Identification of dynamical models of genetic networks

Eugenio Cinquemani, IBIS

23 January 2013

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
GRENOBLE - RHÔNE-ALPES

Outline

- The problem of genetic network identification
- A traditional approach: Boolean networks
- Identification of Ordinary Differential Equation (ODE) models
    - The general problem
    - Linearization methods (steady-state, time series)
    - Boolean-like methods (time series)
- Identification of stochastic models
    - An overall view
    - The Finite State Projection (FSP) Method
- Conclusions

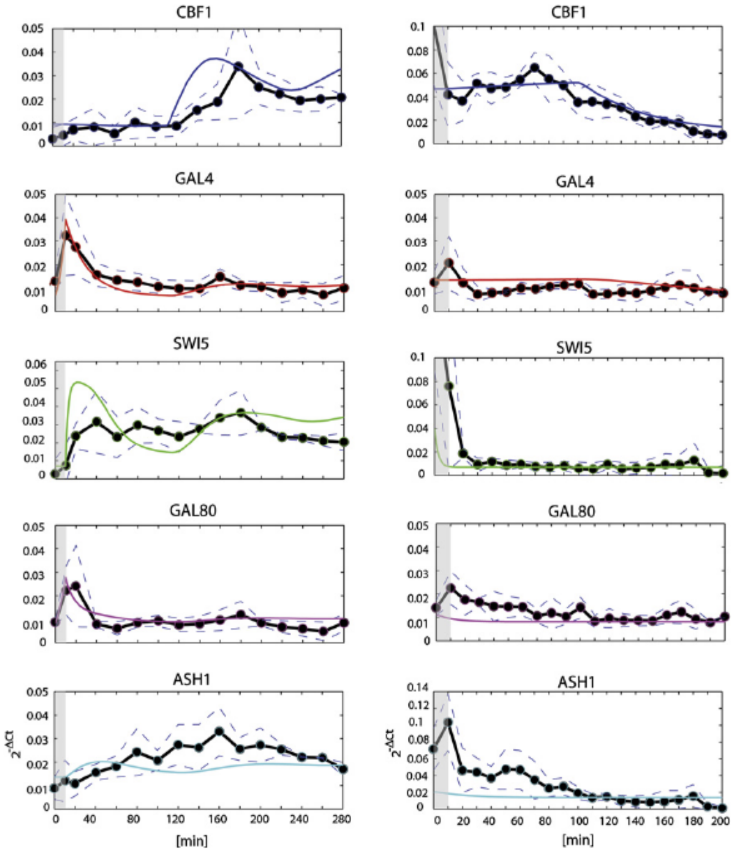
Myself

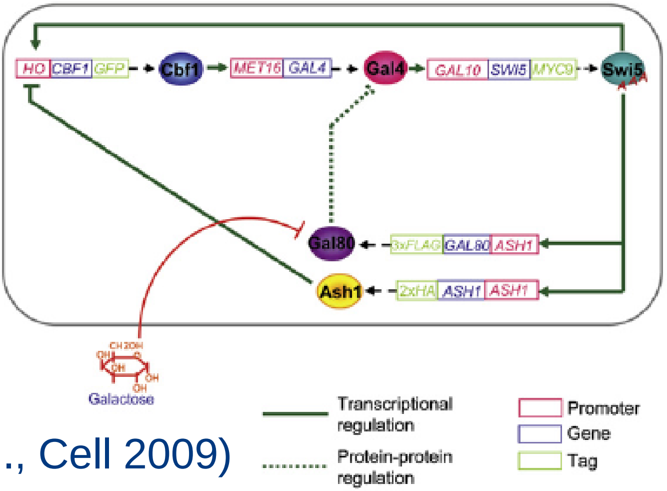
- Formation: Computer Engineering (Laurea), Automation (Ph.D.) (University of Padova, Italy)
- Post-Doc on identification of stochastic models of biological systems (and other stuff, ETH Zurich, Switzerland)
- Since November 2009, Research Scientist at INRIA (IBIS team, Grenoble – Rhône-Alpes)
    - Identification of biochemical reaction (e.g. gene) networks in bacterium *Escherichia coli* and other organisms
    - Methods for identification of regulatory network dynamics

The problem of genetic network identification

Objective

- Determine a mathematical description of the structure and behavior of a network of genes
  - Structure: genes and their interconnection
  - Behavior: inhibition vs. activation, dynamics

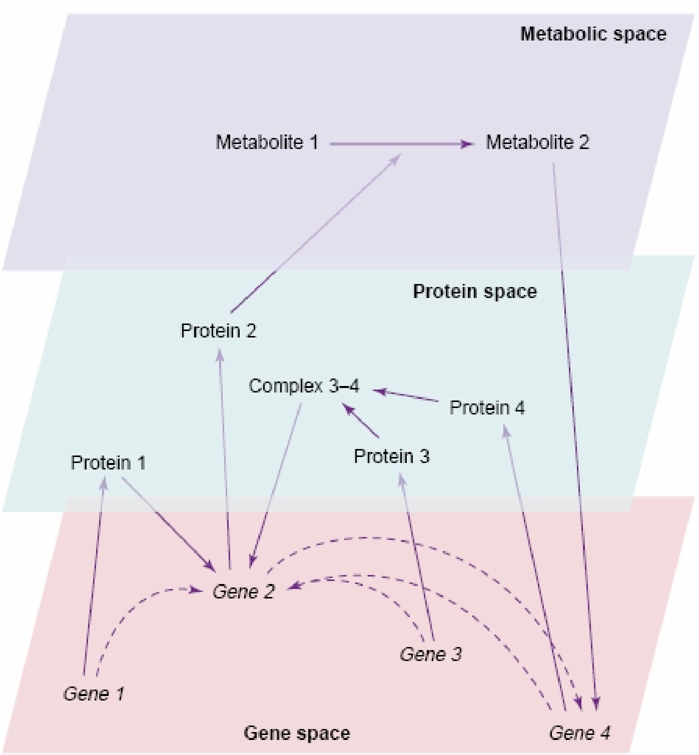


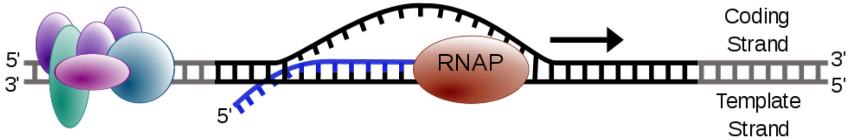


(Cantone et al., Cell 2009)

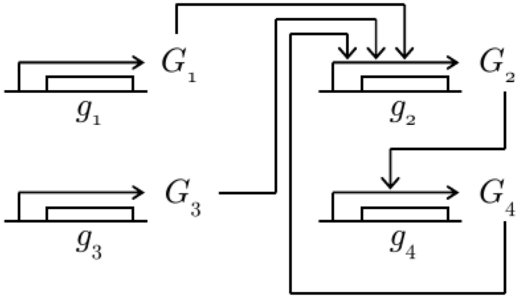
Scale

- Different levels of detail:
  - genes, but also mRNA, transcription factors, protein complexes...
  - expression: binding, DNA unfolding, transcription, translation, ...





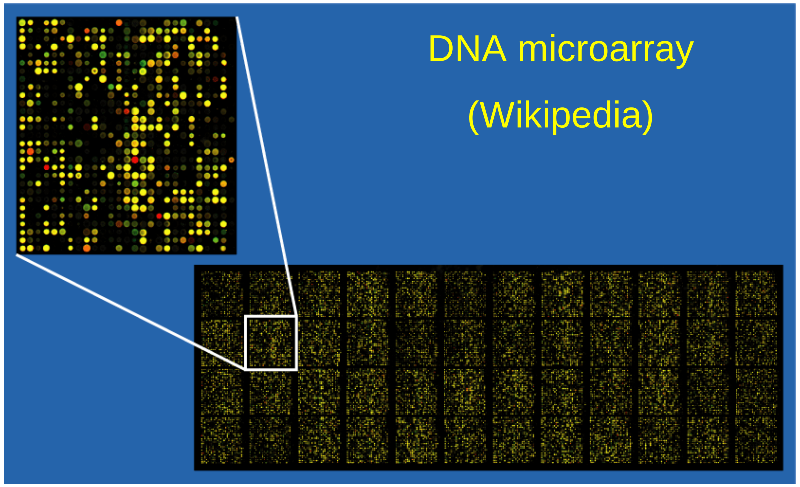
(Wikipedia)



(Brazhnik et al., 2002)

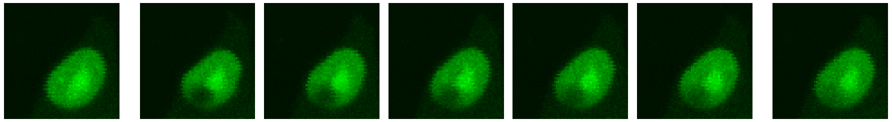
Information content

- Modelling framework depends on available data...
  - Type, quality, quantity
  - System excitation, experimental conditions

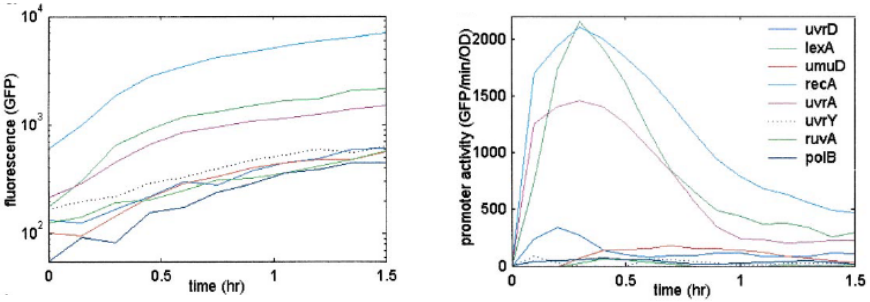


DNA microarray

(Wikipedia)



GFP fusions (courtesy of Z.Lygerou)

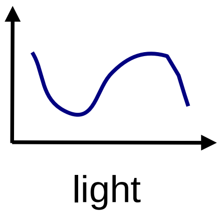


Gene reporter systems (Ronen et al, PNAS 2002)

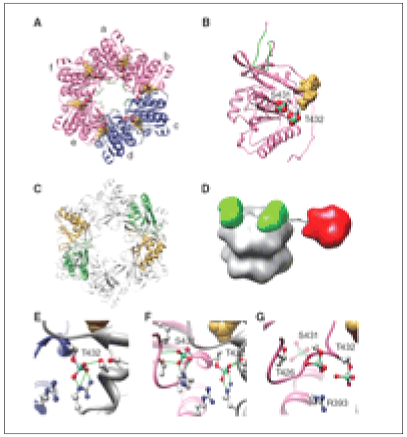
- ... and on the use of the model
  - Understanding the functioning of a biological system
  - Prediction (response of an organism to perturbations/stimuli)
  - Control (industrial exploitation, targeted chemicals for medical therapies...)

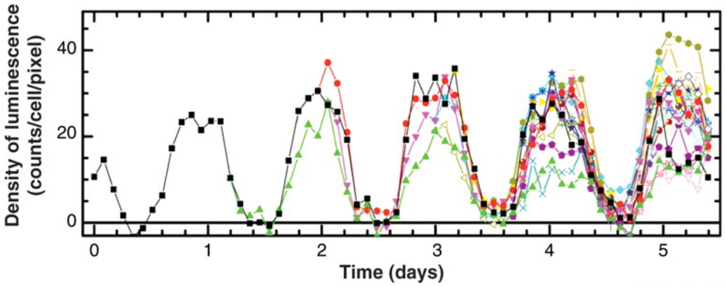
Modelling: A world of tradeoffs

- Qualitative vs. quantitative
- Mechanistic vs. phenomenological
- Fitting accuracy vs. predictive power (overfitting!)
- Complexity vs. identifiability
- Static vs. dynamic
- Black-box vs. grey-box vs. white-box



$$\dot{Y} = AY + BU$$





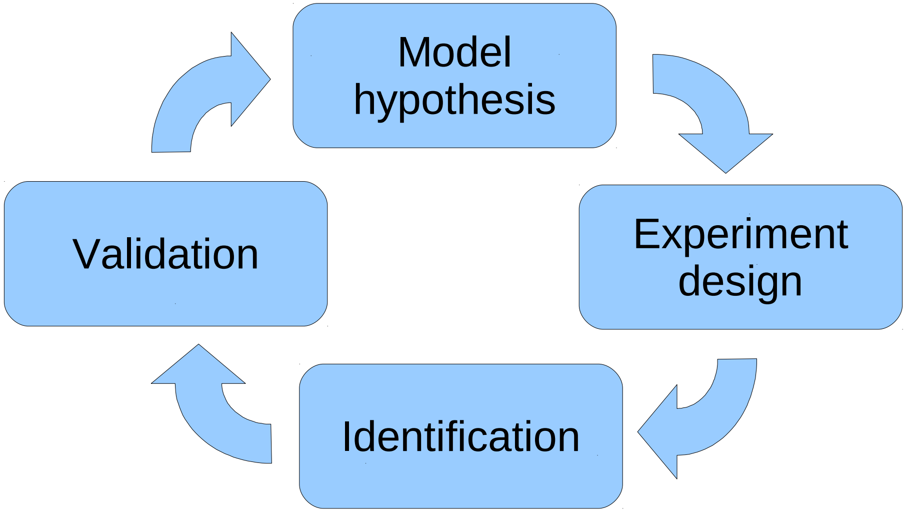
Example:

circadian rhythm

(Johnson et al, *Science,* 2008)

The identification circle

- Model hypothesis:
  - Choice of modelling framework
  - Application of first principles
  - Use of a priori knowledge
- Experiment design:
  - Address unknown model parts
  - Excite system in conditions appropriate for later use
- Identification
  - Collect data via experiment
  - Find model(s) that explains data
- Validation
  - Determine confidence level
  - Test model against new data



Today's focus: formal statement of gene network inference problems and solution with selected methods

A traditional approach: Boolean networks

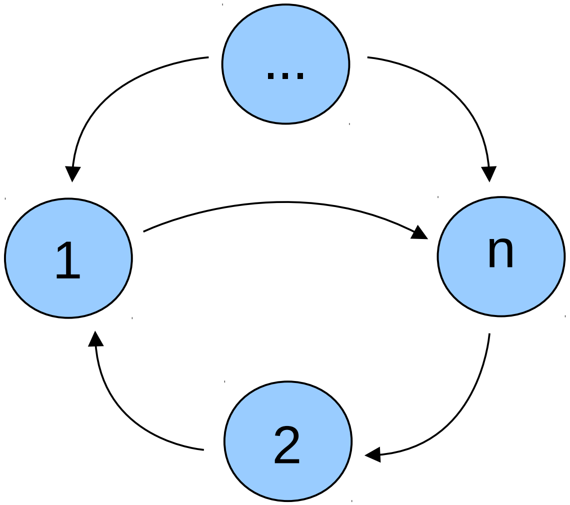
Boolean models

- N Boolean variables representing n genes

$$(X_1, X_2 \ldots, X_n) \in \{0,1\}^n$$

$$X_i = 0 \quad \text{gene not expressed}$$

$$X_i = 1 \quad \text{gene expressed}$$



- Boolean regulation function

$$X_i \text{ expressed iff } b_i(X) = 1$$

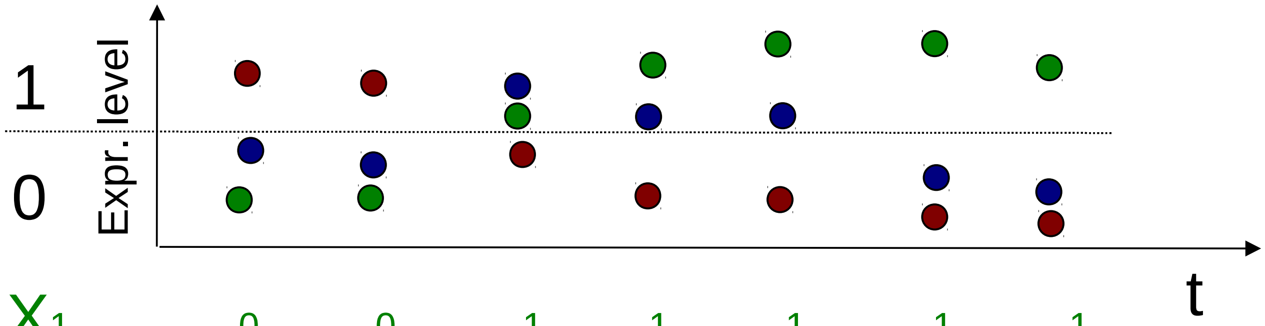
- Dynamic Boolean networks (discrete time):

$$X_i(t+1) = b_i\big(X(t)\big) \qquad i = 1, \ldots, n \qquad t = 0, 1, 2, \ldots$$

- Can associate regulatory interaction graph
  - n nodes (genes), arcs (incoming arcs of node i = effective inputs of b_i)

Identification

- Description of qualitative gene expression data



X_1	0	0	1	1	1	1	1
X_2	1	1	0	0	0	0	0
X_3	0	0	1	1	1	0	0

- Approximation of quantitative data
- Discrete math & graph theory for analysis of stability, oscillations, ...
- Learning of regulation rules from transitions observed in the data

REVerse Engineering ALgorithm

(Liang et al, 1998)

- Based on information-theoretic concepts

$X_1, \ldots, X_n$	random variables
$H(X_i)$	entropy ("variability") of X_i
$M(X_i, X_j)$	mutual information of X_i and X_j
	generalizations to sets of variables

$$\frac{M(X_i, X_j)}{H(X_i)} \in (0,1)$$

$0 =$ X_i is independent of X_j

$1 =$ X_i is fully determined by X_j

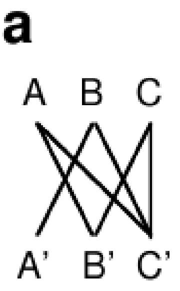
  - Functions of probability distribution of X
  - Estimated from the observed trajectories of X
  - Used to determine the effective inputs of a Boolean update map, e.g.

$$\text{If } \frac{M\big(X_1(t+1), [X_2(t), X_3(t)]\big)}{H\big(X_1(t+1)\big)} = 1 \text{ then } X_1(t+1) = b_1\big(X_2(t), X_3(t)\big)$$

  - Specific form of update map determined from the observed transitions
- May cope with noise (measurement error)
- Worst case: evaluation of all possible combinations of inputs
  - Bound complexity with maximum allowable number of inputs

Simulation example

a



c

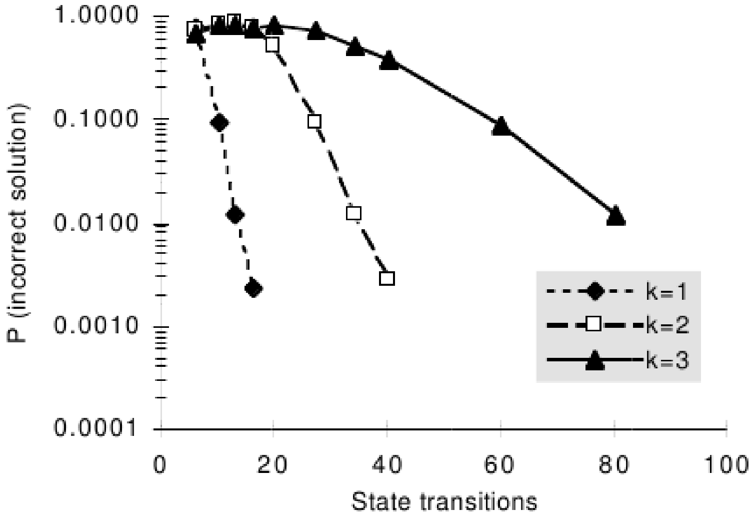
input			output		
A	**B**	**C**	**A'**	**B'**	**C'**
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	0	1	1
1	1	0	1	1	1
1	1	1	1	1	1

b

A' = B

B' = A or C

C' = (A and B) or (B and C) or (A and C)



