Microsoft Research

UNIVERSITY OF OXFORD

The Cell Cycle Switch Computes Approximate Majority

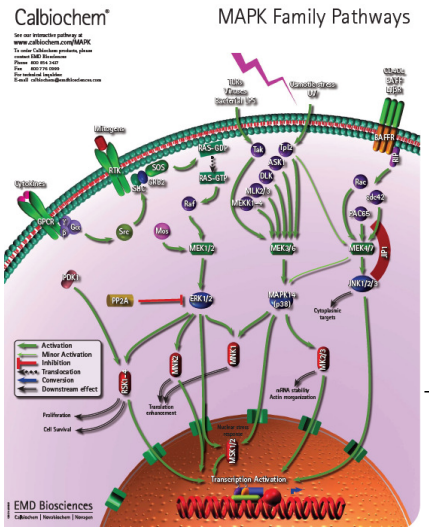
Luca Cardelli, Microsoft Research & Oxford University

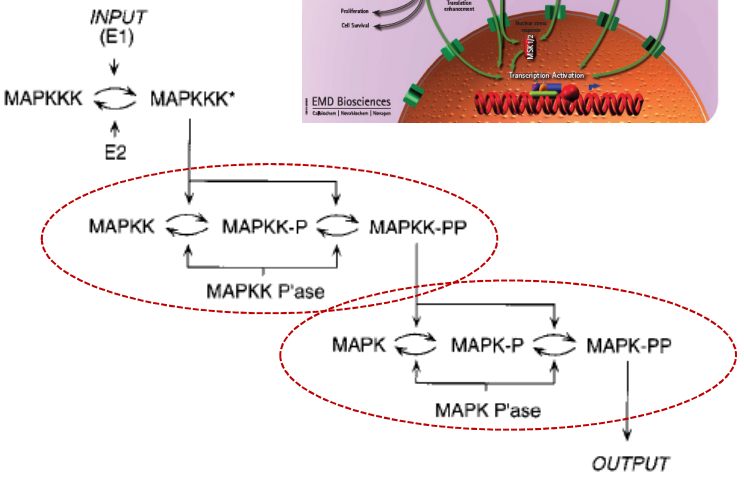
with Attila Csikász-Nagy, King's College London

Caltech, 2015-06-02

Cellular Computation

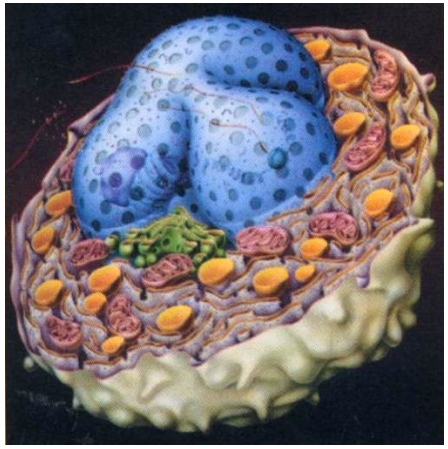
- No survival without computation!
  - Finding food
  - Avoiding predators
- How do cells compute?
  - *Clearly* doing "information processing"
  - What are their computational principles?

