Mathematical Models for Synthetic Biology

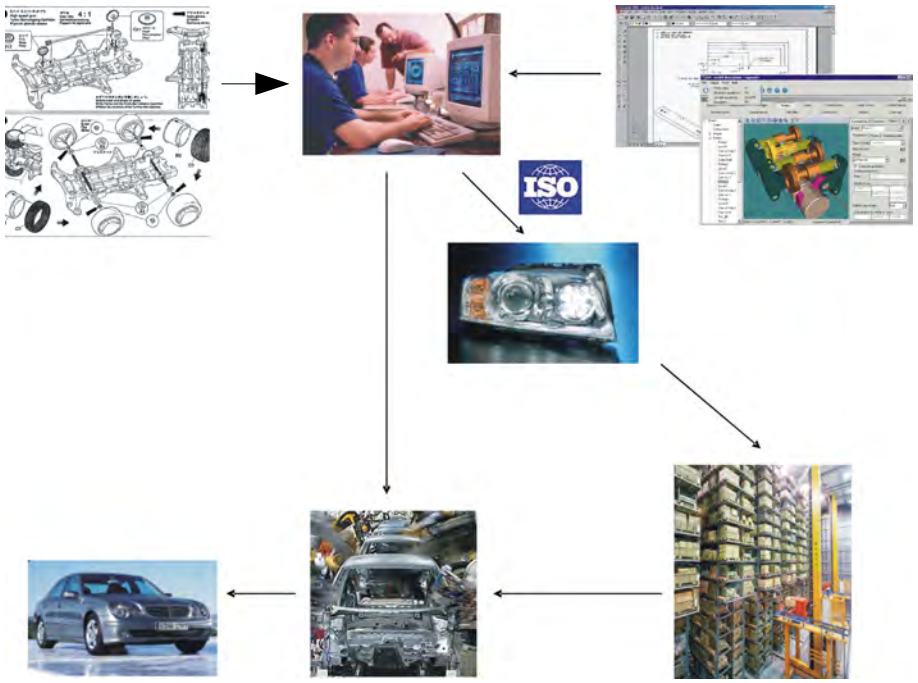
Diego di Bernardo, TIGEM Naples / I,
dibernardo@tigem.it

Jörg Stelling, ETH Zurich / CH,
joerg.stelling@inf.ethz.ch

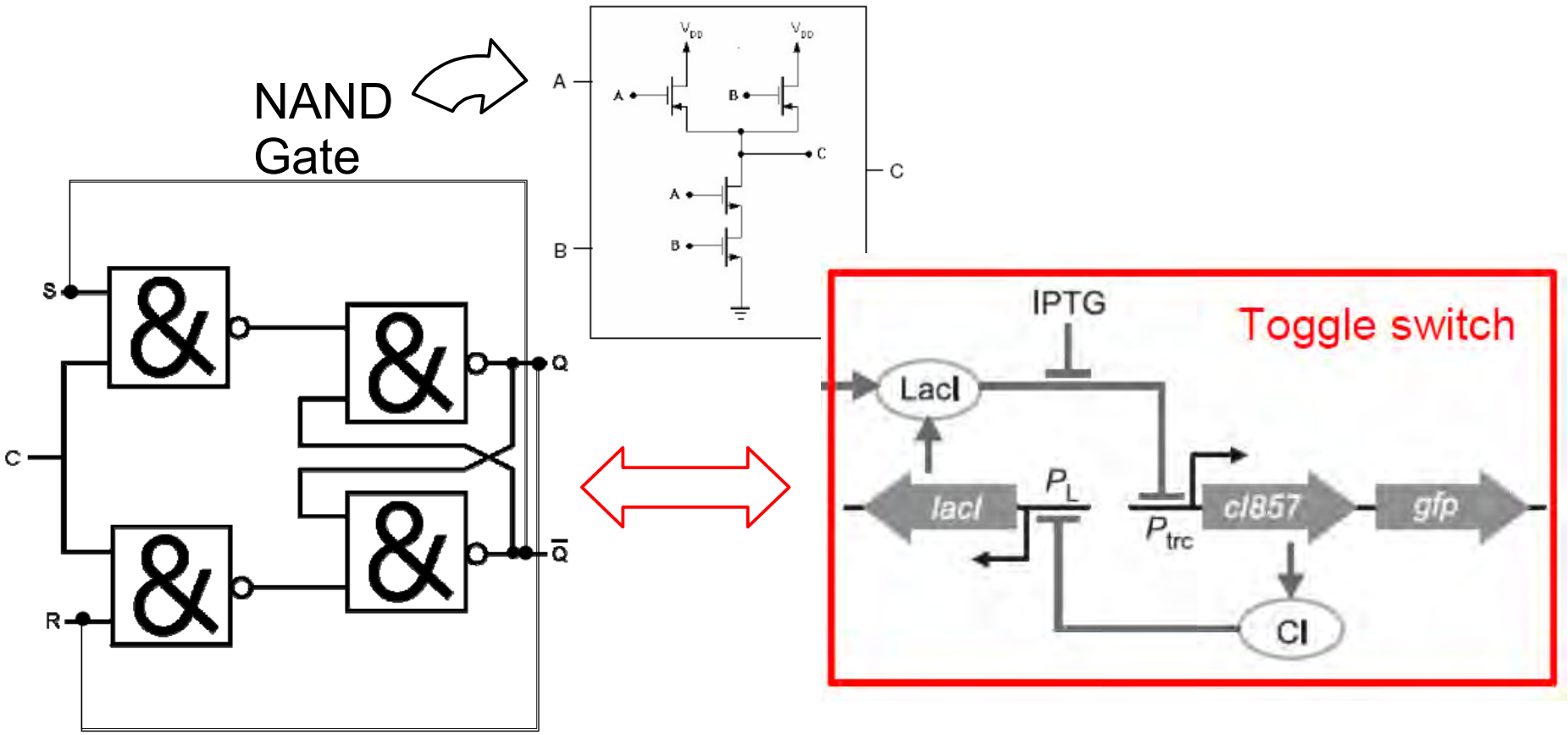
Synthetic Biology 3.0, Zurich, June 2007

Synthetic Biology Vision

- Rational forward-engineering design of ...
- ... robust / reliable biology-based parts and modules with standardized interfaces allowing plug-and-play ...
- ... and their combination into complex systems.

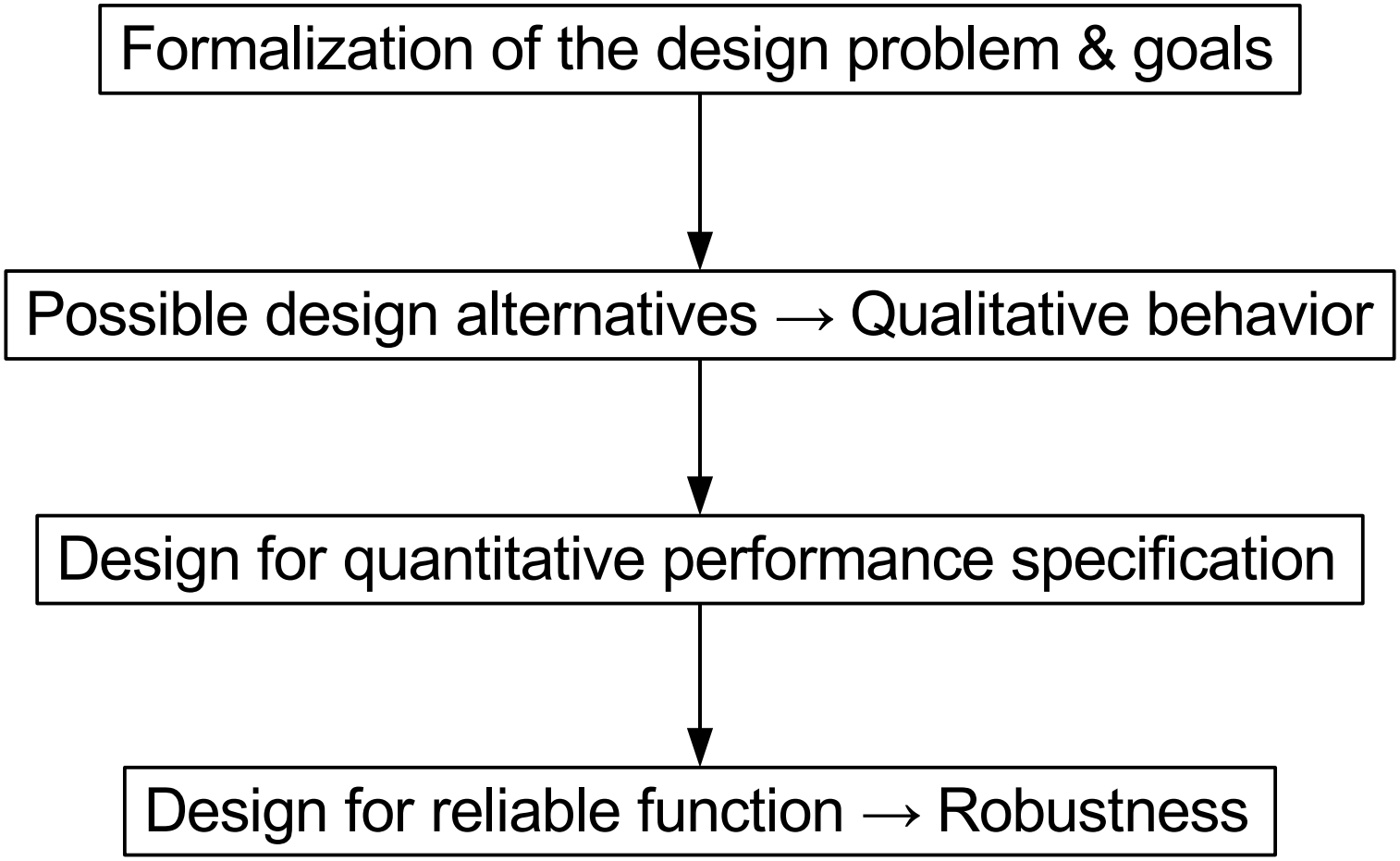


Engineering Design & Synthetic Biology



- Novel design methods / tools because of 'sloppyness', stochasticity, and limited insulation of components in biology.

Circuits: Model-based Design Process



Steps in Model Development

(I) Verbal biological model ⟶ Biochemical reaction network

$$\alpha_{1,j} \cdot X_1 + \ldots + \alpha_{n,j} \cdot X_n \overset{k_j}{\rightarrow} \beta_{1,j} \cdot X_1 + \ldots + \beta_{n,j} \cdot X_n$$

System of elementary chemical reactions

(II) Biochemical reaction network ⟶ Mathematical model

$$\frac{dc_i}{dt} = \sum_{j=1}^{r} k_j \cdot (\beta_{i,j} - \alpha_{i,j}) \cdot \prod_{l \in S_j} c_l^{\alpha_{l,j}}$$

System of ordinary differential equations (ODEs)

Steps in Model Development

(I) Verbal biological model → Biochemical reaction network

- Most important aspects of the system ?
- Complete knowledge on components / interactions ?
- Exact mechanisms of interactions ?

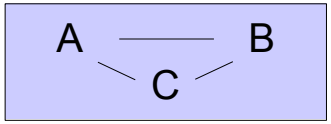
(II) Biochemical reaction network → Mathematical model

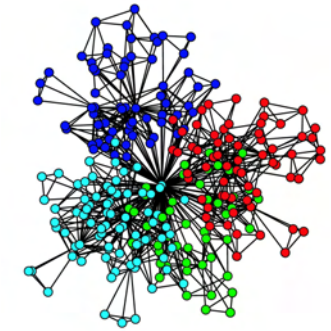
- Level of detail for the mathematical descriptions ?
- Modeling approach (qualitative / mechanistic / ...) ?
- Experimental data for identification & validation ?

Modeling Approaches

Interaction-based

topology

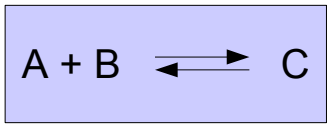


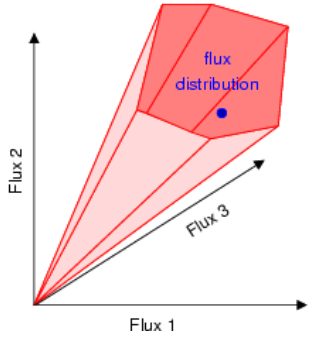


Graph theory

Constraint-based

stoichiometry

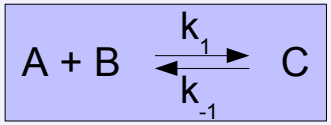


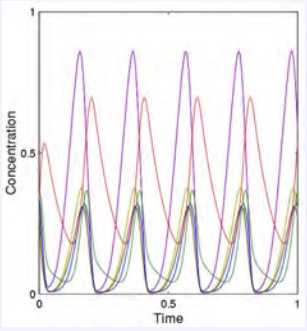


Structural analysis

Mechanism-based

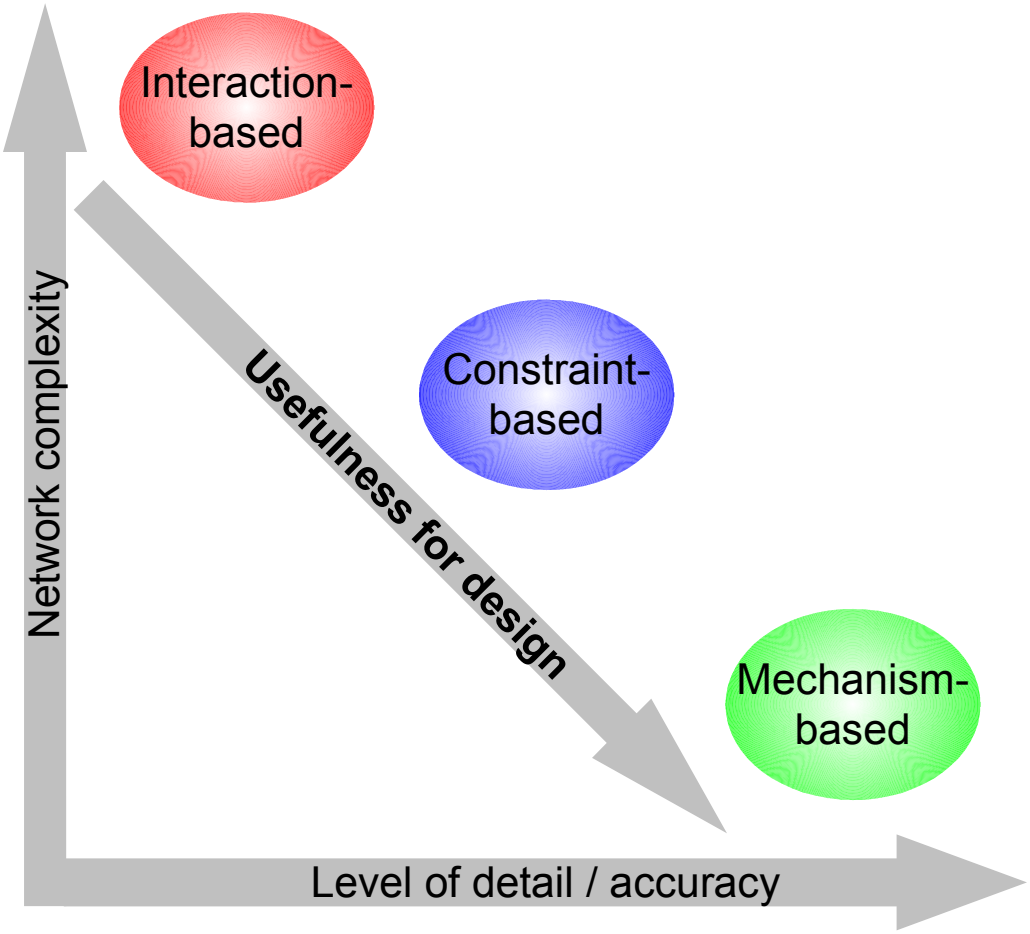
biochemistry





Dynamic analysis

Modeling Approaches: Comparison



Dynamic Systems Analysis: Approach

- Analyze engineered circuits as dynamic (bio)chemical reaction networks → Description of reaction kinetics.
- Based on first principles: Conservation of mass (and energy and possibly other constraints).
- Theoretical background: Chemical kinetic theory.
- (Ordinary) differential equations / Stochastic processes.

Reaction Kinetics: Law of Mass Action

- Law of mass action → Concentrations of reacting molecules in thermodynamic equilibrium.

- Product of concentrations taken to the power of the stoichiometric factors (reaction order) equals a constant (dependent on temperature, pressure, ...).

- Example: $1\,A + 2\,B \rightleftharpoons 1\,C$

$$\frac{[A]\cdot[B]^2}{[C]} = k(T, p)$$

Reaction Kinetics: Dynamic Systems

- Reaction network → System of elementary reactions:

$$\alpha_{1,j} \cdot X_1 + \ldots + \alpha_{n,j} \cdot X_n \xrightarrow{k_j} \beta_{1,j} \cdot X_1 + \ldots + \beta_{n,j} \cdot X_n$$

- Law of mass action → System of differential equations:

$$\frac{dc_i(t)}{dt} = \sum_{j=1}^{q} k_j \cdot (\beta_{i,j} - \alpha_{i,j}) \cdot \prod_l c_l(t)^{\alpha_{l,j}}$$

- Equivalence to: $\frac{d\,\boldsymbol{c}(t)}{dt} = \boldsymbol{N} \cdot \boldsymbol{r}(t)$

Reaction Kinetics: Dynamic Models

$$\frac{d\,\boldsymbol{c}(t)}{dt} = \boldsymbol{N}\cdot\boldsymbol{r}\left(\boldsymbol{c}(t), \boldsymbol{u}(t), \boldsymbol{p}, t\right)$$

- Reactand concentrations $\boldsymbol{c}(t) \rightarrow$ To be determined.
- Stoichiometric matrix $\boldsymbol{N} \rightarrow$ Systems invariant.
- Reaction rates $\boldsymbol{r} \rightarrow$ Time- and state-dependent:
  - Kinetic rate law $\boldsymbol{r}(\cdot) \rightarrow$ From reaction structure.
  - Parameters (kinetic constants) $\boldsymbol{p} \rightarrow$ Identification.
  - Inputs $\boldsymbol{u}(t) \rightarrow$ Additional (time-varying) influences.

ODE Models: General Form

$$\frac{d\,\boldsymbol{x}(t)}{dt} = f\left(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}, t\right)$$

- System of ordinary, first-order, linear or nonlinear differential equations (ODEs) characterized by:
  - Right hand sides $\boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{p})$ = function in $\mathbb{R}^{n_x}$.
  - System states $\boldsymbol{x}(t)$ = $n_x \, x \, 1$ state vector.
  - Parameters $\boldsymbol{p}$ = $n_p \, x \, 1$ parameter set.
  - Inputs $\boldsymbol{u}(t)$ = $n_u \, x \, 1$ input vector.

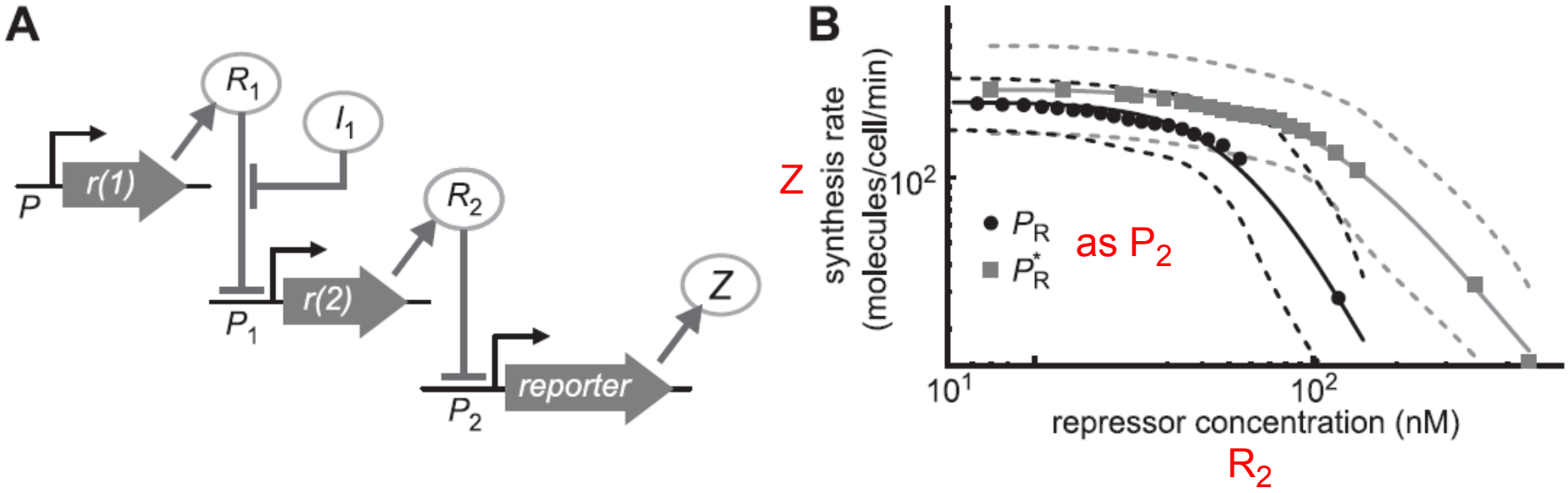
ODE Models: Solution

$$\frac{d\,\boldsymbol{x}(t)}{dt} = f\left(\boldsymbol{x}(t), \boldsymbol{p}, t\right) \quad , \quad \boldsymbol{x}(t_0) = \boldsymbol{x_0}$$

- Existence and uniqueness of solution to the initial value problem (IVP) of finding $\boldsymbol{x}(t)$ with given $\boldsymbol{x_0}$ guaranteed.

- Three possible "solution" methods:
  - Analytical → Only applicable for simple systems.
  - Numerical → Always possible for well-posed IVPs.
  - Graphical → Qualitative analysis methods.

