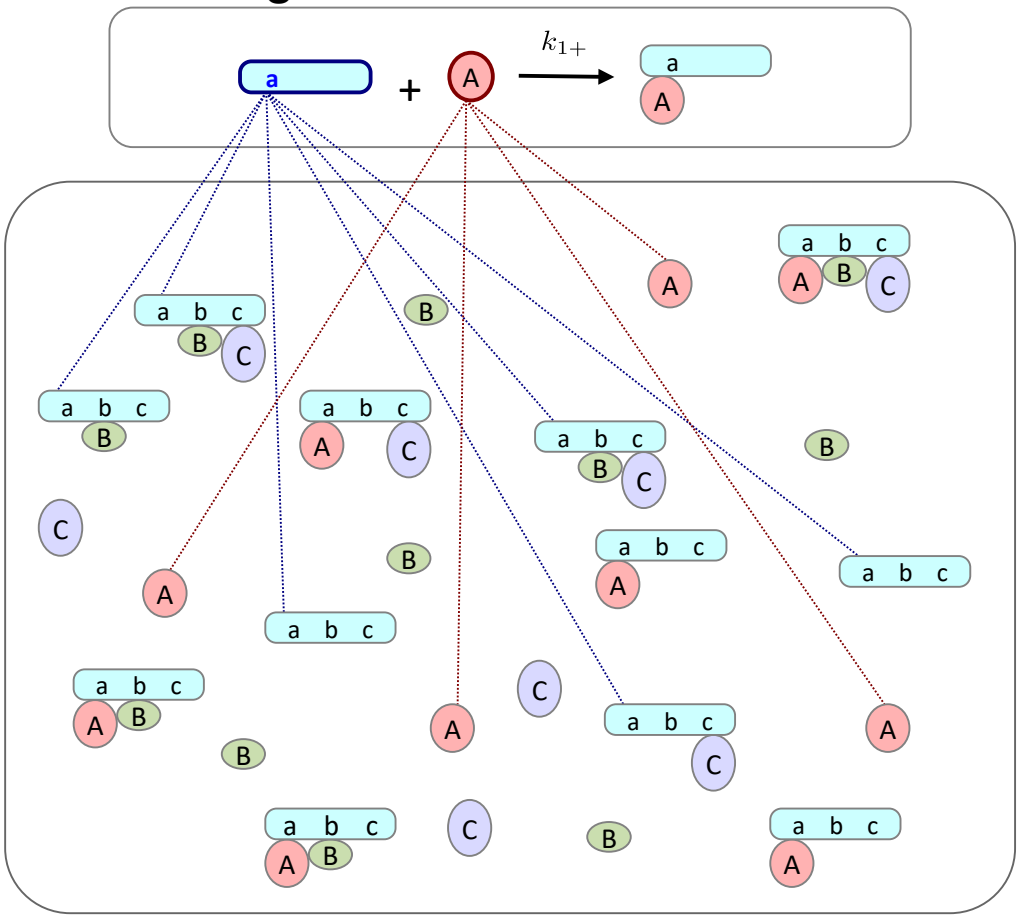
11 species
12 reactions

Direct Methods (NFsim)