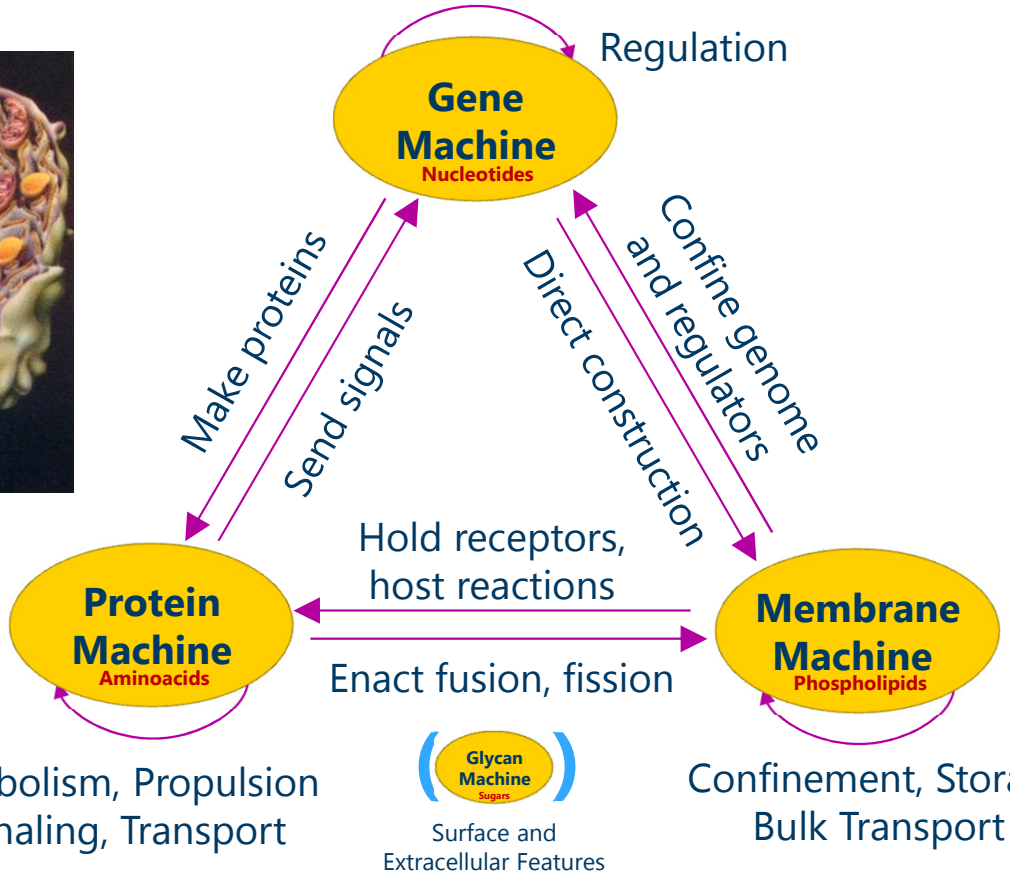




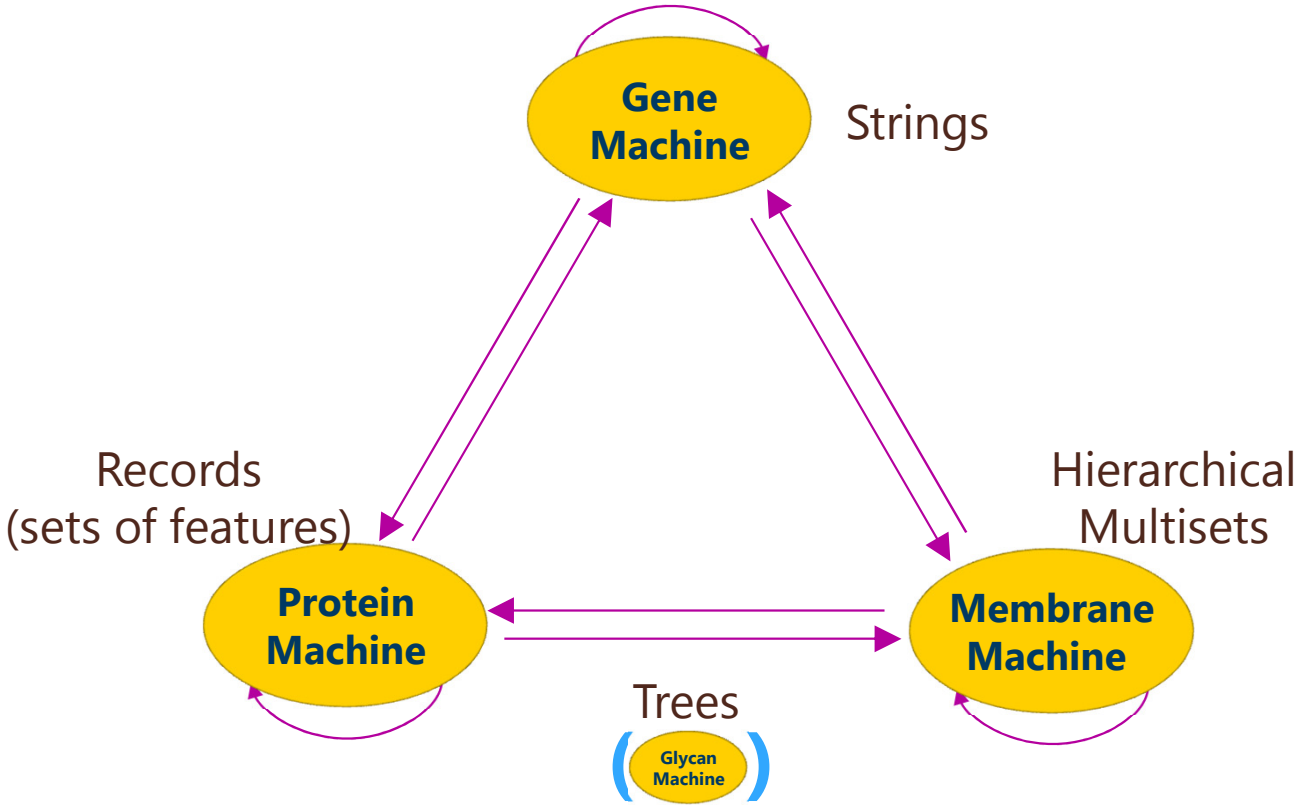
Ultrasensitivity in the mitogen-activated protein **cascade**, Chi-Ying F. Huang and James E. Ferrell, Jr., 1996, *Proc. Natl. Acad. Sci. USA*, 93, 10078-10083.

Abstract Machines of Biochemistry

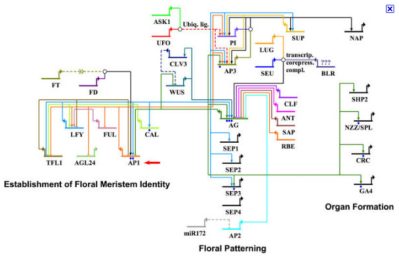




Bioinformatics View (Data Structures)



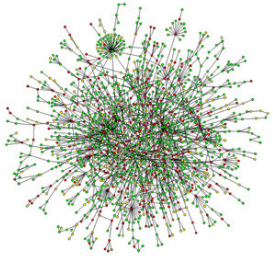
Systems Biology View (Networks)



