Computational Modeling in Biology

Lecture 7

‘I want to understand everything,’ said Miro. ‘I want to know everything and put it all together to see what it all means.’

‘Excellent project,’ she said. ‘It will look very good on your resume.’

- Card (1982)

from Haefner, Modeling Biological Systems

Lectures 7 and 8

- General modeling principles
  - Models, systems, and classifications
  - Two approaches to modeling
  - Occam's Razor
  - Model Objectives
  - Model formulation
  - Parameter Estimation
  - Model Validation and Discrimination
  - Model Analysis
- A gene regulation modeling example
  - Endo16 cis-region quantitative relationships
- Combinatorial Gene Network modeling
  - Static Graph
  - Weight matrix
  - Boolean Networks
  - Bayesian Networks
  - Finite State Model
- A gene network inference example (from microarray data)

Systems and Models

- System: a collection of interrelated objects
- Model: a description of a system
- Models are abstract, and conceptually simple
- Three primary technical uses of models in science:
  - Understanding of a system
  - Prediction of an unknown state of a system
  - Control a system to produce a desirable outcome
- Secondary uses of models:
  - Conceptual framework for organizing/coordinating empirical research
  - A summary mechanism
  - Identify areas of ignorance
  - Insight

Classification of Models

- Forms of models
  - Conceptual or verbal
  - Diagrammatic
  - Physical (tinker toys)
  - Formal (Mathematical)
- Mathematical Classification
  - Mechanistic vs. descriptive
  - Dynamic vs. static
  - Continuous vs. discrete
  - Stochastic vs. deterministic
  - Spatially homogeneous vs. spatially heterogeneous
  - Analytical vs. numerical

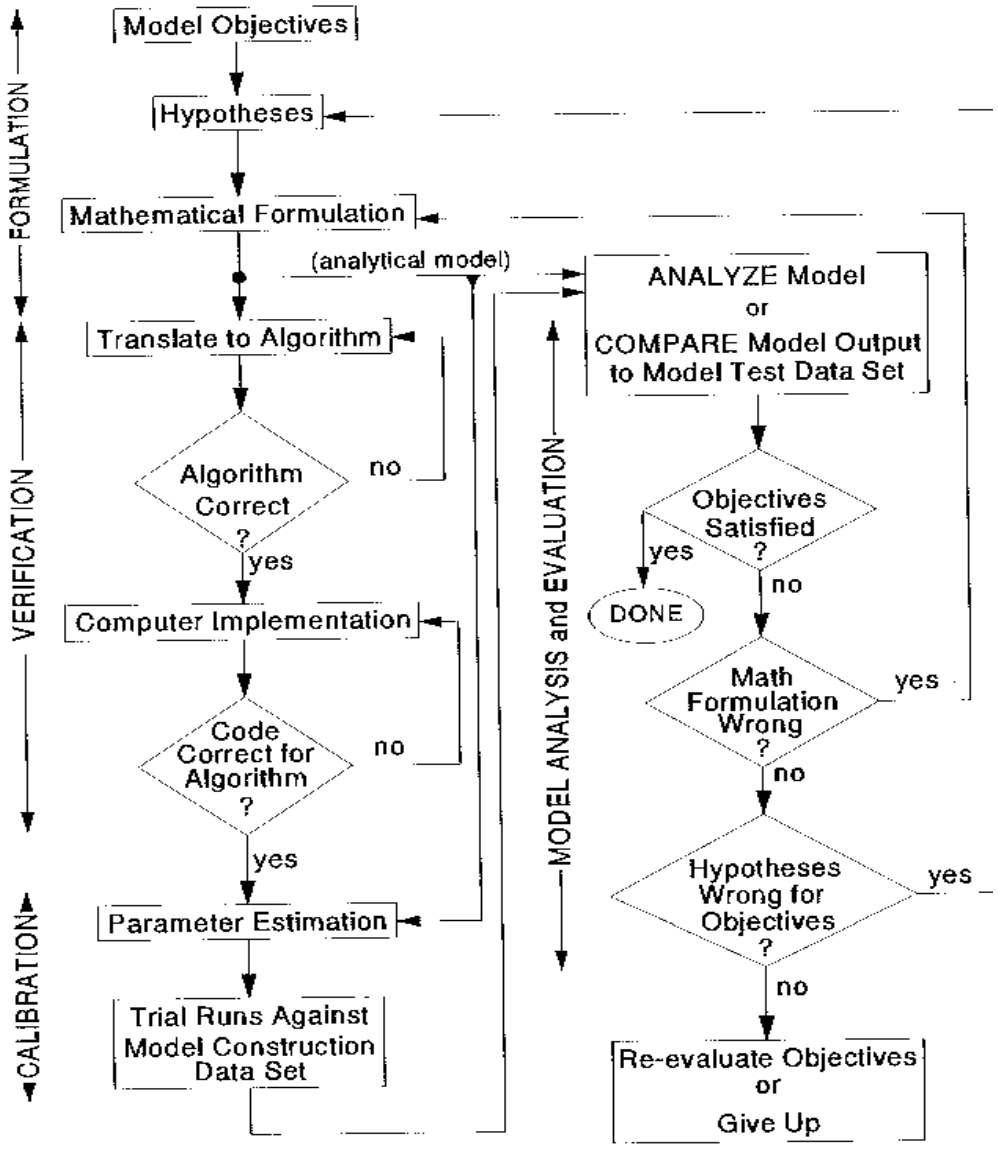
Modeling

- Classical Approach

Eating an elephant:

Q: Where do you start?