reaction rules



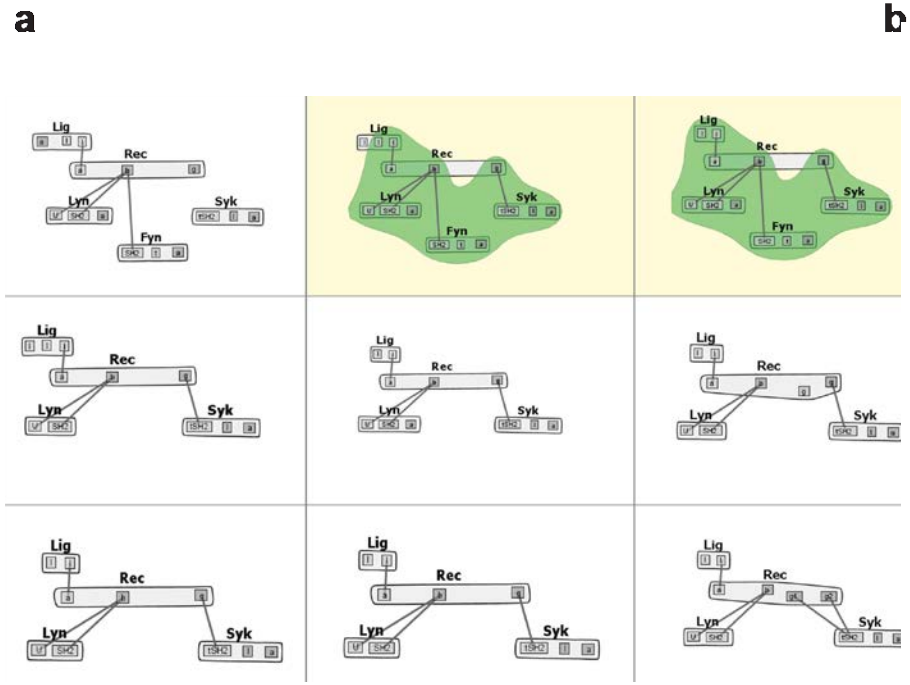
event generation



system configuration

total propensity $24k_{1+} + 6k_{1-} + 30k_{2+} + 6k_{2-} + 21k_{3+} + 5k_{3-}$

FcεRI signaling models



NFSIM can simulate models of greatly increased complexity with manageable increase in cost.

MW Sneddon, JRF, and T. Emonet (2011) *Nat. Methods*, **8**, 177

RuleBender

An eclipse RCP application

rulebender.org

RuleBender

Model | Simulation | Results

egfr_net.bngl

```
#Dephosphorylayion
 egfr(Y1068~pY) -> egfr(Y1068~Y)  km3
 egfr(Y1148~pY) -> egfr(Y1148~Y)  km3

# Shc transphosph

 egfr(r!2,Y1148~pY!1).Shc(PTB!1,Y317~Y) ->  egfr(r!2,Y1148~pY!1).Shc(
 Shc(PTB!1,Y317~pY)  ->  Shc(PTB!1,Y317~Y)  km14


# Y1068 activity
 egfr(Y1068~pY) + Grb2(SH2,SH3)   <-> egfr(Y1068~pY!1).Grb2(SH2!1,SH3
 egfr(Y1068~pY) + Grb2(SH2,SH3!2) <-> egfr(Y1068~pY!1).Grb2(SH2!1,SH3
 egfr(Y1068~pY!1).Grb2(SH2!1,SH3) + Sos(dom) <-> egfr(Y1068~pY!1).Grb

# Y1148 activity

 egfr(Y1148~pY) + Shc(PTB,Y317~Y)    <-> egfr(Y1148~pY!1).Shc(PTB!1,Y
 egfr(Y1148~pY) + Shc(PTB,Y317~pY)   <-> egfr(Y1148~pY!1).Shc(PTB!1,Y
 egfr(Y1148~pY) + Shc(PTB,Y317~pY!1).Grb2(SH2!1,SH3)   <-> \
```

Contact Map

Problems

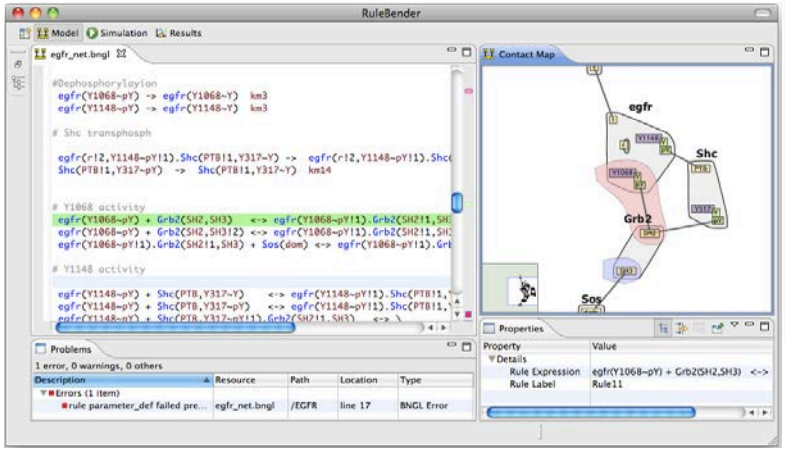
1 error, 0 warnings, 0 others

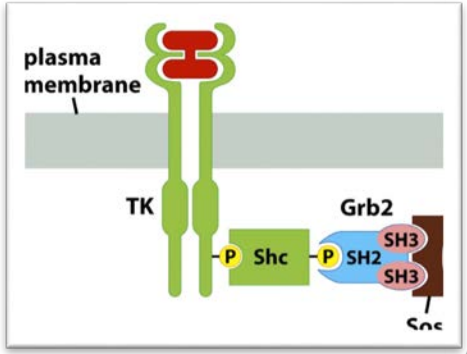
Description	Resource	Path	Location	Type
Errors (1 item)				
rule parameter_def failed pre...	egfr_net.bngl	/EGFR	line 17	BNGL Error