Gene Regulatory Networks

Gene Machine

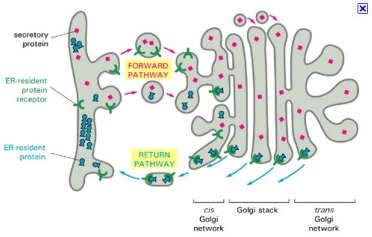
Biochemical Networks



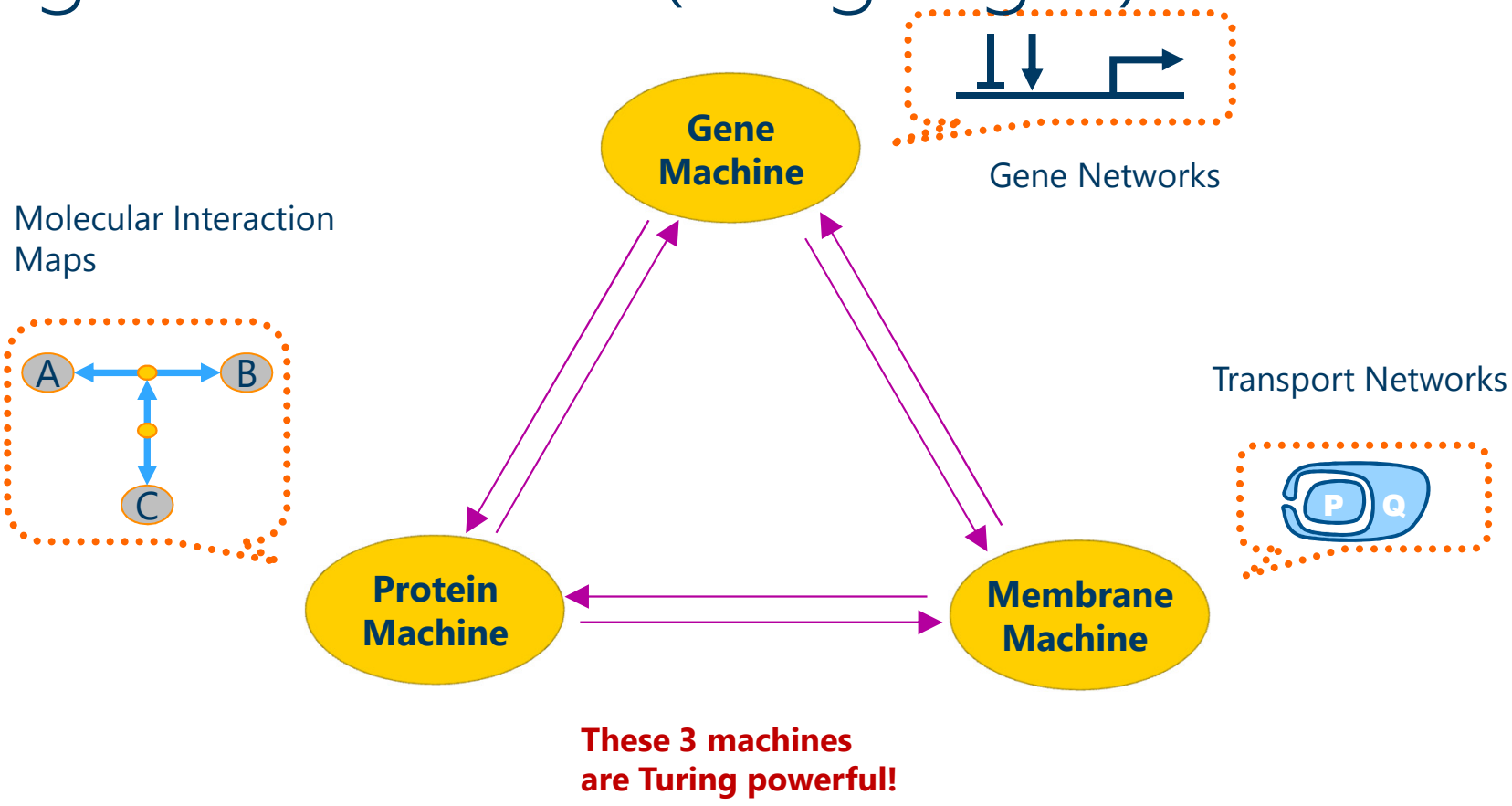
Protein Machine

Transport Networks

Membrane Machine

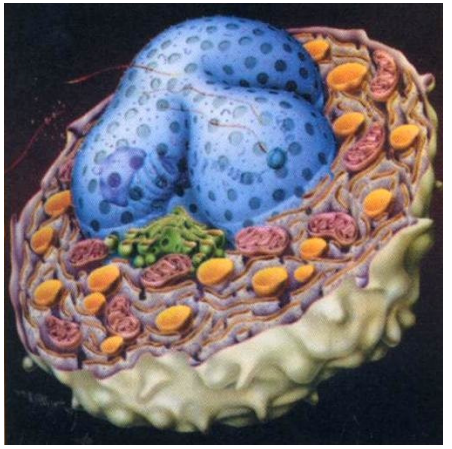


Algorithmic View (Languages)



More concretely

- Give substance to the claim that "cells compute"
  - Yes, but *what* do they compute?
- Catch nature red-handed in the act of running a computational task
  - Something that a computer scientist would recognize as an *algorithm*



Chemical Algorithms

Can *Chemistry* Compute?

- If we believe that biology can do computation…
  - It must be somehow based on chemistry
- So, can chemistry compute, and how?
  - That is in itself a very interesting question with non-trivial answers

Chemical Programming Examples

specification

Y := min(X1, X2)

Y := max(X1, X2)

program

X1 + X2 -> Y

X1 -> L1 + Y
X2 -> L2 + Y
L1 + L2 -> K
Y + K -> 0

max(X1,X2)=
(X1+X2)-min(X1,X2)

(but is not computed "sequentially": it is a form of concurrent computation)

A Consensus Problem

- Population Consensus
  - Given two populations of x and y "agents"
  - we want them to "reach consensus"
  - by converting *all* agents to x or to y
    depending on which population was in majority initially

- Population Protocols Model
  - Finite-state identity-free agents (molecules) interact in
    randomly chosen pairs ($\Rightarrow$ stochastic symmetry breaking)
  - Each interaction (collision) can result in state changes
  - Complete connectivity, no centralized control (well-mixed solution)

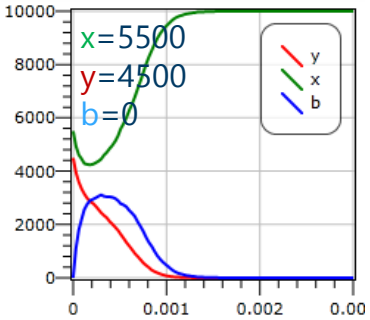
specification

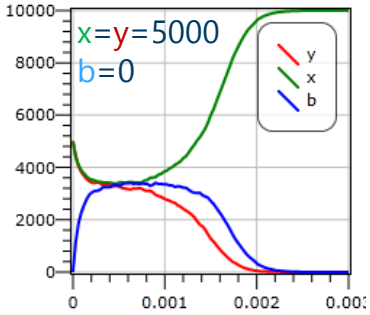
$X,Y := X+Y, 0 \quad \text{if} \quad X_0 \geq Y_0$

$X,Y := 0, X+Y \quad \text{if} \quad Y_0 \geq X_0$

A Consensus Algorithm

- Approximate Majority (AM) Algorithm
  - Uses a third "undecided" population b
  - Disagreements cause agents to become undecided
  - Undecided agents agree with any non-undecided agent

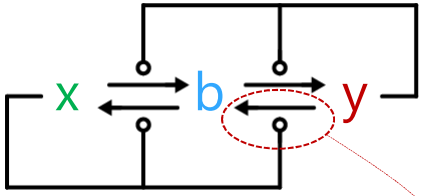




Dana Angluin · James Aspnes · David Eisenstat

A Simple Population Protocol for Fast Robust Approximate Majority

catalysis



chemical reaction network

$x + y \to^{r} y + b$

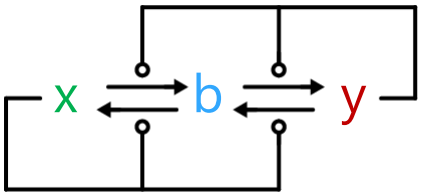
$y + x \to^{r} x + b$

$b + x \to^{r} x + x$

$b + y \to^{r} y + y$

A Biological Implementation

Approximate Majority (AM)



1) Bistable
   Even when initially x=y (stochastically)

2) Fast (asymptotically optimal)
   O(*log n*) convergence time

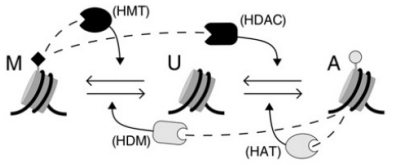
3) Robust to perturbation
   above a threshold, initial majority wins *whp*

