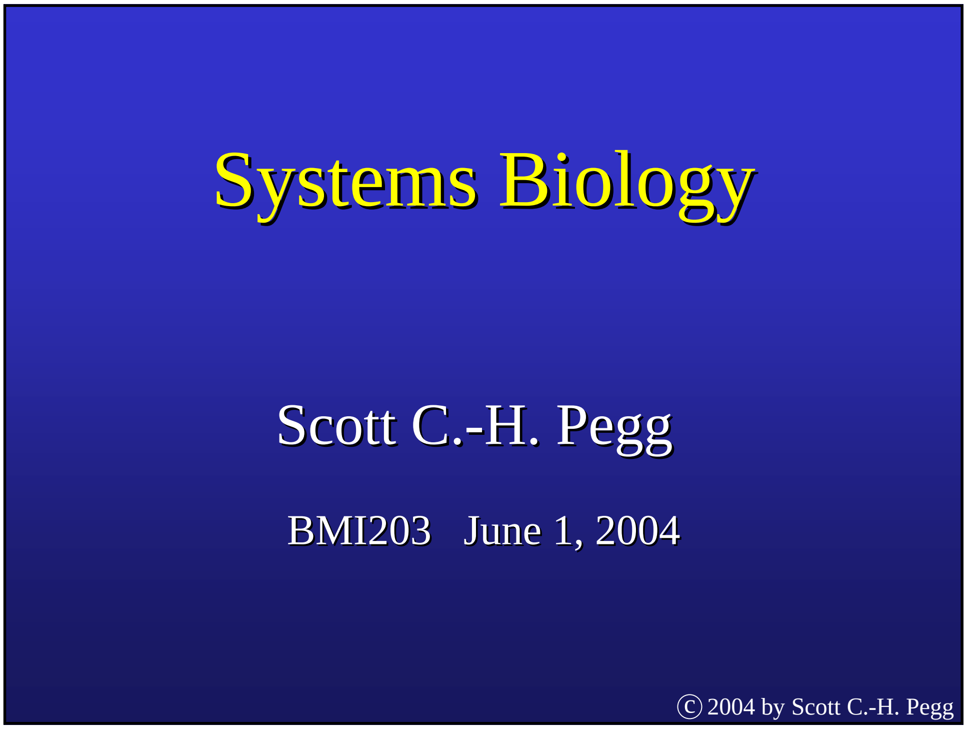
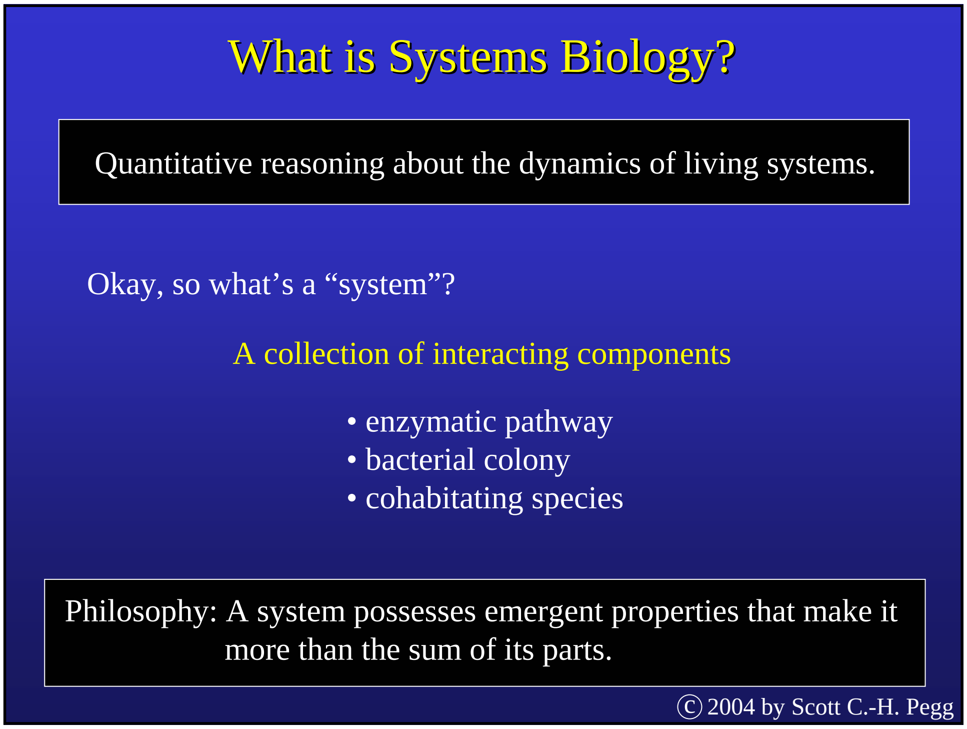


Systems Biology

Scott C.-H. Pegg

BMI203   June 1, 2004



What is Systems Biology?

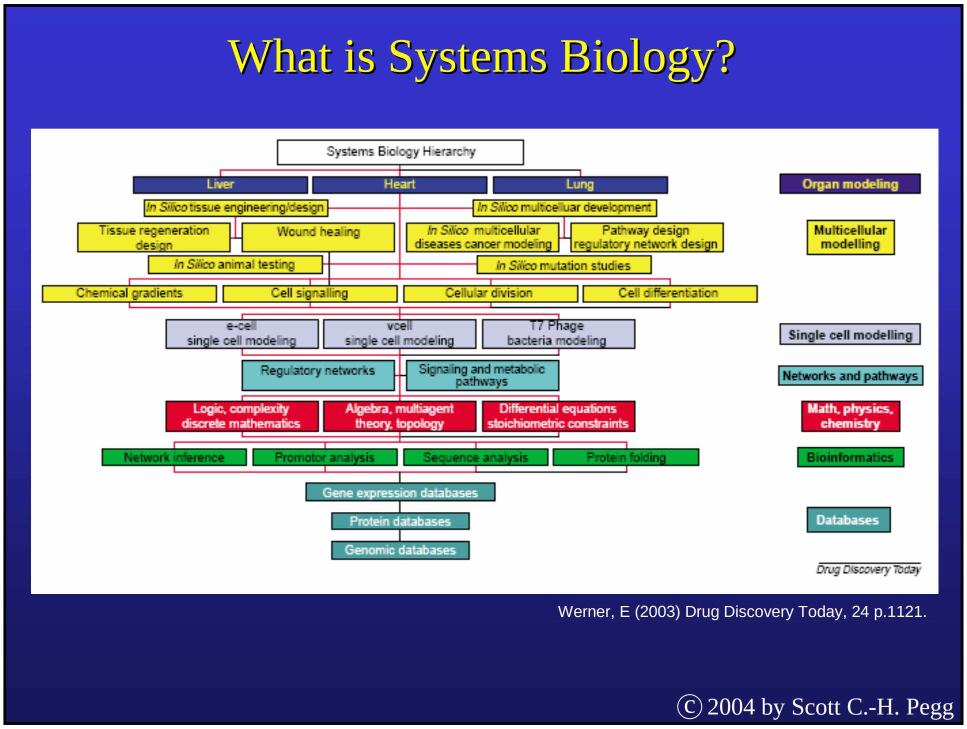
Quantitative reasoning about the dynamics of living systems.

Okay, so what’s a “system”?

A collection of interacting components

- enzymatic pathway
- bacterial colony
- cohabitating species

Philosophy: A system possesses emergent properties that make it more than the sum of its parts.

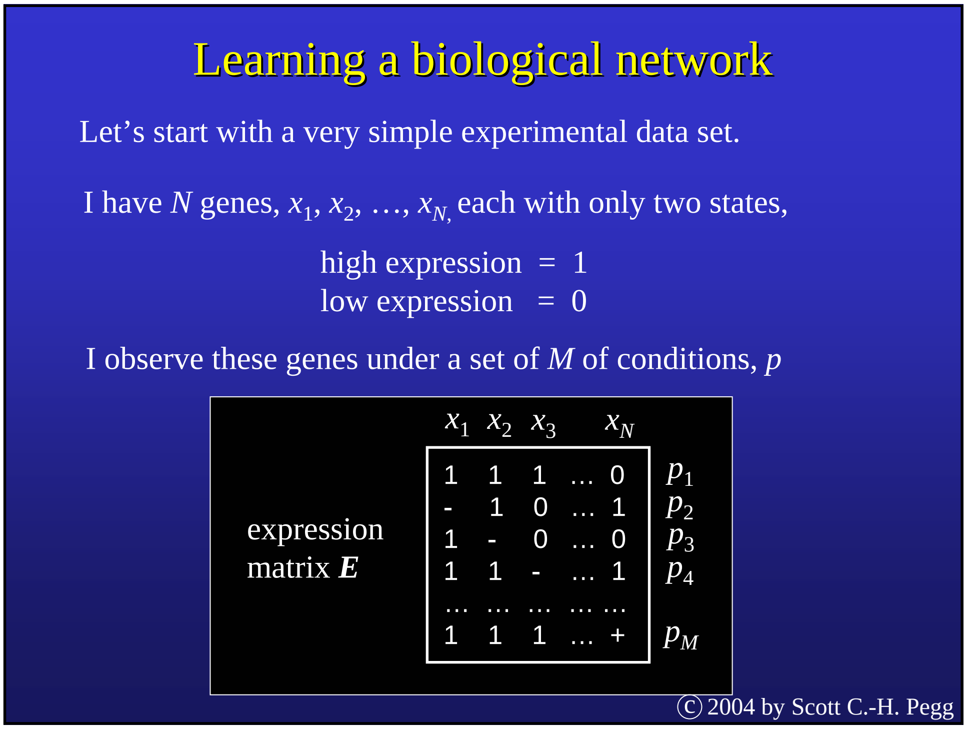


What is Systems Biology?

Systems Biology Hierarchy
Liver
Heart
Lung
Organ modeling
In Silico tissue engineering/design
In Silico multicelluar development
Tissue regeneration design
Wound healing
In Silico multicellular diseases cancer modeling
Pathway design regulatory network design
Multicellular modelling
In Silico animal testing
In Silico mutation studies
Chemical gradients
Cell signalling
Cellular division
Cell differentiation
e-cell single cell modeling
vcell single cell modeling
T7 Phage bacteria modeling
Single cell modelling
Regulatory networks
Signaling and metabolic pathways
Networks and pathways
Logic, complexity discrete mathematics
Algebra, multiagent theory, topology
Differential equations stoichiometric constraints
Math, physics, chemistry
Network inference
Promotor analysis
Sequence analysis
Protein folding
Bioinformatics
Gene expression databases
Protein databases
Genomic databases
Databases
Drug Discovery Today

Werner, E (2003) Drug Discovery Today, 24 p.1121.

Ⓒ 2004 by Scott C.-H. Pegg



Learning a biological network

Let’s start with a very simple experimental data set.

