Learning the Structure of Biochemical Signaling Pathways

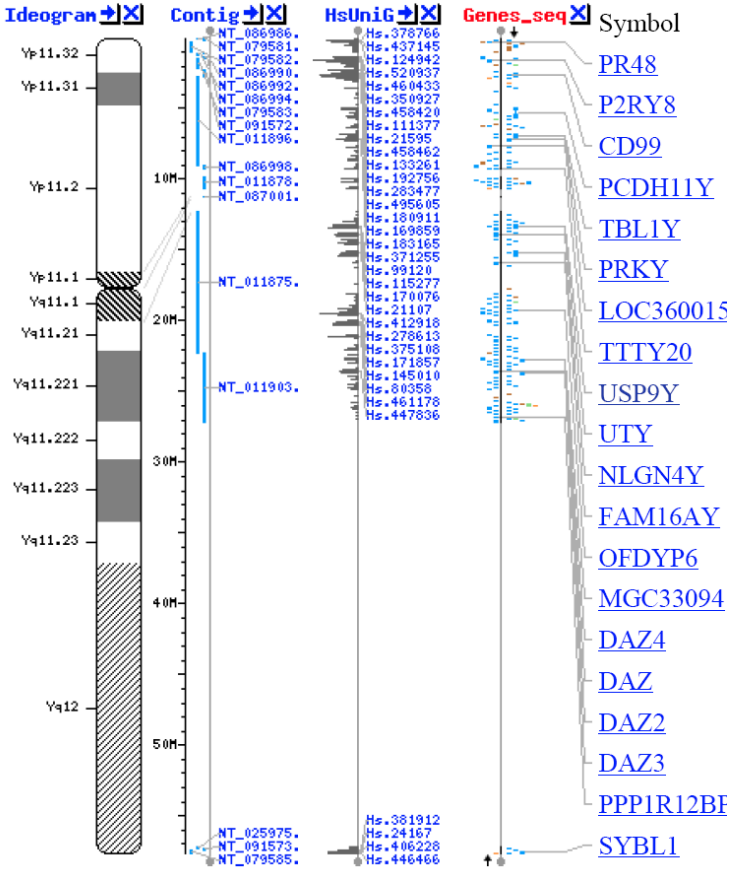
Holly Waisanen

and

Joshua Apgar

Human Genome Project

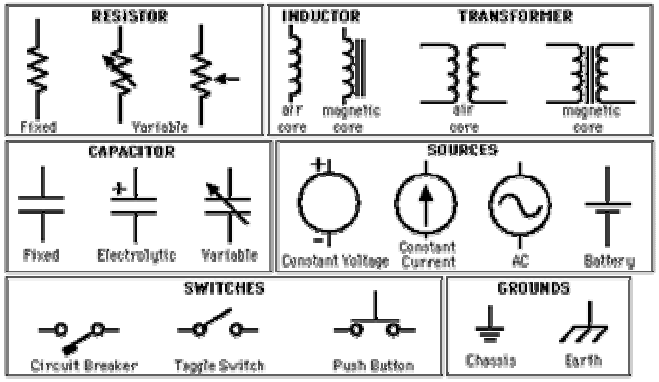
System



Parts Catalog

What's Missing?

