Bioinformatics

Modelling dynamic behaviour (Systems biology)

David Gilbert

Bioinformatics Research Centre

www.brc.dcs.gla.ac.uk

Department of Computing Science, University of Glasgow

Lecture outline

- Biochemical reactions
- Modelling with Ordinary Differential Equations
- Kinetics : Mass Action
- Examples
  - Signalling & metabolic pathways
  - Oscillators & Amplifiers
- Analysis
- ODE simulators

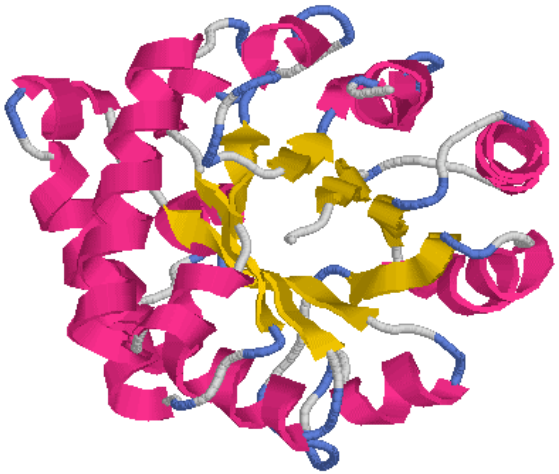
Genes to systems

DNA

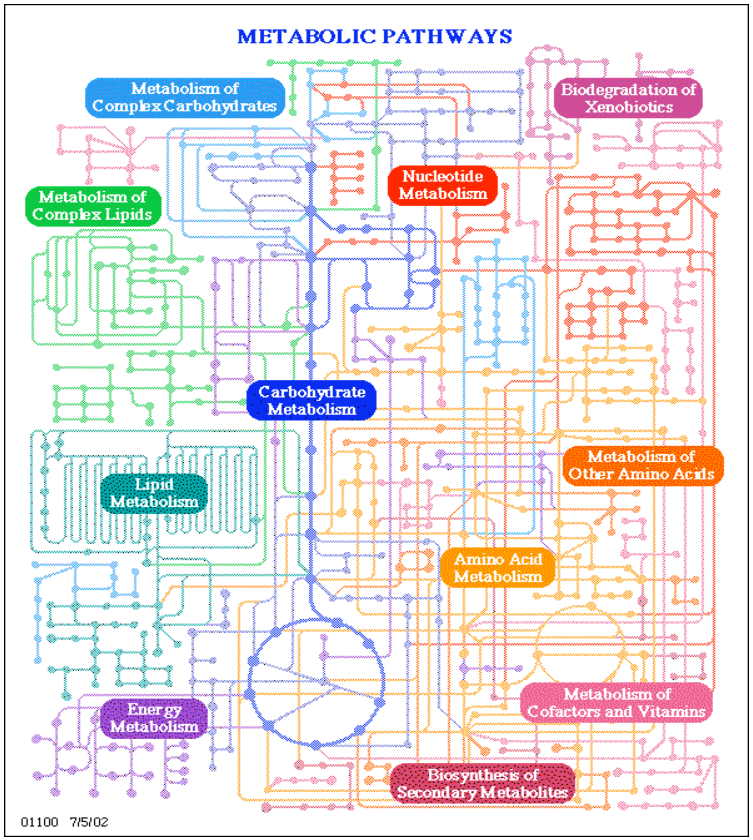
"gene"

mRNA

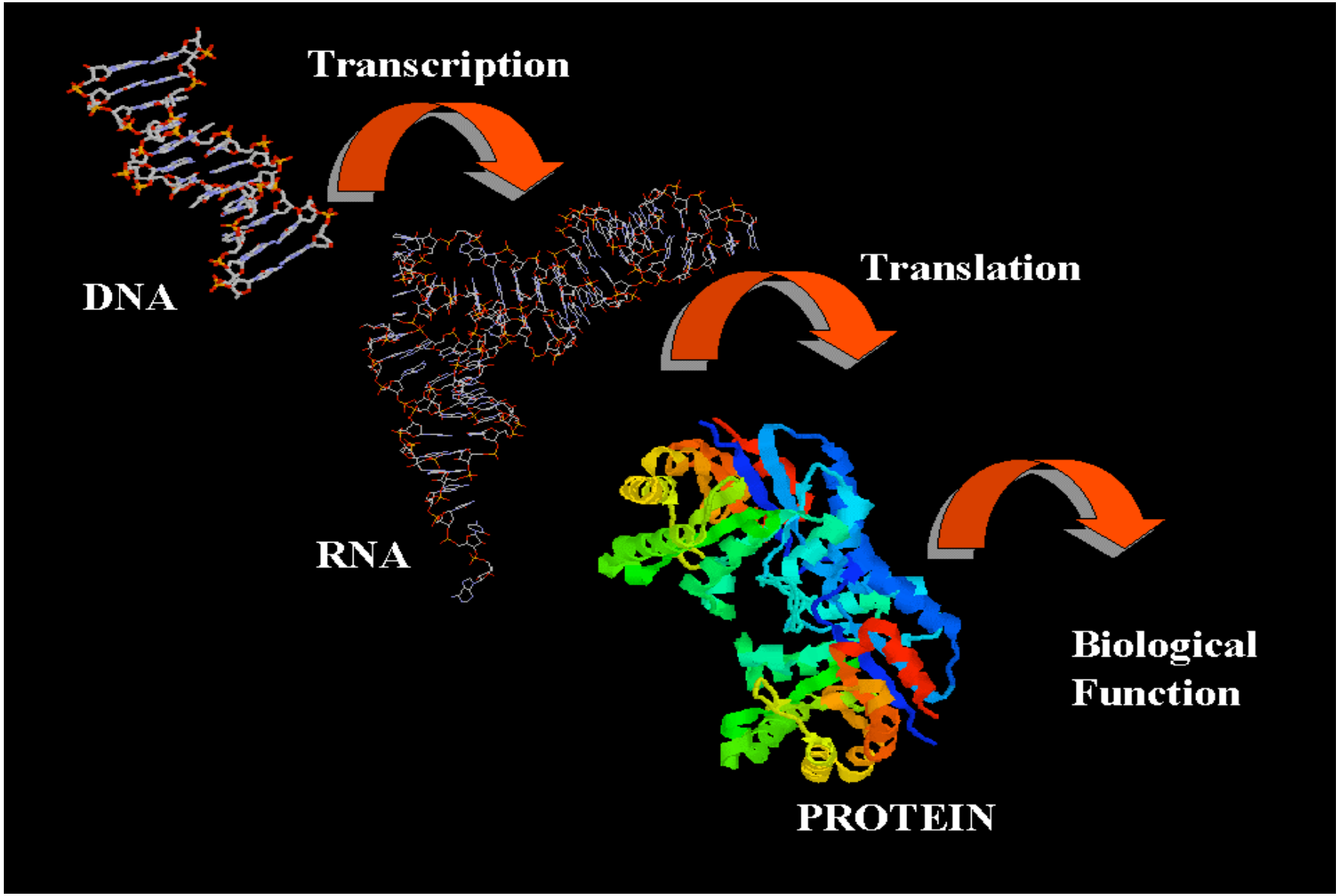
Protein sequence



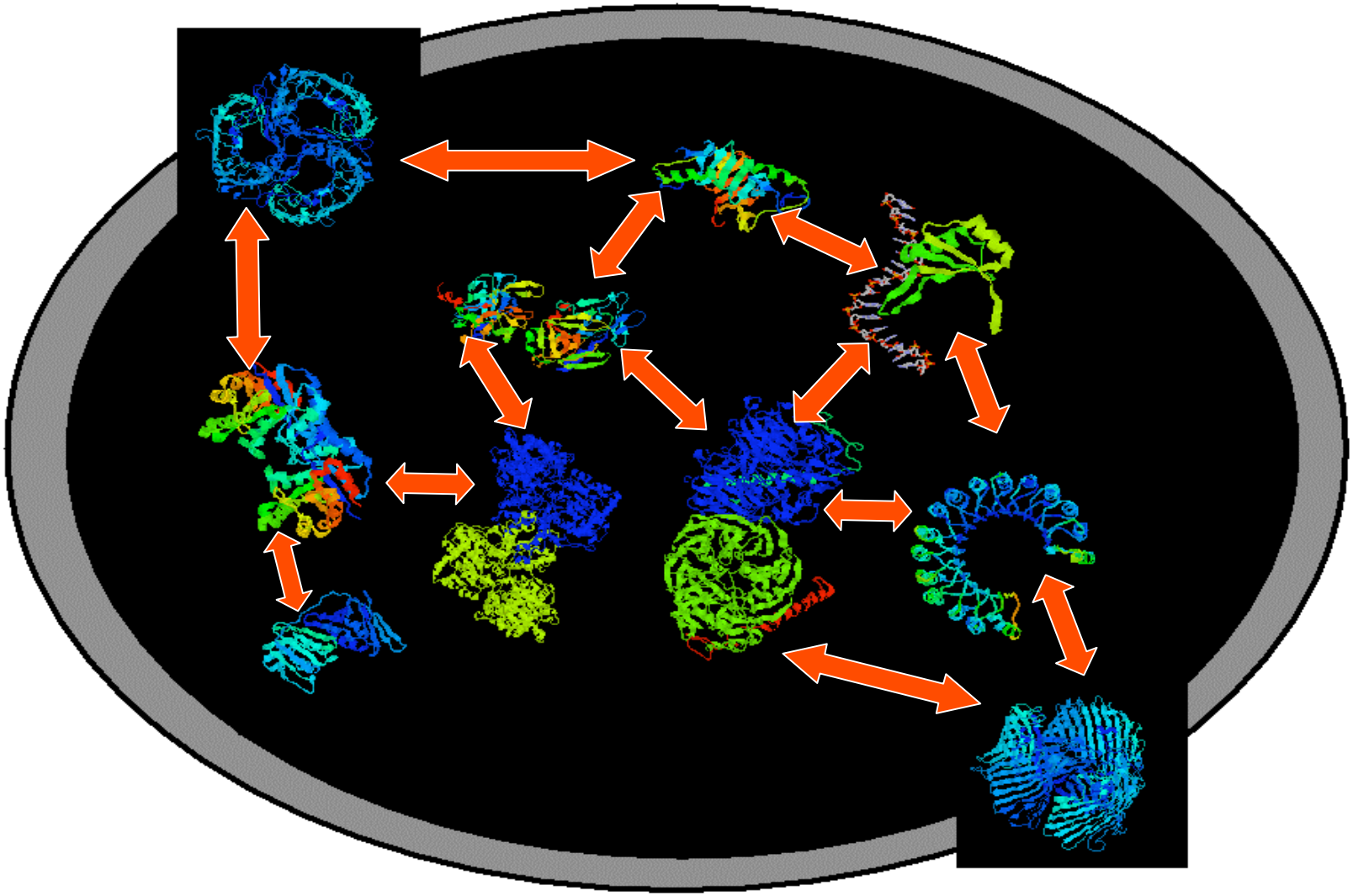
Folded Protein



Behaviour of the gene …



… their interaction



Terminology: Pathways or Networks?

- Pathways implies ‘paths’ - sequences of objects
- Networks - more complex connectivity
- Both are represented by ***graphs***
- Networks: generic; Pathways: specific (?)
  - ‘Signal transduction networks’
  - ‘The ERK signal transduction pathway’

What is a pathway?

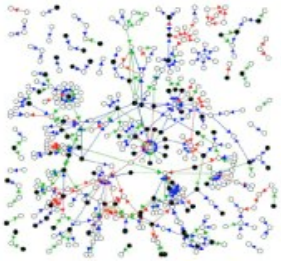
- An ordered sequence of proteins and substrates
- A series of biochemical reactions
- An evolutionary product
- A biological system (living cell)
- A biochemical network/graph

Issues:

  - Organism-specific adaptations
  - Which enzyme sets are involved?

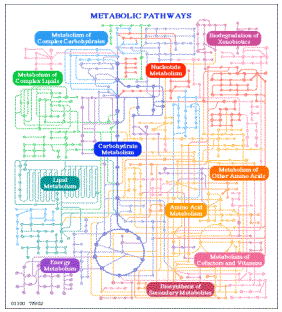
Networks

- Gene regulation

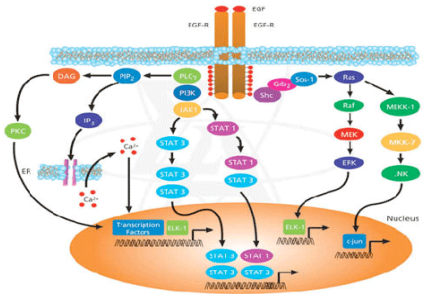


- Protein-protein interaction

- Metabolic



- Signalling



- Developmental

Some Definitions

- Catalyst - substance that increases the rate of a chemical reaction without being consumed in the process.
- Enzyme
  - biological catalyst
  - mainly they are proteins
  - Highly specific for a particular reaction
- Coenzyme - enhances the activity of an enzyme
- substrate (reactant) - consumed in a reaction
- product - produced by a reaction

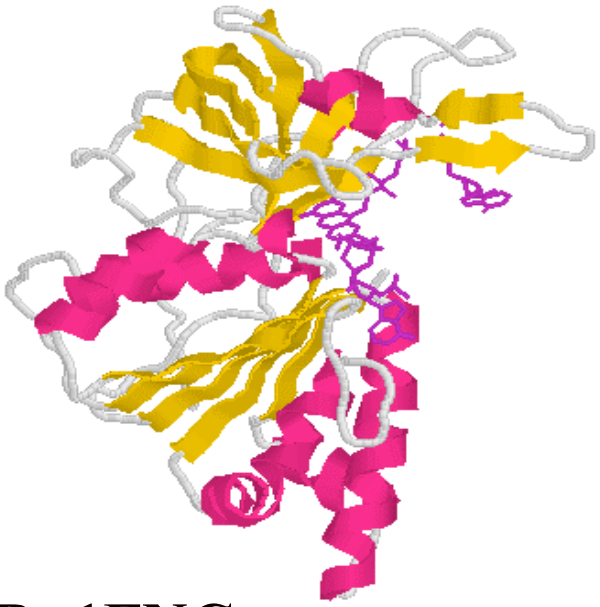
$$NADH + H^+ + C_2H_4O \Leftrightarrow C_2H_5OH + NAD^+$$

Alcohol dehydrogenase

(cofactor) (ion) (acetaldehyde) (ethanol) (cofactor)

EC Classification (EC)

- Classified according to Enzyme Nomenclature (IUBMB)
  (http://www.chem.qmw.ac.uk/iubmb/enzyme/)
- Six major biochemical reactions
- Denoted in four figures (EC X.X.X.X) according to the reaction



PDB: 1FNC

EC 1.18.1.2

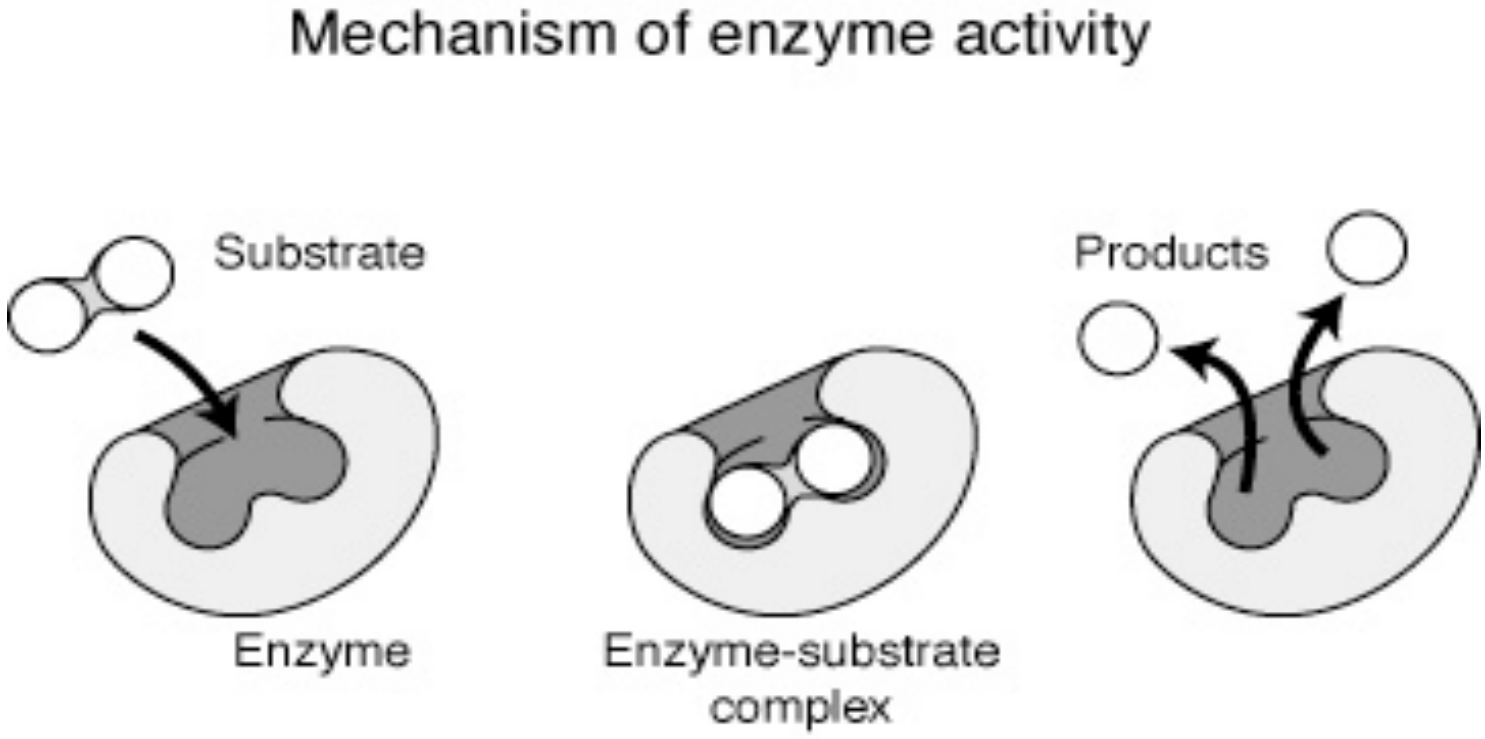
First digit:
Describe the enzyme overall reaction.
EC 1. is the oxidoreductases that catalyse oxido-reductions reaction.
The substrate oxidised is regarded as an hydrogen or electron donor.

Second digit:
Describes substrate acted on by enzyme.
EC 1.18. is acting on reduced ferredoxin as donor.

Third digit:
Describe type of acceptor.
EC 1.18.1. acceptor is NAD(+) or NADP(+)

Fourth digit:
Is the enzyme's arbitrarily assigned serial number in its subsubclass

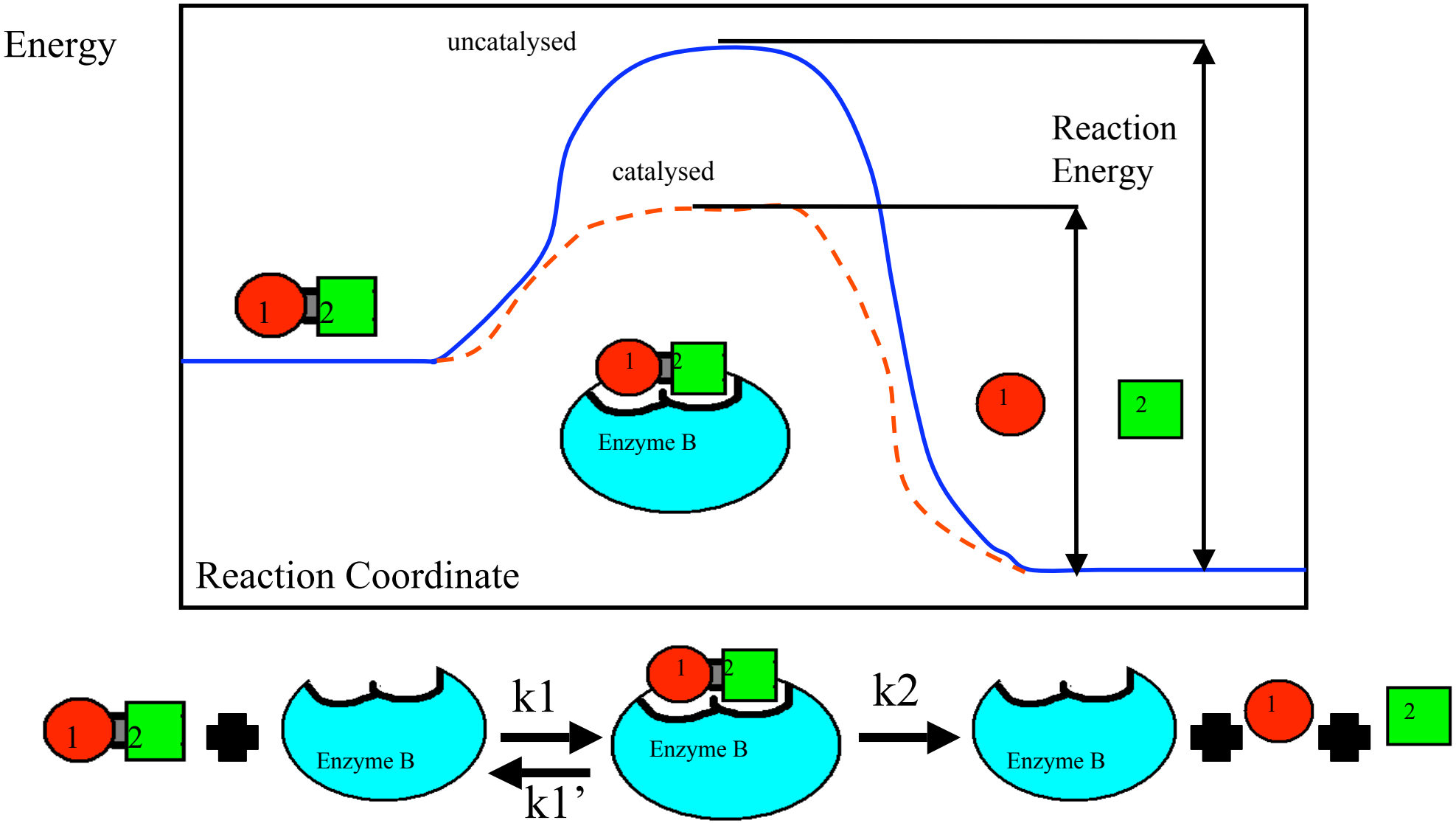
Mechanism of Enzyme Activity



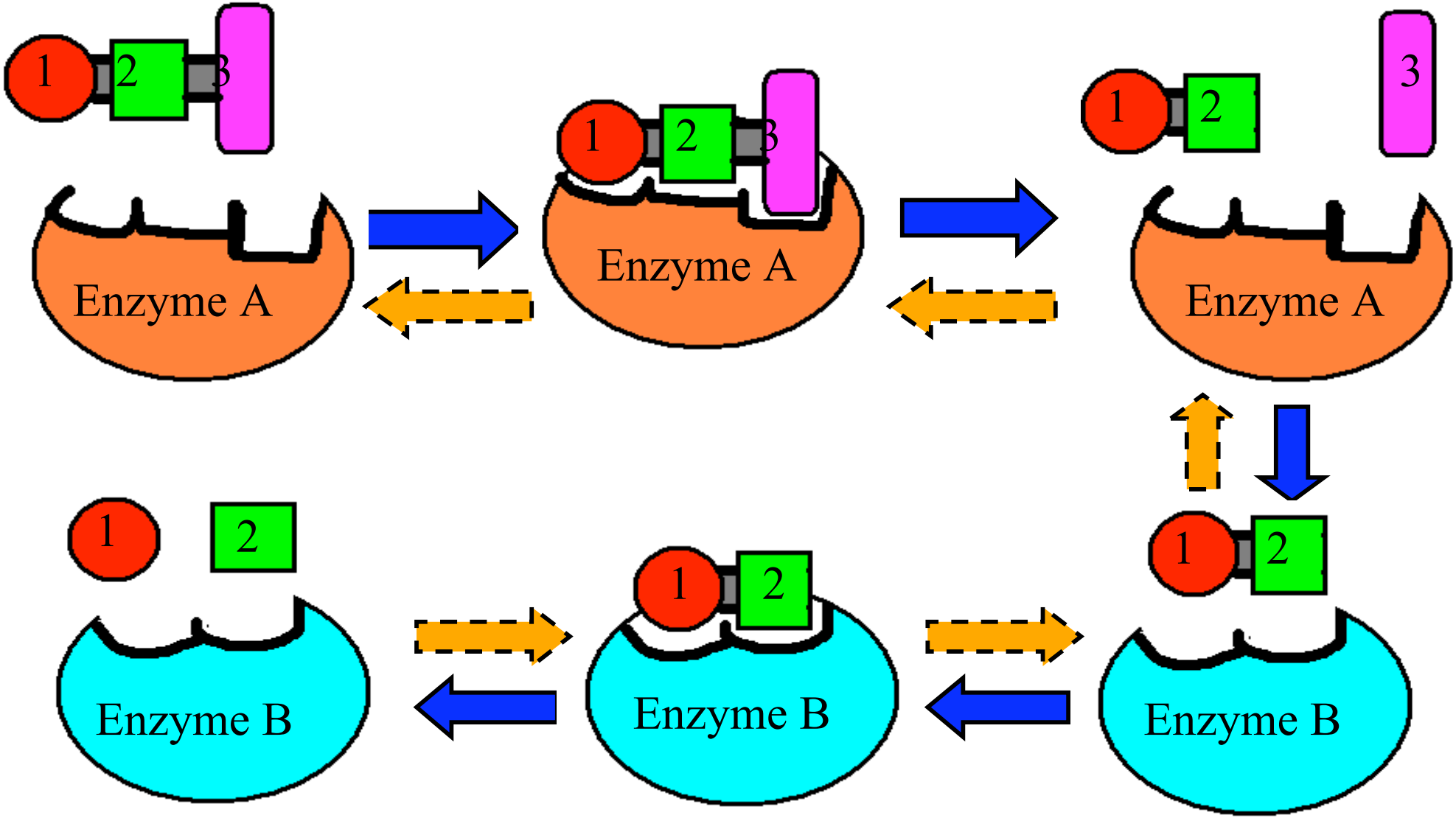
$$S + E \Leftrightarrow ES \Leftrightarrow E + P$$

http://www.accessexcellence.org/AB/GG/

Transition State Diagram



Pathway = A series of Enzymatic Reactions

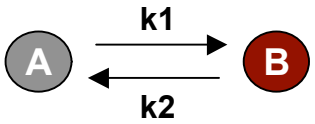


Slide from Richard Orton

What is modelling?