I have *N* genes, x_1, x_2, …, $x_{N,}$ each with only two states,

high expression = 1
low expression = 0

I observe these genes under a set of *M* of conditions, *p*

expression matrix ***E***

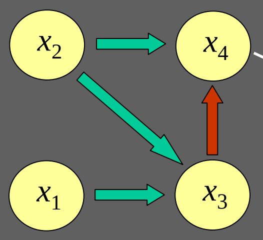
$$\begin{array}{cccccc} x_1 & x_2 & x_3 & & x_N & \\ 1 & 1 & 1 & \dots & 0 & p_1 \\ - & 1 & 0 & \dots & 1 & p_2 \\ 1 & - & 0 & \dots & 0 & p_3 \\ 1 & 1 & - & \dots & 1 & p_4 \\ \dots & \dots & \dots & \dots & \dots & \\ 1 & 1 & 1 & \dots & + & p_M \end{array}$$

Ⓒ 2004 by Scott C.-H. Pegg

Learning a biological network

I assume that any given gene either

(1) influences another gene to increase expression
(2) influences another gene to decrease expression
(3) has no influence on a given gene



x_2	1	0	1	0
x_3	1	1	0	0
x_4	**0**	**0**	**1**	**0**

"truth table"

Learning a biological network

How do we connect the genes in a network that is consistent with my observed expressions?

For each gene x_i, we determine a minimum set of genes whose levels must be included as input to x_i's truth table.

We consider all pairs of rows (i, j) in E such that the expression level of x_i differs.

x_1	x_2	x_3	x_4	
1	1	1	0	p_1
-	1	0	1	p_2
1	-	0	0	p_3
1	1	-	1	p_4
1	1	1	+	p_5

for x_4, I consider rows

(1, 2)
(1, 4)
(2, 3)
(3, 4)

Learning a biological network

For each pair, we build the set of genes which also differed.

x_1	x_2	x_3	x_4	
1	1	1	0	p_1
-	1	0	1	p_2
1	-	0	0	p_3
1	1	-	1	p_4
1	1	1	+	p_5

(1, 2) = $\{x_1, x_3\}$

(1, 4) = $\{x_3\}$

(2, 3) = $\{x_1, x_2\}$

(3, 4) = $\{x_2\}$

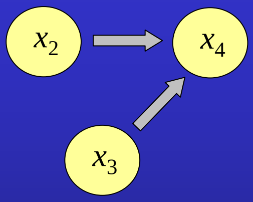
To construct the network with the minimum number of required edges, we want the smallest number of nodes required to explain the changes in x_i's expression.

For x_4, this set is $\{x_2, x_3\}$

This is a classic problem known as minimum set covering, and is solved by well-known applications of branch-and-bound algorithms.

Learning a biological network

The genes in the covering set are connected to x_i,

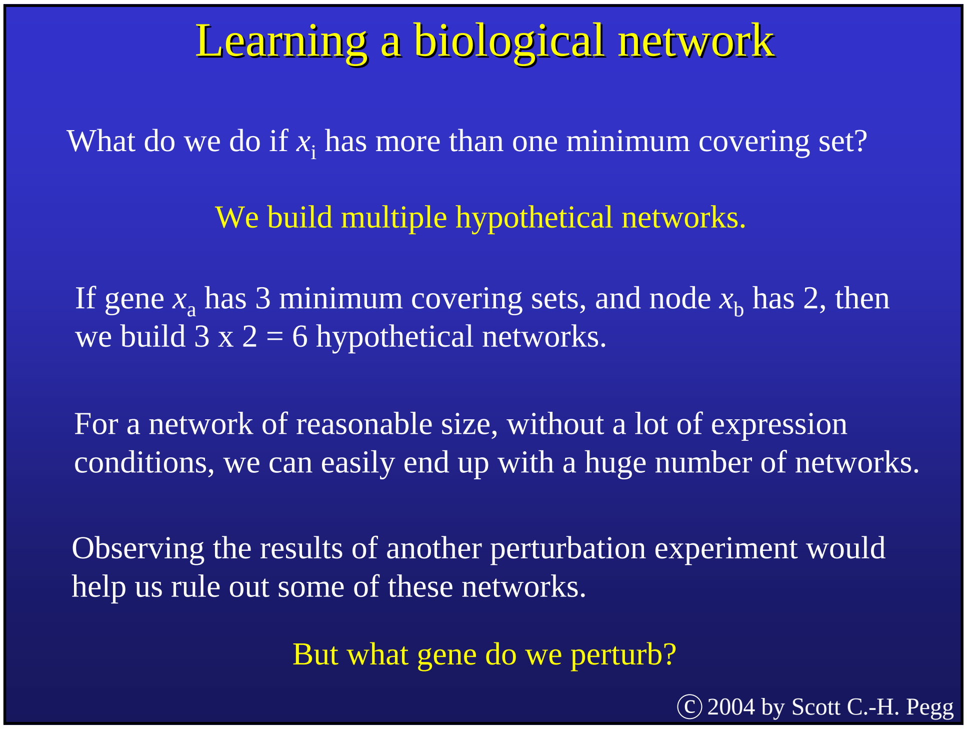


covering set of $x_4 = \{x_2, x_3\}$

We now take our covering set and construct a truth table for x_i

x_1	x_2	x_3	x_4	
1	1	1	0	p_1
-	1	0	1	p_2
1	-	0	0	p_3
1	1	-	1	p_4
1	1	1	+	p_5

x_2	1	0	1	0
x_3	1	1	0	0
x_4	**0**	*****	**1**	**0**



Learning a biological network

What do we do if x_i has more than one minimum covering set?

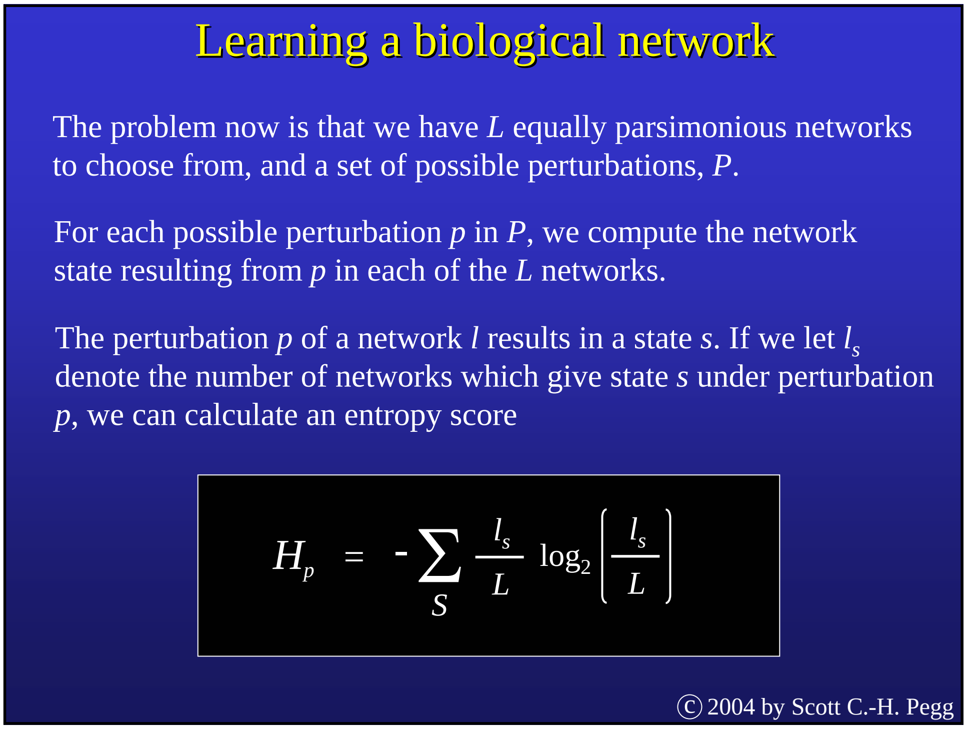
We build multiple hypothetical networks.

If gene x_a has 3 minimum covering sets, and node x_b has 2, then we build 3 x 2 = 6 hypothetical networks.

For a network of reasonable size, without a lot of expression conditions, we can easily end up with a huge number of networks.

Observing the results of another perturbation experiment would help us rule out some of these networks.

But what gene do we perturb?



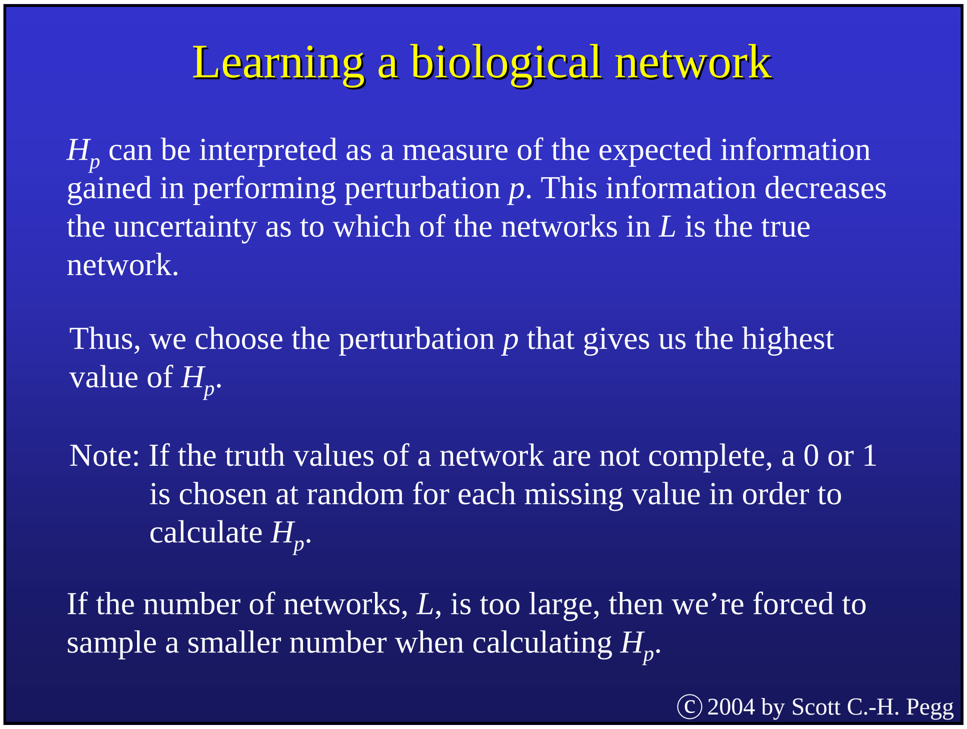
Learning a biological network

The problem now is that we have L equally parsimonious networks to choose from, and a set of possible perturbations, P.

For each possible perturbation p in P, we compute the network state resulting from p in each of the L networks.