Dana Angluin · James Aspnes · David Eisenstat

A Simple Population Protocol for Fast Robust Approximate Majority

2007

Epigenetic Switch



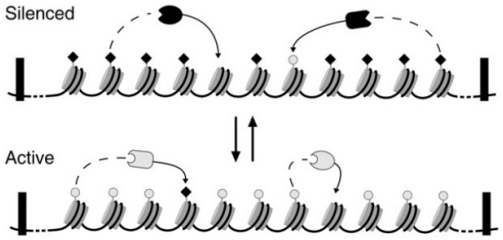
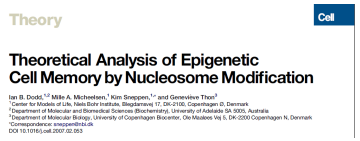


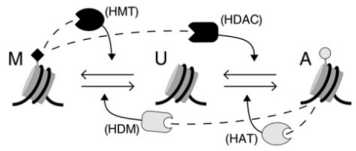
Figure 1. Basic Ingredients of the Model



2007

Here We Got Lucky

- We can claim that the epigenetic switch is a *direct* biological implementation of an algorithm
  - Although we may have to qualify that with some notion of approximation of the (enzymatic) kinetics

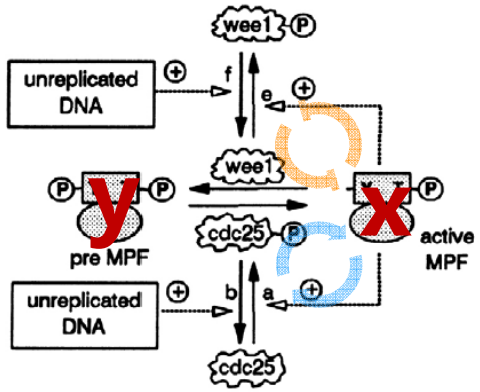


- In most cases the biological implementation seems more *indirect* or *obfuscated*
  - *"Nature is subtle but not malicious - Einstein"* Ha! think again!
  - Other implementations of Approximate Majority seem more convoluted and… approximate

The Cell Cycle Switch

Universal control mechanism regulating onset of M-phase
Paul Nurse

- This basic network is universal in Eukaryotes [P. Nurse]
  - The *switching function* and the *basic network* is *the same* from yeast to us.
  - In particular detail, in frog eggs:



Double positive feedback on x
Double negative feedback on x
No feedback on y. Why ???

Journal of Cell Science 106, 1153-1168 (1993)
Printed in Great Britain © The Company of Biologists Limited 1993

Numerical analysis of a comprehensive model of M-phase control in *Xenopus* oocyte extracts and intact embryos

Bela Novak* and John J. Tyson†
Department of Biology, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24060-0406, USA
*Permanent address: Department of Agricultural Chemical Technology, Technical University of Budapest, 1521 Budapest Gellert Ter 4, Hungary
†Author for correspondence

  - The function is very well-studied. But why this network structure?
  - That is, why this peculiar *algorithm*?

How to model "Influence"

"True" molecular interactions.

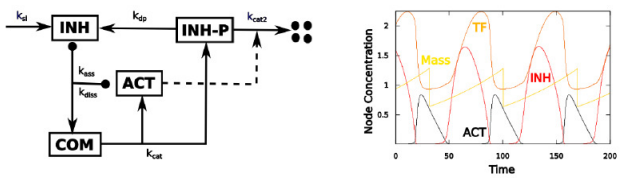


Figure 3: a) Schematic diagram of a simplified SIMM model [17]. The activa-

"Equivalent" influence interactions.

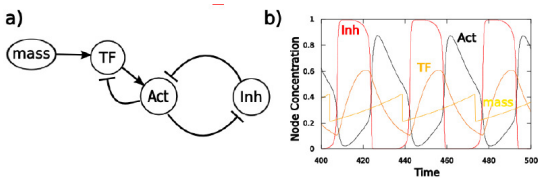


Figure 4: a) Schematic diagram of a primitive cell cycle in the reinitz framework.

Chemical Reaction Network ⟷ Influence Network

Evolving a Primitive Eukaryotic Cell Cycle Model

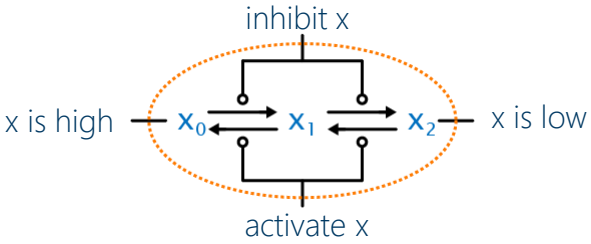
Malte Lücken, Jotun Hein, Bela Novak

Instead of modeling basic interactions, such as binding, synthesis, and degradation of molecular components, this framework models interactions simply as activation or inhibition. This approach also reduces the number of nodes necessary in the network, as e.g. the inhibitor binding tightly to the activator to form a complex, which produces phosphorylated inhibitor to be degraded under catalysis by the activator, is now simply a double negative feedback loop shown in Figure 1. This type of interaction is the basis of both aforementioned molecular model, therefore they can both be summarized in a single Reinitz model.

The Triplet Model of Influence

activation
inhibition
catalysis

=



Usually modeled by sigmoid (e.g. Hill or Reinitz) functions

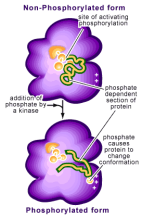
Functional Motifs in Biochemical Reaction Networks

John J. Tyson[1] and Béla Novák[2]

$$\frac{dX_i}{dt} = \gamma_i \frac{[A_i(1-X_i) - B_i X_i]}{A_i + B_i}, \quad i = 1, \ldots, N,$$

$$A_i = \exp\left\{\sigma_i\left(\alpha_{i0} + \sum_{j=1}^{N} \alpha_{ij} X_j\right)\right\}, \quad B_i = \exp\left\{\sigma_i\left(\beta_{i0} + \sum_{j=1}^{N} \beta_{ij} X_j\right)\right\}, \qquad (4)$$