Input entropies

H(A)	1.00
H(B)	1.00
H(C)	1.00
H(A,B)	2.00
H(B,C)	2.00
H(A,C)	2.00
H(A,B,C)	3.00

$$H(X) = -\sum p(x) \log p(x)$$
$$H(X,Y) = -\sum p(x,y) \log p(x,y)$$

$$M(X,Y) = H(X) + H(Y) - H(X,Y)$$
$$M(X,[Y,Z]) = H(X) + H(Y,Z) - H(X,Y,Z)$$

Determination of inputs for element A ①

H(A') 1.00			
H(A',A) 2.00	M(A',A) 0.00	M(A',A) / H(A')	0.00
H(A',B) 1.00	M(A',B) 1.00	**M(A',B) / H(A')**	**1.00**
H(A',C) 2.00	M(A',C) 0.00	M(A',C) / H(A')	0.00

② **Rule table for A**
rule no. 2

input	output
B	**A'**
0	0
1	1

Determination of inputs for element B ③

H(B') 0.81			
H(B',A) 1.50	M(B',A) 0.31	M(B',A) / H(B')	0.38
H(B',B) 1.81	M(B',B) 0.00	M(B',B) / H(B')	0.00
H(B',C) 1.50	M(B',C) 0.31	M(B',C) / H(B')	0.38
H(B',[A,B]) 2.50	M(B',[A,B]) 0.31	M(B',[A,B]) / H(B')	0.38
H(B',[B,C]) 2.50	M(B',[B,C]) 0.31	M(B',[B,C]) / H(B')	0.38
H(B',[A,C]) 2.00	M(B',[A,C]) 0.81	**M(B',[A,C]) / H(B')**	**1.00**

④ **Rule table for B**
rule no. 14

input		output
A	**C**	**B'**
0	0	0
0	1	1
1	0	1
1	1	1

Determination of inputs for element C ⑤

H(C') 1.00			
H(C',A) 1.81	M(C',A) 0.19	M(C',A) / H(C')	0.19
H(C',B) 1.81	M(C',B) 0.19	M(C',B) / H(C')	0.19
H(C',C) 1.81	M(C',C) 0.19	M(C',C) / H(C')	0.19
H(C',[A,B]) 2.50	M(C',[A,B]) 0.50	M(C',[A,B]) / H(C')	0.50
H(C',[B,C]) 2.50	M(C',[B,C]) 0.50	M(C',[B,C]) / H(C')	0.50
H(C',[A,C]) 2.50	M(C',[A,C]) 0.50	M(C',[A,C]) / H(C')	0.50
H(C',[A,B,C]) 3.00	M(C',[A,B,C]) 1.00	**M(C',[A,B,C]) / H(C')**	**1.00**

⑥ **Rule table for C**
rule no. 170

input			output
A	**B**	**C**	**C'**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Discussion

- Well established analysis/identification methods
- Large understanding of dynamic effects of Boolean maps
- Effective network reconstruction for qualitative data
- Wasteful use of quantitative data due to discrete approximation:
  New experimental techniques allow for more!

Identification of ODE models

The model family

- Vector of concentrations: $x = (x_1, \ldots, x_n) \in \mathbb{R}^n_{\geq 0}$
- ODE model: $\dot{x}_i = f_i(x, u, \theta) - \Gamma_i(x, u, \theta)$

$f_i \geq 0$	synthesis rate functions
$\Gamma_i \geq 0$	degradation rate functions
$\theta \in \Theta$	unknown parameters (and structure)
$u(t)$	perturbation input

- Depending on the identification approach and on the data, specific (parametric) form for rate functions
- Common choice: unregulated degradation

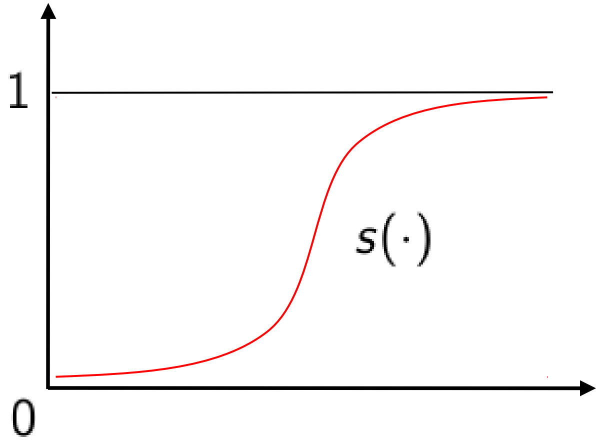
$$\Gamma_i(x_i) = \gamma_i x_i$$

Model family: examples

- Linear model plus saturation (Jaeger et al, Nature 2004):

$$f_i = s_i(\sum_j A_{i,j} x_j + b_i)$$

s_i	sigmoidal functions
$A_{i,j}$	gene connectivity matrix
b_i	basal expression rate



- Piecewise affine models (Glass & Kauffman, 1973, de Jong, ...):

$$f_i = \kappa_i^j, \quad x \in \Delta_j$$

$\{\Delta_j\}$ hyperrectangular partitioning of $\mathbb{R}^n$

The data

- Measurement model

$$y_i(t) = h_i\big(x_i(t), e\big), \qquad \begin{cases} h_i & \text{output function} \\ e & \text{(random) measurement noise} \end{cases}$$

(not always specified in all details)

- Data set

$$\mathcal{D} = \{y^m(t_k):\ k = 1, \ldots, K,\ m = 1, \ldots, M\}$$

K measurement times

M time series (possibly different inputs)

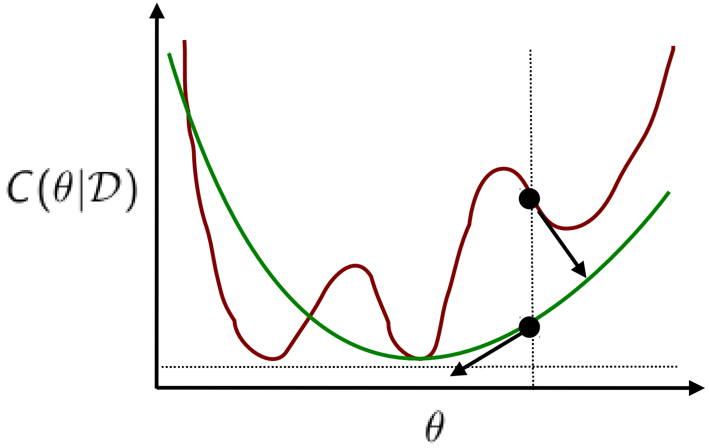
- Typically, observations of protein concentrations and/or their synthesis rates

The problem

- Identification: find “the best” model of the data in a family of alternatives
- Typical formulation: optimization of a (problem-dependent) cost function

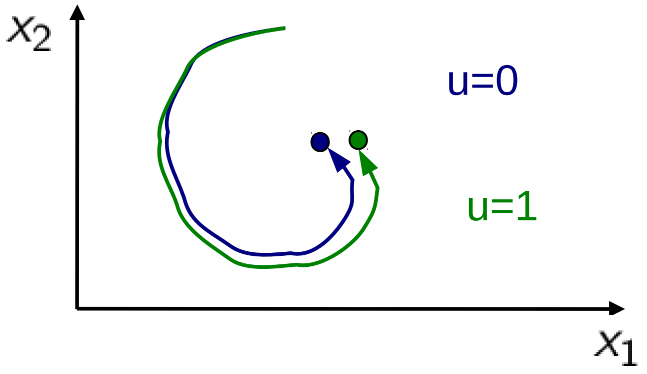
$$\text{minimize } C(\theta|\mathcal{D}) \text{ with respect to } \theta \in \Theta$$

- Cost function describes the ability of a model to explain the data
  - Minimization of the data fitting error
  - Penalization of overly complicated models to avoid overfitting
- In general, cost function is non-convex
  - Non-uniqueness of the solution
  - Optimization heuristics are needed



Linearization methods: steady state

- Working assumption:
  - all concentrations converge to an equilibrium
  - small, fixed perturbations modify the system equilibrium
  - perturbations are known, equilibria can be measured



- What perturbations ?
  - Changes in concentration of chemicals in the medium
  - Gene knockout/overexpression

- Idea: infer local dynamics around unperturbed equilibrium from several known perturbations of the system

Linearized dynamics

- True dynamics without perturbation

$$\dot{x} = \phi(x, u), \qquad u(t) \equiv 0 \text{ implies } x(t) \to x^*$$

- Linearization about equilibrium

$$\frac{d}{dt}(x - x^*) = \phi(x, u) = D_x\phi(x^*)(x - x^*) + D_u\phi(x^*)u + \text{h.o.t.}$$

- Perturbed equilibria

$$u(t) \equiv \bar{u} \text{ implies } x(t) \to x^* + \bar{x}, \text{ where}$$

$$0 = D_x\phi(x^*)(x^* + \bar{x} - x^*) + D_u\phi(x^*)\bar{u} + h.o.t \simeq A\bar{x} + B\bar{u}$$

Identification of linearized model

- Perform repeated perturbation experiments until equilibrium

$$u(t) \equiv \bar{u}_m \text{ implies } x(t) \to \bar{x}_m, \quad m = 1, \ldots, M$$

- Collect observed results in data matrices

$$U = [\bar{u}_1, \ldots, \bar{u}_M], \quad Y = [\bar{y}_1, \ldots, \bar{y}_M], \text{ where } \bar{y}_m = \bar{x}_m + e_m$$

- Solve the least-squares problem

$$\text{minimize} \quad ||AY + BU|| \quad \text{with respect to } A$$

- Solution well defined if B known and M large enough

Discussion

- A is network regulation matrix, B is (known?) perturbation effect

$A_{i,j} > 0$ gene j induces expression of gene i ($x_j \uparrow \implies x_i \uparrow$)

$A_{i,j} < 0$ gene j inhibits expression of gene i ($x_j \uparrow \implies x_i \downarrow$)

$A_{i,j} = 0$ gene j is not affected by gene i (x_j indep. of x_i)

- Explicit solution (Frobenius norm):

$$\widehat{A} = BUY^T(YY^T)^{-1}$$

warning: no zero elements (Overfitting !)

- Penalization of complexity: several strategies, e.g. “the Lasso”:

$$\begin{aligned} \min \quad & ||AY + BU|| \\ \text{s.t.} \quad & \sum_j \mathbf{1}(A_{i,j} \neq 0) \leq n_{max} \ \forall i \end{aligned}$$

$$\begin{aligned} \min \quad & \sum_{i,j} |A_{i,j}| \\ \text{s.t.} \quad & ||AY + BU|| \leq \epsilon \end{aligned}$$

Linearization methods: Time Series Network Identification

- Assumes linear dynamics (system evolving near equilibrium)

$$\frac{d}{dt}(x - x^*) = A(x - x^*) + Bu$$

- Allows for time-dependent (small) perturbation experiments
- Attempts to solve the equation

$$\dot{Y} = AY + BU$$

with the following time-course data (from a single experiment)

$$Y = [y(t_1), \dots y(t_K)], U = [u(t_1), \dots, u(t_K)], \quad y(t_k) = x(t_k) - x^* + e_k$$

- In practice derivatives not known, resort to discretized dynamics

Identification from time-series

- Discretized linear dynamics (equidistant measurement samples)

$$x(t_{k+1}) = A^d x(t_k) + B^d u(t_k)$$

- Solution of the approximate equality

$$Y^+ = [A^d \ B^d] \begin{bmatrix} Y^- \\ U \end{bmatrix},$$

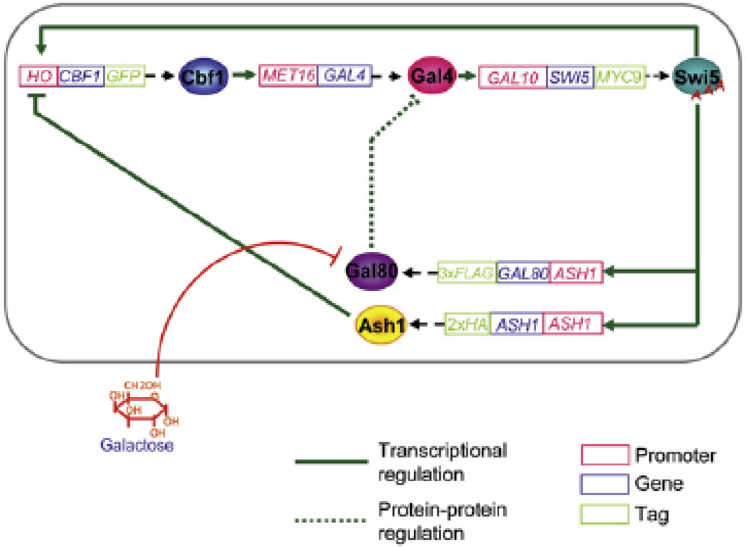
$$\begin{aligned} Y^+ &= [y(t_2), \ldots y(t_K)], \\ Y^- &= [y(t_1), \ldots y(t_{K-1})], \\ U &= [u(t_1), \ldots, u(t_{K-1})] \end{aligned}$$

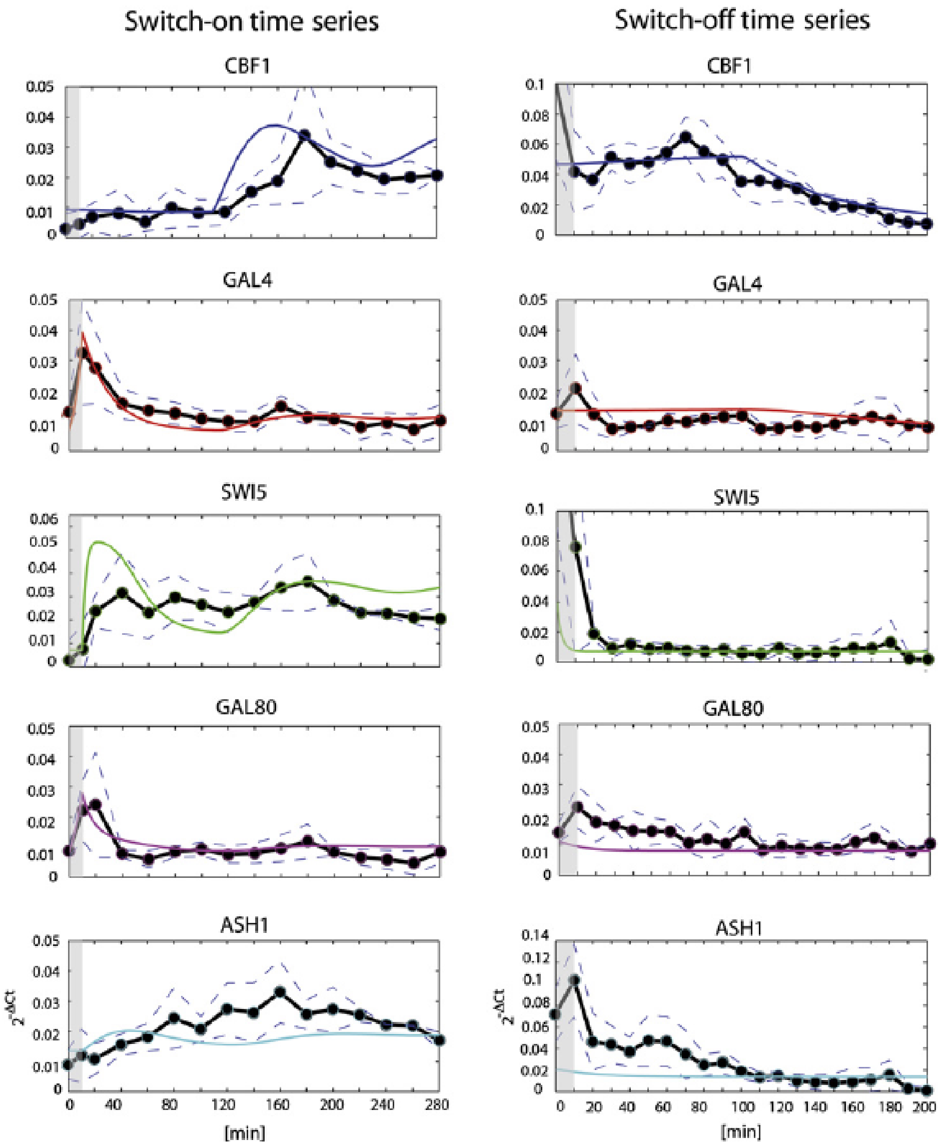
- Also identifies perturbation matrix
- Regularized solution via Principal Component Analysis (PCA)
- Conversion to continuous-time network parameters

Experiment

Synthetic gene regulatory network in Yeast

(Cantone et al., Cell 2009)