The perturbation p of a network l results in a state s. If we let l_s denote the number of networks which give state s under perturbation p, we can calculate an entropy score

$$H_p = -\sum_S \frac{l_s}{L} \log_2\left(\frac{l_s}{L}\right)$$



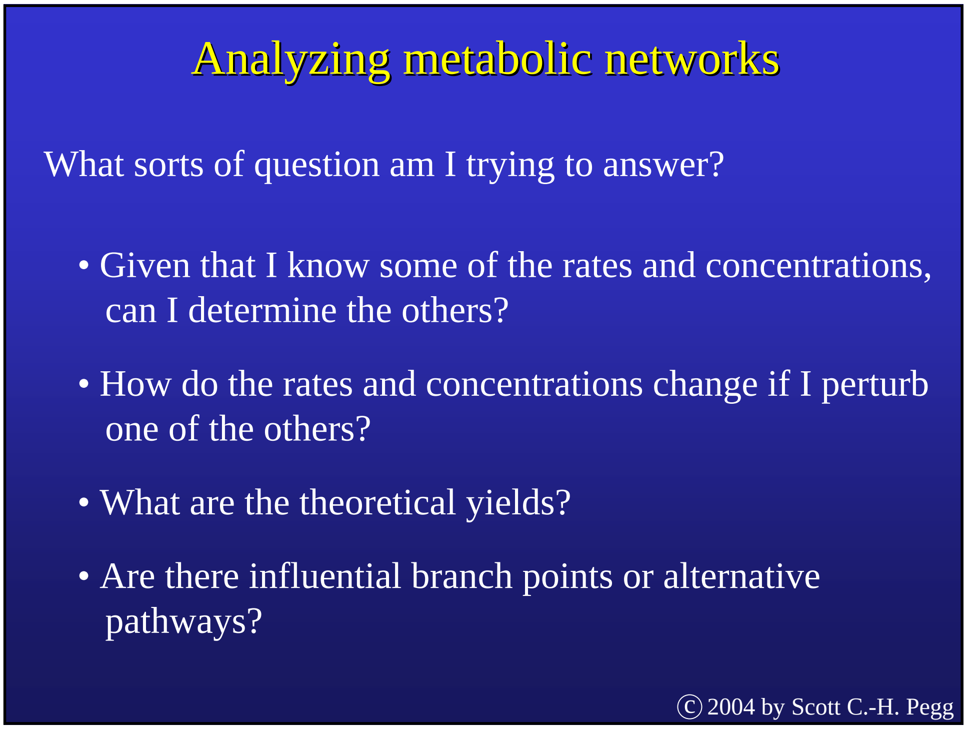
Learning a biological network

H_p can be interpreted as a measure of the expected information gained in performing perturbation p. This information decreases the uncertainty as to which of the networks in L is the true network.

Thus, we choose the perturbation p that gives us the highest value of H_p.

Note: If the truth values of a network are not complete, a 0 or 1 is chosen at random for each missing value in order to calculate H_p.

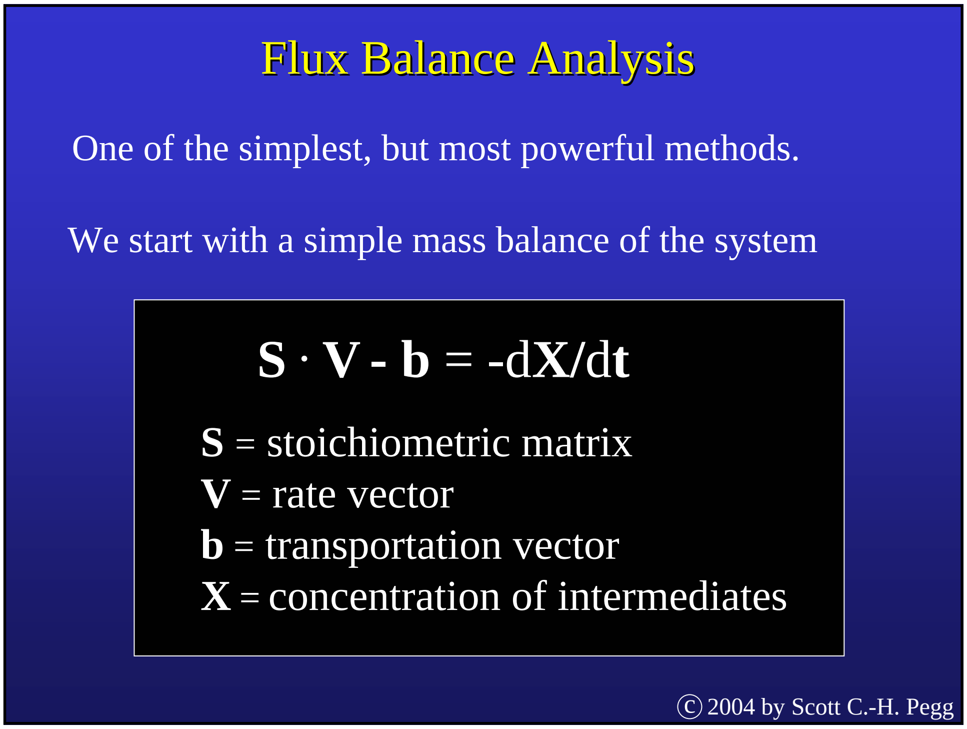
If the number of networks, L, is too large, then we're forced to sample a smaller number when calculating H_p.



Analyzing metabolic networks

What sorts of question am I trying to answer?

- Given that I know some of the rates and concentrations, can I determine the others?
- How do the rates and concentrations change if I perturb one of the others?
- What are the theoretical yields?
- Are there influential branch points or alternative pathways?



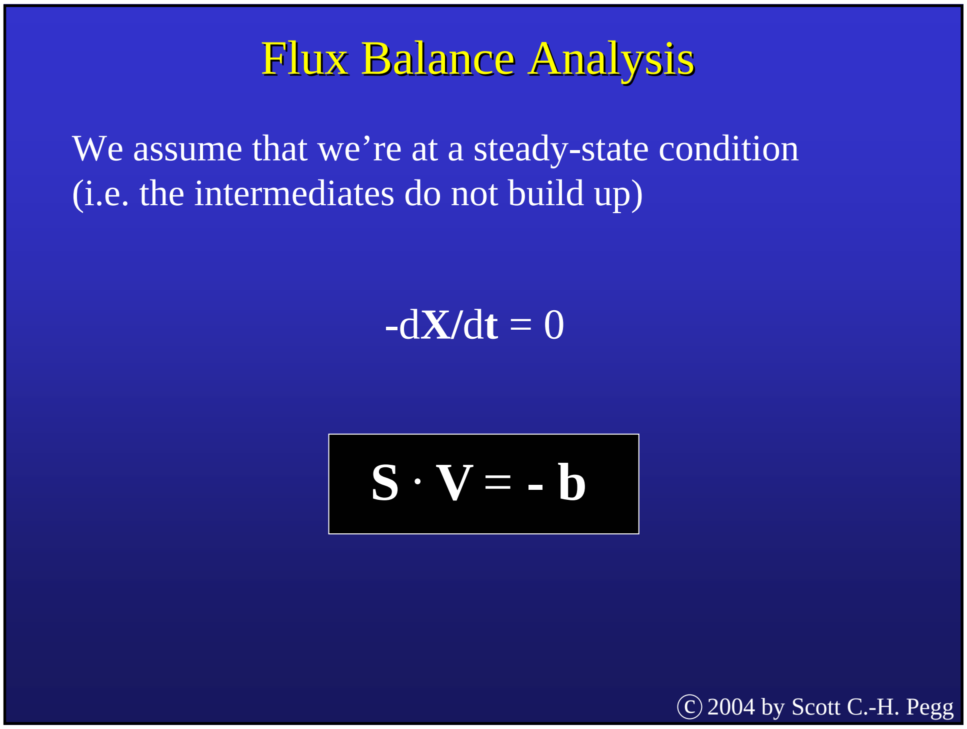
Flux Balance Analysis

One of the simplest, but most powerful methods.

We start with a simple mass balance of the system

$$\mathbf{S} \cdot \mathbf{V} - \mathbf{b} = -d\mathbf{X}/dt$$

S = stoichiometric matrix
V = rate vector
b = transportation vector
X = concentration of intermediates

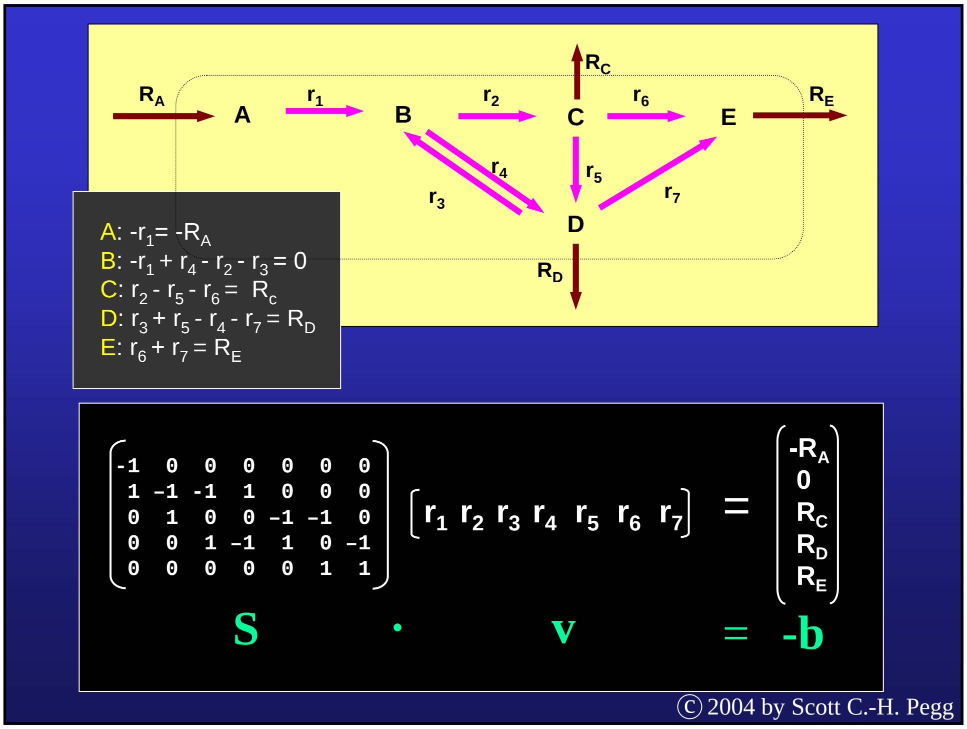


Flux Balance Analysis

We assume that we’re at a steady-state condition (i.e. the intermediates do not build up)