Example: Two-step Repressor Cascade



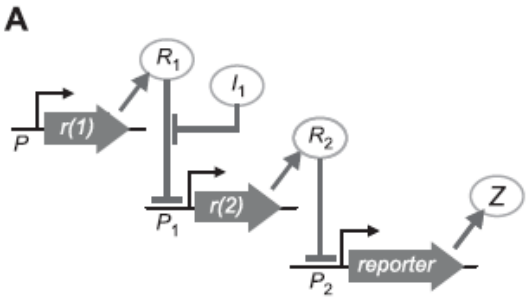
M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).

- Signal-response characteristics → Promoter selection.
- Low-pass filter: High *I* levels – low *Z* synthesis rate.

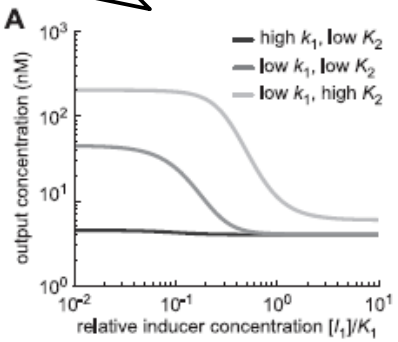
Example: Two-step Repressor Cascade

$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$



Design Cycle

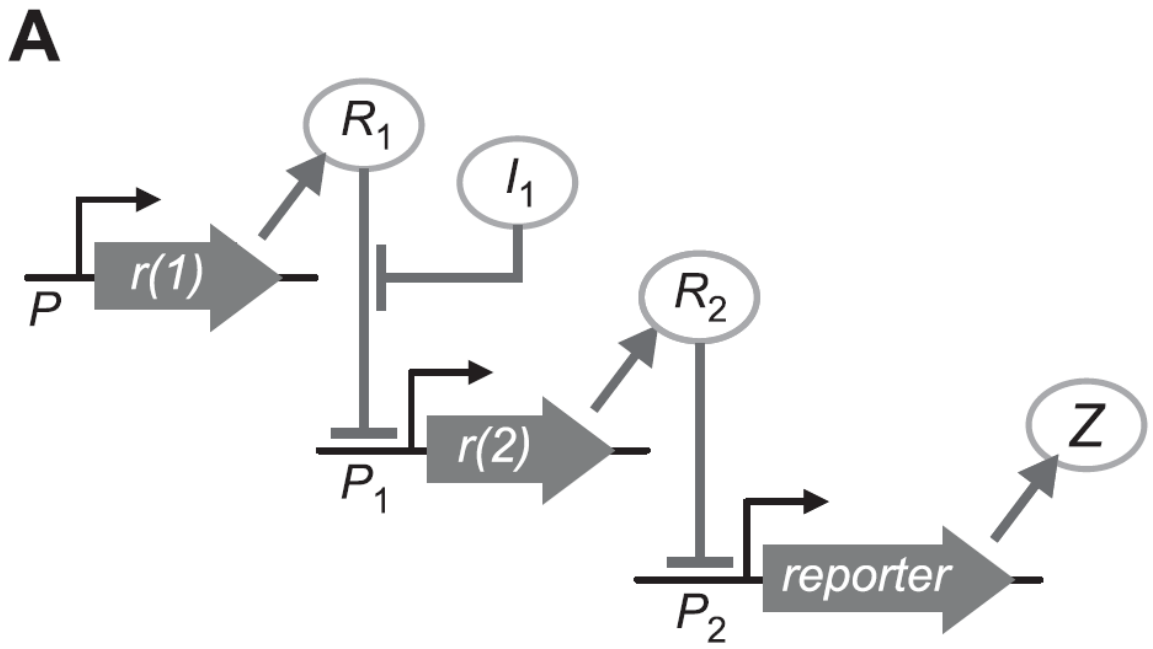


$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$

Example: Two-step Repressor Cascade

M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).

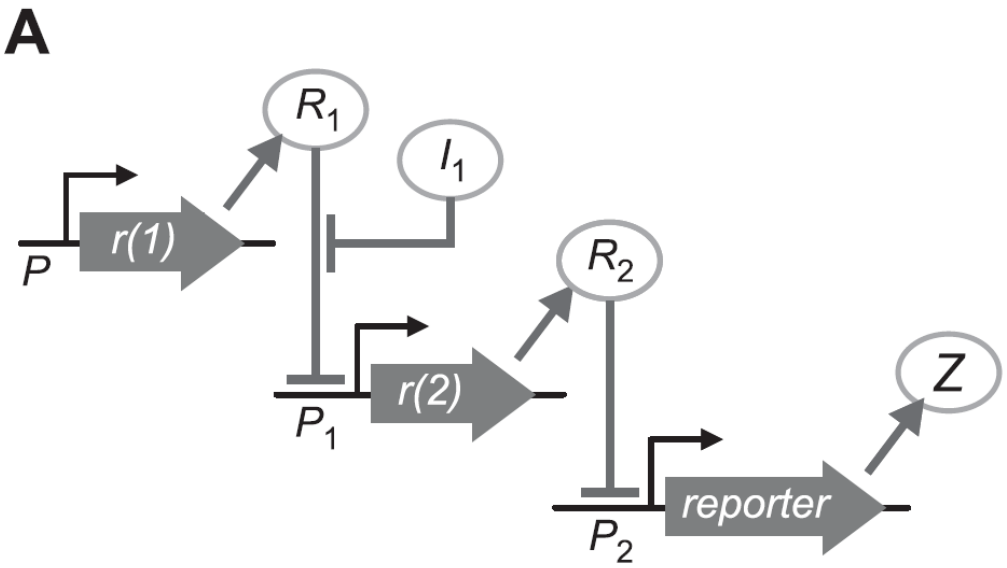


$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$

Example: Two-step Repressor Cascade

M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).



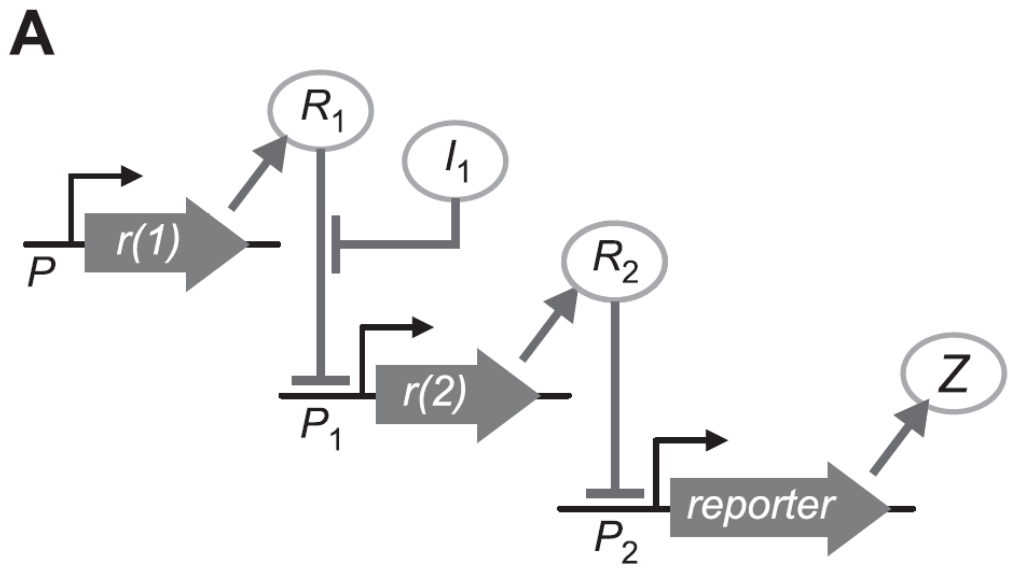
$$\frac{d[R_2]}{dt} = \boxed{a_1 \cdot k_1} + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = \boxed{a_2 \cdot k_2} + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$

- Low constitutive activity of P_1 and P_2.

Example: Two-step Repressor Cascade

M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).



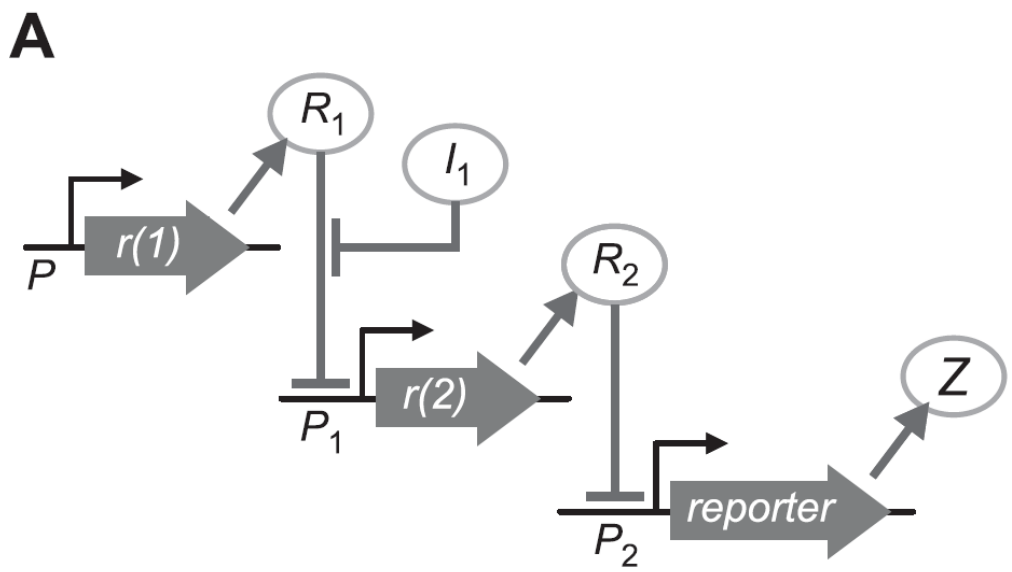
$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$

- Constitutive degradation of all proteins.

Example: Two-step Repressor Cascade

M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).



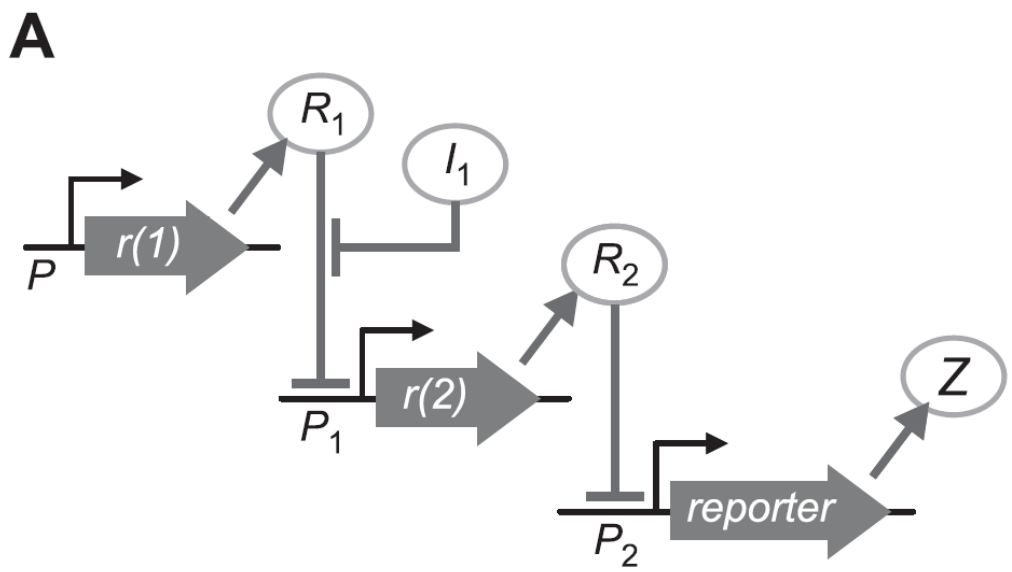
$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \frac{k_2}{1 + ([R_2]/K_2)^{n_2}} - d_2 \cdot [Z]$$

- Binding of R_1 and I_1 → Cooperative transcriptional activation.

Example: Two-step Repressor Cascade

M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).



$$\frac{d[R_2]}{dt} = a_1 \cdot k_1 + \frac{k_1 \cdot ([I_1]/K_1)^{n_1}}{1 + ([I_1]/K_1)^{n_1}} - d_1 \cdot [R_2]$$

$$\frac{d[Z]}{dt} = a_2 \cdot k_2 + \boxed{\frac{k_2}{1 + ([R_2]/K_2)^{n_2}}} - d_2 \cdot [Z]$$

- Cooperative transcriptional repression of P_2 by R_2.

Circuit Models: Generalizations

- Derivation of rate laws or equilibrium binding concentrations for structurally similar reaction networks yields similar basic functional terms.

- Example: Gene G bound by transcription factor T:
  - Without repression: $[G \cdot T] = \frac{[G]^T [T]}{[T] + K}$
  - Competitive repressor R: $[G \cdot T] = \frac{[G]^T [T]}{[T] + K(1 + [T]/K_I)}$
  - Cooperative binding: $[G \cdot T] = \frac{[G]^T [T]^n}{[T]^n + K^n}$

Circuit Models: Generalizations

- General model structure for (simple) genetic circuits:

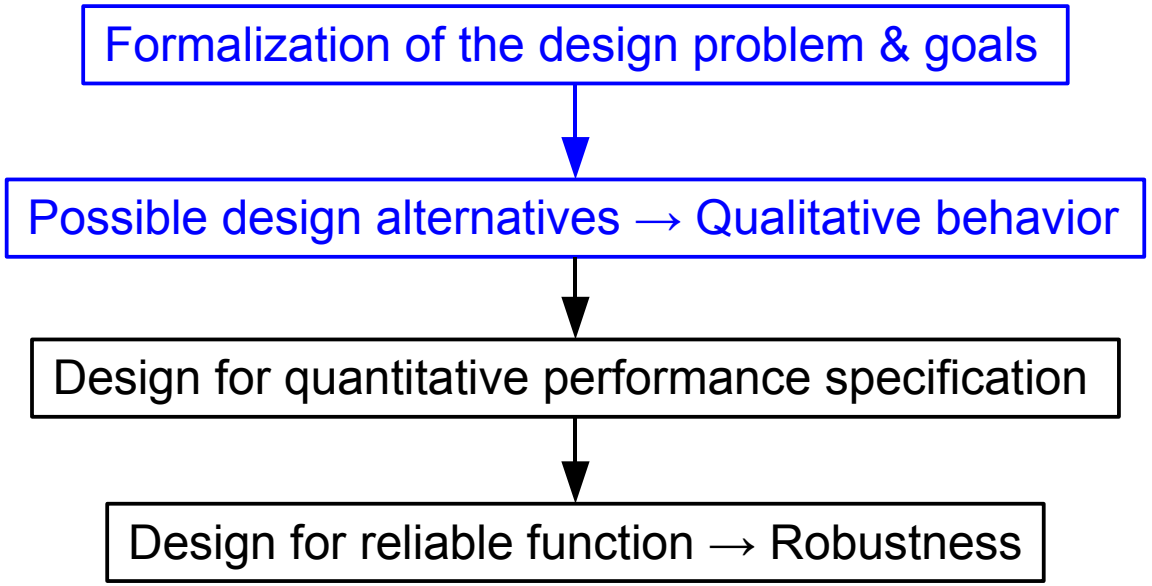
$$\frac{d[X_i]}{dt} = a_i \cdot k_i + \frac{k_i \cdot ([X_j]^n / K_i^n)^\mu}{1 + ([X_j]^n / K_i^n)} - d_i \cdot [X_i]$$

  - Activation of expression of X_i by $X_j \rightarrow \mu = 1$.
  - Repression of expression of X_i by $X_j \rightarrow \mu = 0$.
  - Always: Basal expression / constitutive degradation.

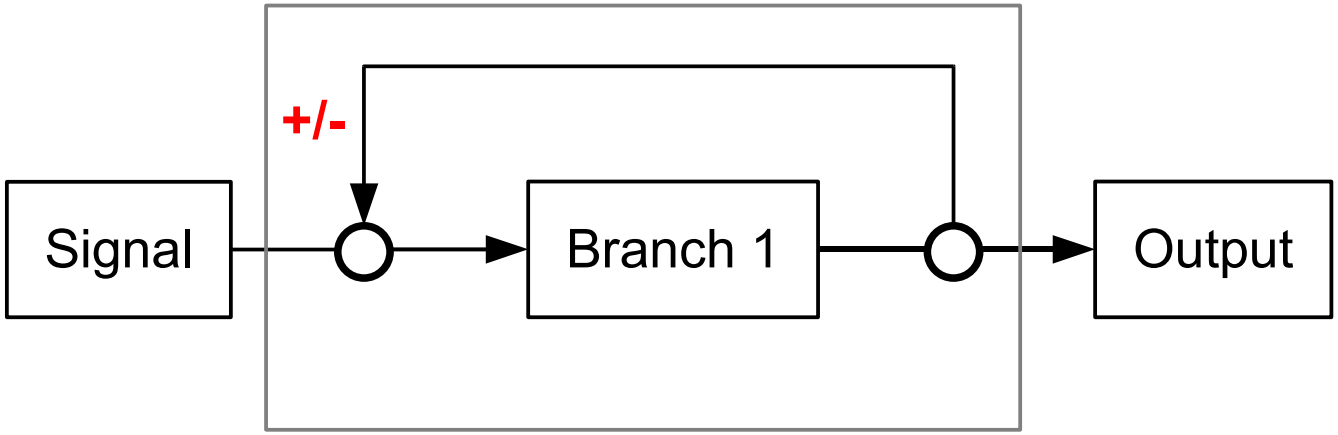
J. theor. Biol. (2000) **203,** 117–133
doi:10.1006/jtbi.2000.1068, available online at http://www.idealibrary.com on IDEAL®

How to make a Biological Switch

JOSHUA L. CHERRY*† AND FREDERICK R. ADLER*‡



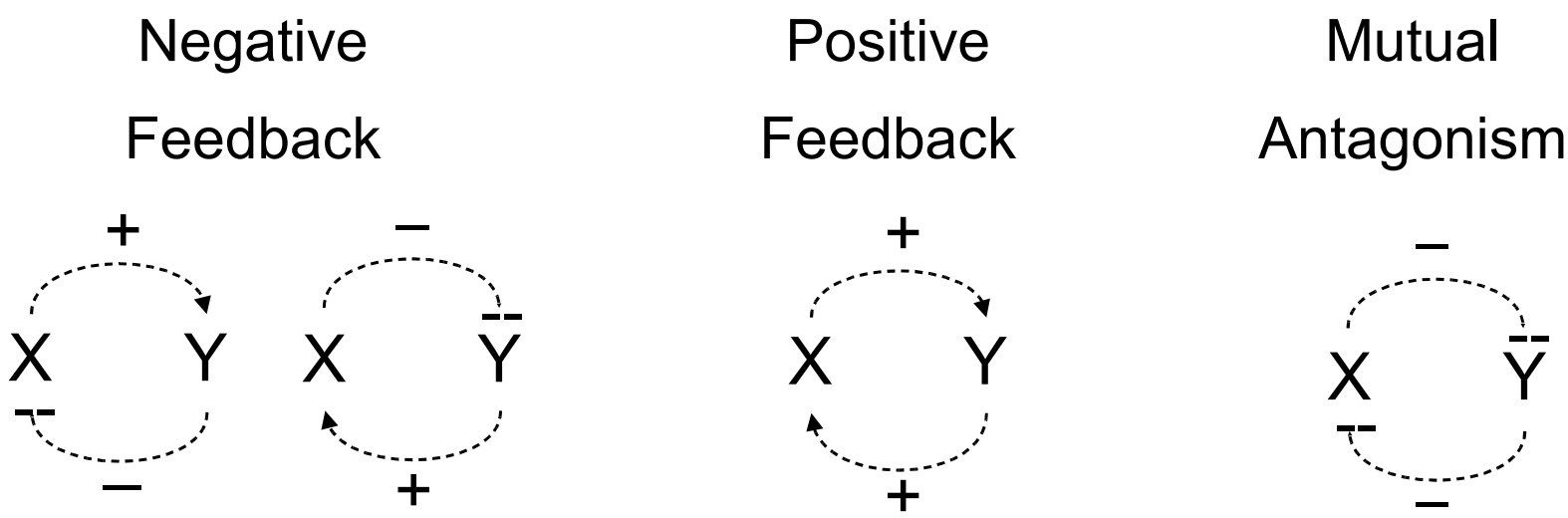
Feedback Systems



- Feedback of module's output signal on the input signal.
- Main categories: Positive feedback / negative feedback.
- Essential for: Controllers, switches, oscillators, ...

Feedback Systems: Simple Types

- Patterns of interactions between two components:

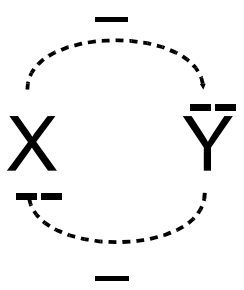


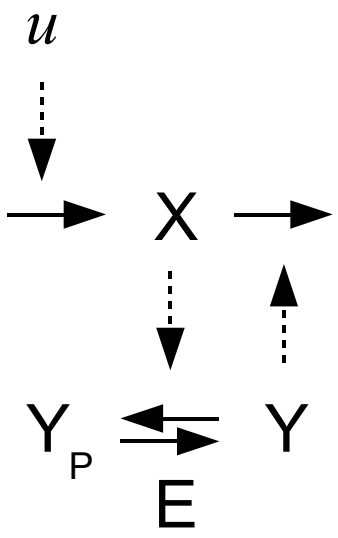
- Positive or negative (net) effect of interactions:

$$\frac{d\,\boldsymbol{x}(t)}{dt} = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}, t) \quad \Rightarrow \quad \frac{\partial f_i(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{p}, t)}{\partial x_j} \neq 0$$

Example Switch: System

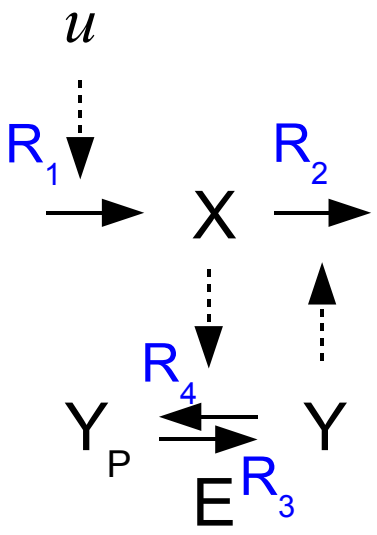
Abstraction





- Component X: Inactivates component Y $\rightarrow$ Y_P.
- Component Y: Degrades component X.
- Input signal u: Control of production rate for X.