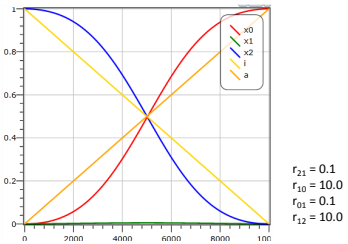
biological mechanism:
(e.g.:) multisite phosphorylation



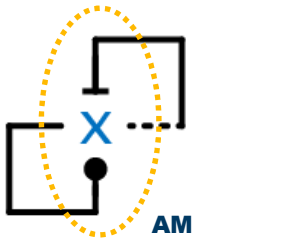
triplet motif

We model them by
4 mass action reactions over
3 species x_0, x_1, x_2

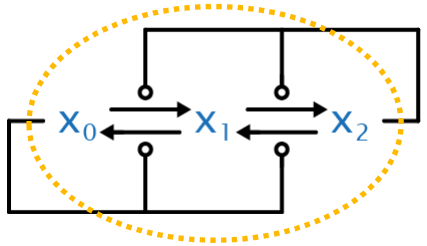
They actually implement a
Hill function of coefficient 2:



For example:



=



Approximate Majority

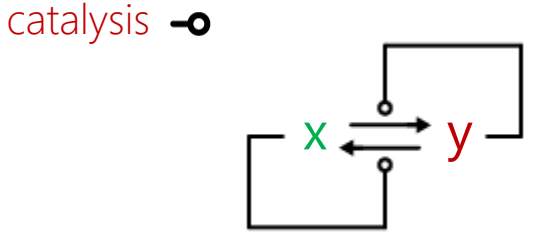
How to Build a Good Switch

- We need first a bistable system: one that has two *distinct* and *stable* states. I.e., given any initial state the system must settle into one of two states
- The settling must be fast (not get stuck in the middle for too long) and robust (must not spontaneously switch back)
- Finally, we need to be able to flip the switch by external inputs

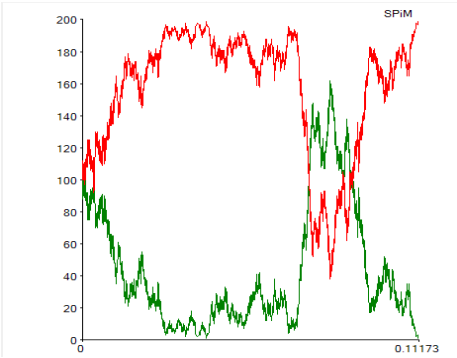
A Bad Algorithm

- Direct Competition
  - x catalyzes the transformation of y into x
  - y catalyzes the transformation of x into y
  - when all-x or all-y, it stops

- This system has two end states, but
  - Convergence to an end state is slow (a random walk)
  - Any perturbation of an end state can start a random walk to the other end state (hence not really *bistable*)



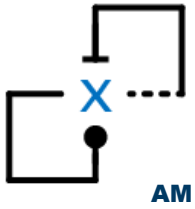
$$y + x \rightarrow x + x$$
$$x + y \rightarrow y + y$$



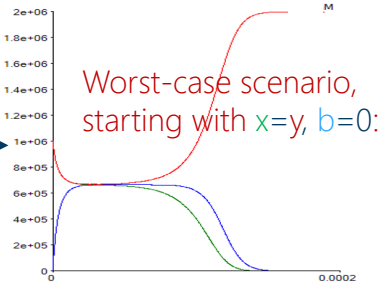
A Good Algorithm

- Approximate Majority (AM)
  - Third, undecided, state b
  - Disagreements cause agents to become undecided
  - Undecided agents believe any non-undecided agent
- With high probability, for *n* agents
  - The total number of interactions before converging is $O(n \log n)$
    ⇒ fast (optimal)
  - The final outcome is correct if the initial disparity is $\omega(\sqrt{n} \log n)$
    ⇒ solution states are robust to perturbations
- Logarithmic time bound in parallel time
  - *Parallel time* is the number of steps divided by the number of agents
  - In parallel time the algorithm converges with high probability in $O(\log n)$

activation
inhibition



x=y

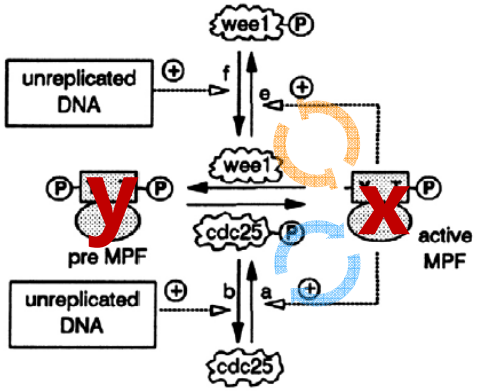


Worst-case scenario,
starting with x=y, b=0:

Dana Angluin · James Aspnes · David Eisenstat
A Simple Population Protocol for Fast Robust Approximate Majority

An "Ugly" Algorithm: Cell Cycle Switch

activation
inhibition



Nobel-prize winning network

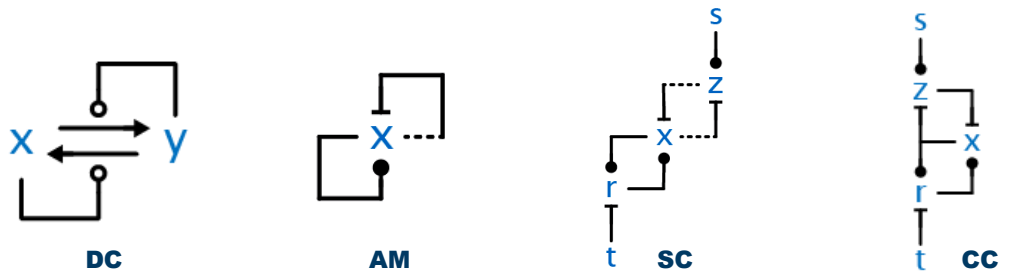
Obfuscation of a distributed algorithm?

- Is it a good algorithm? Is it bad?
- Is it optimal or suboptimal?