- In this context:
  - Translating a biological pathway into mathematics for subsequent analysis

Translating a biological pathway



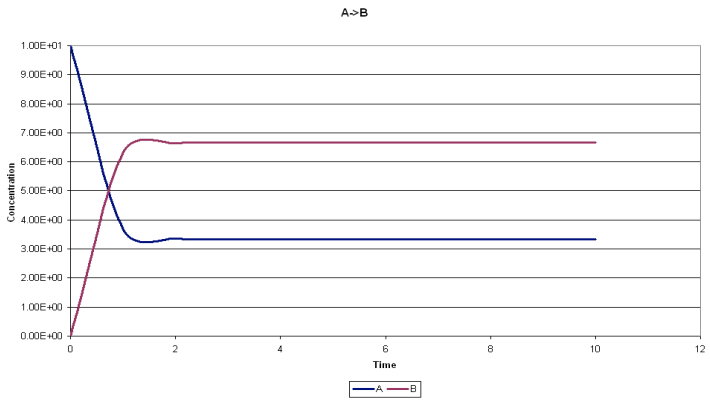
Into mathematics

$$\frac{d[A]}{dt} = -k_1[A] + k_2[B]$$

$$\frac{d[B]}{dt} = k_1[A] - k_2[B]$$

[A] = 10; [B] = 0; k1 = 2; k2 = 1; Time = 10

For subsequent analysis



Slide from
Richard Orton

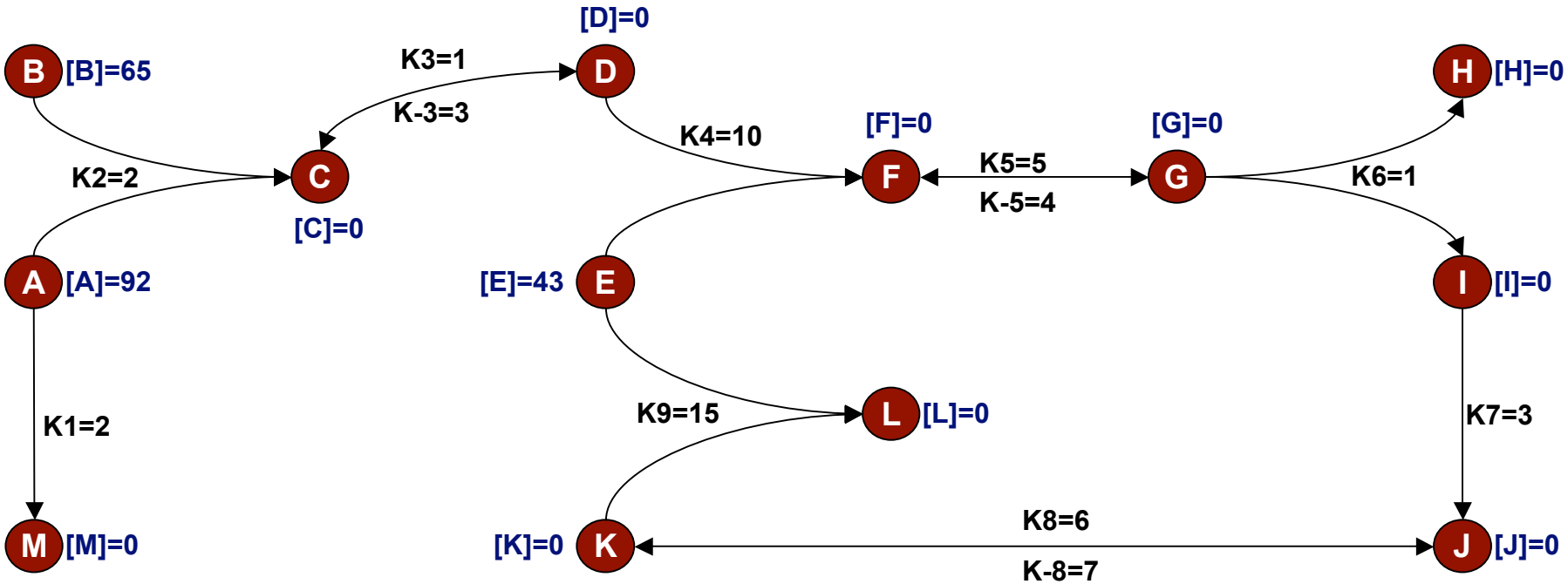
Why model?

- Simplistic answers:
  - Because it's there…
  - Why not?
- Technical answer:
  - "The benefit of formal mathematical models is that they can show whether proposed causal mechanisms are at least theoretically feasible and can help to suggest experiments that might further discriminate between alternatives." (Franks & Tofts, 1994)
- Realistic answers:
  - A computer model can generate new insights
  - A computer model can make testable predictions
  - A computer model can test conditions that may be difficult to study in the laboratory
  - A computer model can rule out particular explanations for an experimental observation
  - A computer model can help you identify what's right and wrong with your hypotheses (could/is the proposed mechanism correct)

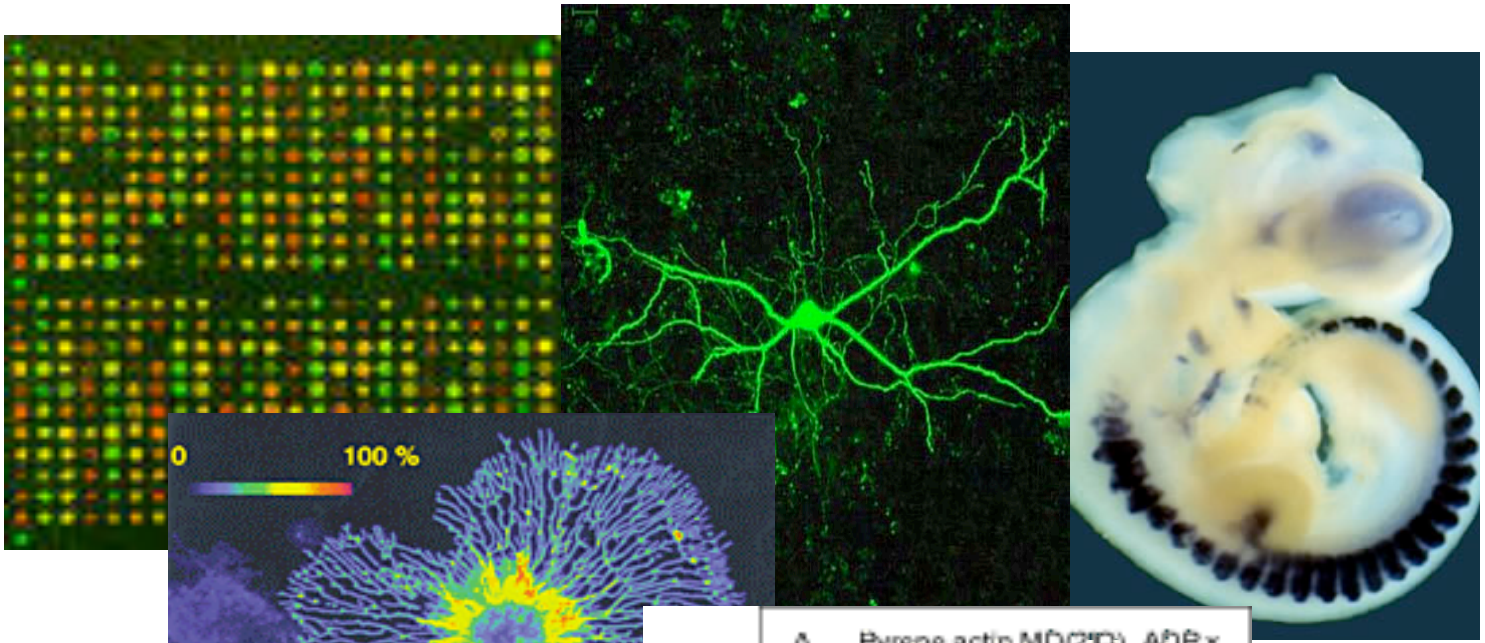
Slide from
Richard Orton

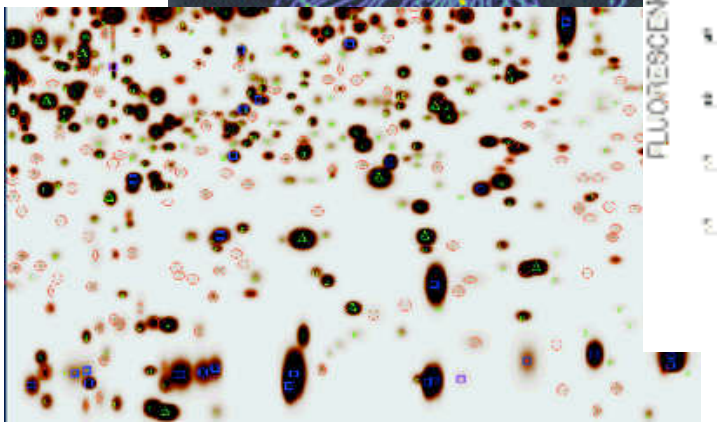
Why model?

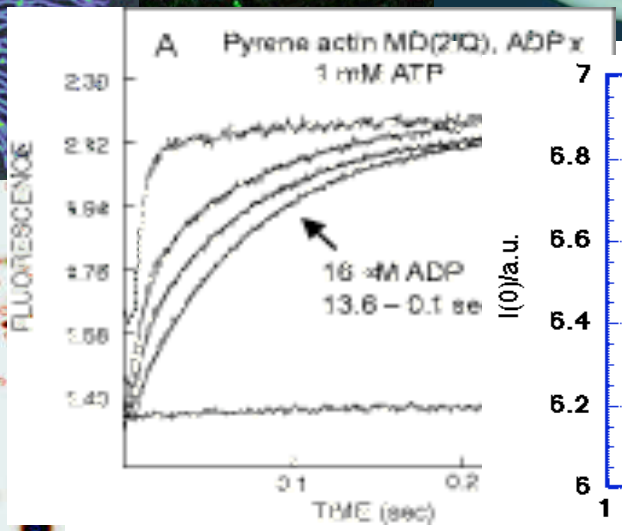
- In a complex pathway, knowing all the proteins involved and what they do, may still not tell you how the pathway works
- Furthermore, if all the initial concentrations and rate constants are known in the pathway, a computer simulation will probably still be needed to show how the system behaves over time

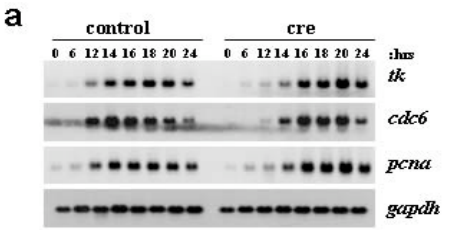


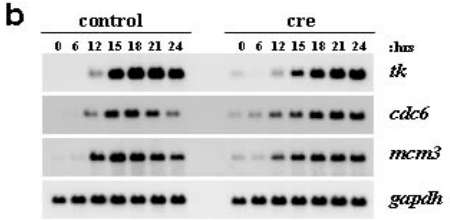
Biology = Concentrations

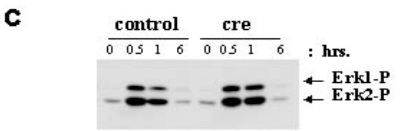


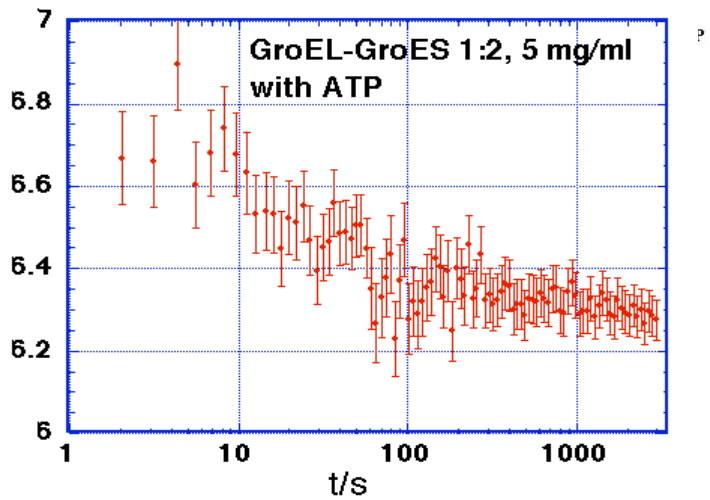












Modelling dynamic beha

...but biological systems contain

- non-linear interaction between components
- positive and negative feedback loops
- complex cross-talk phenomena

The simplest chemical reaction

$$A \rightarrow B$$

- irreversible, one-molecule reaction
- examples: all sorts of decay processes, e.g. radioactive, fluorescence, activated receptor returning to inactive state
- any metabolic pathway can be described by a combination of processes of this type (including reversible reactions and, in some respects, multi-molecule reactions)

The simplest chemical reaction

$$A \rightarrow B$$

various levels of description:

- homogeneous system, large numbers of molecules = ordinary differential equations, **kinetics**
- small numbers of molecules = probabilistic equations, **stochastics**
- spatial heterogeneity = partial differential equations, **diffusion**
- small number of heterogeneously distributed molecules = single-molecule tracking (e.g. cytoskeleton modelling)

Kinetics Description

Main idea: Molecules don't talk

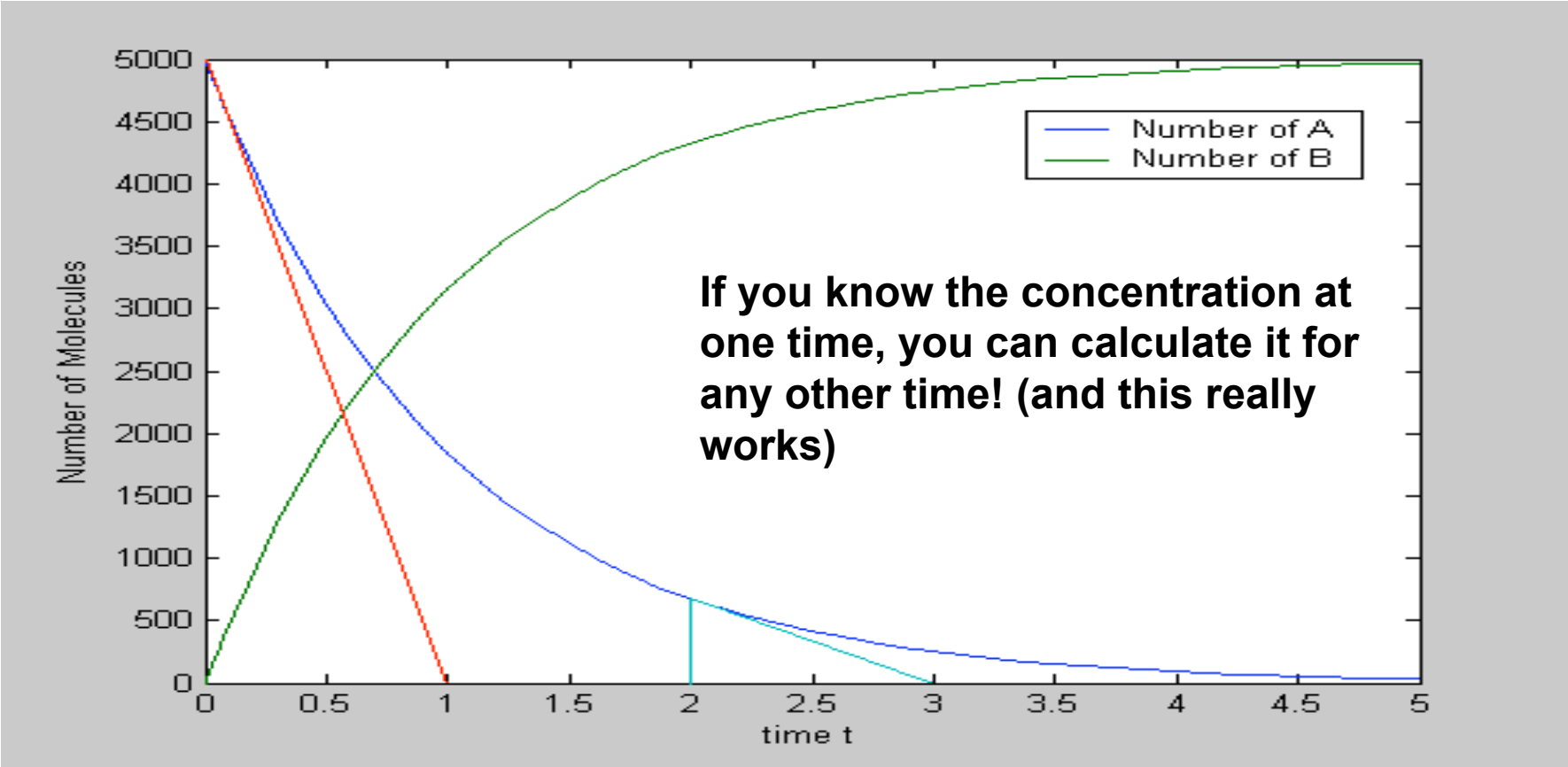
- Imagine a box containing N molecules.
  How many will decay during time t? k*N
- Imagine two boxes containing N/2 molecules each.
  How many decay? k*N
- Imagine two boxes containing N molecules each.
  How many decay? 2k*N
- In general:

$$-\frac{dn(t)}{dt} = \lambda * n(t) \quad \Leftrightarrow \quad n(t) = N_0 e^{-\lambda t}$$

differential equation (ordinary, linear, first-order)

exact solution (in more complex cases replaced by a numerical approximation)

Kinetics Description



Probabilistic Description

Main idea: Molecules are isolated entities without memory

Probability of decay of a single molecule in some small time interval:

$$p_1 = \lambda \Delta t$$

Probability of survival in Δt:

$$p_2 = 1 - p_1 = 1 - \lambda \Delta t$$

Probability of survival for some time t:

$$p = \lim_{x \to \infty} (1 - \lambda \frac{t}{x})^x = e^{-\lambda t}$$

Transition to large number of molecules:

$$n(t) = N_0 e^{-\lambda t} \quad \text{or}$$

$$\frac{dn(t)}{dt} = -\lambda N_0 e^{-\lambda t} = -\lambda n(t)$$

Probabilistic Description – 2

Probability of survival of a single molecule for some time t:

$$p = \lim_{x \to \infty}(1 - \lambda \frac{t}{x})^x = e^{-\lambda t}$$

Probability that exactly x molecules survive for some time t:

$$p_x = (e^{-\lambda t})^x (1 - e^{-\lambda t})^{N_0 - x} \begin{pmatrix} N_0 \\ x \end{pmatrix}$$

Most likely number to survive to time t:

$$\max(x \mid p_x) = N_0 e^{-\lambda t}$$

Spatial heterogeneity

- concentrations are different in different places, n = f(t,x,y,z)
- diffusion superimposed on chemical reactions:

$$\frac{\partial n(t)_{xyz}}{\partial t} = -\lambda n(t)_{xyz} \pm \text{diffusion}$$

- partial differential equation

Summary of Physical Chemistry

- Simple reactions are easy to model accurately
- Kinetic, probabilistic, Markovian approaches lead to the same basic description

$$\frac{dn(t)}{dt} = -\lambda n(t) \Leftrightarrow n(t) = N_0 e^{-\lambda t}$$

- Diffusion leads only to slightly more complexity
- Next step: Everything is decay...

Some (Bio)Chemical Conventions

Concentration of Molecule A = [A], usually in units mol/litre (molar)

Rate constant = k, with indices indicating constants for various reactions (k_1, k_2...)

Therefore:

$$A \rightarrow B$$

$$\frac{d[A]}{dt} = -\frac{d[B]}{dt} = -k_1[A]$$

Description in MATLAB: 1. Simple Decay Reaction

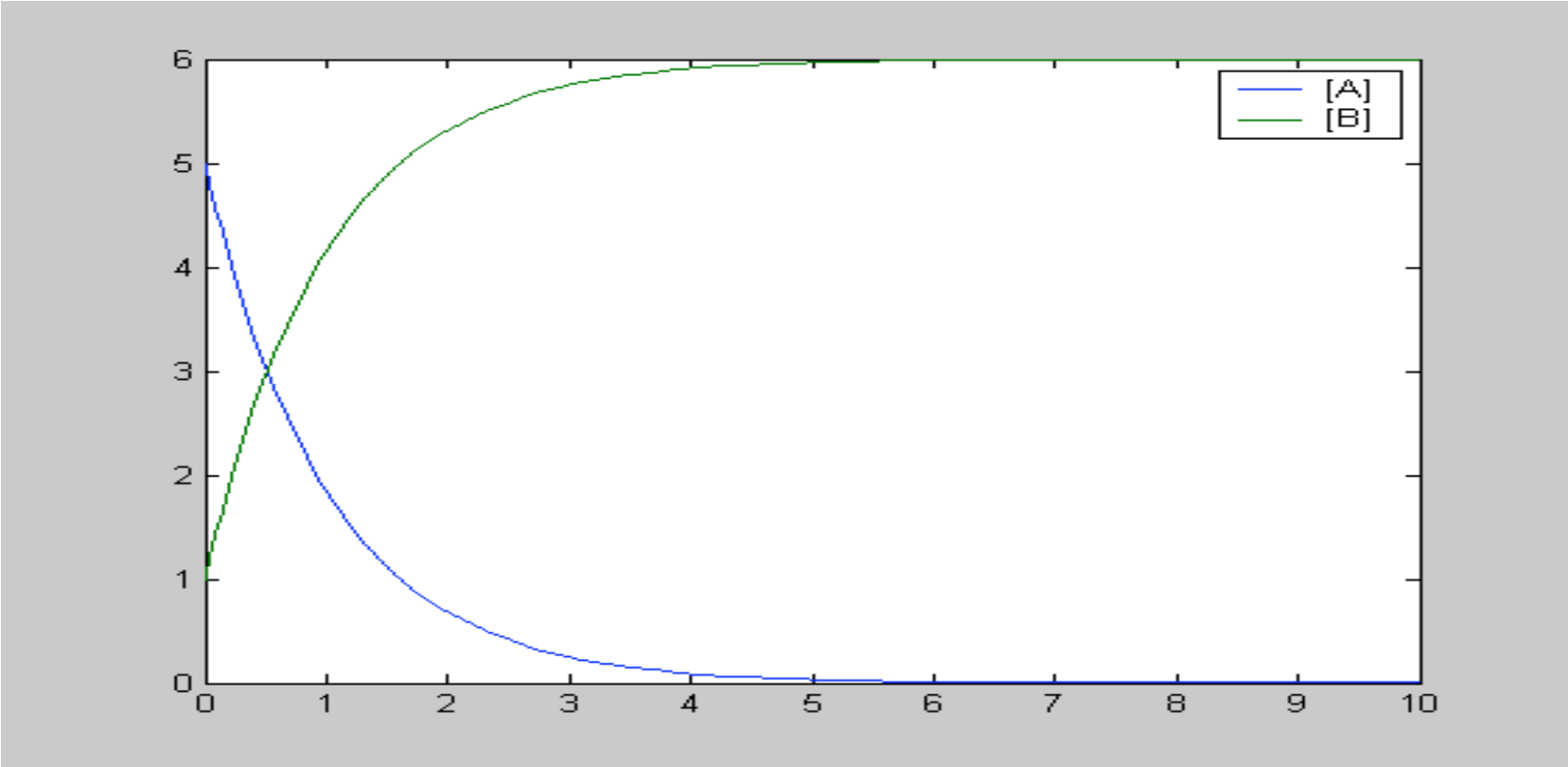
M-file (description of the model)

```
function dydt = decay(t, y)
% A -> B   or   y(1) -> y(2)
k = 1;
dydt = [-k*y(1)
         k*y(1)];
```

Analysis of the model

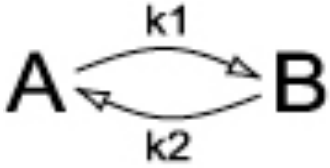
```
>> [t y] = ode45(@decay, [0 10], [5 1]);
>> plot (t, y);
>> legend ('[A]', '[B]');
```

Decay Reaction in MATLAB



Reversible, Single-Molecule Reaction

A ↔ B, or



Differential equations:

forward reverse

$$\frac{d[A]}{dt} = -k_1[A] + k_2[B]$$

$$\frac{d[B]}{dt} = k_1[A] - k_2[B]$$

Main principle: Partial reactions are **independent**!