Example Switch: ODE Model

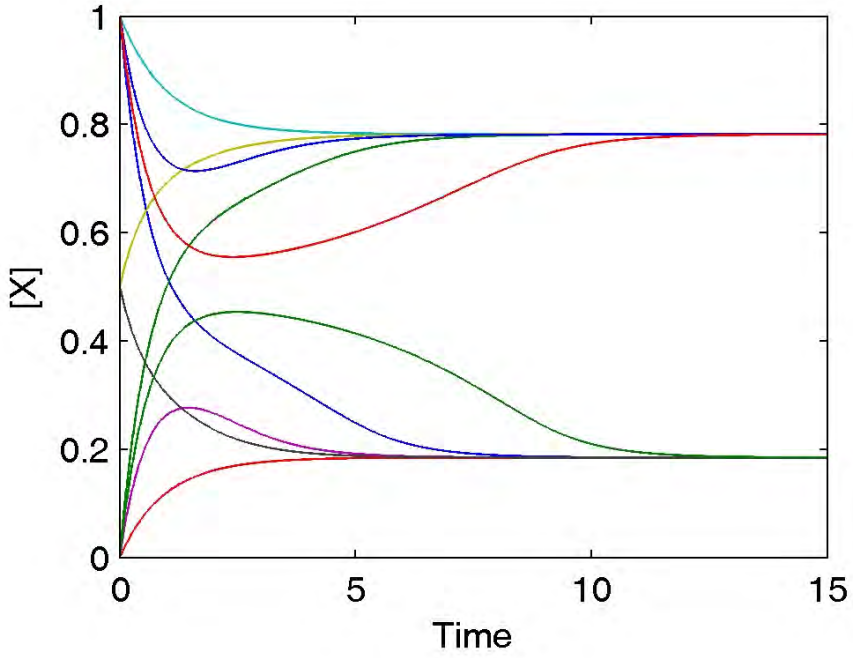


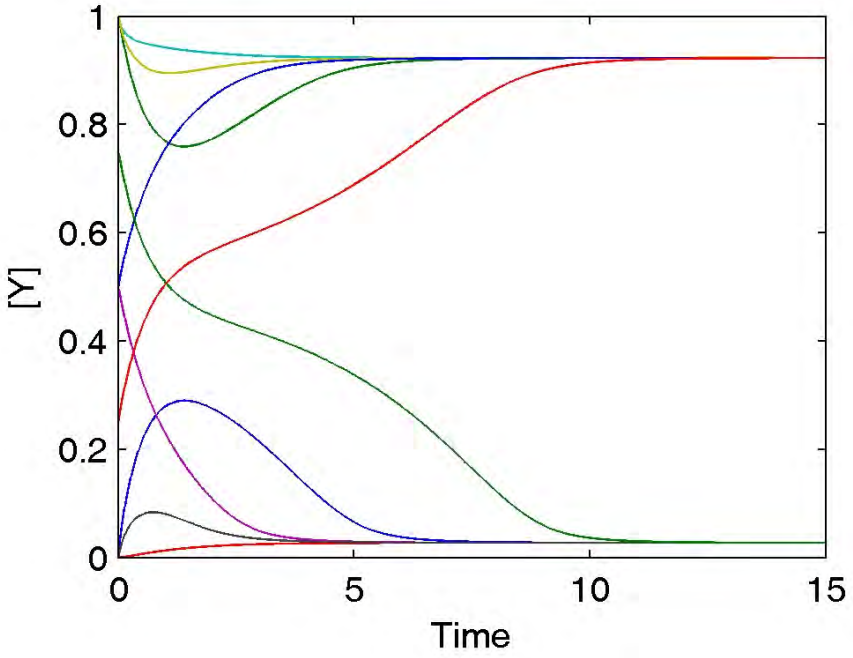
- Assuming constant total concentration of $Y \rightarrow Y^T$:

$$\frac{d[X]}{dt} = k_1 \cdot u - \left(k_2' + k_2 \cdot [Y]\right)[X]$$

$$\frac{d[Y]}{dt} = \frac{k_3 \cdot [E]\left([Y]^T - [Y]\right)}{K_{M3} + [Y]^T - [Y]} - \frac{k_4 [X][Y]}{K_{M4} + [Y]}$$

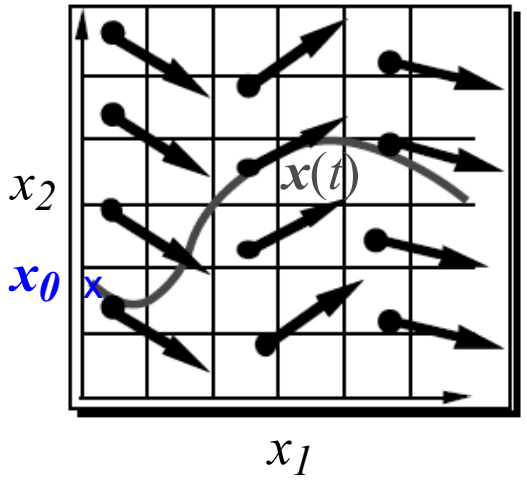
Example Switch: Numerical Solution





- Assume: Different initial concentrations of X / Y.
- Convergence to qualitatively different solutions.

Example Switch: Graphical 'Solution'

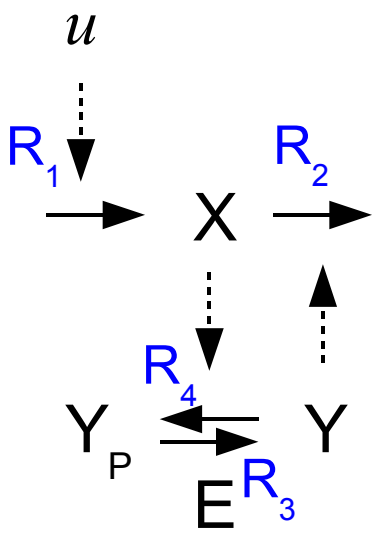


$$\frac{d\,\boldsymbol{x}(t)}{dt} = f(\boldsymbol{x}(t), \boldsymbol{p}, t)$$

$$\boldsymbol{x}(t_0) = \boldsymbol{x}_0$$

- Derivatives $d\boldsymbol{x}(t)/dt$ define vector field in state space.
- Qualitative analysis for two-dimensional systems:
  - Nullclines: Zero velocity in one dimension.
  - Steady states: Zero velocity in both dimensions.

Example Switch: Nullclines



$$\frac{d[X]}{dt} = k_1 \cdot u - \left(k_2' + k_2 \cdot [Y]\right)[X]$$

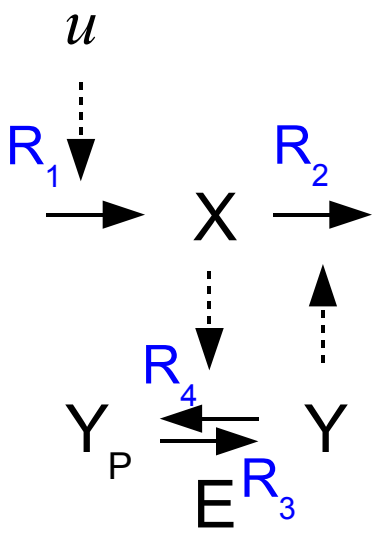
$$\frac{d[Y]}{dt} = \frac{k_3 \cdot [E]\left([Y]^T - [Y]\right)}{K_{M3} + [Y]^T - [Y]} - \frac{k_4 [X][Y]}{K_{M4} + [Y]}$$

- States with zero velocity in one of the directions (nullclines):

$$\frac{d[X]}{dt} = 0 \quad \Rightarrow \quad [Y] = \frac{k_1 \cdot u - k_2' \cdot [X]}{k_2 \cdot [X]}$$

$$\frac{d[Y]}{dt} = 0 \quad \Rightarrow \quad \frac{k_3 \cdot [E]\left([Y]^T - [Y]\right)}{K_{M3} + [Y]^T - [Y]} = \frac{k_4 [X][Y]}{K_{M4} + [Y]}$$

Example Switch: Y-Nullcline



- Y-nullcline in original variables:

$$\frac{k_3 \cdot [E]\left([Y]^T - [Y]\right)}{K_{M3} + [Y]^T - [Y]} = \frac{k_4 [X][Y]}{K_{M4} + [Y]}$$

- Introduction of new variables:

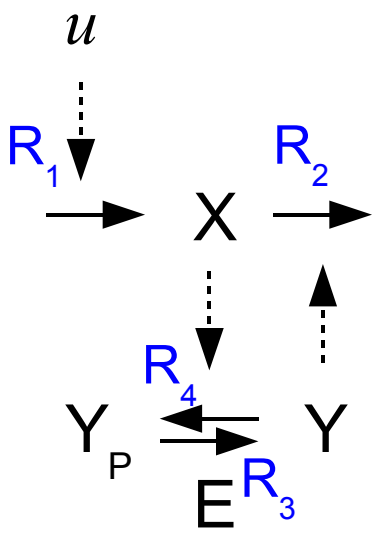
$$y = \frac{[Y]}{[Y]^T} \quad , \quad v_1 = k_3 \cdot [E] \quad , \quad v_2 = k_4 \cdot [X]$$

$$J_1 = \frac{K_{M3}}{[Y]^T} \quad , \quad J_2 = \frac{K_{M4}}{[Y]^T}$$

- Rescaled equation for Y-nullcline:

$$v_1 (1 - y)(J_2 + y) = v_2 \cdot y (J_1 + 1 - y)$$

Example Switch: Y-Nullcline



- Rescaled equation for Y-nullcline:

$$y = \frac{[Y]}{[Y]^T} \quad , \quad v_1 = k_3 \cdot [E] \quad , \quad v_2 = k_4 \cdot [X]$$

$$J_1 = \frac{K_{M3}}{[Y]^T} \quad , \quad J_2 = \frac{K_{M4}}{[Y]^T}$$

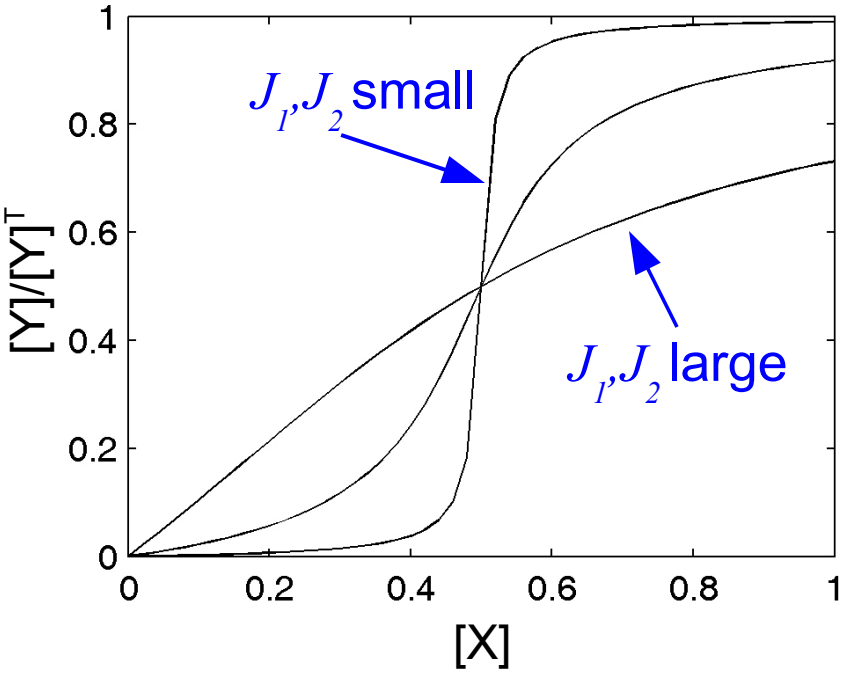
$$v_1 (1-y)(J_2+y) = v_2 \cdot y (J_1+1-y)$$

- Solution in new variables → Goldbeter-Koshland function:

$$y = G(v_1, v_2, J_1, J_2) = \frac{2 v_1 J_2}{B + \sqrt{B^2 - 4(v_2 - v_1) v_1 J_2}}$$

$$B = v_2 - v_1 + v_2 J_1 + v_1 J_2$$

Example Switch: Y-Nullcline

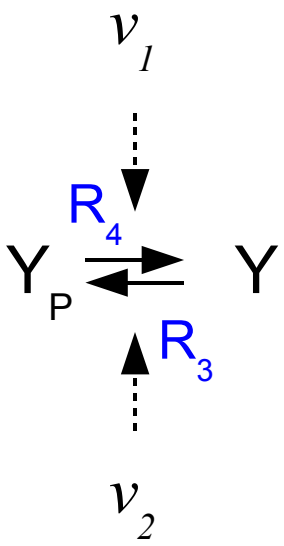


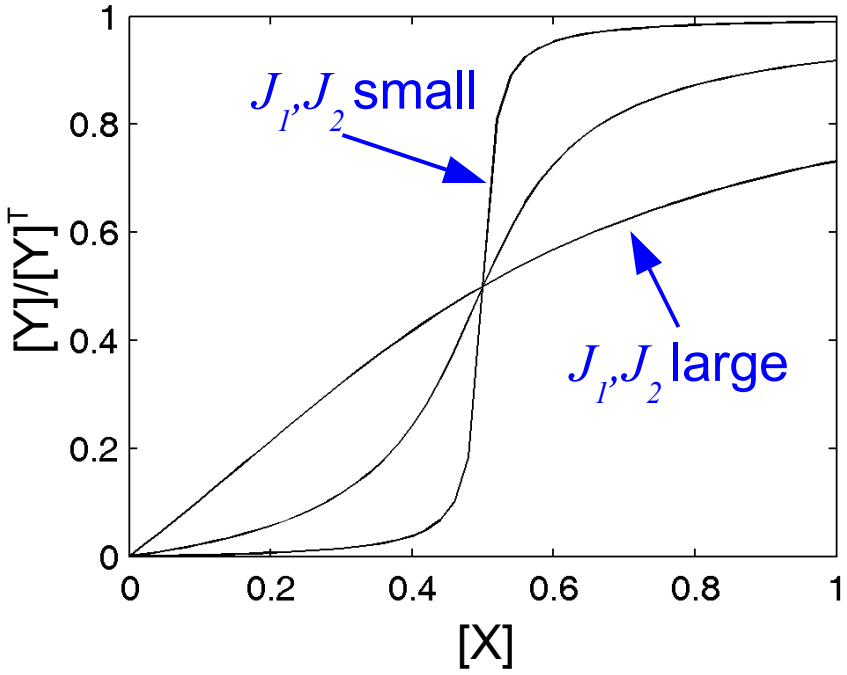
$$y = G(v_1, v_2, J_1, J_2) = \frac{2 v_1 J_2}{B + \sqrt{B^2 - 4(v_2 - v_1) v_1 J_2}} \quad , \quad B = v_2 - v_1 + v_2 J_1 + v_1 J_2$$

$$y = \frac{[Y]}{[Y]^T} \quad , \quad v_1 = k_3 \cdot [E] \quad , \quad v_2 = k_4 \cdot [X] \quad , \quad J_1 = \frac{K_{M3}}{[Y]^T} \quad , \quad J_2 = \frac{K_{M4}}{[Y]^T}$$

- Sigmoidal function of input X $\rightarrow$ Switch-like for $0 < J_1, J_2 << 1$.

Example Switch: Y-Nullcline

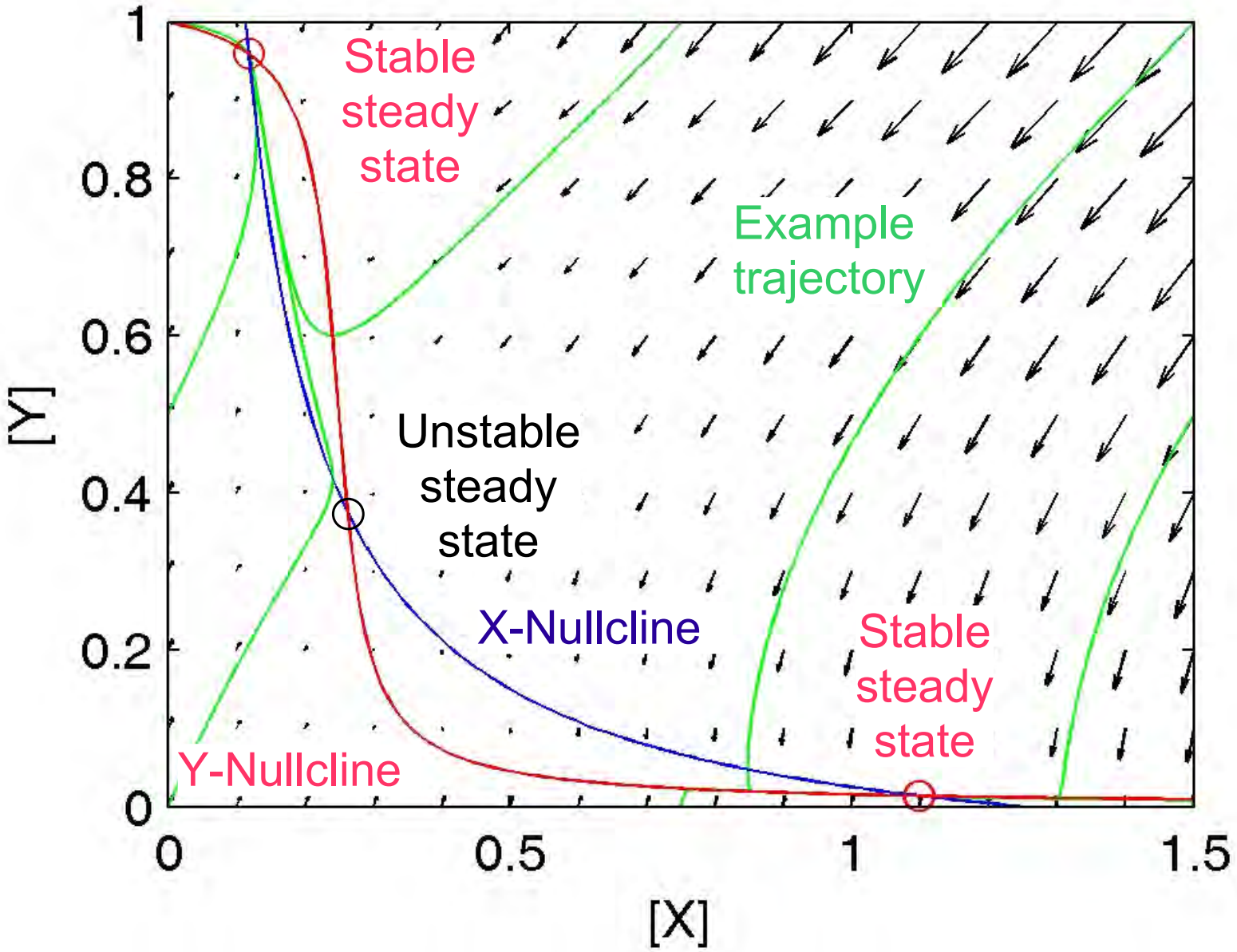




$$y = \frac{[Y]}{[Y]^T} \quad , \quad v_1 = k_3 \cdot [E] \quad , \quad v_2 = k_4 \cdot [X] \quad , \quad J_1 = \frac{K_{M3}}{[Y]^T} \quad , \quad J_2 = \frac{K_{M4}}{[Y]^T}$$

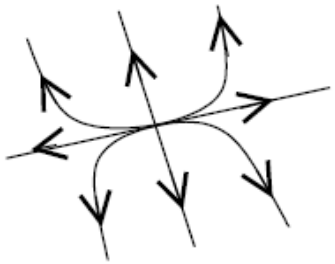
- General: Switch-like functions using reversible reactions.
- Necessary: High affinities and / or excess of total regulator.