$$-d\mathbf{X}/dt = 0$$

$$\boxed{\mathbf{S} \cdot \mathbf{V} = -\mathbf{b}}$$



A: $-r_1 = -R_A$

B: $-r_1 + r_4 - r_2 - r_3 = 0$

C: $r_2 - r_5 - r_6 = R_c$

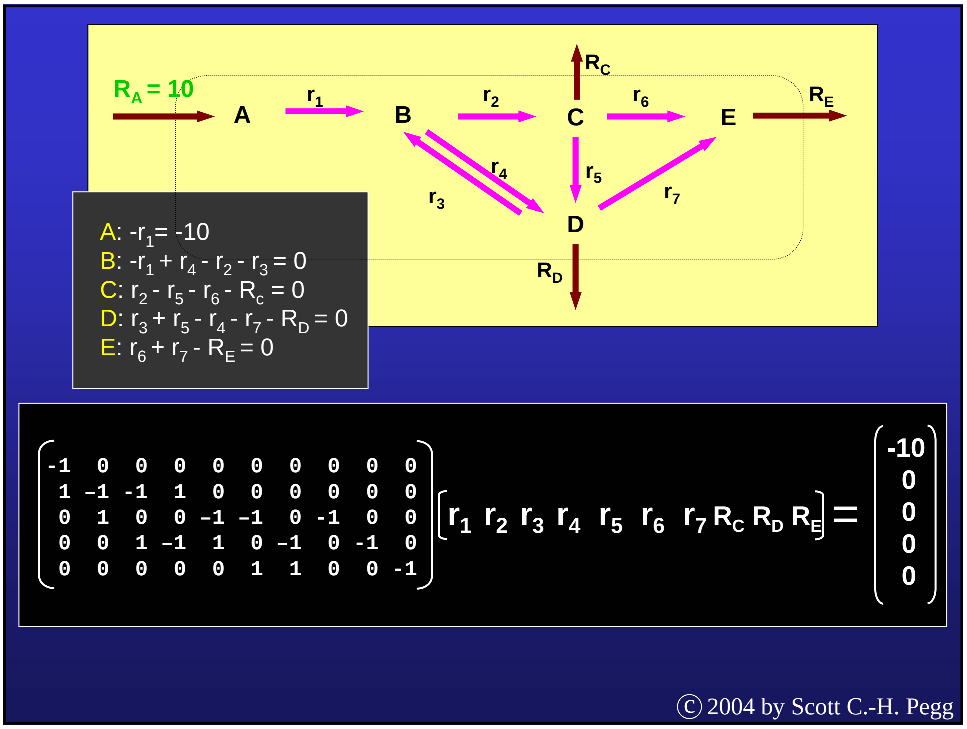
D: $r_3 + r_5 - r_4 - r_7 = R_D$

E: $r_6 + r_7 = R_E$

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{bmatrix} r_1 & r_2 & r_3 & r_4 & r_5 & r_6 & r_7 \end{bmatrix} = \begin{pmatrix} -R_A \\ 0 \\ R_C \\ R_D \\ R_E \end{pmatrix}$$

$$\mathbf{S} \quad \cdot \quad \mathbf{v} \quad = \quad \mathbf{-b}$$

Ⓒ 2004 by Scott C.-H. Pegg



A: $-r_1 = -10$

B: $-r_1 + r_4 - r_2 - r_3 = 0$

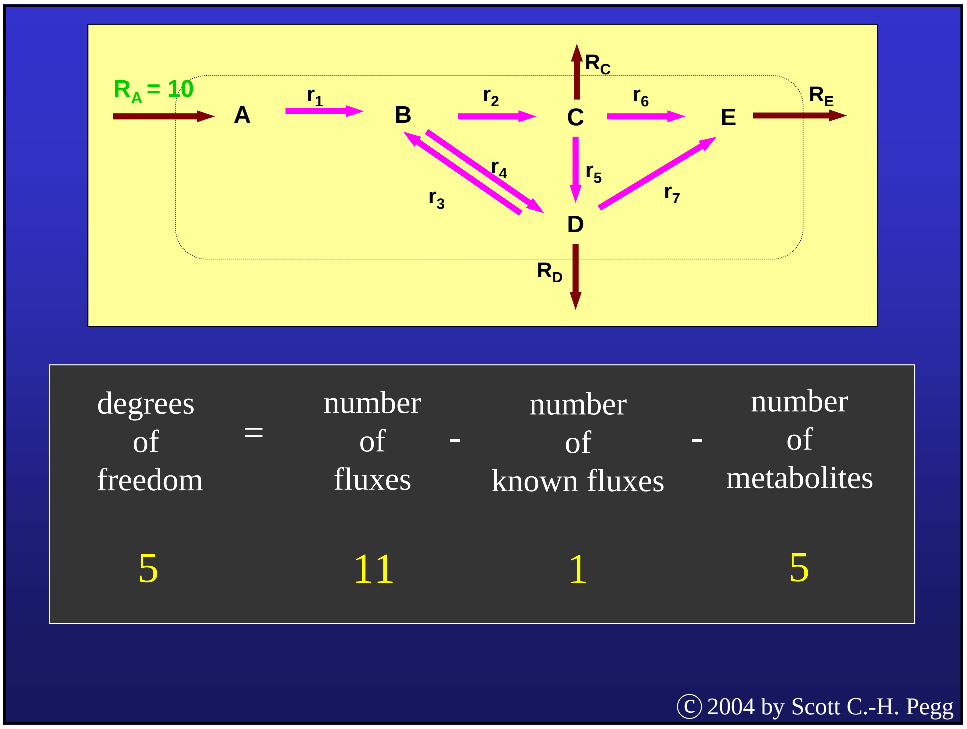
C: $r_2 - r_5 - r_6 - R_c = 0$

D: $r_3 + r_5 - r_4 - r_7 - R_D = 0$

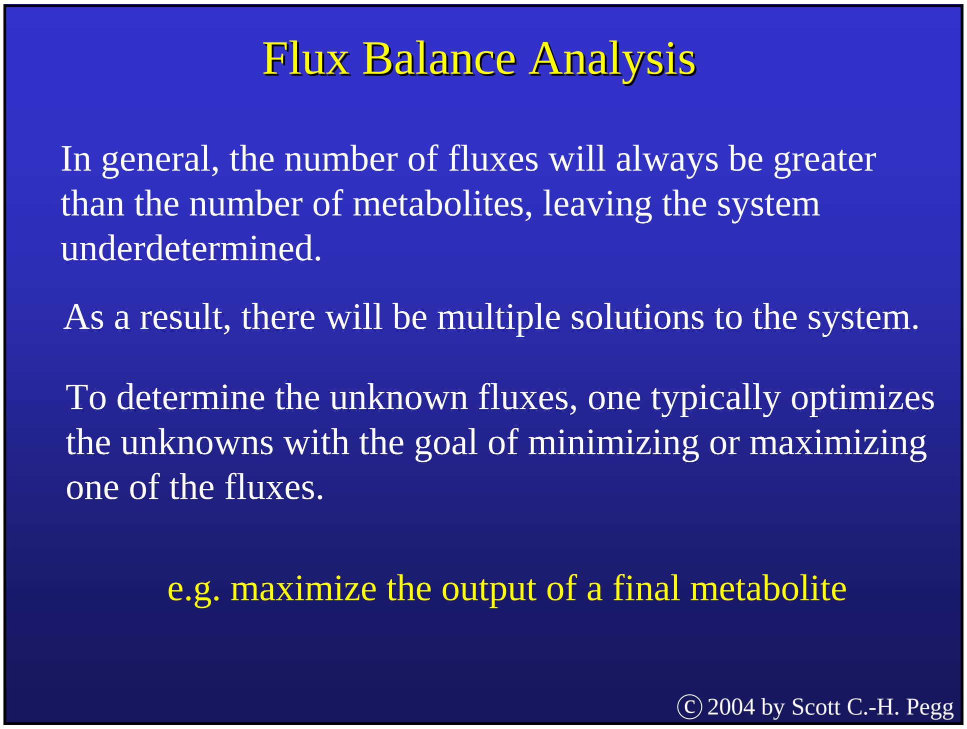
E: $r_6 + r_7 - R_E = 0$

$$\begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & -1 \end{pmatrix} \begin{bmatrix} r_1 & r_2 & r_3 & r_4 & r_5 & r_6 & r_7 & R_C & R_D & R_E \end{bmatrix} = \begin{pmatrix} -10 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Ⓒ 2004 by Scott C.-H. Pegg



degrees of freedom	=	number of fluxes	-	number of known fluxes	-	number of metabolites
5		11		1		5



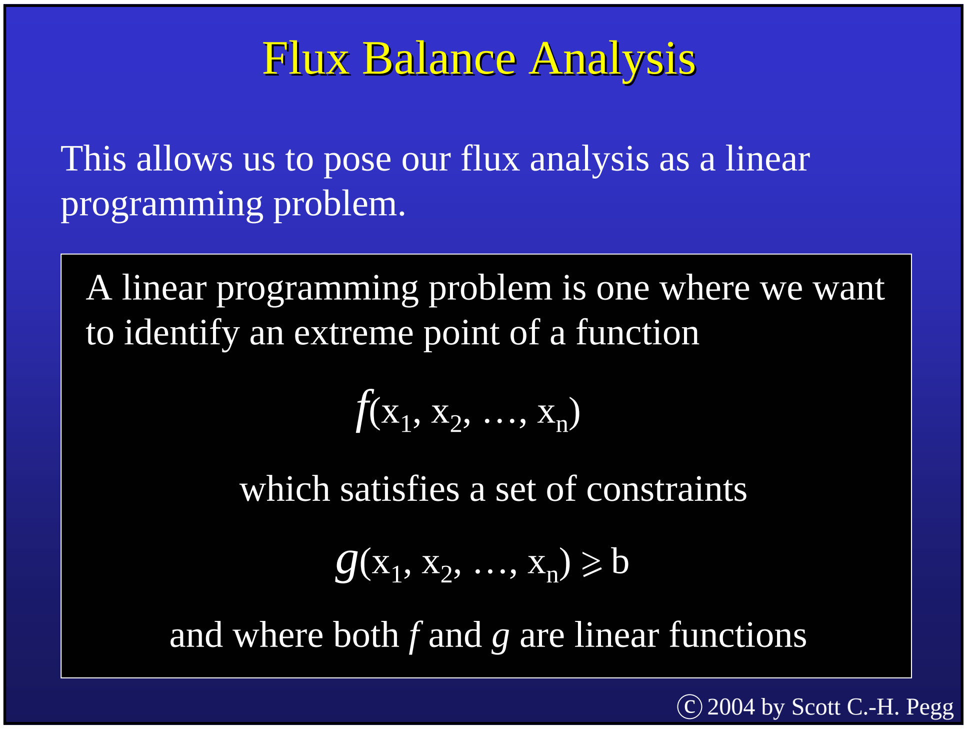
Flux Balance Analysis

In general, the number of fluxes will always be greater than the number of metabolites, leaving the system underdetermined.

As a result, there will be multiple solutions to the system.