Results

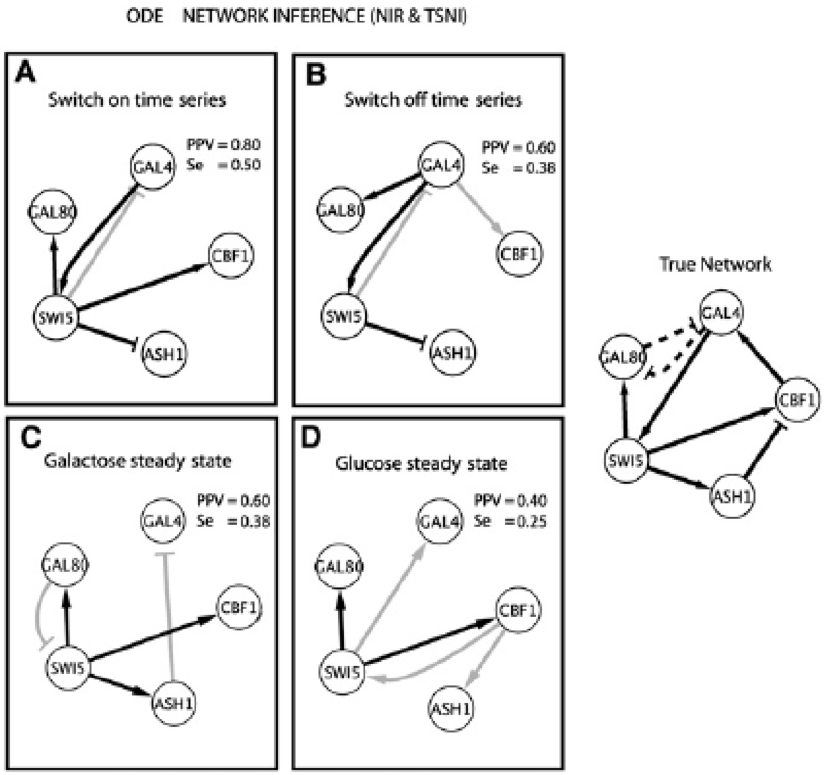


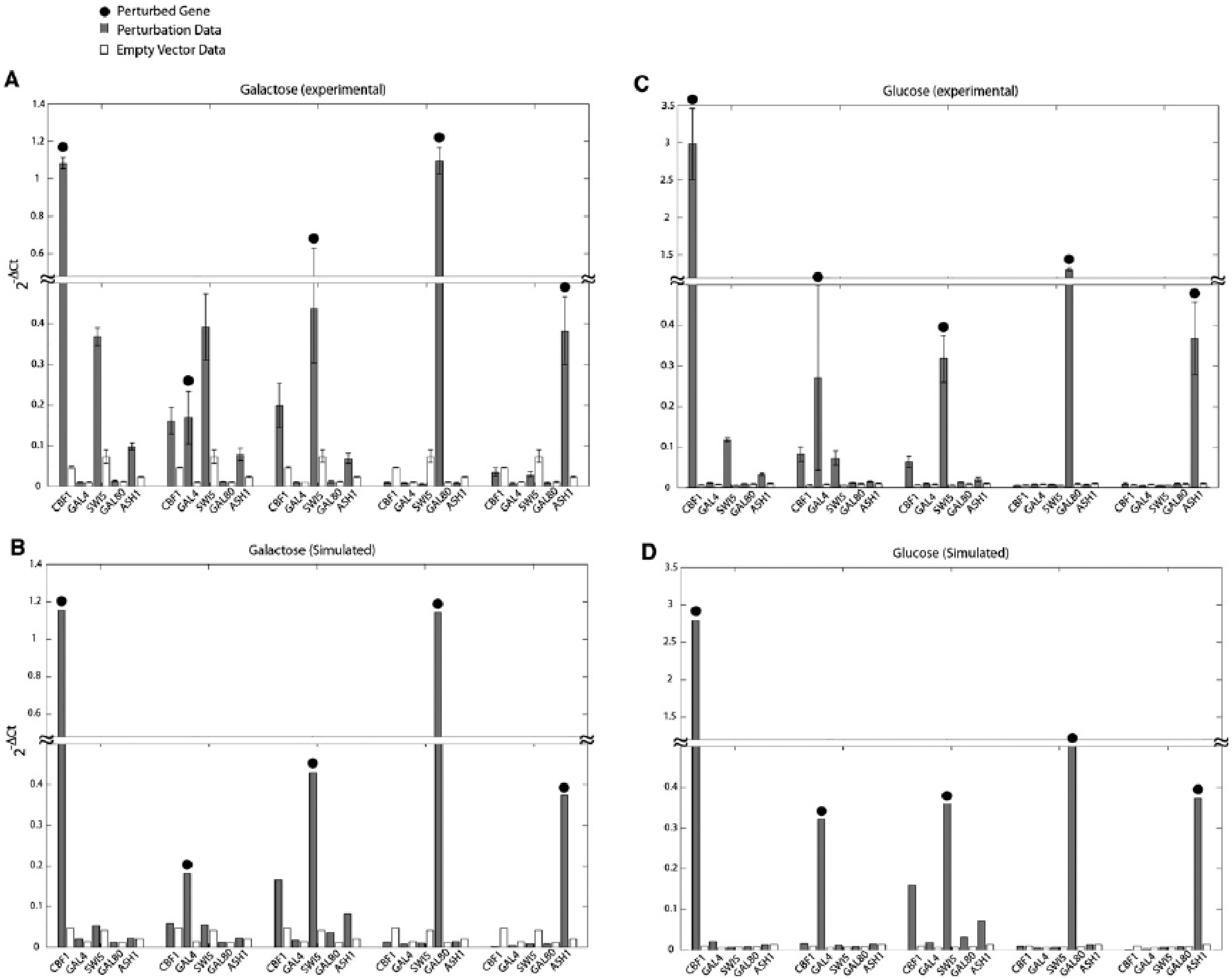
Figure 5. Reverse Engineering of the IRMA Gene Network from Steady-State and Time Series Experimental Data Using the ODE-Based Approach

The true network shows the regulatory interactions among genes in IRMA. Dashed lines represent protein-protein interactions. Directed edges with an arrow end represent activation, whereas a dash end represents inhibition.

(A and B) Inferred network using the TSNI reverse-engineering algorithm and the switch-on and switch-off time series experiments. Solid gray lines represent inferred interactions that are not present in the real network, or that have the wrong direction (FP, false positive). PPV [Positive Predictive Value = TP/(TP + FP)] and Se [Sensitivity = TP/(TP + FN)] values show the performance of the algorithm for an unsigned directed graph. TP, true positive; FN, false negative. The random PPV for the unsigned directed graph is equal to 0.40.

(C and D) Inferred network using the NIR reverse-engineering algorithm and the steady-state experimental data from network gene overexpression in cells grown in galactose or glucose medium, respectively.

Qualitative validation



Boolean-like methods

- Recall Boolean update map:

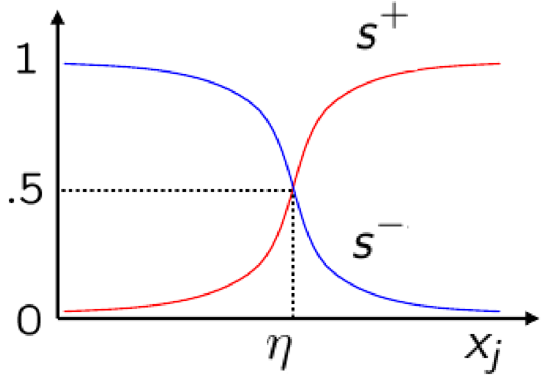
$$X_i^+ = b_i(X), \quad \text{where } b_i = \bigvee_l \bigwedge_j X'_{l,j}, \quad X'_{l,j} \in \{X_j, \neg X_j\}$$

- Algebraic equivalent (Plahte et al, 1998): apply the transformation

$$\begin{aligned} X_j &\rightarrow \sigma^+(x_j) \\ \neg\, expr(X) &\rightarrow 1 - expr(x) \\ expr(X) \wedge expr'(X) &\rightarrow expr(x) \cdot expr'(x) \end{aligned}$$

$$s^+(x_j) = \frac{x_j^d}{x_j^d + \eta^d}$$

$$s^-(x_j) = 1 - s^+(x_j)$$

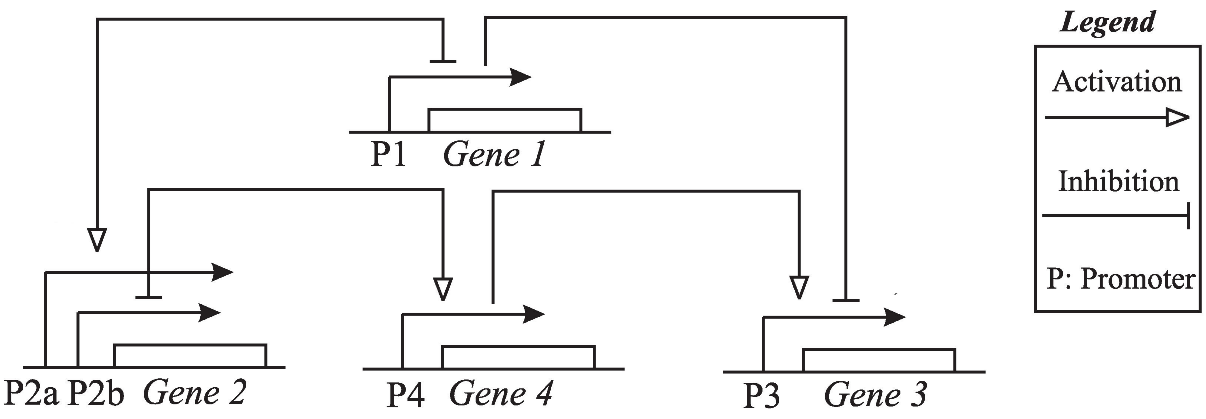


- Boolean-like model: define ODE

$$\dot{x}_i = \kappa_{0,i} + \kappa_{1,i} b_i(x) - \gamma_i x_i$$

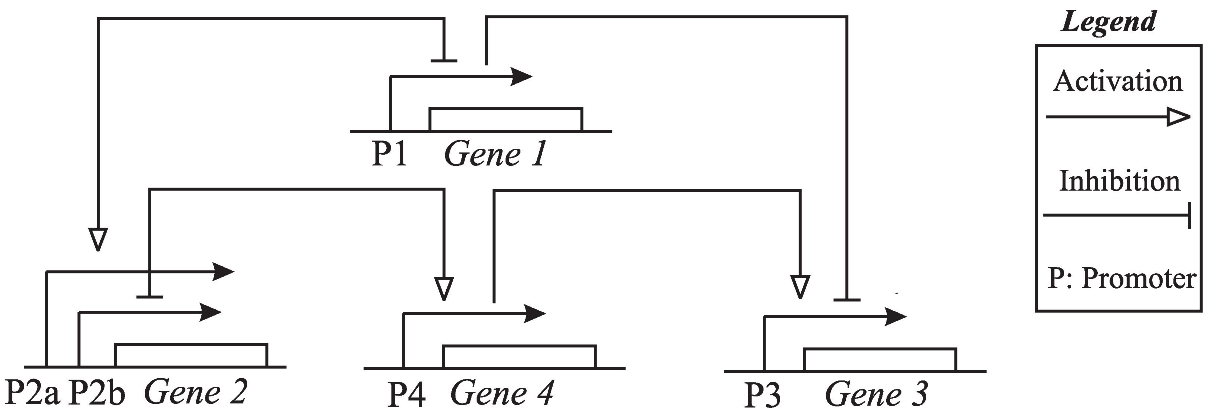
$b_i(x)$ algebraic equivalent of $b_i(X)$

Example (Boolean model)



Gene	Expressed when	Boolean model
1	G2 not expressed	$b_1(X) = \neg X_2$
2	G1 expressed or G4 not expressed	$b_2(X) = X_1 \vee \neg X_4$
3	G4 expressed and G1 not expressed	$b_3(X) = X_4 \wedge \neg X_1$
4	G2 expressed	$b_4(X) = X_2$

Example (Boolean-like ODE)



Gene	More active when
1	G2 low
2	G1 high or G4 low
3	G4 high and G1 low
4	G2 high

ODE model

$$b_1(x) = s^-(x_2)$$

$$b_2(x) = 1 - s^-(x_1) \cdot s^+(x_4)$$

$$b_3(x) = s^+(x_4) \cdot s^-(x_1)$$

$$b_4(x) = s^+(x_2)$$

Plausibility ?

- Experimental evidence that *often* (Gjuvsland et al, 2007)
  - Transcription factors combine into Boolean-like input functions
  - Sigmoidal functions relate transcription factor concentrations and transcription rates
  - Post-transcriptional, transport, (and reaction) processes at equilibrium can be described by sigmoidal functions
- Still a phenomenological framework, but ...
  - Easy to interpret biologically
  - Accurate and flexible

Tractability ?

- General Boolean-like model:

$$\dot{x}_i = \kappa_i^1 + \kappa_i^2 b_i(x) - \gamma_i x_i, \quad \text{where } b_i = \sum_l \prod_j s^{\pm}(x_j|\theta_{l,j})$$

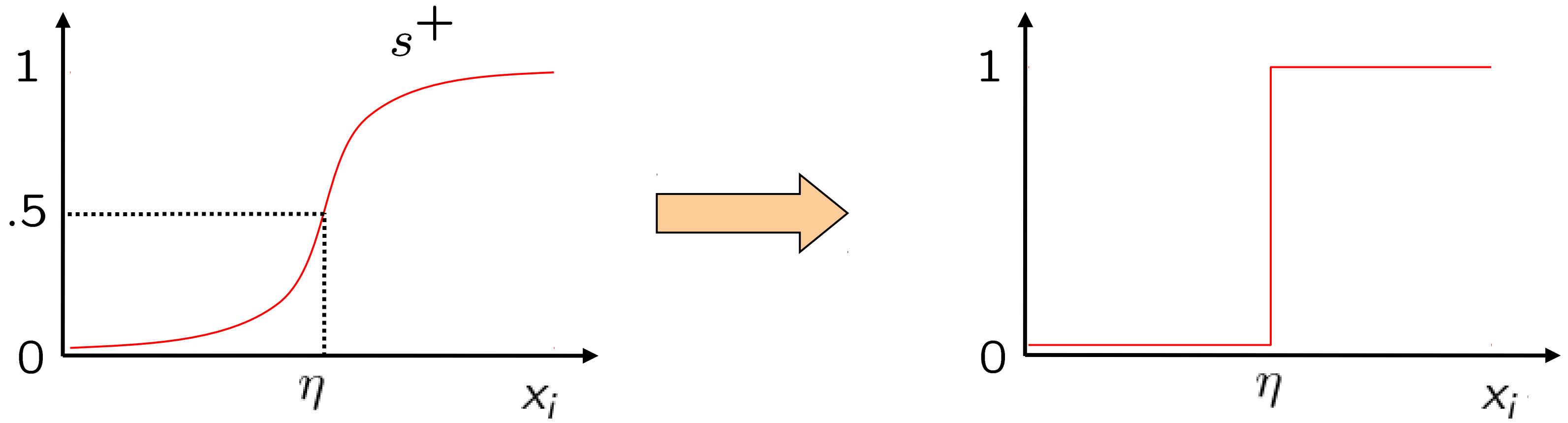
- Structure identification: based on data, decide
  - The number of summands
  - The sigmoids in each product
  - The signs of the sigmoids

... combinatorial explosion and identifiability issues !!

- Parameter identification: paramaters of each sigmoid, rates
- Intractable problem. But, good starting point
  - Structured expression
  - Reduction to specific families of Boolean-like functions
  - Approximation

Piecewise Affine models

- Simple idea: abstract nonlinearities by switches

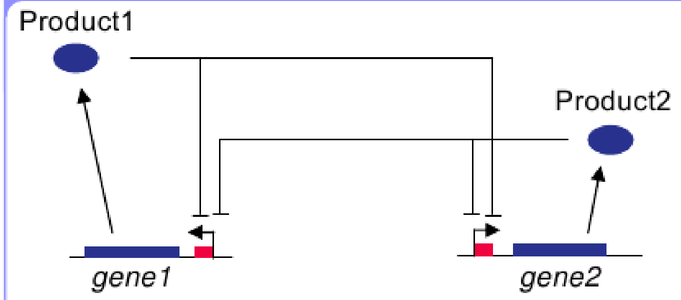


- Dynamical models with Boolean-type events
- Coarse approximation, but ...
- Powerful analysis (de Jong et al. 2004) & identification (Porreca et al, 2009) tools!

Example: double-inhibition network

Courtesy of G.Ferrari-Trecate
(apologies for notational changes...)

Double inhibition network



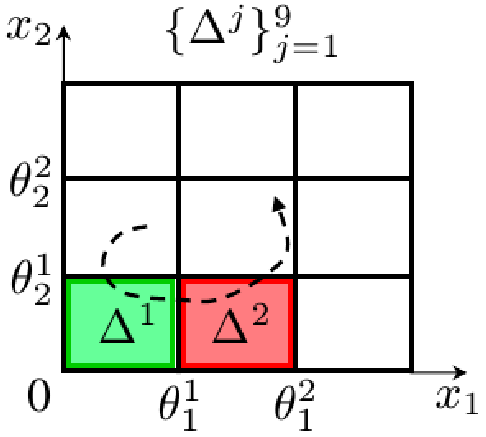
$$\dot{x}_1 = \alpha_{11} b_{11}(x) - \gamma_1 x_1$$

$$\dot{x}_2 = \alpha_{21} b_{21}(x) - \gamma_2 x_2$$

$$b_{11}(x) = s^-(x_1, \theta_1^1) s^-(x_2, \theta_2^1)$$

$$b_{21}(x) = s^-(x_1, \theta_1^2) s^-(x_2, \theta_2^2)$$

Domains and affine dynamics



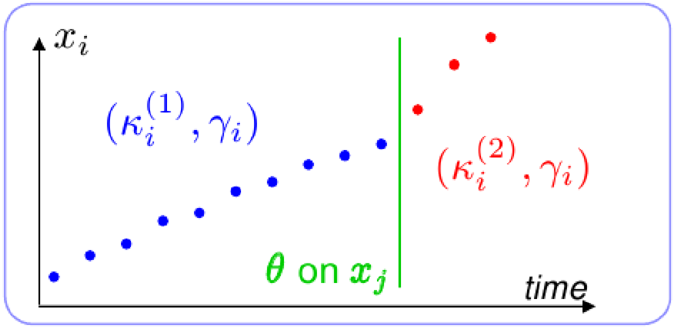
$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{cases} \boxed{\begin{bmatrix}\alpha_{11}\\ \alpha_{21}\end{bmatrix}} - \begin{bmatrix}\gamma_1 & 0\\ 0 & \gamma_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}, & \text{if } x \in \Delta^1\\ \boxed{\begin{bmatrix}0\\ \alpha_{21}\end{bmatrix}} - \begin{bmatrix}\gamma_1 & 0\\ 0 & \gamma_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}, & \text{if } x \in \Delta^2\\ \vdots & \end{cases}$$

PWA models: key features

- thresholds split Ω into **hyperrectangular domains** $\Delta^1, \Delta^2, \ldots$ **:**

$$\left.\begin{array}{l} \dot{x} = \begin{bmatrix} \kappa_1^j \\ \kappa_2^j \\ \vdots \\ \kappa_n^j \end{bmatrix} - \begin{bmatrix} \gamma_1 & 0 & \cdots & 0 \\ 0 & \gamma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \gamma_n \end{bmatrix} x \\ \text{if } x \in \Delta^j \end{array}\right\} \rightarrow \boxed{\text{system of } n \text{ decoupled affine equations}}$$

- switching thresholds and rate parameters define the interactions

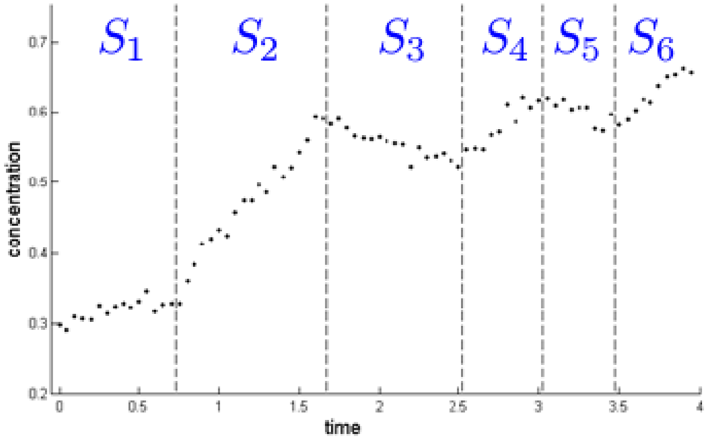


- gene *j* acts on of gene *i*
- interaction: activation/inhibition based on changes in κ_i

PWA models: key features cont'd

decoupling ⇒ local 1st order dynamics for each concentration: if no switches occur over $[t_{k_0}, t_{k_1}]$ there exist $\kappa \geq 0, \gamma > 0$ such that

$$x_i(t_{k_1}) = \frac{\kappa}{\gamma} - \left(\frac{\kappa}{\gamma} - x_i(t_{k_0})\right) e^{-\gamma(t_{k_1} - t_{k_0})}$$