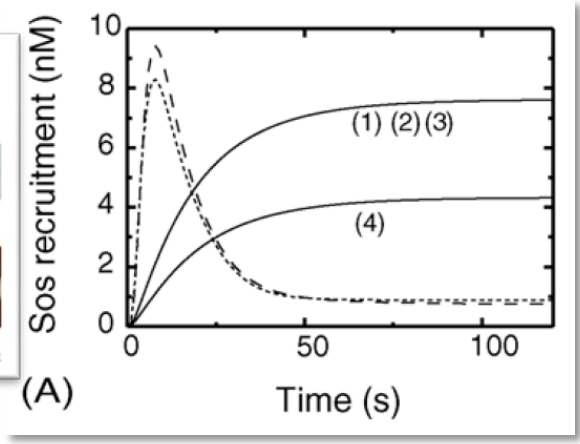
Properties

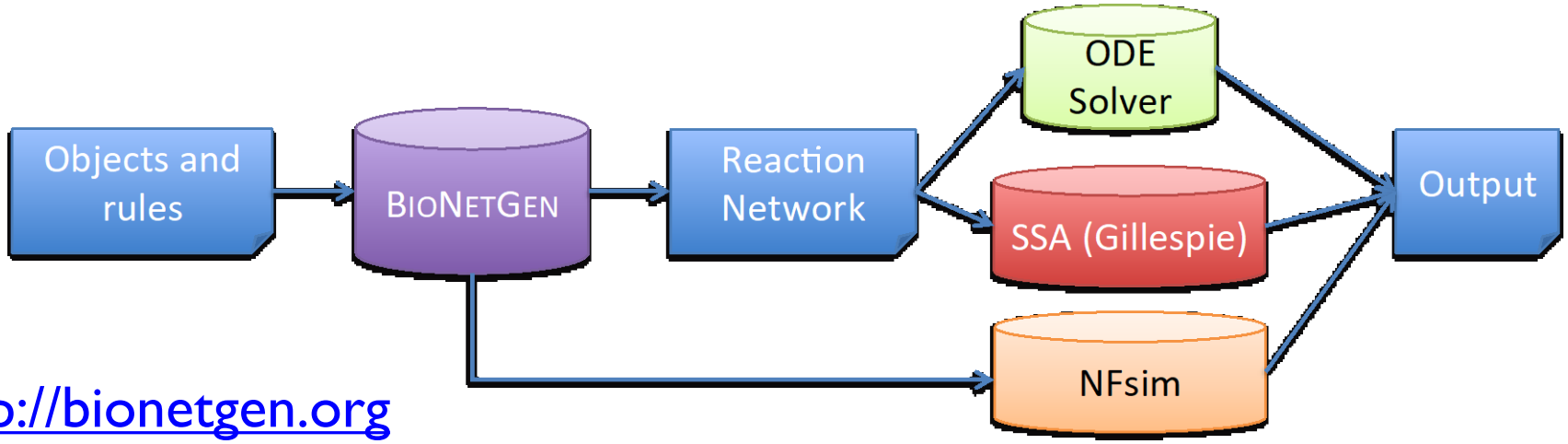
Property	Value
Details	
Rule Expression	egfr(Y1068~pY) + Grb2(SH2,SH3) <->
Rule Label	Rule11