To determine the unknown fluxes, one typically optimizes the unknowns with the goal of minimizing or maximizing one of the fluxes.

e.g. maximize the output of a final metabolite



Flux Balance Analysis

This allows us to pose our flux analysis as a linear programming problem.

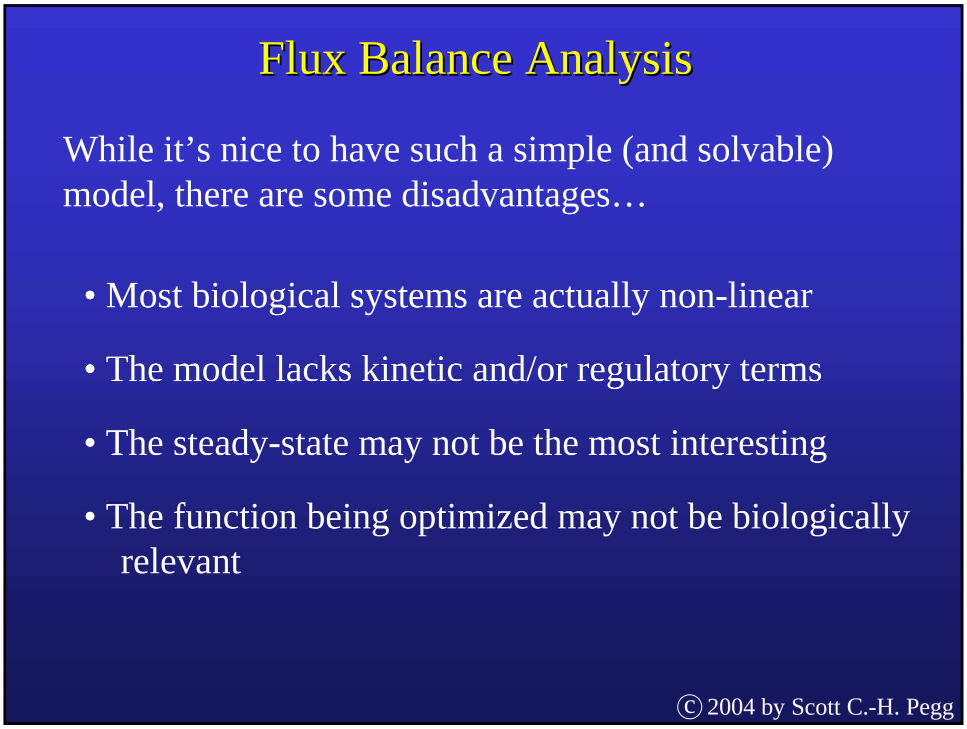
A linear programming problem is one where we want to identify an extreme point of a function

$$f(x_1, x_2, \ldots, x_n)$$

which satisfies a set of constraints

$$g(x_1, x_2, \ldots, x_n) \geqslant b$$

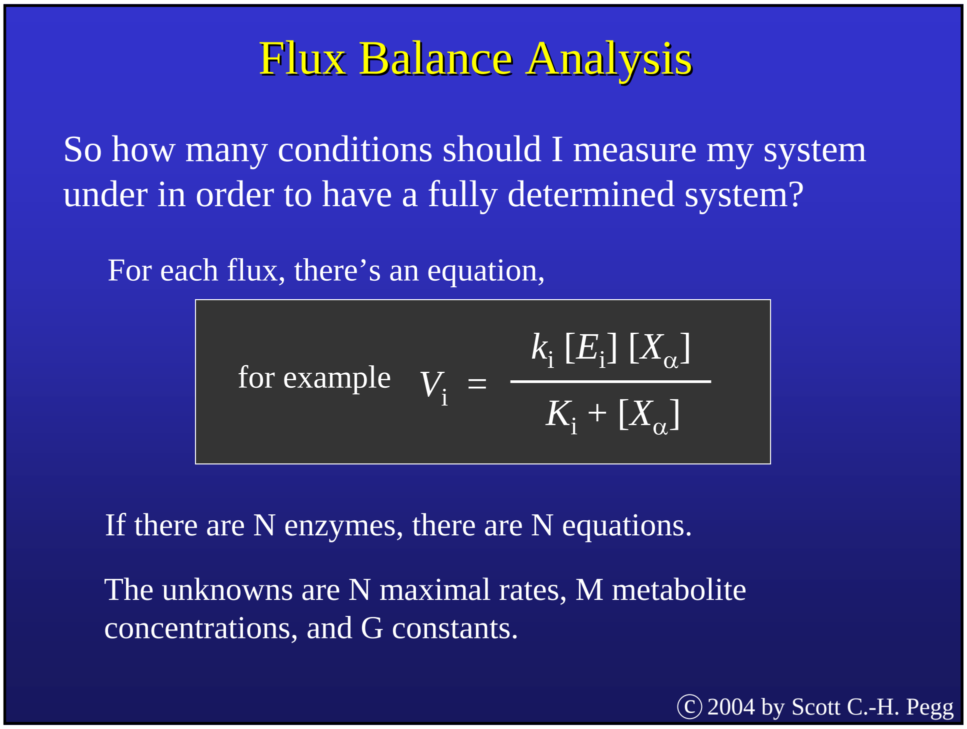
and where both *f* and *g* are linear functions



Flux Balance Analysis

While it’s nice to have such a simple (and solvable) model, there are some disadvantages…

- Most biological systems are actually non-linear
- The model lacks kinetic and/or regulatory terms
- The steady-state may not be the most interesting
- The function being optimized may not be biologically relevant



Flux Balance Analysis

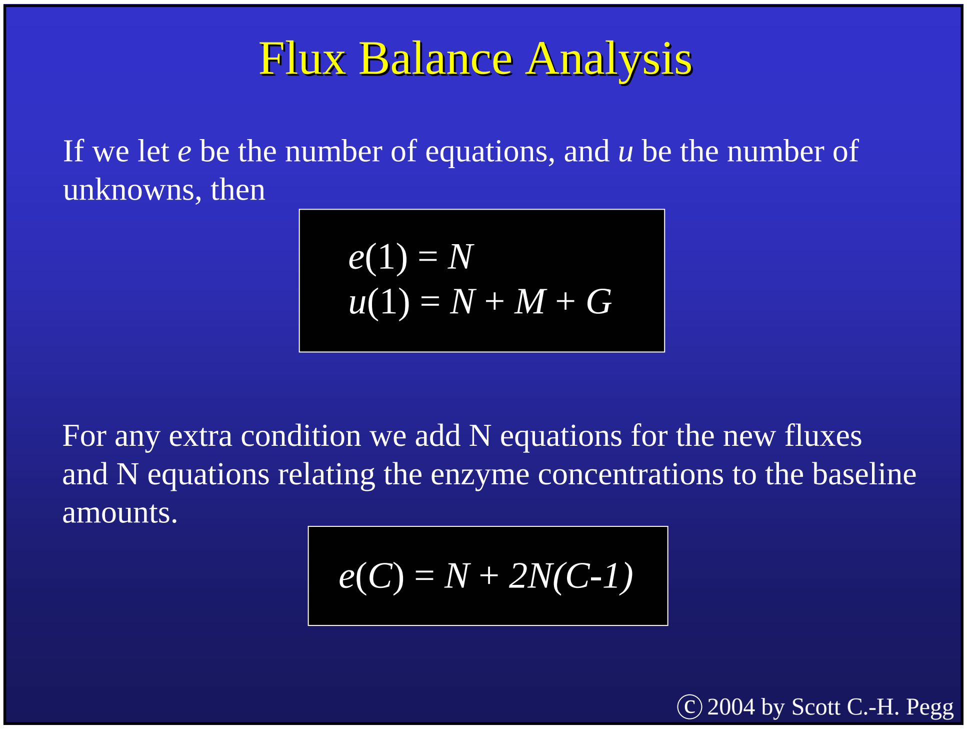
So how many conditions should I measure my system under in order to have a fully determined system?

For each flux, there's an equation,

for example $$V_{\mathrm{i}} = \frac{k_{\mathrm{i}}\,[E_{\mathrm{i}}]\,[X_\alpha]}{K_{\mathrm{i}} + [X_\alpha]}$$

If there are N enzymes, there are N equations.

The unknowns are N maximal rates, M metabolite concentrations, and G constants.



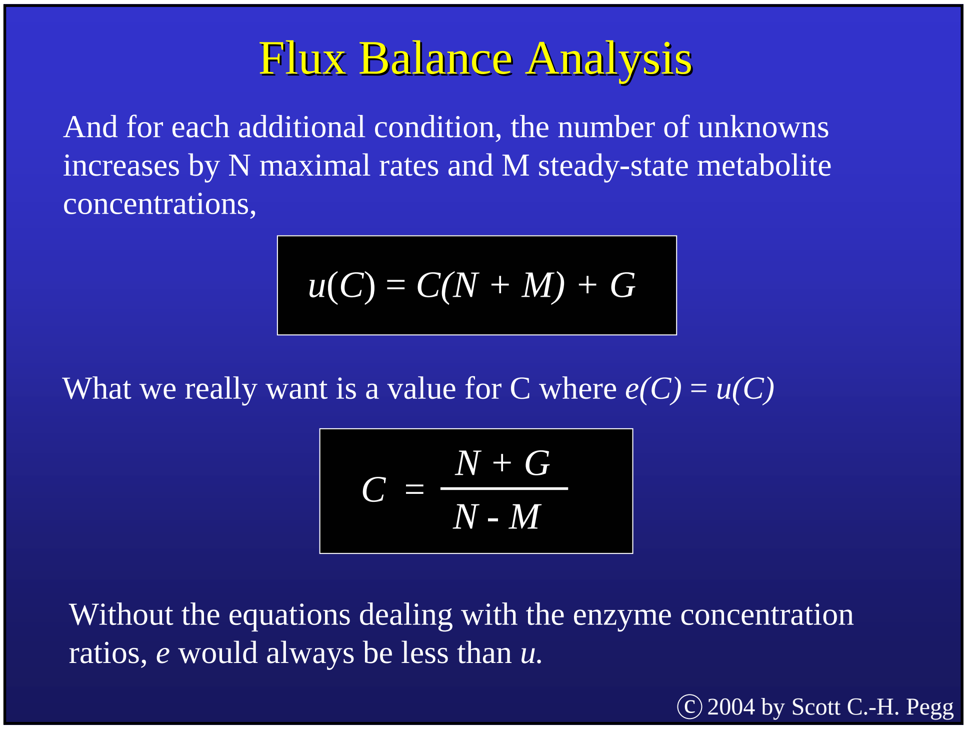
Flux Balance Analysis

If we let *e* be the number of equations, and *u* be the number of unknowns, then

$$e(1) = N$$
$$u(1) = N + M + G$$

For any extra condition we add N equations for the new fluxes and N equations relating the enzyme concentrations to the baseline amounts.