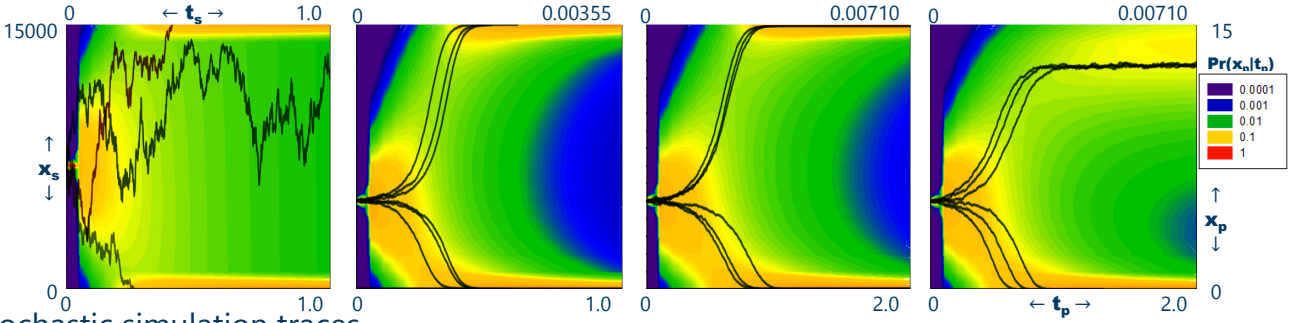
Convergence Analysis

- Switches as computational systems

CC converges in O(log n) time (like AM)
(but 2x slower than AM, and does not fully switch)



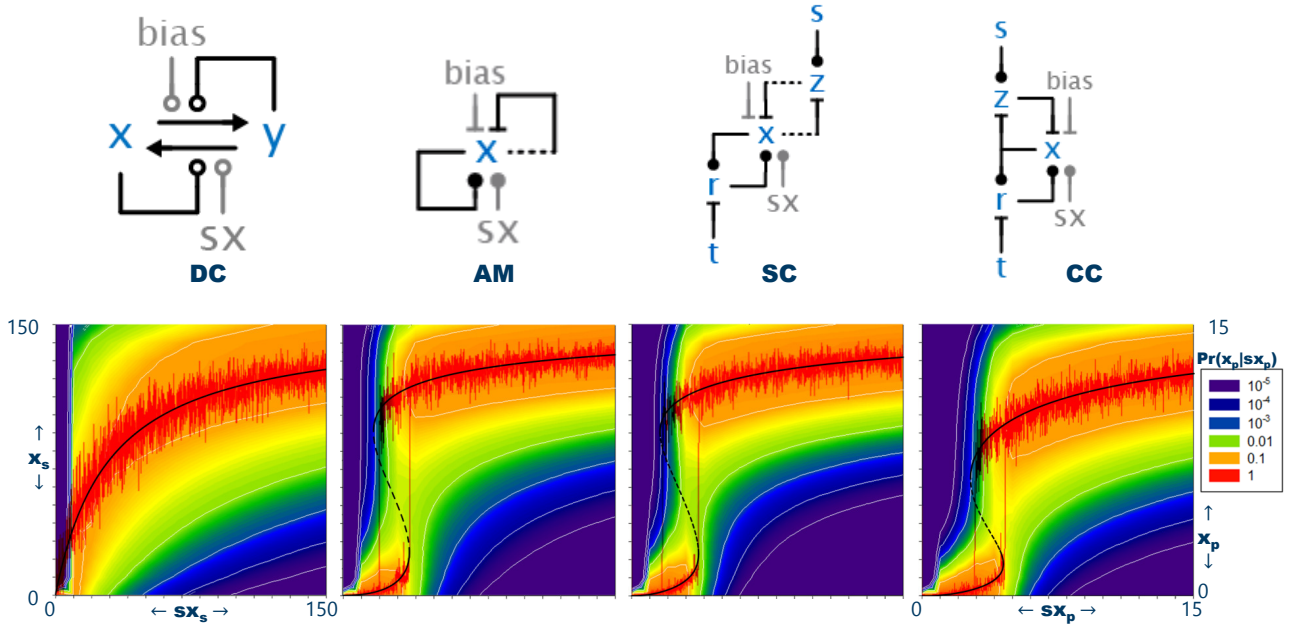
Start symmetrical
(x_0=x_1=x_2 etc.)



Black lines: several stochastic simulation traces
Color: full probability distribution of small-size system

Steady State Analysis

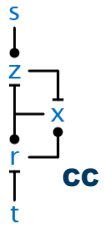
- Switches as dynamical systems

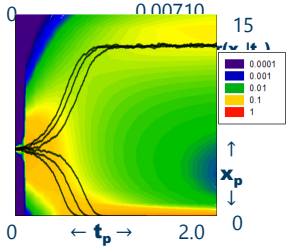


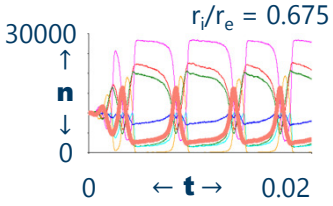
Black lines: deterministic ODE bifurcation diagrams
Red lines: noisy stochastic simulations
Color: full probability distribution of small-size system

Why is CC worse than AM?

- The classical CC has an algorithmic "bug"
  - It works ok but never as well as AM
  - Because s continuously inhibits x through z, so that x cannot fully express





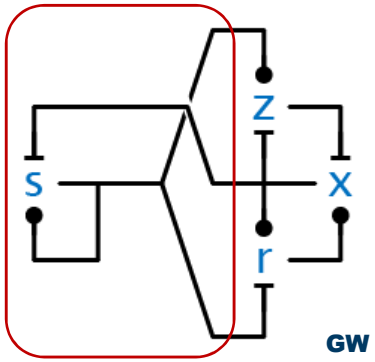


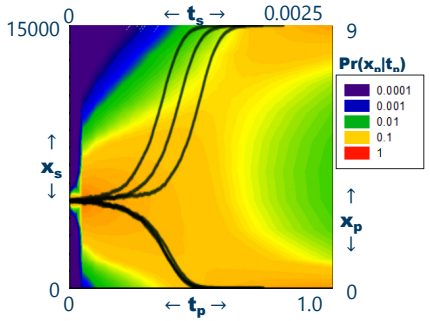
The corresponding cell cycle oscillator is also depressed

- So let's fix the bug!
  - Easy: let x inhibit s and t "in retaliation"
  - Q: Why didn't nature fix it?

Nature fixed it!

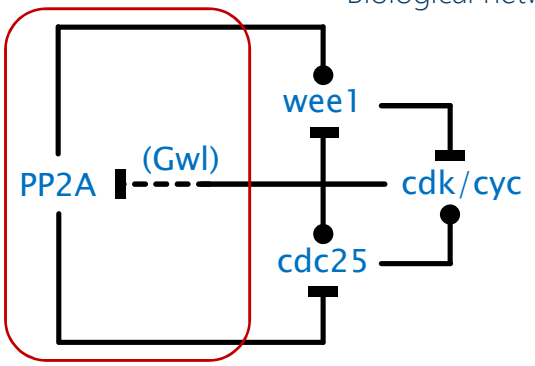
- There is another known feedback loop
  - By which x suppresses s “in retaliation” via the so-called Greatwall loop
  - Also, s and t happen to be the same molecule (=s)





Full activation!