Basic RBM workflow with BioNetGen



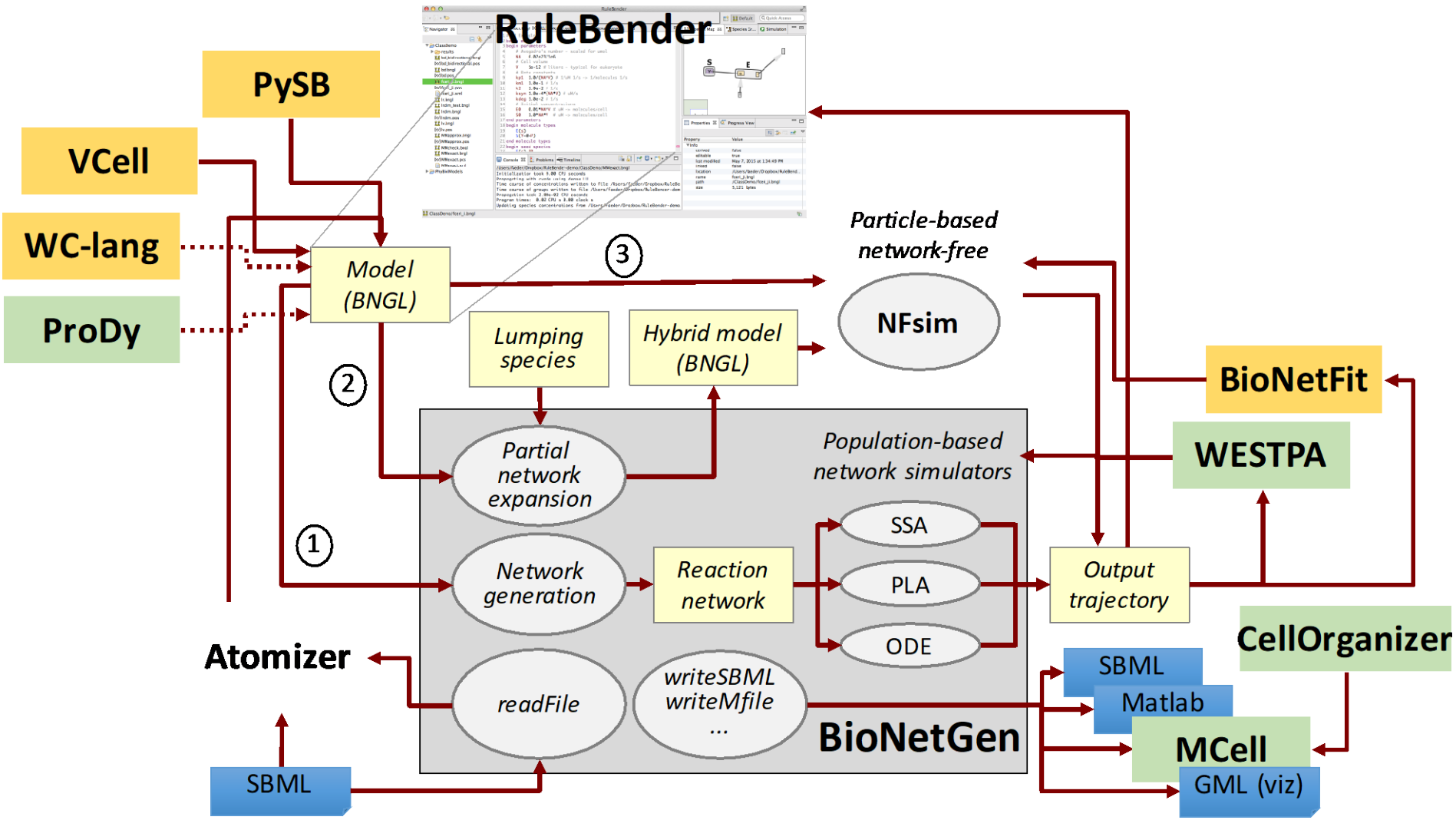






http://bionetgen.org
http://rulebender.org
http://nfsim.org

BioNetGen and Friends