Example Switch: Qualitative Behavior

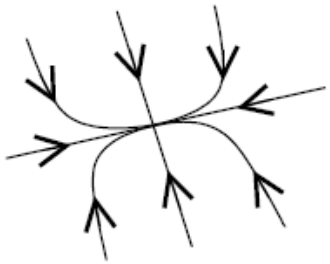


Example Switch: Stability

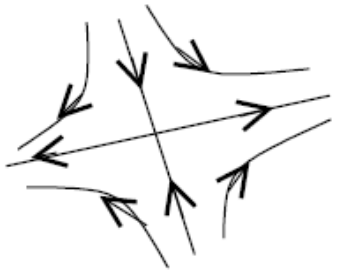
- Classification of steady states (nodes) according to directions of the vector field:



unstable node



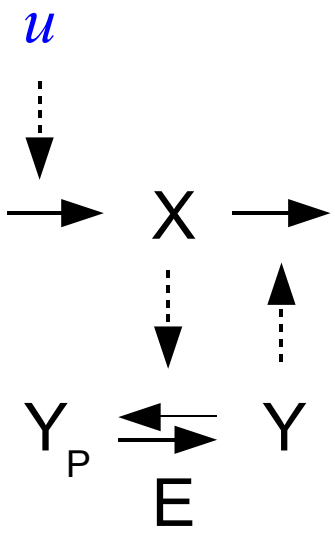
stable node

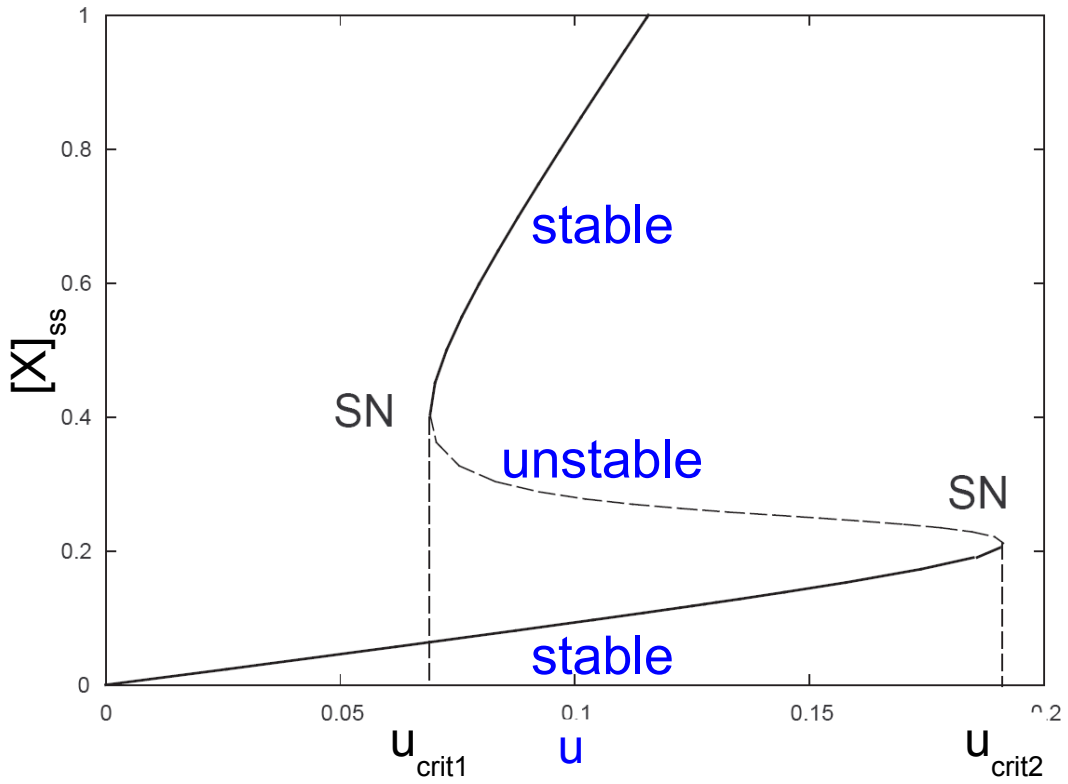


saddle point (unstable)

- Stability: Global versus local (w.r.t. 'small' perturbations).

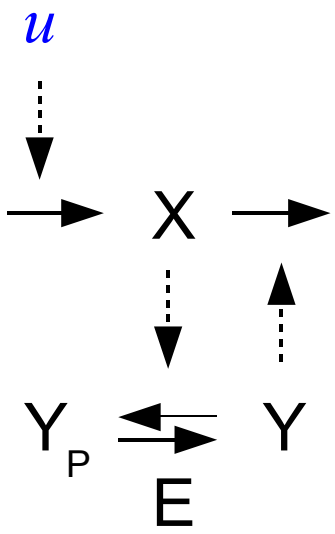
Example Switch: Response to Input

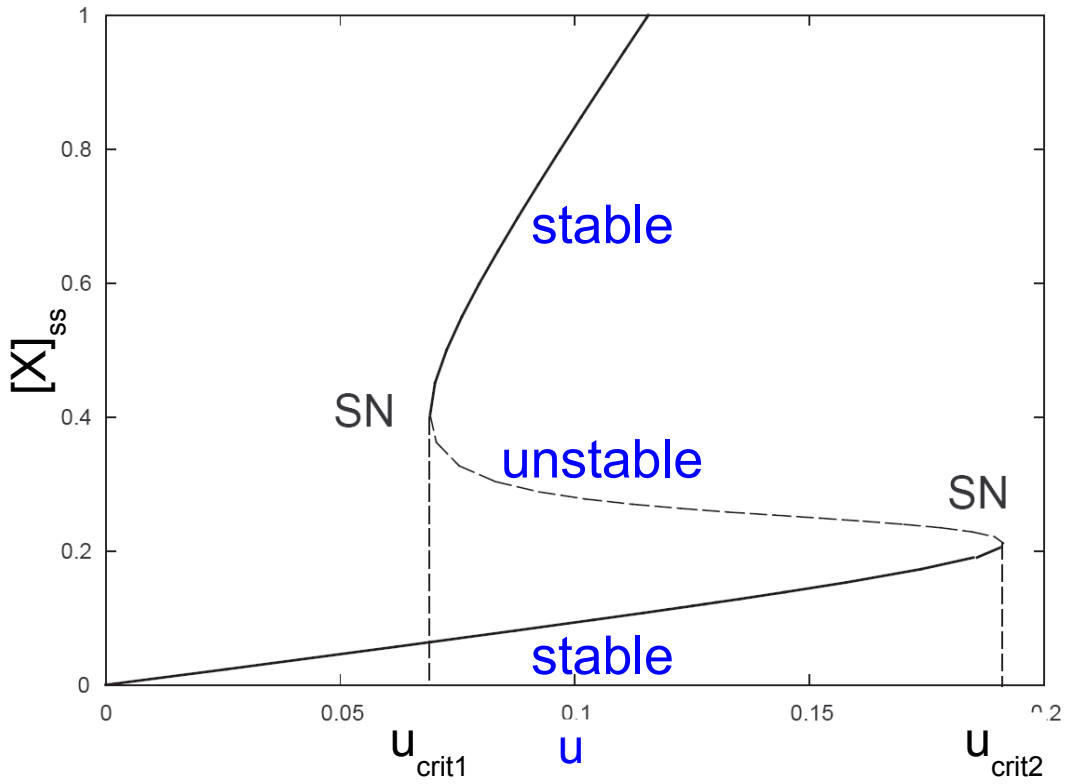




- X-Nullcline: $\frac{d[X]}{dt} = 0 \Rightarrow [Y] = \frac{k_1 \cdot u - k_2' \cdot [X]}{k_2 \cdot [X]}$
- Bifurcation: Change of the number of attractors in a (nonlinear) dynamic system upon parameter changes.

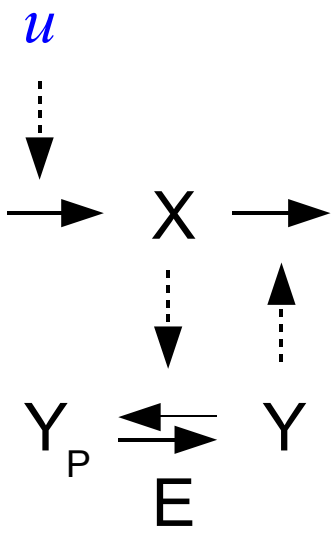
Example Switch: Response to Input

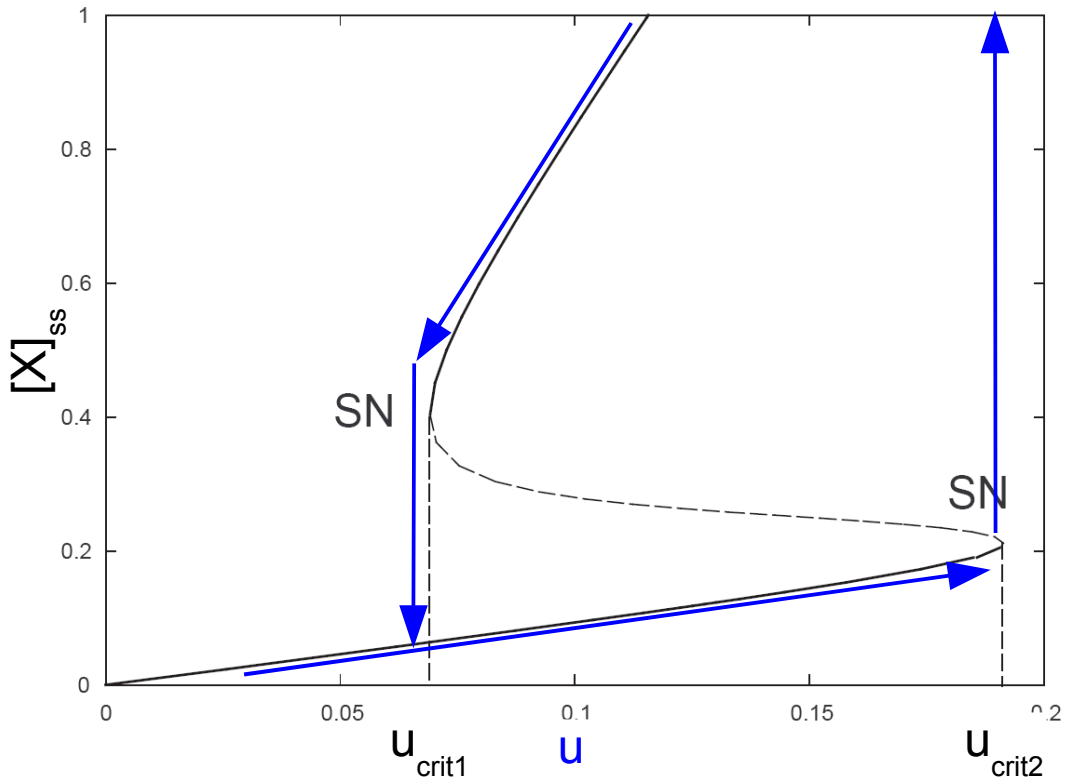




- For $u < u_{crit1}$ and $u > u_{crit2}$: Globally monostable system.
- For $u_{crit1} \leq u \leq u_{crit2}$: Bistable system → Switch possible.

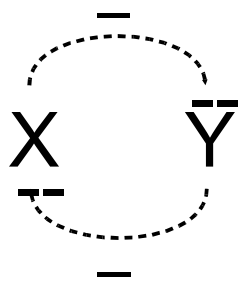
Example Switch: Response to Input





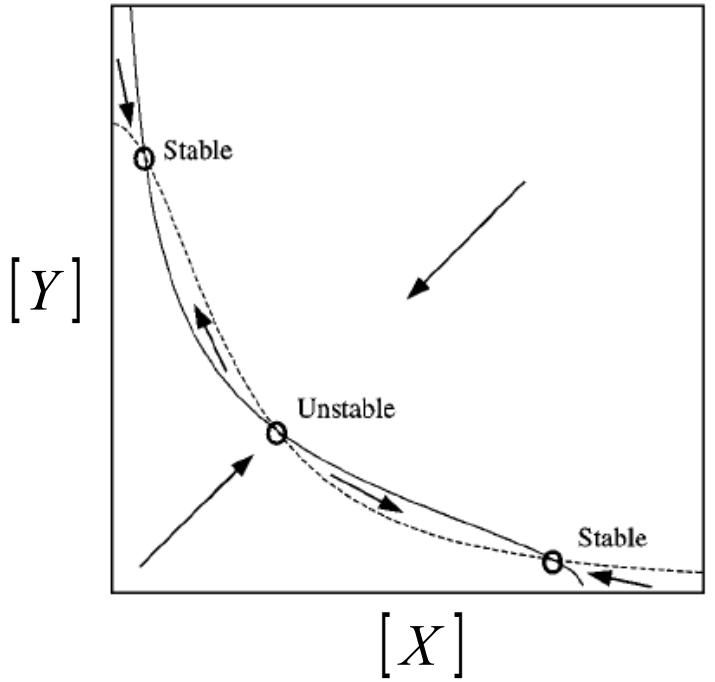
- History dependence of the system's state (here with respect to changes in the input): Hysteresis.
- Functional implication for circuit behavior: Memory.

Switches: Generalization



$$\frac{d[X]}{dt} = f([Y]) - d_1[X]$$

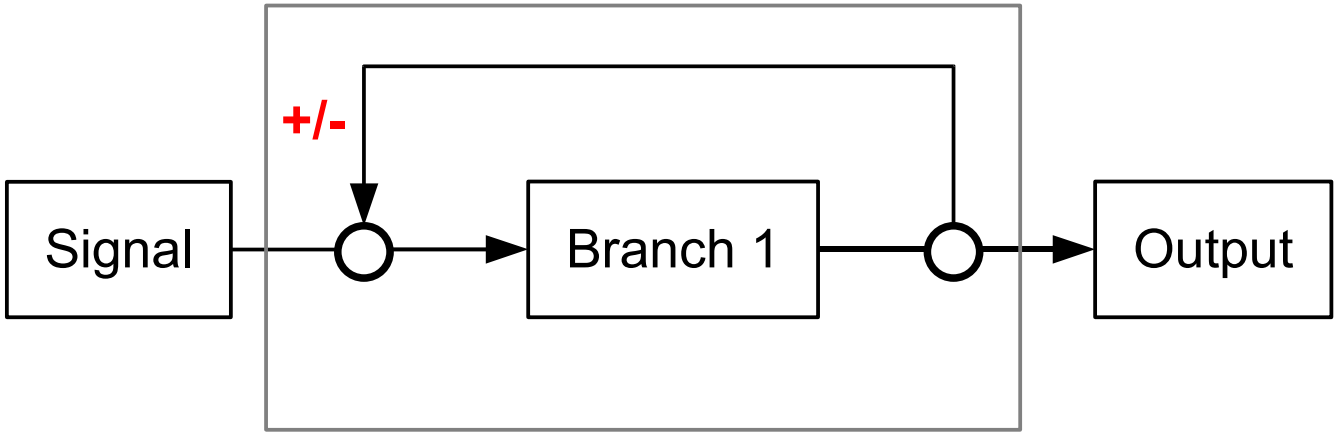
$$\frac{d[Y]}{dt} = g([X]) - d_2[Y]$$



J. Cherry & F. Adler, J. theor. Biol. 203:117 (2000).

- Analysis of alternative designs for biological switches.
- Phase plane analysis, multiplicity of steady states.
- Mechanisms: Cooperativity (at least in one branch).

Feedback Systems



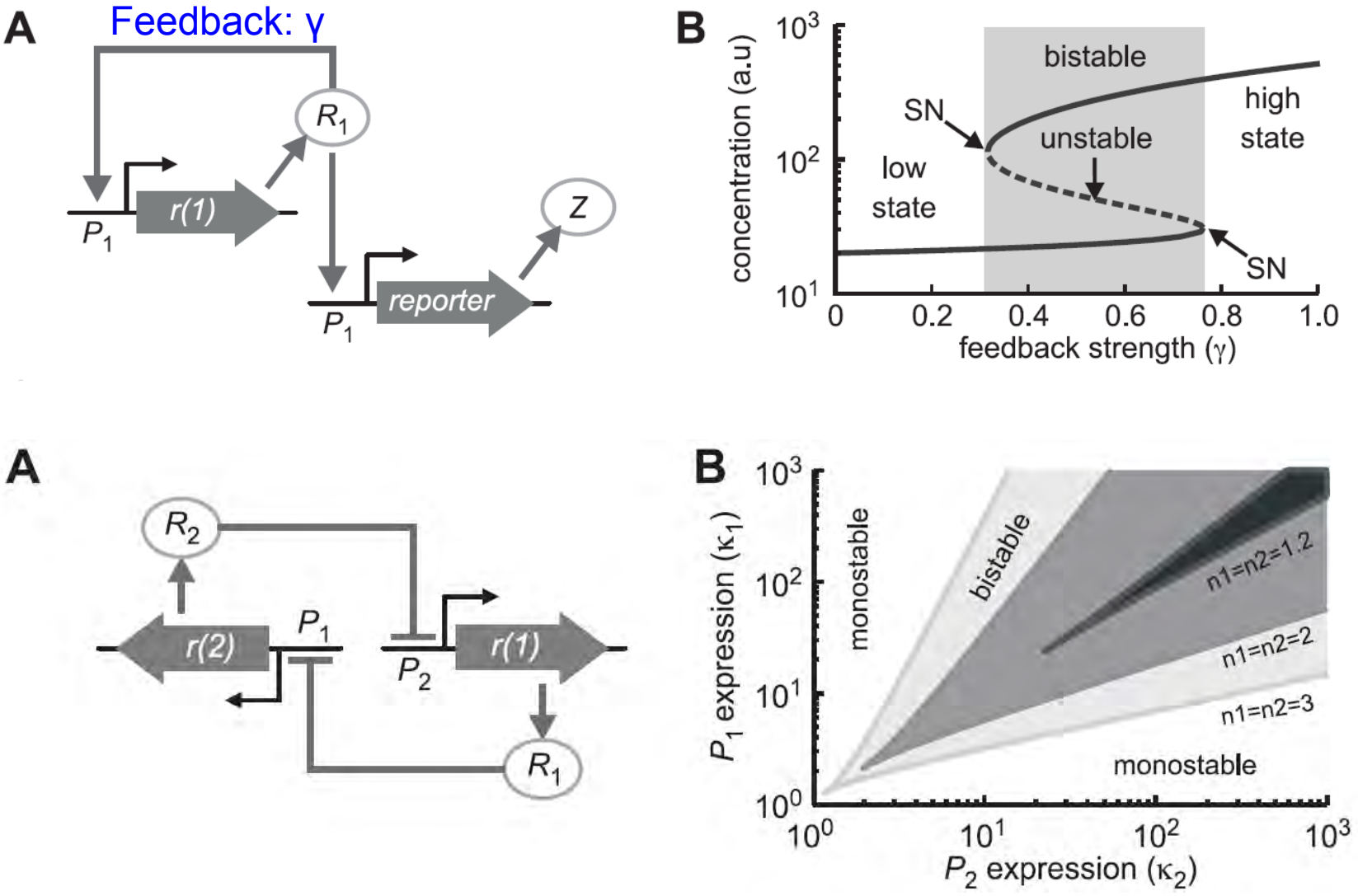
- Main categories: Positive feedback / negative feedback.
- Essential for: Controllers, switches, oscillators, ...
- And beyond switches relying on mutual repression ... ?

Positive Feedback: Functions

- Simple positive feedback systems:
  - Multiple (stable / unstable) steady states possible.
  - Phenomenon in nonlinear systems: Hysteresis.

- Functions in biological networks:
  - Discrete decisions from continuous signals.
  - Irreversibility of decisions, e.g. in development.

Positive Feedback: Realizations



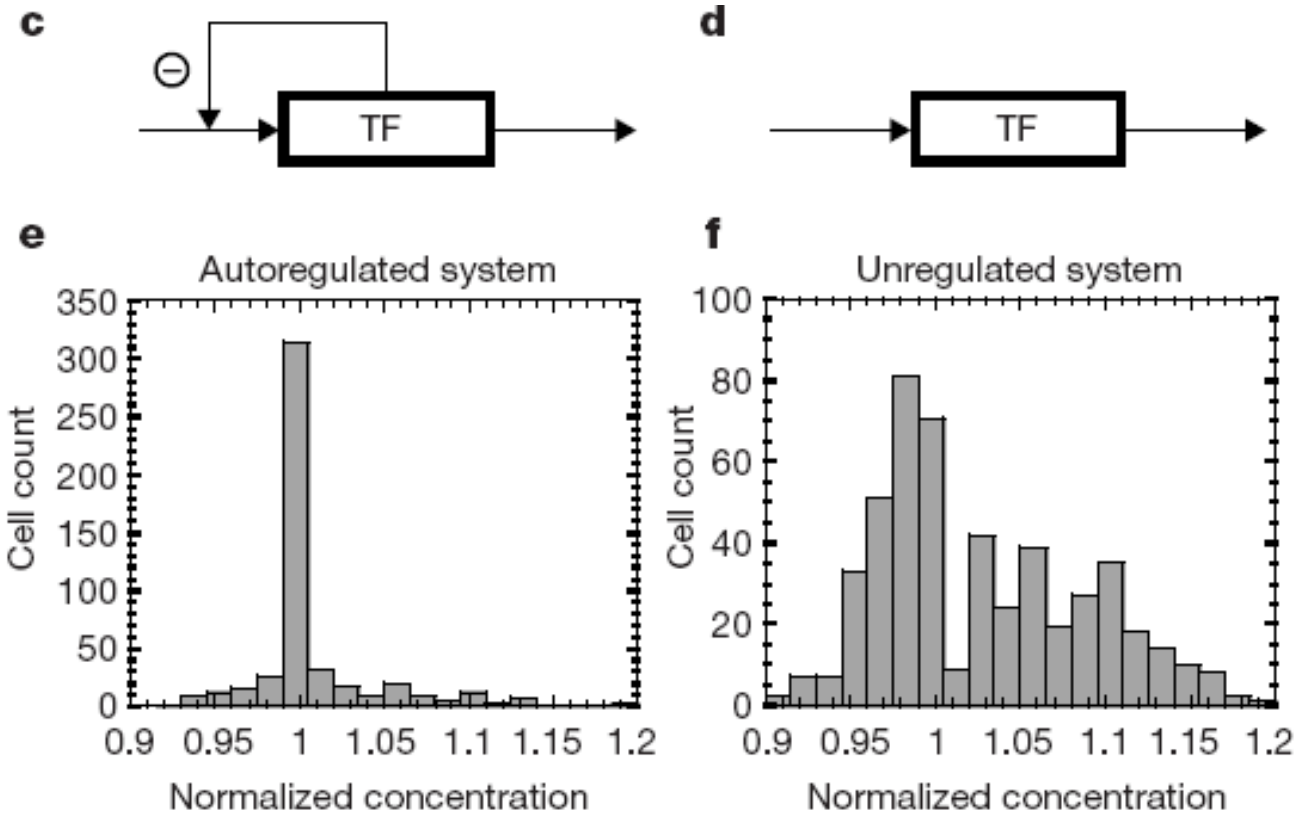
M. Kaern & R. Weiss, in Szallasi / Periwal / Stelling (eds.) *Systems modelling in cell biology*, MIT Press (2006).

Negative Feedback: Functions

- Simple negative feedback systems:
  - Approaching steady state (transient dynamics).
  - Existence of a unique steady state.