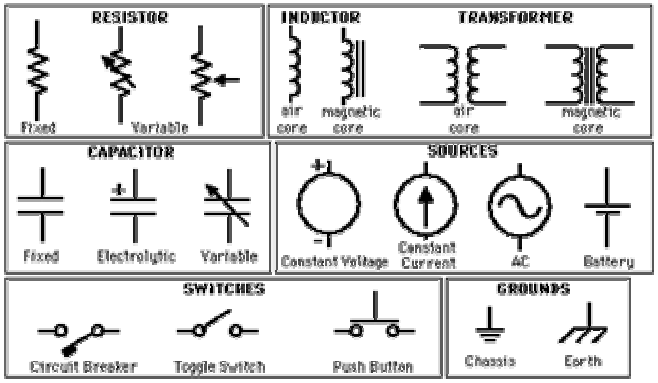
System



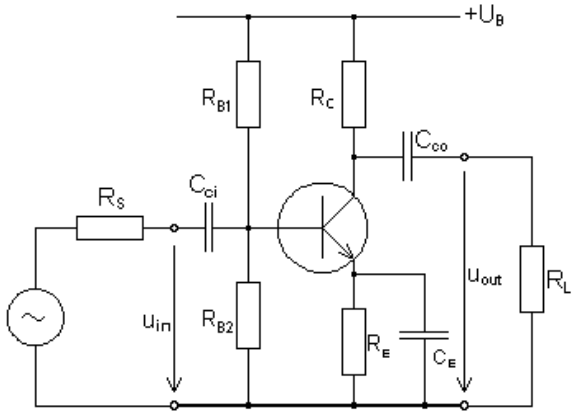
Parts Catalog

Models

System

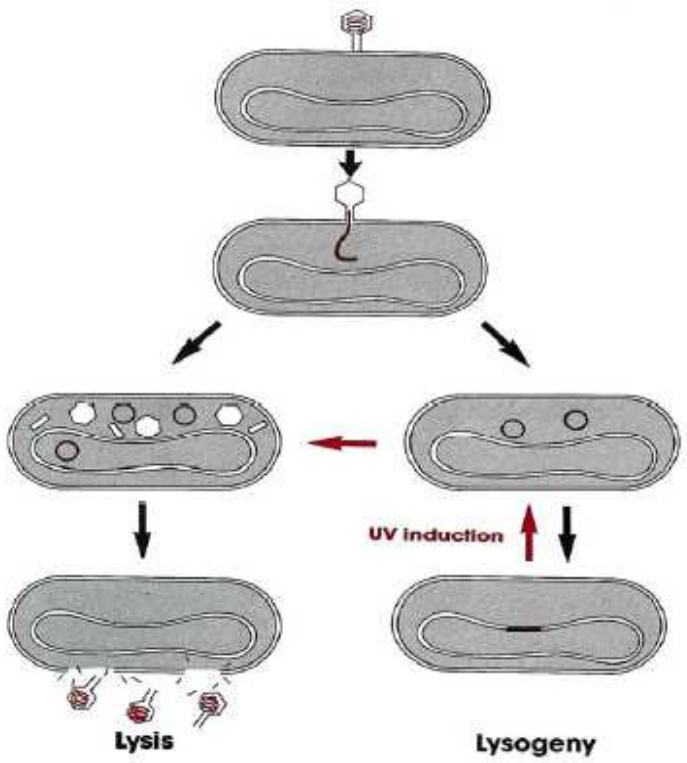


Parts Catalog

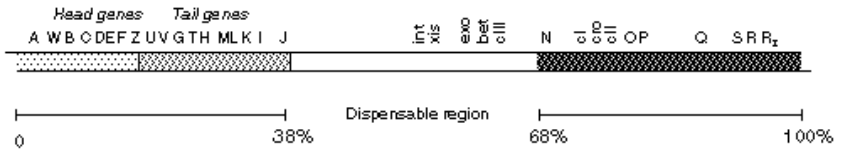


Model

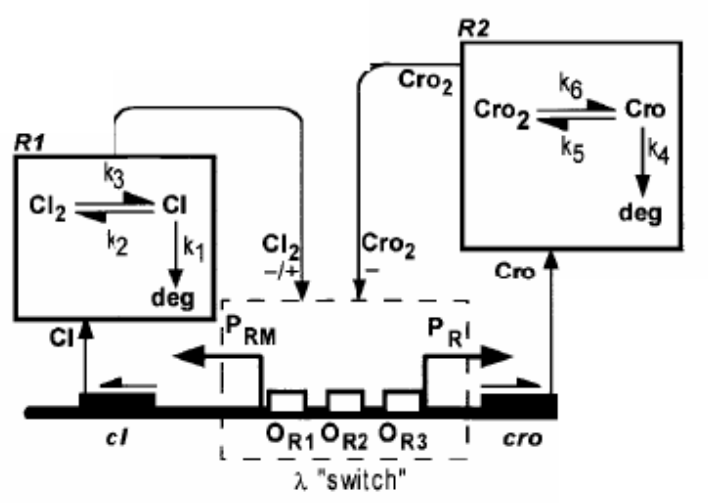
Bacteriophage-λ Lysis Lysogeny



System

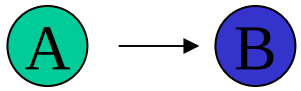
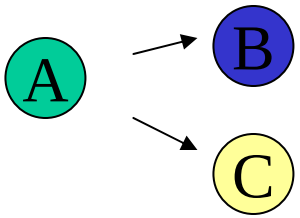
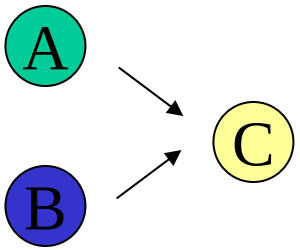


Parts Catalog

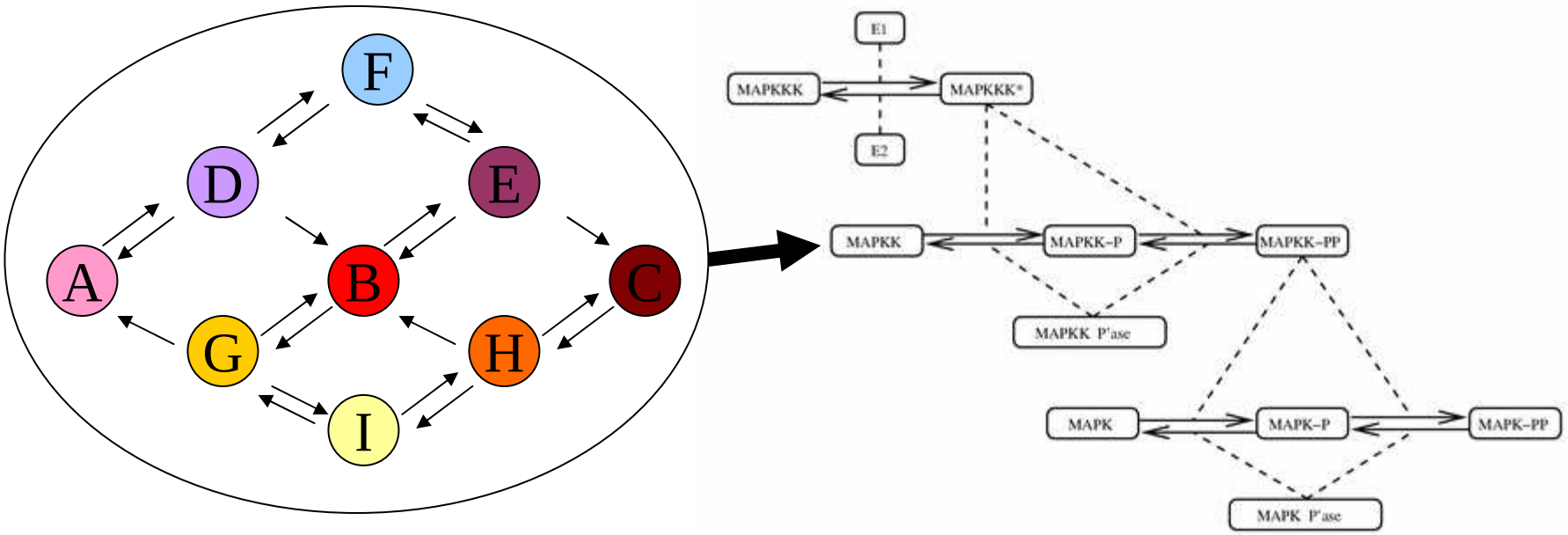


Model

Reaction Channels

Reaction	Type	Products	Rate
	Unimolecular	1	$dx_b = a_{a,b} x_a dt$
	Unimolecular	2	$dx_b = a_{a,bc} x_a dt$ $dx_c = a_{a,bc} x_a dt$
	Bimolecular	1	$dx_c = a_{ab,c} x_a x_b dt$