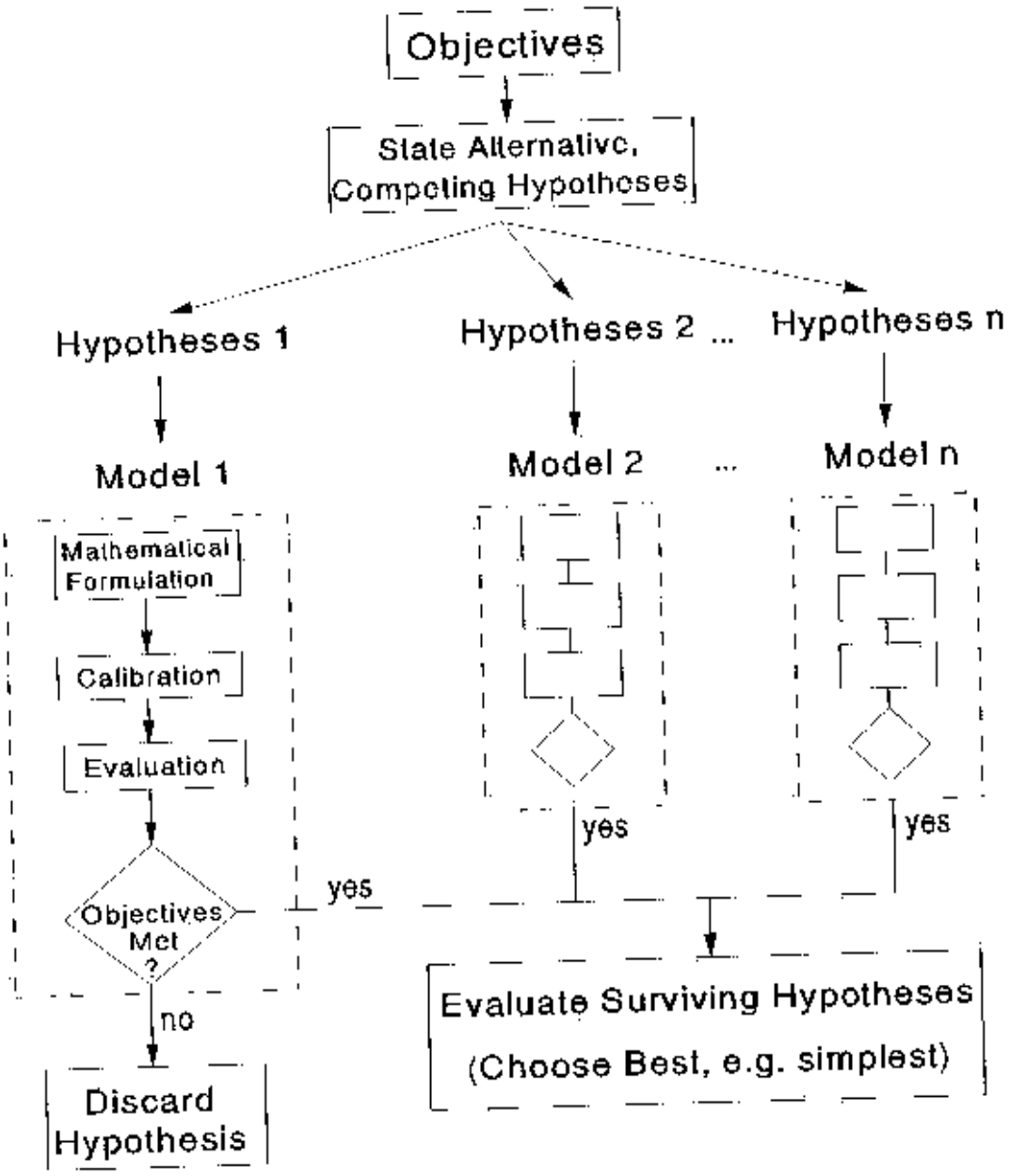
A: Wherever you start, eat it one bite at a time



Alternative Models

- Problems with the classical approach: new data must be used for the validation of every pass through the process
- Solution: use multiple hypotheses and models in parallel
- Which model(s) is(are) the best? Use a method to discriminate between models

Multiple Models Approach



Modeling vs. Function Fitting

- Insight
- One can fit a polynomial of degree n through any n-1 points, but that does not offer any better understanding of the phenomena modeled
- Even if the model doesn't fit perfectly, if it is simple it will be accepted, even if only as an estimator
- Eg. Linear regression as an estimator of a relationship between two samples from different populations

Choosing the Best Model

Parsimony Argument (Occam's Razor)

All things being equal, the "smallest" model that explains the observations and fits the objectives should be accepted

In reality, the smallest means the model which optimizes a certain scoring function (e.g. least nodes, most robust, least assumptions, etc.)

Model Objectives

- Eg. Construct a gene network model that:
  - Describes known genes interactions well
  - Predicts interactions not known so far
  - Allows for perturbations in the description
  - Defines the parameters (temporal and spatial scales) over which the model is described
  - Can be clearly validated (empirically and theoratically)

Qualitative and Quantitative Model Formulation

1. Qualitative: Understanding the model abstractly

- Elements of model formulation
  - Objects (state variables)
  - Material flow
  - Information flow
  - Sources and sinks
  - Parameters (constants)
  - Driving and Aux. Variables

- Principles of Qualitative Formulation
  - What are the question that are being answered? (objectives)
  - What quantities are needed to answer the questions?
  - What equations answer them?
  - Etc.

- Model simplifications
  - Minimize state variables
    - Convert a variable into a constant
    - Aggregate state variables
  - Make stronger assumptions
    - Convert functions of state variables into constants
    - Convert non-linear relationships into linear
  - Remove temporal complexity
    - Convert random models into deterministic models
    - Convert driving variables to constants
  - Remove spatial complexity

2. Quantitative model formulation

- This is the only exact formulation of relationships among objects in a system
- Mathematical formalisms:
  - Discrete, Difference Equations
  - Continuous, Differential Equations
- Analytical Solution:
  - Closed functional form of relationship(s) among state variables

What If the Equations Cannot be Solved: Simulation and Numerical Techniques

- Divide time in small pieces and assume discrete model
- Simulations provide valuable insight in the behavior of the equations describing the system
- Many available simulating environments (languages)
- Numerical techniques used to find solutions to difficult ODEs and PDEs (Euler, Runge-Kutta).