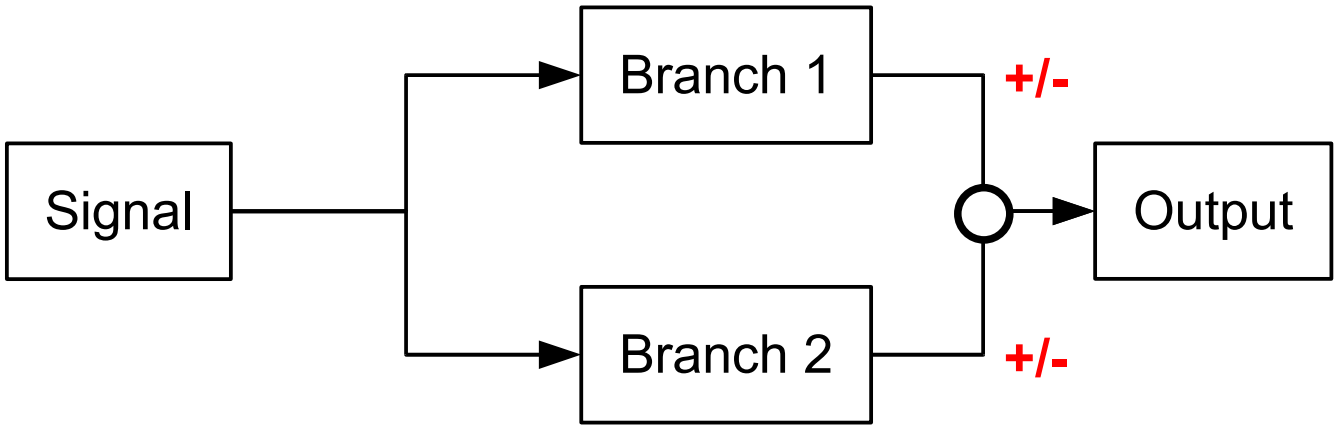
- Functions in biological networks:
  - Set point regulation → Homeostasis.
  - Rejection of external or internal perturbations.

Negative Feedback: Realization



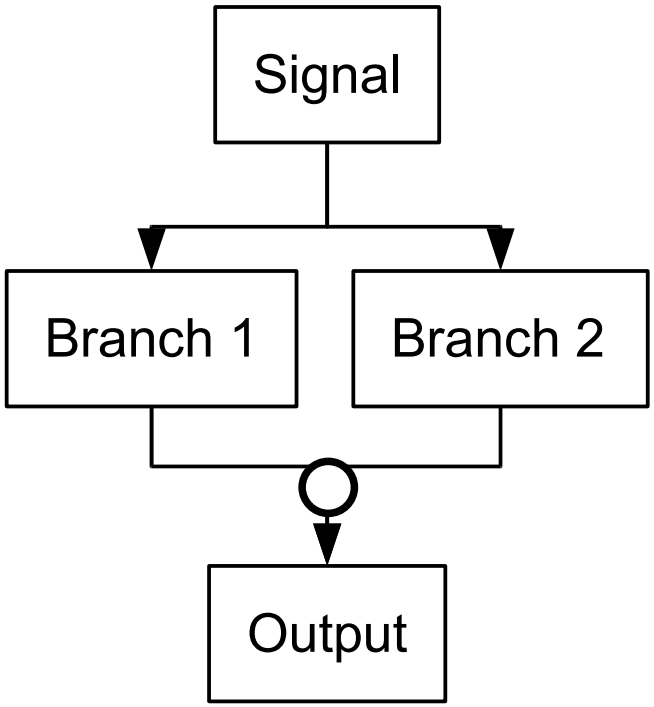
From: Becskei & Serrano (2000) Nature 405: 591-593.

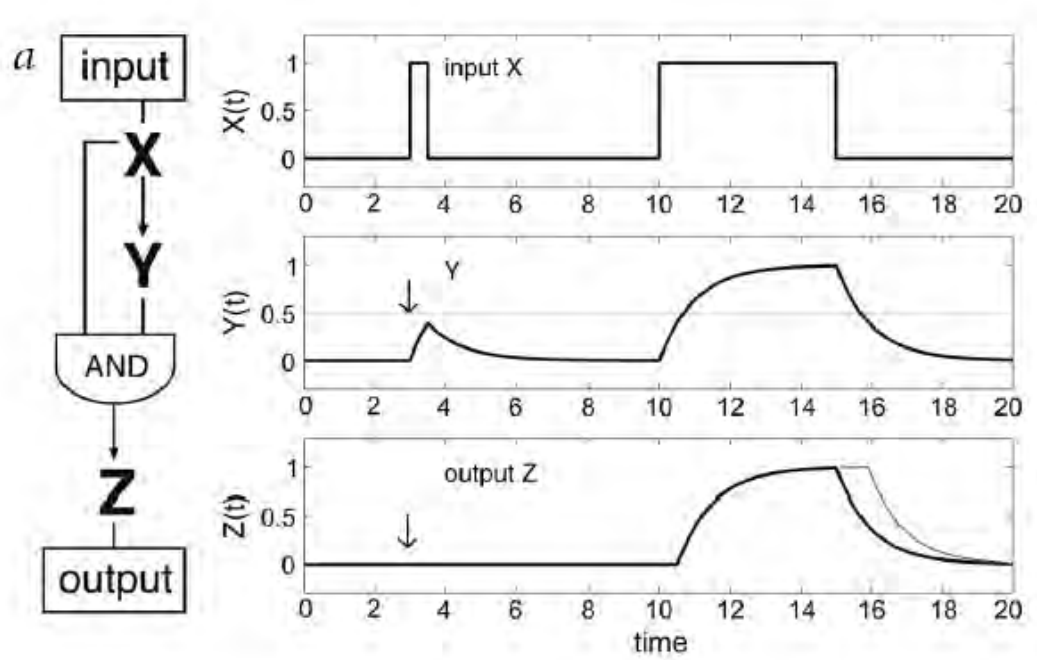
Feedforward Systems



- Common input and output, propagation via separate paths.
- Behavior depends on signs and timing for the branches.

Feedforward Systems: Functions





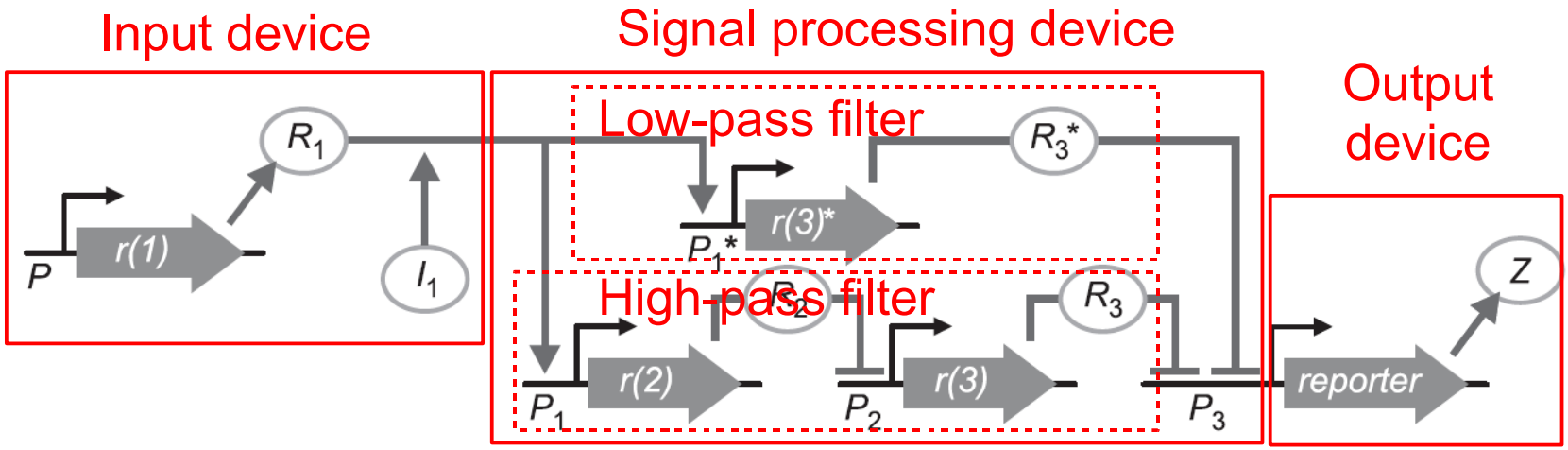
From: Shen-Orr et al. (2002) Nat. Genetics 31: 64-68.

- Positive branch OR delayed negative branch: Pulse generator.
- Negative low-pass NOR negative high-pass: Bandpass filter.
- Positive branch AND positive branch: Low-pass frequency filter.
- Many others: Speed-up of signaling, signal filtering, ...

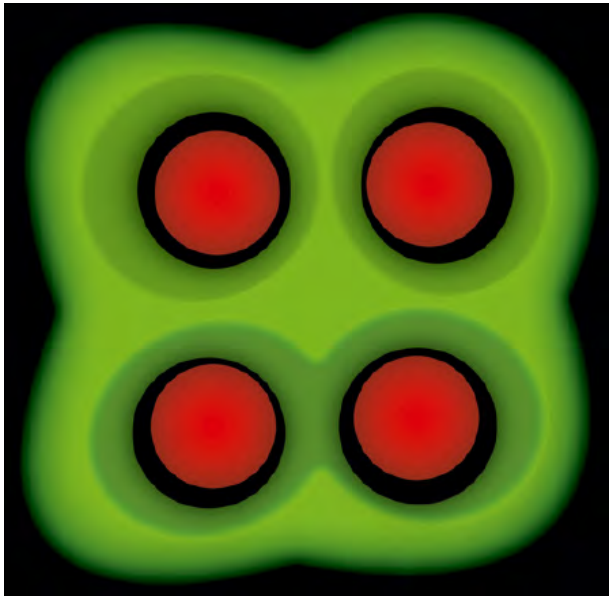
Complex Circuits: Basic Approaches

- Alternative #1: Augmentations at the module level:
  - Additional feedback / feedforward loops.
  - Aim: More complicated systems dynamics.
- Alternative #2: Combination of modules:
  - Modules with defined input / output behavior.
  - More complicated circuits through linking basic elements (cascades, switches, oscillators, ...).

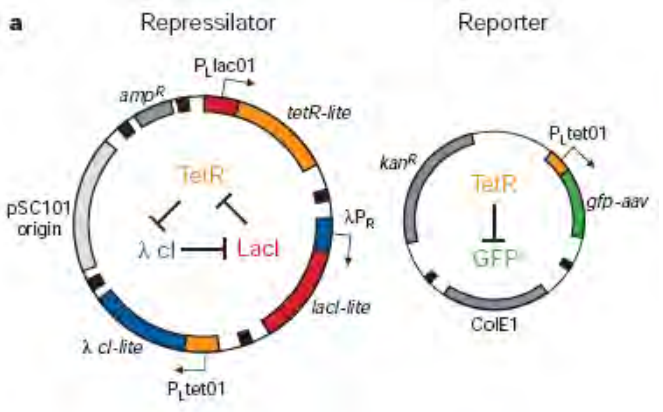
Example: Pattern Generator



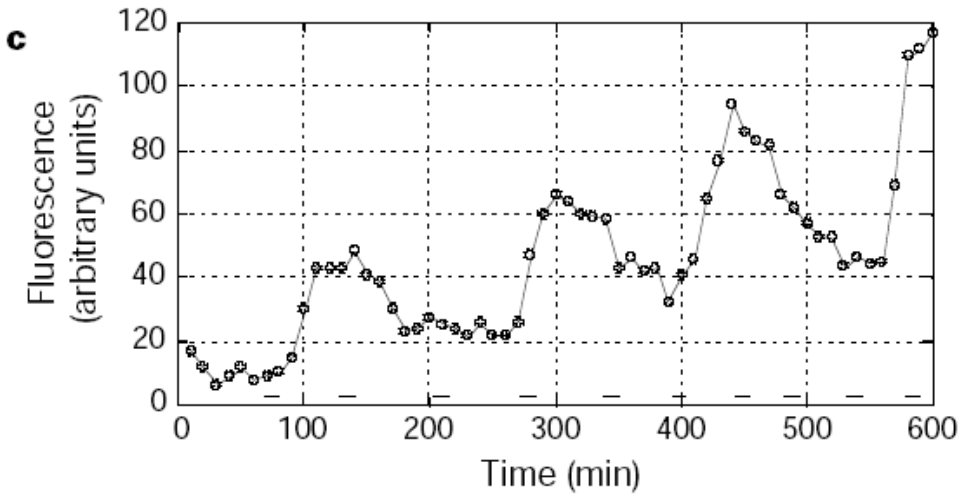
- Combination of simple standard building blocks: Genetic filters.
- Design: Modularization and specific interconnections.



Example: Repressilator

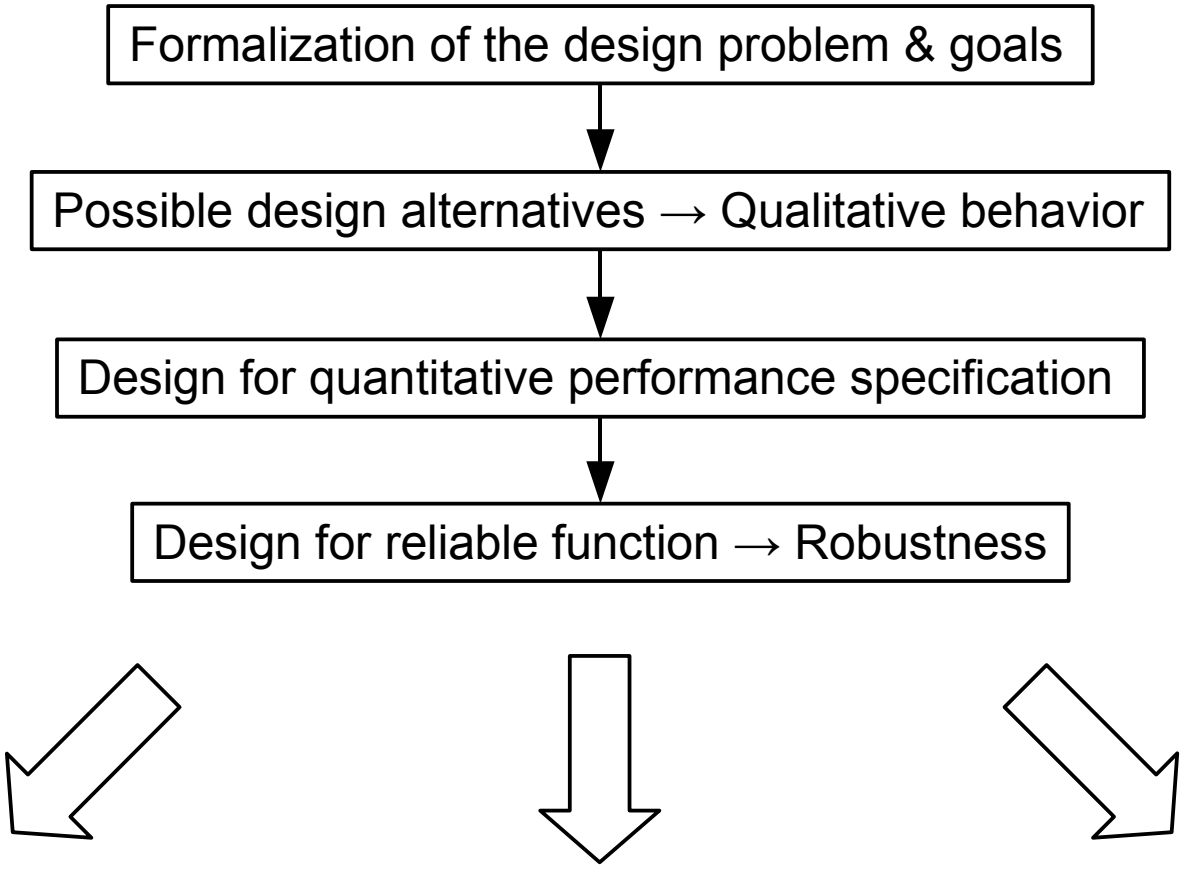


M. Elowitz & S. Leibler, Nature 403:335 (2000).



- Proof-of-principle for oscillator design, yet:
  - Stable oscillations not achieved.
  - High sensitivity to molecular noise.

Challenges: Models & Reality



How to analyze performance?

How to obtain parameters?

How to deal with noise?

Further Reading

- M. Kaern & R. Weiss. Synthetic gene regulatory systems. In: Szallasi / Periwal / Stelling (eds.) *System modeling in cell biology.* (MIT Press, Cambridge / MA) (2006).

- J.J. Tyson, K.C. Chen & B. Novak. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. Curr Opin Cell Biol. 15, 221 – 231 (2003).

- J.L. Cherry & F.R. Adler. How to make a biological switch. J. theor. Biol. 203: 117 – 133 (2000).

- E. Andrianantoandro, S. Basu, D. K. Karig & R. Weiss. Synthetic biology: new engineering rules for an emerging discipline. Molecular Systems Biology 2: 0028 (2006).

Reverse-engineering gene networks

Diego di Bernardo

TIGEM

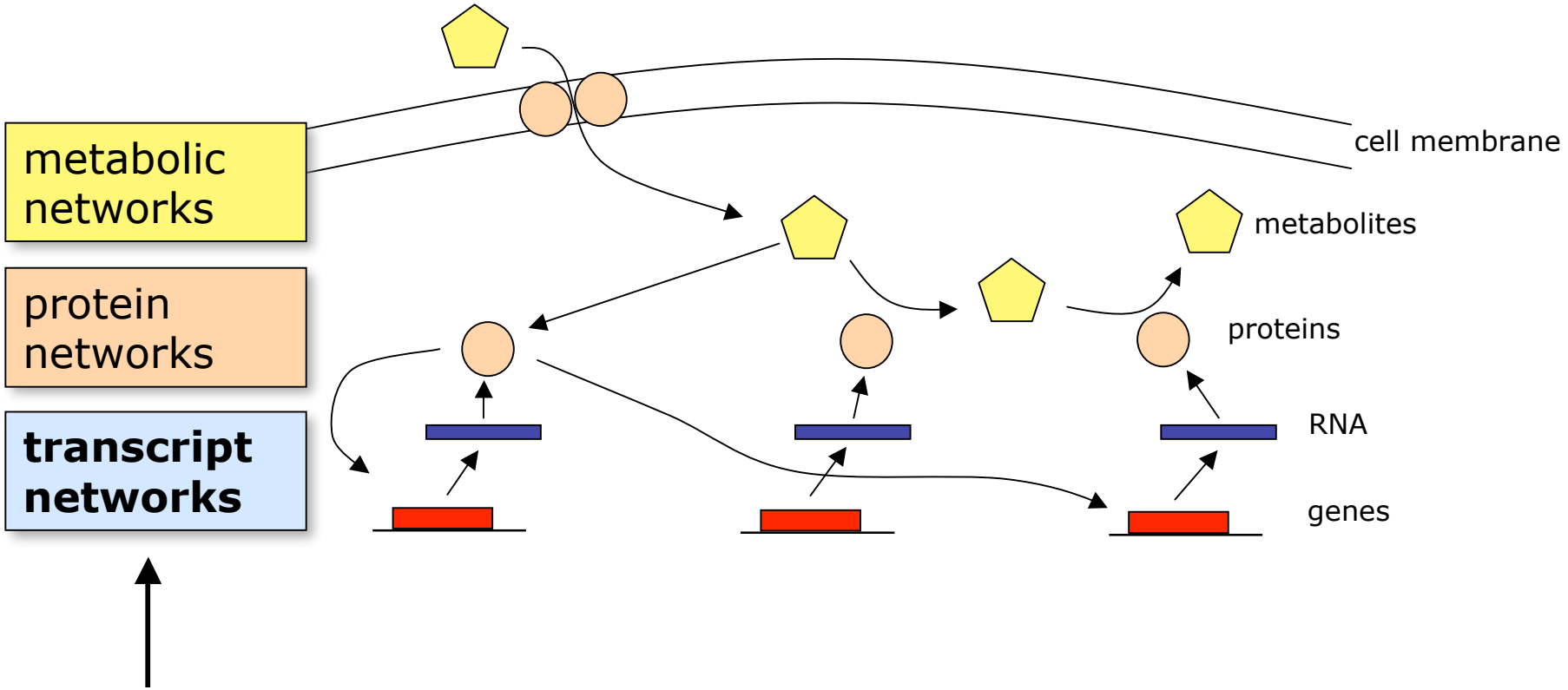
Telethon **I**nstitute of **GE**netics and **M**edicine

www.tigem.it

Overview:

- Networks in Biology
- Reverse-engineering gene networks of unknown topology (de novo)
- Parametrisation of network with known topology

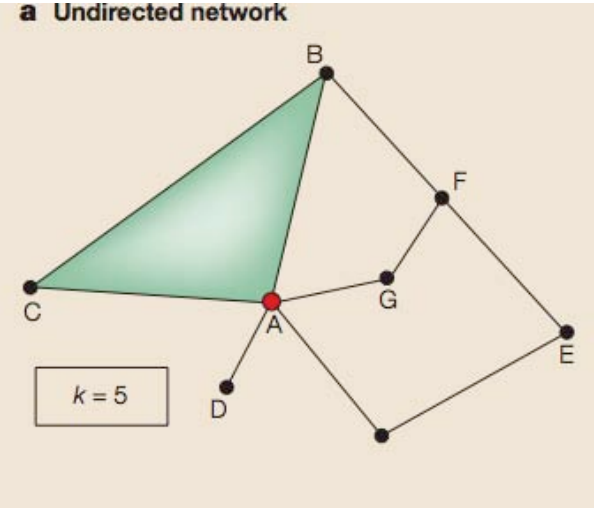
Gene Networks

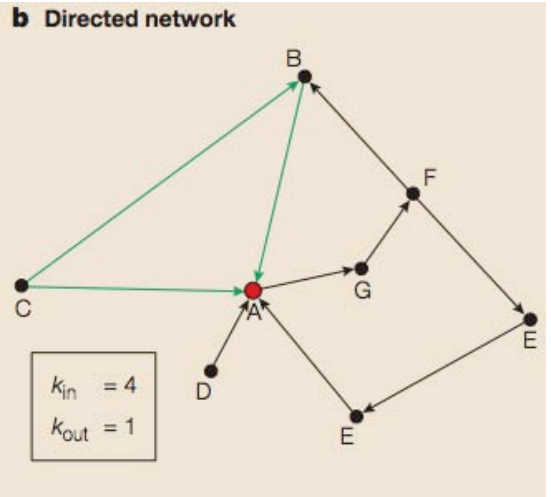


Our focus: methods to decode transcription regulation networks

How can we describe gene interactions: Network theory

- The cell is the result of many sub-components working together
- Graph (network) theory is useful to describe such systems
- Definitions:
  - *graph* G={V,E} where V is a set of verteces or nodes, and E is a set of edges
  - degree k: number of edges connected to a node
  - *digraph*: the edges have a direction
  - *P(k) degree distributin*: probability that a node has degree k: P(k)=N(k)/N
  - *C(k) clustering:* if node A is connected to node B, and B to C, are A and C connected?