Networks of Reaction Channels



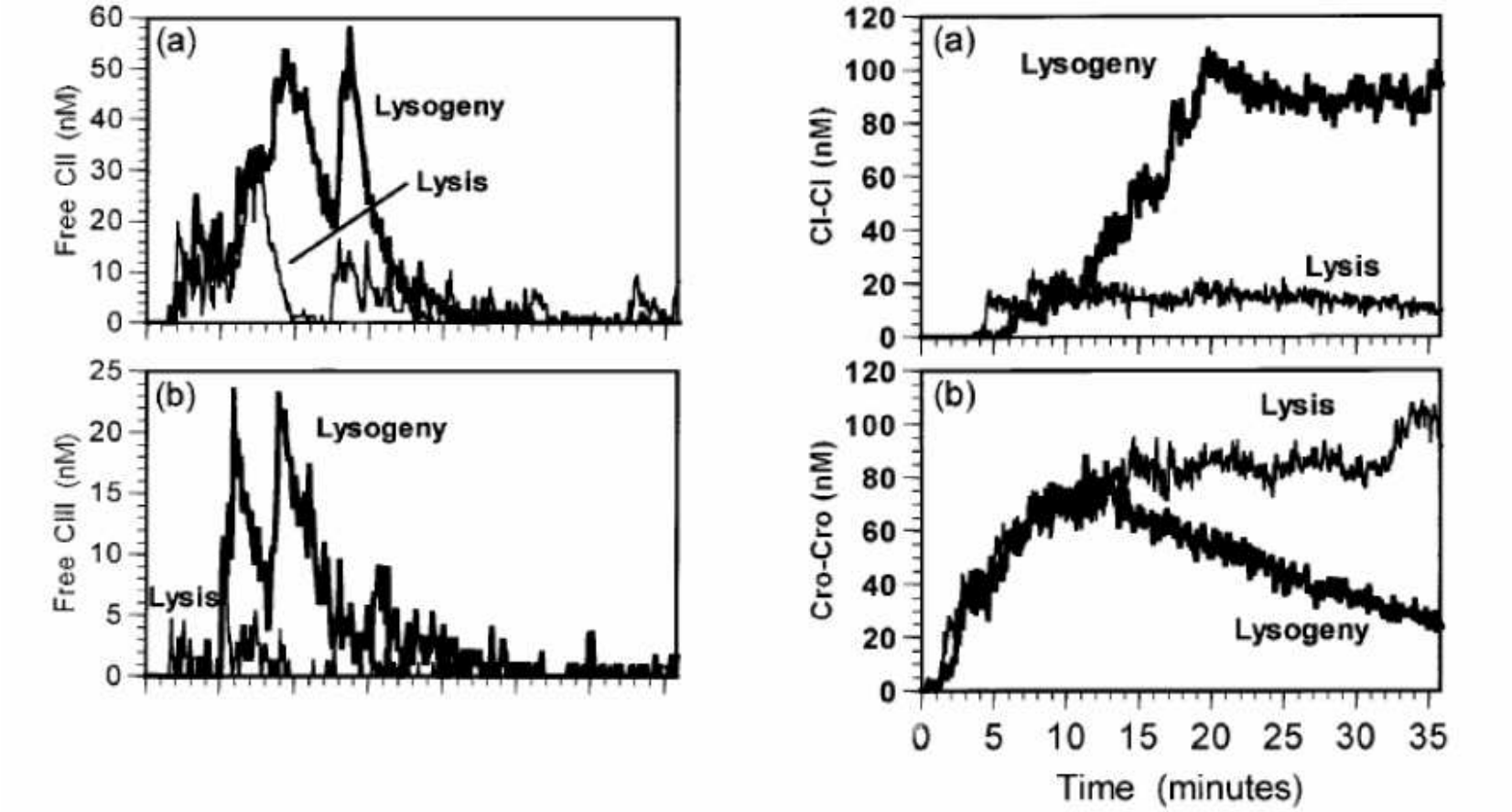
Mass Action Kinetics

$$dx = \left(A^{(1)} x + A^{(2)} x \otimes x \right) dt$$

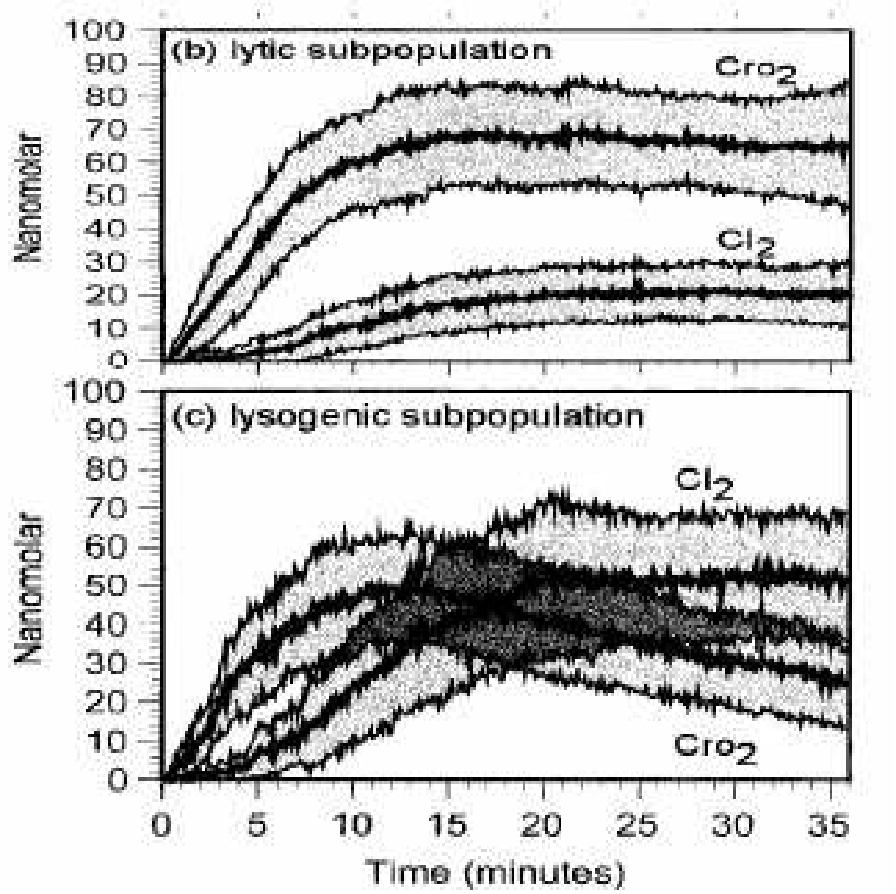
$A^{(1)}_{i,j}$ = Probability that a given $x_i \rightarrow x_j$ in time dt

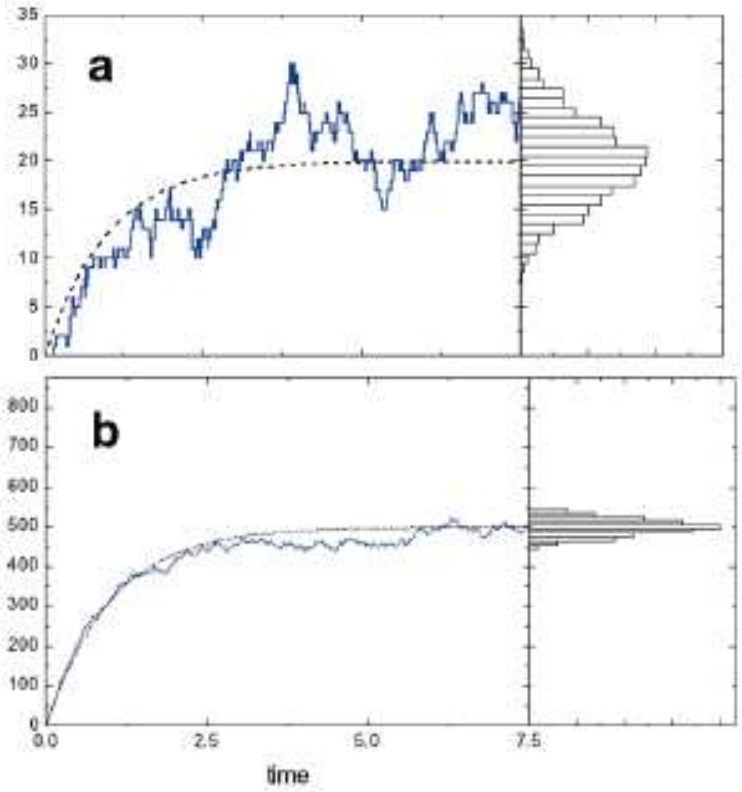
$A^{(2)}_{ij,k}$ = Probability that a given pair $x_i + x_j \rightarrow x_k$ in time dt

Systems Are Discrete

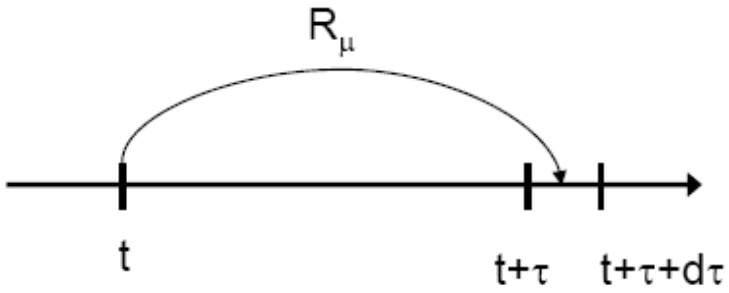


Systems Are Stochastic





Gillespie *Exact Method*



Probablility that the next reaction will be μ at time τ:

$$p(\tau,\mu)d\tau = p_0(\tau)a_\mu d\tau \qquad p_0(\tau) = \exp\left(-\sum_\mu a_\mu \tau\right)$$

Which gives the joint probability distribution:

$$p(\tau,\mu)d\tau = a_\mu \exp\left(-\sum_\mu a_\mu \tau\right)d\tau$$

Simulating the Gillespie Method

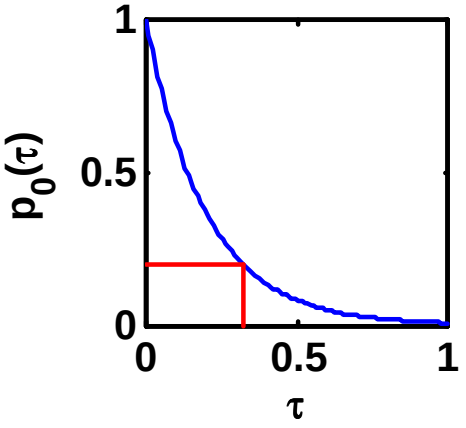
The Joint distribution can be broken into two simple distributions:

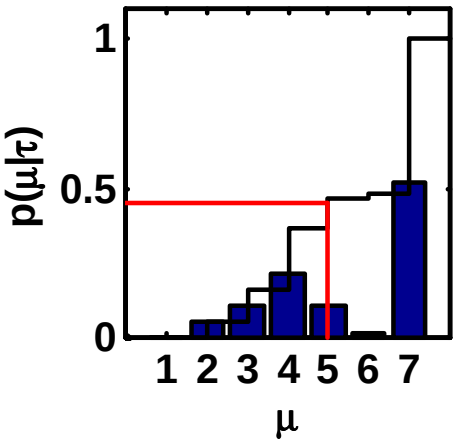
- The next reation time distribution:

$$p(\tau)d\tau = \left(\sum_{\mu} a_{\mu} d\tau\right) \exp\left(-\sum_{\mu} a_{\mu}\tau\right)$$

- The next reation distribution:

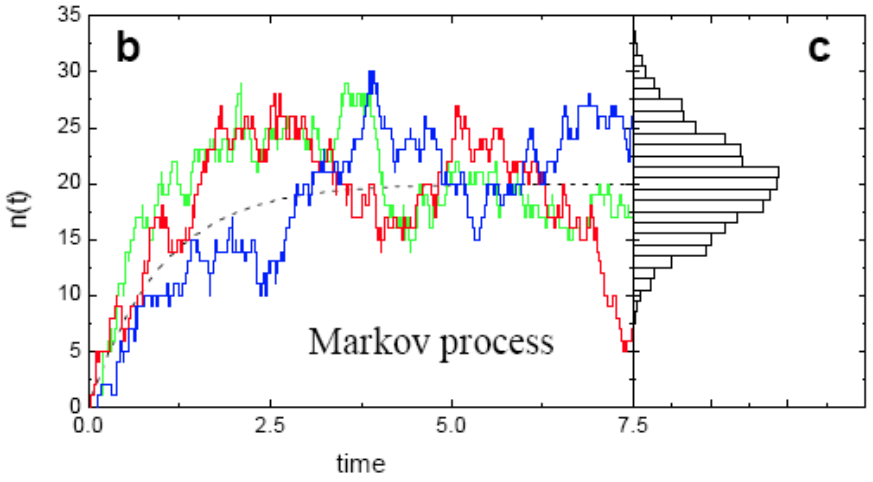
$$p(\mu \mid \tau)d\tau = \frac{a_{\mu}}{\sum_{\mu} a_{\mu}} d\tau$$



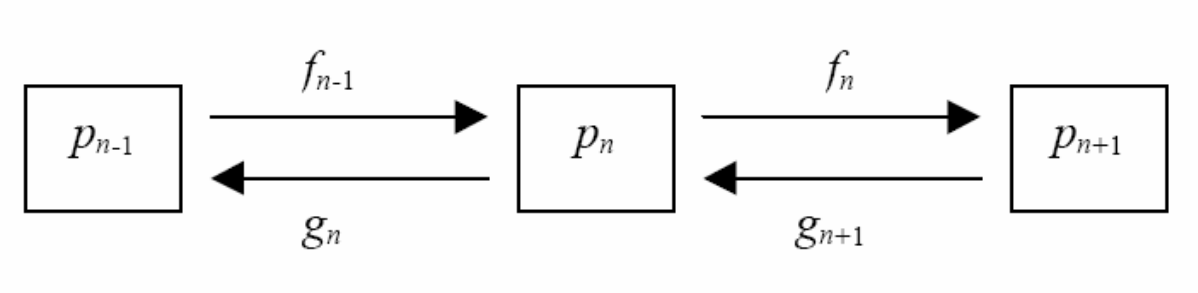


Gillespie Method Generates a Sample Path

- Method is “Exact” in the sense that it makes no averaging assumptions
- Gives a sample path a not a distribution



A Continuous Approximation



Starting with the Master Equation:

$$\frac{dp_n}{dt} = -(f_n + g_n)p_n + f_{n-1}p_{n-1} + g_{n+1}p_{n+1}$$

Approximating as continuous functions and Taylor expanding:

$$f(n-1)p(n-1) = f(n)p(n) - \frac{\partial}{\partial n} f(n)p(n) + \frac{1}{2}\frac{\partial^2}{\partial n^2} f(n)p(n)$$

$$f(n+1)p(n+1) = f(n)p(n) + \frac{\partial}{\partial n} f(n)p(n) + \frac{1}{2}\frac{\partial^2}{\partial n^2} f(n)p(n)$$

This gives the Fokker Plank Equation:

$$\frac{dp(n,t)}{dt} = -\frac{\partial}{\partial n}\left[\left(f(n) - g(n) \right) p(n) - \frac{1}{2}\frac{\partial}{\partial n} p(n) \right]$$

Fokker Plank Steady State

At Steady State:

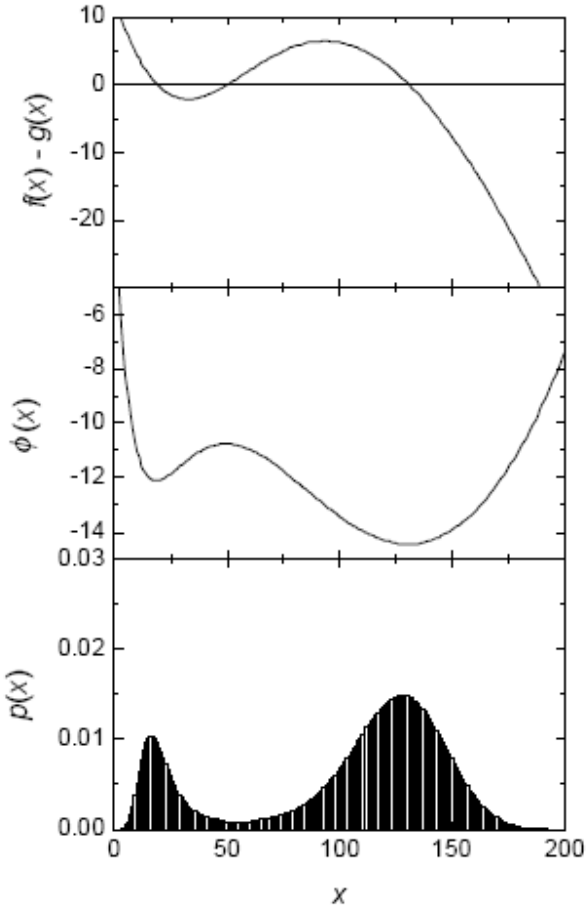
$$0 = -\frac{\partial}{\partial n}\left[\left(f(n) - g(n) \right) p(n) - \frac{1}{2}\frac{\partial}{\partial n} p(n) \right]$$

$$\left(f(n) - g(n) \right) p(n) - \frac{1}{2}\frac{\partial}{\partial n} p(n) = C$$

But from positivity c = 0 so:

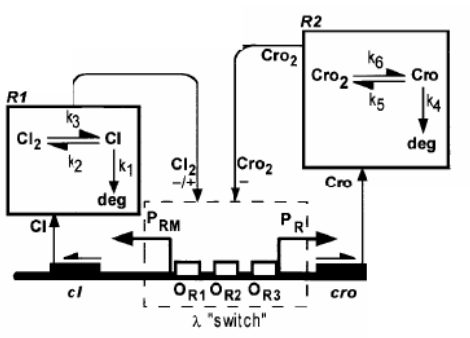
$$\frac{1}{p(n)}\frac{\partial}{\partial n} p(n) = 2\left(f(n) - g(n) \right)$$