Parameter Estimation (Optimization problems)

- Every quantitative formulation will have some data dependent constants that will be determined when the model is "fitted" to the data.
- Fitting here is understood as minimizing some error function between the model prediction and the observed data
- Example: the statistical model of additive and multiplicative effects on microarray data

Optimization and Search

- Methods:
  - Linear regression (least squares)
    - Fitting a linear equation (of the state variables), and determining the coefficients
  - Non-linear methods (iterative)
    - Gradient methods (steepest descent)
      - Finds a local optimum
    - Simplex method
    - Stochastic Methods can find an optimum in the presence of local optima:
      - Simulated Annealing
      - Tabu Search
      - Genetic Algorithms

Graph Models

Combinatorial Optimization

- Discrete Models
- Parsimony arguments

Model Validation

Basics of Scientific Falsification: modus tollens

$A \Rightarrow B$
$\neg B$
_____
$\neg A$

Eg.
Whenever A goes up gene B goes up too.
Gene B is down,
Thus A is down too..

~~$A \Rightarrow B$~~
~~B~~
~~_____~~
~~A~~

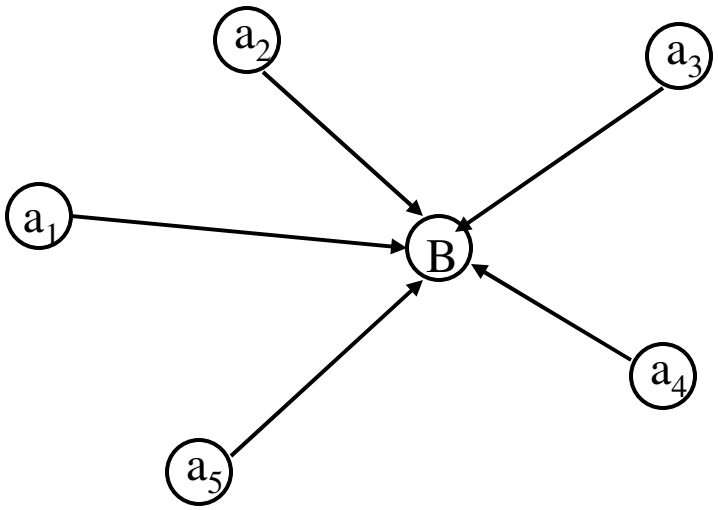
Eg.
Whenever A goes up gene B goes up too.
Gene B is up,
Thus A is up too.

A is almost always a conjunction of conditions in reality:

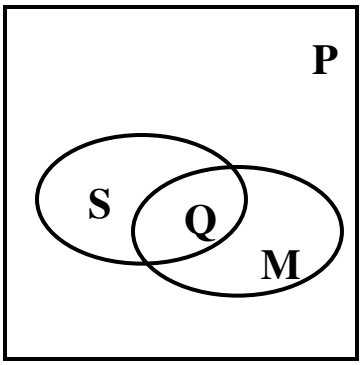
$a_1 \wedge a_2 \wedge a_3 \ldots \wedge a_n \Rightarrow B$
$\neg B$
_____
$\neg(a_1 \wedge a_2 \wedge a_3 \ldots \wedge a_n)$

But in reality we cannot tell which ai's caused, $\neg a_1 \vee \neg a_2 \vee \neg a_3 \ldots \vee \neg a_n$ to be false.

<u>This is especially true in network modeling, where to find which of the a_i's is false additional experiments need be performed.</u>

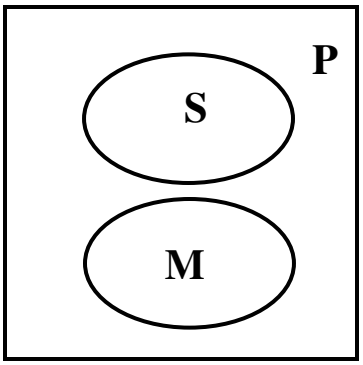


Model Reliability and Adequacy

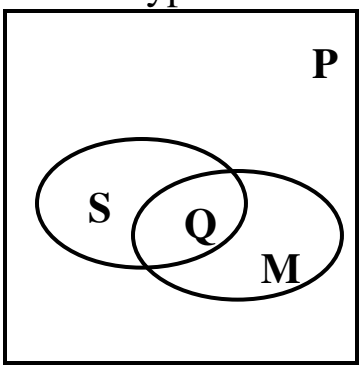


P is the set of all possible observations
S set of all observations made on the study system
M is the set of all model outputs
Q=S∩M

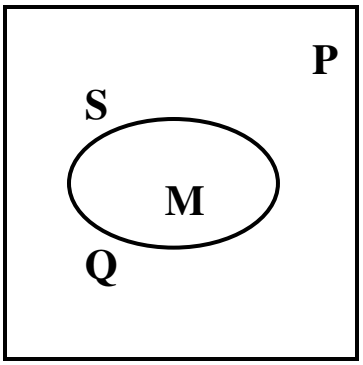
Useless model

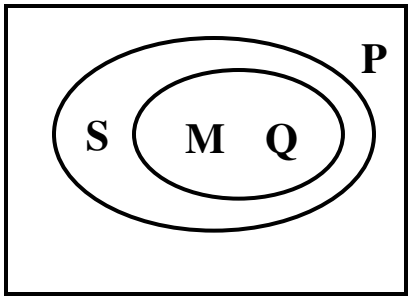


Typical

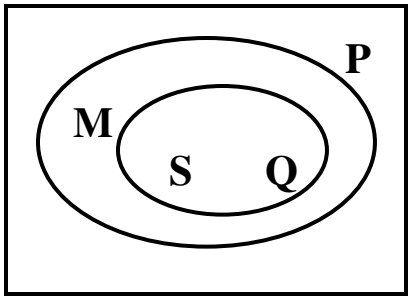


Dream situation





Incomplete model



Complete, but erring model

Model reliability: |Q|/|M|
Model adequacy: |Q|/|S|
Increasing difficulty of validation: Q, Adequacy, Reliability