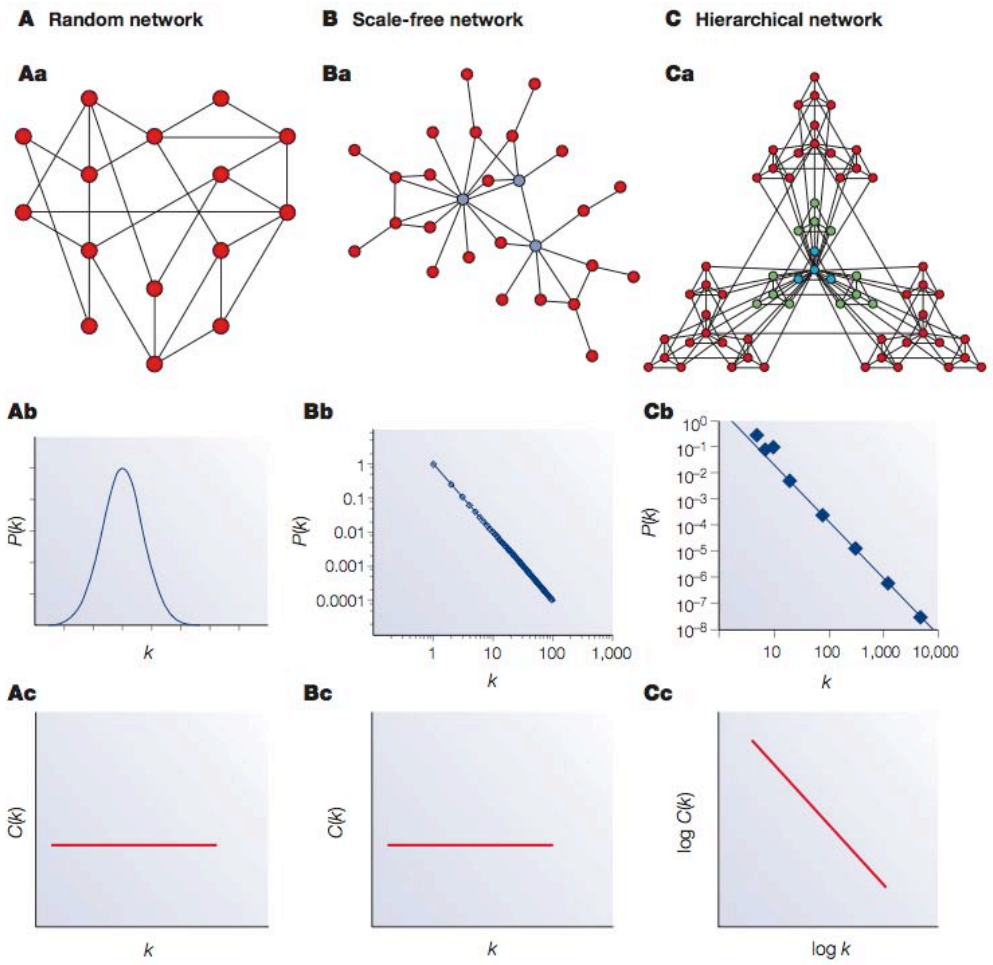




Barabasi et al, Nature Review Genetics, 2004, 5:101: http://www.nd.edu/~networks/PDF/Wuchty03_NatureGenetics.pdf

Types of network

- *Random networks:*
  - *Node have similar degrees*
- *Scale-free networks:*
  - *$P(k)=k^{-g}$ few nodes have a lot of edges (hubs)*
  - *Internet, gene networks, social networks*
- *Hierarchical networks*
  - *Modules*
  - *Scale-free*



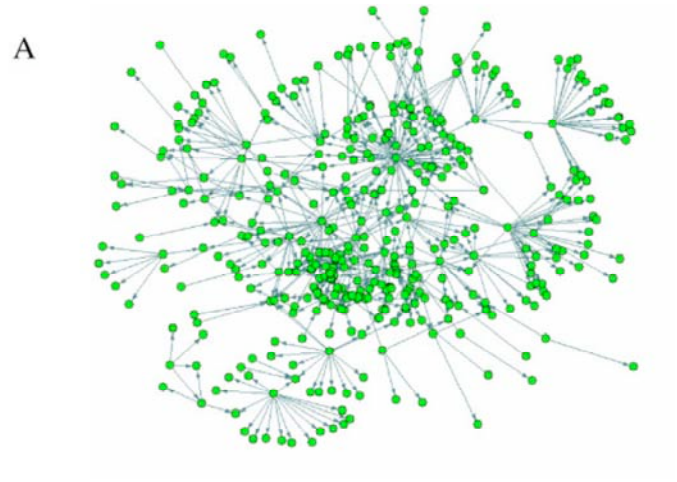
Barabasi et al, Nature Review Genetics, 2004, 5:101: http://www.nd.edu/~networks/PDF/Wuchty03_NatureGenetics.pdf

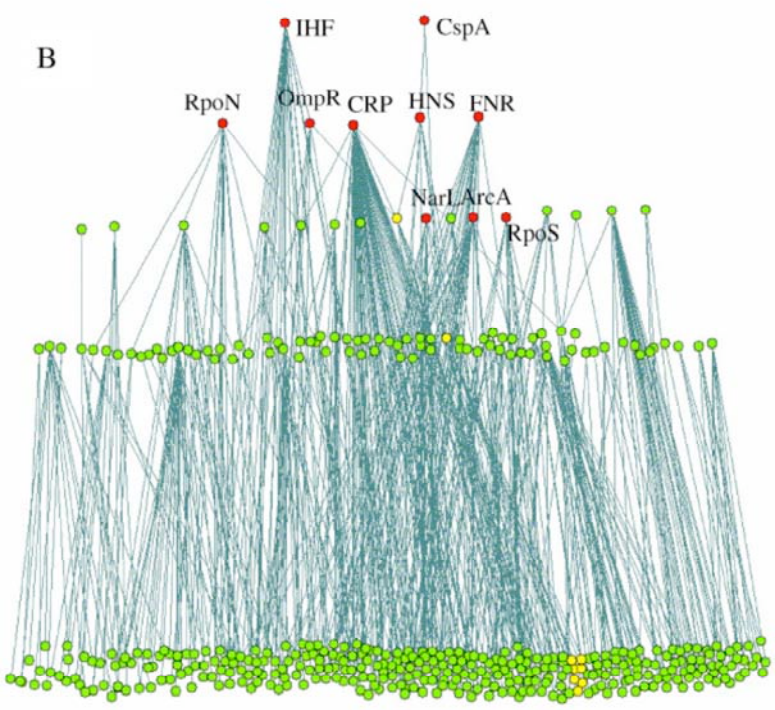
Biological networks

- Biological processes can be represented as networks:
  - Transcriptional networks (protein-DNA)=digraph
    - Nodes: genes and proteins
    - Edges: a TF activaes/inhibits a gene
  - Protein-protein networks = graph
    - Nodes: proteins
    - Edges: the two proteins interact
  - Metabolic networks:
    - Nodes: metabolites
    - Edges: there is an enzyme transforming the two products

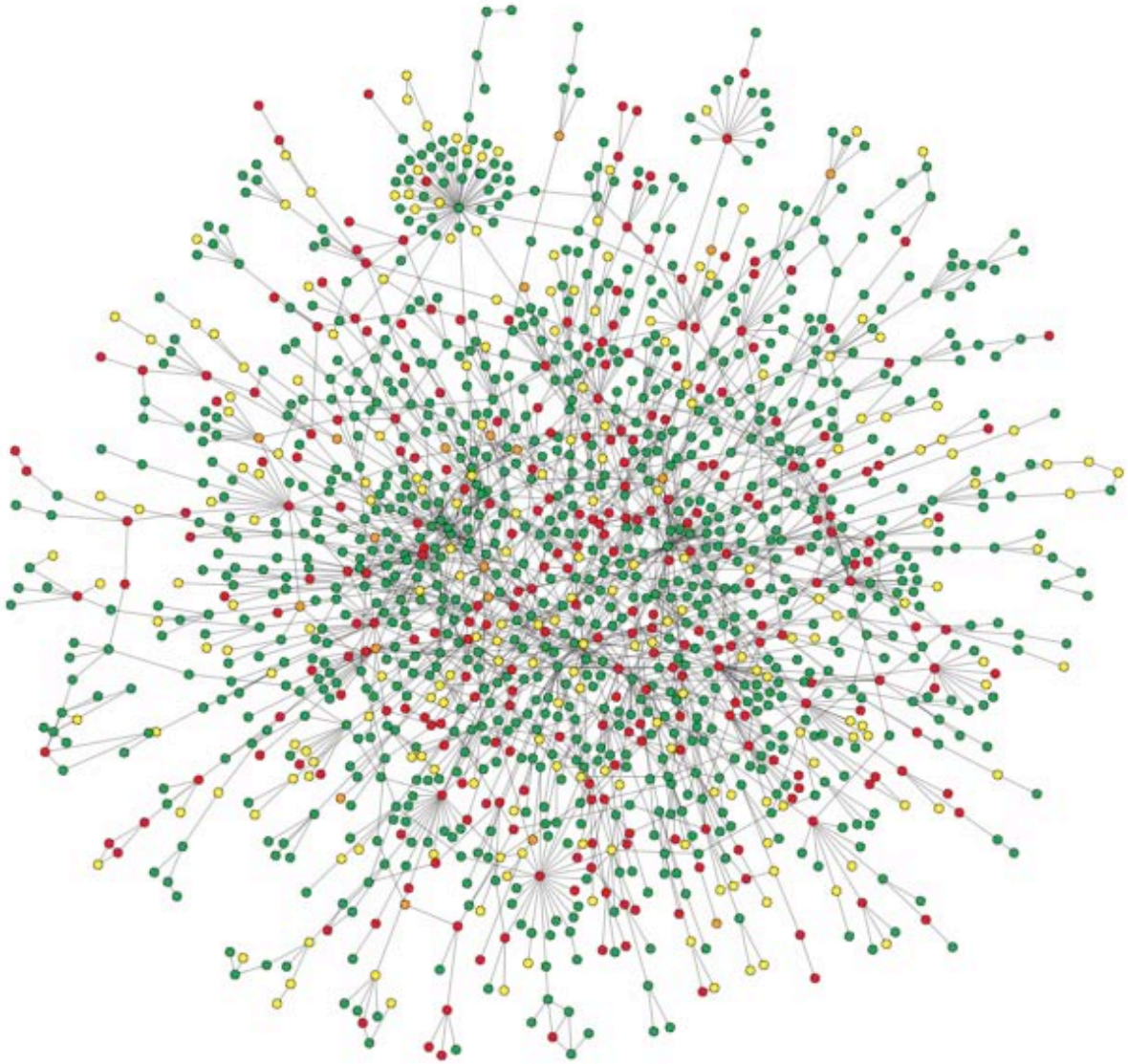
Why “de novo”? example of transcriptional network (E. coli):

- From the structure of the network we can learn its function.
- For synthetic biology: what are the genes that we “replace” in the cell doing?



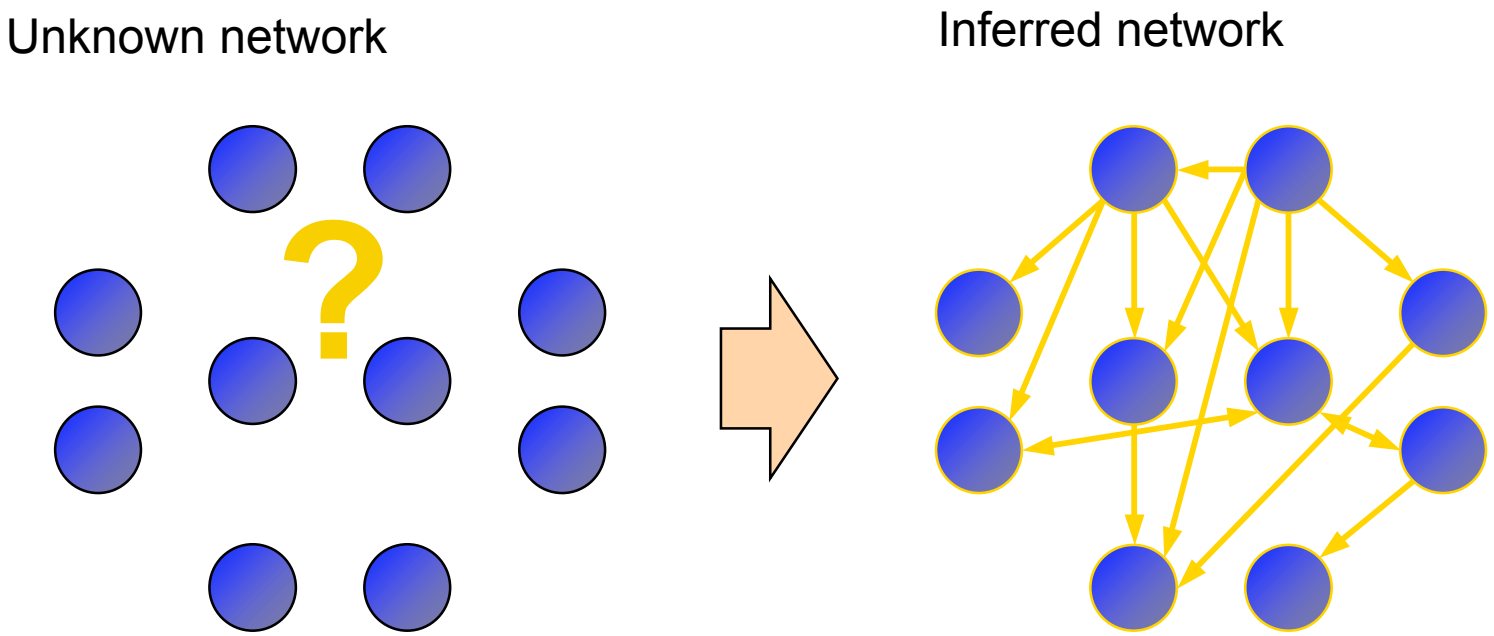


What info can we gain? protein-protein interaction network (yeast S. cerevisiae)

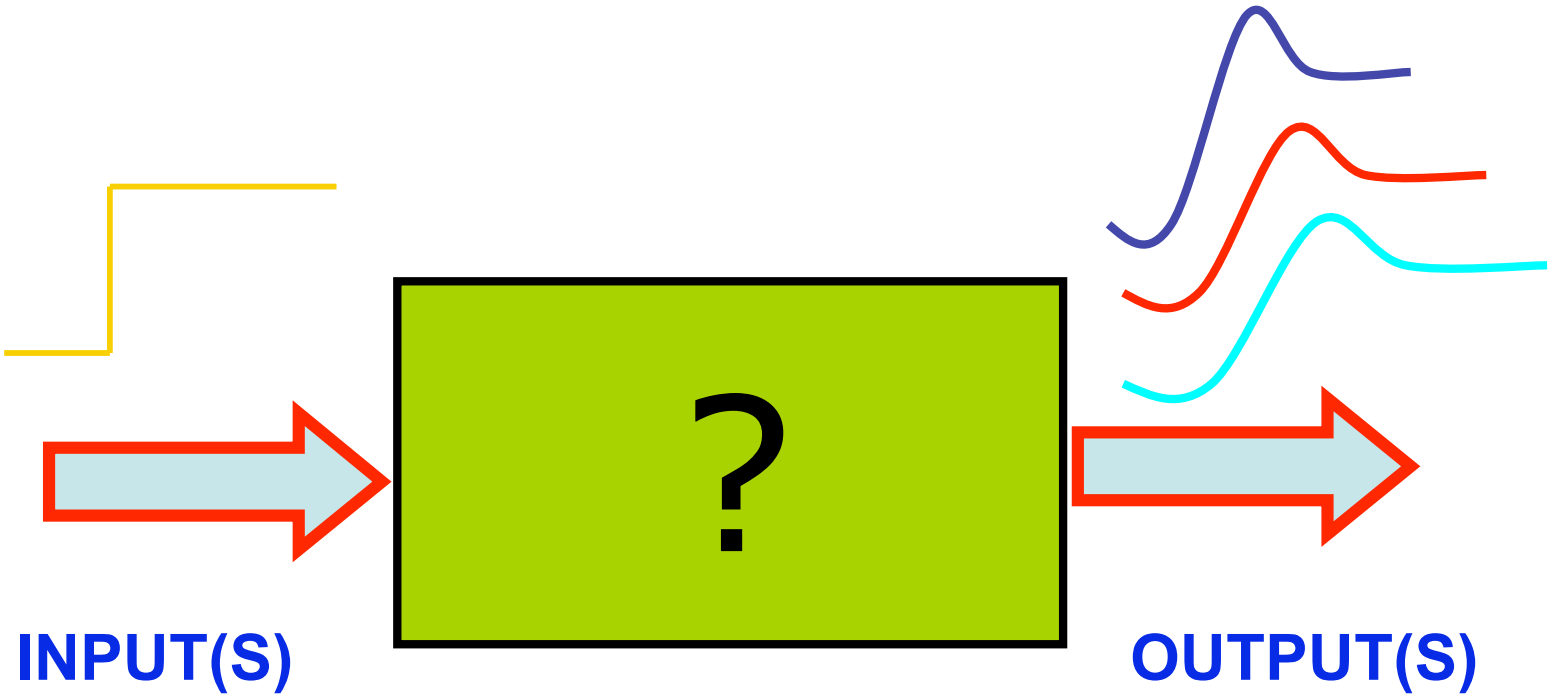


dibernardo.tigem.it

Reverse engineering (or inference) gene networks:



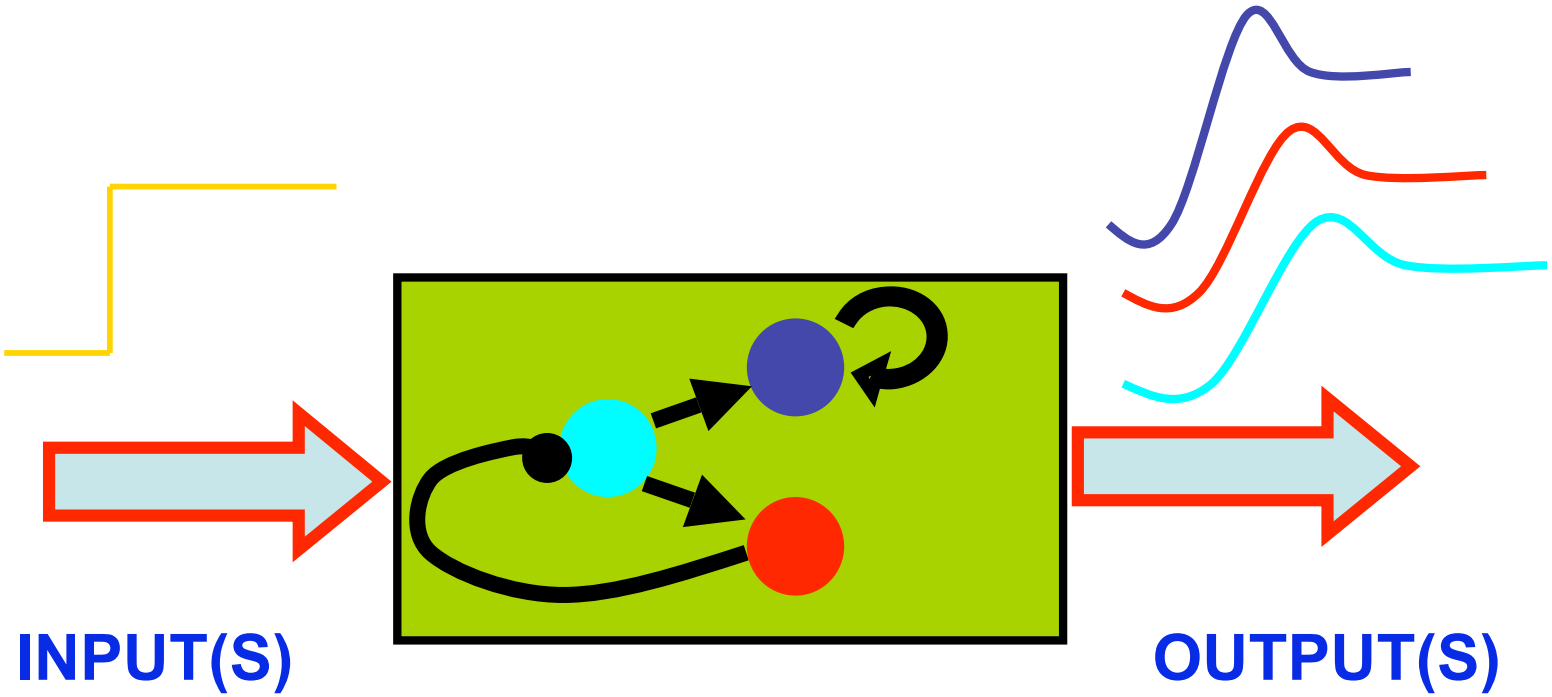
“System Identification” or “reverse engineering”



Input: perturbations to the system (i.e. gene overexpression)