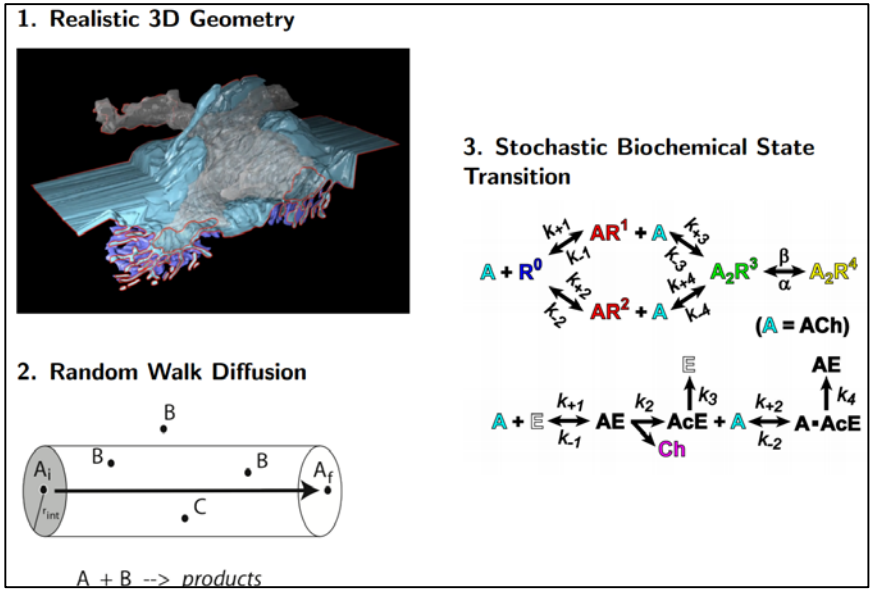


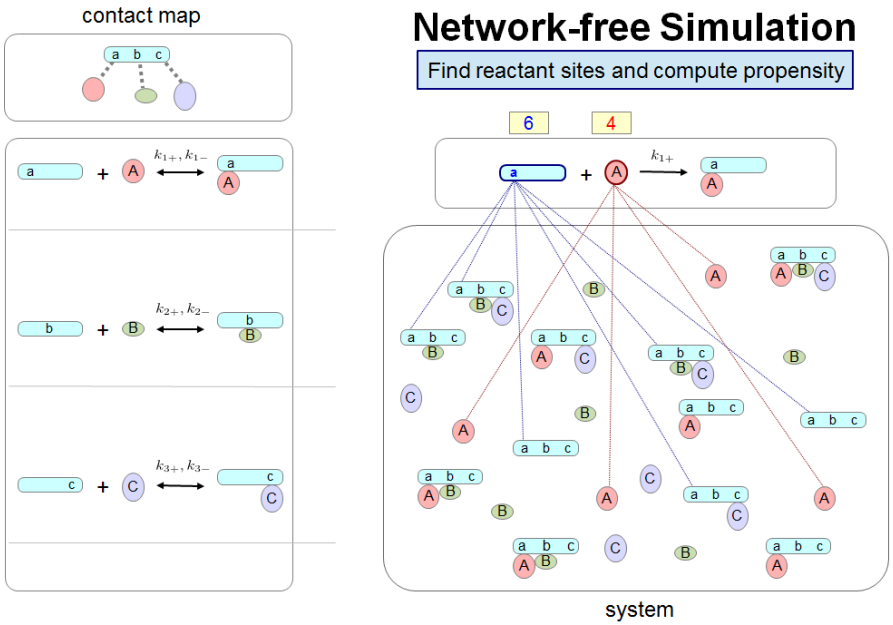
Development of network-free MCell

Beyond a well-mixed cell….