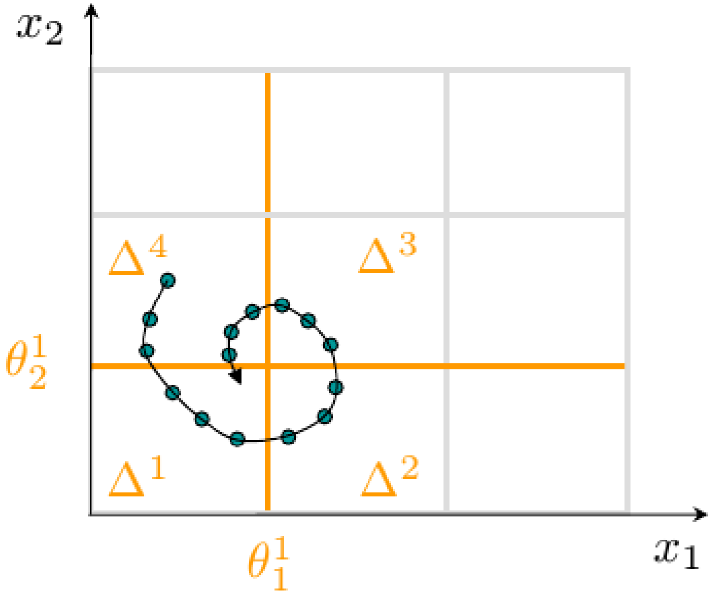


Data can be split in *segments* S_j generated by rate parameters (κ^j, γ)

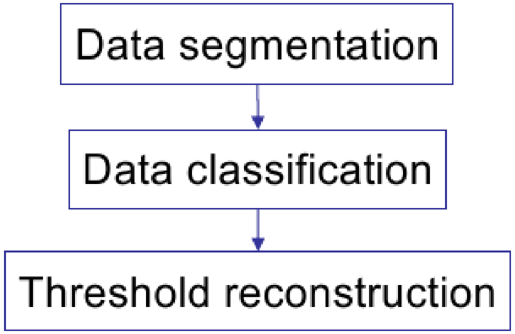
PWA model identification

Goal: reconstruct from data

- **number of submodels**
  (excited during the experiment)
- **switching thresholds**
  (defining the domains)
- **rate parameters**
  (on the reconstructed domains)



Identification algorithm


Data segmentation

Data classification

Threshold reconstruction

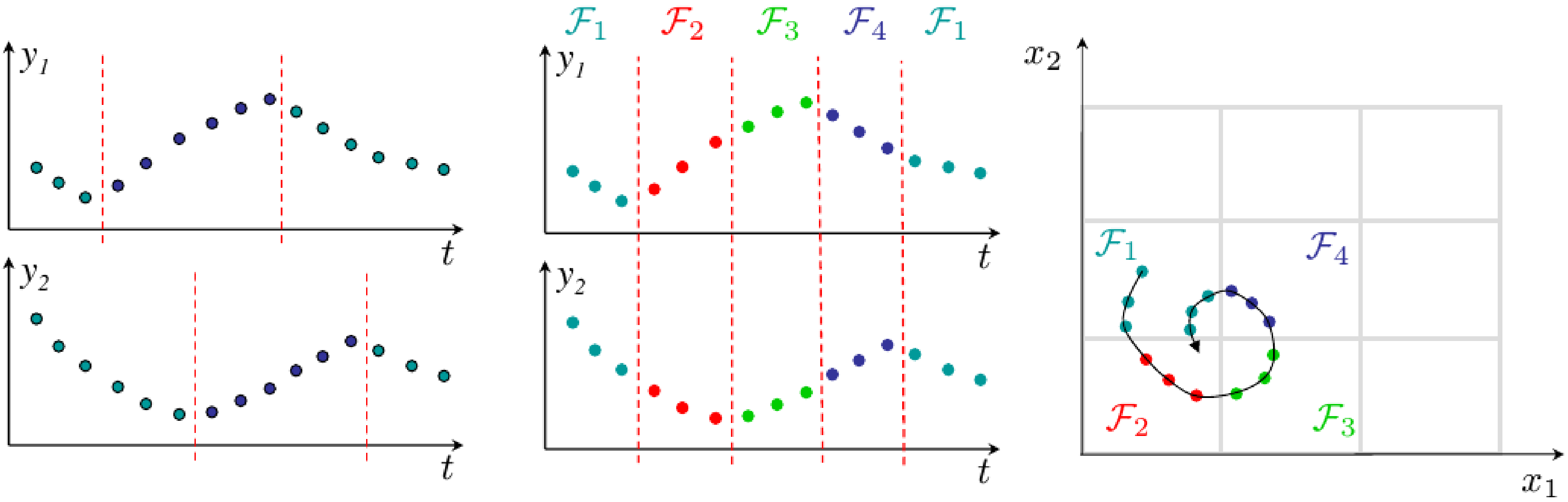
$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{cases} \begin{bmatrix}\alpha_{11}\\ \alpha_{21}\end{bmatrix} - \begin{bmatrix}\gamma_1 & 0\\ 0 & \gamma_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}, & \text{if } x \in \Delta^1\\ \begin{bmatrix}0\\ \alpha_{21}\end{bmatrix} - \begin{bmatrix}\gamma_1 & 0\\ 0 & \gamma_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}, & \text{if } x \in \Delta^2\\ \vdots & \end{cases}$$

Data segmentation and classification

- Given one time series
  - Variable sampling time
  - Extends to multiple time series

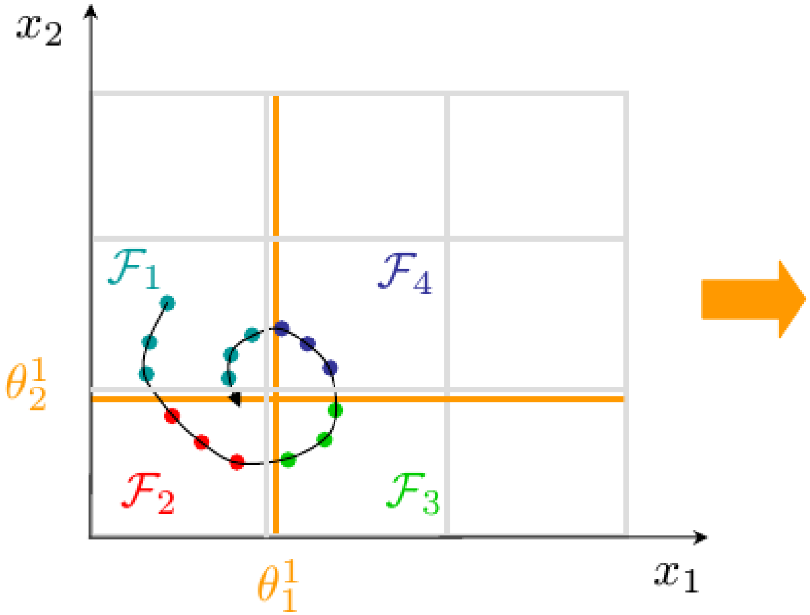
$$y_i(t_k) = x_i(t_k) + e_k, \qquad i = 1, \ldots, n$$
$$e_k \sim \mathcal{N}(0, \sigma^2) \qquad k = 1, \ldots, K$$

- Use statistical procedures to
  - Find segments with exponential behavior in each concentration profile (fit parameters and check that fitting residuals are compatible with noise)
  - Partition data into sets with the same exponential model



Threshold reconstruction

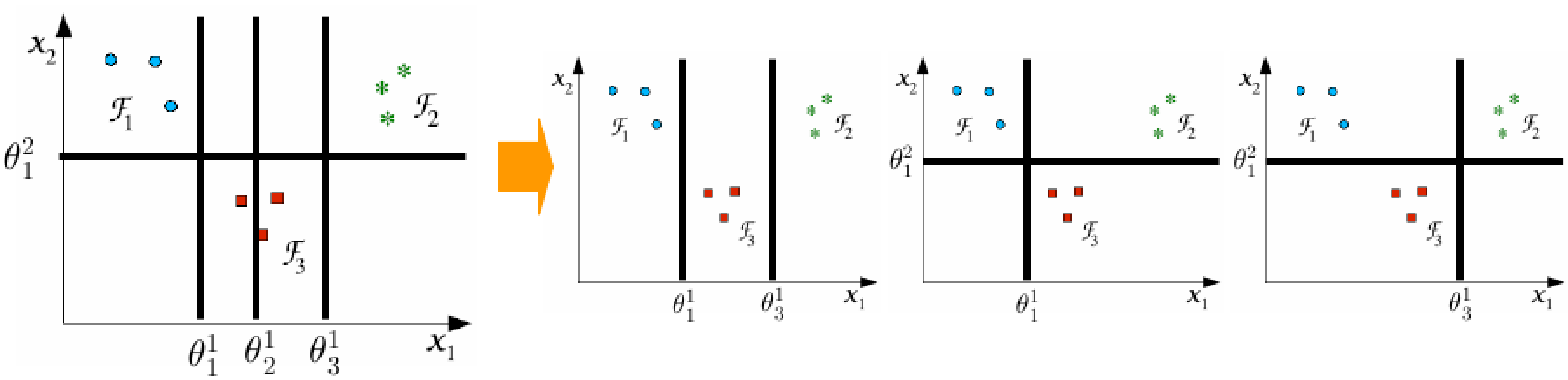
- Find *minimal* sets of thresholds that separate data clusters (multicuts)
  - Find all thresholds that separate two clusters
  - Define and exploit partial order relations among multicuts to find the minimal ones
  - Combinatorial number of multicuts: exploit branch-and-bound optimization techniques to avoid exploring all possibilities



- Two cuts
- Only one multicut = only one possible GRN

Optimal models

- Search of minimal multicuts: complexity reduction
- Identifiability issues:
  - Cannot discriminate certain models on the basis of the data (pool of equivalent models providing alternative biological hypotheses)
  - Cannot fix thresholds, only bounds can be established



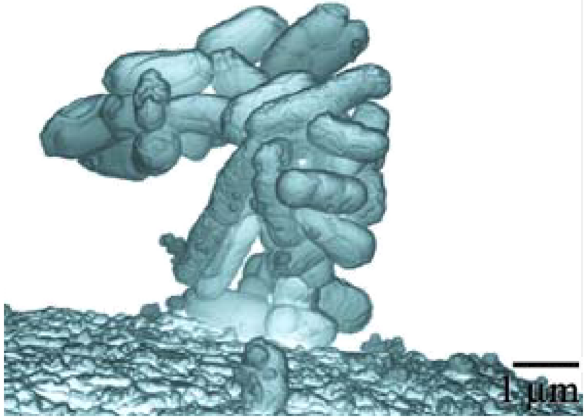
Three minimal multicuts = three possible GRNs

Example: carbon starvation in E.coli

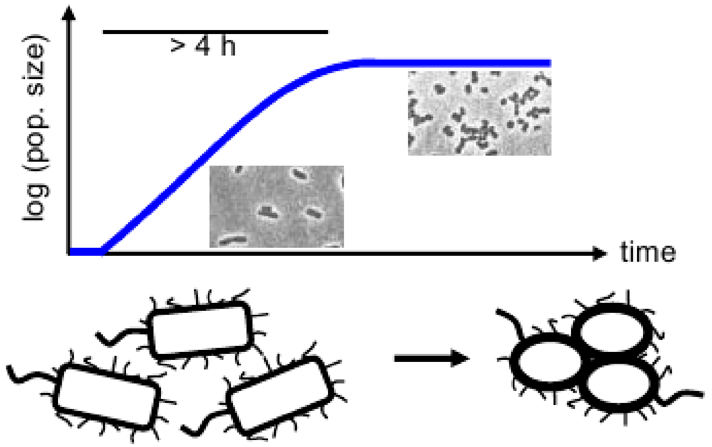
Nutritional stress → Transition from exponential to stationary phase → Changes in:

- morphology,
- metabolism,
- gene expression,
- …

Escherichia coli



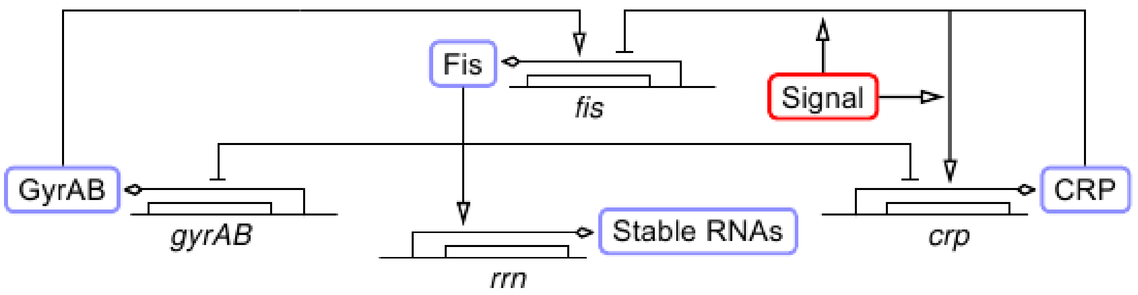
Low-temperature electron micrograph of a cluster of *E. coli* bacteria. Photo by Eric Erbe, digital colorization by Christopher Pooley, both of USDA, ARS, EMU.



Model and simulation

(Ropers et al., *Biosystems*, 2006)

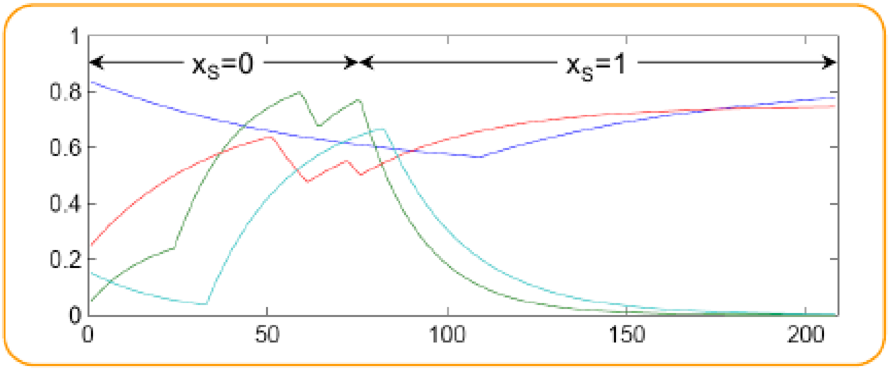
Simplified model



$$
\begin{aligned}
\dot{x}_{CRP} &= \kappa^0_{CRP} + \kappa^1_{CRP}\, s^-(x_{Fis}, \theta^1_{Fis}) \boxed{s^+(x_{CRP}, \theta^1_{CRP})} s^+(x_S, \theta_S) - \gamma_{CRP}\, x_{CRP} \\
\dot{x}_{Fis} &= \kappa^1_{Fis}\, (1 - \boxed{s^+(x_{CRP}, \theta^1_{CRP})} s^+(x_S, \theta_S)) \\
&\quad + \kappa^2_{Fis}\, s^+(x_{GyrAB}, \theta_{GyrAB})\, (1 - \boxed{s^+(x_{CRP}, \theta^1_{CRP})} s^+(x_S, \theta_S)) - \gamma_{Fis}\, x_{Fis} \\
\dot{x}_{GyrAB} &= \kappa_{GyrAB}\, s^-(x_{Fis}, \theta^3_{Fis}) - \gamma_{GyrAB}\, x_{GyrAB} \\
\dot{x}_{rrn} &= \kappa_{rrn}\, s^+(x_{Fis}, \theta^2_{Fis}) - \gamma_{rrn}\, x_{rrn}
\end{aligned}
$$

non identifiable interactions

simulation given x_0, x_S



Identification from simulated data

Cut #	Variable	Threshold value	Interaction	Correct? (Y/N)
1	CRP	0.61	activator of the synthesis of Fis	N
2	CRP	0.64	activator of the synthesis of Fis	N
3	CRP	0.71	inhibitor of the synthesis of Stable RNAs	N
4	CRP	0.74	inhibitor of the synthesis of Fis	N
5	Fis	0.09	inhibitor of the synthesis of CRP	Y
6	Fis	0.23	activator of the synthesis of Fis	N
7	Fis	0.49	activator of the synthesis of Stable RNAs	Y
8	Fis	0.75	inhibitor of the synthesis of GyrAB	Y
9	GyrAB	0.48	activator of the synthesis of Fis	N
10	GyrAB	0.50	activator of the synthesis of Fis	Y
11	GyrAB	0.55	inhibitor of the synthesis of Stable RNAs	N
12	GyrAB	0.67	activator of the synthesis of CRP	N
13	Stable RNAs	0.04	inhibitor of the synthesis of Fis, activator of the synthesis of Stable RNAs	N
14	Stable RNAs	0.18	inhibitor of the synthesis of CRP	N
15	Stable RNAs	0.53	inhibitor of the synthesis of Fis	N
16	Stable RNAs	0.55	activator of the synthesis of GyrAB	N
17	Stable RNAs	0.64	inhibitor of the synthesis of Fis, activator of the synthesis of Stable RNAs	N
18	Signal	0.50	inhibitor of the synthesis of Fis	Y

Minimal multicuts found

Multicut composed of cuts #:	Correct? (Y/N)
1, 5, 7, 8, 10	N, Y, Y, Y, Y
1, 7, 8, 10, 12	N, Y, Y, Y, N
5, 7, 8, 10, 18	Y, Y, Y, Y, Y
7, 8, 10, 12, 18	Y, Y, Y, N, Y
7, 8, 10, 14, 18	Y, Y, Y, N, Y

the best minimal multicut captures all interactions that are identifiable from the data

Models with unate structure

- Unate functions: Boolean rules monotone in each input variable
  - Transcription factors with unambiguous role (activator XOR repressor)
  - Arguably, the experimentally observable rules ? (↔ identifiability)
  - Includes most of the known gene activation rules
- Boolean-like ODE model: preserves monotonicity properties
  - Model:

$$b_i(x) = \prod_{l=1}^{n_i} \tau_l, \quad \tau_l = 1 - \prod_{j \in J_l} \left(1 - s^{\pm}(x_j)\right) \quad \text{where} \quad s^{\pm}(x_j) = \begin{cases} s^{+}(x_j), & \text{or} \\ s^{-}(x_j), \end{cases}$$

  - Sign pattern:

$$p = (p_1, \dots p_n), \qquad p_j = \begin{cases} 1, & \text{if } s^{\pm}(x_j) = s^{+}(x_j), \\ -1 & \text{if } s^{\pm}(x_j) = s^{-}(x_j), \\ 0 & \text{if } j \notin J_l \ \forall l \end{cases} \quad j = 1, \dots, n$$

Example: $p = (-1, 1)$: $s^{-}(x_1)s^{+}(x_2)$, $1 - s^{+}(x_1)s^{-}(x_2)$, $s^{-}(x_1)s^{+}(x_2) + \frac{1}{2}s^{+}(x_2)$, . . .