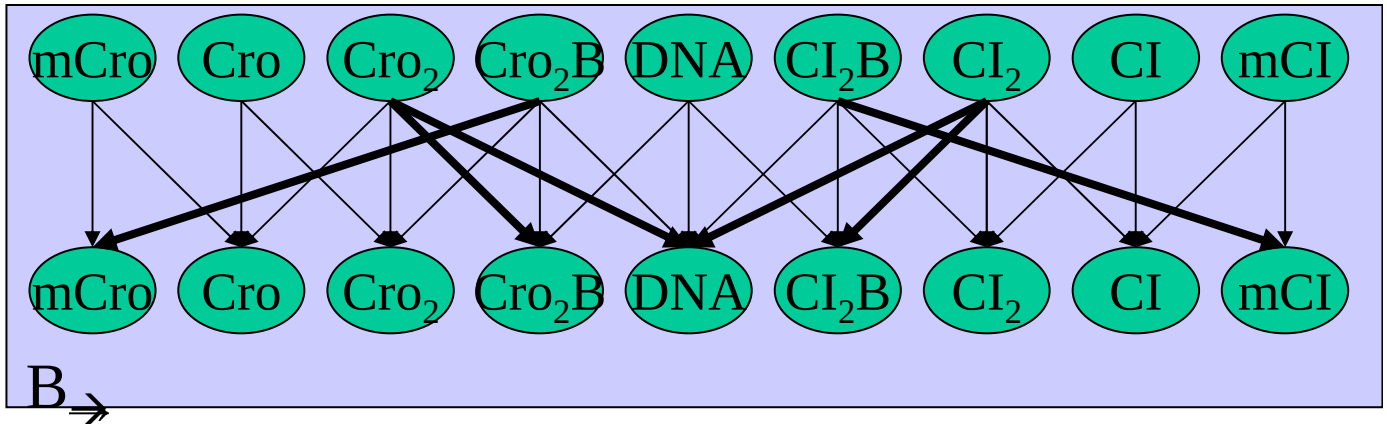
$$p(n) = \frac{A}{f(n) + g(n)} e^{-\varphi(n)} \qquad \varphi(n) = 2\int_0^n \frac{g(n') - f(n')}{f(n') + g(n')} dn'$$



Bayesian Networks

BN – graphical model for probabilistic relationships between variables





Node probabilities are independent given node parents

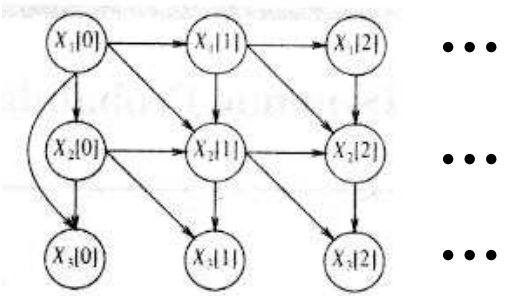
– $P_B(x_1, \ldots, x_n) = \prod_{i=1}^{n} P_B(x_i|pa(X_i))$

Dynamic Bayesian Networks

DBN – models stochastic evolution of variables over time

- Assumes time invariant evolution!
- Same independence given parents as BN, with $pa(X_i[t]) \subseteq \{X_j[t-1]\}$

Consider as a constrained semiinfinite BN

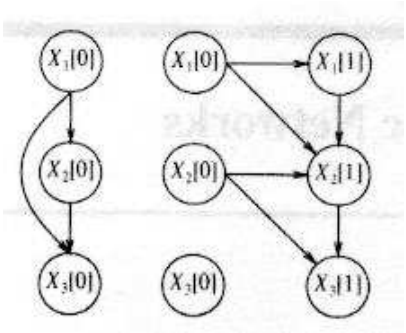


…

…

…

Or parametrize by B_0 and $B_{\rightarrow}$



Model Selection for BNs

Define a BN $\Phi = (G, \Theta)$

- G structure
  - Which connections, which entries in A matrix are nonzero
- Θ parameters
  - Arise in conditional probabilities, values of nonzero entries in A, rate constants

Once structure is fixed, easier to find parameters → maximum likelihood

Find structure - max P(G|X)

- i.e. maximize the probability that G is the correct model given that X is the data observed

Complete information

- Markov field is fully observed → can examine transitions independently
- Given N observations of the DBN up to time n_t
  - $N*n_t$ independent realizations of the BN $B_{\rightarrow}$
  - N independent realizations of the BN B_0

May use standard techniques for model selection on the constrained semiinfinite Bayesian network

-or-

Model selection using many realizations of smaller networks $B_{\rightarrow}$ and B_0

Model Selection for BNs

By Bayes rule: $P(G|X) \propto P(X|G)P(G)$

- $P(G)$ = prior probability of model G
- $P(X|G)$ = likelihood → need to compute

Task:

$\arg\max_G P(G|X) = \arg\max_G \log P(X|G) + \log P(G)$

Model Selection for BNs → likelihood

$P(X|G) = \int P(X|G,\Theta)\, P(\Theta|G)\, d\Theta = E_{\Theta}[P(X|G,\Theta)]$

- Integral hard to compute, requires priors on parameters

Likelihood penalties – a general class of model selection criteria

- Rather than comparing $P(X|G)$, compare $P(X|G, \hat{\theta}_G)$
- $\hat{\theta}_G(X) = \arg\max_{\theta} P(X|G,\theta)$
  - $\hat{\theta}_G$ is ML estimate of θ given X assuming G is correct model
- Penalty comes from limiting comparison to only a single parameter for a given model

- $BIC = \log P(X|G, \hat{\theta}_G) - K/2 \log N$
- $AIC = \log P(X|G, \hat{\theta}_G) - K/2$

Model Selection for BNs – BIC

Recall Likelihood $P(X|G) = \int P(X|G,\Theta)\, P(\Theta|G)\, d\Theta$

Use Laplace approximation for the integral and take logarithm

- Laplace approximation assumes θ Gaussian around $\hat{\theta}_G$, i.e. $P(\Theta|G)$ Gaussian

- BIC ignores any terms in approximation that are not O(N)
  - Other terms may be computed for added accuracy

$$BIC = \log P(X|G, \hat{\theta}_G) - K/2 \log N$$

–K = # of parameters in model G

BIC ←→ MDL

Regret – difference in code length between selected and baseline

Shtarkov: to minimize maximum regret, code data X according to distribution:

$$Q(X) = P(X|G, \hat{\theta}_G) / \text{comp}_N(G, \Theta)$$

where $\text{comp}_N(G, \Theta) = \Sigma_X P(X|G, \hat{\theta}_G) = \min R_{max}$

MDL Code length = - log Q(X)

- Recall from notes: $\log \text{comp}_N(G, \Theta) \approx K/2 \log N$
  - Used a Laplace approximation here too!

→ MDL code length = $-\log P(X|G, \hat{\theta}_G) + K/2 \log N$

Minimum description length = - BIC score!