Output: measure response to perturbations (40’000 genes)

To infer a network means to find what is inside the “black box”

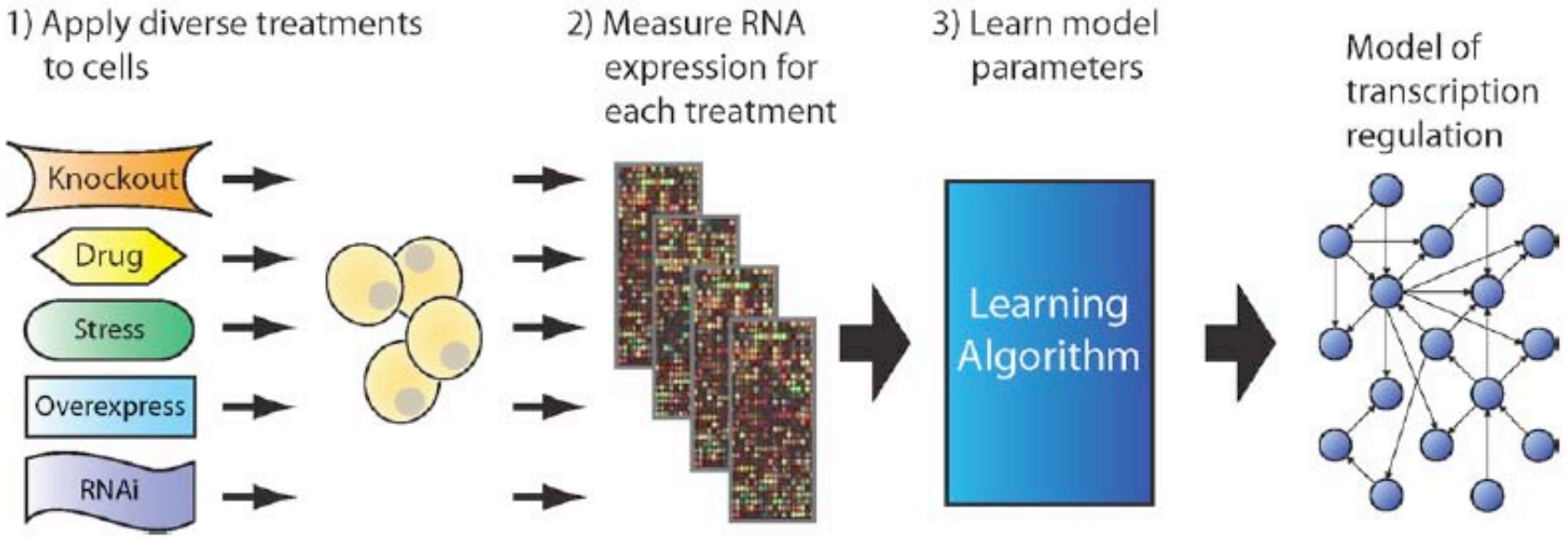


Measuring cell activity: experimental methods

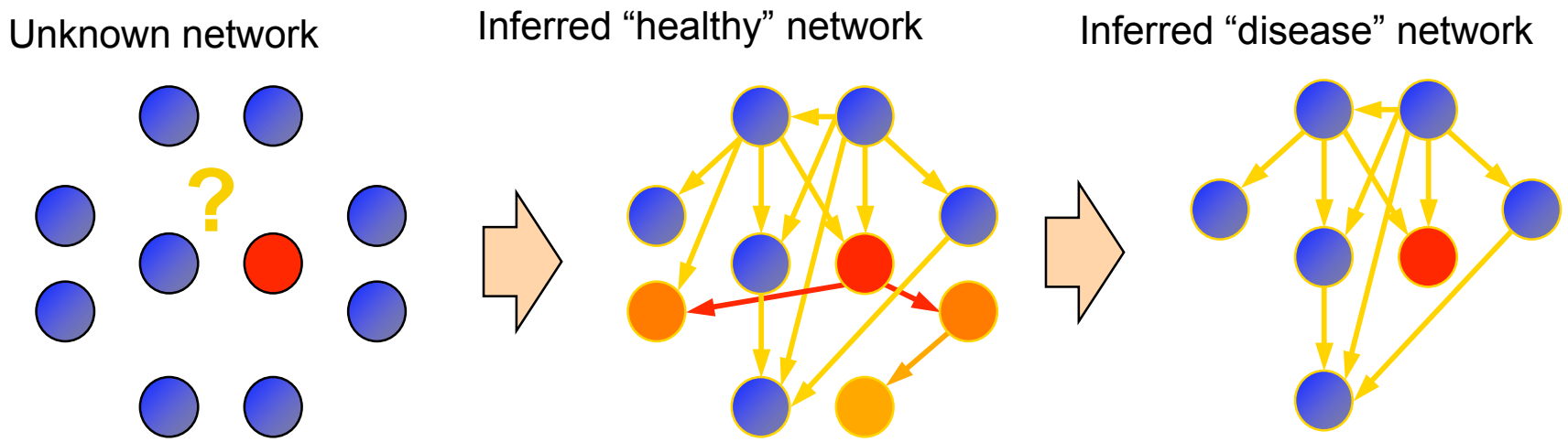
- We need to measure input and output of the cell to tackle the identification process:
  - There are at least 40'000 genes, i.e. 40'000 species of mRNA and 40'000 species of proteins…and counting
  - A "revolution" has been the creation of microarrays to measure mRNAs levels simultaneously for all the genes
  - This is not yet possible for proteins or metabolites…but we are almost there…

Reverse engineering gene networks

Goal: Learn structure and function from expression data



Reverse-engineering networks can help in understanding the disease:

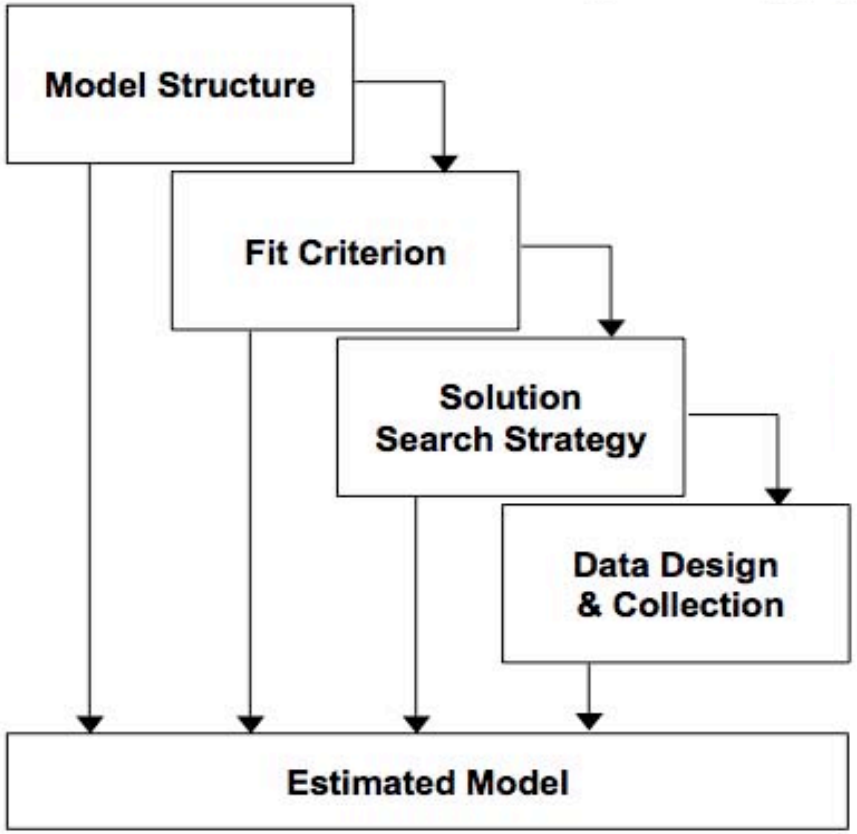


Methods to reverse-engineer gene networks:

- Given the experimental data, how can we reverse-engineer the network?

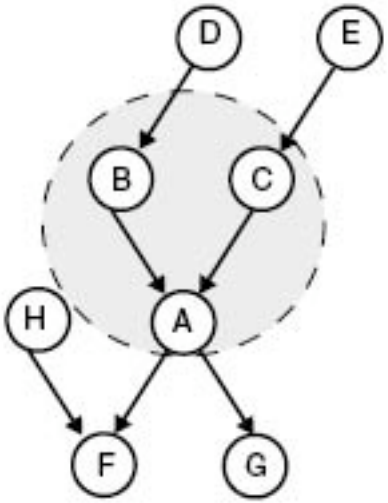
Reverse-engineering strategy:

- Choose a model
- Choose a fit criterion (cost function) to measure the fit of the model to the data
- Define a strategy to find the parameters that best fit the data (i.e. that minimise cost function)
- Perform appropriate experiments to collect the experimental data:



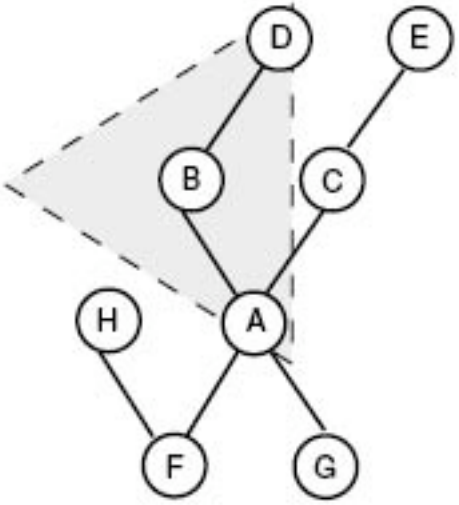
Reverse-engineering strategy:

Bayesian Networks



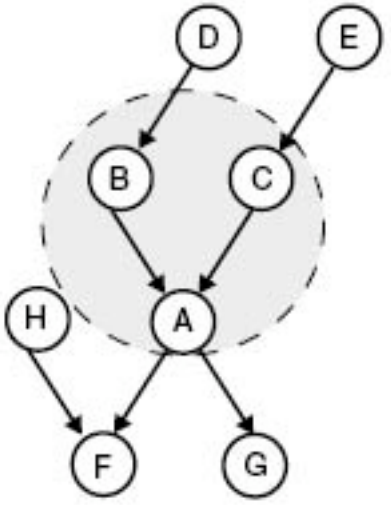
$P(A/B,C,D,E)=P(A/B,C)$

Information-theoretic



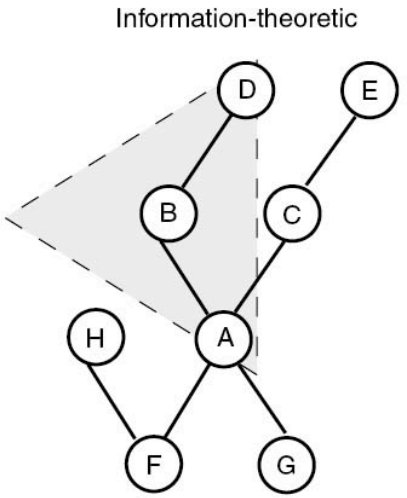
$MI(A,H)=0$
$MI(A,B)>0$
$0<MI(A,D)\leq \min\{MI(A,B), MI(B,D)\}$

Ordinary differential equations



$dA/dt=\theta_1 A+\theta_2 B+\theta_3 C$
or more generally:
$dA/dt=f(A,B,C,\theta)$

Reverse-engineering strategy: Information-theoretic approach



MI(A,H)=0
MI(A,B)>0
0<MI(A,D)≤min{MI(A,B), MI(B,D)}

- Assume that the joint probability can be computed as a combination of 2nd order probabilities, i.e. look only at pair of genes.
- Compute Mutual Information I(x,y) for a pair of gene:

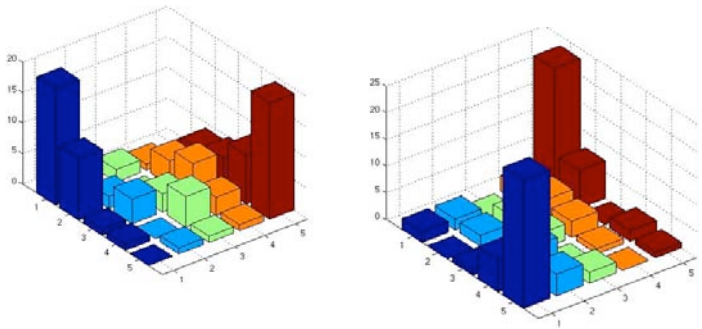
$$\mathrm{MI}_{i,j} = H_i + H_j - H_{ij}$$

where H, the entropy, is defined as:

$$H_i = -\sum_{k=1}^{n} p(x_k)\log(p(x_k))$$

- The MI can be computed directly as:

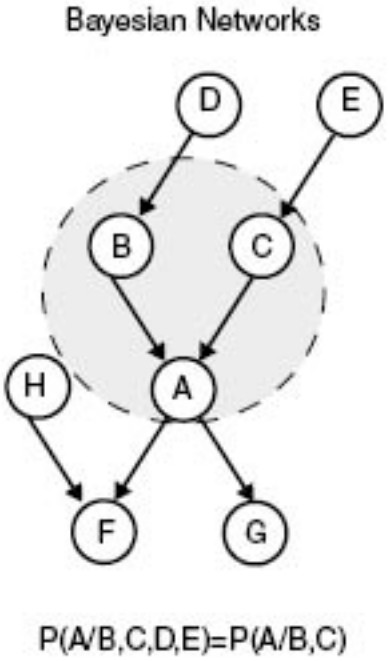
$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} p(x,y)\ln\frac{p(x,y)}{p(x)p(y)}$$



- In practice:

$$I(\hat{\pi}) = \sum_{ij}\frac{n_{ij}}{n}\log\frac{n_{ij}n}{n_{i+}n_{+j}}$$

Reverse-engineering strategy: Bayesian Networks



- Using the Markov rule

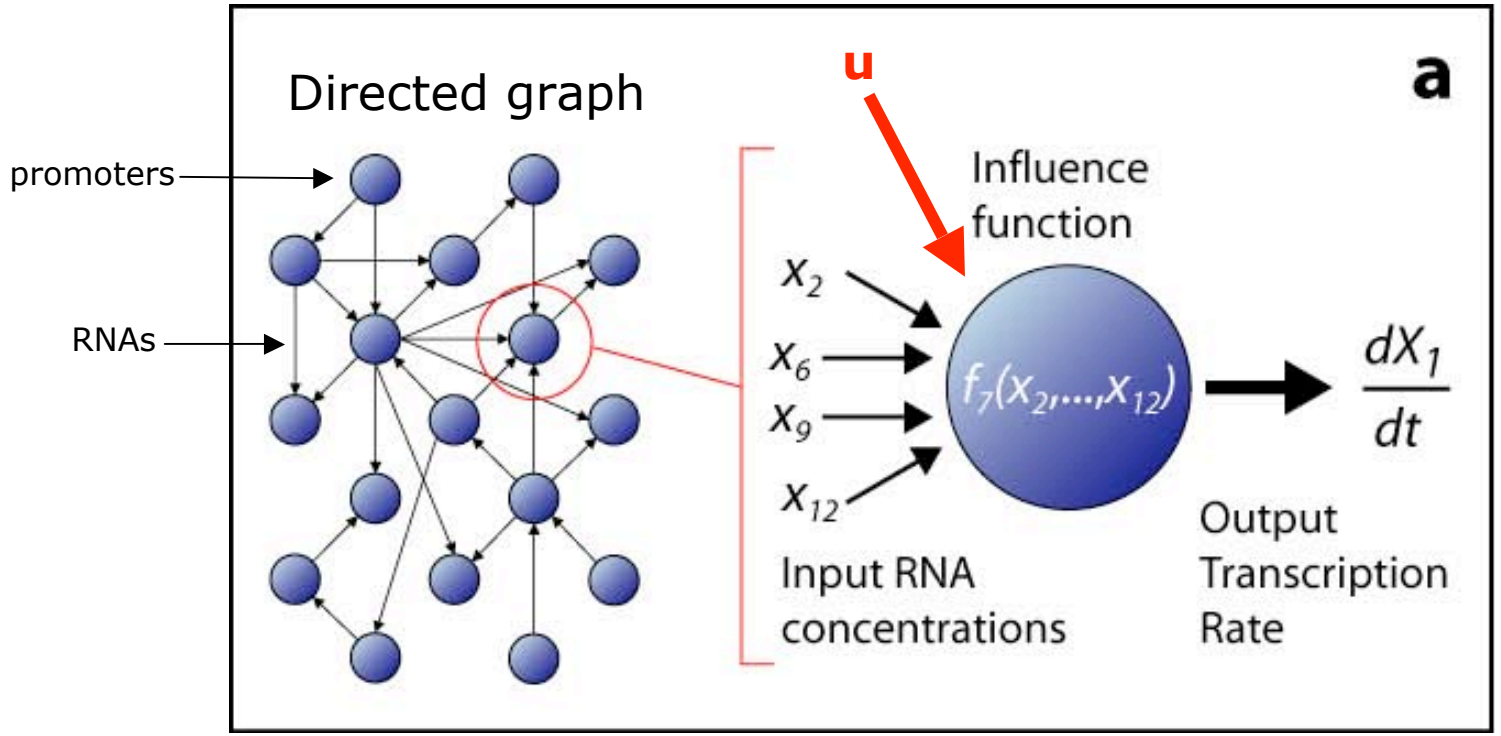
$$P(X_1,\ldots,X_n)=\prod_{i=1}^{N}P(X_i=x_i||X_j=x_j,\ldots,X_{j+p}=x_{j+p})$$

- Choose a network topology G
- Compute joint probability function P(D/G)
- Score each network (i.e. BDe)

$$P(G/D)=\frac{P(D/G)*P(G)}{P(D)}$$

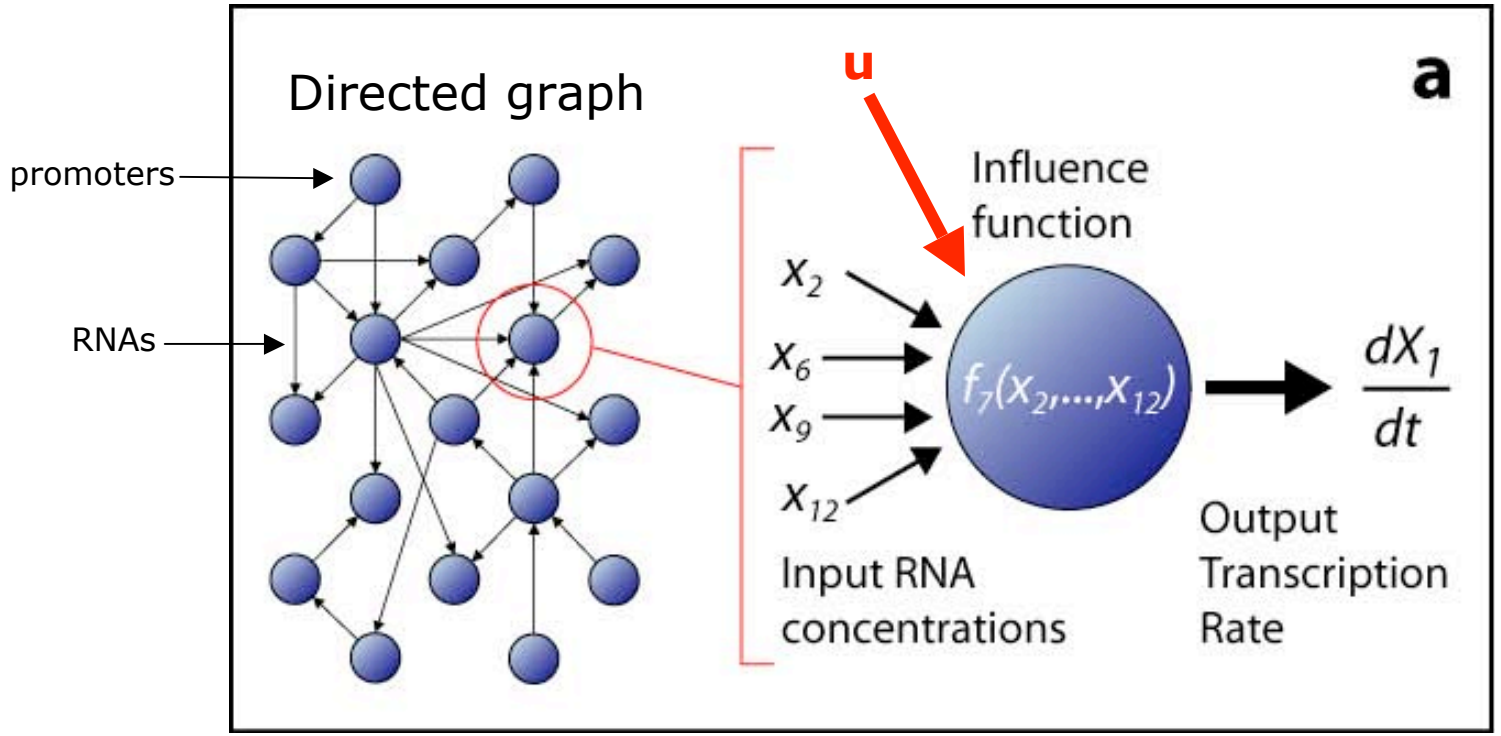
- Iterate the above steps and choose among the networks the one with highest score

Reverse-engineering strategy: ODEs



$$dX_1/dt = 0 = a_2 X_2 + a_6 X_6 + a_9 X_9 + a_{12} X_{12} + u$$

Reverse-engineering strategy: ODEs



$$dX_1/dt = 0 = a_2 X_2 + a_6 X_6 + a_9 X_9 + a_{12} X_{12} + u$$

Model structure:

$$\begin{cases} x'_{11}(t) = a_{11}x_{11}+a_{12}x_{21}+\ldots+a_{1n}x_{n1} + u_1 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{Overexpression of gene 1} \\ x'_{n1}(t) = a_{n1}x_{11}+a_{n2}x_{21}+\ldots+a_{nn}x_{n1} + 0 \end{cases}$$

x_{ij} i:gene number j: experiment number

Or in matrix format:

$$x'=Ax+u$$

?