$b(x)$ is nondecreasing (resp. nonincreasing) in x_j if $p_j = 1$ (resp. $p_j = -1$)

. . . and so is any synthesis rate $g_i(x) = \kappa_{0,i} + \kappa_{1,i} b_i(x)$, provided $\kappa_{0,i}, \kappa_{1,i} \geq 0$

Identification via sign patterns: rationale

- Given: protein concentrations & synthesis rates (recall $\dot{x}_i = g_i(x) - \gamma_i(x)$)
- Step 1: Exploit monotonicity properties to invalidate sign patterns

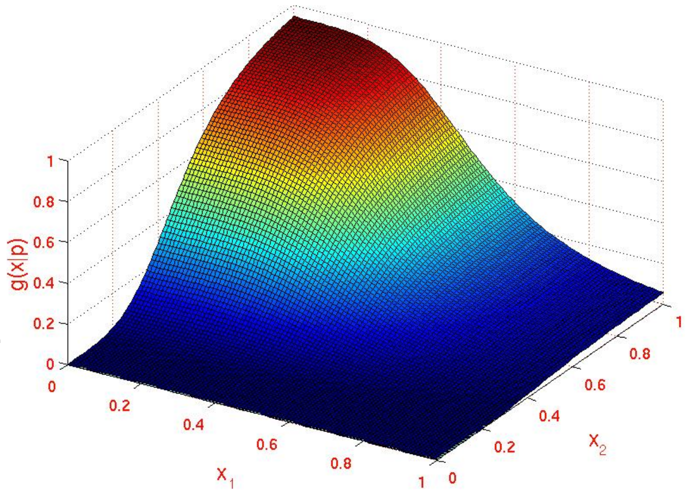
Example. $g(x|p)$, $x = (x_1, x_2)$, unknown $p = (p_1, p_2)$.

Given (x, g_i), (x', g_i') with $x_1 > x_1'$, $x_2 < x_2'$, $g_i > g_i'$.

Can exclude: $p = (-1, 1) = \left(\text{sign}(x_1' - x_1), \text{sign}(x_2' - x_2)\right)$.

Can also exclude: $p = (0, 1)$, $p = (-1, 0)$, $p = (0, 0)$.

Note: Parameter values play no role here!



- Step 2: Search best fitting model structure with valid sign pattern
  - Enumerate valid sign patterns of increasing level of complexity
  - Fit model structures with valid sign pattern to the data
    - *Parametrization of model structures S(p) with sign pattern p*
    - *Prior knowledge embedded in the definition of S(p)*
  - Evaluate fitted models based on a statistical test on the fitting errors

Sign patterns: definitions and properties

- Given data pairs: $(x^1, g^1), \ldots, (x^m, g^m)$, with $g^k = g(x^k|p)$
- Definition: p is *inconsistent* if the property

$$p_j(x_j^k - x_j^l) \geq 0,\ j = 1, \ldots, n \implies g(x^k|p) - g(x^l|p) \geq 0$$

is falsified for some k,l

- Definition: subpattern and superpattern

		Complexity

Superpatterns	1 1 -1 1
	1 1 -1 -1
	1 -1 -1 1
	1 -1 -1 -1
		4

	1 1 -1 0 1 -1 -1 0	3
Pattern	1 0 -1 0	2
Subpatterns	1 0 0 0 0 0 -1 0	1
	0 0 0 0	0

- Subpatterns of inconsistent patterns are also inconsistent
- Superpatterns of consistent patterns are also consistent
- Minimal consistent and maximal inconsistent patterns exist

Algorithm 1: original version (full data)

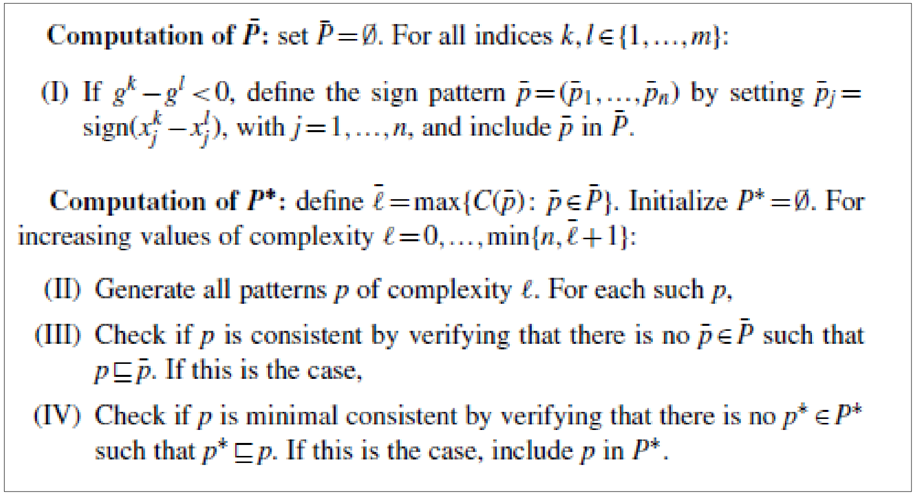
- Protein concentrations & synthesis rates
- Time-course noisy data, known variance:

$$\tilde{x}_i^k = x_i^k + e_i^k \qquad \tilde{g}_i^k = g_i^k + \epsilon_i^k$$

$$x_i^k = x_i(t_k) \qquad g_i^k = g\big(x(t_k)\big)$$

with $k = 1, \ldots, m$ and zero-mean Gaussian noise

$$v_e(x_i^k) = \mathrm{var}(e_i^k) \qquad v_\epsilon(g_i^k) = \mathrm{var}(\epsilon_i^k)$$



Computation of $\bar{P}$: set $\bar{P} = \emptyset$. For all indices $k, l \in \{1, \ldots, m\}$:

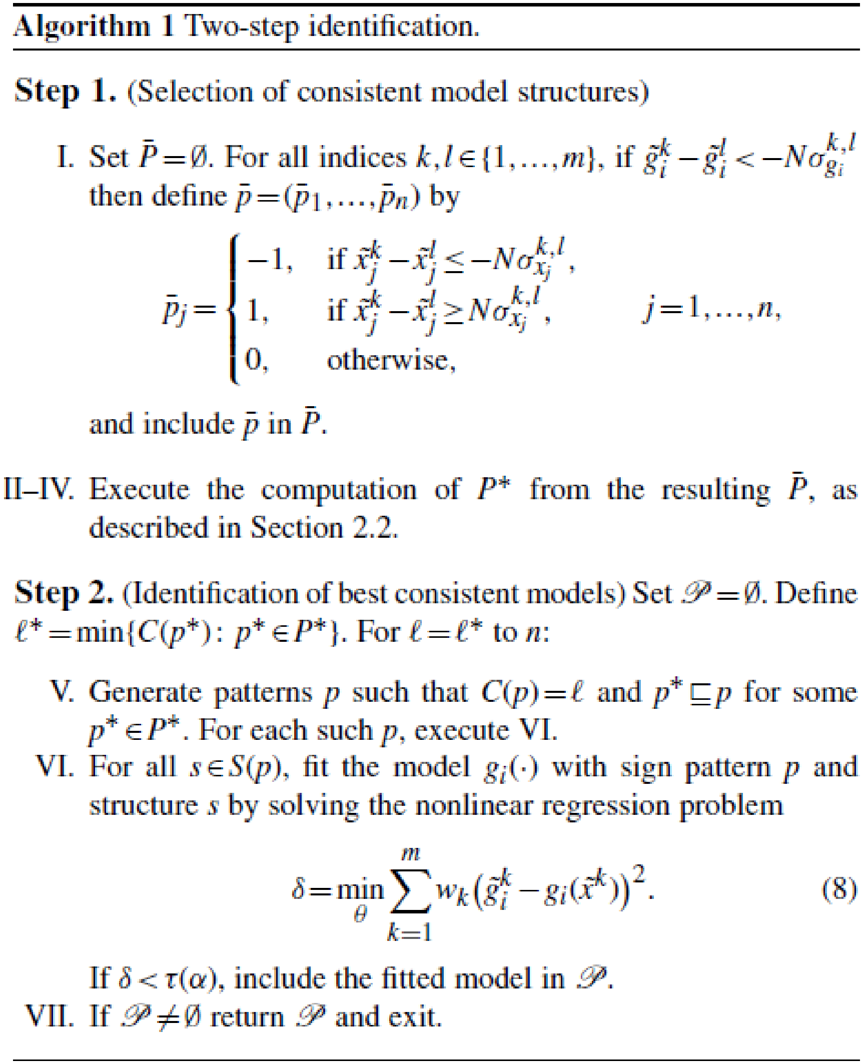
(I) If $g^k - g^l < 0$, define the sign pattern $\bar{p} = (\bar{p}_1, \ldots, \bar{p}_n)$ by setting $\bar{p}_j = \mathrm{sign}(x_j^k - x_j^l)$, with $j = 1, \ldots, n$, and include $\bar{p}$ in $\bar{P}$.

Computation of P^*: define $\bar{\ell} = \max\{C(\bar{p}) : \bar{p} \in \bar{P}\}$. Initialize $P^* = \emptyset$. For increasing values of complexity $\ell = 0, \ldots, \min\{n, \bar{\ell} + 1\}$:

(II) Generate all patterns p of complexity ℓ. For each such p,

(III) Check if p is consistent by verifying that there is no $\bar{p} \in \bar{P}$ such that $p \sqsubseteq \bar{p}$. If this is the case,

(IV) Check if p is minimal consistent by verifying that there is no $p^* \in P^*$ such that $p^* \sqsubseteq p$. If this is the case, include p in P^*.



Algorithm 1 Two-step identification.

Step 1. (Selection of consistent model structures)

I. Set $\bar{P} = \emptyset$. For all indices $k, l \in \{1, \ldots, m\}$, if $\tilde{g}_i^k - \tilde{g}_i^l < -N\sigma_{g_i}^{k,l}$ then define $\bar{p} = (\bar{p}_1, \ldots, \bar{p}_n)$ by

$$\bar{p}_j = \begin{cases} -1, & \text{if } \tilde{x}_j^k - \tilde{x}_j^l \leq -N\sigma_{x_j}^{k,l}, \\ 1, & \text{if } \tilde{x}_j^k - \tilde{x}_j^l \geq N\sigma_{x_j}^{k,l}, \\ 0, & \text{otherwise}, \end{cases} \qquad j = 1, \ldots, n,$$

and include $\bar{p}$ in $\bar{P}$.

II–IV. Execute the computation of P^* from the resulting $\bar{P}$, as described in Section 2.2.

Step 2. (Identification of best consistent models) Set $\mathscr{P} = \emptyset$. Define $\ell^* = \min\{C(p^*) : p^* \in P^*\}$. For $\ell = \ell^*$ to n:

V. Generate patterns p such that $C(p) = \ell$ and $p^* \sqsubseteq p$ for some $p^* \in P^*$. For each such p, execute VI.

VI. For all $s \in S(p)$, fit the model $g_i(\cdot)$ with sign pattern p and structure s by solving the nonlinear regression problem

$$\delta = \min_{\theta} \sum_{k=1}^{m} w_k \big(\tilde{g}_i^k - g_i(\tilde{x}^k)\big)^2. \qquad (8)$$

If $\delta < \tau(\alpha)$, include the fitted model in $\mathscr{P}$.

VII. If $\mathscr{P} \neq \emptyset$ return $\mathscr{P}$ and exit.

Comments

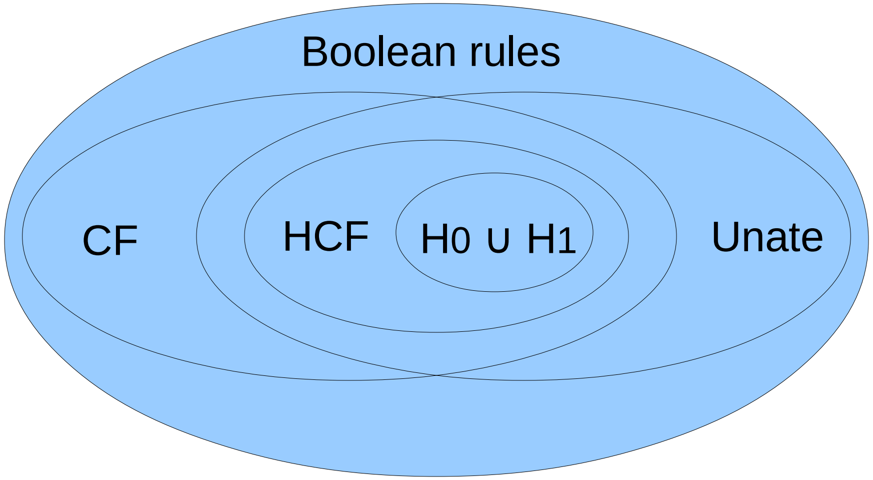
- Separate identification of regulation function of each gene
- Hierarchical search of model structures of increasing complexity
  - Stops when a good model is found (statistical test on the model residuals)
  - Favors simple over complicated models
  - Returns pool of biological alternatives
- What is a statistically good model?
  - Under the null hypothesis that the estimated model is correct, the fitting residual is distributed as $\chi^2(m)$
  - Use this property to define confidence levels (threshold on the fitting residuals) on the model estimate
- Limitations: Nonconvex parameter fitting, Data requirements

Case study: unate models with canalizing structure

- Goal: use a priori knowledge to reduce the family of network structures
- Intuition: many Boolean expression rules are unlikely/uncommon
- Evidence: (Szallasi et al 1998, Kauffman et al 2004, ...)

  out of 139 gene activation rules analyzed in (Harris et al., 2002), 99% are "Canalizing Functions", 95% are "Hierarchically Canalizing Functions", 90% are "$H_0 \cup H_1$"

  - CFs: at least one (canalizing) value of at least one (canalizing) variable determines the value of the function
  - HCFs: when the canalizing variable takes its non-canalizing value, a second variable is canalizing, etc.



We focus on H_0 U H_1

The class H0 U H1

- Class H0: $b_i(X) = X'_{j_1} \wedge X'_{j_2} \wedge \cdots \wedge X'_{j_\ell}$ $\qquad X'_{l,j} \in \{X_j, \neg X_j\}$

- Class H1: $b_i(X) = X'_{j_1} \wedge X'_{j_2} \wedge \cdots \wedge X'_{j_{\ell-2}} \wedge (X'_{j_{\ell-1}} \vee X'_{j_\ell})$

- Boolean-like ODE model with H0 U H1-structure:

$$\dot{x}_i = \kappa_i^1 + \kappa_i^2 b_i(x) - \gamma_i x_i$$

$$b_i(x) = \begin{cases} s^{\pm}(x_{j_1}) \cdot s^{\pm}(x_{j_2}) \cdots s^{\pm}(x_{j_\ell}) \\ s^{\pm}(x_{j_1}) \cdot s^{\pm}(x_{j_2}) \cdots s^{\pm}(x_{j_{\ell-2}})(1 - s^{\mp}(x_{j_{\ell-1}}) \cdot s^{\mp}(x_{j_\ell})) \end{cases}$$

Structure: $\ell, \ (j_1, j_2, \ldots, j_\ell), \ H_0 \text{ vs. } H_1$

Parameters: $\kappa_i^1, \ \kappa_i^2, \ \text{sigmoids' parameters (threshold, cooperativity)}$

Identification of H0 U H1 models

- Given concentration and synthesis rate measurements

$$y_i(t_k) = x_i(t_k)(1 + e_{i,k}) \qquad z_i(t_k) = f_i\big(x(t_k)\big)(1 + \epsilon_{i,k}) \qquad i = 1, \dots, n$$

$$e_{i,k} \sim \mathcal{N}(0, \sigma_e^2) \qquad \epsilon_{i,k} \sim \mathcal{N}(0, \sigma_\epsilon^2) \qquad k = 1, \dots, K$$

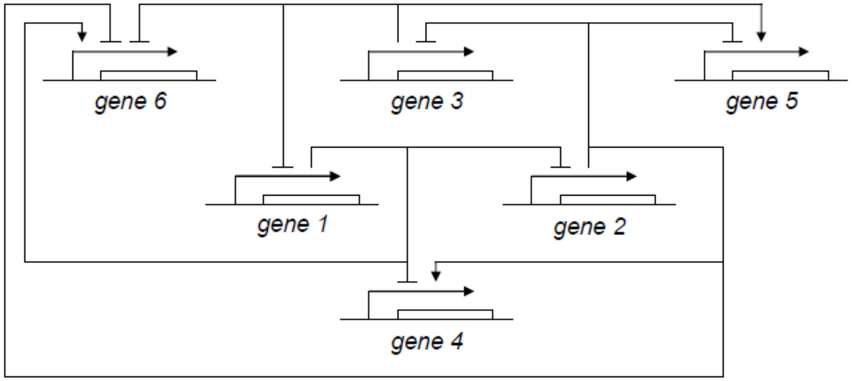
  - For known degradation rate, can compute synthesis rates from x:

$$f_i(x) = \kappa_i^1 + \kappa_i^2 b_i(x) = \dot{x}_i + \gamma_i x_i$$

(Ronen et al 2002, Brown et al 2008,...)