Validating Dynamical Systems

- Data Independence
  - Separate data for model validation from data used for generating hypotheses and estimating parameters
  - This way a circular argument is avoided; otherwise calibration is performed and not validation
- Modeling single or multiple responses
  - System-wide vs. single variable validation
- Unreplicated Models (no variability)
  - Turing Test (80% of the experts, 80% of the time)
  - Observed vs. predicted regression
  - Indices
  - Goodness of fit
- Replicated Models (variability)
  - Single Value (t-test, ANOVA)
  - Time-Series (ANOVA, corrected for autocorrelation)

Model Discrimination

Choosing "the best" model

- Likelihood functions: which model is more likely to be the best based on the data fit?
- Modeling models: knowing more about the system means choosing the better model with higher probability
- Bayesian Inference: calculating the probability of being correct

Model Analysis

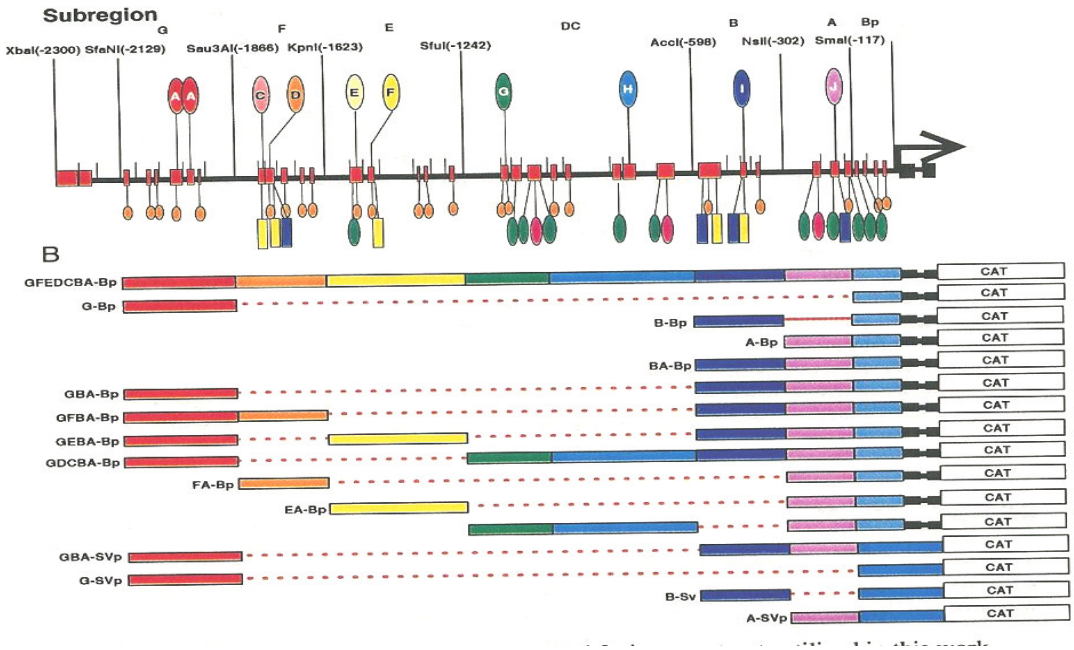
- Finding out general properties of the models by actively manipulating model components
  - Uncertainty analysis
  - Parameter sensitivity
    - Single
    - Multiple
  - Error Analysis
  - Analysis of Model Behavior
    - Equilibria and Nullclines (behavior near equilibria)
  - Stability to Perturbations

A Gene Regulation Modeling Example

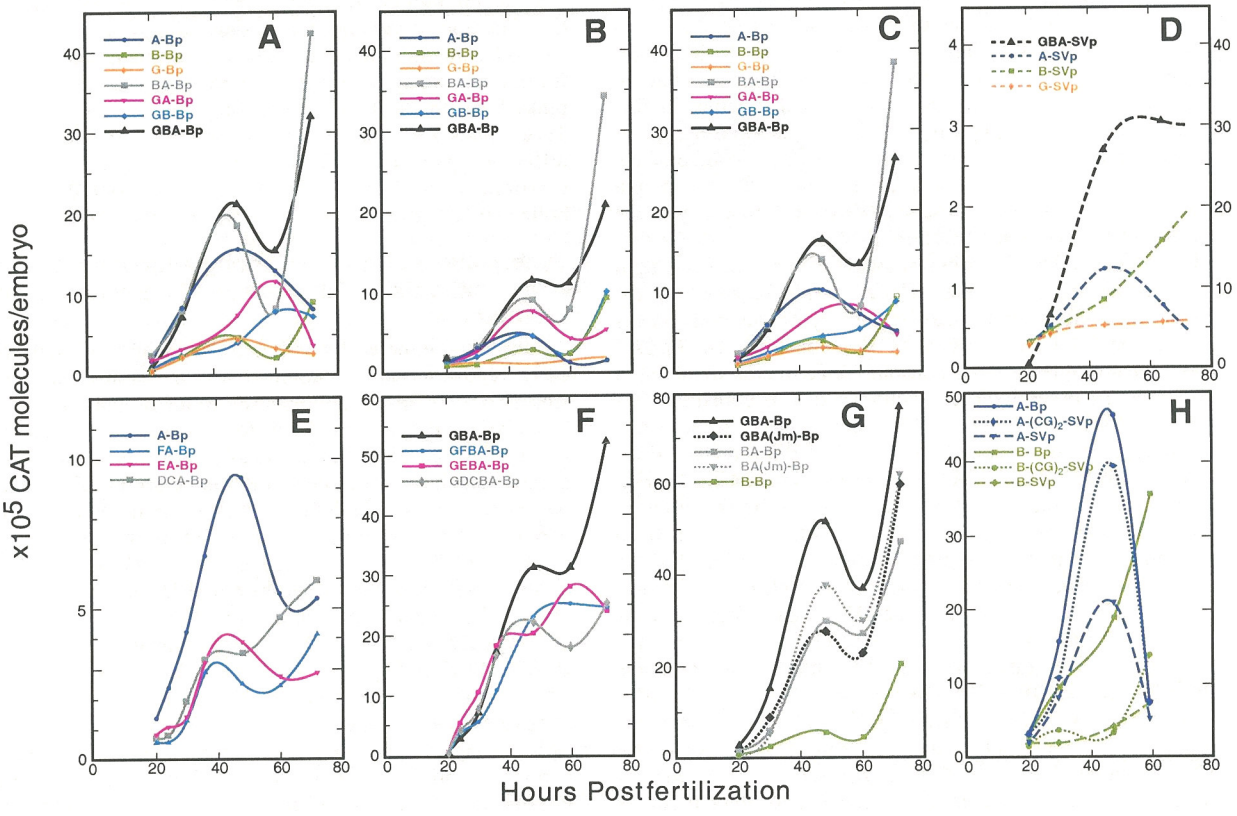
"Quantitative functional interrelations within the cis-regulatory system of the S. purpuratus Endo16 gene"