Biological network



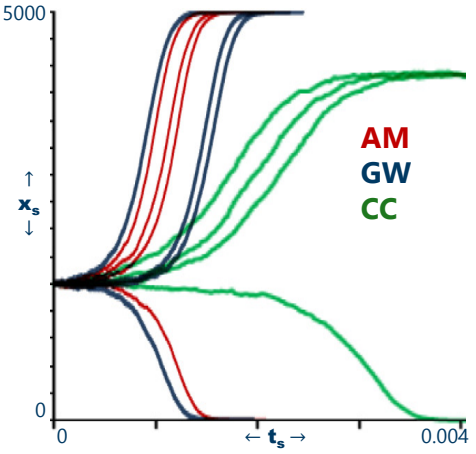
- s and x now are antagonists: they are the two halves of the switch, mutually inhibiting each other (through intermediaries).

More surprisingly

- The fix makes it faster too!
  - The extra feedback also speeds up the decision time of the switch, making it about as good as the 'optimal' AM switch:

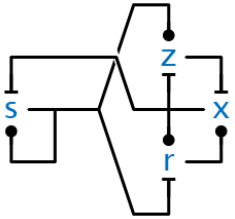
Conclusion:
Nature is trying as hard as it can to implement an AM-class algorithm!

The "classical" cell cycle switch is only half of the picture: the extra feedback completes it *algorithmically*.



Publications

- Our paper appeared:
  - Suggesting GW is a better switch than CC. *September 2012*



OPEN

SUBJECT AREAS:
COMPUTATIONAL BIOLOGY

The Cell Cycle Switch Computes Approximate Majority

Luca Cardelli[1] & Attila Csikász-Nagy[2,3]

- Another paper that same week:
  - Showing experimentally that the Greatwall loop is a necessary component of the switch, i.e. the not-as-good-as-AM network has been 'refuted'

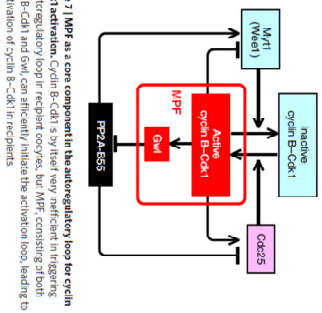


Figure 7 | MPF as a core component in the autoregulatory loop for cyclin B–Cdk1 activation. Cyclin B–Cdk1 's synthesis is inefficient in triggering the autoregulatory loop in recipient oocytes, but MPF, consisting of both cyclin B–Cdk1 and Gwl, can efficiently initiate the activation loop, leading to full activation of cyclin B–Cdk1 in recipients.

ARTICLE

Received 6 Jul 2012 | Accepted 14 Aug 2012 | Published 11 Sep 2012 DOI: 10.1038/ncomms2062

Greatwall kinase and cyclin B-Cdk1 are both critical constituents of M-phase-promoting factor

Masatoshi Hara[1,†], Yusuke Abe[1,†], Toshiaki Tanaka[2], Takayoshi Yamamoto[1,†], Eiichi Okumura[1] & Takeo Kishimoto[1]

But again, is CC (or GW) the "same" as AM?

- Our evidence for computational content of biochemical networks is so far
  - Quantitative, covering both kinetic and steady state behavior of *what* networks do
  - But empirical (based on simulations/numerical solutions)
  - And it does not yet explain *how* the CC/GW network relates to the AM network, that is, how each *piece* of CC/GW corresponds to each *piece* of AM
- Analytical evidence is harder to obtain
  - The proofs of the computational properties (optimality etc.) for the AM algorithm are hard and do not generalize easily to more complex networks
  - Quantitative theories of behavioral equivalence and behavioral approximation, e.g. in process algebra, are still lacking (although rich qualitative theories exist)

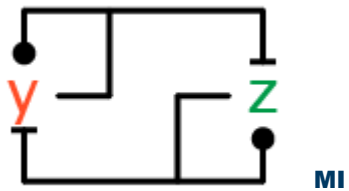
Network Morphisms

When does a (complex) network implement a (simpler) algorithm?

Antagonistic Networks

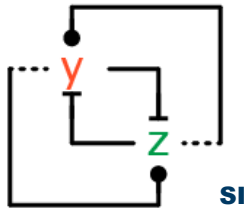
activation
inhibition

1 vs. 1
Mutual Inhibition & Self Activation



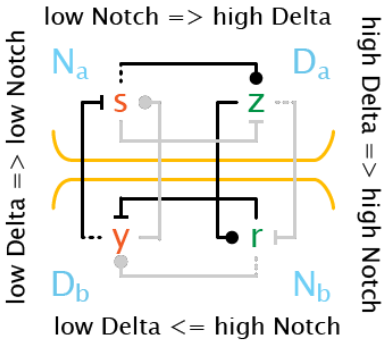
MI

1 vs. 1
Mutual Inhibition & Mutual Anti-activation

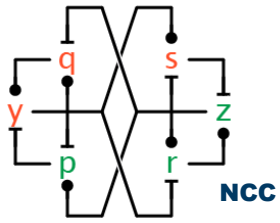


SI

2 vs. 2

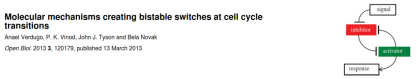


3 vs. 3



NCC

Cell cycle transitions

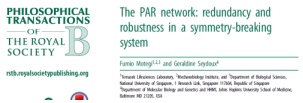


Molecular mechanisms creating bistable switches at cell cycle transitions

Anael Verdugo, P. K. Vinod, John J. Tyson and Béla Novak

Open Biol. 2013 **3**, 120179, published 13 March 2013

Polarity establishment

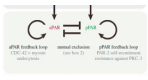


PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B

rstb.royalsocietypublishing.org

The PAR network: redundancy and robustness in a symmetry-breaking system

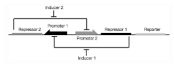
Fumio Motegi[1,2] and Geraldine Seydoux[3]



