



Modeling with Multicellularity, Arrays, and Connections

Eric Mjolsness

Bruce Shapiro

JPL/Caltech

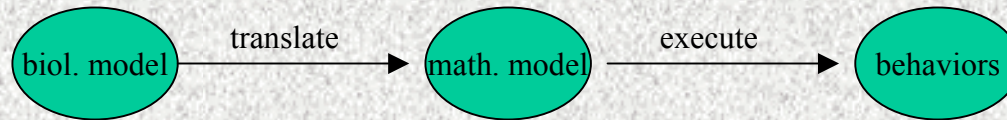
Why Automatic Model Generation?

- A tool for systematic model exploration
- Vary everything:
 - Reactions, species, states
 - Compartments, their connections
 - Level of simulation detail
 - Dynamics assumed for reactions, compartments, etc.
- Other benefits
 - * Large, consistent models can be generated from concise descriptions, and eventually from data
 - Ease translation from biological to mathematical models with computer support



How to Automate Model Generation?

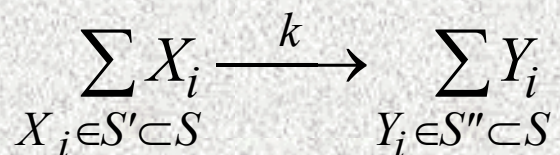
- *Reify* biological models, mathematical models, and translations between them using *computer algebra*



- Use math models to limit complexity
 - enables generalization
- Use structured indexing to organize biological objects, collections, relationships

Canonical Forms for Translation: Chemical reactions

- Input Canonical Form for Chemical Reaction



- Output Canonical Form: Terms in an ODE

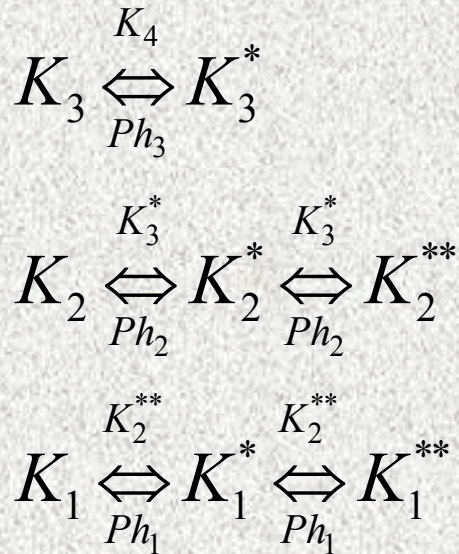
$$\tau_i \frac{dX_i}{dt} = \sum_{\alpha} c_{i\alpha} \prod_j X_j^{n_{i\alpha j}}$$



Representational Issues

- Label, tree, graph structures of species
 - E.g. MAPK scaffold $S[i,j,k]$, S-S, polymers, dendromers
 - Tree of numerical/symbolic labels – syntax tree
 - Compartment # labels: variable-length index list in tree
- Static compartment graph
- Dynamic nodes (cell division), ~static links
- Dynamic links

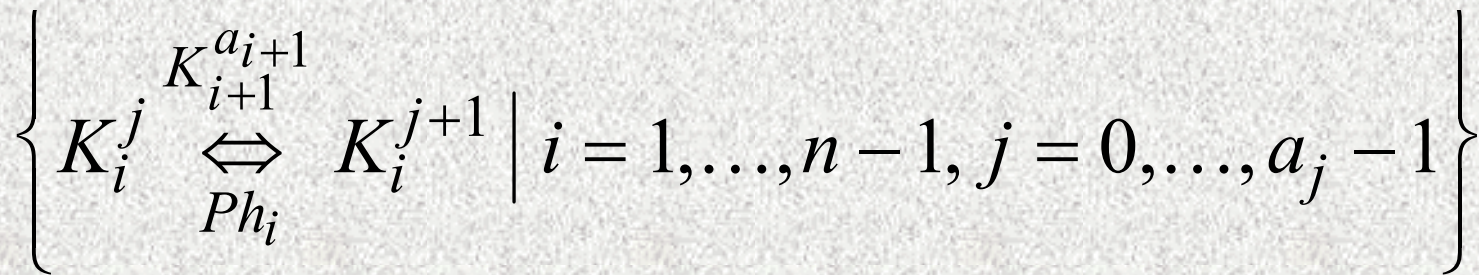
MAPK Reactions in Solution and Indexing



- Biological Model for pheromone response in yeast: cascade of two-way catalyses, chained to produce active state [Ferrell]
- Indexing permits succinct summary

$$\left\{ K_i^j \xrightleftharpoons[Ph_i]{K_{i+1}^{a_{i+1}}} K_i^{j+1} \mid i = 1, \dots, n - 1, j = 0, \dots, a_j - 1 \right\}$$