Jose Juan Tapia

MCell

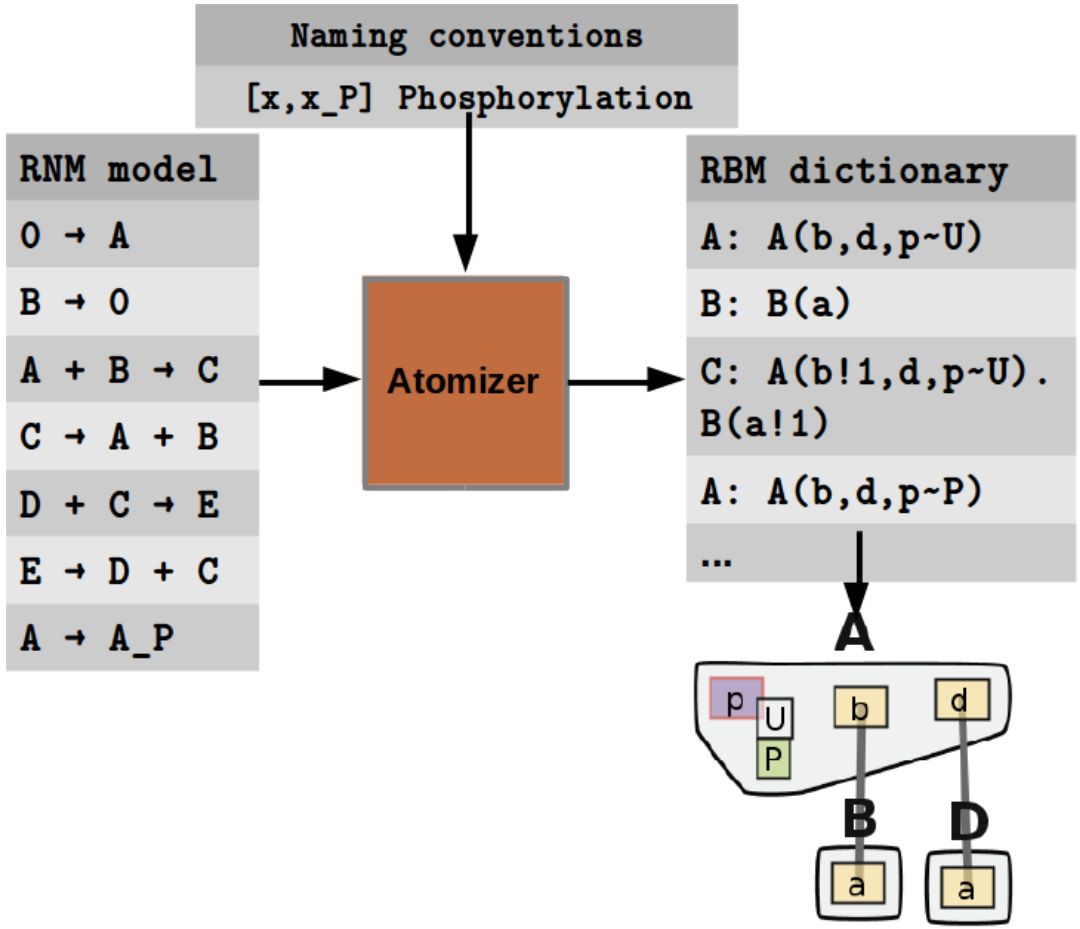




Particle-based

Atomizer can recover implicit molecular structure reaction networks

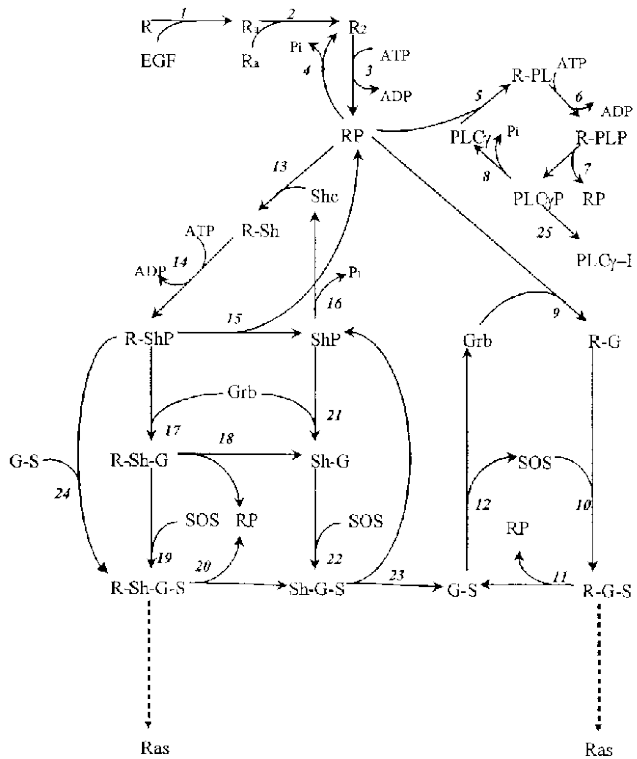
1. Reaction stoichiometry
2. Naming conventions
3. Annotation

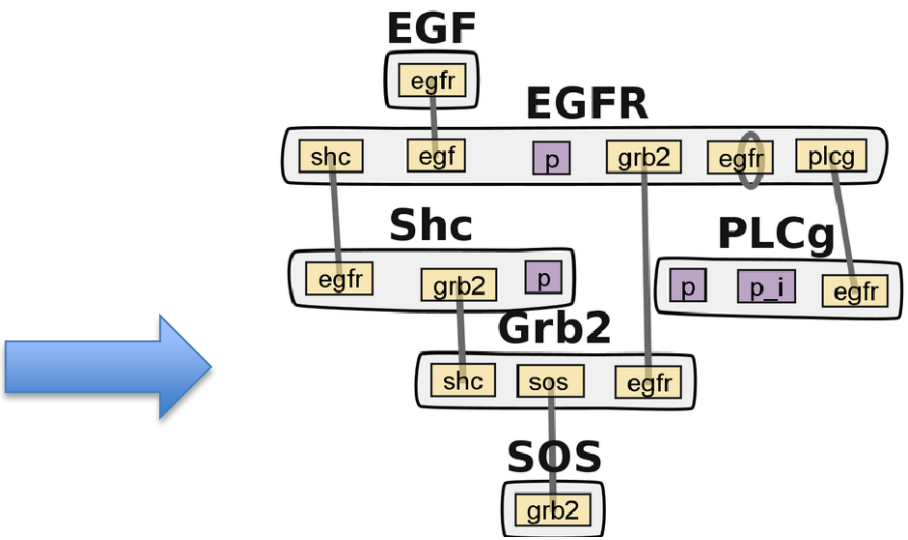


Tapia and Faeder (2013) *Proc. ACM-BCB*.