Comments on BIC

- Does not require priors on parameters
  - Effect of parameter priors disappears with large N
- Good for large N → performance for small N?
- Consistent estimate → finds true model with large N (if true model is in model class)
- Intuitive – penalizes complex models without explicit priors on models
  - Avoids overfitting
- If model priors available, may augment BIC
  - Recall $\log P(G|X) = \log P(X|G) + \log P(G)$

    $\approx \log P(X|G, \hat{\theta}_G) - K/2 * \log(N) + \log P(G)$

Model Selection for BNs → AIC

Consider choosing Q(X) to minimize K-L distance between Q(X) and actual P(X|G,θ)

$D(Q(X)||P(X|\theta)) = E_{Q(X)}[\log(Q(X)) - \log P(X|G,\theta)]$

With fixed G, don't know actual θ (or P(X|G,θ)) → expect over θ

i.e. find $Q(X) = \arg\min_{Q(X)} E_{\theta}[D(Q(X)||P(X|G,\theta))]$

Using Laplace approximation and similar analysis as before

- Assumes θ Gaussian around the ML estimate from the data θ^(X)

→code length of Q(X) minimizing the expected K-L distance is:

$$AIC = \log P(X|G, \hat{\theta}_G) - K/2$$

AIC vs. BIC

$$BIC = \log P(X|G, \hat{\theta}_G) - K/2 \log N$$

$$AIC = \log P(X|G, \hat{\theta}_G) - K/2$$

- BIC – minimized maximum regret
- AIC – minimized expected K-L distance
  - some kind of average regret?
- AIC not consistent
  - Okay, since true model probably not in model class
- AIC better than BIC for small N
- Both include natural penalty on model complexity (without using explicit structure priors!)

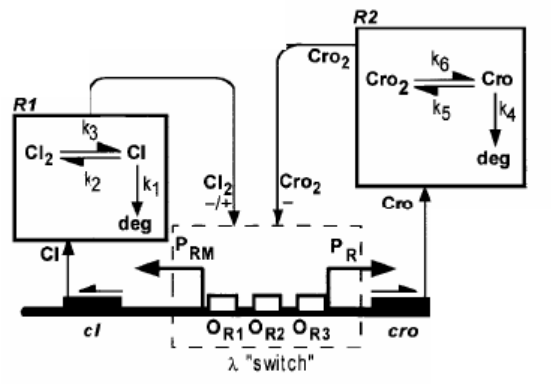
Computation from Data – local search

For each fixed structure G and given data X, compute $\hat{\theta}_G(X) = \arg\max_\theta P(X|G,\theta)$

In practice, begin with some structure G and add or delete edges

If new structure gets higher BIC score, keep it, else revert and try again

Incomplete Information



Markov field no longer fully observed

- Can't separate into many independent realizations of $B_{\rightarrow}$

Structural EM solution

- Given the model and data, complete the state information
- Use model selection criteria on completed data to find a better model structure

EM – Expectation Maximization

Given G^n and θ^n, compute $P(X|Y, G^n, \theta^n)$

– Complete the data

For each G and X, compute

$\theta\hat{}_G(X) = \arg\max_\theta P(X|G,\theta)$

– for BIC score

Find $G^{n+1} = \arg\max_G$

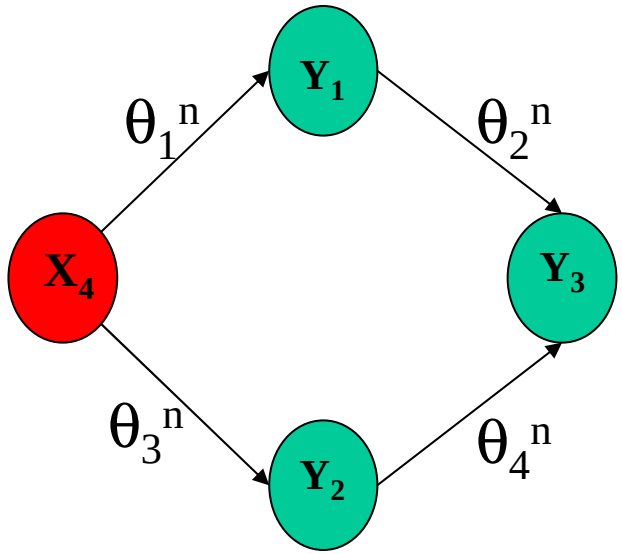
$E_{P(X|Y,Gn,\theta n)}[\log P(X|G, \theta\hat{}_G) - K/2 \log N]$

– Max over models

Given G^{n+1} and observed data Y, find

$\theta^{n+1} = \arg\max_\theta E_{P(X|Y,Gn,\theta n)}[\log P(X|G^{n+1}, \theta)]$

– Max over parameters, expected ML



EM – Expectation Maximization

Given G^n and θ^n, compute $P(X|Y, G^n, \theta^n)$

- **Complete the data**

For each G and X, compute

$\theta\hat{}_G(X) = \arg\max_\theta P(X|G,\theta)$

- for BIC score

Find $G^{n+1} = \arg\max_G$

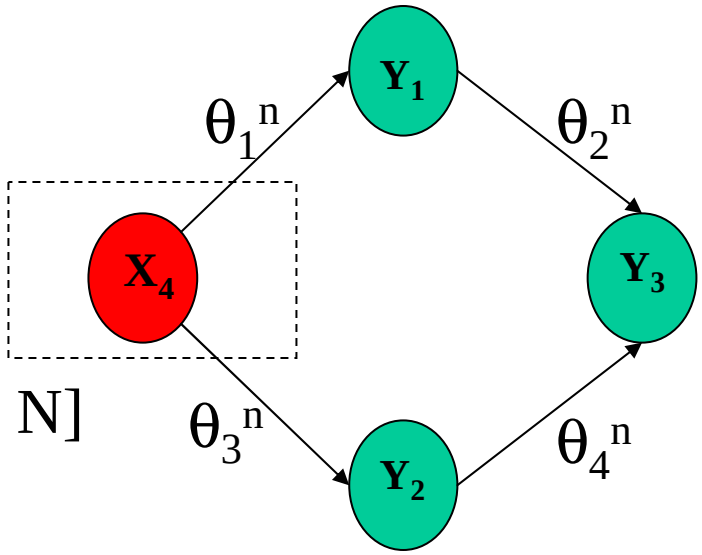
$E_{P(X|Y,Gn,\theta n)}[\log P(X|G, \theta\hat{}_G) - K/2 \log N]$

- Max over models

Given G^{n+1} and observed data Y, find

$\theta^{n+1} = \arg\max_\theta E_{P(X|Y,Gn,\theta n)}[\log P(X|G^{n+1}, \theta)]$

- Max over parameters, expected ML



EM – Expectation Maximization

Given G^n and θ^n, compute $P(X|Y, G^n, \theta^n)$

– Complete the data

For each G and X, compute

$\theta\hat{}_G(X) = \arg\max_\theta P(X|G,\theta)$

– for BIC score

Find $G^{n+1} = \arg\max_G$

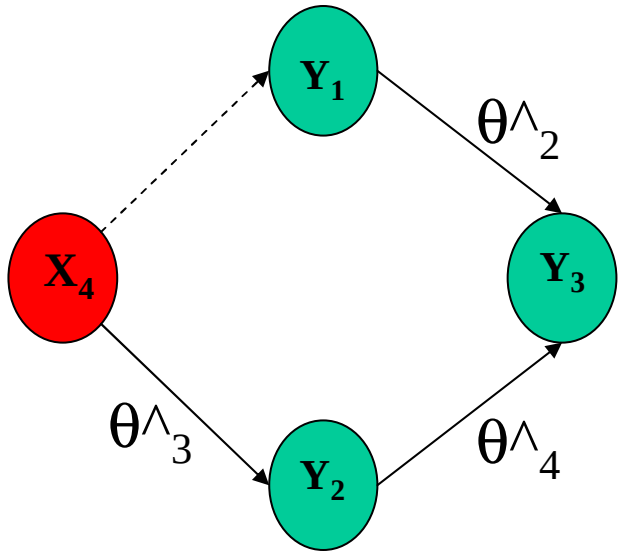
$E_{P(X|Y,Gn,\theta n)}[\log P(X|G, \theta\hat{}_G) - K/2 \log N]$