Model structure:

$$\begin{cases} x'_{11}(t) = a_{11}x_{1n} + a_{12}x_{2n} + \ldots + a_{1n}x_{nn} + 0 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{ Overexpression of gene n} \\ x'_{n1}(t) = a_{n1}x_{1n} + a_{n2}x_{2n} + \ldots + a_{nn}x_{nn} + u_n \end{cases}$$

x_{ij} i:gene number j: experiment number

Or in matrix format:

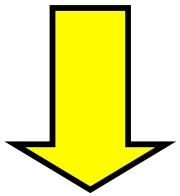
$$x' = Ax + u$$

?

Fit criterion and search solution strategy:

- Perturb one gene x_i at at time and measure the response of the other genes **at steady-state**:

$$\underline{x}'(t) = 0 = A\,\underline{x} + \underline{u}$$



$$A\,\underline{x} = -\underline{u}$$

- Repeat the experiment overexpressing all of the N genes:

$$A\,X = -U \quad \Rightarrow A = -UX^{-1}$$ *not robust to noise*

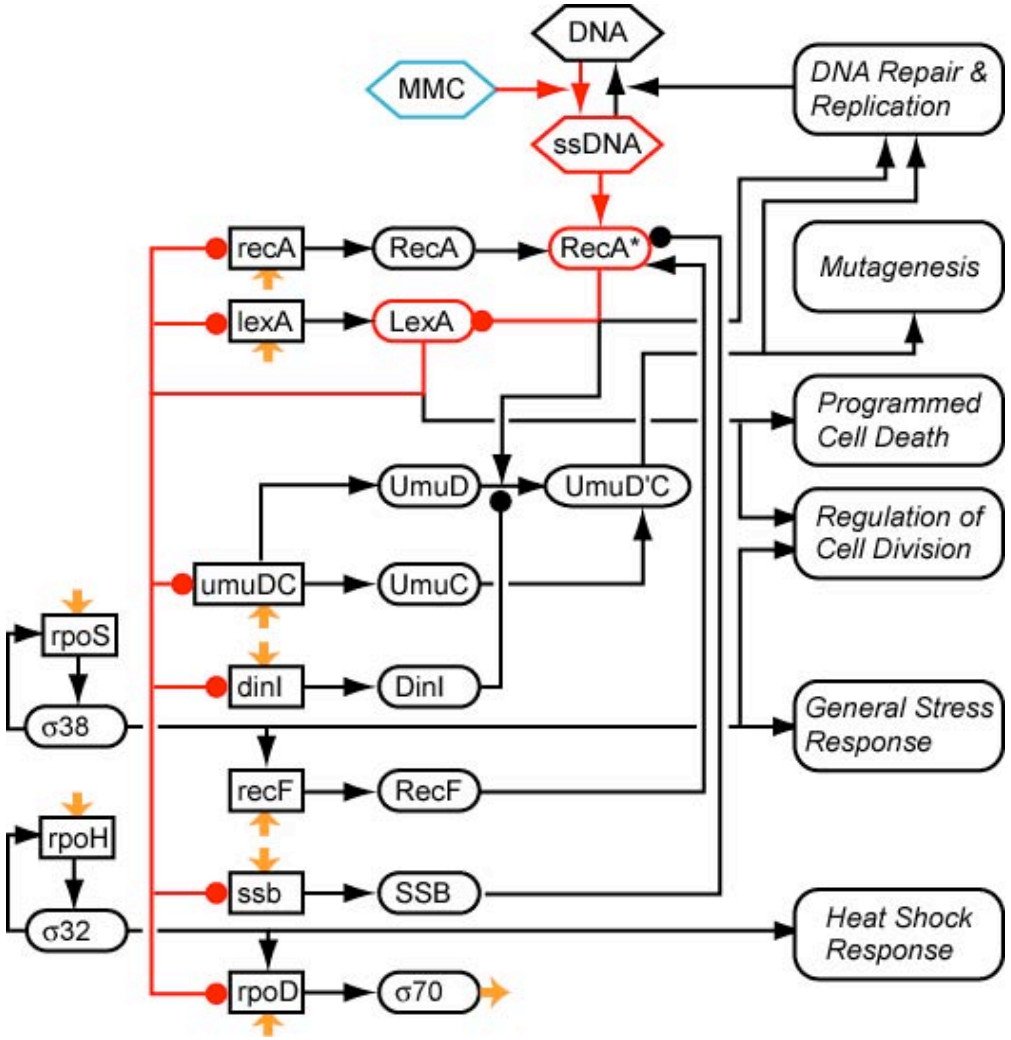
A (N x N), X (N x N), U (NxN)

? known known

Pilot study: *E. coli* DNA-damage repair pathway (SOS pathway)

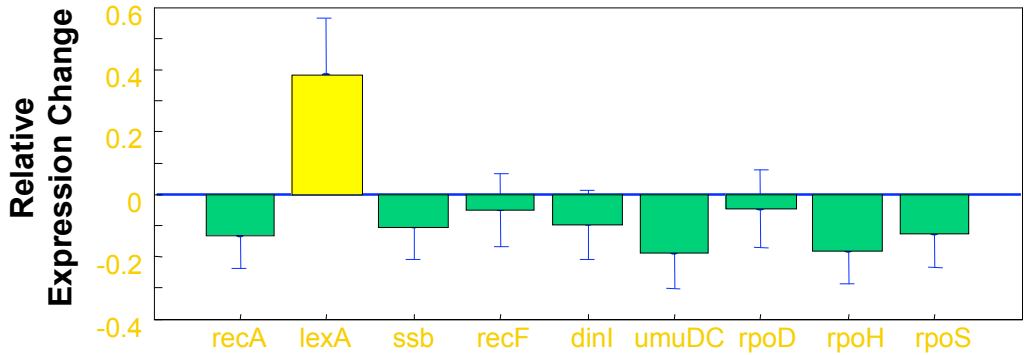
DNA-damage repair potentially involves 100s of genes

Applied NIR to 9 transcript subnetwork



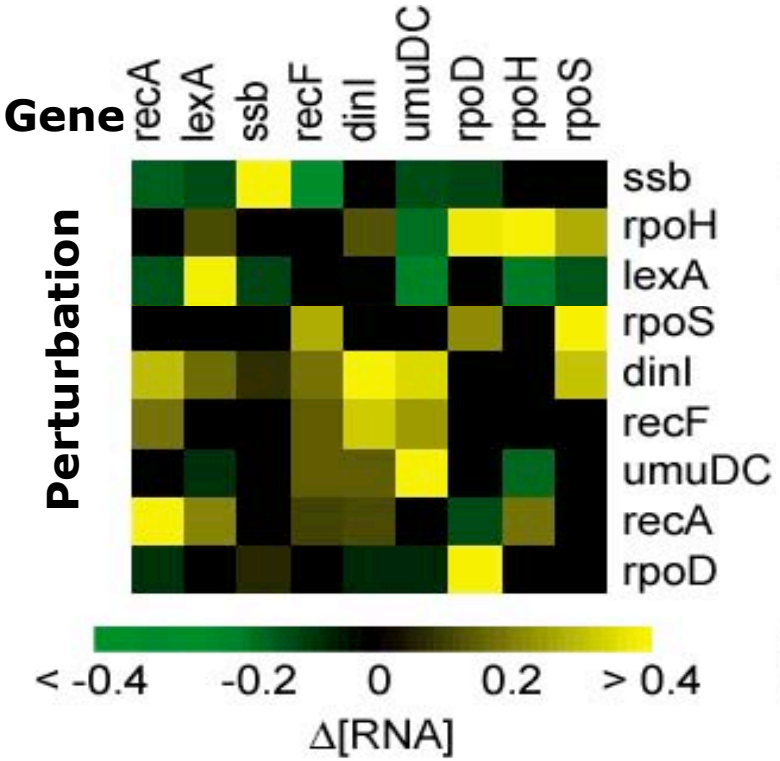
Data design and collection:

Example perturbation: lexA



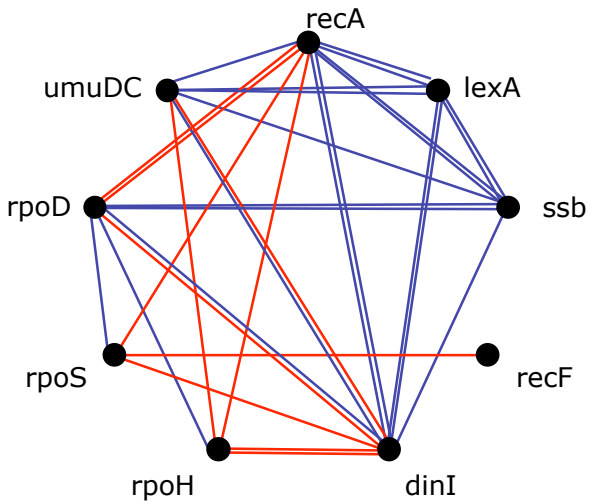
7-9 training perturbations used to recover 9 gene SOS subnetwork

Insignificant changes set to zero during data preprocessing



SOS subnetwork model identified by NIR

Graphical model



Quantitative regulatory model

Connection strengths

	recA	lexA	ssb	recF	dinI	umuD	rpoD	rpoH	rpoS
recA	0.40	-0.18	-0.01	0	0.10	0	-0.01	0	0
lexA	0.39	-0.67	-0.01	0	0.09	-0.07	0	0	0
ssb	0.04	-1.19	-0.28	0	0.05	0	0.03	0	0
recF	0	0	0	0	0	0	0	0	0
dinI	0.28	0	0	0	-1.09	0.16	-0.04	0.01	0
umuDC	0.11	-0.40	-0.02	0	0.20	-0.15	0	0	0
rpoD	-0.17	0	-0.02	0	0.03	0	-0.51	0.02	0
rpoH	0.10	0	0	0	0.01	-0.03	0	0.52	0
rpoS	0.22	0	0	-1.68	0.67	0	0.08	0	-2.92

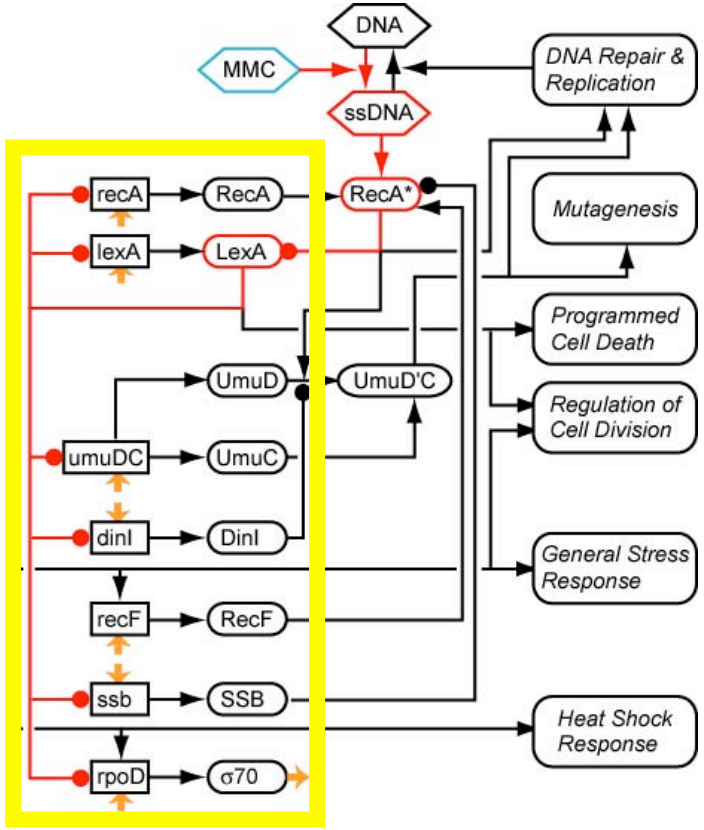
Majority of previously observed influences discovered despite high noise (68% N/S)

Methods to find parameters of known networks:

- Given the experimental data, how can we find physical parameters of a known network?

- Known network means that:
  - We known the topology
  - We know the kind of interaction (protein-dna; protein-protein; rna-rna; etc.)

Parameter fitting strategy: ODEs



Assigning numbers to the arrows: Parameterizing a gene regulation network by using accurate expression kinetics

Michal Ronen[†], Revital Rosenberg[†], Boris I. Shraiman[‡], and Uri Alon[†§¶]

Departments of [†]Molecular Cell Biology and [§]Physics of Complex Systems, Weizmann Institute of Science, Rehovot 76100, Israel; and [‡]Bell Laboratories, Lucent Technologies, Murray Hill, NJ 07974

- Build network model (known topology)
- Measure mRNA (or protein) levels
- Find parameters of your model:

$$\dot{X}_{ij}(t) = \frac{\beta_i}{1 + (A_j(t)/k_i)}$$

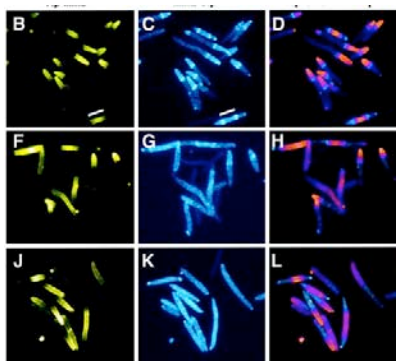
unknowns

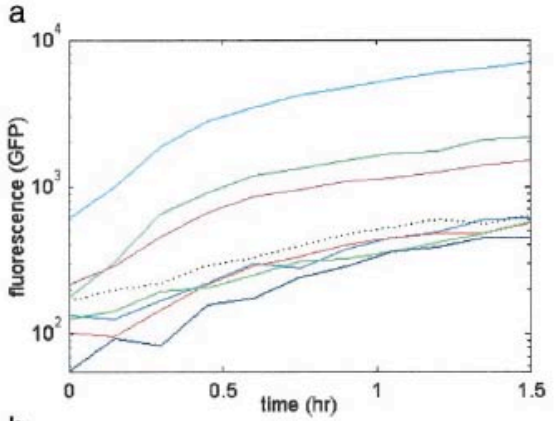
$$1/\dot{X}_i(t) = u_i(t) = a_i \cdot A(t) + b_i$$

knowns

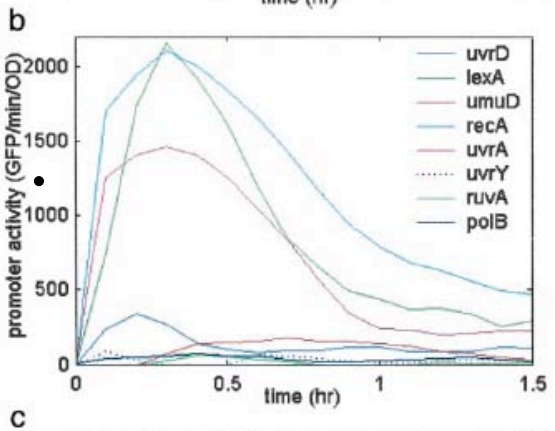
- For N genes, we have 2N unknown with M equations, if we choose M>=2N we can solve the problem with linear algebra.
- More complex cases (non-linear in the parameters) require optimisation techniques like Simulated Annealing

Parameter fitting strategy: ODEs

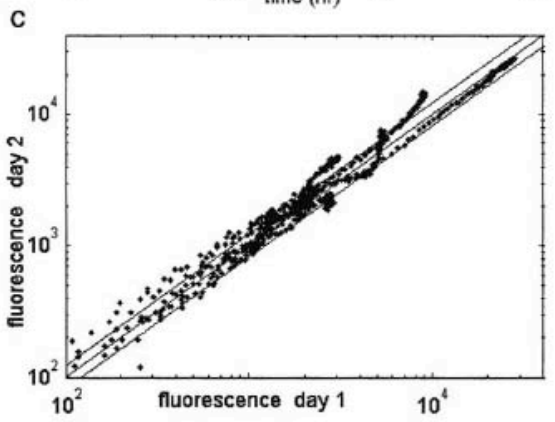




$X_{ij}(t)$



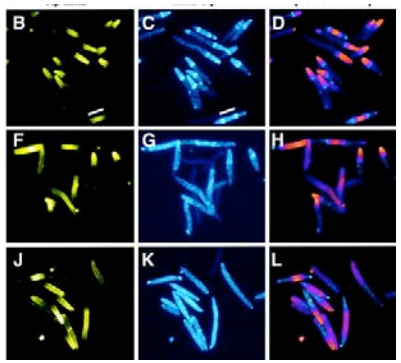
$X_{ij}(t)$

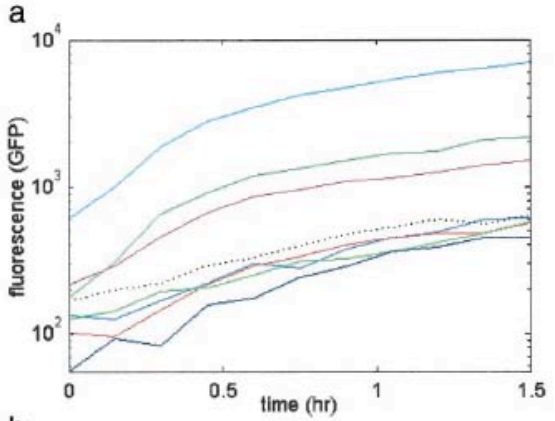


- CASE 1 A(t) activity of protein LexA is known:
  - For N genes, we have 2N unknown with M equations, if we choose M>=2N we can solve the problem with linear algebra.
  - More complex cases (non-linear in the parameters) require optimisation techniques like Simulated Annealing

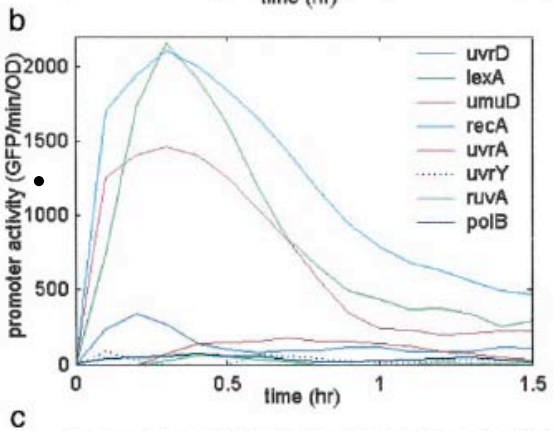
$$1/\dot{X}_i(t) = u_i(t) = a_i \cdot A(t) + b_i$$

Parameter fitting strategy: ODEs

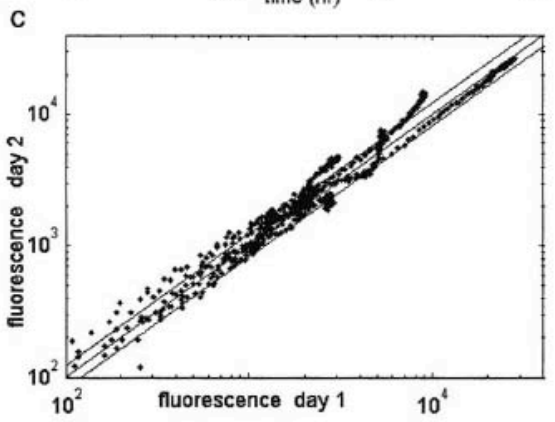




$X_{ij}(t)$



$X_{ij}(t)$



- CASE 2 A(t) activity of protein LexA is not known:
  - For N genes, we have 2N+M unknown with M equations
  - We have an infinity of solutions of dimension 2N
  - We choose one using Singular Value Decomposition

$$1/\dot{X}_i(t) = u_i(t) = a_i \cdot A(t) + b_i$$

Parameter fitting strategy: ODEs

Table 1. The effective kinetic parameters for the SOS system (±SD)

Gene	k	β	E	Function
uvrA	0.09 ± 0.04	2,800 ± 300	0.14	Nucleotide excision repair
lexA	0.15 ± 0.08	2,200 ± 100	0.10	Transcriptional repressor
recA	0.16 ± 0.07	3,300 ± 200	0.12	Mediates LexA autocleavage, blocks replication forks
umuD	0.19 ± 0.1	330 ± 30	0.21	Mutagenesis repair
polB	0.35 ± 0.15	70 ± 10	0.31	Trans-lesion DNA synthesis, replication fork recovery
ruvA	0.37 ± 0.1	30 ± 2	0.22	Double-strand break repair
uvrD	0.65 ± 0.3	170 ± 20	0.20	Nucleotide excision repair, recombinational repair
uvrY	0.51 ± 0.25	300 ± 200	0.45	SOS operon of unknown function, additional roles in two-component signaling
lacZ	—	—	1.53	Unrelated to SOS system

E is the mean error for the promoter activity prediction (see *Methods*).

Connection strengths

	recA	lexA	ssb	recF	dinI	umuD	rpoD	rpoH	rpoS
recA	0.40	-0.18	-0.01	0	0.10	0	-0.01	0	0
lexA	0.39	-0.67	-0.01	0	0.09	-0.07	0	0	0
ssb	0.04	-1.19	-0.28	0	0.05	0	0.03	0	0
recF	0	0	0	0	0	0	0	0	0
dinI	0.28	0	0	0	-1.09	0.16	-0.04	0.01	0
umuDC	0.11	-0.40	-0.02	0	0.20	-0.15	0	0	0
rpoD	-0.17	0	-0.02	0	0.03	0	-0.51	0.02	0
rpoH	0.10	0	0	0	0.01	-0.03	0	0.52	0
rpoS	0.22	0	0	-1.68	0.67	0	0.08	0	-2.92

Our lab: TIGEM, Naples, Italy

Diego di Bernardo

http://dibernardo.tigem.it

Mukesh Bansal (physics)

Giusy Della Gatta (biology)

Giulia Cuccato, Ph.D. (biology)

Francesco Iorio (computer science)

Velia Siciliano (biology)

Vincenzo Belcastro (computer science)

Lucia Marucci (mathematics)

Mario Lauria, Ph.D. (computer science)