Model Generation for Solution Phase MAPK Pathway



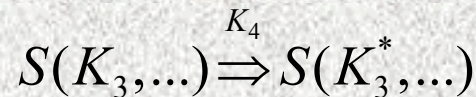
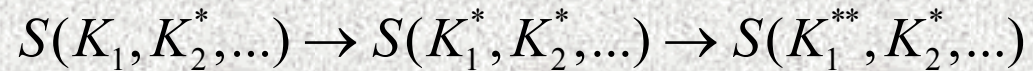
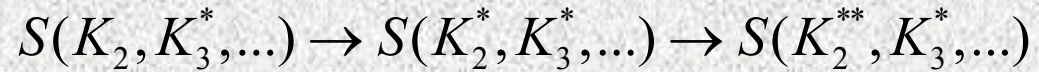
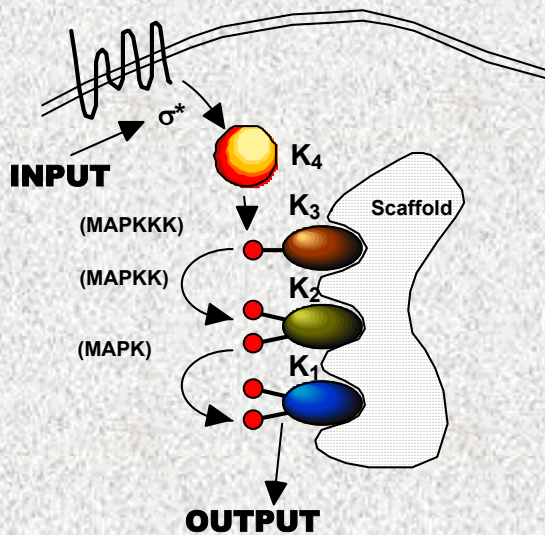
```
solutionReactions = genReacts [K, 3, {2, 2, 1, 1}, kpase]
```

```
{ K[1, 0]  $\xrightleftharpoons[kpase[1]]{K[2,2]}$  K[1, 1], K[1, 1]  $\xrightleftharpoons[kpase[1]]{K[2,2]}$  K[1, 2],
  K[2, 0]  $\xrightleftharpoons[kpase[2]]{K[3,1]}$  K[2, 1], K[2, 1]  $\xrightleftharpoons[kpase[2]]{K[3,1]}$  K[2, 2], K[3, 0]  $\xrightleftharpoons[kpase[3]]{K[4,1]}$  K[3, 1] }
```

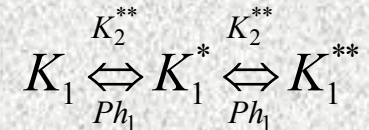
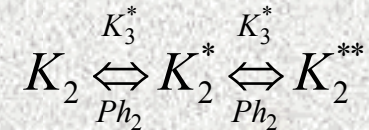
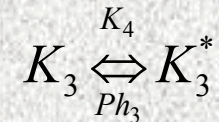
- 5 High Level Reactions generated
- Equivalent to 30 elementary reactions (also generated, not shown here)

MAPK Cascade in Scaffold

- Main catalyses become unidirectional and direct:



vs.