– Max over models

Given G^{n+1} and observed data Y, find

$\theta^{n+1} = \arg\max_\theta E_{P(X|Y,Gn,\theta n)}[\log P(X|G^{n+1}, \theta)]$

– Max over parameters, expected ML



EM – Expectation Maximization

Given G^n and θ^n, compute $P(X|Y, G^n, \theta^n)$

– Complete the data

For each G and X, compute

$\theta\hat{}_G(X) = \arg\max_\theta P(X|G,\theta)$

– for BIC score

Find $G^{n+1} = \arg\max_G$

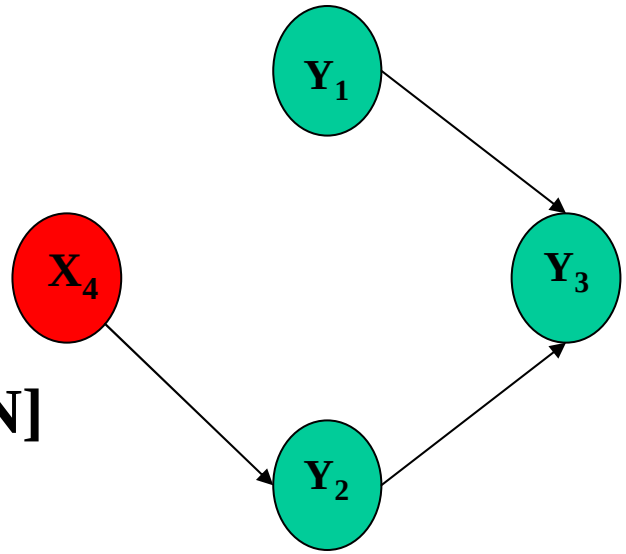
$E_{P(X|Y,Gn,\theta n)}[\log P(X|G, \theta\hat{}_G) – K/2 \log N]$

– Max over models

Given G^{n+1} and observed data Y, find

$\theta^{n+1} = \arg\max_\theta E_{P(X|Y,Gn,\theta n)}[\log P(X|G^{n+1}, \theta)]$

– Max over parameters, expected ML



EM – Expectation Maximization

Given G^n and θ^n, compute $P(X|Y, G^n, \theta^n)$

– Complete the data

For each G and X, compute

$\theta\hat{}_G(X) = \arg\max_\theta P(X|G,\theta)$

– for BIC score

Find $G^{n+1} = \arg\max_G$

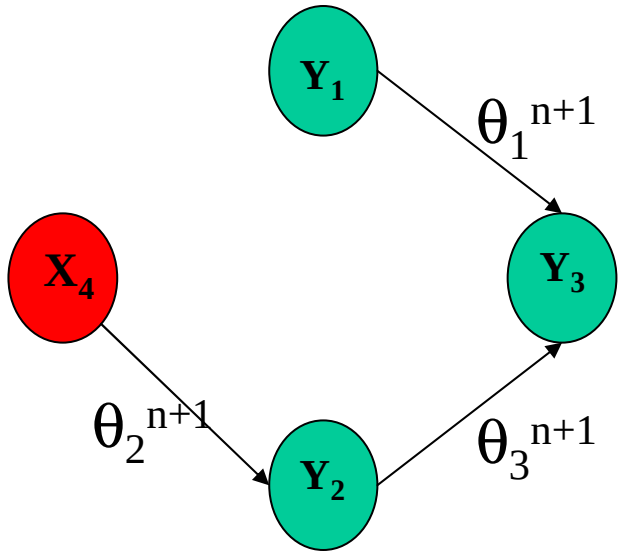
$E_{P(X|Y,Gn,\theta n)}[\log P(X|G, \theta\hat{}_G) - K/2 \log N]$

– Max over models

Given G^{n+1} and observed data Y, find

$\theta^{n+1} = \arg\max_\theta E_{P(X|Y,Gn,\theta n)}[\log P(X|G^{n+1}, \theta)]$

– Max over parameters, expected ML



EM in practice

- Compute full probability distribution of completions of data
  - By simulation methods described before
- Computing ML and expected ML
- Convergence
  - Enough to find an improving model in each step

Dynamic Optimization

$$\begin{cases} \dot{x} = f(x(p,t), p) \\ x(0) = x_0 \end{cases}$$

$$p = \arg\min \Psi(x(p, T_f))$$

$$\frac{d}{dp}\Psi = ?$$

Computing First Order Sensitivities

$$\frac{d}{dp}\Psi(x(p,T_f)) = \left(\frac{\partial \Psi}{\partial x}\frac{dx}{dp}\right|_{t=T_f}$$

$$\frac{d}{dt}\frac{dx}{dp} = \frac{d}{dp}\frac{dx}{dt} = \frac{d}{dp} f(x(p), p)$$

$$\frac{d}{dp}\dot{x} = \frac{d}{dp} f(x(p,t), p) = \frac{\partial f}{\partial x}\frac{dx}{dp} + \frac{\partial f}{\partial p}\frac{dp}{dp}$$

$$\frac{d}{dt}\frac{dx}{dp} = \frac{\partial f}{\partial x}\frac{dx}{dp} + \frac{\partial f}{\partial p}\frac{dp}{dp}$$

Integrate the Sensitivity System Along with The Dynamic System

$$\begin{cases} \dfrac{d}{dt}\begin{bmatrix} x \\ \dfrac{dx}{dp} \end{bmatrix} = \begin{bmatrix} f(x(p,t),p) \\ \dfrac{\partial f}{\partial x}\dfrac{dx}{dp} + \dfrac{\partial f}{\partial p} \end{bmatrix} \\ \left[\begin{bmatrix} x \\ \dfrac{dx}{dp} \end{bmatrix}\right]_{t=0} = \begin{bmatrix} x_0 \\ 0 \end{bmatrix} \end{cases}$$

Adjoint Method

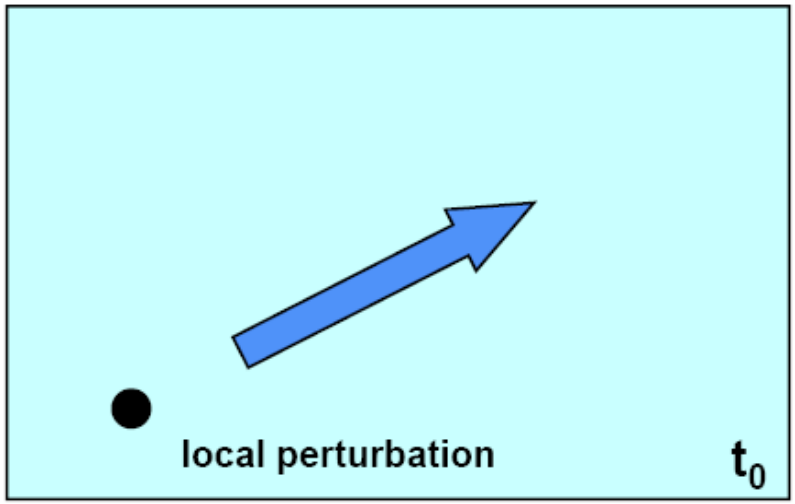
$$\begin{cases} \dot{\lambda} \equiv -\dfrac{\partial f}{\partial x}\lambda \\ \lambda_{T_f} \equiv \left.\dfrac{\partial \Psi}{\partial x}\right|_{T_f} \end{cases}$$