Reversible, single-molecule reaction – 2

Differential Equation:

$$\frac{d[A]}{dt} = -k_1[A] + k_2[B]$$

$$\frac{d[B]}{dt} = k_1[A] - k_2[B]$$

Equilibrium (=steady-state):

$$\frac{d[A]_{equi}}{dt} = \frac{d[B]_{equi}}{dt} = 0$$

$$-k_1[A]_{equi} + k_2[B]_{equi} = 0$$

$$\frac{[A]_{equi}}{[B]_{equi}} = \frac{k_2}{k_1} = K_{equi}$$

Description in MATLAB: 2. Reversible Reaction

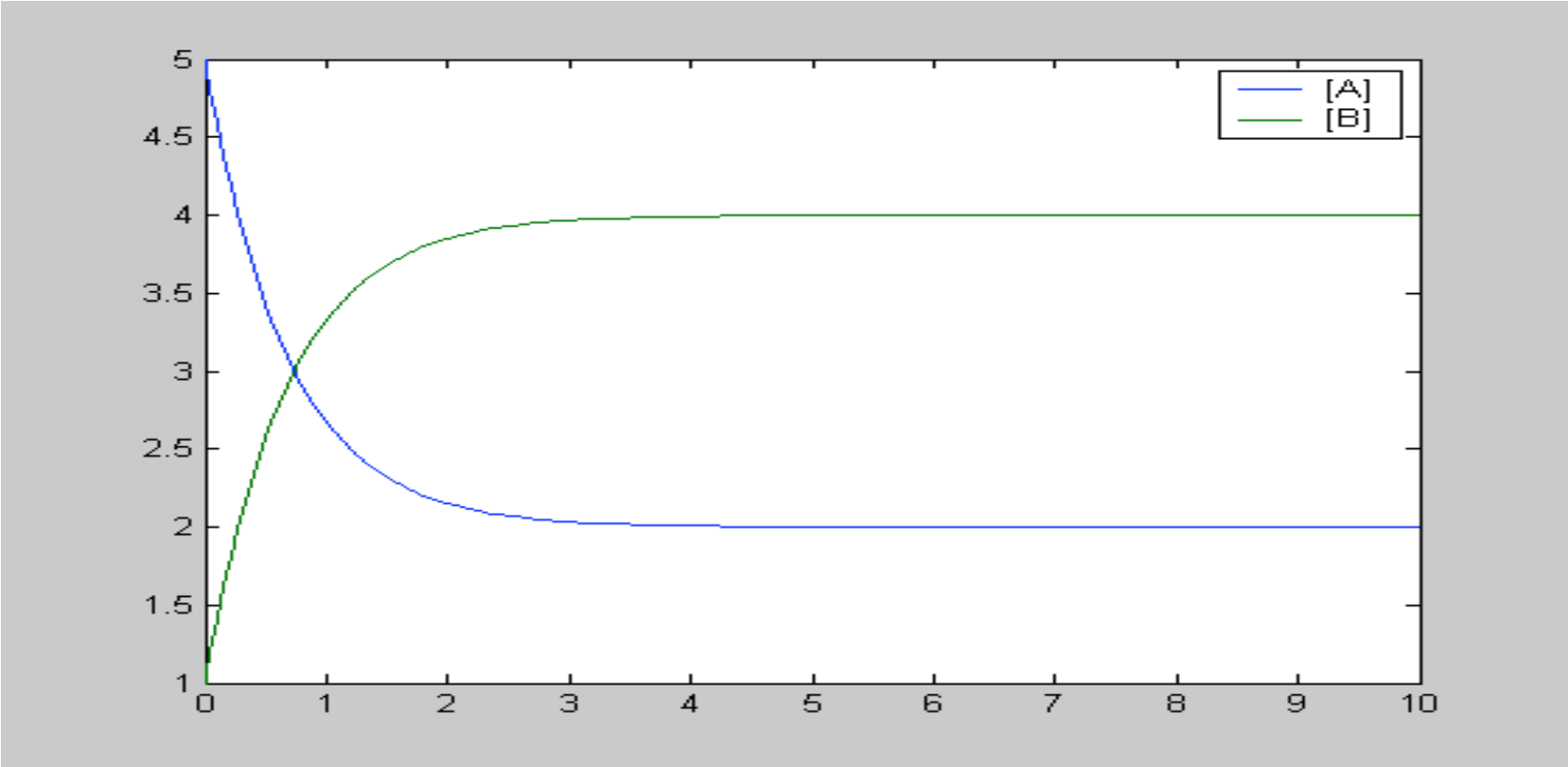
M-file (description of the model)

```
function dydt = isomerisation(t, y)
% A <-> B    or    y(1) <-> y(2)
k1 = 1;
k2 = 0.5;
dydt = [-k1*y(1)+k2*y(2)     % d[A]/dt
        k1*y(1)-k2*y(2)      % d[B]/dt
        ];
```

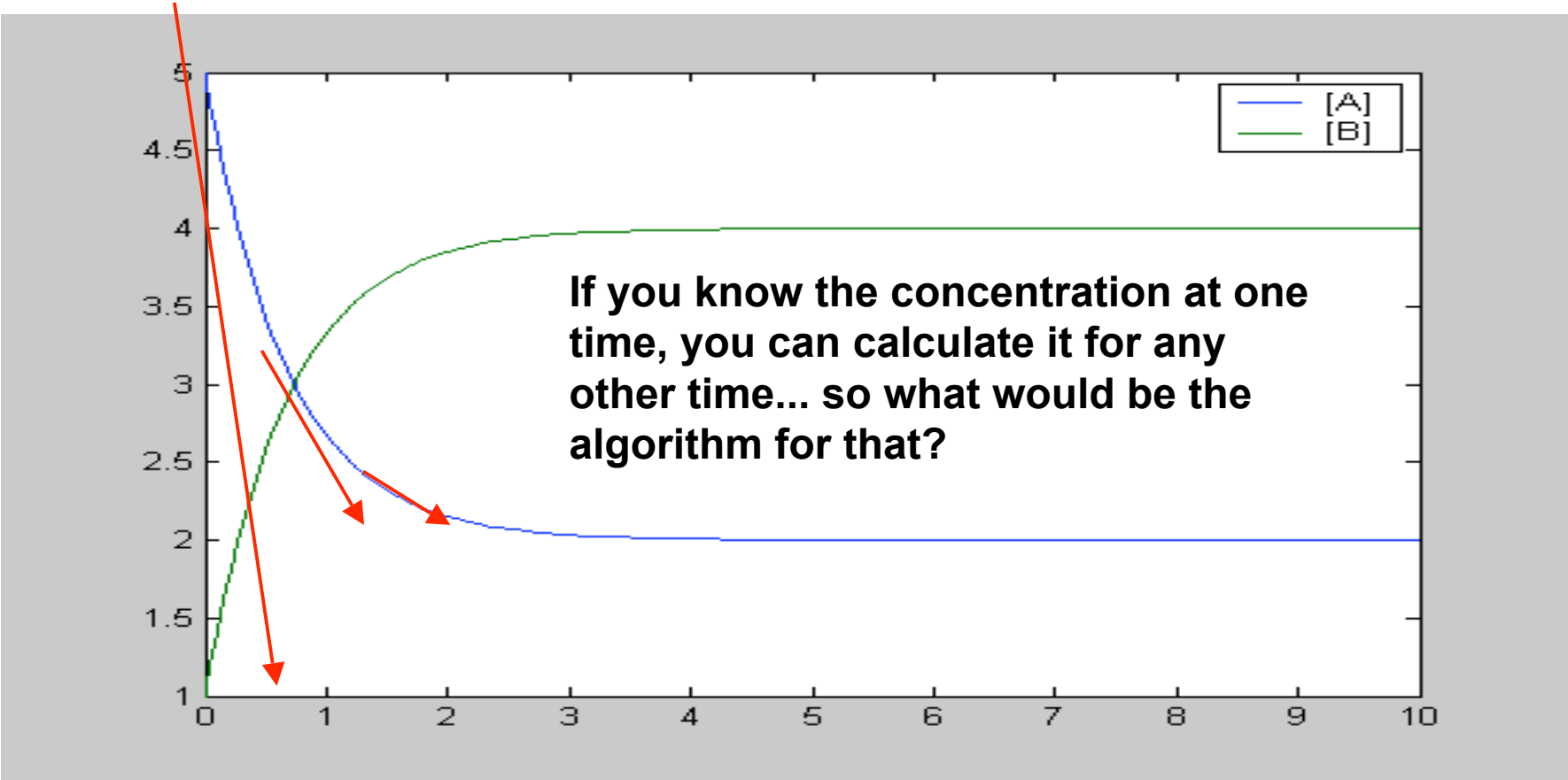
Analysis of the model

```
>> [t y] = ode45(@isomerisation, [0 10], [5
  1]);
>> plot (t, y);
>> legend ('[A]', '[B]');
```

Isomerization Reaction in MATLAB



Isomerization Reaction in MATLAB



Euler's method - pseudocode

$$y_{n+1} = y_n + hf(t_n, y_n)$$

```
1.      define f(t,y)
2.      input t0 and y0.
3.      input h and the number of steps, n.
4.      for j from 1 to n do
        a.      m = f(t0,y0)
        b.      y1 = y0 + h*m
        c.      t1 = t0 + h
        d.      Print t1 and y1
        e.      t0 = t1
        f.      y0 = y1
5.      end
```

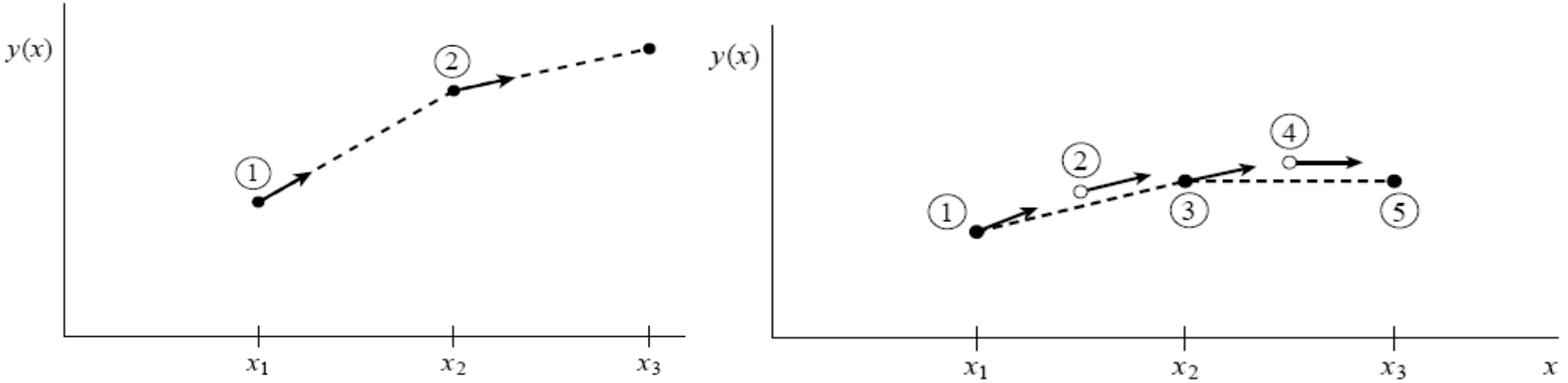
Where
One step of Euler's integration from t_n to $t_n+1 = t_n + h$ is:
$Y_{n+1} = y_n + hf(t_n,y_n)$ where h is the (time) step and
$f(t_n,y_n)$is the differential equation

Improving Euler's method

$$y_{n+1} = y_n + hf(t_n, y_n)$$

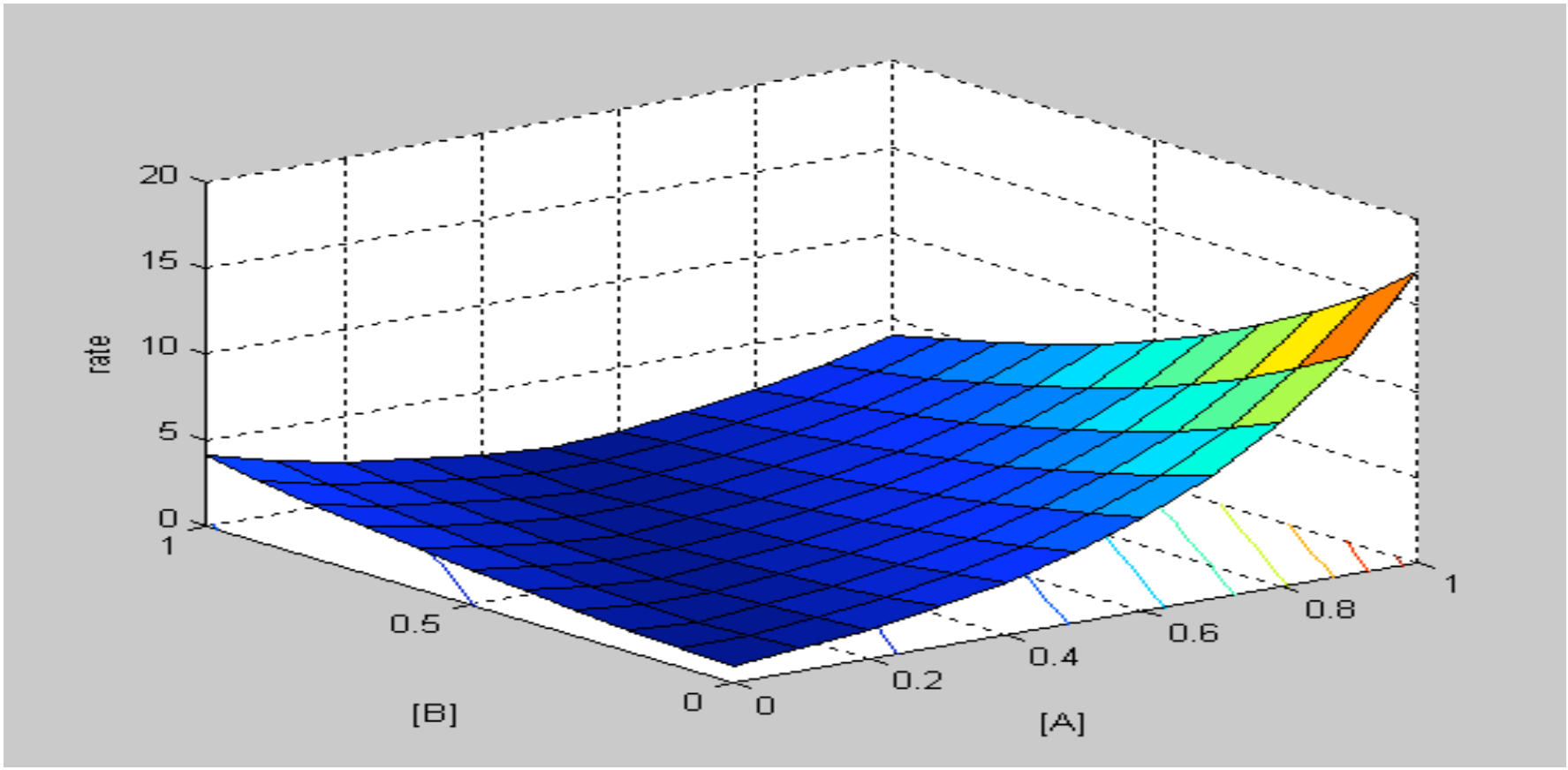
$$\downarrow$$

$$y_{n+1} = y_n + hf\left(t_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}hf(t_n, y_n)\right)$$



(second-order Runge-Kutta method)

Isomerization Reaction in MATLAB



Irreversible, two-molecule reaction

The last piece of the puzzle

A+B→C

Differential equations:

$$\frac{d[A]}{dt} = \frac{d[B]}{dt} = -\frac{d[C]}{dt}$$

$$\frac{d[A]}{dt} = -k[A][B]$$

Non-linear!

Underlying idea: Reaction probability = Combined probability that both [A] and [B] are in a "reactive mood":

$$p(AB) = p(A)\,p(B) = k_1^*[A]k_2^*[B] = k[A][B]$$

A simple metabolic pathway

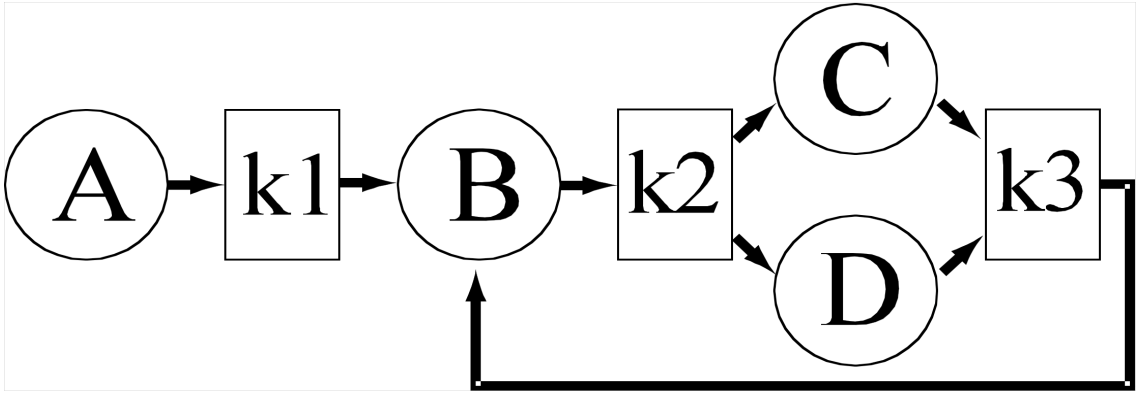
A→B←→C+D

Differential equations:

d/dt	decay	forward	reverse
[A]=	`-k1[A]`		
[B]=	`+k1[A]`	`-k2[B]`	`+k3[C][D]`
[C]=		`+k2[B]`	`-k3[C][D]`
[D]=		`+k2[B]`	`-k3[C][D]`

Metabolic Networks as Bigraphs

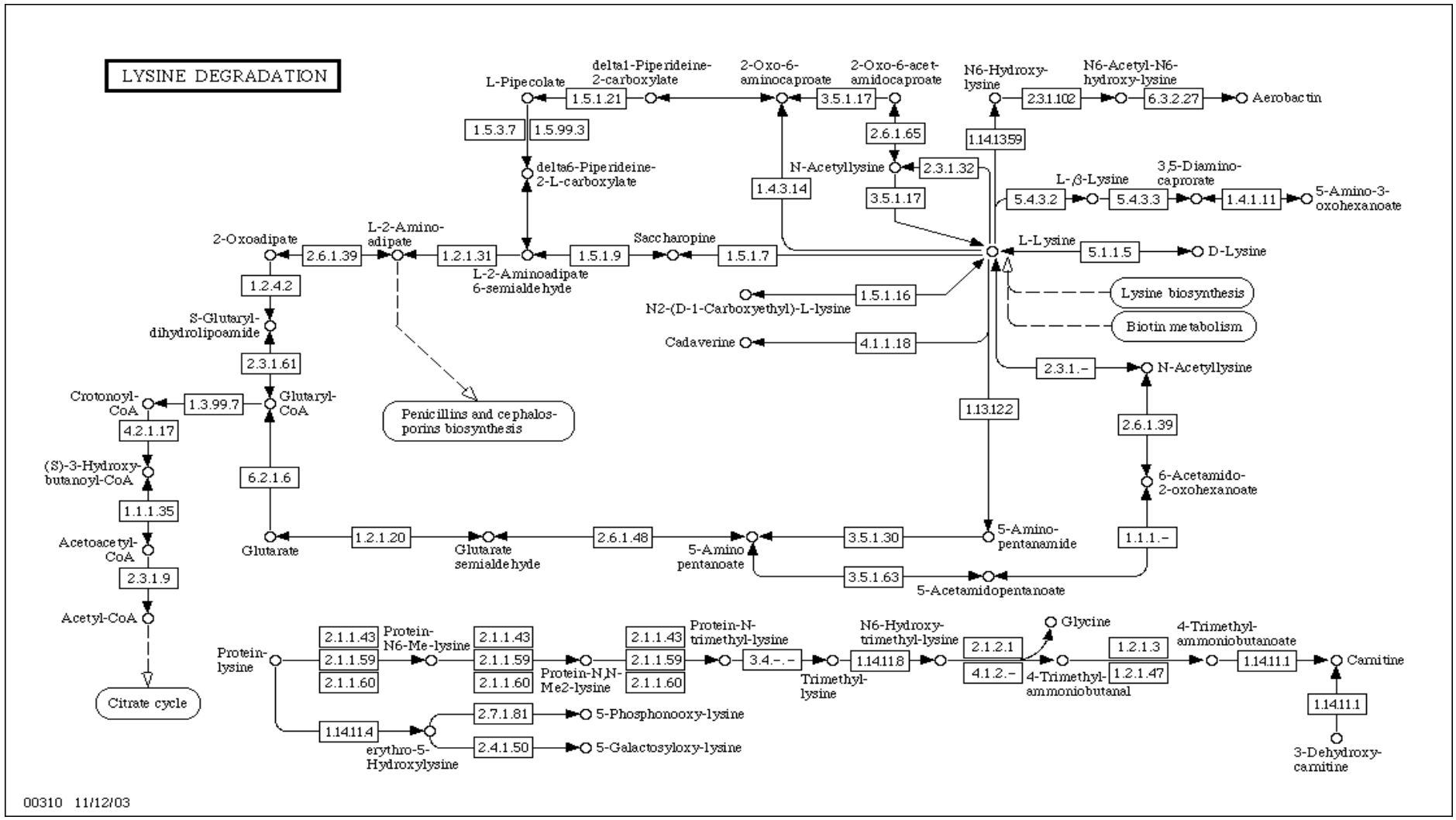
A→B←→C+D



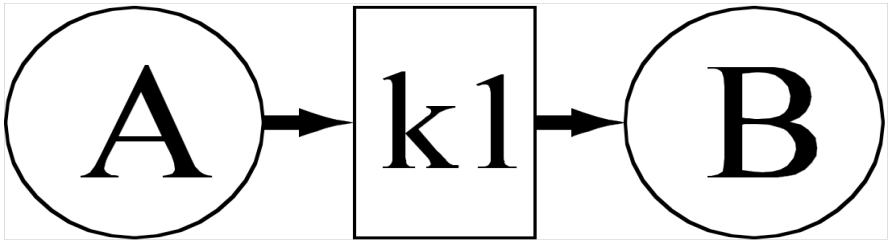
	k1	k2	k3
A	**-1**	**0**	**0**
B	**1**	**-1**	**1**
C	**0**	**1**	**-1**
D	**0**	**1**	**-1**

d/dt	decay	forward	reverse
[A]	**-k1[A]**		
[B]	**+k1[A]**	**-k2[B]**	**+k3[C][D]**
[C]		**+k2[B]**	**-k3[C][D]**
[D]		**+k2[B]**	**-k3[C][D]**

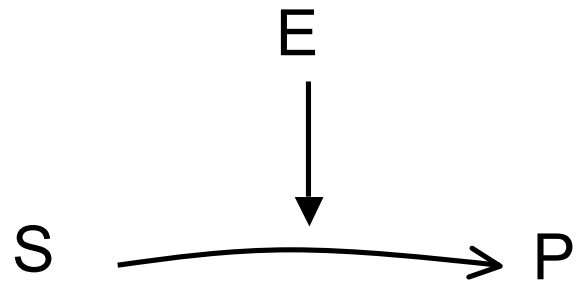
Biological description → bigraph → differential equations

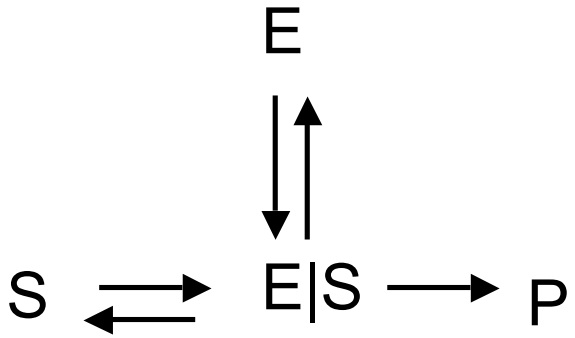


Biological description → bigraph → differential equations



Mass action



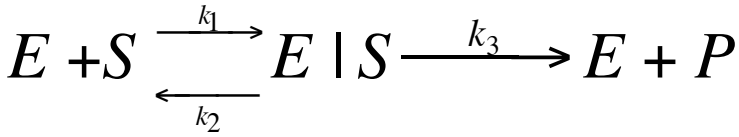


- S: substrate,
- P: product
- E: enzyme
- E|S substrate-enzyme complex

$$E + S \underset{k_2}{\overset{k_1}{\rightleftharpoons}} E \mid S \xrightarrow{k_3} E + P$$

Mass action equations

1. E + S -(k1)→ E|S
2. E|S -(k2)→ E+S
3. E|S -(k3)→ E+P



OR

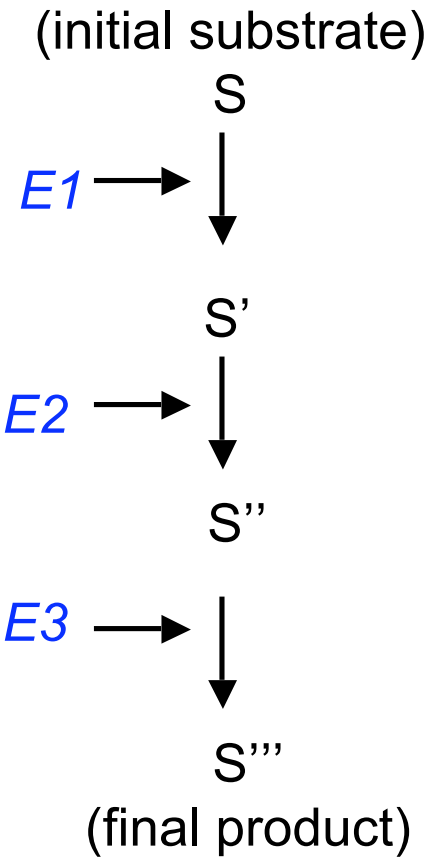
1. E + S =(k1/k2)= E|S
2. E|S -(k3)→ E+P

?Can you code the differential equations?

Metabolic pathways vs Signalling Pathways

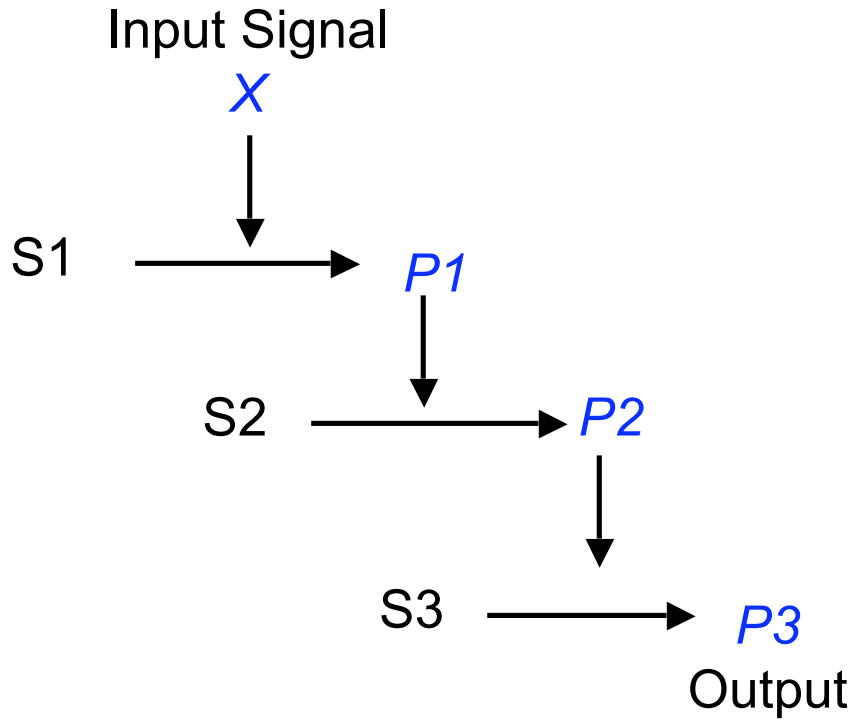
(can you give the mass-action equations?)