$$e(C) = N + 2N(C\text{-}1)$$



Flux Balance Analysis

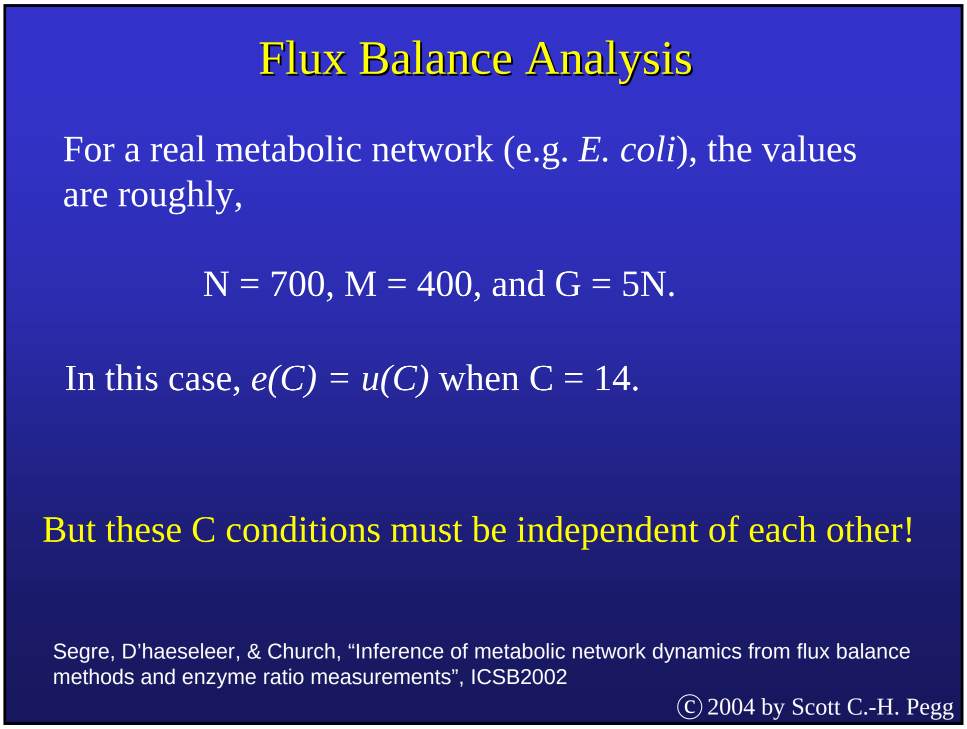
And for each additional condition, the number of unknowns increases by N maximal rates and M steady-state metabolite concentrations,

$$\boxed{u(C) = C(N + M) + G}$$

What we really want is a value for C where *e(C)* = *u(C)*

$$\boxed{C = \frac{N + G}{N - M}}$$

Without the equations dealing with the enzyme concentration ratios, *e* would always be less than *u*.



Flux Balance Analysis

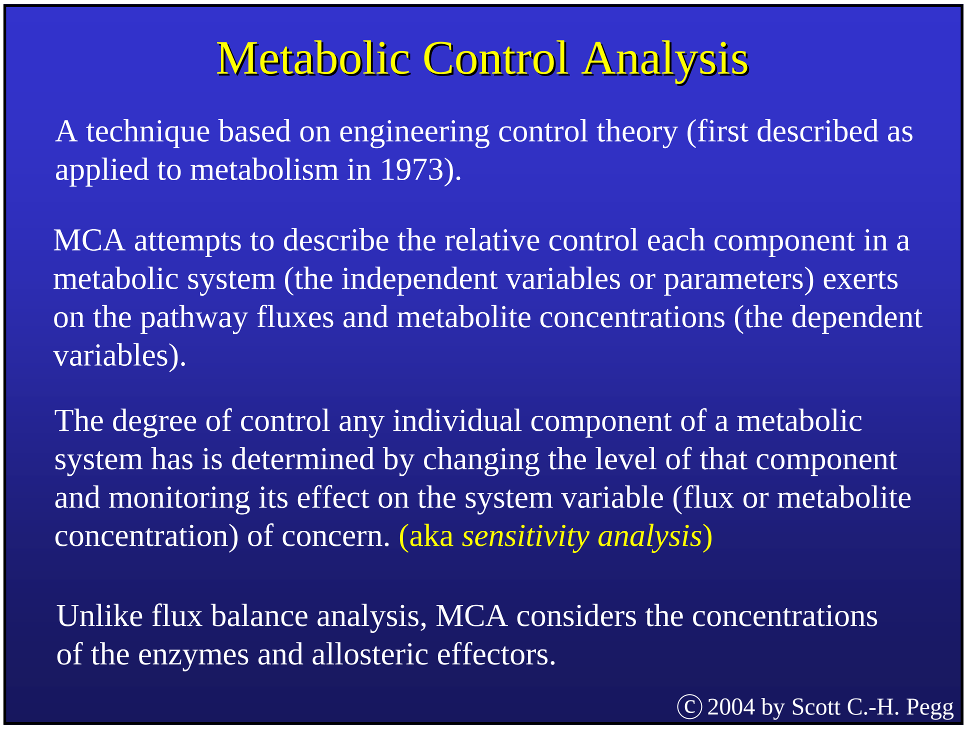
For a real metabolic network (e.g. *E. coli*), the values are roughly,

N = 700, M = 400, and G = 5N.

In this case, *e(C)* = *u(C)* when C = 14.

But these C conditions must be independent of each other!

Segre, D'haeseleer, & Church, "Inference of metabolic network dynamics from flux balance methods and enzyme ratio measurements", ICSB2002



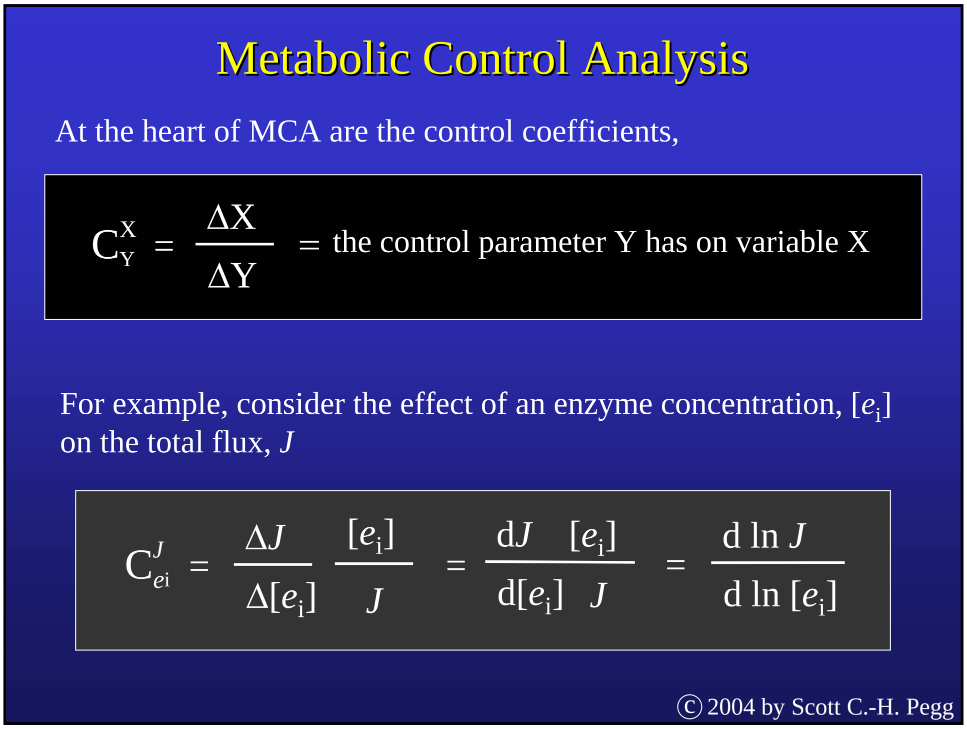
Metabolic Control Analysis

A technique based on engineering control theory (first described as applied to metabolism in 1973).

MCA attempts to describe the relative control each component in a metabolic system (the independent variables or parameters) exerts on the pathway fluxes and metabolite concentrations (the dependent variables).

The degree of control any individual component of a metabolic system has is determined by changing the level of that component and monitoring its effect on the system variable (flux or metabolite concentration) of concern. (aka *sensitivity analysis*)

Unlike flux balance analysis, MCA considers the concentrations of the enzymes and allosteric effectors.



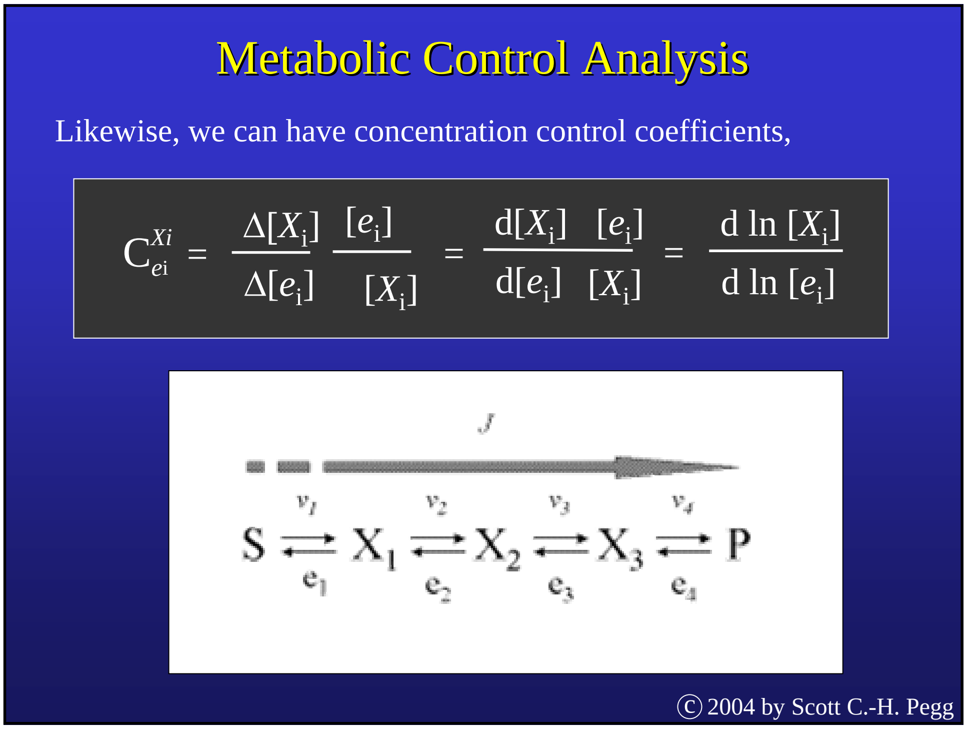
Metabolic Control Analysis

At the heart of MCA are the control coefficients,

$$C^{X}_{Y} = \frac{\Delta X}{\Delta Y} = \text{the control parameter Y has on variable X}$$