- Solution phase reactions still required to restore system after signaling

Reactions in MAP Kinase Cascade

- Phosphorylation in Solution
- Binding to Scaffold
- Phosphorylation in Scaffold

```
phosphorylationReactions = genScafPhosReacts[S, {2, 2, 1, 1
```

```
{S[0, 2, -1] → S[1, 2, -1], S[0, 2, 0] → S[1, 2, 0],
  S[0, 2, 1] → S[1, 2, 1], S[1, 2, -1] → S[2, 2, -1],
  S[1, 2, 0] → S[2, 2, 0], S[1, 2, 1] → S[2, 2, 1], S[-1, 0, 1] →
  S[-1, 1, 1] → S[-1, 2, 1], S[0, 0, 1] → S[0, 1, 1], S[0, 1, 1]
  S[1, 0, 1] → S[1, 1, 1], S[1, 1, 1] → S[1, 2, 1], S[2, 0, 1] → S
  S[2, 1, 1] → S[2, 2, 1], S[-1, -1, 0]  $\xrightleftharpoons{K[4,1]}$  S[-1, -1, 1],
  S[-1, 0, 0]  $\xrightleftharpoons{K[4,1]}$  S[-1, 0, 1], S[-1, 1, 0]  $\xrightleftharpoons{K[4,1]}$  S[-1, 1, 1],
  S[-1, 2, 0]  $\xrightleftharpoons{K[4,1]}$  S[-1, 2, 1], S[0, -1, 0]  $\xrightleftharpoons{K[4,1]}$  S[0, -1, 1]
```

Canonical Forms – Math Models

- ODE's $\tau_i \dot{v}_i = f_i(\{v_j\})$
 $f_i \in \text{closure}(+, -, *, /, \infty, \exp, \log)$

$$\tau_i \dot{v}_i = \sum_{\alpha} c_{i\alpha} \prod_j v_j^{n_{i\alpha j}}$$

$$\tau_i \dot{v}_i = g\left(\sum_j T_{ij} v_j + h_i\right) - \lambda_i v_i$$

- Hybrid systems

$$\tau_i \dot{v}_i = \sum_{\alpha} \Theta(h_{i\alpha}(\mathbf{v})) f_{i\alpha}(\mathbf{v})$$

$$f_{i\alpha}, h_{i\alpha} \in \text{closure}(+, -, *, /, \infty, \exp, \log)$$

Canonical Forms – Math Models

- Variable-structure systems

$$\tau_i \dot{v}_i = A_i f_i \left(v_i, \{ C_{ij} A_j(t) v_j \} \right)$$

$$A_i(t + \Delta t) = \hat{g}_i \left(\mathbf{v}(t), A_i(t), \{ C_{ij} A_j(t) \} \right)$$

$$C_{ij}(t + \Delta t) = h \left(C_{ij}(t), \mathbf{v}(t), \mathbf{v}(t + \Delta t), A(t), A(t + \Delta t) \right)$$

Array Abstractions: Index Domains and Value Fields

- Integer index domain: $i, j \in D(t)$
- Field of vectors (vector-valued array): \mathbf{v}_i
- nbrs = field of domains:

$$f_i = \sum_{j \in \text{nbrs}(i)} g_j = \sum_j C_{ij} g_j$$



Taxonomy of Structure/ Connectivity Dynamics

- $A = \text{const}$, $C(i, j) = A_i A_j C_a(i, j)$
 - C_a from a library
- index arithmetic: $i = f(j)$. Problem: indexing fanout.
- $A = \text{const}$, $C(i, j) = A_i A_j \text{thresh}(\text{const expression})$
 - Function of indices only
 - Function of real constants as well (e.g. fixed positions)
- $A = \text{vbl}$, $C = C(A's, i, j)$
 - $C(A, i, j)$ has alternatives as above
 - May need a real graph data structure e.g. using linked lists, hash tables

Taxonomy of Structure/ Connectivity Dynamics

- $A=vbl, C=C(A's, i, j)$
- $C vbl: C(v_i, v_j, i, j)$
 - frequent connectivity checking
 - Need a real graph structure
 - May need spatial data structures e.g. k-d trees, etc. to check C

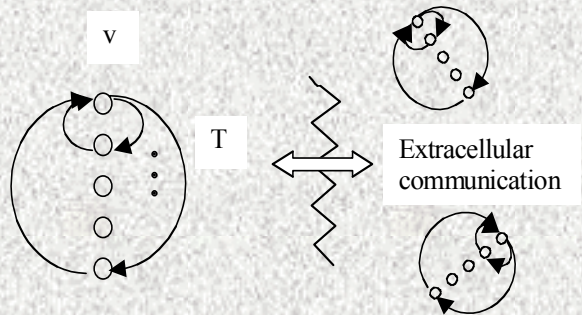
$$\tau_i \dot{v}_i = A_i f_i \left(v_i, \{C_{ij} A_j(t) v_j\} \right)$$

$$A_i(t + \Delta t) = \hat{g}_i \left(\mathbf{v}(t), A_i(t), \{C_{ij} A_j(t)\} \right)$$

$$C_{ij}(t + \Delta t) = h \left(C_{ij}(t), \mathbf{v}(t), \mathbf{v}(t + \Delta t), A(t), A(t + \Delta t) \right)$$