Metabolic



Classical enzyme-product pathway

Signalling cascade

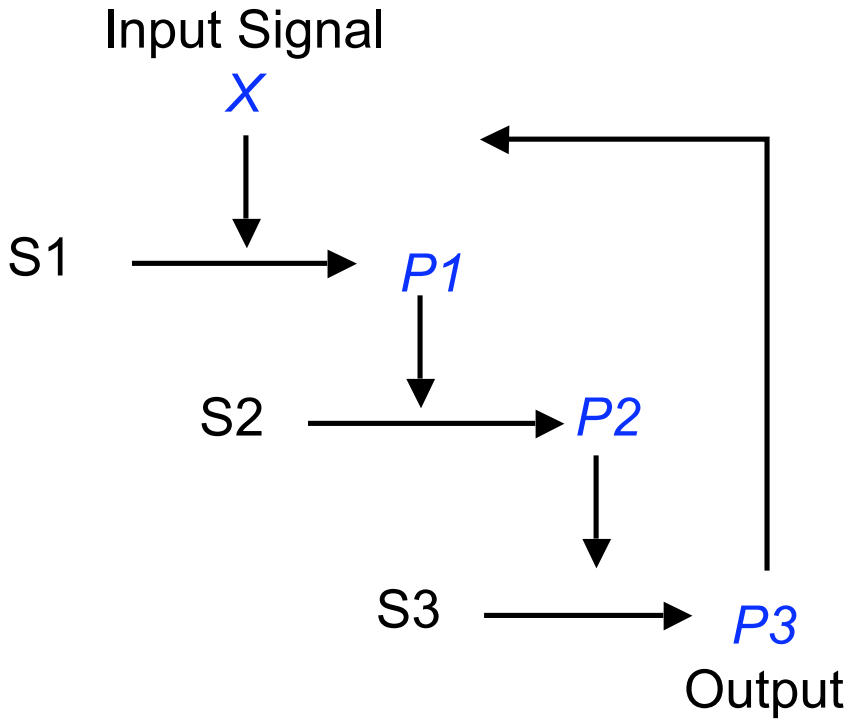


Product become enzyme at next stage

Feedback loops (signalling cascades)

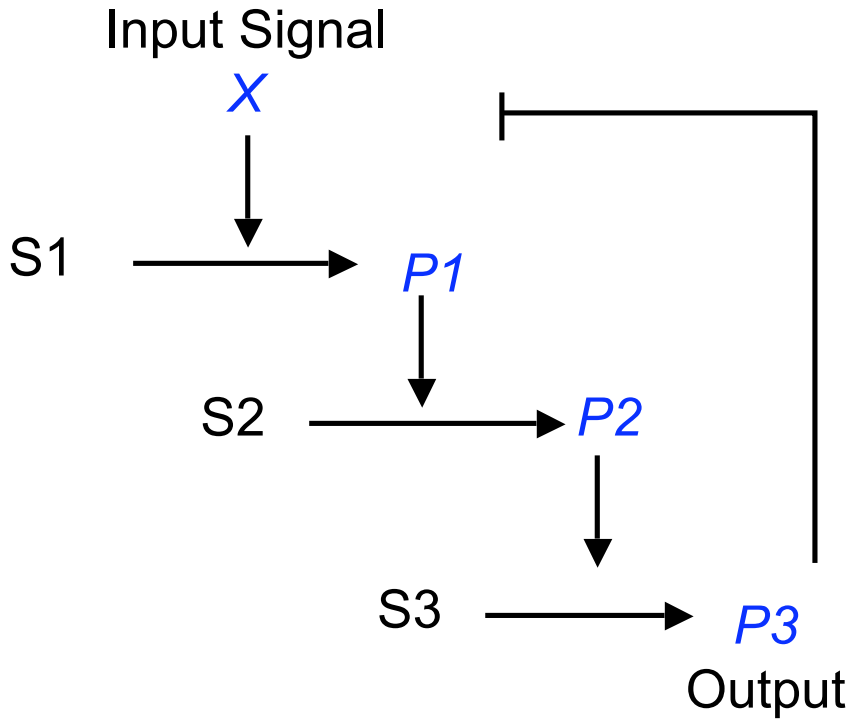
P3 + S1 = P3|S1 →P3+P1

Positive feedback



P3 + X = P3|X

Negative feedback



Biological description → bigraph → differential equations

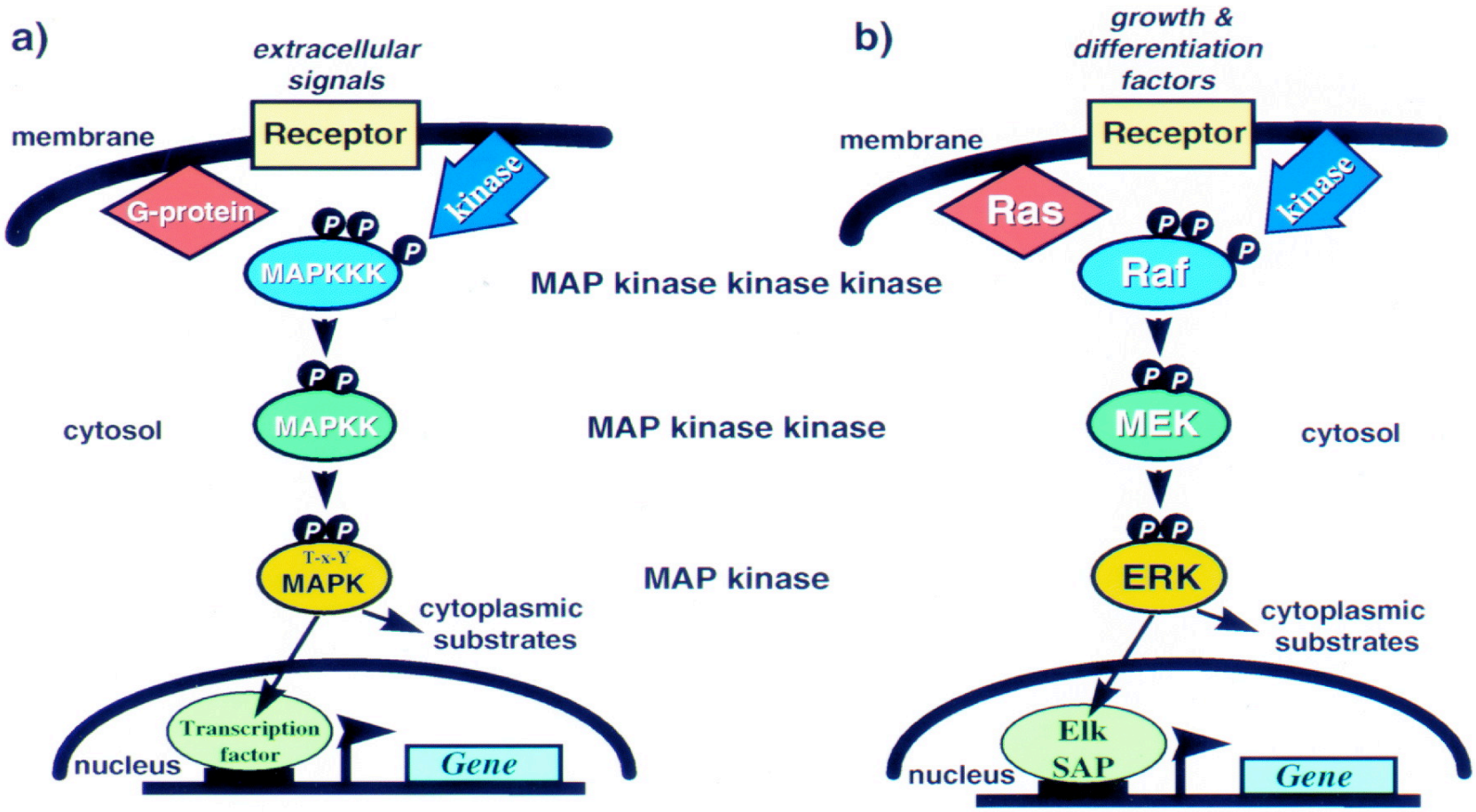
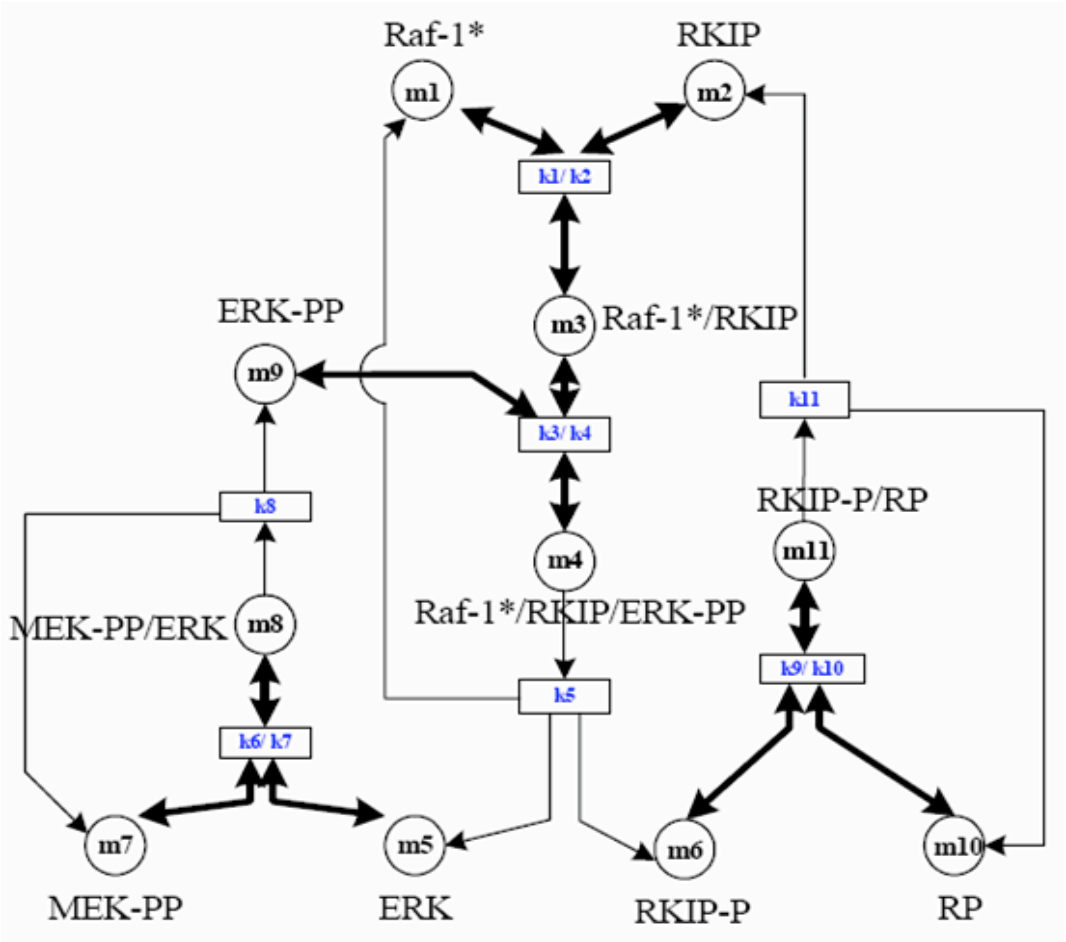


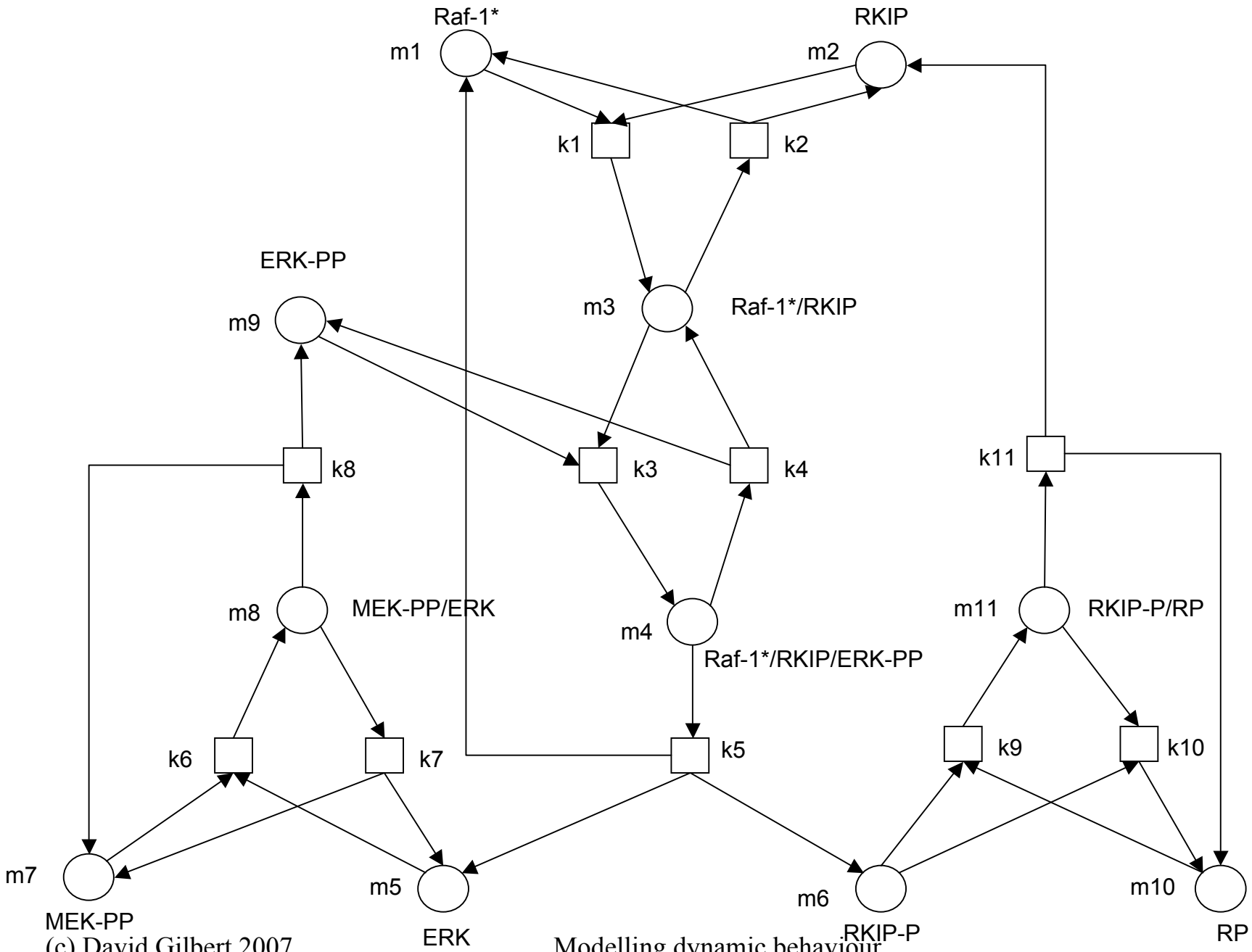
Fig. courtesy of W. Kolch

The Raf-1/RKIP/ERK pathway

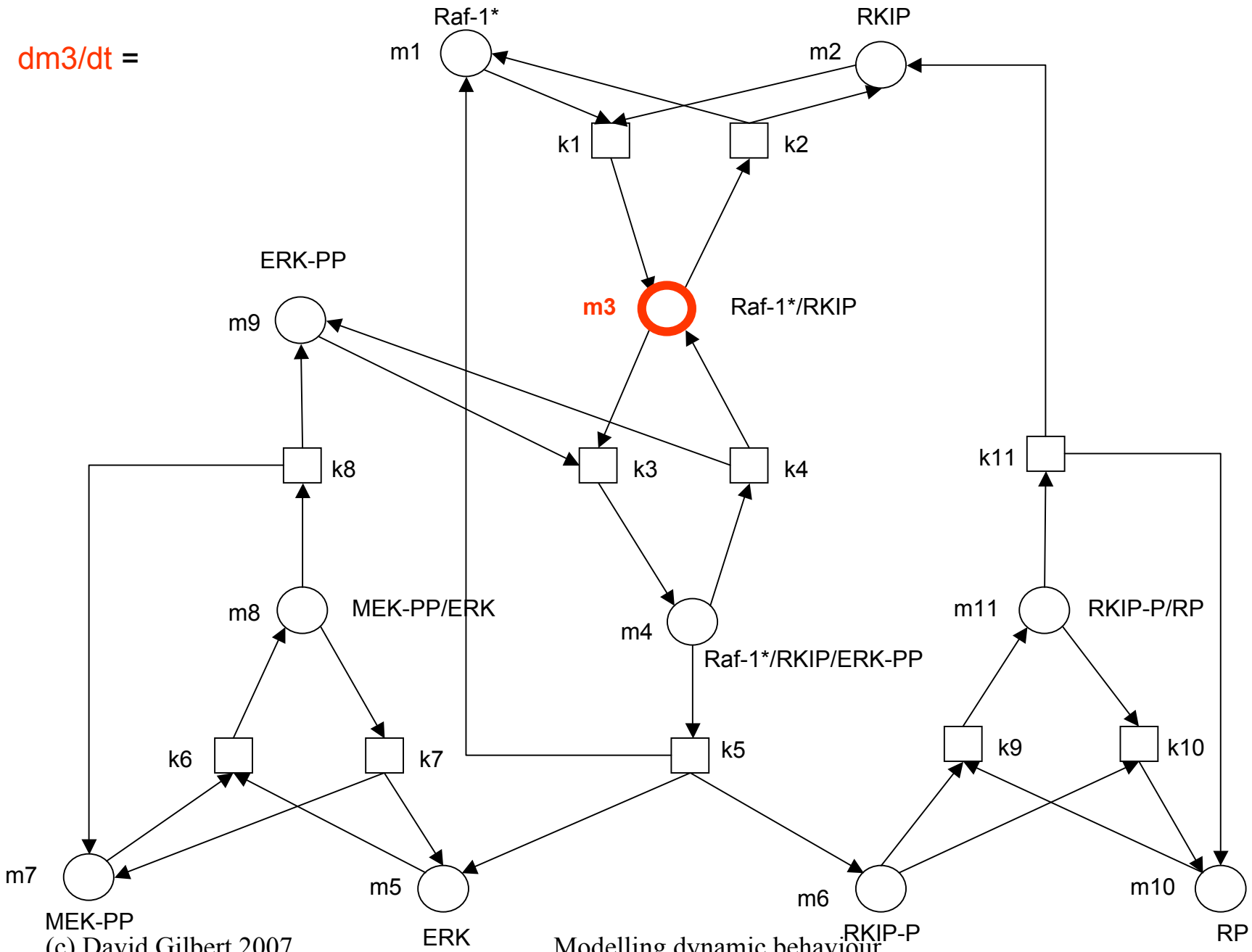


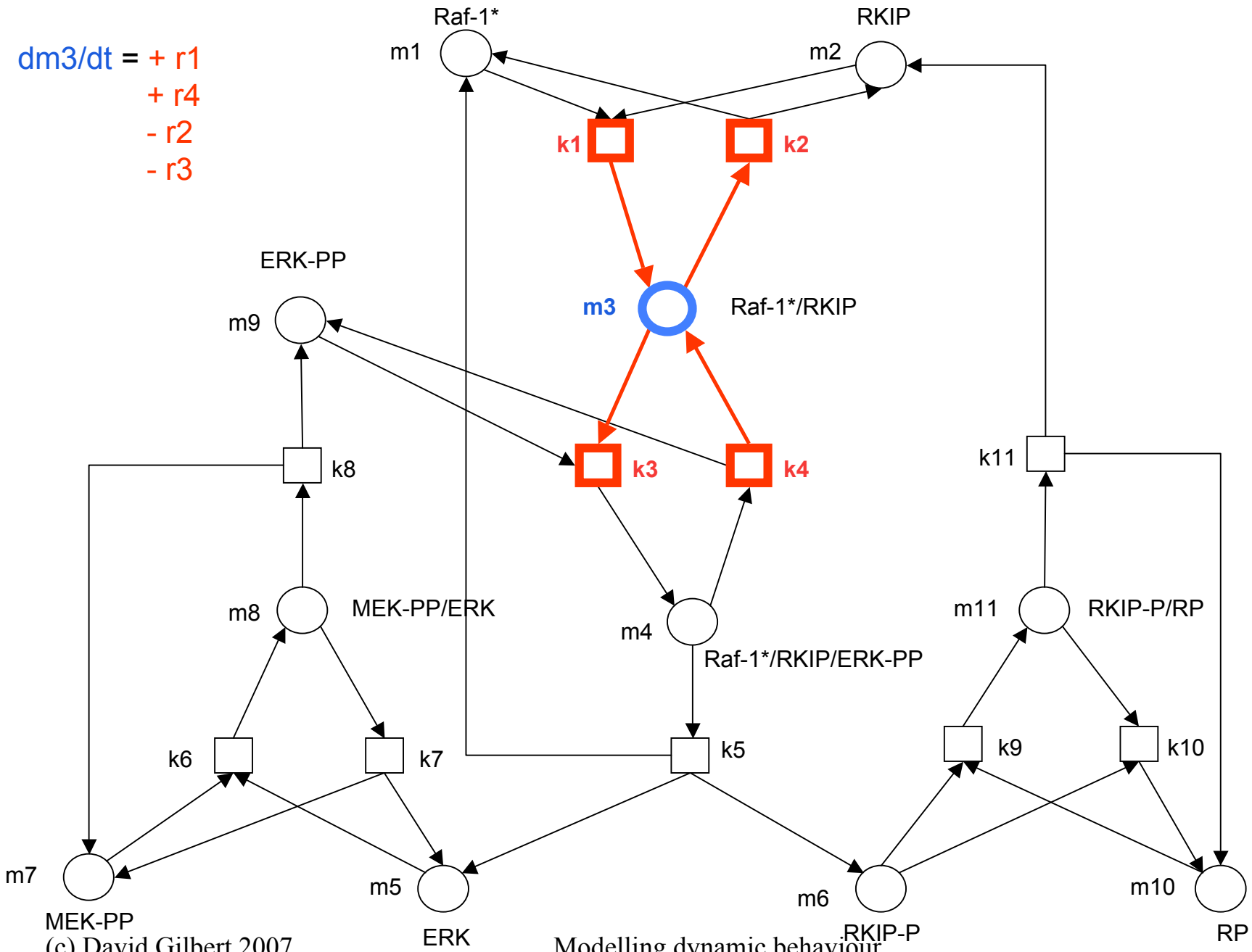
Can you model it?

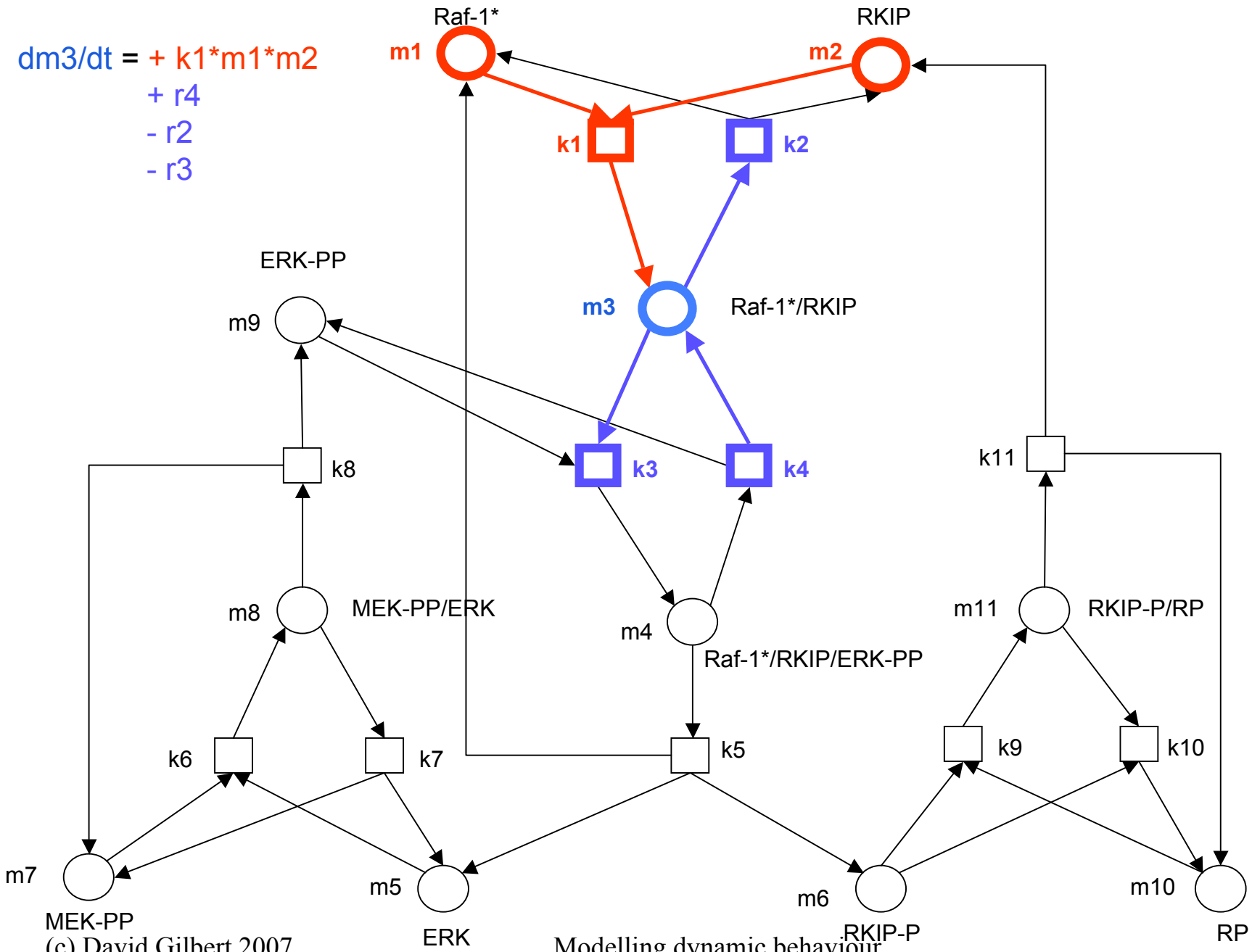
(11x11 table, 34 entries)