Yuh et al, 1996

Experiment Setup: Fishing Expedition



Observations: Expression profiles have local similarities



Modeling Ideas

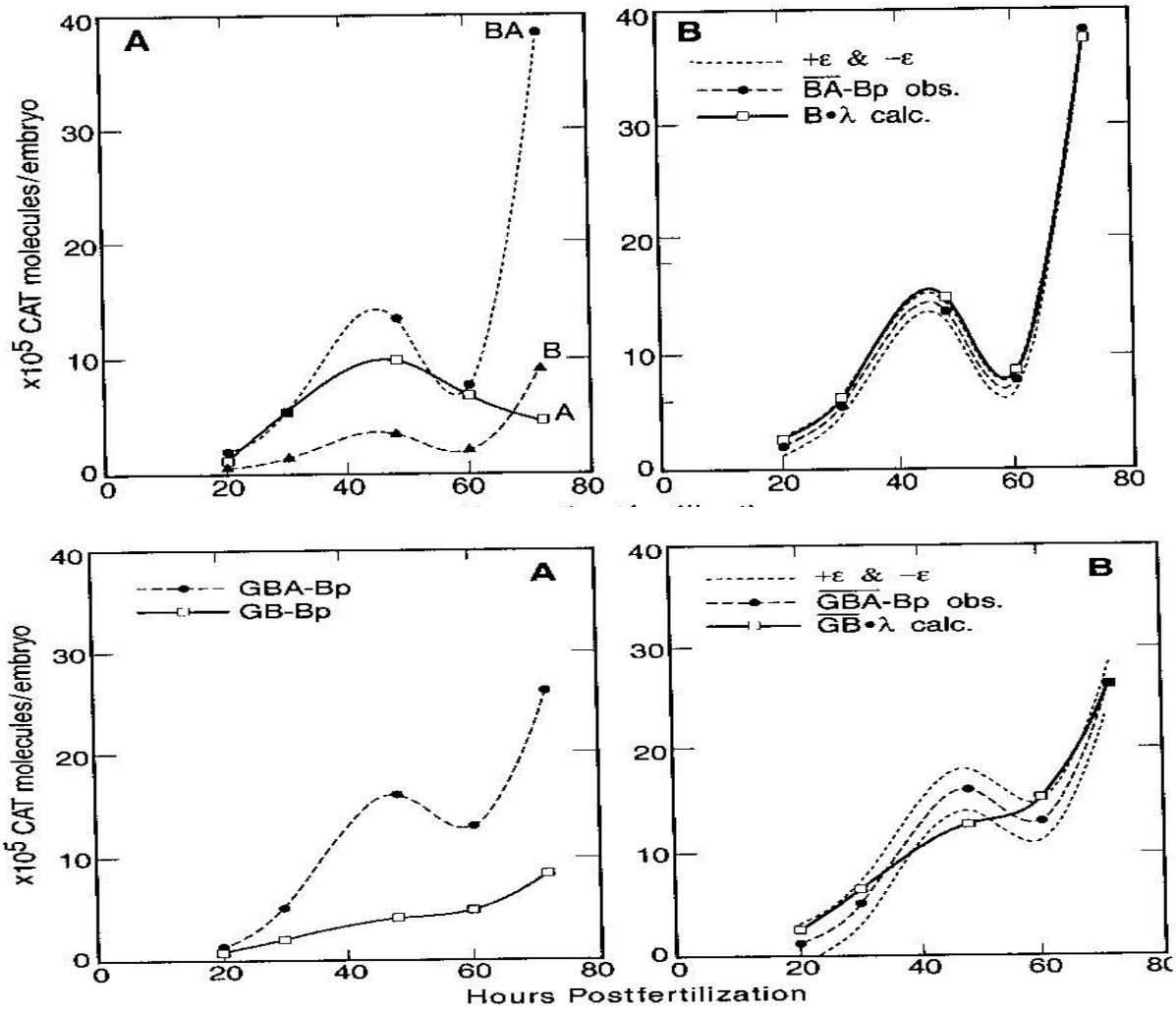
- Cis-region modules contribute uniquely to the gene expression
- Activators and inhibitor modules
- The contribution of the individual activator modules is additive
- Model Formulation: expression of region AB is the sum of the expressions of regions A and B

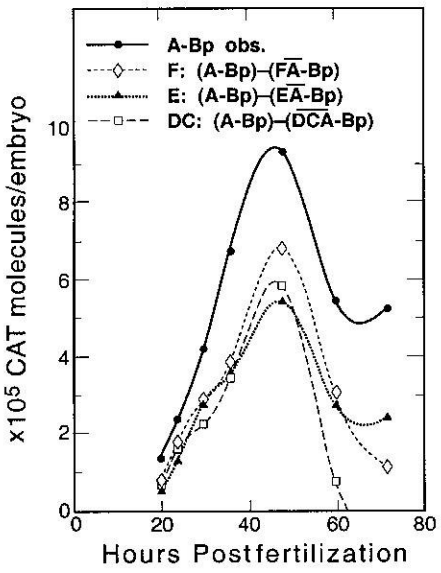
  $AB = A+B$
- Model Validation: least squares fit (linear regression)