Automating structure recovery

"Atomization"





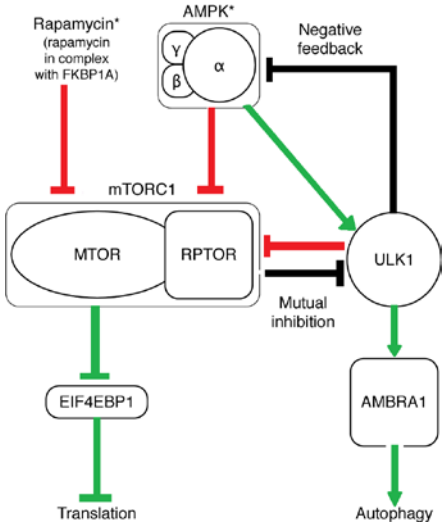
Motivation

- **Find molecules and interactions used to build model**
- **Find out what has been modeled?**
- **Find the implicit assumptions**
- **Automate study of relaxing model assumptions**

Visualizing rule-based models

John Sekar
Jose Juan Tapia

Autophagy network



Szymanska et al. (2015), *PLOS One*

Rule-based model

Rules are code!

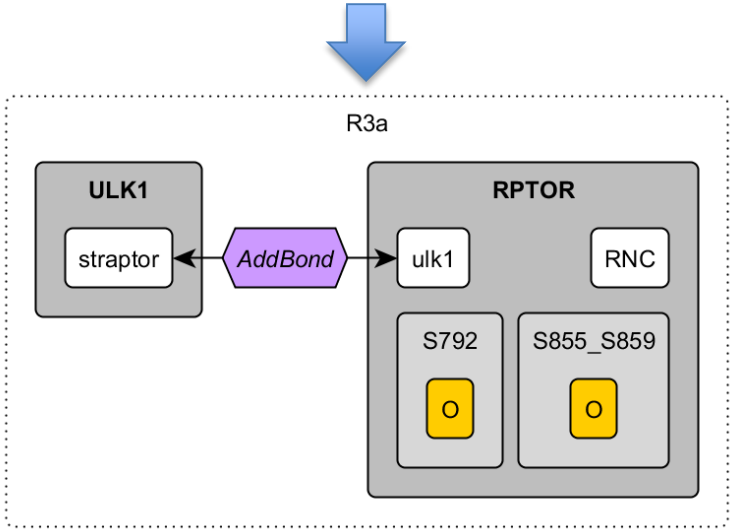
R1: rapa(mtor)+MTOR(HEAT,FRB)<->rapa(mtor!1).MTOR(HEAT,FRB!1) a1,d1
R2: MTOR(HEAT,FRB)+RPTOR(WD40)<->MTOR(HEAT!1,FRB).RPTOR(WD40!1) a2,d2
R3a: RPTOR(RNC,ulk1,S792~O,S855_S859~O)+ULK1(straptor) ->RPTOR(RNC,ulk1!1,S792~O,S855_S859~O).ULK1(straptor!1) a3
R3b: RPTOR(ulk1!1).ULK1(straptor!1)->RPTOR(ulk1)+ULK1(straptor) d3
R3c: RPTOR(ulk1!1,S792~P,S855_S859~PP).ULK1(straptor!1)->RPTOR(ulk1,S792~P,S855_S859~PP)+ULK1(straptor) d3max
R4a: RPTOR(RNC,ulk1,S792~O,S855_S859~O)+EIF4EBP1(RCR,S65_T70~O)-
>RPTOR(RNC!1,ulk1,S792~O,S855_S859~O).EIF4EBP
R4b: RPTOR(RNC!1).EIF4EBP1(RCR!1)->RPTOR(RNC
R5a: AMPK(ulk1,T172~P)+ULK1(stampk,S758~O)->AM
R5b: AMPK(ulk1!1).ULK1(stampk!1)->AMPK(ulk1)+ULK1
R6: MTOR(HEAT!1).RPTOR(WD40!1,ulk1!2).ULK1(strap
>MTOR(HEAT!1).RPTOR(WD40!1,ulk1!2).ULK1(straptor
R7: MTOR(HEAT!1).RPTOR(WD40!1,RNC!2).EIF4EBP1
>MTOR(HEAT!1).RPTOR(WD40!1,RNC!2).EIF4EBP1(R
R8a: RPTOR(ulk1!1,S792~O).ULK1(straptor!1,S317~P,S
R8b: RPTOR(ulk1!1,S855_S859~O).ULK1(straptor!1,S31
R8c: RPTOR(ulk1!1,S855_S859~P).ULK1(straptor!1,S31
R9: ULK1(straptor,S317~P,S778~P)+AMBRA1(ST~O)->
R10: ULK1(straptor,S317~P,S778~P)+AMPK(ST~O)->U
R11a: AMPK(ulk1!1,T172~P,ST~O).ULK1(stampk!1,strap
R12: AMPK(T172~P)+RPTOR(S792~O)->AMPK(T172~P
R13a: ULK1(S758~P)->ULK1(S758~O) u0
R13b: EIF4EBP1(S65_T70~P)->EIF4EBP1(S65_T70~O)
R13c: RPTOR(S792~P)->RPTOR(S792~O) u1
R13d: RPTOR(S855_S859~P)->RPTOR(S855_S859~O)
R13e: RPTOR(S855_S859~PP)->RPTOR(S855_S859~P
R13f: AMBRA1(ST~P)->AMBRA1(ST~O) u0
R13g: AMPK(ST~P)->AMPK(ST~O) u2
R13h: ULK1(straptor,S317~P)->ULK1(straptor,S317~O) u0
R13i: ULK1(S778~P)->ULK1(S778~O) u0
R13j: RPTOR(S792~P)->RPTOR(S792~O) u0