- Estimate
  - Structure: $\ell, \ (j_1, j_2, \dots, j_\ell), \ H_0 \text{ vs. } H_1$
  - Parameters: $\kappa_i^1, \ \kappa_i^2, \ \theta_j$ (possibly depending on i)

Test on a repressilator system



$$\dot{x}_1 = \kappa_{0,1} + \kappa_{1,1}\sigma^-(x_3) - \gamma_1 x_1,$$

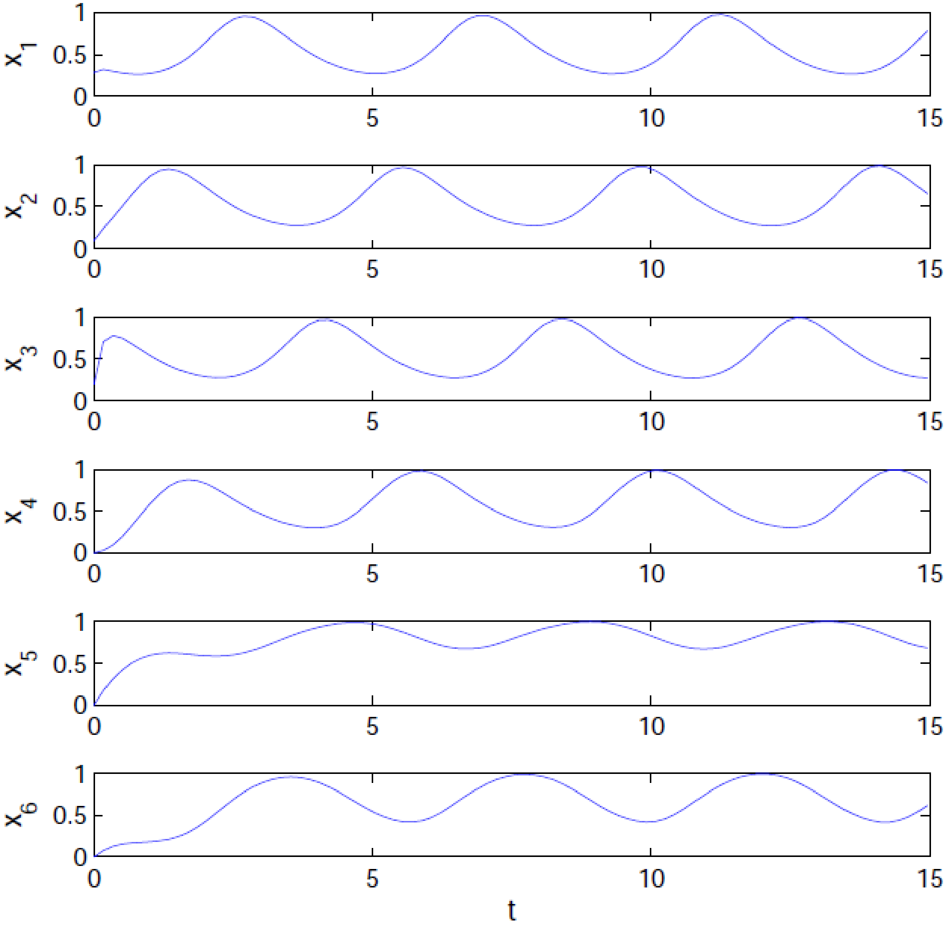
$$\dot{x}_2 = \kappa_{0,2} + \kappa_{1,2}\sigma^-(x_1) - \gamma_2 x_2,$$

$$\dot{x}_3 = \kappa_{0,3} + \kappa_{1,3}\sigma^-(x_2) - \gamma_3 x_3,$$

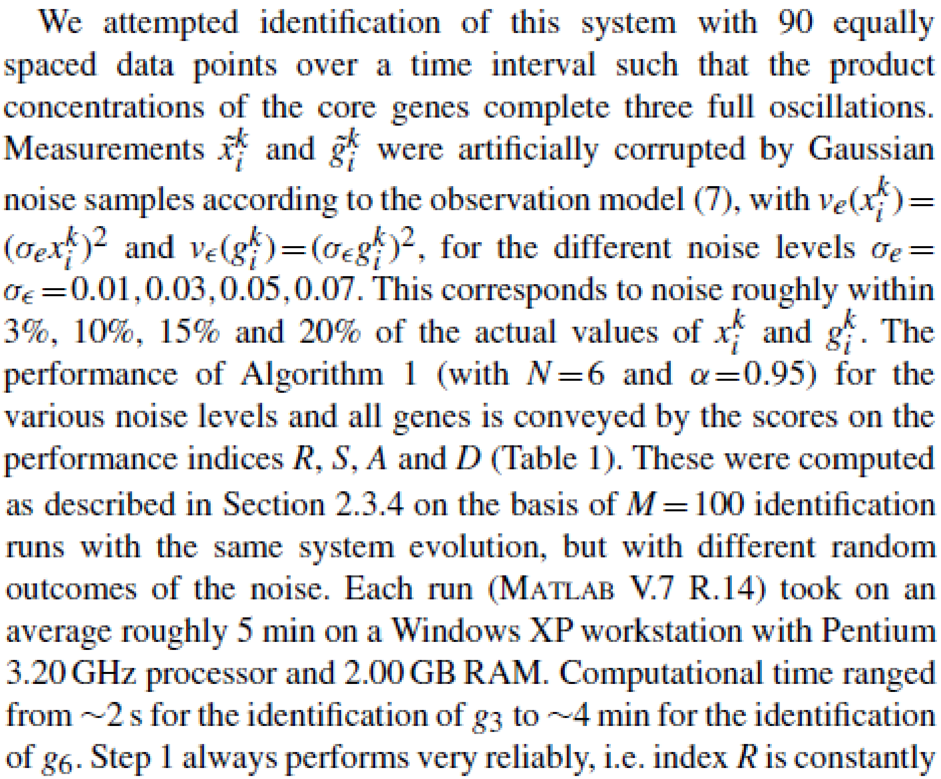
$$\dot{x}_4 = \kappa_{0,4} + \kappa_{1,4}\sigma^-(x_1)\sigma^+(x_2) - \gamma_4 x_4,$$

$$\dot{x}_5 = \kappa_{0,5} + \kappa_{1,5}[1 - \sigma^+(x_2)\sigma^-(x_3)] - \gamma_5 x_5,$$

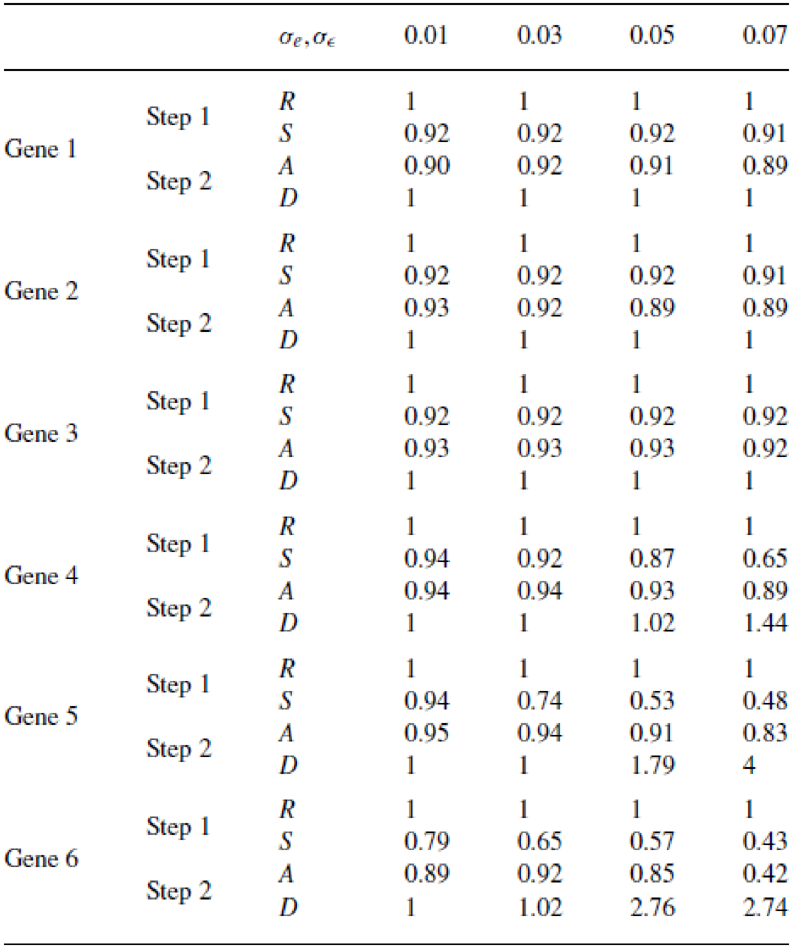
$$\dot{x}_6 = \kappa_{0,6} + \kappa_{1,6}[1 - \sigma^+(x_2)\sigma^+(x_3)]\sigma^+(x_1) - \gamma_6 x_6.$$



Performance results



We attempted identification of this system with 90 equally spaced data points over a time interval such that the product concentrations of the core genes complete three full oscillations. Measurements $\tilde{x}_i^k$ and $\tilde{g}_i^k$ were artificially corrupted by Gaussian noise samples according to the observation model (7), with $\nu_e(x_i^k)=(\sigma_e x_i^k)^2$ and $\nu_\epsilon(g_i^k)=(\sigma_\epsilon g_i^k)^2$, for the different noise levels $\sigma_e=\sigma_\epsilon=0.01, 0.03, 0.05, 0.07$. This corresponds to noise roughly within 3%, 10%, 15% and 20% of the actual values of x_i^k and g_i^k. The performance of Algorithm 1 (with $N=6$ and $\alpha=0.95$) for the various noise levels and all genes is conveyed by the scores on the performance indices R, S, A and D (Table 1). These were computed as described in Section 2.3.4 on the basis of $M=100$ identification runs with the same system evolution, but with different random outcomes of the noise. Each run (MATLAB V.7 R.14) took on an average roughly 5 min on a Windows XP workstation with Pentium 3.20 GHz processor and 2.00 GB RAM. Computational time ranged from ~2 s for the identification of g_3 to ~4 min for the identification of g_6. Step 1 always performs very reliably, i.e. index R is constantly



		$\sigma_e, \sigma_\epsilon$	0.01	0.03	0.05	0.07
Gene 1	Step 1	R	1	1	1	1
		S	0.92	0.92	0.92	0.91
	Step 2	A	0.90	0.92	0.91	0.89
		D	1	1	1	1
Gene 2	Step 1	R	1	1	1	1
		S	0.92	0.92	0.92	0.91
	Step 2	A	0.93	0.92	0.89	0.89
		D	1	1	1	1
Gene 3	Step 1	R	1	1	1	1
		S	0.92	0.92	0.92	0.92
	Step 2	A	0.93	0.93	0.93	0.92
		D	1	1	1	1
Gene 4	Step 1	R	1	1	1	1
		S	0.94	0.92	0.87	0.65
	Step 2	A	0.94	0.94	0.93	0.89
		D	1	1	1.02	1.44
Gene 5	Step 1	R	1	1	1	1
		S	0.94	0.74	0.53	0.48
	Step 2	A	0.95	0.94	0.91	0.83
		D	1	1	1.79	4
Gene 6	Step 1	R	1	1	1	1
		S	0.79	0.65	0.57	0.43
	Step 2	A	0.89	0.92	0.85	0.42
		D	1	1.02	2.76	2.74

	Index	Range	Description
Step 1	R eliability	[0,1]	Probability that the true p is deemed consistent
	S electivity	[0,1]	Percentage of sign patterns eliminated from the search in Step 2
Step 2	A ccuracy	[0,1]	Probability that the true structure is In the pool of identified models
	D ispersion	≥1	Average number of models in the pool

Simulated identification on *E.coli* model

- 6-gene carbon starvation response network
- Model in exponential growth phase
- All but third equation have H0 U H1-structure (all have unate structure)

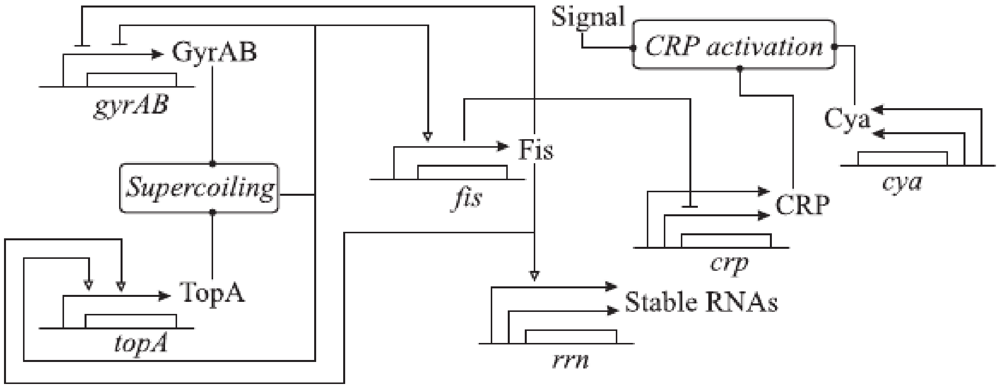


FIGURE 1. Key global regulators and regulatory interactions taking place during the transition from stationary to exponential growth phase in *E.Coli*.

(Ropers et al, Biosystems 2006)

$$\dot{x}_1 = \kappa_1^1 + \kappa_1^2 \;-\gamma_1\; x_1$$

$$\dot{x}_2 = \kappa_2^1 + \;\kappa_2^3\; \sigma^-(x_3) - \gamma_2\; x_2$$

$$\dot{x}_3 = \kappa_3^1\; \sigma^-(x_3) \;+\; \kappa_3^2\; \sigma^+(x_4)\, \sigma^-(x_5)\, \sigma^-(x_3) \;-\gamma_3\; x_3$$

$$\dot{x}_4 = \kappa_4\; (1 - \sigma^+(x_4)\, \sigma^-(x_5))\; \sigma^-(x_3) - \gamma_4\; x_4$$

$$\dot{x}_5 = \kappa_5\; \sigma^+(x_4)\, \sigma^-(x_5)\, \sigma^+(x_3) - \gamma_5\; x_5$$

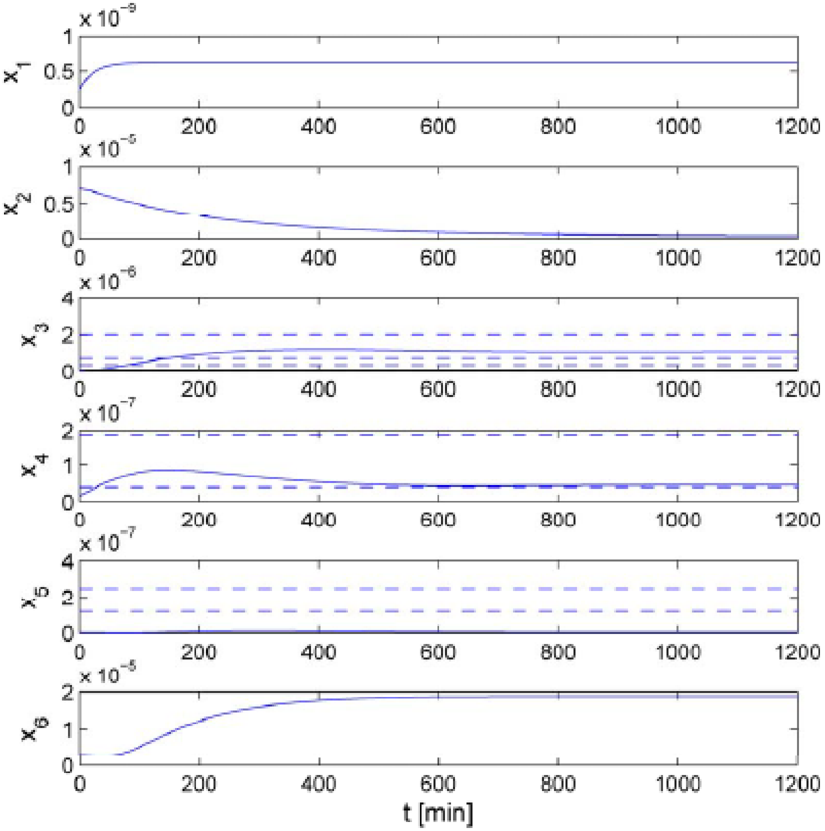
$$\dot{x}_6 = \kappa_6^1\; \sigma^+(x_3) + \kappa_6^2 - \gamma_6\; x_6$$

$x_1,\; x_2,\; x_3,\; x_4,\; x_5,\; x_6 \;=$

Cya, CRP, Fis, GyrAB, TopA

Stable RNAs

Identification scenario



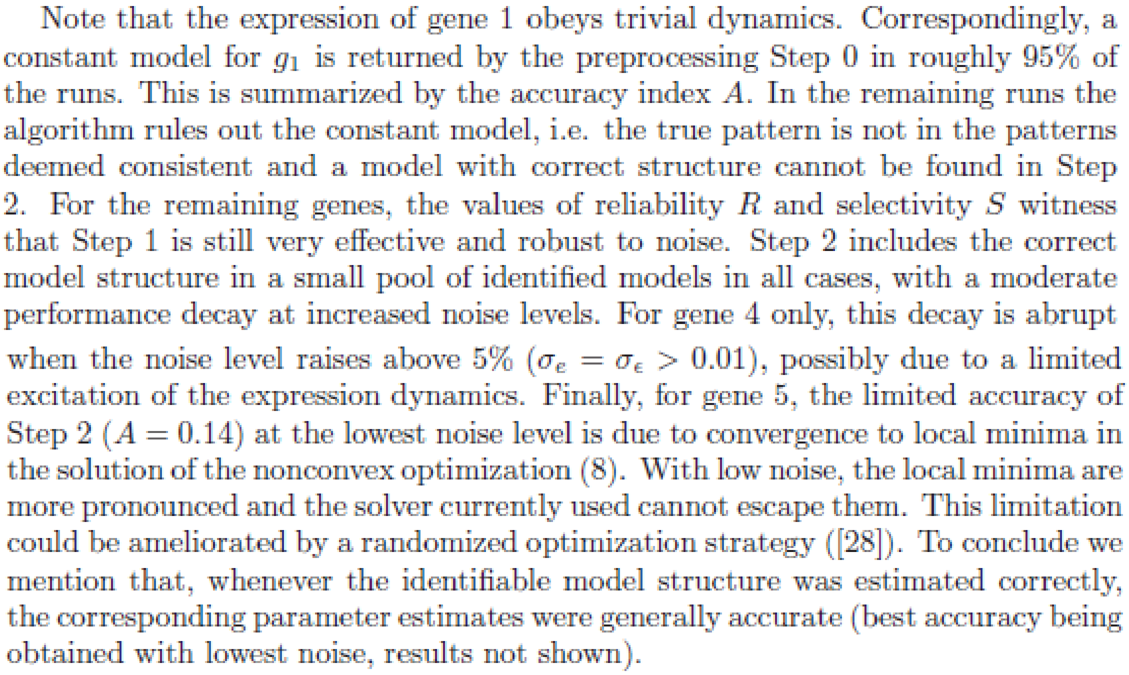
- Simulated data collected every 10 min
- Measurements over 1200 min
- Various noise levels
- Performance from 100 simulated runs
- Realistic parameters and initial cond.
- Dynamics excited in the experiment:

$$
\begin{aligned}
g_1 &= \kappa_{0,1}, & g_4 &\simeq \kappa_{1,4}\sigma^-(x_4)\sigma^-(x_3),\\
g_2 &= \kappa_{0,2} + \kappa_{1,2}\sigma^-(x_3), & g_5 &\simeq \kappa_{1,5}\sigma^+(x_4)\sigma^+(x_3),\\
g_3 &\simeq \kappa_{0,3} + \kappa_{1,3}\sigma^+(x_4)\sigma^-(x_3), & g_6 &= \kappa_{0,6} + \kappa_{1,6}\sigma^+(x_3).
\end{aligned}
$$

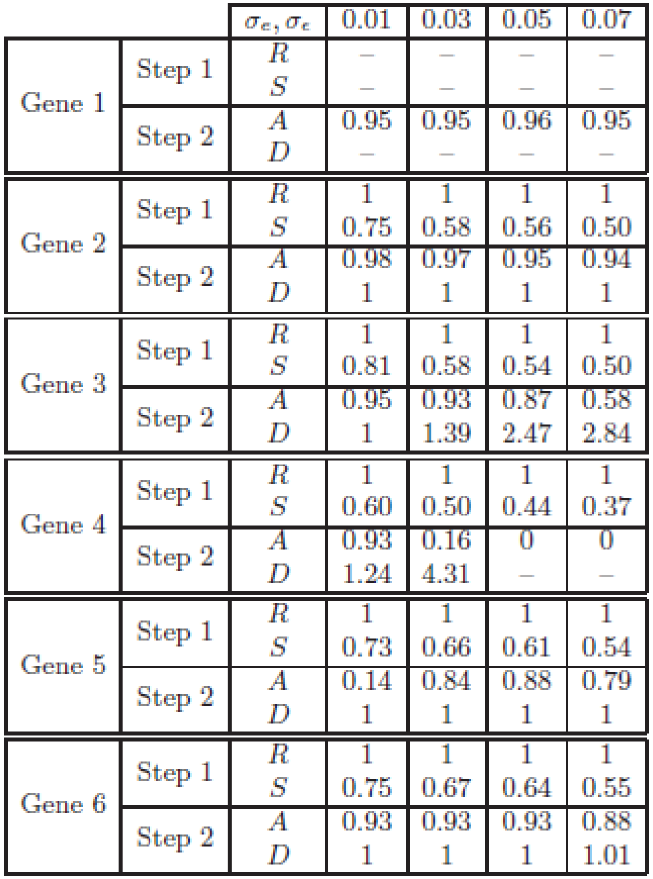
- All excited dynamics have $H_0 \cup H_1$-structure

  Use this as a “reference” model

Results on *E.coli*



Note that the expression of gene 1 obeys trivial dynamics. Correspondingly, a constant model for g_1 is returned by the preprocessing Step 0 in roughly 95% of the runs. This is summarized by the accuracy index A. In the remaining runs the algorithm rules out the constant model, i.e. the true pattern is not in the patterns deemed consistent and a model with correct structure cannot be found in Step 2. For the remaining genes, the values of reliability R and selectivity S witness that Step 1 is still very effective and robust to noise. Step 2 includes the correct model structure in a small pool of identified models in all cases, with a moderate performance decay at increased noise levels. For gene 4 only, this decay is abrupt when the noise level raises above 5% ($\sigma_\varepsilon = \sigma_\epsilon > 0.01$), possibly due to a limited excitation of the expression dynamics. Finally, for gene 5, the limited accuracy of Step 2 ($A = 0.14$) at the lowest noise level is due to convergence to local minima in the solution of the nonconvex optimization (8). With low noise, the local minima are more pronounced and the solver currently used cannot escape them. This limitation could be ameliorated by a randomized optimization strategy ([28]). To conclude we mention that, whenever the identifiable model structure was estimated correctly, the corresponding parameter estimates were generally accurate (best accuracy being obtained with lowest noise, results not shown).