Table 1. Synergistic models for modules G, B and A linked to endogenous basal promoter

Model†	ε (% max)*	λ‡	λ/function#
$\overline{BA}$=B•λ	0.227 (2%)	4.2	4.2
$\overline{BA}$=(B+A)•λ	9.07 (24%)	1.6	1.6
$\overline{BA}$=A•λ	6.49 (17%)	0.69	0.83
$\overline{GBA}$=$\overline{GB}$ •λ	1.99 (8%)	3.1	3.1
$\overline{GBA}$=$\overline{BA}$ •λ	4.35 (17%)	0.78	0.78
$\overline{GBA}$=$\overline{BA}$•G•λ	3.58 (14%)	0.39	0.62
$\overline{GBA}$=A•B•G•λ	4.65 (18%)	0.26	0.64
$\overline{GBA}$=$\overline{GB}$•A•λ	3.97 (15%)	0.50	0.70
$\overline{GBA}$=(G+B+A)•λ	4.40 (17%)	1.23	1.23
$\overline{GBA}$=B•A•λ	3.09 (12%)	0.59	0.77
$\overline{GBA}$=$\overline{GBA}$ (J_m)•λ	7.0 (9%)	1.42	1.42

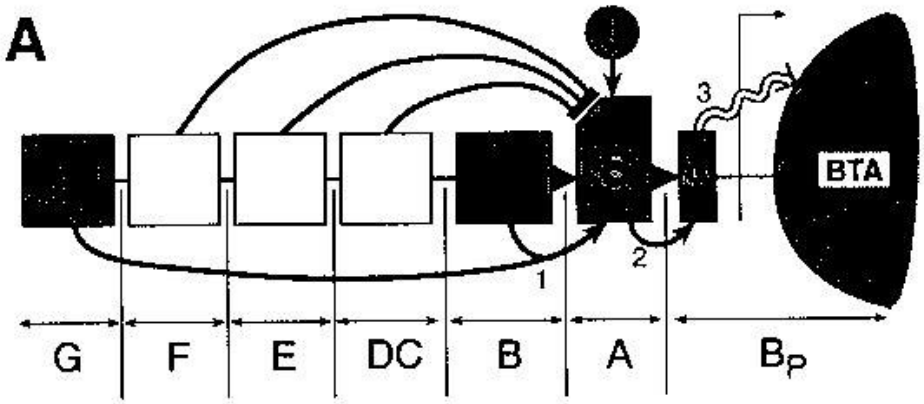
†Overline indicates physical linkage of modules in expression construct.





Similar experiments done for the inhibition modules

Putting it all together



What Is a Gene Network?

Gene Regulatory Systems

"Programs built into the DNA of every animal."
Eric H. Davidson

- *Cis regulatory elements*: DNA sequence (specific sites)
  - promoters;
  - enhancers;
  - silencers;
- *Trans regulatory factors*: products of regulatory genes
  - generalized
  - specific (Zinc finger, leucine zipper, etc.)

Known properties of real gene regulatory systems:

- cis-trans specificity
- small number of trans factors to a cis element: 8-10
- cis elements are programs
- regulation is event driven (asynchronous)
- regulation systems are noisy environments
- Protein-DNA and protein-protein regulation
- regulation changes with time

Gene Regulatory Networks

Gene Networks: models of measurable properties of Gene Regulatory Systems.

Gene networks model functional elements of a Gene Regulation System together with the regulatory relationships among them in a computational formalism.

Types of relationships: causal, binding specificity, protein-DNA binding, protein-protein binding, etc.

Existing Formalisms

Combinatorial	**Physical**
• Static Graph Models • Boolean Networks • Weight Matrix (Linear) Models • Bayesian Networks	• Stochastic Models • Difference / Differential Equation Models • Chemical/Physical Models • Concurrency models

Combinatorial Formalisms

Gene regulation networks are modeled as *graphs.*

The general syntax is:

- **Nodes**: functional units (genes, proteins, metabolites, etc.);
- **Edges**: dependencies;
- **Node states**: measurable (observable) properties of the functional units, can be discrete or continuous, deterministic or stochastic;

– **Graph Annotation**: a,i,+,-,w

– **Gates**: nodes with an associated function, its input and the resulting output;

– **Topology:** wiring, can be fixed or time-dependent;

– **Dynamics:** (i.e. static,dynamic);

– **Synchrony:** synchronous, asynchronous;

– **Flow:** Quantity that is conveyed (flows) through the edges in a dynamic network

Graph-based Models

	Nodes	Edges	Labels:		Flow
			Nodes	Edges	
Static Graph Models	Genes	Dependency	On/Off	Nature of dependency (i.e.Strength)	None, Static
Boolean Networks	Genes	Dependency	Boolean Function of Inputs, Discrete	Information 0/1	Information
Weight Matrix (Linear)	Genes	Activation / Inhibition	Linear combination of inputs, continuous	Weight	Normalized node states
Bayesian Networks	Random Variables	Stochastic Dependence /Independence	Stochastic	Conditional Probabilities	None, Static

Static Graph Models

Network: directed graph G=(V,E), where V is set of vertices, and E set of edges on V.

The nodes represent genes and an edge between v_i and v_j symbolizes a dependence between v_i and v_j.

The dependencies can be *temporal* (causal relationship) or *spatial* (cis-trans specificity).

The graph can be annotated so as to reflect the nature of the dependencies (e.g. promoter, inhibitor), or their strength.