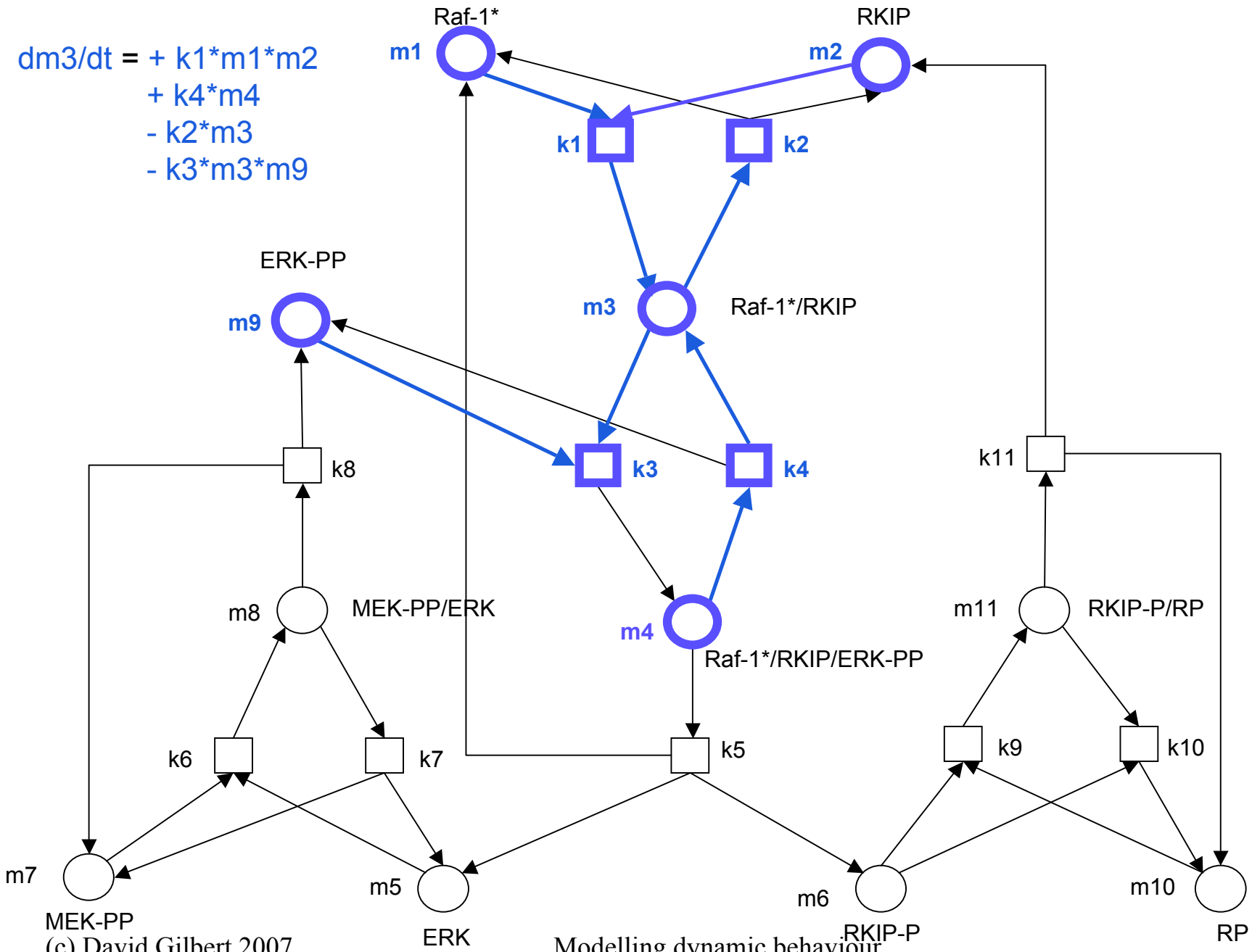

Raf-1*
RKIP
m1
m2
k1
k2
ERK-PP
m3
Raf-1*/RKIP
m9
k8
k3
k4
k11
m8
MEK-PP/ERK
m4
Raf-1*/RKIP/ERK-PP
m11
RKIP-P/RP
k6
k7
k5
k9
k10
m7
m5
m6
m10
MEK-PP
ERK
RKIP-P
RP

dm3/dt =


Raf-1*
m1
RKIP
m2
k1
k2
ERK-PP
m9
m3
Raf-1*/RKIP
k8
k3
k4
k11
m8
MEK-PP/ERK
m4
Raf-1*/RKIP/ERK-PP
m11
RKIP-P/RP
k6
k7
k5
k9
k10
m7
MEK-PP
m5
ERK
m6
RKIP-P
m10
RP


dm3/dt = + r1
+ r4
- r2
- r3
Raf-1*
RKIP
m1
m2
k1
k2
ERK-PP
m3
Raf-1*/RKIP
m9
k3
k4
k8
k11
m8
MEK-PP/ERK
m4
Raf-1*/RKIP/ERK-PP
m11
RKIP-P/RP
k6
k7
k5
k9
k10
m7
m5
m6
m10
MEK-PP
ERK
RKIP-P
RP


dm3/dt = + k1*m1*m2
+ r4
- r2
- r3
Raf-1*
RKIP
m1
m2
k1
k2
ERK-PP
m3
Raf-1*/RKIP
m9
k3
k4
k8
k11
m8
MEK-PP/ERK
m4
Raf-1*/RKIP/ERK-PP
m11
RKIP-P/RP
k6
k7
k5
k9
k10
m7
m5
m6
m10
MEK-PP
ERK
RKIP-P
RP


dm3/dt = + k1*m1*m2
+ k4*m4
- k2*m3
- k3*m3*m9
Raf-1*
RKIP
m1
m2
k1
k2
ERK-PP
m3
Raf-1*/RKIP
m9
k3
k4
k8
k11
m8
MEK-PP/ERK
m4
Raf-1*/RKIP/ERK-PP
m11
RKIP-P/RP
k6
k7
k5
k9
k10
m7
m5
m6
m10
MEK-PP
ERK
RKIP-P
RP

Description in MATLAB: 3. The RKIP/ERK pathway

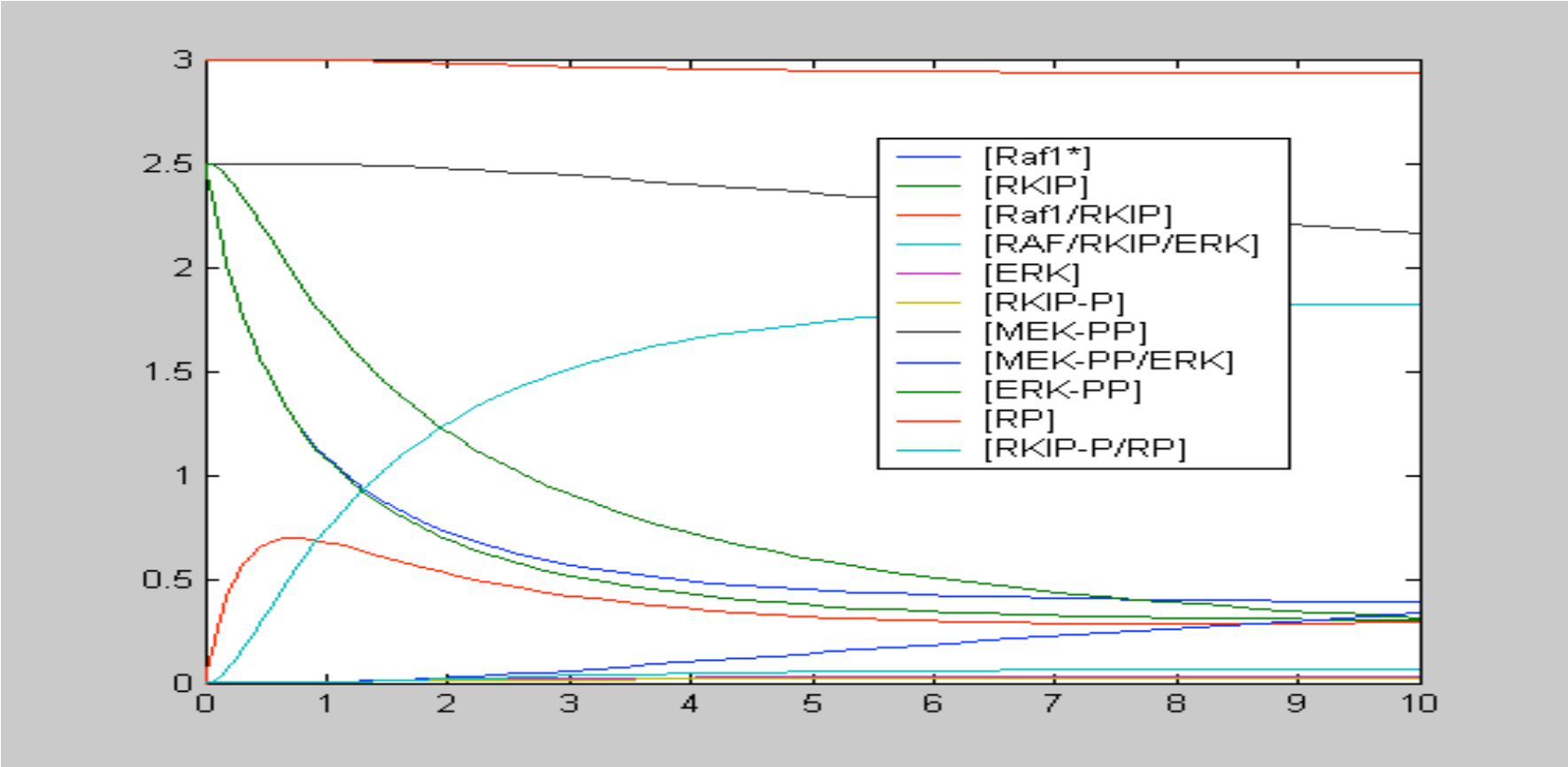
```
function dydt = erk_pathway_wolkenhauer(t, y)
% from Kwang-Hyun Cho et al., Mathematical Modeling...
k1 = 0.53;
k2 = 0.0072;
k3 = 0.625;
k4 = 0.00245;
k5 = 0.0315;
k6 = 0.8;
k7 = 0.0075;
k8 = 0.071;
k9 = 0.92;
k10 = 0.00122;
k11 = 0.87;
```

Description in MATLAB: 3. The RKIP/ERK pathway

Analysis of the model:

```
>> [t y] =
  ode45(@erk_pathway_wolkenhauer, [0
  10], [2.5 2.5 0 0 0 0 2.5 0 2.5 3
  0]); %(initial values!)
>> plot (t, y);
>> legend ('[Raf1*]', '[RKIP]',
  '[Raf1/RKIP]', '[RAF/RKIP/ERK]',
  '[ERK]', '[RKIP-P]', '[MEK-PP]',
  '[MEK-PP/ERK]', '[ERK-PP]', '[RP]',
  '[RKIP-P/RP]' );
```

The RKIP/ERK pathway in MATLAB



Further Analyses in MATLAB et al.

All initial concentrations can be varied at will, e.g. to test a concentration series of one component (sensitivity analysis)

Effect of slightly different k-values can be tested (stability of the model with respect to measurement/estimation errors)

Effect of inhibitors of each reaction (changed k-values) can be predicted

Concentrations at each time-point are predicted exactly and can be tested experimentally

Example of Sensitivity Analysis

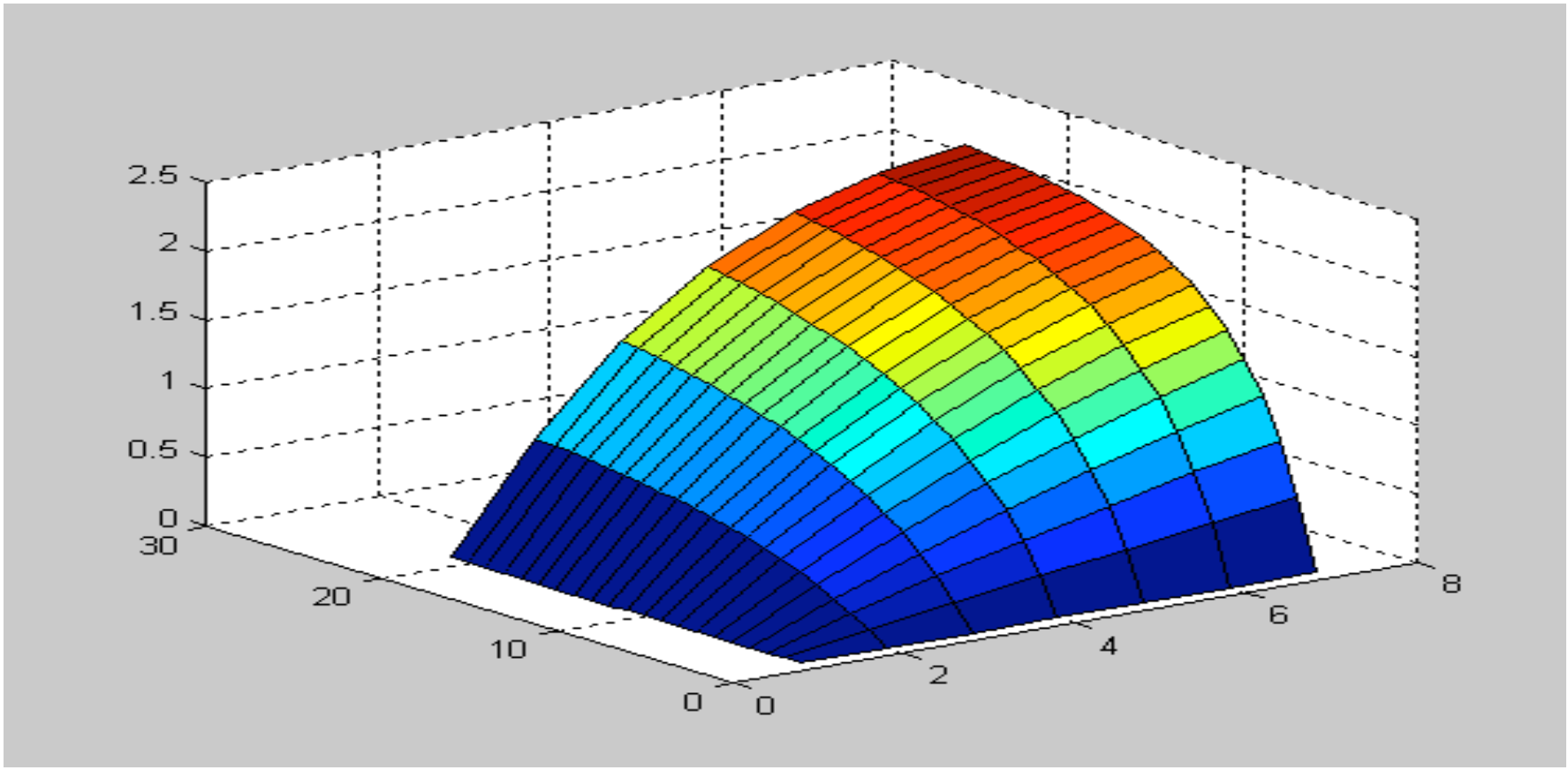
```
function [tt,yy] = sensitivity(f, range, initvec,
  which_stuff_vary, ep, step, which_stuff_show, timeres);

timevec = range(1):timeres:range(2);
vec = [initvec];
[tt y] = ode45(f, timevec, vec);
yy = y(:,which_stuff_show);

for i=initvec(which_stuff_vary)+step:step:ep;
    vec(which_stuff_vary) = i;
    [t y] = ode45(f, timevec, vec);
    tt = [t];
    yy = [yy y(:,which_stuff_show)];
end
```

Example of Sensitivity Analysis

```
>> [t y] =
  sensitivity(@erk_pathway_wolkenhau
  er, [0 1], [2.5 2.5 0 0 0 0 2.5 0
  2.5 3 0], 5, 6, 1, 8, 0.05);
>> surf (y);
```

varies concentration of m5 (ERK-PP) from 0..6, outputs concentration of m8 (ERK/MEK-PP), time range [0 1], steps of 0.05. Then plots a surface map.

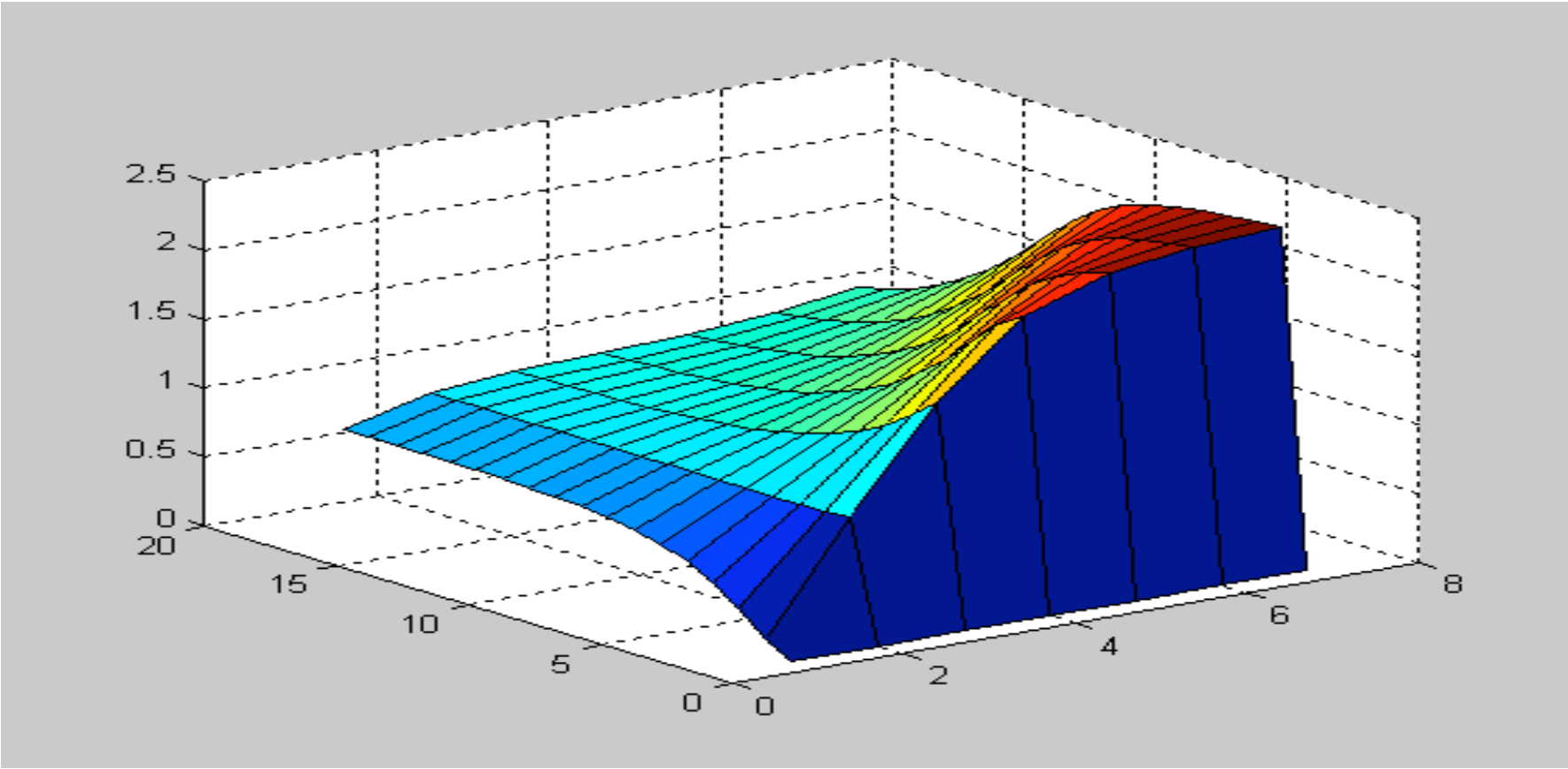
Example of Sensitivity Analysis



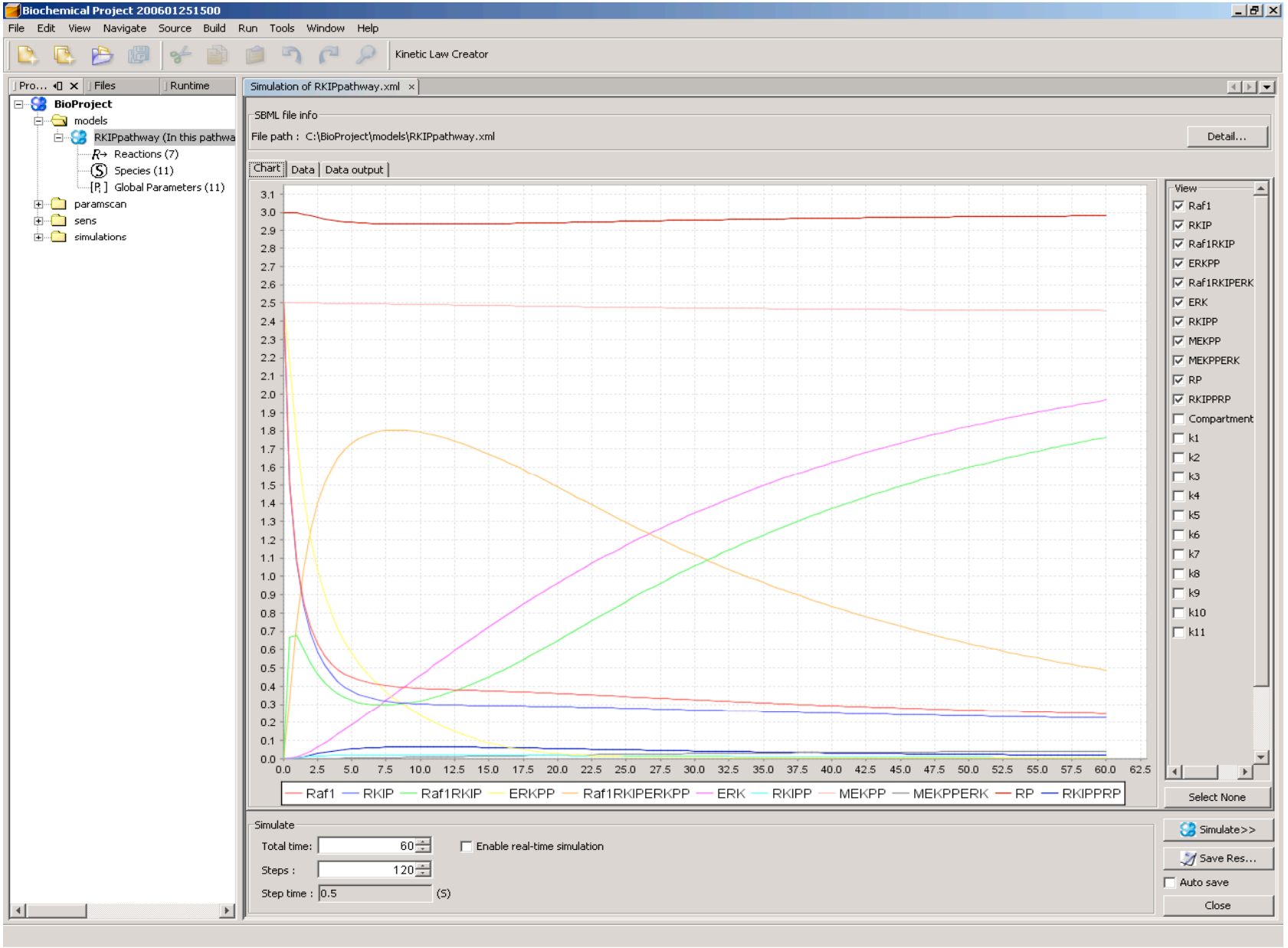
after Cho et al. (2003) CSMB

Example of Sensitivity Analysis

(longer time course)



Simulation in BioNessie

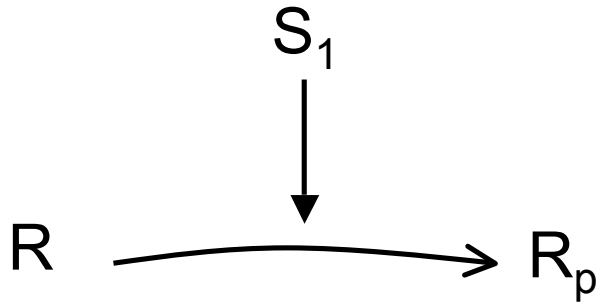


Modelling and modularisation

Example:

Signalling pathway cascades

Mass action for enzymatic reaction - phosphorylation

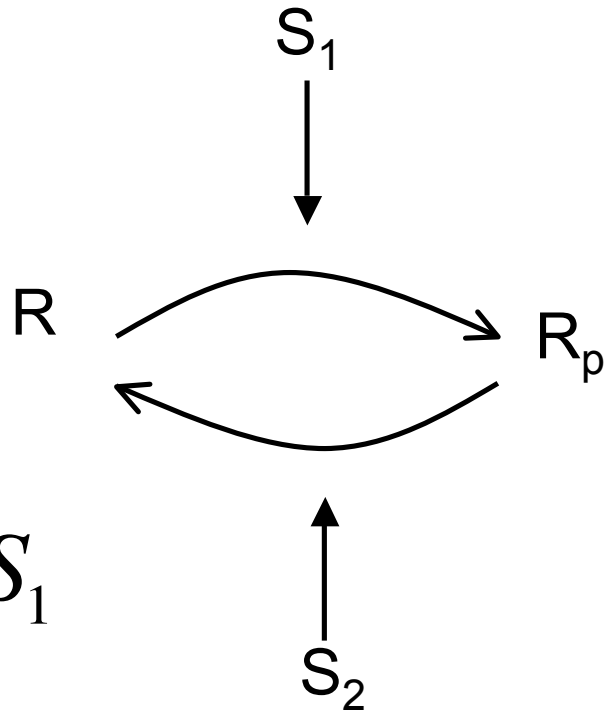


- R: substrate,
- R_p: product (phosphorylated R)
- S_1: enzyme (kinase)
- $R|S_1$ substrate-enzyme complex

$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftarrows}} R \mid S_1 \xrightarrow{k_3} R_p + S_1$$

Phosphorylation - dephosphorylation loop
Mass action model 1

- R: unphosphorylated form
- R_p: phosphorylated form
- S_1: kinase
- S_2: phosphotase
- $R|S_1$ unphosphorylated+kinase complex
- $R|S_2$ unphosphorylated+phosphotase complex



$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} R|S_1 \xrightarrow{k_3} R_p + S_1$$

$$R + S_2 \xleftarrow{k_3'} R_p|S_2 \underset{k_2'}{\overset{k_1'}{\rightleftharpoons}} R_p + S_2$$

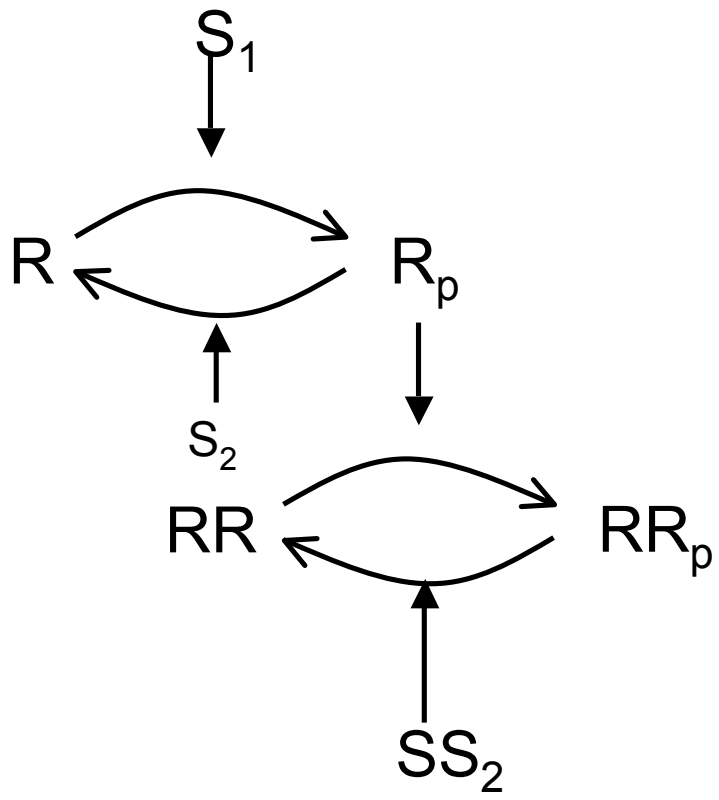
Phosphorylation cascade: 2-stage, Mass Action model 1

$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} R \mid S_1 \xrightarrow{k_3} R_p + S_1 -$$

$$R + S_2 \xleftarrow{k_3'} R \mid S_2 \underset{k_2'}{\overset{k_1'}{\rightleftharpoons}} R_p + S_2$$

$$RR + R_p \underset{kk_2}{\overset{kk_1}{\rightleftharpoons}} RR \mid R_p \xrightarrow{kk_3} RR_p + R_p$$

$$RR + SS_2 \xleftarrow{kk_3'} RR \mid SS_2 \underset{kk_2'}{\overset{kk_1'}{\rightleftharpoons}} RR_p + SS_2$$