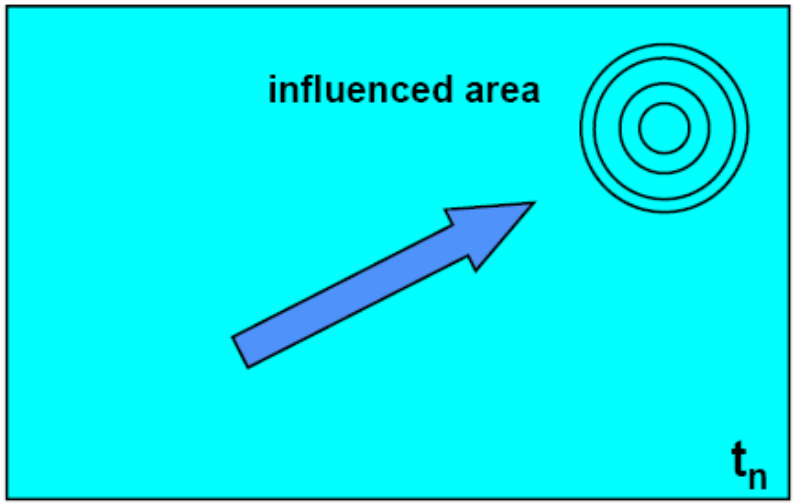
$$\frac{d}{dt}\frac{d\psi}{dp} = \frac{d}{dp}\frac{d}{dt}\psi = \frac{d}{dp}\frac{d\psi}{dx}\frac{dx}{dt} = -\frac{df}{dp}\lambda$$

Adjoint Sensitivity System

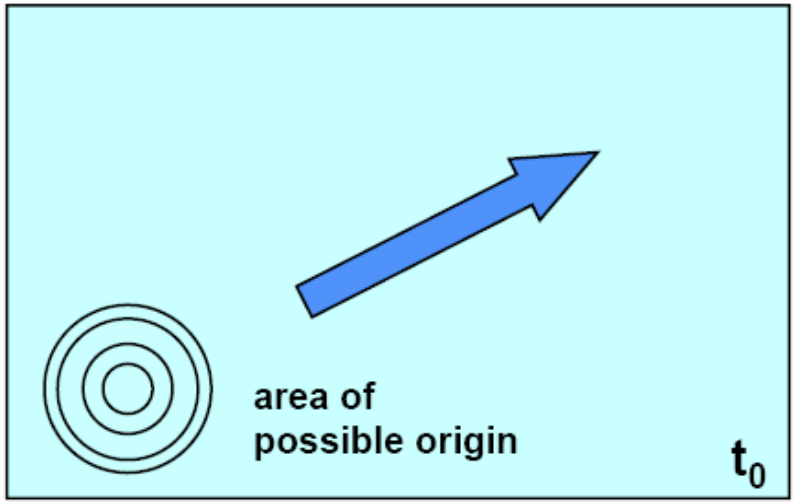
$$\begin{cases} \dfrac{d}{dt}\begin{bmatrix} x \\ \lambda \\ \dfrac{d\psi}{dp} \end{bmatrix} = \begin{bmatrix} f \\ -\dfrac{\partial f}{\partial x}\lambda \\ -\dfrac{df}{dp}\lambda \end{bmatrix} \\ \begin{bmatrix} x(0) \\ \lambda(T_f) \\ \dfrac{d\psi}{dp} \end{bmatrix} = \begin{bmatrix} x_0 \\ \left.\dfrac{\partial \Psi}{\partial x}\right|_{T_f} \\ 0 \end{bmatrix} \end{cases}$$

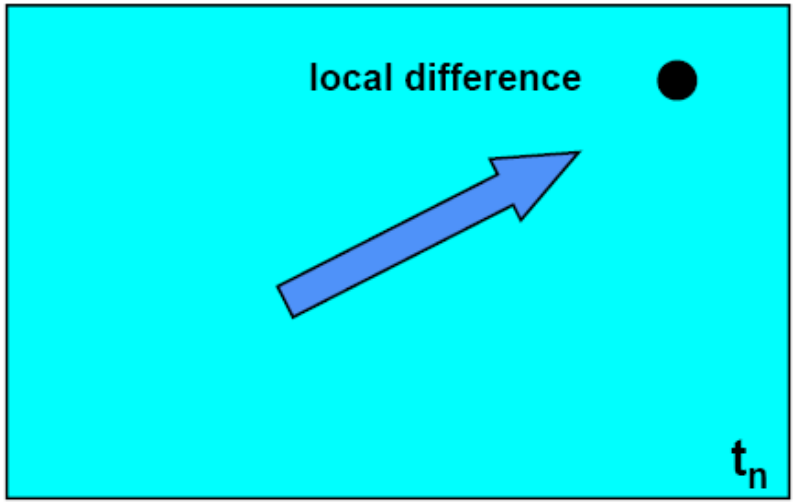
Forward Model





Adjoint Model





Conclusions

- Biochemical Signaling Pathways can be formulated as a DPN
- This formulation allows the structure of the network to be learned even in the case of partial observability
- Significant numerical challenges exist to make this feasible for large scale networks

References:

[1] Hirotugu Akaike. A new look at the statistical model identification. IEEE Trans. on Automatic Control, 19(6):716–723, 1974.

[2] Chachuat B, Singer AB, and Barton PI. Global mixed-integer dynamic optimization. AIChE Journal, 2004. In Press.

[3] Andrew Barron, Rissanen Jorma, and Bin Yu. The minimum description length principle in coding and modeling. IEEE Trans. on Information Theory, 44(6):2743–2760, 1998.

[4] Kholodenko BN, Kiyatkin A, Bruggeman FJ, Sontag E, Westerhoff HV, and Hoek JB. Untangling the wires: A strategy to trace functional interactions in signaling and gene networks. PNAS, 99(23):15245–15245, 2002.

[5] Heckerman D. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.

[6] Munther Dahleh. Lecture notes on mdl, from 17 march 05, 2005. [7] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society, 39(1):1–38, 1977.

[8] Gillespie DT. General method for numerically simulating stochastic time evolution of coupled chemicalreactions. Journal of Computational Physics, 22(4):403–434, 1976.

[9] Gillespie DT. Exact stochastic simulation of coupled chemical-reactions. Journal of Phys. Chem., 81(25):2340–2361, 1977.

[10] Sontag E, Kiyatkin A, and Kholodenko BN. Inferring dynamic architecture of cellular networks using time series of gene expression, protein and metabolite data. Bioinformatics, 20(12):1877–1886, 2004.

[11] Nir Friedman. Learning belief networks in the presence of missing values and hidden variables. In International Conference in Machine Learning, 1997.

[12] Nir Friedman, Kevin Murphy, and Stuart Russell. Learning the structure of dynamic probabilistic networks. In Uncertainty in Artificial Intelligence, 1998.

[13] David Heckerman. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.

[14] Michael Jordan and Chris Bishop. Introduction to Graphical Models (Ch 10).

[15] Sachs K, Gifford D, Jaakkola T, Sorger P, and Lauffenburger DA. Bayesian network approach to cell signaling pathway modeling. Sci STKE, 148:PE38, 2002.

[16] Friedman N, Murphy K, and Russell S. Learning the structure of dynamic probabilistic networks. UAI-1998, 1998.

[17] Schuster S and Hofer T. Determining all extremem semi-positive conservation relations in chemical-reaction systems - a test criterion for conservativity. Journal of the Chemical Society-Farady Trans., 87(16):2561–2566, 1991.

[18] Ghahramani Z. Learning dynamic bayesian networks. In Lecture Notes in Artificial Intelligence, 1997.