Why rule visualization is essential

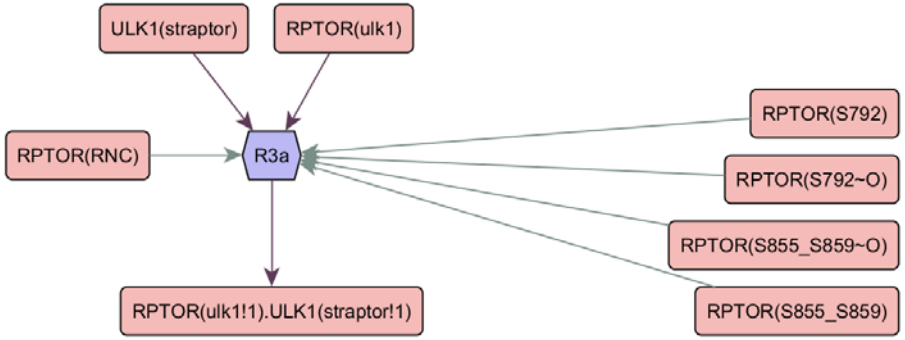
1. Debugging
2. Model reuse / sharing
3. Communicating with collaborators
4. Model comparison
5. Model integration

Automated rule visualization

R3a: RPTOR(RNC,ulk1,S792~O,S855_S859~O) + ULK1(straptor) -> \
RPTOR(RNC,ulk1!1,S792~O,S855_S859~O).ULK1(straptor!1) a3



Compact rule visualization



Regulatory graph

Faeder Lab Current and Past

John Sekar, PhD
Jose Juan Tapia
Robert Sheehan
Cihan Kaya (w/ I. Bahar)
Sanjana Gupta (w/ R. Lee)
Kyle Davidson
Rob Clark, PhD

Leonard Harris, PhD (Vanderbilt)
Natasa Miskov-Zivanov, PhD (Pitt-ECE)
Dipak Barua, PhD (UMIST)

Jintao Liu, PhD (UCSD)
Justin Hogg, PhD (Apple)

Arshi Arora (MSKCC)
Qingyang Ding (Tsinghua)
Tyler McLaughlin (Rice)
Ilya Korsunksy (NYU)
Mate Nagy (Yale)
Niketh Nair (Oak Ridge)
Lori Stover (Macmillan)

Computational Collaborators

Lily Chylek

BioNetGen

RuleBender
NFsim
Atomizer

MCell
WESTPA

Experimental Collaborators

Penelope Morel *(PITT-Immunology)*
Lawrence Kane
Michael Turner
William Hawse
Ashley Menk

Sarah Gaffen *(PITT-Immunology)*

Peter Setlow *(UCHC)*
Yongqing Li *(ECU)*
Shiwei Wang

Alexander Sorkin (PITT-Cell Biology)

Funding

NIH-NIGMS P41, P50, R01
NIH-NIAID R01
NSF Expeditions, EMT
NIH-NCRR CTSI
DOD MURI