		$\sigma_\varepsilon, \sigma_\epsilon$	0.01	0.03	0.05	0.07
Gene 1	Step 1	R	–	–	–	–
		S	–	–	–	–
	Step 2	A	0.95	0.95	0.96	0.95
		D	–	–	–	–
Gene 2	Step 1	R	1	1	1	1
		S	0.75	0.58	0.56	0.50
	Step 2	A	0.98	0.97	0.95	0.94
		D	1	1	1	1
Gene 3	Step 1	R	1	1	1	1
		S	0.81	0.58	0.54	0.50
	Step 2	A	0.95	0.93	0.87	0.58
		D	1	1.39	2.47	2.84
Gene 4	Step 1	R	1	1	1	1
		S	0.60	0.50	0.44	0.37
	Step 2	A	0.93	0.16	0	0
		D	1.24	4.31	–	–
Gene 5	Step 1	R	1	1	1	1
		S	0.73	0.66	0.61	0.54
	Step 2	A	0.14	0.84	0.88	0.79
		D	1	1	1	1
Gene 6	Step 1	R	1	1	1	1
		S	0.75	0.67	0.64	0.55
	Step 2	A	0.93	0.93	0.93	0.88
		D	1	1	1	1.01

Algorithm 2: extension to partial data

- Assuming only protein concentrations are available:
  1. Reconstruct missing information (synthesis rates, variances)
  2. Apply Algorithm 1 (unchanged)
- Option 1: Deconvolution

$$\dot{x}_i(t) = -\gamma_i x_i(t) + g_i(t), \quad g_i(t) = \kappa_{0,i} + \kappa_{1,i} b_i\big(x(t)\big) \text{ is a forcing input}$$

  - Well established (Bayesian) methods for regularized estimates
  - Severe over- and under-smoothing observed in practice
- Option 2 (our choice): Data fitting + Bootstrapping

Choose basis functions for $x_i(\cdot)$, e.g. cubic splines

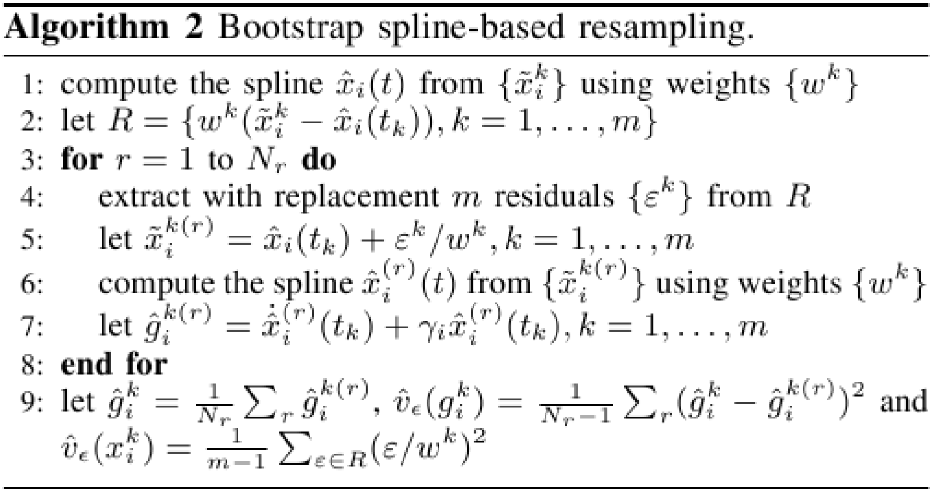
Compute estimate $\hat{x}_i$ by fitting data $\tilde{x}_i^k$, and $\hat{\dot{x}}_i = \dot{\hat{x}}_i$ by explicit differentiation

Reconstruct the synthesis rates $\tilde{g}_i^k = \dot{\hat{x}}_i(t_k) + \gamma_i \tilde{x}_i^k$

Utilize the fitting errors $\tilde{x}_i^k - \hat{x}_i(t_k)$ to reproduce the noise statistics

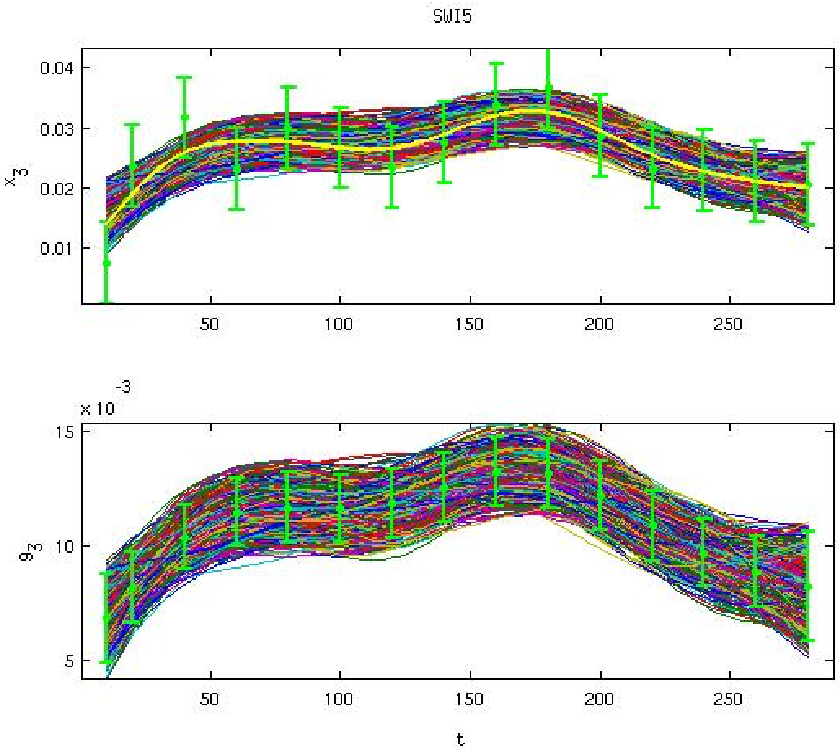
Residual resampling

- Randomized procedure to infer statistics of any functional of the regression curve
- Applicable to any type of regression curve (But sensitive to this choice!)
- Our implementation computes statistics of protein concentration and synthesis rate measurements from a single protein concentration dataset.



Algorithm 2 Bootstrap spline-based resampling.

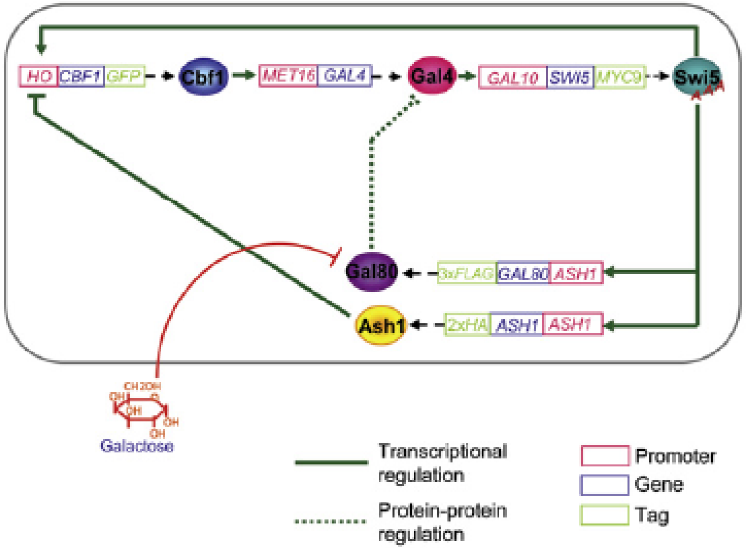
```
compute the spline $\hat{x}_i(t)$ from $\{\tilde{x}_i^k\}$ using weights $\{w^k\}$
let $R = \{w^k(\tilde{x}_i^k - \hat{x}_i(t_k)), k = 1, \ldots, m\}$
**for** $r = 1$ to $N_r$ **do**
    extract with replacement $m$ residuals $\{\varepsilon^k\}$ from $R$
    let $\tilde{x}_i^{k(r)} = \hat{x}_i(t_k) + \varepsilon^k / w^k, k = 1, \ldots, m$
    compute the spline $\hat{x}_i^{(r)}(t)$ from $\{\tilde{x}_i^{k(r)}\}$ using weights $\{w^k\}$
    let $\hat{g}_i^{k(r)} = \dot{\hat{x}}_i^{(r)}(t_k) + \gamma_i \hat{x}_i^{(r)}(t_k), k = 1, \ldots, m$
**end for**
let $\hat{g}_i^k = \frac{1}{N_r} \sum_r \hat{g}_i^{k(r)}$, $\hat{v}_\epsilon(g_i^k) = \frac{1}{N_r - 1} \sum_r (\hat{g}_i^k - \hat{g}_i^{k(r)})^2$ and
$\hat{v}_\epsilon(x_i^k) = \frac{1}{m-1} \sum_{\varepsilon \in R} (\varepsilon / w^k)^2$
```

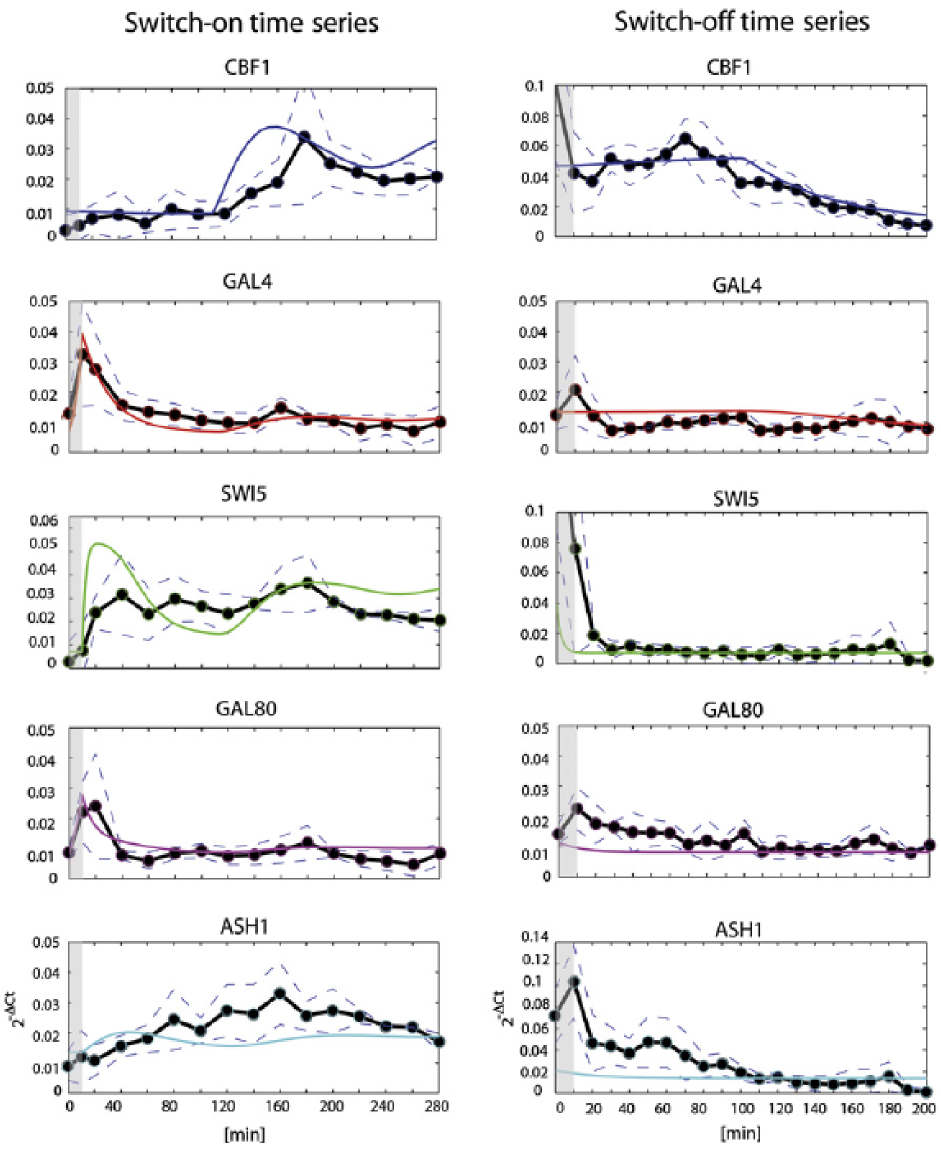


Experiment on IRMA

Synthetic gene network

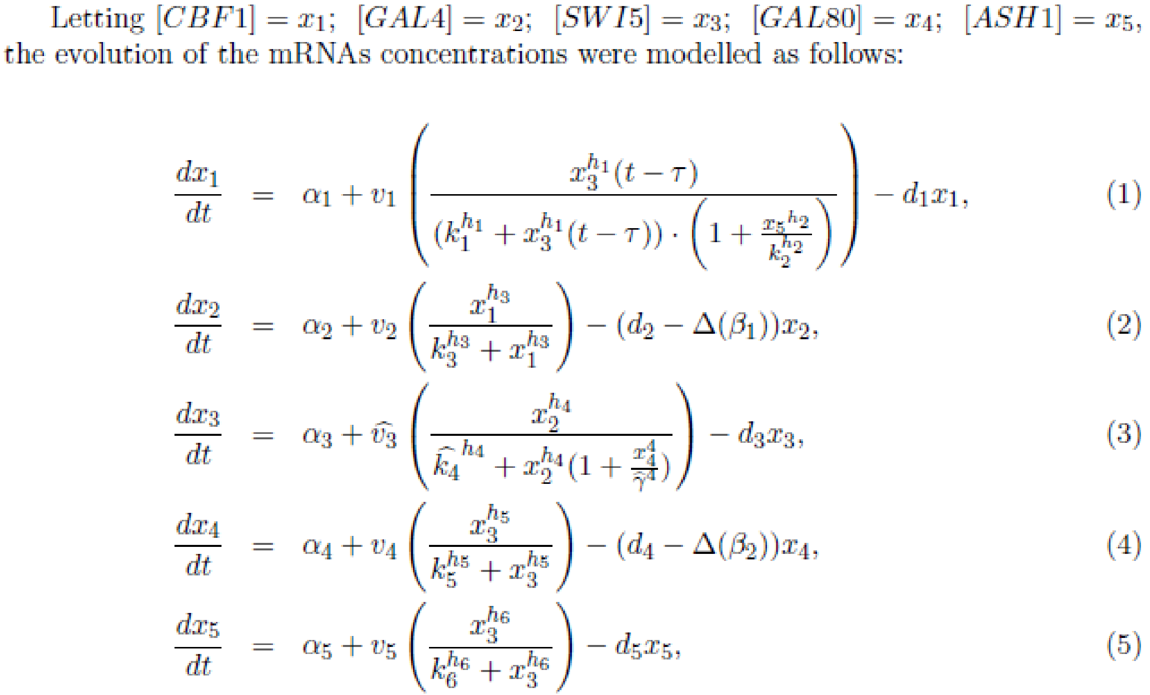
in Yeast (Cantone et al., Cell 2009)





Mathematical model

(Cantone *et al.*, Cell 2009)



Letting $[CBF1] = x_1$; $[GAL4] = x_2$; $[SWI5] = x_3$; $[GAL80] = x_4$; $[ASH1] = x_5$, the evolution of the mRNAs concentrations were modelled as follows:

$$\frac{dx_1}{dt} = \alpha_1 + v_1 \left(\frac{x_3^{h_1}(t-\tau)}{(k_1^{h_1} + x_3^{h_1}(t-\tau)) \cdot \left(1 + \frac{x_5{}^{h_2}}{k_2^{h_2}}\right)} \right) - d_1 x_1, \quad (1)$$

$$\frac{dx_2}{dt} = \alpha_2 + v_2 \left(\frac{x_1^{h_3}}{k_3^{h_3} + x_1^{h_3}} \right) - (d_2 - \Delta(\beta_1))x_2, \quad (2)$$

$$\frac{dx_3}{dt} = \alpha_3 + \widehat{v_3} \left(\frac{x_2^{h_4}}{\widehat{k_4}^{\,h_4} + x_2^{h_4}(1 + \frac{x_4^4}{\widehat{\gamma}^4})} \right) - d_3 x_3, \quad (3)$$

$$\frac{dx_4}{dt} = \alpha_4 + v_4 \left(\frac{x_3^{h_5}}{k_5^{h_5} + x_3^{h_5}} \right) - (d_4 - \Delta(\beta_2))x_4, \quad (4)$$

$$\frac{dx_5}{dt} = \alpha_5 + v_5 \left(\frac{x_3^{h_6}}{k_6^{h_6} + x_3^{h_6}} \right) - d_5 x_5, \quad (5)$$

- We attempt identification in the class of models with H_0 U H_1-structure
  - Different but similar analytical form
  - Test for flexibility of the approach
  - Known delays can be accounted for

Results: full data

- Comparison with TSNI (Cantone *et al.*, Cell 2009)
- True protein concentrations (very few data points)
- Rates simulated from the model ("what-if" performance test)
- Evaluation of network reconstruction performance, but not of parameter fit
- PPV=TD/TD+FD and Se=TD/TD+FU (T=True, D=Detected, U=Undetected edges)