Properties:

- Fixed Topology (doesn't change with time)
- Static
- Node States: Deterministic

Boolean Networks

Boolean network: a graph G(V,E), annotated with a set of states $X=\{x_i \mid i=1,\ldots,n\}$, together with a set of Boolean functions $B=\{b_i \mid i=1,\ldots,n\}$, $b_i : \{0,1\}^k \rightarrow \{0,1\}$ **.**

Gate: Each node, v_i, has associated to it a function , with inputs the states of the nodes connected to v_i.

Dynamics: The state of node v_i at time t is denoted as $x_i(t)$. Then, the state of that node at time t+1 is given by:

$$x_i(t+1) = b_i(x_{i1}, x_{i2}, \ldots, x_{ik})$$

where x_{ij} are the states of the nodes connected to v_i.

General Properties of BN:

- Fixed Topology (doesn't change with time)
- Dynamic
- Synchronous
- Node States: Deterministic, discrete (binary)
- Gate Function: Boolean
- Flow: Information

Exhibit synergetic behavior:

- redundancy
- stability (attractor states)

BN and Biology

Microarrays quantify transcription on a large scale.

The idea is to infer a regulation network based solely on transcription data.

Discretized gene expressions can be used as descriptors of the states of a BN. The wiring and the Boolean functions are reverse engineered from the microarray data.

BN and Biology, Cont'd.

From mRNA measures to a Regulation Network:

1 *Continuous gene expression values are discretized as being 0 or 1 (on, off), (each microarray is a binary vector of the states of the genes);*

2 *Successive measurements (arrays) represent successive states of the network i.e. X(t)->X(t+1)->X(t+2)…*

3 *A BN is reverse engineered from the input/output pairs: (X(t),X(t+1)), (X(t+1),X(t+2)), etc.*

Weight Matrix (Linear) Models

The network is an annotated graph G(V,E): each edge ($v_i v_j$) has associated to it a weight w_{ij}, indicating the "strength" of the relationship between v_i and v_j.

$W=(w_{ij})_{nxn}$ is referred to as the *weight matrix*.

Dynamics: The state of node v_i at time t is denoted as $x_i(t)$.

$$x_i(t+1) = f_i\left(\sum_{j=1}^{n} w_{ij} x_j(t)\right)$$

where the next state of a node is a linear combination of all other nodes' states.

Properties of Weight Matrix Models

- Fixed Topology (doesn't change with time)
- Dynamic
- Synchronous
- Node States: Deterministic, continuous
- Gate: linear combinations of inputs
- Flow: Normalized node states

Weight Matrix Models and Biology

Used to model transcriptional regulation.

Gene expression (microarray) data is reverse engineered to obtain the weight matrix W, as in the Boolean networks.

The number of available experiments is smaller than the number of genes modeled, so genes are grouped in similarity classes to lower the under constrained-ness.

These models have been used on existing data to obtain good results.

Bayesian Networks

Bayesian Network: An annotated directed acyclic graph G(V,E), where the nodes are random variables X_i, together with a conditional distribution $P(X_i \mid \text{ancestors}(X_i))$ defined for each X_i.

A Bayesian network uniquely specifies a joint distribution:

$$p(X) = \prod_{i=1}^{n} p(X_i \mid \text{ancestors}(X_i))$$

Various Bayesian networks can describe a given set of Random variables' values. The one with the highest score is chosen.

General Properties

- Fixed Topology (doesn't change with time)
- Node States: Stochastic
- Flow: Conditional Probabilities

Model Comparison

How do we compare all these formalisms?

- Biologically
    - descriptive models that capture reality well
    - predictive models useful to a biologist
- Combinatorially
    - ease of analysis
    - utilization of existing tools
    - syntax and semantics

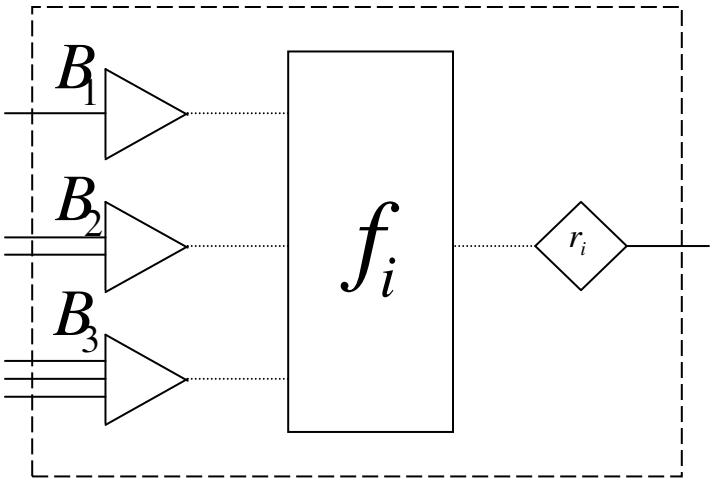
Towards a Consensus Model

Desired properties of a general descriptive model:

- combinatorial model
- asynchronous
- capturing the complex cis element information processing
- deterministic states representing measurable quantities
- stable (i.e. resistant to small perturbations of states)
- describing the flow of both information and concentration

Reverse Engineering must be possible!

Finite-State Linear Model



- **input**: states of binding sites (attached/detached)
- **function**: Boolean of the binding states
- **output**: rate of production of a substance

Dynamics: event driven, asynchronous. The production rate changes only if the Boolean combination of the binding sites' states is T.

Flow: both information(dashed lines) and concentration.(full lines).

Brazma, 2000

This model describes well both continuous (concentration) and discrete (information, binding sites' states) behavior.

Further improvements:

- capture concentrations of both mRNA and proteins.
- introduce noisy gates

Finite State Linear Model
With Von Neumann Error

Network Inference and Existing Data

A general interdisciplinary modeling strategy:

1. Network model of a Regulatory system
2. "Hardwire" existing data/literature into model
3. Infer new relationships from such model
4. Perform experiments to validate new relationships
5. Extend model if necessary, go back to second step.