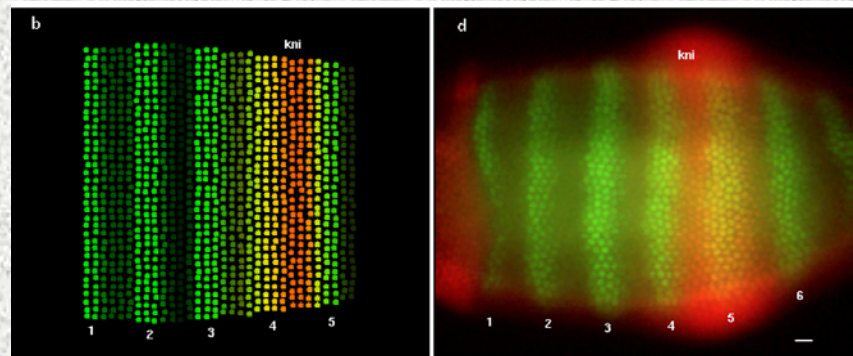
Developmental Model with Transcriptional Regulation

- Gene Regulation Network (GRN) model
- Grammar for changing # of compartments



$$\tau_i \dot{v}_i = g \left(\sum_j T_{ij} v_j + h_i \right) - \lambda_i v_i$$

[J. Theor. Biol. 152: 429-453]



Drosophila eve stripe expression in model (right) and data (left). Green: *eve* expression, red: *kni* expression.

From [Reinitz and Sharp, Mech. of Devel., 49:133-158, 1995].

Variable-Structure Systems: Developmental Grammars

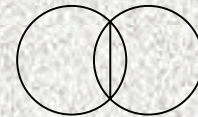
$$\tau_{i\alpha} \frac{d\mathbf{v}_{i\alpha}}{dt} = A_i \left[\mathbf{g}(\mathbf{u}_{i\alpha}) - \lambda_{i\alpha} \mathbf{v}_{i\alpha} \right]$$

$$\mathbf{u}_{i\alpha} = \sum_r \chi_i^r \hat{\Gamma}_{\alpha;\alpha}^r \mathbf{T}^r \cdot \mathbf{v}_{i\alpha} + \sum_r \chi_i^r \sum_\beta \hat{\Gamma}_{\alpha\beta;\alpha}^r \Lambda_{ij} \mathbf{T}_\beta^r \cdot \mathbf{v}_{j\beta} + \mathbf{h}_{i\alpha}$$

$$A_{i_1 \dots i_{n+1}, \alpha}(t + \Delta t) = \sum_r \chi_{i_1 \dots i_{n+1}}^r(t) \hat{\Gamma}_{\alpha;\alpha}^r A_{i_1 \dots i_{n+1}, \alpha}(t) + \sum_r \chi_{i_1 \dots i_n}^r(t) \hat{\Gamma}_{\beta; \alpha i_{n+1}}^r A_{i_1 \dots i_n, \beta}(t) + \dots$$

$$C_{ij}(t + \Delta t) = \Theta(\Lambda_{ij}(t + \Delta t))$$

$$\Lambda_{ij}(t) = \sum_{\alpha\beta} A_{i\alpha} A_{j\beta} \Lambda_{\alpha\beta}(\mathbf{x}_i - \mathbf{x}_j, \rho_{i\alpha}, \rho_{j\alpha})$$

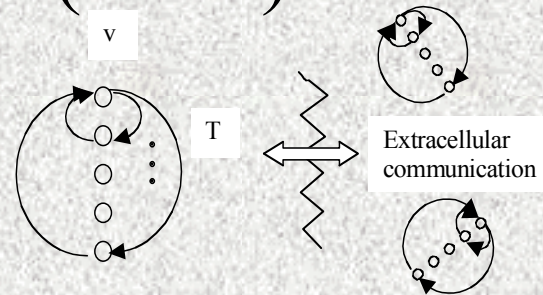


$$\frac{d\mathbf{x}_i}{dt} = \eta \left(F_i^{\text{external}} + \nabla_{\mathbf{x}_i} \sum_j \sum_{\alpha\beta} A_{i\alpha} A_{j\beta} \Lambda_{\alpha\beta}(\mathbf{x}_i - \mathbf{x}_j, \rho_{i\alpha}, \rho_{j\alpha}) \mathbf{v}_{i\alpha} \cdot \mathbf{Q}_{\alpha\beta} \cdot \mathbf{v}_{j\beta} \right)$$