Phosphorylation cascade: 3-stage, Mass-Action model 1

$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} R \mid S_1 \xrightarrow{k_3} R_p + S_1$$

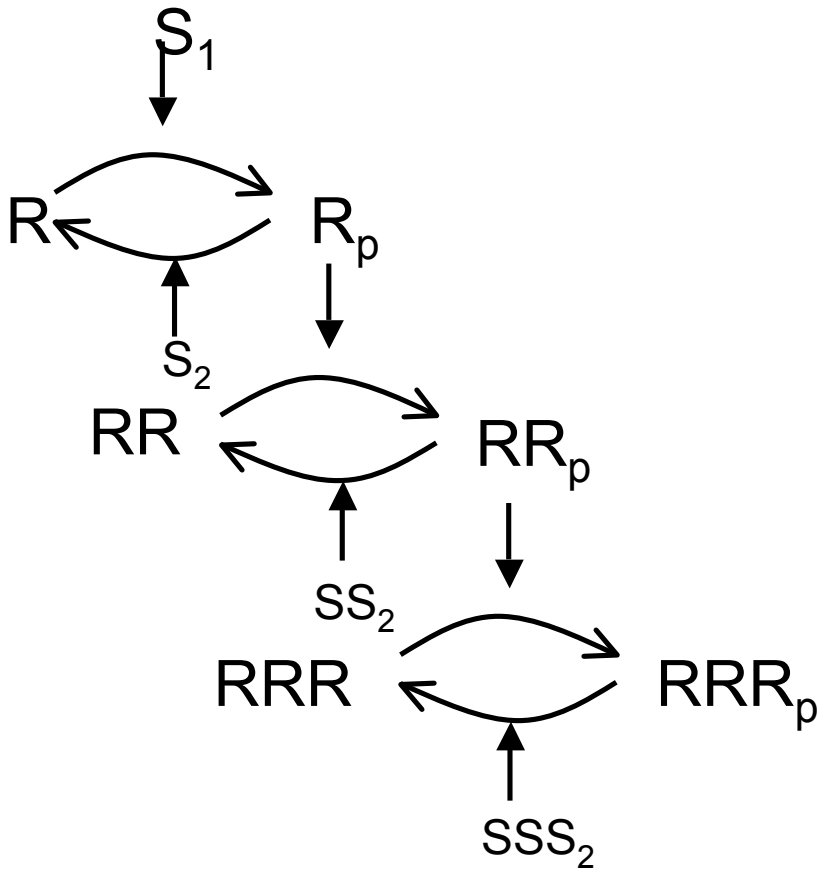
$$R + S_2 \xleftarrow{k_3'} R_p \mid S_2 \underset{k_2'}{\overset{k_1'}{\rightleftharpoons}} R_p + S_2$$

$$RR + R_p \underset{kk_2}{\overset{kk_1}{\rightleftharpoons}} RR \mid R_p \xrightarrow{kk_3} RR_p + R_p$$

$$RR + SS_2 \xleftarrow{kk_3'} RR_p \mid SS_2 \underset{kk_2'}{\overset{kk_1'}{\rightleftharpoons}} RR_p + SS_2$$

$$RRR + RR_p \underset{kkk_2}{\overset{kkk_1}{\rightleftharpoons}} RRR \mid RR_p \xrightarrow{kkk_3} RRR_p + RR_p$$

$$RRR + SSS_2 \xleftarrow{kkk_3'} RRR_p \mid SSS_2 \underset{kkk_2'}{\overset{kkk_1'}{\rightleftharpoons}} RRR_p + SSS_2$$



Phosphorylation cascade + negative feedback: 3-stage, Mass Action, model 1

$$RRR_p + S_1 \underset{ki'}{\overset{ki}{\rightleftharpoons}} RRR_p \mid S_1$$

$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} R \mid S_1 \xrightarrow{k_3} R_p + S_1$$

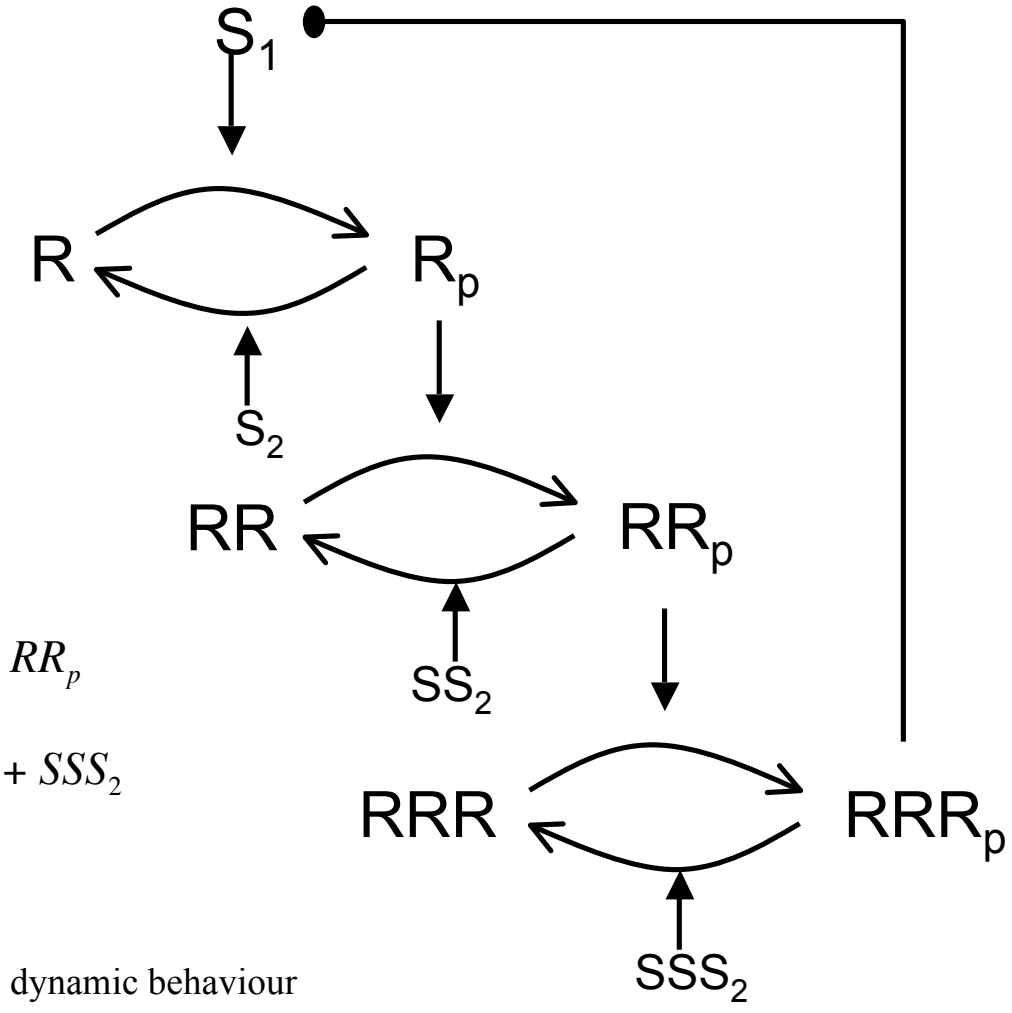
$$R + S_2 \xleftarrow{k_3'} R_p \mid S_2 \underset{k_2'}{\overset{k_1'}{\rightleftharpoons}} R_p + S_2$$

$$RR + R_p \underset{kk_2}{\overset{kk_1}{\rightleftharpoons}} RR \mid R_p \xrightarrow{kk_3} RR_p + R_p$$

$$RR + SS_2 \xleftarrow{kk_3'} RR_p \mid SS_2 \underset{kk_2'}{\overset{kk_1'}{\rightleftharpoons}} RR_p + SS_2$$

$$RRR + RR_{p1} \underset{kkk_2}{\overset{kkk_1}{\rightleftharpoons}} RRR \mid RR_p \xrightarrow{kkk_3} RRR_p + RR_p$$

$$RRR + SSS_2 \xleftarrow{kkk_3'} RRR_p \mid SSS_2 \underset{kkk_2'}{\overset{kkk_1'}{\rightleftharpoons}} RRR_p + SSS_2$$



Phosphorylation cascade + negative feedback: 3-stage, Inhibitor on 2nd stage, Mass Action

$$RRR_p + S_1 \underset{ki'}{\overset{ki}{\rightleftharpoons}} RRR_p \mid S_1$$

$$R + S_1 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} R \mid S_1 \xrightarrow{k_3} R_p + S_1$$

$$R + S_2 \xleftarrow{k_3'} R_p \mid S_2 \underset{k_2'}{\overset{k_1'}{\rightleftharpoons}} R_p + S_2$$

$$RR + R_p \underset{kk_2}{\overset{kk_1}{\rightleftharpoons}} RR \mid R_p \xrightarrow{kk_3} RR_p + R_p$$

$$RR + SS_2 \xleftarrow{kk_3'} RR_p \mid SS_2 \underset{kk_2'}{\overset{kk_1'}{\rightleftharpoons}} RR_p + SS_2$$

$$U + RR \underset{ku_2}{\overset{ku_1}{\rightleftharpoons}} U \mid RR$$

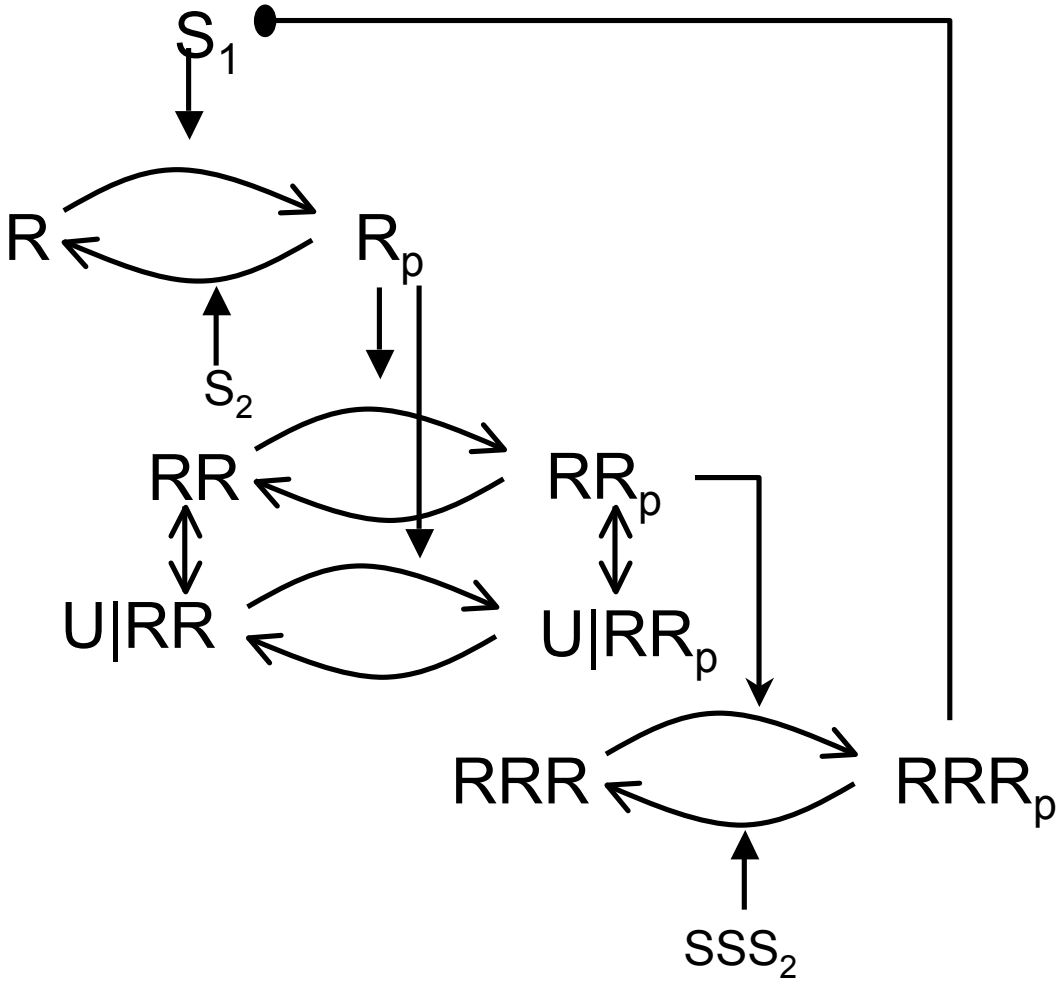
$$U + RR_p \underset{ku_2}{\overset{ku_1}{\rightleftharpoons}} U \mid RR_p$$

$$U \mid RR + R_p \underset{kk_2}{\overset{kk_1}{\rightleftharpoons}} U \mid RR \mid R_p \xrightarrow{kk_3} U \mid RR_p + R_p$$

$$U \mid RR + SS_2 \xleftarrow{kk_3'} U \mid RR_p \mid SS_2 \underset{kk_2'}{\overset{kk_1'}{\rightleftharpoons}} U \mid RR_p + SS_2$$

$$RRR + RR_p \underset{kkk_2}{\overset{kkk_1}{\rightleftharpoons}} RRR \mid RR_p \xrightarrow{kkk_3} RRR_p + RR_p$$

$$RRR + SSS_2 \xleftarrow{kkk_3'} RRR_p \mid SSS_2 \underset{kkk_2'}{\overset{kkk_1'}{\rightleftharpoons}} RRR_p + SSS_2$$



Further Analyses

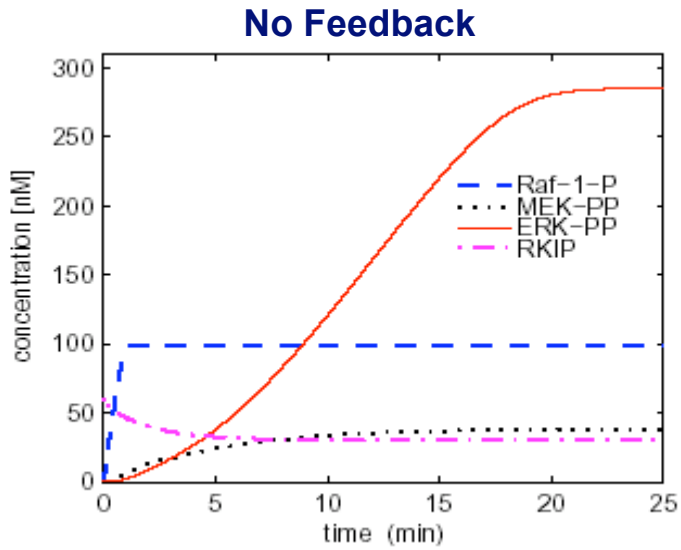
All initial concentrations can be varied at will, e.g. to test a concentration series of one component (sensitivity analysis)

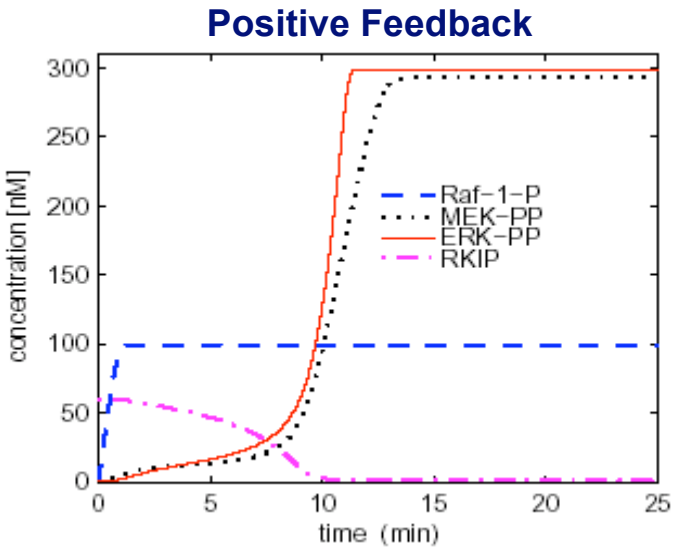
Effect of slightly different k-values can be tested (stability of the model with respect to measurement/estimation errors)

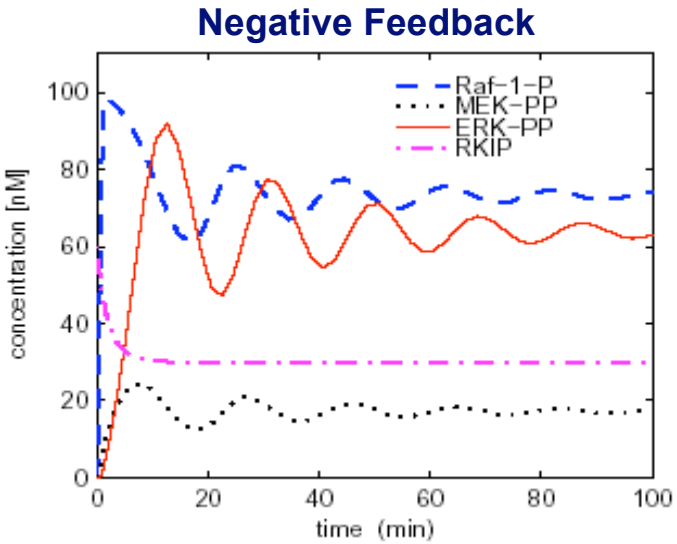
Effect of inhibitors of each reaction (changed k-values) can be predicted

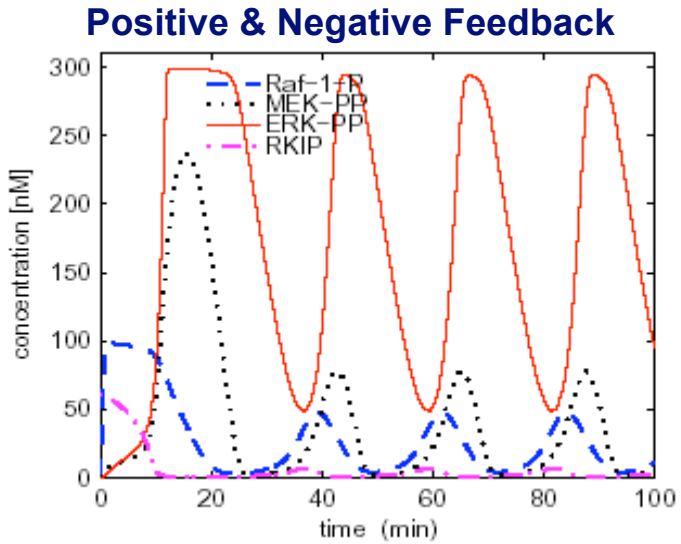
Concentrations at each time-point are predicted exactly and can be tested experimentally

Combination of positive & negative feedback: Simulation

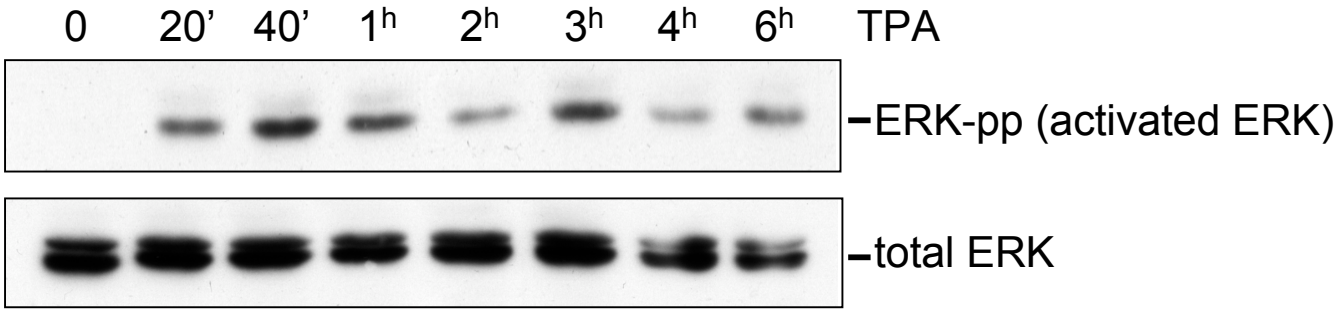




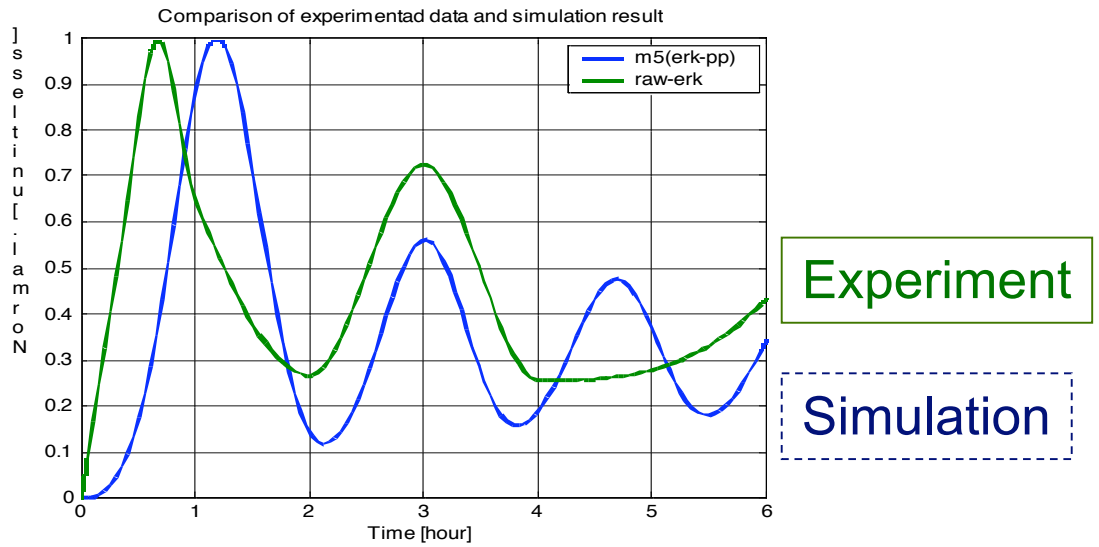




Combination of positive & negative feedback: Simulation vs. Experimental Data



Western blots COS1 cell lysates



Experiment

Simulation

SBML: http://www.sbml.org

- The Systems Biology Markup Language (SBML) is a computer-readable format for representing models of biochemical reaction networks. SBML is applicable to metabolic networks, cell-signaling pathways, regulatory networks, and many others.
- SBML has been evolving since mid-2000 through the efforts of an international group of software developers and users. Today, SBML is supported by over 75 software systems including Gepasi. Also an SBML->MatLab converter
- Advances in biotechnology are leading to larger, more complex quantitative models. The systems biology community needs information standards if models are to be shared, evaluated and developed cooperatively. SBML's widespread adoption offers many benefits, including:
  - enabling the use of multiple tools without rewriting models for each tool
  - enabling models to be shared and published in a form other researchers can use even in a different software environment
  - ensuring the survival of models (and the intellectual effort put into them) beyond the lifetime of the software used to create them.

SBML - XML Based Language

```
<sbml>
<model>
        <listOfCompartments> <compartment/> </listOfCompartments>
        <listOfSpecies> <specie/>          < /listOfSpecies>
        <listOfReactions>
                <reaction>
                        <listOfReactants>
                                <specieReference/>
                        </listOfReactants>
                        <listOfProducts>
                                <specieReference/>
                        </listOfProducts>
                        <kineticLaw>
                                <listOfParameters>
                                <parameter/>
                                </listOfParameters>
                        </kineticLaw>
                </reaction>
        </listOfReactions>
</model>
</sbml>
```

SBML Example

Specie representation: m1 in RKIP model:
<specie name="**m1**" compartment="**compartment**" initialAmount="**2.5**" boundaryCondition="**false**" />

Reaction representation: k1 in RKIP model: m1 + m2 -> m3 (rate = k1 = 0.53)
<reaction name="**k1**" reversible="**false**">

<listOfReactants>
<specieReference specie="**m1**" stoichiometry="**1**" />
<specieReference specie="**m2**" stoichiometry="**1**" />
</listOfReactants>

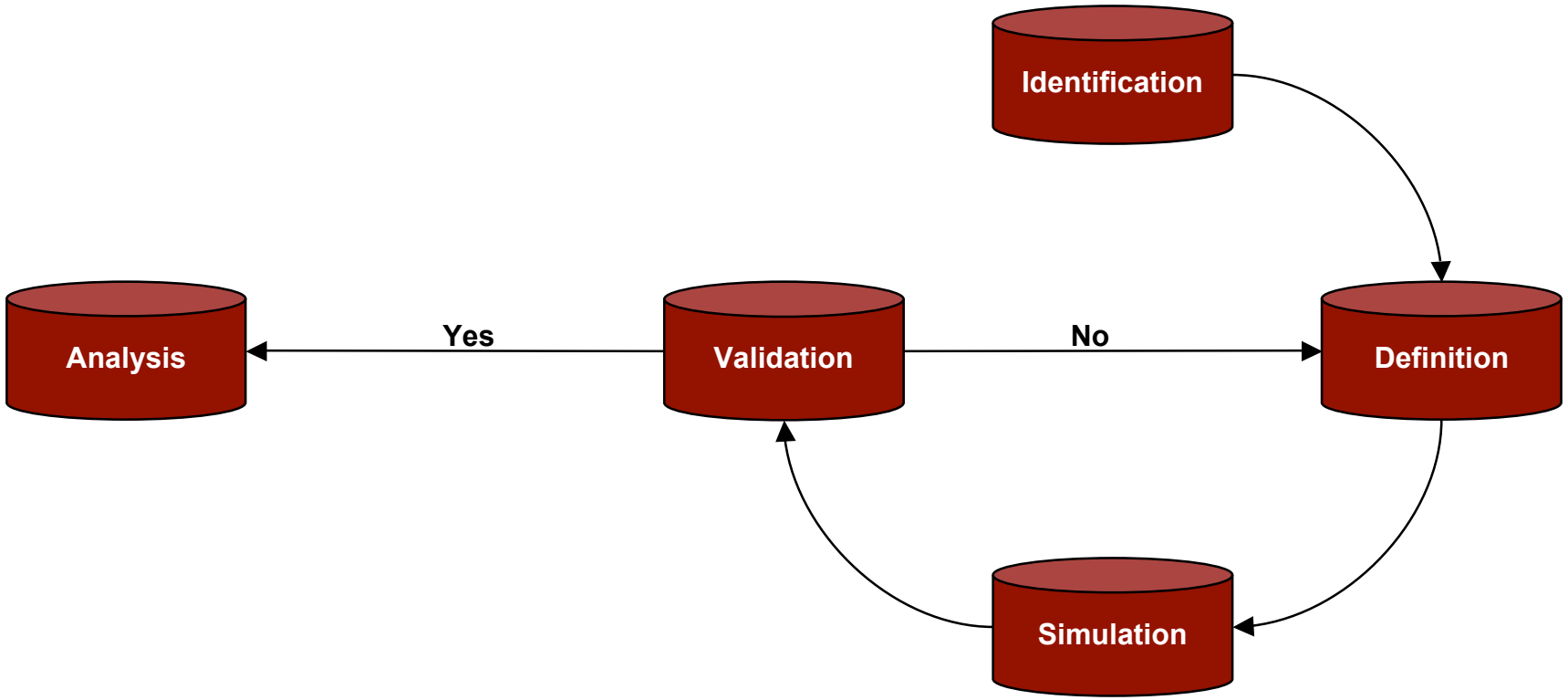
<listOfProducts>
<specieReference specie="**m3**" stoichiometry="**1**" />
</listOfProducts>

<kineticLaw formula="**k_1*m1*m2**">
<listOfParameters>
<parameter name="**k_1**" value="**0.53**" />
</listOfParameters>
</kineticLaw>

</reaction>

Slide from
Richard Orton

How to model



Slide from
Richard Orton

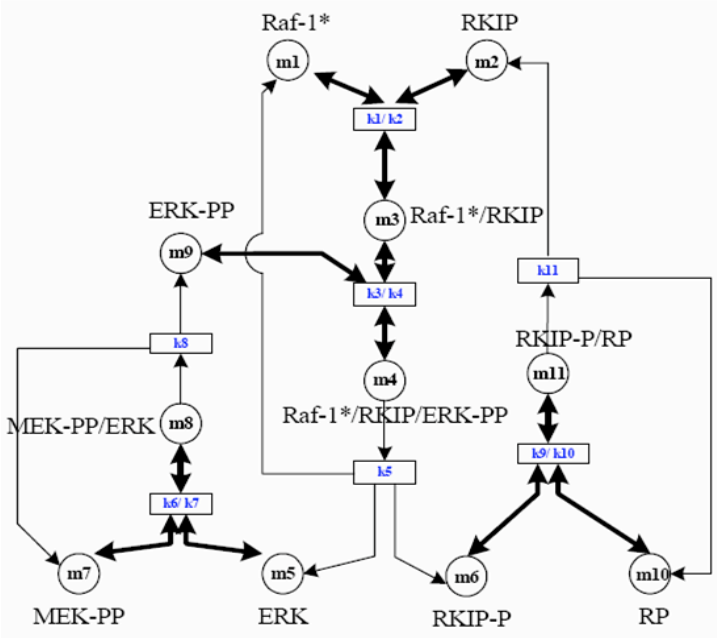
How to model…1: Identification

- Identify the biological pathway to model (what)
  - RKIP
  - EGF and NGF activated MAPK
- Or, more importantly, identify the biological question to answer (why)
  - What influence does the Raf Kinase Inhibitor Protein (RKIP) have on the Extracellular signal Regulated Kinase (ERK) signalling pathway?
  - How do EGF and NGF cause differing responses in ERK activation, transient and sustained, respectively?