For example, consider the effect of an enzyme concentration, $[e_i]$ on the total flux, J

$$C^{J}_{ei} = \frac{\Delta J}{\Delta [e_i]} \frac{[e_i]}{J} = \frac{dJ \quad [e_i]}{d[e_i] \quad J} = \frac{d \ln J}{d \ln [e_i]}$$

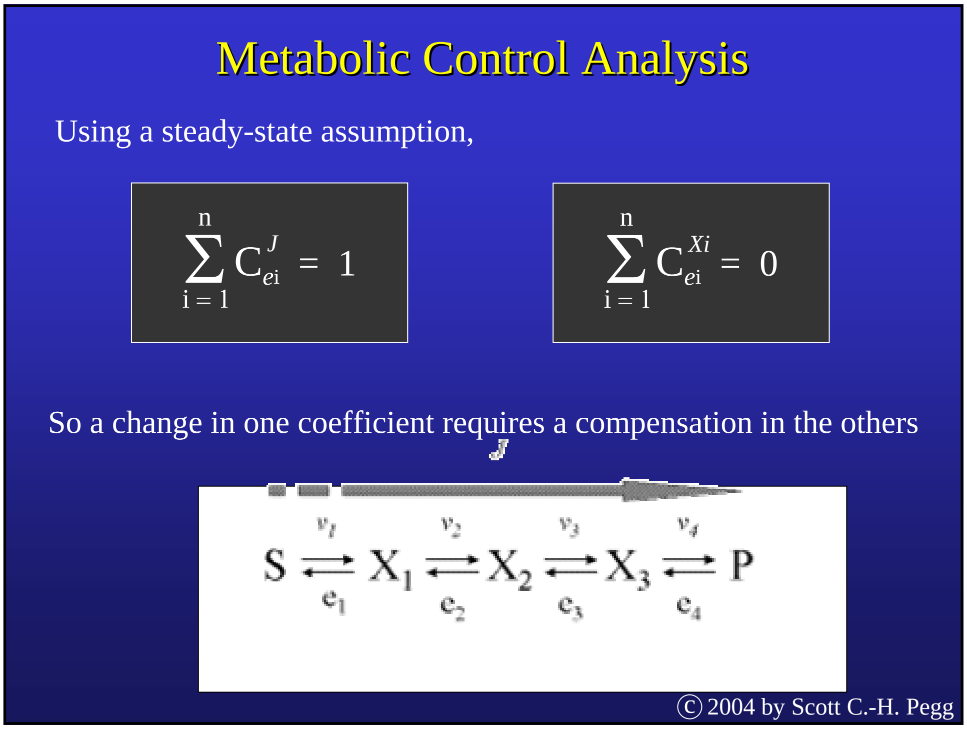


Metabolic Control Analysis

Likewise, we can have concentration control coefficients,

$$C^{Xi}_{ei} = \frac{\Delta[X_i]}{\Delta[e_i]}\,\frac{[e_i]}{[X_i]} = \frac{d[X_i]}{d[e_i]}\,\frac{[e_i]}{[X_i]} = \frac{d\ln[X_i]}{d\ln[e_i]}$$

J
v1
v2
v3
v4
S
X1
X2
X3
P
e1
e2
e3
e4



Metabolic Control Analysis

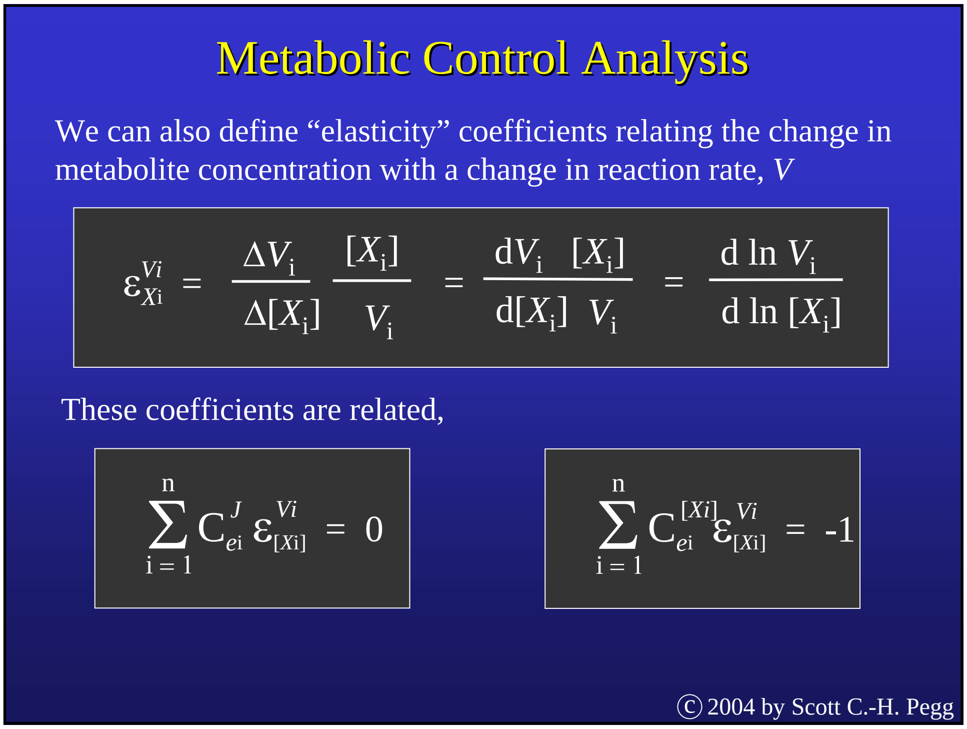
Using a steady-state assumption,

$$\sum_{i=1}^{n} C^{J}_{ei} = 1$$

$$\sum_{i=1}^{n} C^{Xi}_{ei} = 0$$

So a change in one coefficient requires a compensation in the others

J
v1
v2
v3
v4
S
X1
X2
X3
P
e1
e2
e3
e4



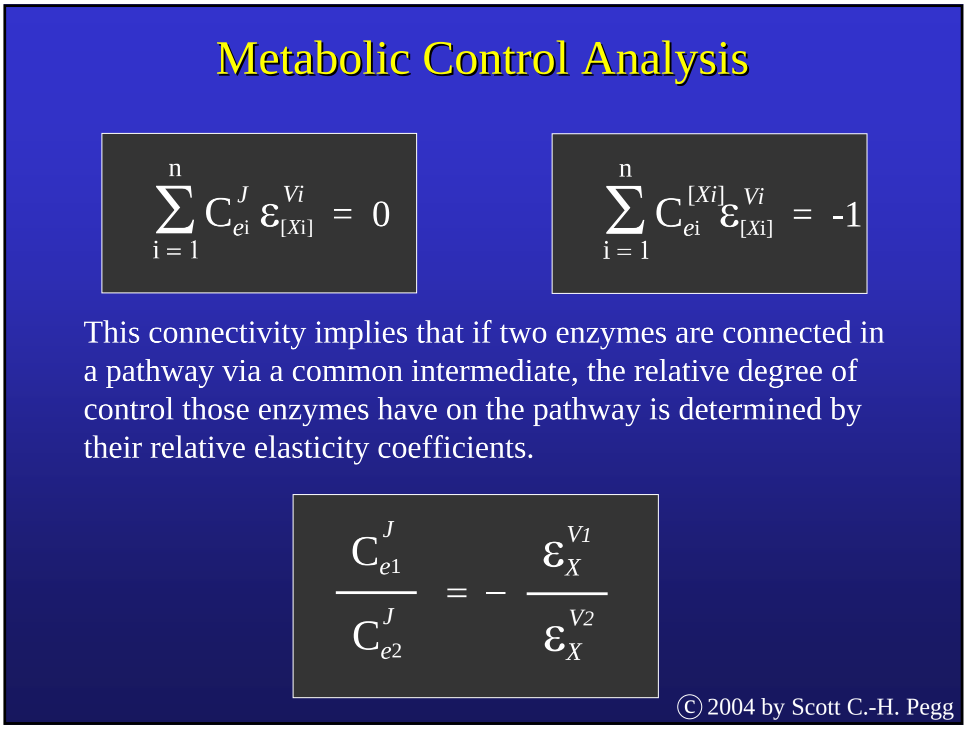
Metabolic Control Analysis

We can also define “elasticity” coefficients relating the change in metabolite concentration with a change in reaction rate, *V*

$$\varepsilon^{Vi}_{Xi} = \frac{\Delta V_\mathrm{i}}{\Delta[X_\mathrm{i}]}\,\frac{[X_\mathrm{i}]}{V_\mathrm{i}} = \frac{\mathrm{d}V_\mathrm{i}\;\;[X_\mathrm{i}]}{\mathrm{d}[X_\mathrm{i}]\;\;V_\mathrm{i}} = \frac{\mathrm{d}\ln V_\mathrm{i}}{\mathrm{d}\ln[X_\mathrm{i}]}$$

These coefficients are related,

$$\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{C}^{J}_{ei}\,\varepsilon^{Vi}_{[Xi]} = 0 \qquad\qquad \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{C}^{[Xi]}_{ei}\,\varepsilon^{Vi}_{[Xi]} = -1$$



Metabolic Control Analysis

$$\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{C}^{J}_{ei}\,\varepsilon^{Vi}_{[Xi]} = 0 \qquad\qquad \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{C}^{[Xi]}_{ei}\,\varepsilon^{Vi}_{[Xi]} = -1$$

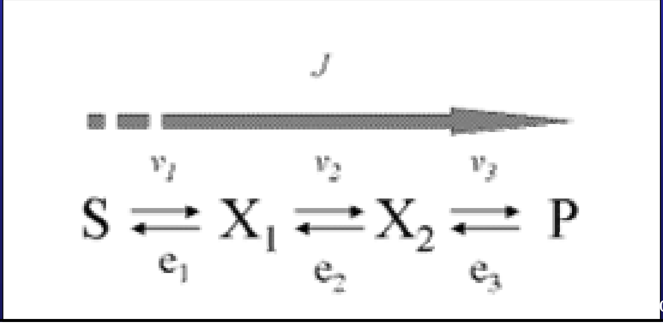
This connectivity implies that if two enzymes are connected in a pathway via a common intermediate, the relative degree of control those enzymes have on the pathway is determined by their relative elasticity coefficients.

$$\frac{\mathrm{C}^{J}_{e1}}{\mathrm{C}^{J}_{e2}} = -\,\frac{\varepsilon^{V1}_{X}}{\varepsilon^{V2}_{X}}$$

Metabolic Control Analysis

Some simple algebra allows us to express the control coefficients in terms of just the elasticity coefficients,

$$C^{J}_{ei} = \frac{\varepsilon^{V2}_{X1}\ \varepsilon^{V3}_{X2}}{\varepsilon^{V1}_{X1}\,\varepsilon^{V3}_{X2} - \varepsilon^{V2}_{X1}\,\varepsilon^{V3}_{X2} - \varepsilon^{V2}_{X2}\,\varepsilon^{V1}_{X1}}$$