Modeling large complexes: Transcriptional Regulation

- Gene Regulation Network (GRN) model

$$\tau_i \dot{v}_i = g \left(\sum_j T_{ij} v_j + h_i \right) - \lambda_i v_i$$



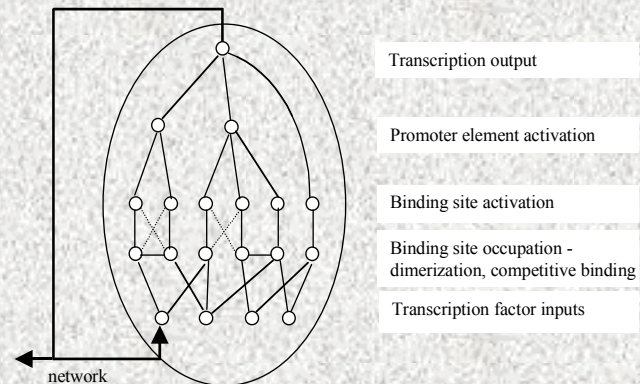
- Heirarchical Cooperative Activation Model

$$\tau_i \dot{X}_i = \frac{J u_i}{1 + J u_i} - \lambda_i X_i$$

$$u_i = \prod_{\alpha \in i} \frac{1 + J_{\alpha} \tilde{v}_{\alpha}}{1 + \hat{J}_{\alpha} \tilde{v}_{\alpha}}$$

$$\tilde{v}_{\alpha} = \frac{\tilde{K}_{\alpha} \tilde{u}_{\alpha}}{1 + \tilde{K}_{\alpha} \tilde{u}_{\alpha}}$$

$$\tilde{u}_{\alpha} = \prod_{b \in \alpha} \frac{1 + K_b v_{j(b)}^{n(b)}}{1 + \hat{K}_b v_{j(b)}^{n(b)}}$$



Mixing Different Types of Reactions

- Cellerator Arrows Here (qv.)
- Enzyme kinetics
- Transcriptional reg alternative models
 - Hill's function for single inputs
 - GRN
 - HCA including competitive binding

Circuit Inference

- Scoring functions
 - Minimize deviation from observed behavior
 - Maximize fitness in a simulation
- Methods
 - Local minimum finders
 - Simulated annealing – Lam-Delosme version (qv.)
 - Genetic Algorithms – but maintain diversity

Conclusions

- Labeling individual species may require variable-length lists or even trees.
- There is a progression of canonical forms for dynamics including connectivity, corresponding to a progression of required specifications, data structures and solvers.
- Please implement Lam-Delosme simulated annealing algorithm for circuit inference.