Slide from Richard Orton

How to model…2: Definition

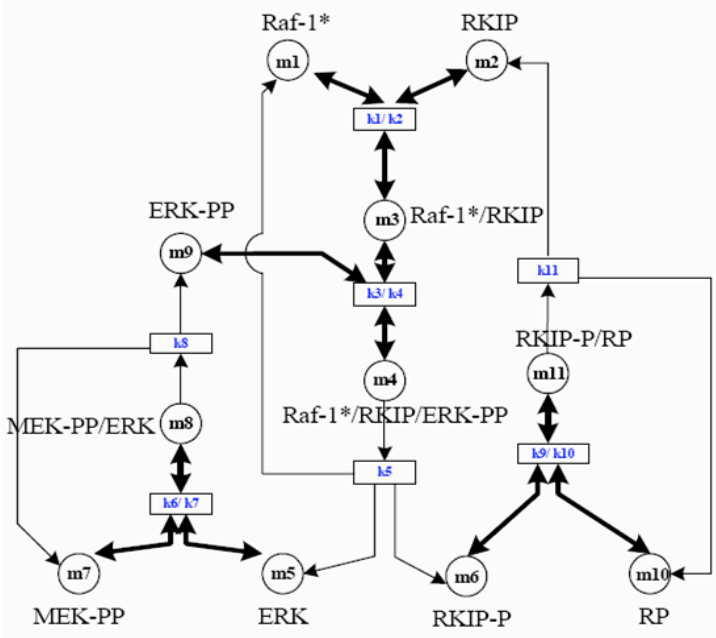
- This is the key step and is not trivial
- Draw a detailed picture of the pathway to model
  - Define all the proteins/molecules involved
  - Define the reactions they are involved in
  - Where do you draw the model boundary line?
- Check the literature
  - What is known about the pathway and proteins?
  - What evidence is there that protein A binds directly to protein B?
  - Protein C also binds directly to protein B: does it compete with protein A or do they bind to protein B at different sites?
  - Trust & Conflicts: it is important to recognize which evidence to trust and which to discard (talk to the people in the wet lab)
- Simplifying assumptions
  - Many biological processes are very complex and not fully understood
  - Therefore, developing a model often involves making simplifying assumptions
  - For example, the activation of Raf by Ras is very complicated and not fully understood but it is often modelled as:
    - Raf + Ras-GTP = Raf/Ras-GTP -> Raf-x + Ras-GTP
  - Although this is a simplification, it is able to explain the observed data



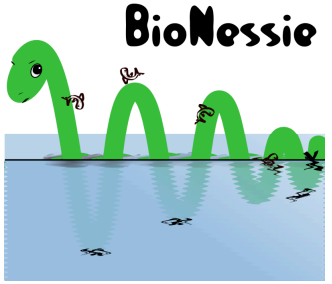
Slide from Richard Orton

How to model…2: Definition

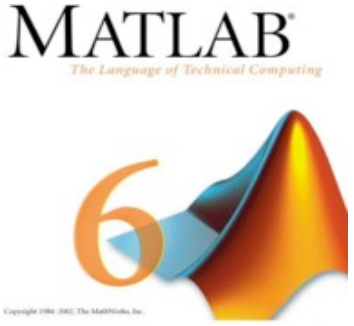
- Define the kinetic types
  - Each reaction has a specific kinetic type
  - All the reactions in the RKIP model are mass action (plain, uncatalysed kinetic type):
    - V = k1[m1][m2] - k2[m3]
  - Another common kinetic type is Michaelis Menten (enzyme catalysis):
    - V = Vmax[S] / (Km+[S])
- Define the rate constants (k's, km's, Vmax's etc)
- Define the initial concentrations
- Check the literature
  - What values have been previously reported?
  - What values are used in similar models?
  - Do you trust them? Are there any conflicts?
  - Measure them yourself in the wet lab
  - Parameter estimation techniques: estimate some parameters based on others and observed data

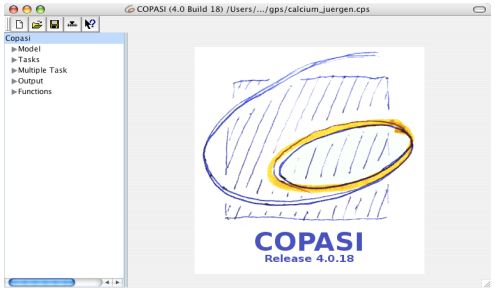


How to model…3: Simulation



- Once the model has been constructed and parameter data has been assigned you can simulate (run) the model
- This is a relatively straightforward step as there are many software tools available to simulate differential equation based models
- For example:
  - BioNessie
  - MatLab
  - Copsai / Gepasi
  - CellDesigner
  - Jarnac
  - WinScamp
  - Many many more
- Runtime options include setting the time to run the model for ~~and the number~~ of data points to take



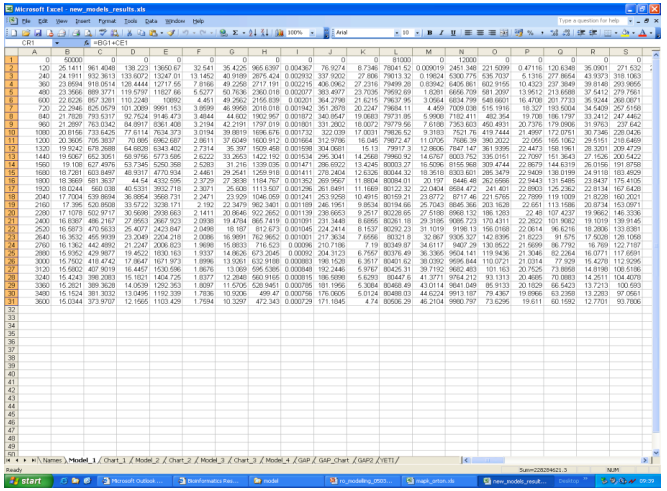


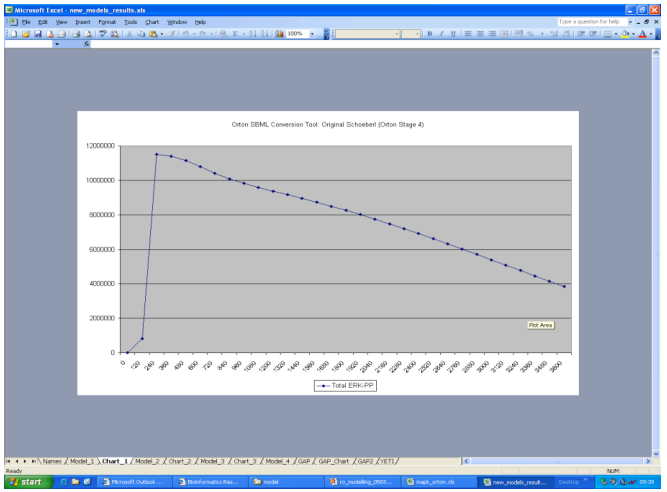
Slide from Richard Orton

Slide from Richard Orton

How to model…4: Validation

- Simulating the model typically returns a table of data which shows how each specie's concentration varies over time
- This table can then be used to generate graphs of specie concentrations
- Do the model results match the experimental data?
  - Yes: validation
  - No: back to definition and check for errors
    - Simple typos
    - Wrong kinetics
    - Over simplifications of processes
    - Missing components from the model
    - Incorrect parameter data
- The model can then be validated further by checking the system behaves correctly when things are varied:
  - It might be known how the system behaves when you over-express or knockout a component
  - The model should be able to recreate this behaviour
- If the model's results do not match known biology, we cannot rely on predictions about unknown biology





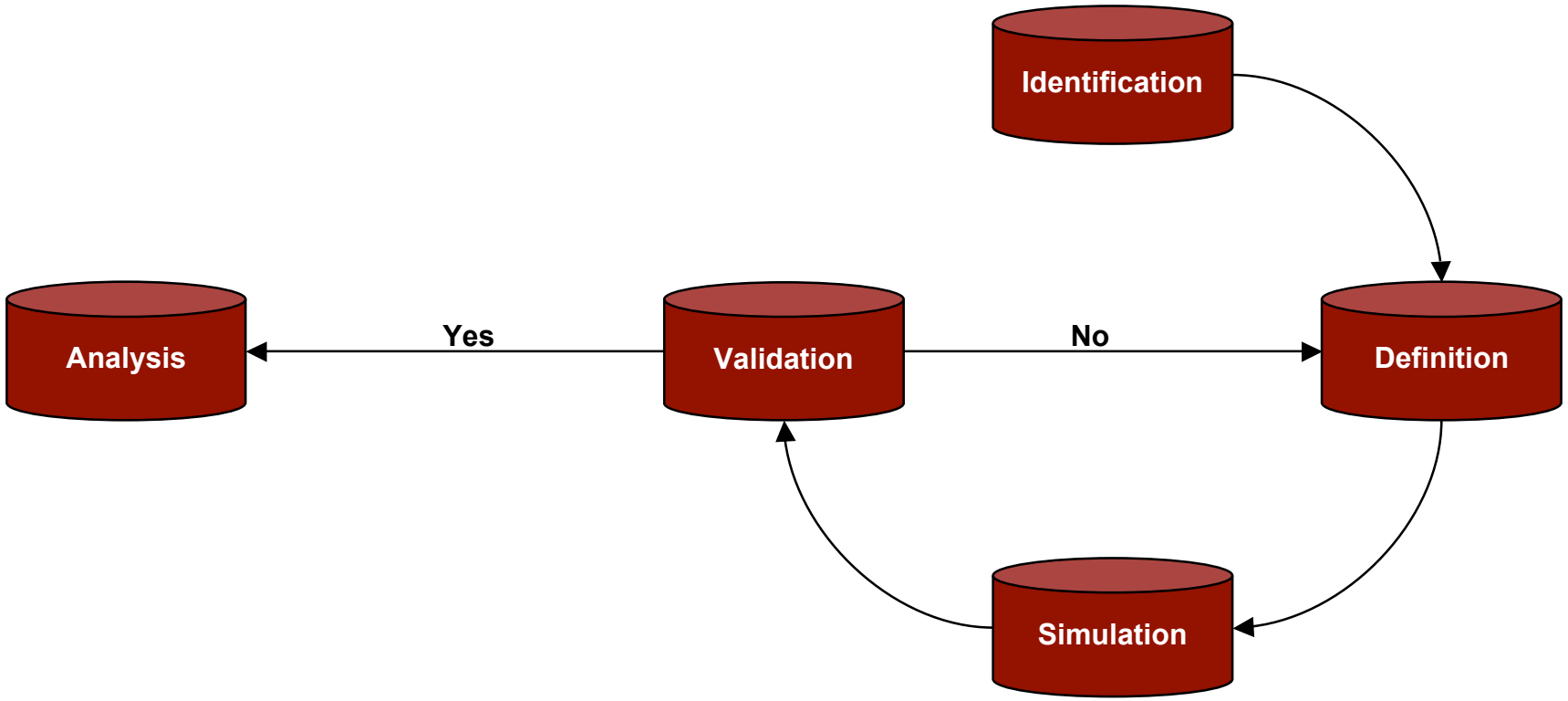
Slide from Richard Orton

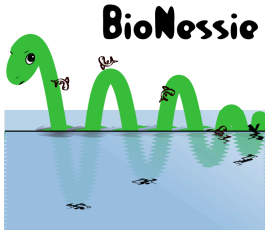
How to model…5: Analysis

- After the model has been validated we can then analyse and interpret the results
  - What do the results imply or suggest?
  - What do they tell us that is new and that we did not know/understand before?
  - What predictions can we make?
- Sensitivity analysis can be used to identify the key steps and components in the pathway as well as monitoring how robust the system is:
  - Vary an initial concentration or rate by a small amount and see what affect it has on the system as a whole: small changes in a key value are likely to have a large affect
  - How robust is the system to changes?
- Knockout experiments are easy to do in a model: for example, simply set the initial concentration of the desired component to 0
  - Knockout experiments can be used to identify which components are essential and which are redundant
  - Can also knockout reactions (set rate to 0) to identify essential and redundant reactions in the system

Slide from
Richard Orton

How to model…Overview





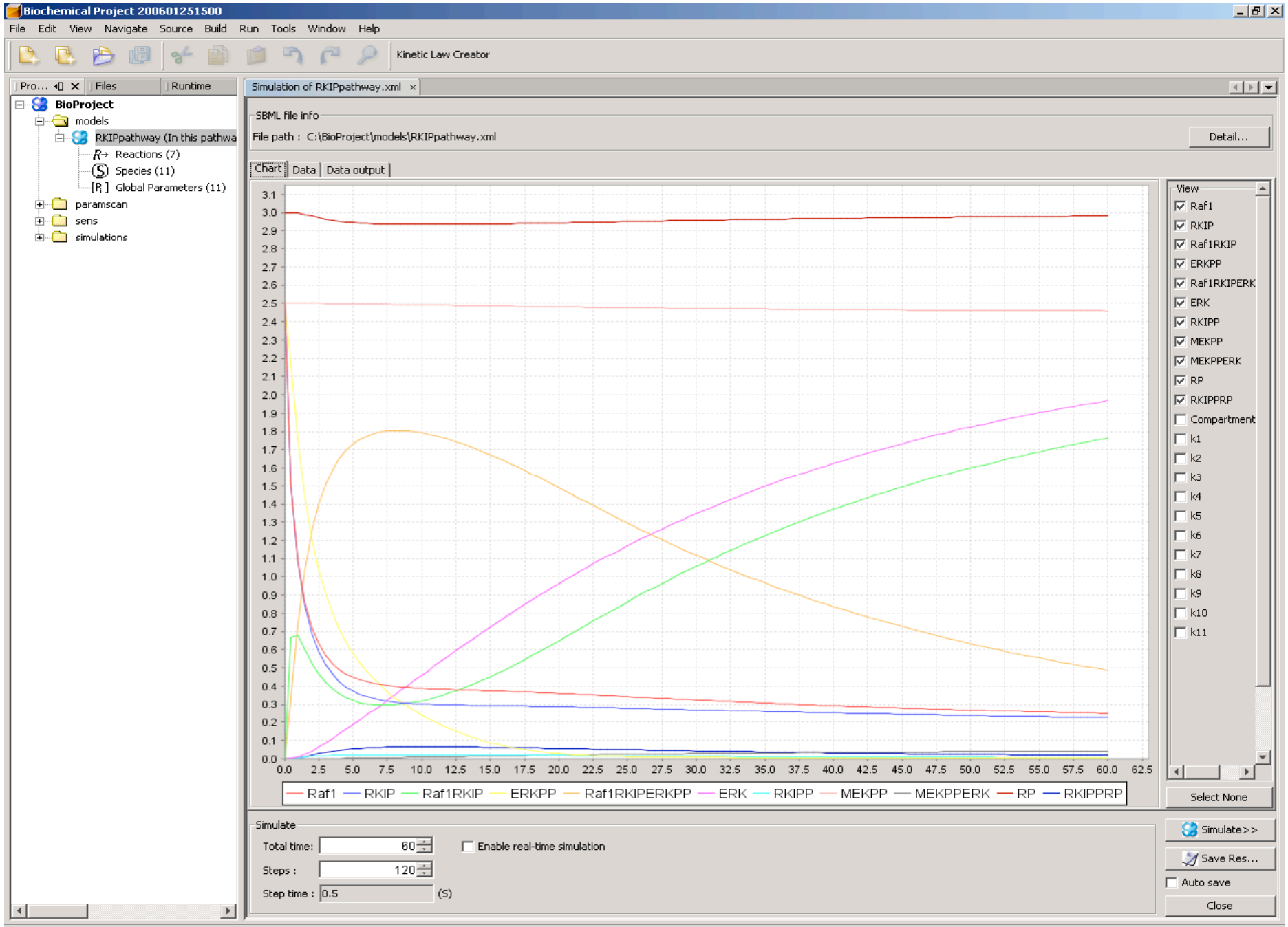
BioNessie ODE workbench

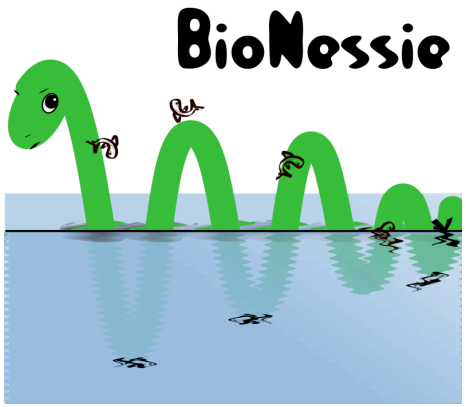
- Platform independent
  - Windows, Linux (i386 or AMD64) and Mac Os with Intel i386.
  - Released on 5th October 2006 for internal use.
  - JAVA Web Start

- Simulation
  - Multithreaded: simulation of different models at the same time.
  - User-friendly data viewer and printable data output

- SBML model construction
  - Graphical tool supports creation & editing of SBML biochemical models
  - Kinetic Law creation and management

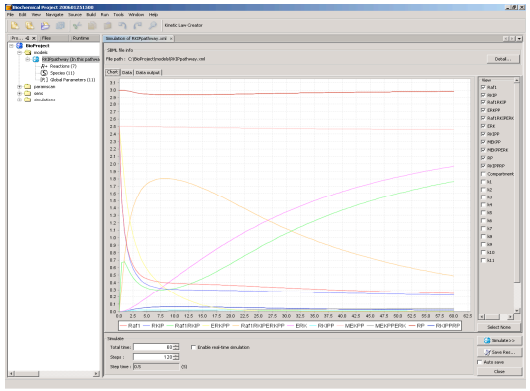
- Parameter Scanning
- Sensitivity Analysis
- Grid
- Model Version Control System
- Model Development Management
- Optimisation

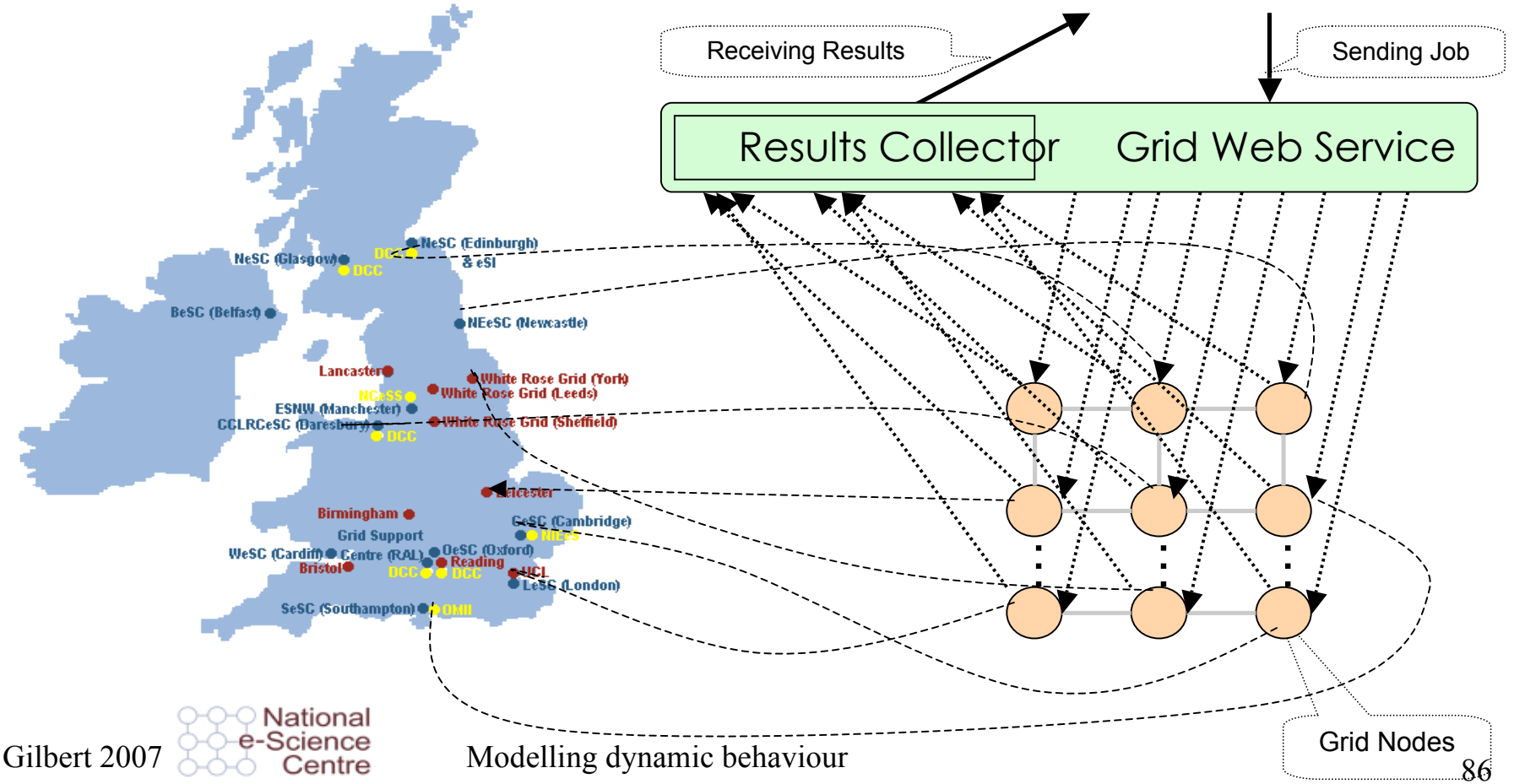

Biochemical Project 200601251500
File Edit View Navigate Source Build Run Tools Window Help
Kinetic Law Creator
Pro... Files Runtime
BioProject
models
RKIPpathway (In this pathwa
Reactions (7)
Species (11)
Global Parameters (11)
paramscan
sens
simulations
Simulation of RKIPpathway.xml
SBML file info
File path : C:\BioProject\models\RKIPpathway.xml
Detail...
Chart Data Data output
View
Raf1
RKIP
Raf1RKIP
ERKPP
Raf1RKIPERK
ERK
RKIPP
MEKPP
MEKPPERK
RP
RKIPPRP
Compartment
k1
k2
k3
k4
k5
k6
k7
k8
k9
k10
k11
Select None
Raf1 RKIP Raf1RKIP ERKPP Raf1RKIPERKPP ERK RKIPP MEKPP MEKPPERK RP RKIPPRP
Simulate
Total time: 60
Enable real-time simulation
Steps : 120
Step time : 0.5 (S)
Simulate>>
Save Res...
Auto save
Close



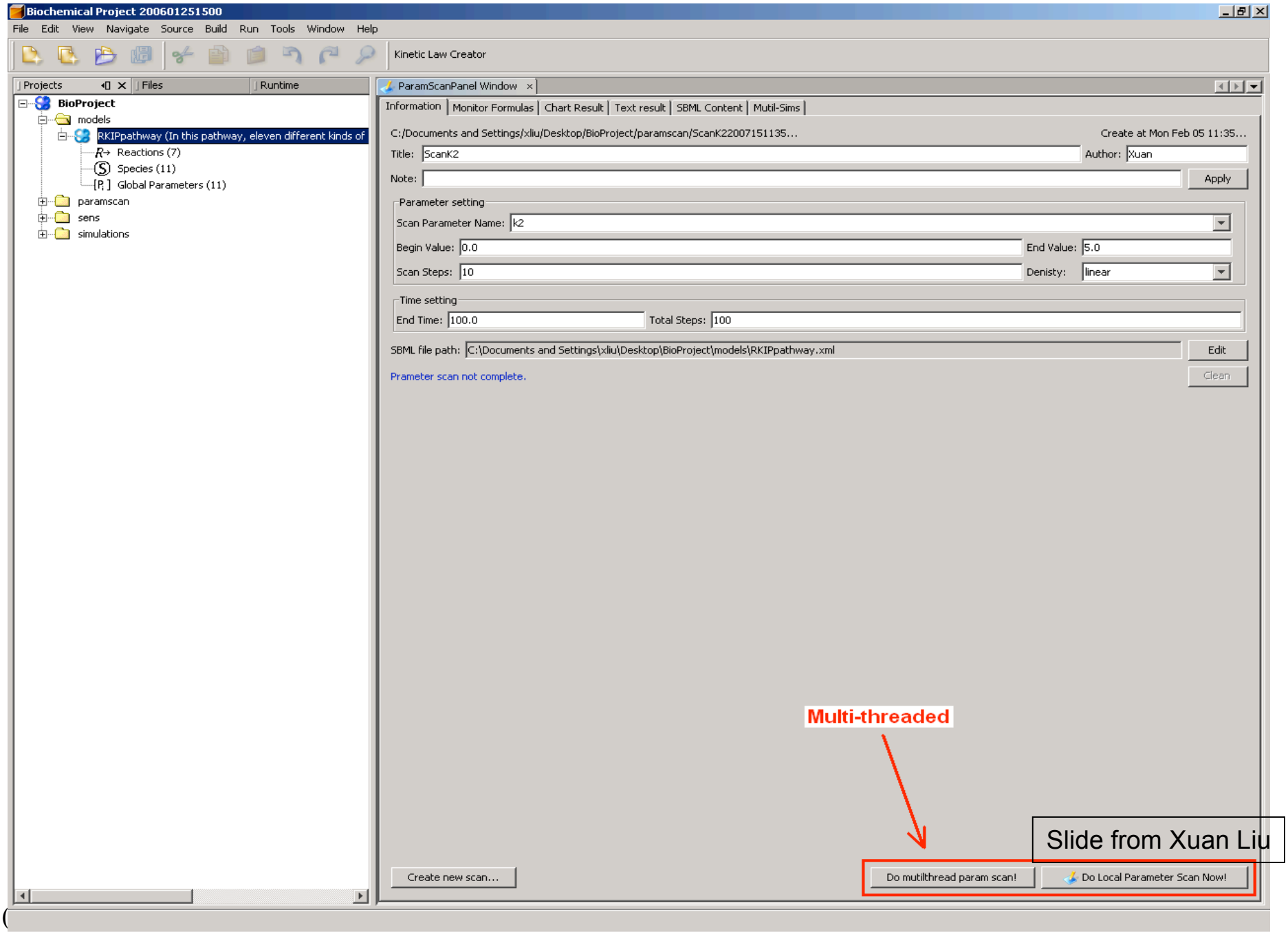
Simulator, analyser ...& *go-faster* on the Grid!