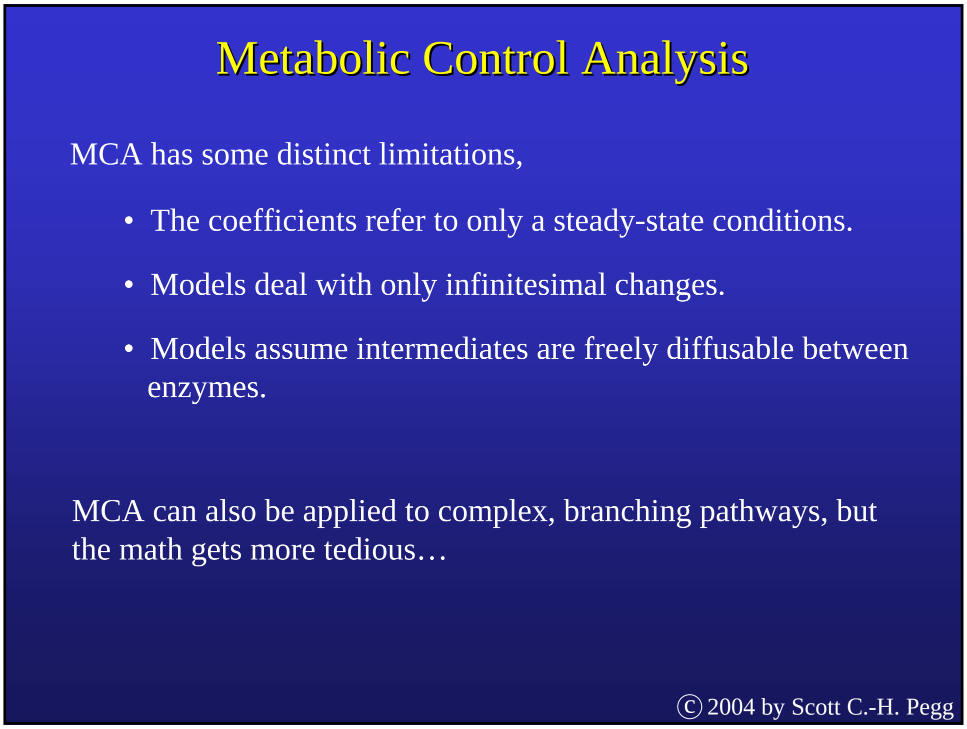
Metabolic Control Analysis

We can also define “response” coefficients to model the effects of an external influence, *A*

$$R^{J}_{A} = C^{J}_{ei}\,\varepsilon^{Vi}_{[A]}$$

Or, if the external influence effects more than one enzyme,

$$R^{J}_{A} = \sum_{i=1}^{n} C^{J}_{ei}\,\varepsilon^{Vi}_{[A]}$$

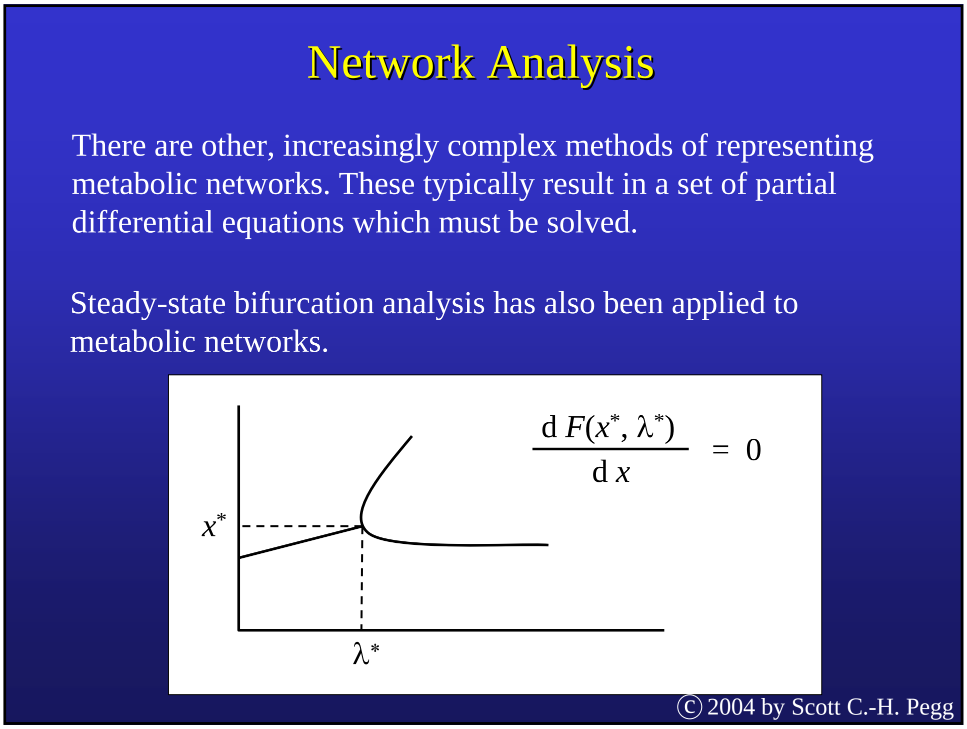


Metabolic Control Analysis

MCA has some distinct limitations,

- The coefficients refer to only a steady-state conditions.
- Models deal with only infinitesimal changes.
- Models assume intermediates are freely diffusable between enzymes.

MCA can also be applied to complex, branching pathways, but the math gets more tedious…

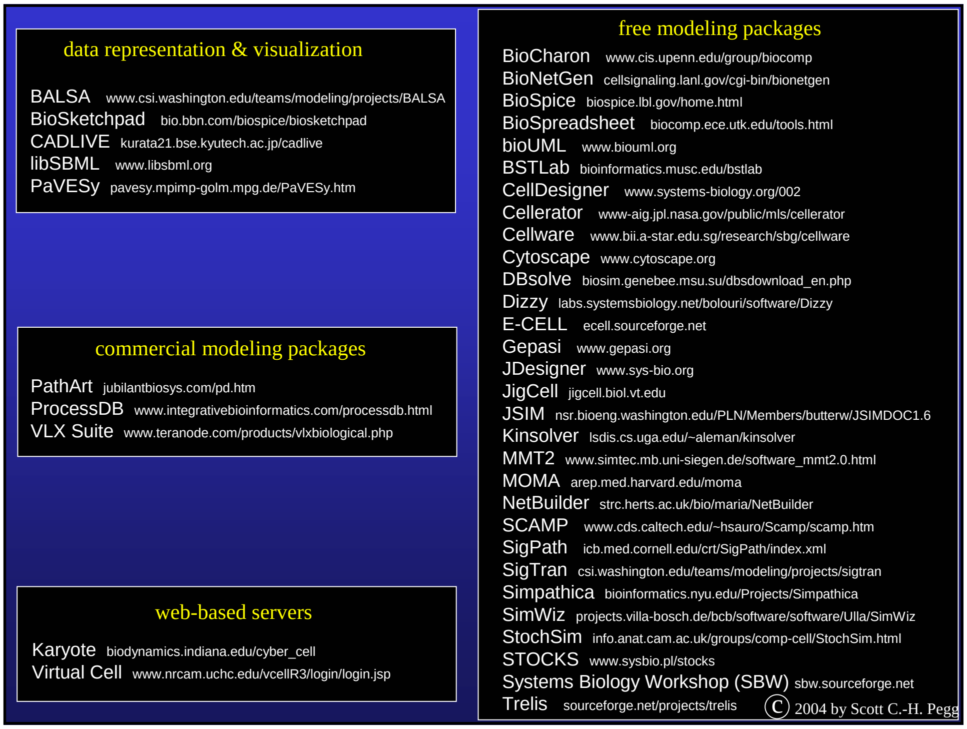


Network Analysis

There are other, increasingly complex methods of representing metabolic networks. These typically result in a set of partial differential equations which must be solved.

Steady-state bifurcation analysis has also been applied to metabolic networks.

$$\frac{\mathrm{d}\,F(x^*, \lambda^*)}{\mathrm{d}\,x} = 0$$



data representation & visualization

BALSA www.csi.washington.edu/teams/modeling/projects/BALSA
BioSketchpad bio.bbn.com/biospice/biosketchpad
CADLIVE kurata21.bse.kyutech.ac.jp/cadlive
libSBML www.libsbml.org
PaVESy pavesy.mpimp-golm.mpg.de/PaVESy.htm

commercial modeling packages

PathArt jubilantbiosys.com/pd.htm
ProcessDB www.integrativebioinformatics.com/processdb.html
VLX Suite www.teranode.com/products/vlxbiological.php

web-based servers

Karyote biodynamics.indiana.edu/cyber_cell
Virtual Cell www.nrcam.uchc.edu/vcellR3/login/login.jsp

free modeling packages

BioCharon www.cis.upenn.edu/group/biocomp
BioNetGen cellsignaling.lanl.gov/cgi-bin/bionetgen
BioSpice biospice.lbl.gov/home.html
BioSpreadsheet biocomp.ece.utk.edu/tools.html
bioUML www.biouml.org
BSTLab bioinformatics.musc.edu/bstlab
CellDesigner www.systems-biology.org/002
Cellerator www-aig.jpl.nasa.gov/public/mls/cellerator
Cellware www.bii.a-star.edu.sg/research/sbg/cellware
Cytoscape www.cytoscape.org
DBsolve biosim.genebee.msu.su/dbsdownload_en.php
Dizzy labs.systemsbiology.net/bolouri/software/Dizzy
E-CELL ecell.sourceforge.net
Gepasi www.gepasi.org
JDesigner www.sys-bio.org
JigCell jigcell.biol.vt.edu
JSIM nsr.bioeng.washington.edu/PLN/Members/butterw/JSIMDOC1.6
Kinsolver lsdis.cs.uga.edu/~aleman/kinsolver
MMT2 www.simtec.mb.uni-siegen.de/software_mmt2.0.html
MOMA arep.med.harvard.edu/moma
NetBuilder strc.herts.ac.uk/bio/maria/NetBuilder
SCAMP www.cds.caltech.edu/~hsauro/Scamp/scamp.htm
SigPath icb.med.cornell.edu/crt/SigPath/index.xml
SigTran csi.washington.edu/teams/modeling/projects/sigtran
Simpathica bioinformatics.nyu.edu/Projects/Simpathica
SimWiz projects.villa-bosch.de/bcb/software/software/Ulla/SimWiz
StochSim info.anat.cam.ac.uk/groups/comp-cell/StochSim.html
STOCKS www.sysbio.pl/stocks
Systems Biology Workshop (SBW) sbw.sourceforge.net
Trelis sourceforge.net/projects/trelis