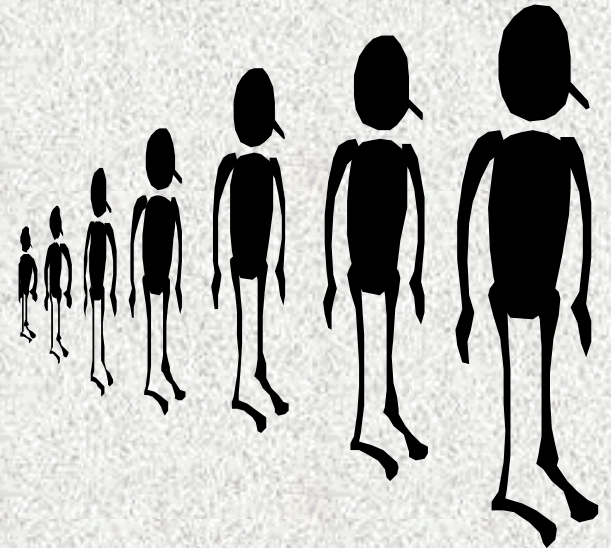
JPL

Canonical Forms – Math Models

- ODE's $\tau_i \dot{v}_i = \sum_{\alpha} c_{i\alpha} \prod_j v_j^{n_{\alpha j}}$, or
 $\tau_i \dot{v}_i = g\left(\sum_j T_{ij} v_j + h_i\right) - \lambda_i v_i$, or
 $\tau_i \dot{v}_i = f_i(\{v_j\})$ $f_i \in \text{closure}(+, -, *, /, \infty, \exp, \log)$
- Hybrid systems $\tau_i \dot{v}_i = \sum_{\alpha} \Theta(h_{i\alpha}(\mathbf{v})) f_{i\alpha}(\mathbf{v})$
 $f_{i\alpha}, h_{i\alpha} \in \text{closure}(+, -, *, /, \infty, \exp, \log)$
- Variable-Structure Systems
 $\tau_i \dot{v}_i = A_i f_i(\mathbf{v})$
 $A_i(t + \Delta t) = \hat{g}_i(\mathbf{v}(t), A_i(t), \{C_{ij} A_i(t)\})$
 $C_{ij}(t + \Delta t) = h(C_{ij}(t), \mathbf{v}(t), \mathbf{v}(t + \Delta t), A(t), A(t + \Delta t))$

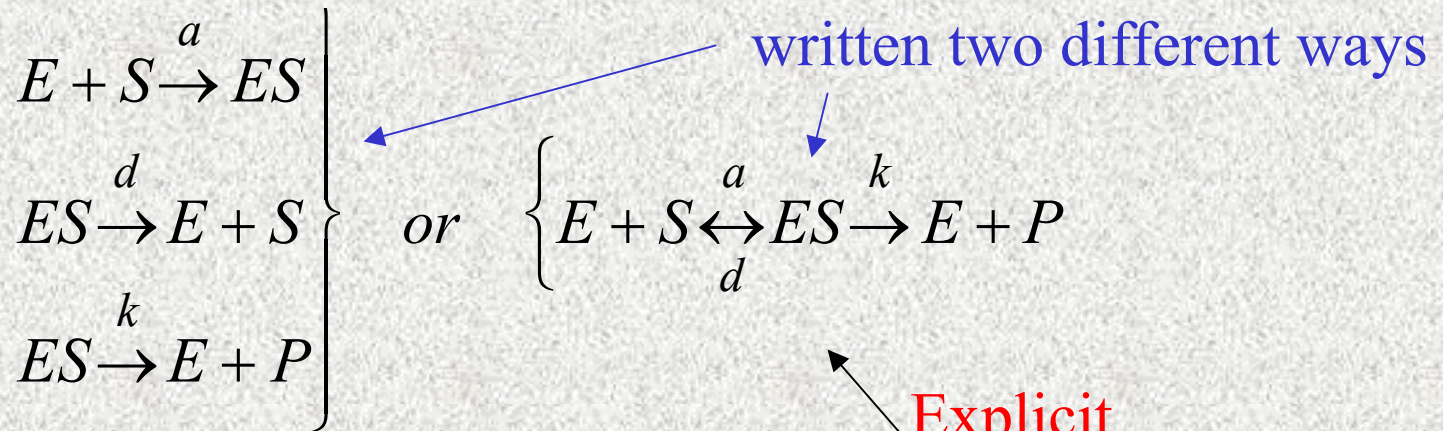
Types of Interactions

- Association/Dissassociation $A + B \rightarrow AB, AB \rightarrow A + B$
- Conversion $A \rightarrow B$
- Degradation $A \rightarrow \phi$
- Creation $\phi \rightarrow A$
- Enzymatic $S \xrightarrow{E} P$ $S \xrightleftharpoons[e]{E} P$
- Transcription (Gene \Rightarrow RNA)
- Post-transcriptional Processing
- Translation (RNA \Rightarrow Protein)
- Diffusion
- and more ...