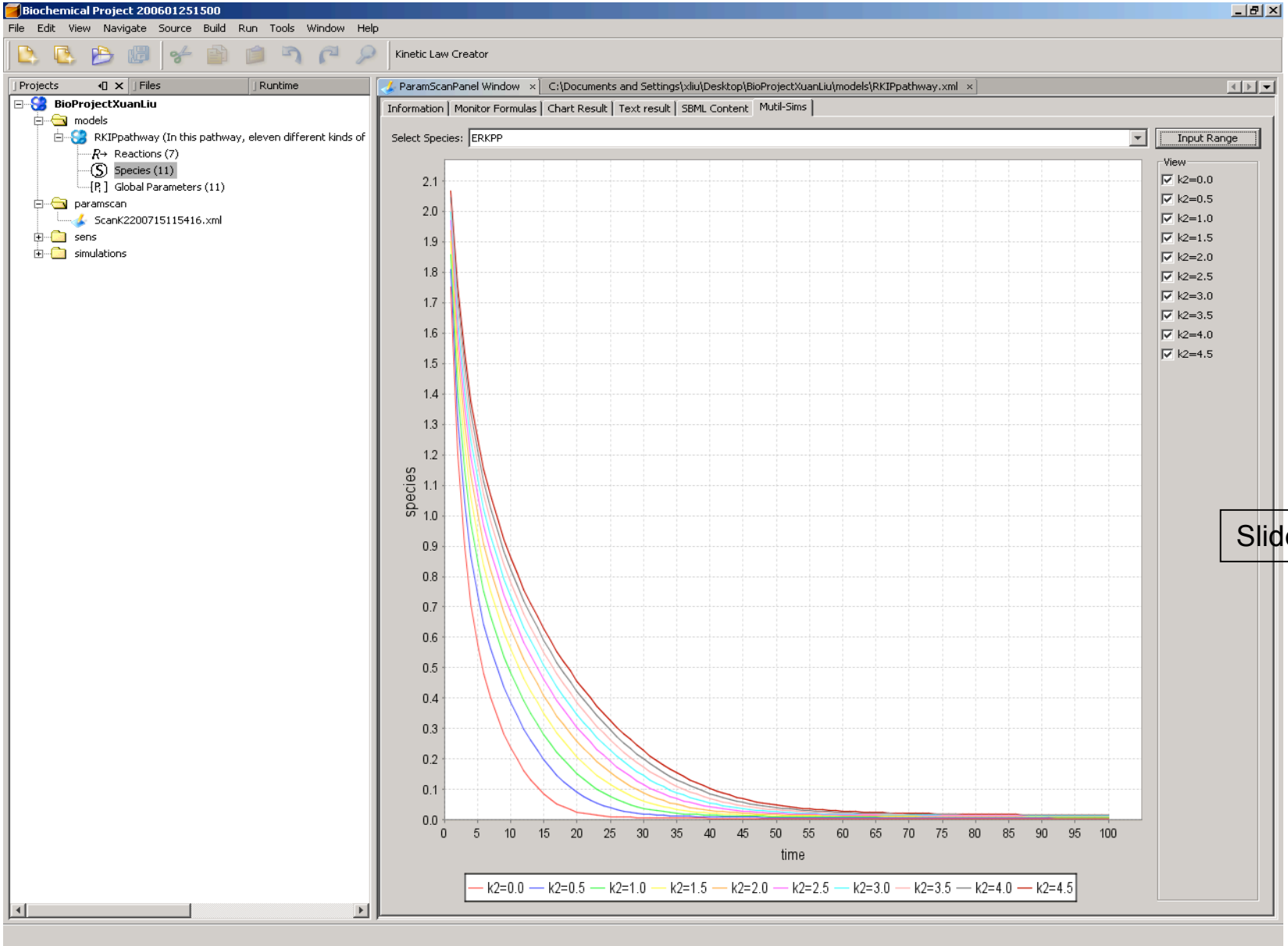
Xuan Liu, Vladislav Vyshemirsky,
Gary Gray, Jipu Jiang, Femi Ajayi
(David Gilbert, Richard Sinnott)





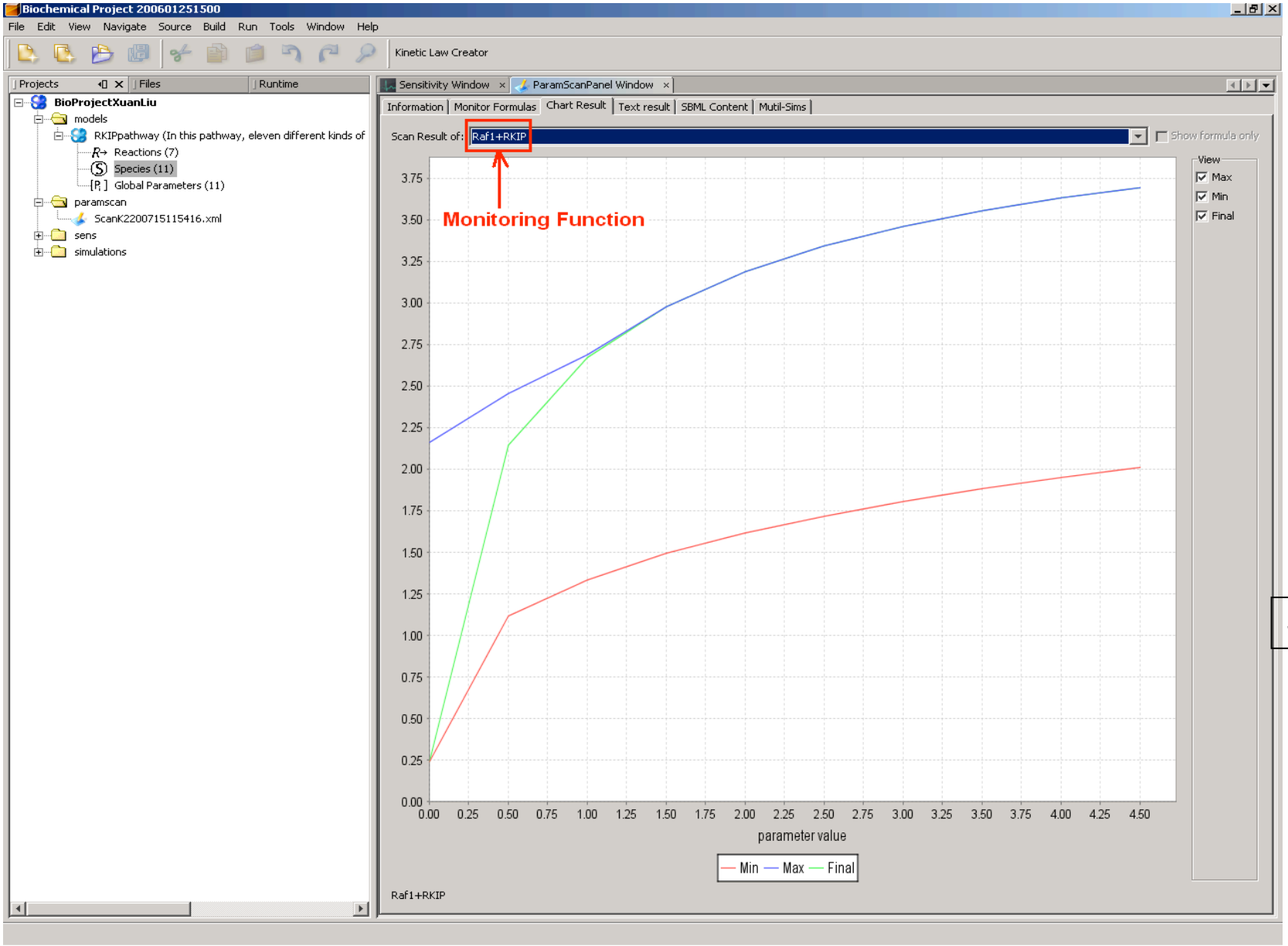
Multi-threaded Parameter Scan





Slide from Xuan Liu

This plot shows the whole trace of selected species - ERKPP for a parameter scan in RKIPpathway.xml of parameter K2 from 0 through 4.5 in steps of 0.5 with linear density for the timecourse of 100 timesteps of 100 time units.



Slide from Xuan Liu

This plot shows the min. max and final values of monitoring function Raf1+RKIP for a parameter scan in RKIPpathway.xml of parameter K2 from 0 through 5 in steps of 0.5 with linear density for the timecourse of 100 timesteps of 100 time units.

Sensitivity analysis

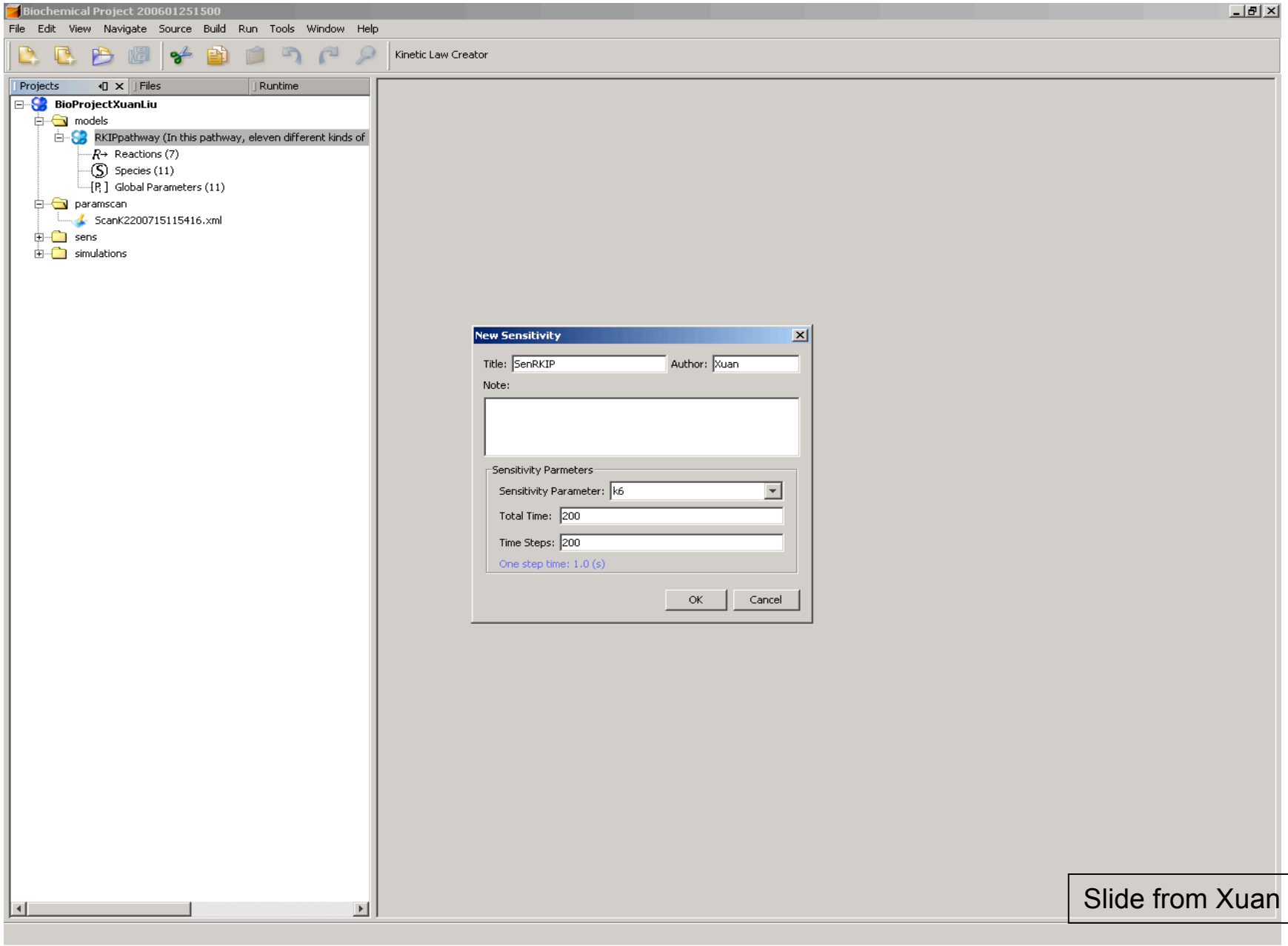
- Sensitivity analysis investigates the changes in the system outputs or behavior with respect to the parameter variations. It is a general technique for establishing the contribution of individual parameter values to the overall performance of a complex system.

- Sensitivity analysis is an important tool in the studies of the dependence of a system on external parameters, and sensitivity considerations often play an important role in the design of control systems.

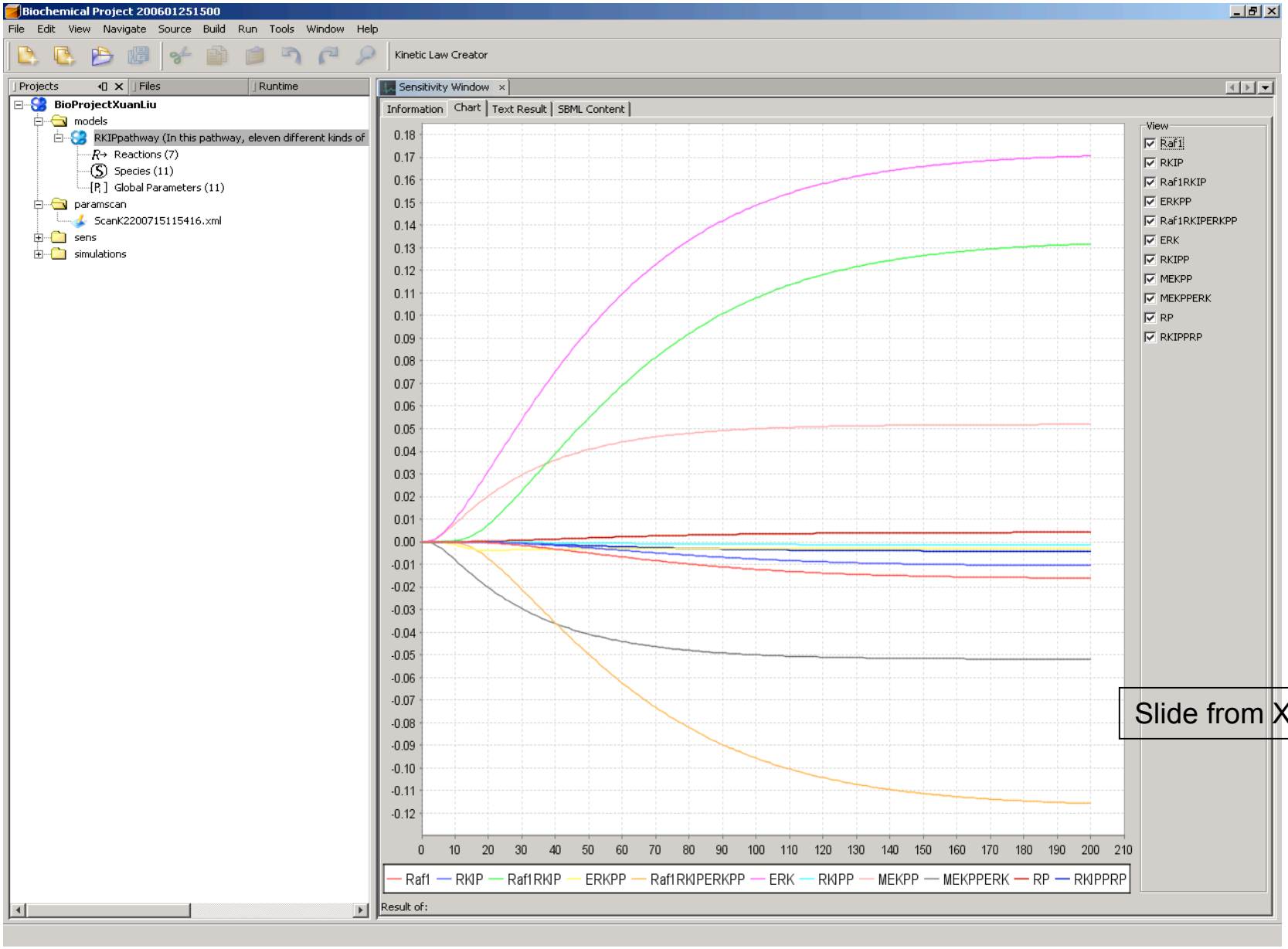
- Parameter sensitivity analysis can also be utilised to validate a model's response and iteratively, to design experiments that support the estimation of parameters

Slide from Xuan Liu

Sensitivity Analysis Creation in BioNessie



Slide from Xuan Liu

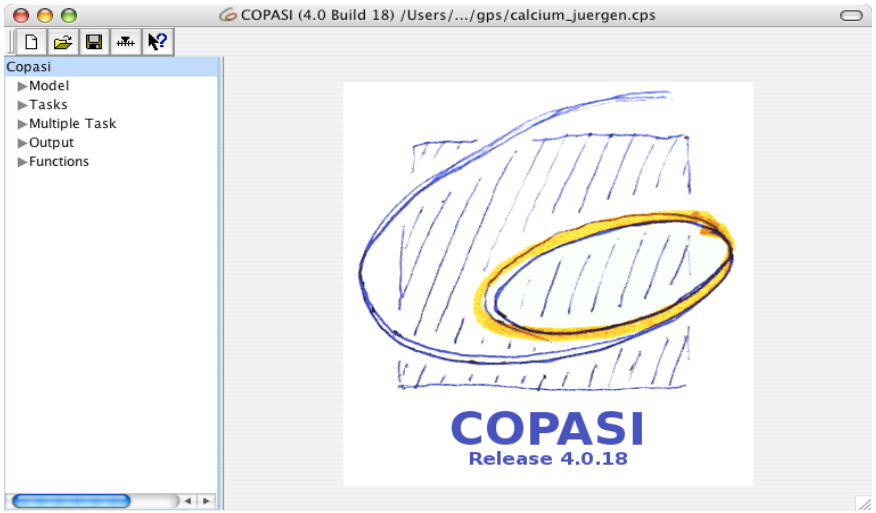


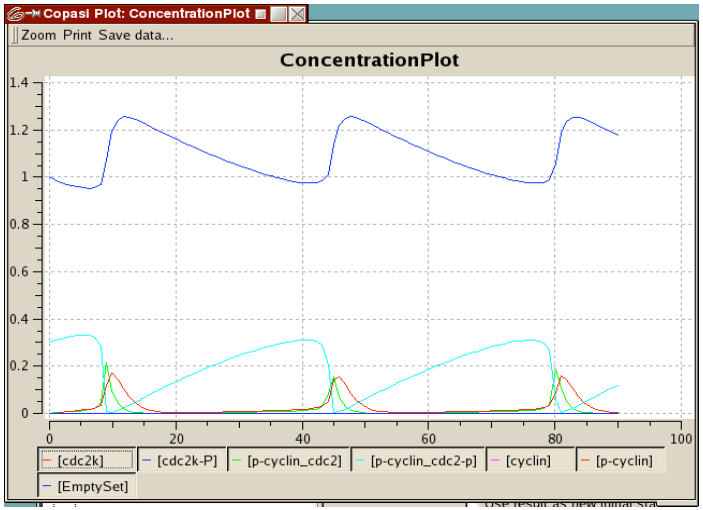
Slide from Xuan Liu

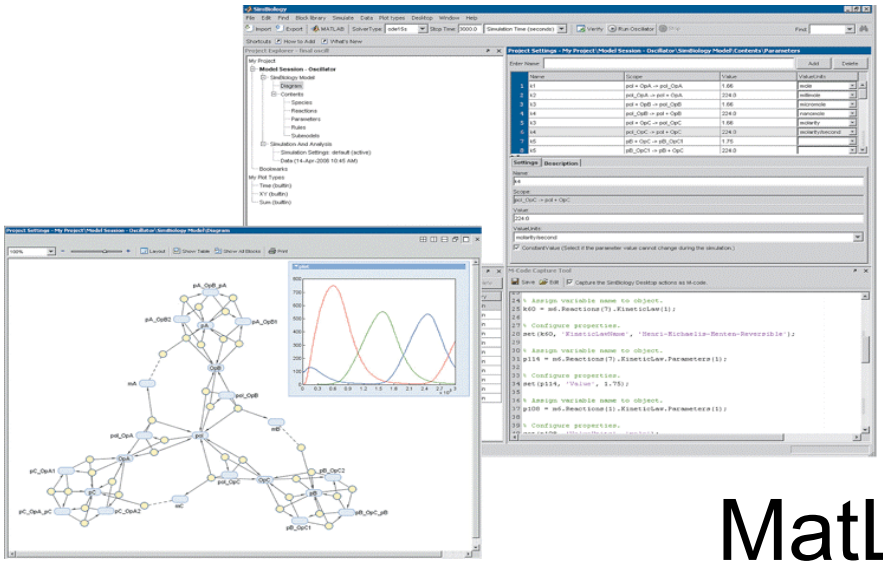
This creates a plot of the sensitivity of species Raf1, RKIP, Raf1RKIP, ERKPP, Raf1RKIPERKPP, ERK, RKIPP, MEKPP, MEKPPERK, RP and RKIPPRP to the values of the parameter K6 for the timecourse of 200 timesteps of 200 time units.

Other simulators include…

Copasi



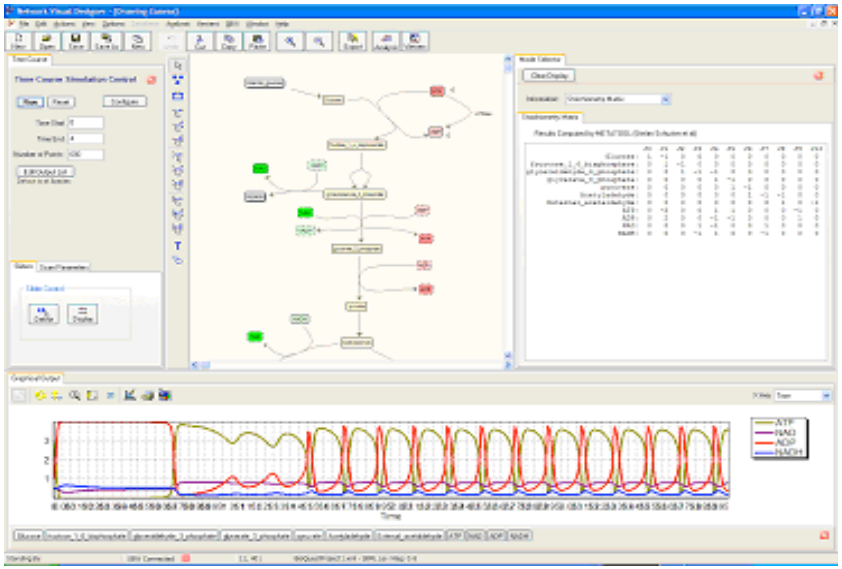




MatLab & SimBio

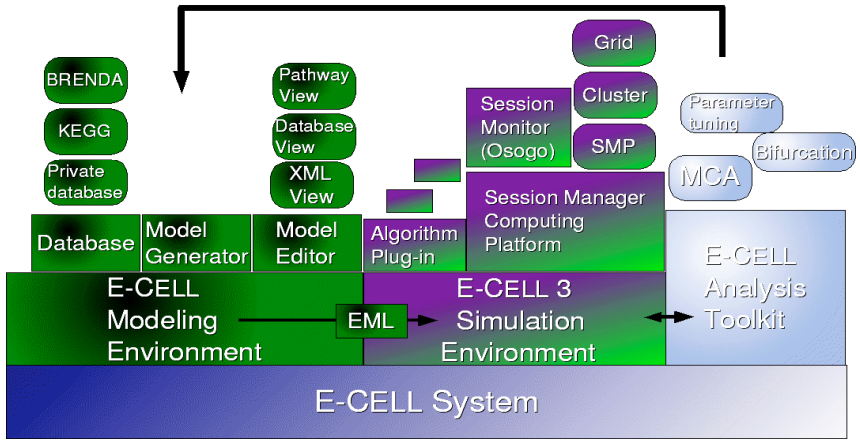
more simulators …

Systems Biology Workbench



SBW - standard parts

E-CELL Development Overview



Conclusions and Outlook

- Differential equations allow exact predictions of systems behaviour in a unified formalism
- Modelling = *in silico* experimentation
- Difficulties:
  - translation from biology
    - modular model building interfaces, e.g. Gepasi/COPASI, Genomic Object Net, E-cell, Ingeneue
  - managing complexity explosion
    - pathway visualization and construction software
    - standardized description language, e.g. Systems Biology Markup Language (SBML)
  - lack of biological data
    - perturbation-based parameter estimation, e.g. metabolic control analysis (MCA)
    - constraints-based modelling, e.g. flux balance analysis (FBA)
    - semi-quantitative differential equations for inexact knowledge

Lecture outline

- Biochemical reactions
- Modelling with Ordinary Differential Equations
- Kinetics: Mass Action
- Examples
  - Signalling & metabolic pathways
  - Oscillators & Amplifiers
- Analysis
- ODE simulators