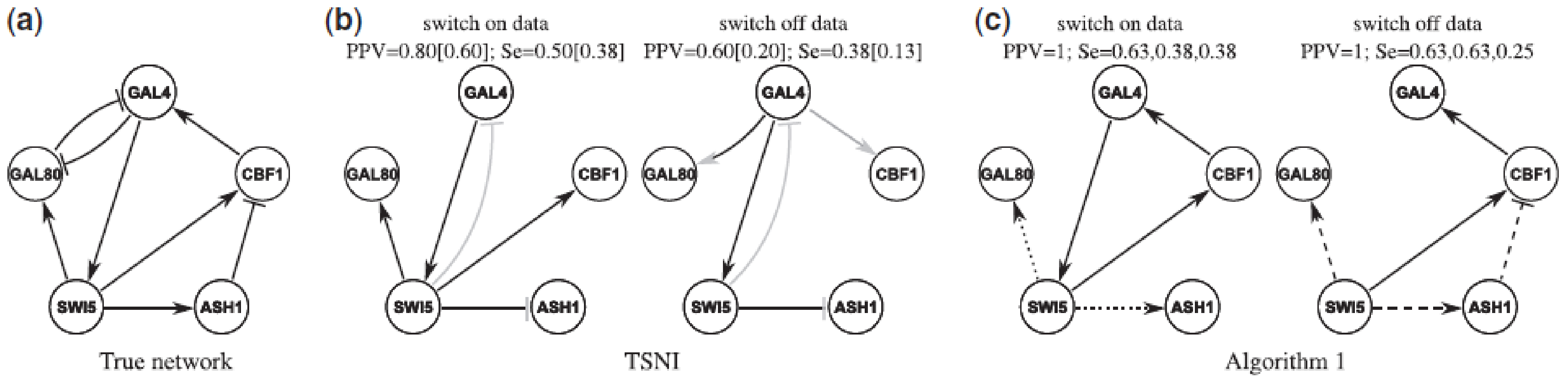
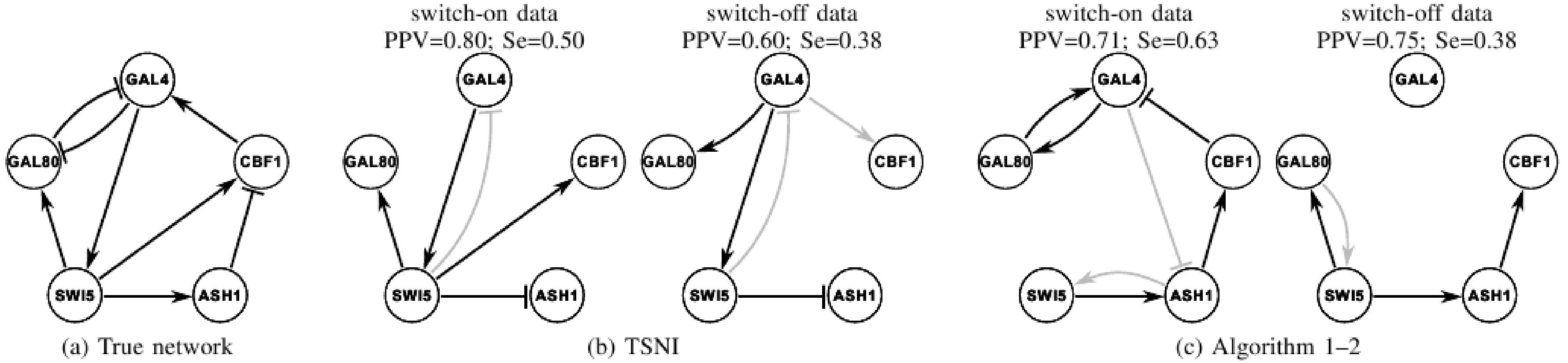


Fig. 1. (a) True network of interactions in IRMA. Results obtained by (b) the TSNI algorithm (Cantone *et al.*, 2009) and by (c) Algorithm 1. Grey arcs (respectively, grey-end markers) denote incorrect direction (respectively, sign) of the inferred interactions. Values of PPV and Se for the signed directed graph, when different from the unsigned case, appear in square brackets. The three values of Se in (c) refer to increasing noise levels, while dashed and dotted arcs denote interactions inferred only for $\sigma_\epsilon < 0.3$ and $\sigma_\epsilon < 0.1$, respectively.

Porreca *et al,* Bioinformatics 2010

Results: partial data

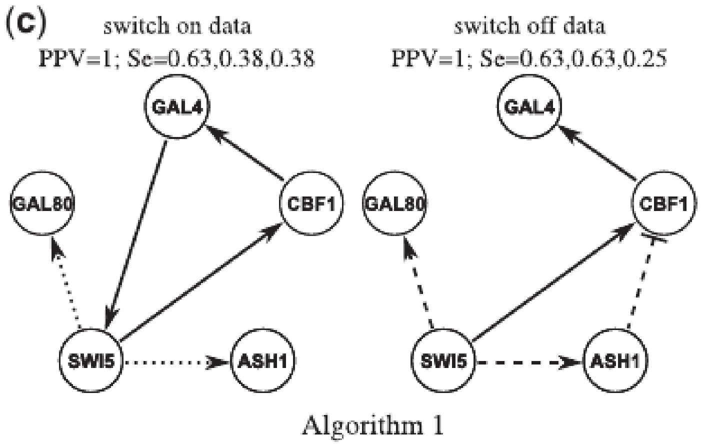


(a) True network (b) TSNI (c) Algorithm 1–2

Fig. 1: (a) True network of interactions in IRMA. Results obtained by (b) the TSNI algorithm [27] and by (c) Algorithms 1 and 2. Gray edges denote incorrect direction of the inferred interactions.

- Additional assumptions (no self-regulation)
- Loss of accuracy
  - Parameter estimates (when applicable, not shown)
  - Sign of interaction (possibly due to low data quality)
  - Direction of regulation (bad!)
- Still better than TSNI...

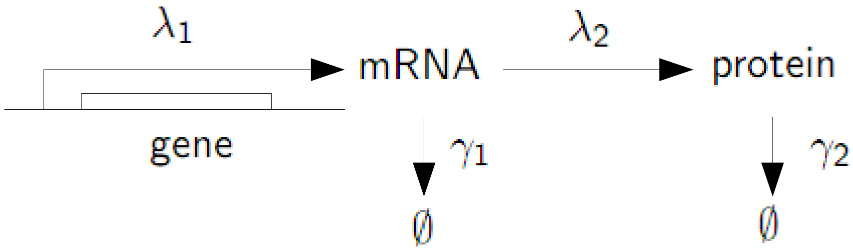
To be compared with...



Identification of stochastic models: A quick view

Introduction: stochastic gene expression

- At the cell level, protein synthesis depends on *random* events
  - Binding/unbinding of activators/repressors and RNApol to DNA, ...
  - Environmental conditions (temperature, availability of free RNAP,...)
- Classical stochastic gene expression model:
  - Describes the formation and degradation of single molecules
  - Time resolution, no spatial resolution (homogeneous reaction volume)

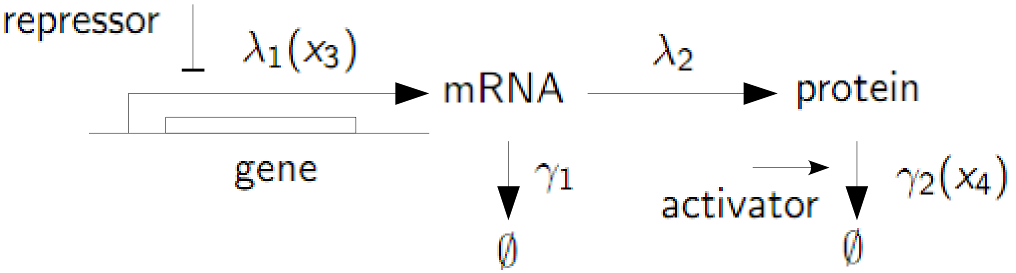


$$\begin{aligned}
\mathbf{x}_1 &= \text{number of mRNA molecules}\\
\mathbf{x}_2 &= \text{number of protein molecules}\\
\lambda_1, \lambda_2 &= \text{prob. of molecule formation per unit time}\\
\gamma_1, \gamma_2 &= \text{prob. of molecule degradation per unit time}
\end{aligned}$$

$$\begin{aligned}
p(\mathbf{x}_1 = x_1; t+\delta) = {} & p(\mathbf{x}_1 = x_1 - 1; t)\cdot\lambda_1\delta + p(\mathbf{x}_1 = x_1 + 1; t)\cdot\gamma_1\delta + p(\mathbf{x}_1 = x_1; t)\cdot\lambda_2\delta +\\
& p(\mathbf{x}_1 = x_1; t)\cdot(1 - \lambda_1\delta - \gamma_1\delta - \lambda_2\delta)\\
p(\mathbf{x}_2 = x_2; t+\delta) = {} & p(\mathbf{x}_2 = x_2 - 1; t)\cdot\lambda_2\delta + p(\mathbf{x}_2 = x_2 + 1; t)\cdot\gamma_2\delta + p(\mathbf{x}_2 = x_2; t)\cdot(1 - \lambda_2\delta - \gamma_2\delta)
\end{aligned}$$

Regulation and noise

- Example: regulated gene expression and protein degradation



$$x_3 = \text{number of repressor molecules}$$

$$x_4 = \text{number of activator molecules}$$

$$\lambda_1(x_3) = \kappa_1 \cdot 1/(1 + x_3) \ (e.g.)$$

$$\gamma_2(x_4) = \kappa_2 \cdot x_4/(1 + x_4) \ (e.g.)$$

- This modelling framework describes the random nature of the events *internal* to the gene expression mechanism (*intrinsic noise*)
- Random fluctuations of the event rates, due to changes *external* to the gene expression mechanism, are not modelled (*extrinsic noise*)

[Many contributors: Paulsson, Elowitz, Alon, Arkin, ...]

Network modeling: Chemical Master Equation

- Generalization of the stochastic modelling framework seen before to any biochemical (regulatory) network

$$\dot{p}(\mathbf{x};t) = \sum_{\mu=1}^{M} p(\mathbf{x}-s_\mu;t)a_\mu(\mathbf{x}-s_\mu) - p(\mathbf{x};t)\cdot\sum_{\mu=1}^{M} a_\mu(\mathbf{x})$$

$\mathbf{x}$ = random vector of the number of molecules of every species, one per entry

μ = reaction index (from 1 to M possible reactions)

s_μ = state change associated to the μ-th reaction

a_μ = propensity (prob. per unit time) of μ-th reaction (state dependent)

- Infinite-dimensional linear equation in the probabilities p
- No closed-form (exact) solution except in trivial cases
- Finite complexity approximations exist ([Gillespie, Khammash, …])

Experimental observation of empirical probabilities

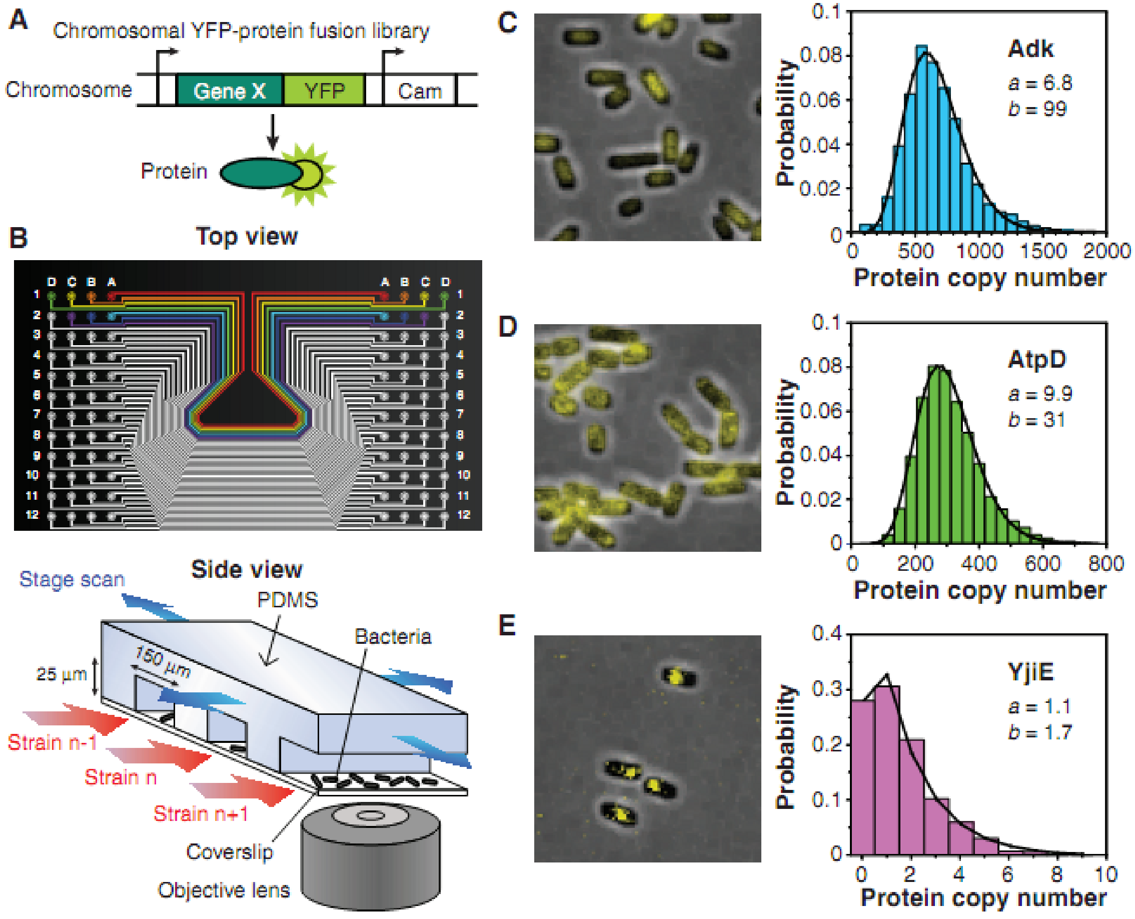


Fig. 1. Quantitative imaging of a YFP-fusion library. (**A**) Each library strain has a YFP translationally fused to the C terminus of a protein in its native chromosomal position. (**B**) A poly(dimethylsiloxane) (PDMS) microfluidic chip is used for imaging 96 library strains. *E. coli* cells of each strain are injected into separate lanes and immobilized on a polylysine-coated coverslip for automated fluorescence imaging with single-molecule sensitivity. (**C** to **E**) Representative fluorescence images overlaid on phase-contrast images of three library strains, with respective single-cell–protein level histograms that are fit to gamma distributions with parameters *a* and *b*. Protein levels are determined by deconvolution (*18*). The protein copy number per average cell volume, or the concentration, was determined as described in the main text and the SOM (*18*). (C) The cytoplasmic protein Adk uniformly distributed intracellularly. (D) The membrane protein AtpD distributed on the cell periphery. (E) The predicted DNA-binding protein YjiE with clear intercellular localization. Single YjiE-YFPs can be visualized because they are localized. Note that, unlike (C) and (D), the gamma distribution asymmetrically peaks near zero if *a* is close to or less than unity.

[Taniguchi *et al.*, Science 329, 533 (2010)]

Identification: Fitting empirical probabilities

- Assumption: the network structure is known
- Let θ be model parameters to be determined (e.g. rate constants)
- Data: Gene expression histograms y_k at times t_k, k=1,...N
  - Fluorescence assumed proportional to the number of reporter molecules
- Model predictions: Computation of p(x;t|θ)
  - Exact solution, if available (basically never...)
  - Approximate solutions: e.g. by simulation (Stochastic Simulation Algorithm, a.k.a. Gillespie algorithm, and the like) or by analytical approximation (e.g. the Finite State Projection (FSP) method, see next)
- Parameter identification: Fit model to data by solving

$$\hat{\theta} = \arg\min_{\theta} \sum_{k=1}^{N} ||y_k - p^{obs}(t_k|\theta)||$$

where $p^{obs}(t|\theta)$ is a function of p(•;t|θ) fixed by the experimental setup

  - e.g. marginal probabilities of the (subset of) observed gene(s)

The Finite State Projection method

- Analytical approximation of the Chemical Master Equation
  - Method: Munsky and Khammash, J. Chem. Phys 124 (2006)
  - Use in identification: Munsky et al, Mol Syst Biol 5:318 (2009)
- Guarantee of achieving desired approximation accuracy
- Of course, better precision implies increasing computational cost
- Basic idea: Out of all possible network states, restrict computations to a finite set of most probable network states
  - Practical for systems that traverse a reasonably small set of states with high probability
- For the math and the details of the algorithm, let's go through:

M.Khammash, “The Chemical Master Equation in Gene Networks: Complexity and Approaches”

http://www.cds.caltech.edu/~murray/wiki/images/d/d9/Khammash_master-15aug06.pdf

pages 26-33.

Identification: Other methods

- Moment matching: [e.g. work by J.Hespana]
    - Instead of probabilities, consider vector of all moments z and a truncation z*

    $$z(t) = \begin{bmatrix} Ex(t) & Ex(t)^T x(t) & \dots \end{bmatrix}^T, \qquad z^*(t) = \begin{bmatrix} Ex(t) & Ex(t)^T x(t) \end{bmatrix}^T$$

    evolving according to the equations depending on the model parameters

    $$\dot{z}(t) = Bz(t), \qquad \dot{z}^*(t) \simeq B^* z^*(t)$$

    and fit the equation for z* to the corresp. empirical statistics from many cells
- At stochastic steady state: [Taniguchi *et al.*, Science 329, 533 (2010)]
    - System evolves until stochastic equilibrium where p does not change
    - Use asymptotic approximation with a Gamma distribution

    $$p(x; t) \to d(x) \text{ for } t \to +\infty$$

    to fit (combinations of the) model parameters

Discussion

- Evidence for fundamental role of intrinsic and extrinsic noise

  (e.g. Elowitz et al, *Science*, 2002)

- Identification of stochastic models of genetic networks being developed and applied

  (Munsky, Khammash et al 2009, Zechner *et al* 2012)

- Great interest for near-future experimental techniques

Conclusions

- Masses of data wait for being processed. Automated processing unavoidable
- Modern experimental techniques enable inference of quantitative dynamic models at population and (sometimes) single cell level, even more to come
- Numerous applications in medicine, (bio)chemical industry etc.
- A lot of work in progress for model identification methods
- Intriguing mathematical problems
- Nonstandard identification problems: a lot to use, a lot to invent
- Exciting interdisciplinary activity
- Opportunities for internships & research projects !

... Thank you!

eugenio.cinquemani@inria.fr

Appendix: Principal Component Analysis

- Singular Value Decomposition (SVD) of a matrix

$$M \in \mathbb{R}^{p \times q}, \quad p < q$$

$$M = USV^T = \begin{bmatrix} U_1 & \cdots & U_p \end{bmatrix} \left[\begin{array}{ccc|c} s_1 & & & \\ & \ddots & & 0 \\ & & s_p & \end{array}\right] \begin{bmatrix} V_1^T \\ \vdots \\ V_q^T \end{bmatrix}$$

$U, \; V$ orthogonal matrices, $s_1 \geq s_2 \geq \ldots \geq s_p \geq 0$

- PCA principle: eliminate contributions from smallest singular values

$$M = \sum_{i=1}^{p} s_i U_i V_i^T \simeq \sum_{i=1}^{r} s_i U_i V_i^T, \quad r < p$$

- i=1 , ... , r are called the principal components of M

PCA in linear regression

- Problem: find combination H of rows of M that is closest to Y^+:

$$\text{minimize} \quad \|Y^+ - HM\|, \qquad H = [A^d \ B^d], \quad M = \begin{bmatrix} Y^- \\ U \end{bmatrix}$$

- Idea: exploit PCA to project Y^+ on the approximate row space of M
- Define:

$$H = Y^+ V S^\dagger U^T, \quad S^\dagger = \left[\begin{array}{cccccc} s_1^{-1} & & & & & \\ & \ddots & & & & \\ & & s_r^{-1} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \\ \hline & & & 0 & & \end{array}\right] \quad \text{s.t.} \quad S^\dagger S = \left[\begin{array}{cccccc|c} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & 0 & & & 0 \\ & & & & \ddots & & \\ & & & & & 0 & \\ \hline & & & 0 & & & \end{array}\right]$$

Then:

$$H \cdot M = (Y^+ V S^\dagger U^T) \cdot (U S V^T) = Y^+ V S^\dagger S V^T = \sum_{i=1}^{r} (Y^+ \cdot V_i) \cdot V_i^T$$

- Low-rank solution, elimination of noise (non-principal components)