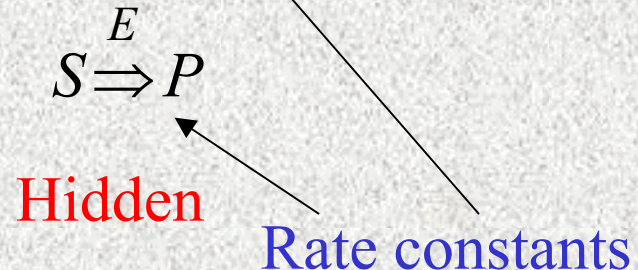


Enzyme Kinetic (Catalytic) Reaction

- Enzyme E catalyzes the production of product P from substrate (source) S

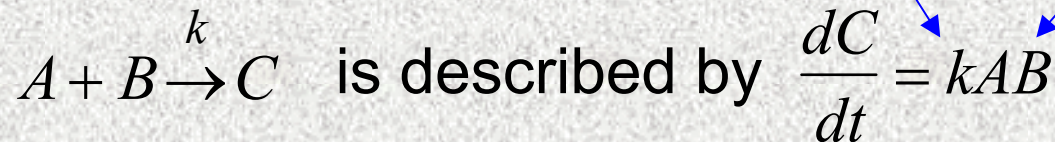


- Write more compactly as



Translation of Biochemical Formula to ODE

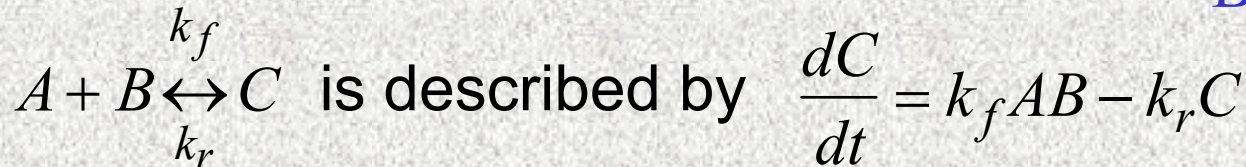
- Law of Mass Action



rate constant

Concentrations

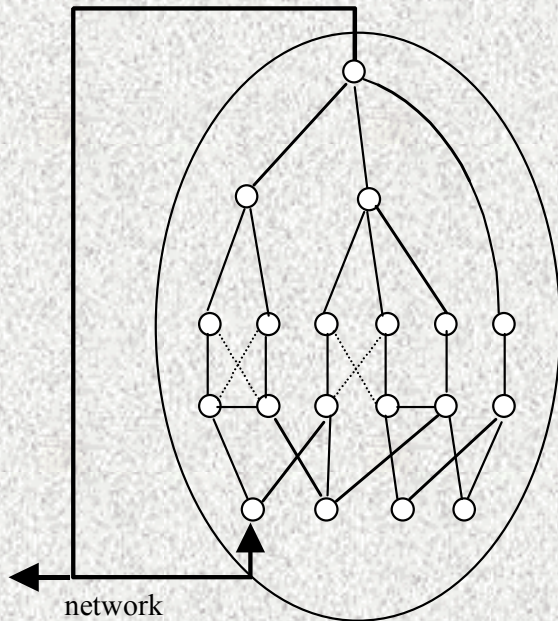
- Two-way Reaction



Similar ODE's
can be written for
B and C

- Complex reactions built from simple reactions

Modeling Eucaryotic Transcription Complexes: Hierarchical Cooperative Activation (HCA) Model



Transcription output

Promoter element activation

Binding site activation

Binding site occupation -
dimerization, competitive binding

Transcription factor inputs

$$\tau_i \frac{dv_i}{dt} = [\textit{transcribing}]_i - \lambda_i v_i$$

$$[\textit{transcribing}]_i = g(u_i) = \frac{J u_i}{1 + J u_i}$$

$$u_i = \prod_{\alpha \in i} \left(\frac{1 + J_{\alpha} P_{\alpha}}{1 + \hat{J}_{\alpha} P_{\alpha}} \right)$$

$$P_{\alpha} = g_{\alpha}(\tilde{u}_{\alpha}) = \frac{\tilde{K}_{\alpha} \tilde{u}_{\alpha}}{1 + \tilde{K}_{\alpha} \tilde{u}_{\alpha}}$$

$$\tilde{u}_{\alpha} = \prod_{b \in \alpha} \left(\frac{1 + K_b v_{j(b)}^{n(b)}}{1 + \hat{K}_b v_{j(b)}^{n(b)}} \right)$$

$$f_{cb} = \frac{K_b v_{j(b)}^{n(b)}}{1 + K_b v_{j(b)}^{n(b)}} \quad \hat{f}_{cb} = \frac{\hat{K}_b v_{j(b)}^{n(b)}}{1 + \hat{K}_b v_{j(b)}^{n(b)}}$$