Network Inference Example

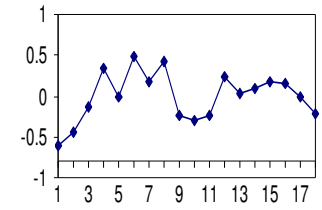
Chen et al, 1999

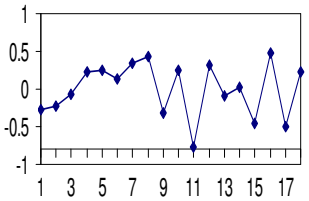
A Simple Static Graph Model From Microarray Data

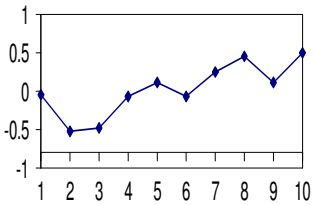
- Motivation
  - Time-series data of gene expressions in yeast
  - Is it possible to elucidate regulatory relationships for the up/down patterns in the curves?
  - Could one select a gene network from many candidates, based on a parsimony argument?
- Grand Model:
  - Graphs with nodes = genes
  - Edges labeled A, I, N, determined from the data
  - The graph is a putative regulatory network, and has too many edges
  - Since the model over-fits the data, there is a need for additional assumptions
  - Parsimony argument: few regulators

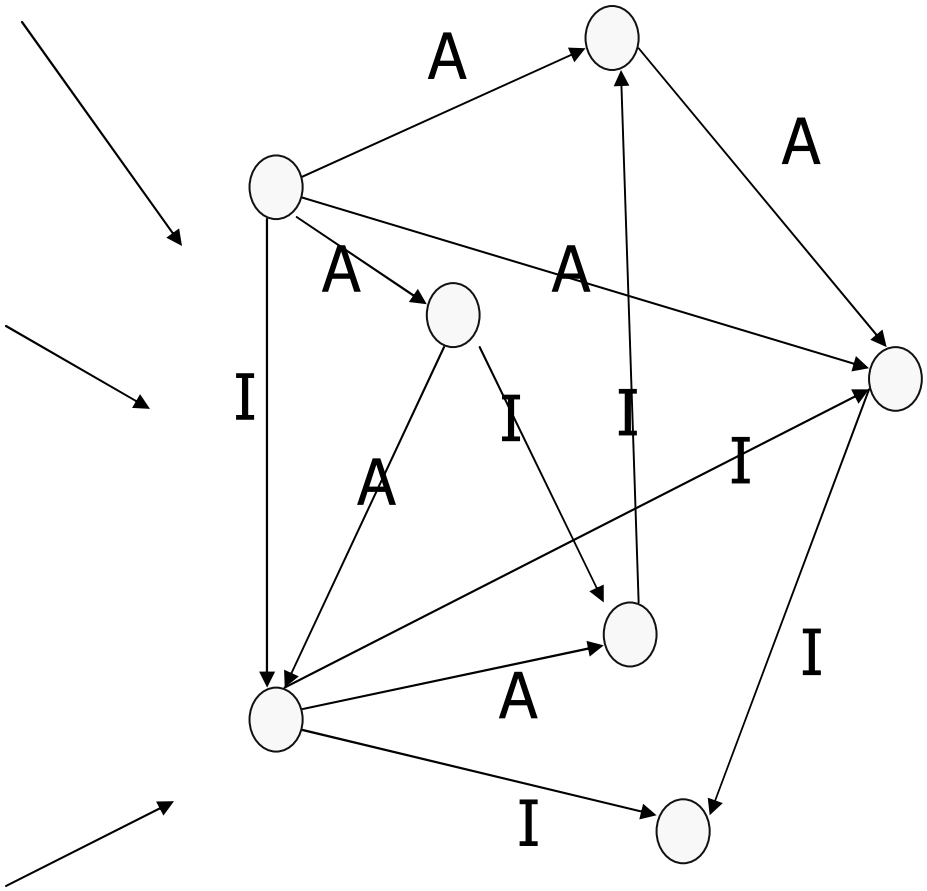
Raw data: putative regulation

Reduce the time-series data set to a graph G_{ai} where each edge is labeled A, I.









1. Filter
2. Cluster
3. Curve Smoothing

Optimizing the Graph

Goal: Given a directed graph G_{ai} with edges labeled A or I and weighted, output a labeling of vertices which optimizes:

$$f(G_{ai}) = \sum_{v_i \in V(G_{ai})} max(v_i[I] \cdot v_i[A]) - C(count(A) + count(I))$$

General optimization technique (Sim. Ann.)

Bibliography:

Eric H. Davidson *Genomic Regulatory Systems*, Academic Press, 2001

John von Neumann. *Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components*. Automata Studies. Princeton University Press, 1956. pp. 43--98.

Alvis Brazma and Thomas Schlitt *Reverse Engineering of Gene Regulatory Networks: a Finite State Linear Model*, BITS 2000, Heidelberg, Germany

Dana Pe'er et al. *Inferring Subnetworks from Perturbed Expression Profiles*, ISMB 2001, Copenhagen Denmark

Trey Ideker et al. *Integrated Genomic and Proteomic Analyses of a Systematically Perturbed Metabolic Network*, (2001) Science, v292, pp 929-934

Bas Dutilh *Analysis of Data from Microarray Experiments, the State of the Art in Gene Network Reconstruction*, 1999, Literature Thesis, Utrecht University, Utrecht, The Nederlands

Hidde de Jong *Modeling and Simulation of Genetic Regulatory Systems: A Literature Review*, 2001, submitted

Haefner, *Modeling Biological Systems,* Chapman & Hall, 1996

Yuh, Moore, and Davidson, *Quantitative functional interrelations within the cis-regulatory system of the S. purpuratus Endo16 gene*, 1996, Development 122

Chen, Filkov, Skiena, *Identifying Gene Regulatory Networks from Experimental Data*, RECOMB 1999, Lyon, France