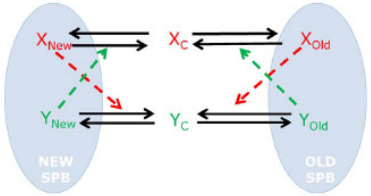
Gene networks

Construction of a genetic toggle switch in *Escherichia coli*

Timothy S. Gardner[1,2], Charles R. Cantor[1] & James J. Collins[1,2]



Septation Initiation

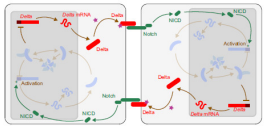


Dynamics of SIN Asymmetry Establishment

Archana Bajpai[1], Anna Fookstsova[2], Jun-Song Chen[2], Dannel McCollum[3], Masamitsu Sato[4,5], Rafael E. Carazo-Salas[6], Kathleen L. Gould[2], Attila Csikasz-Nagy[1,7]*

PLOS Computational Biology | www.ploscompbiol.org

Delta-Notch



Patterning embryos with oscillations: structure, function and dynamics of the vertebrate segmentation clock

Andrew C. Oates[1,2], Luis G. Morelli[1,2] and Saúl Ares[3,4,*]

Lateral Inhibition through Delta-Notch Signaling: A Piecewise Affine Hybrid Model*

Ronojoy Ghosh and Claire J. Tomlin

M.D. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.): HSCC 2001, LNCS 2034, pp. 232–246, 2001.
© Springer-Verlag Berlin Heidelberg 2001

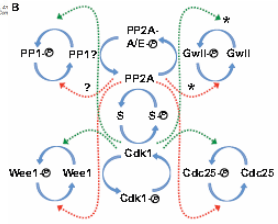
The "new" cell cycle switch

Phosphorylation network dynamics in the control of cell cycle transitions

Daniel Fisher[1,*], Liliana Krasinska[1,2], Damien Coudreuse[3,4] and Béla Novák[1,2]

[1]Institut de Génétique Moléculaire de Montpellier, IGMM, CNRS UMR 5535, Université Montpellier I and II, 34293 Montpellier, France
[2]Institute of Genetics and Development of Rennes, CNRS UMR 6290, 35043 Rennes, France
[3]Oxford Centre for Integrative Systems Biology, Department of Biochemistry, University of Oxford, South Parks Road, Oxford OX1 3QU, UK
[*]Author for correspondence (daniel.fisher@igmm.cnrs.fr)
[†]These authors contributed equally to this work

Journal of Cell Science 122, 69–73
© 2012. Published by The Company of Biologists Ltd
doi: 10.1242/jcs.100271



New Cell Cycle Switch Network

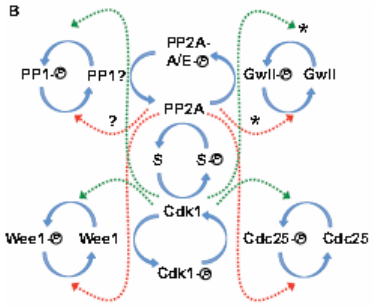
- A recent paper presents a more complete view of the cell cycle switch
- N.B. "phosphorylation network dynamics" here is the same as our x_0-x_1-x_2 motif

Phosphorylation network dynamics in the control of cell cycle transitions

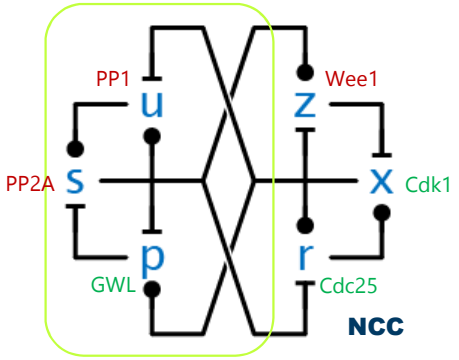
Daniel Fisher[1,*], Liliana Krasinska[1,‡], Damien Coudreuse[2,‡] and Béla Novák[3,‡]
[1]Institut de Génétique Moléculaire de Montpellier, IGMM, CNRS UMR 5535, Université Montpellier I and II, 34293 Montpellier, France
[2]Institute of Genetics and Development of Rennes, CNRS UMR 6290, 35043 Rennes, France
[3]Oxford Centre for Integrative Systems Biology, Department of Biochemistry, University of Oxford, South Parks Road, Oxford OX1 3QU, UK
*Author for correspondence (daniel.fisher@igmm.cnrs.fr)
‡These authors contributed equally to this work

Journal of Cell Science 125, 4703–4711

doi: 10.1242/jcs.106351

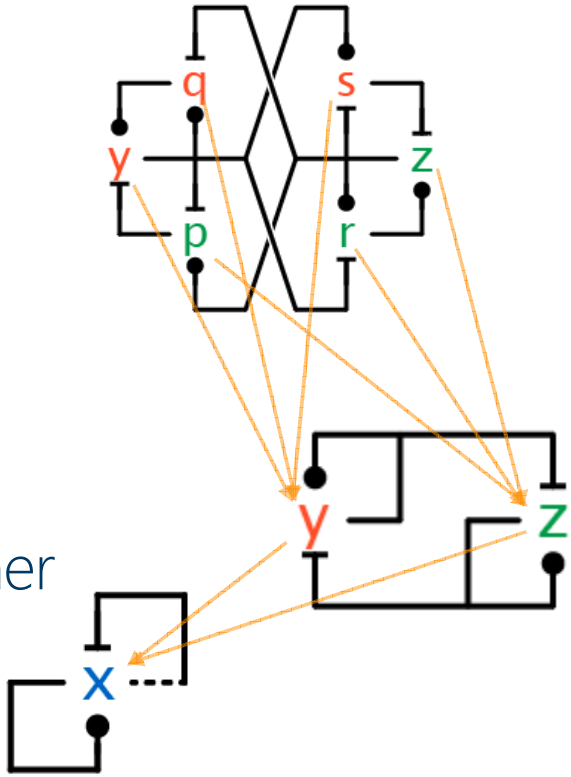


Mutual inhibition between *three* species each



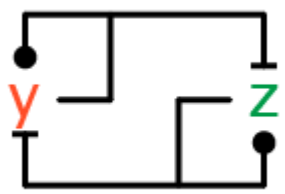
Comparing networks

- How can we compare different networks?
  - Different number of species
  - Different number of reactions
  - Apparently unrelated connectivity
- So that we can compare their function?
  - Does antagonism (in network structure) guarantee bistability (in function)?
- We do it by *mapping* networks onto one another so that they *emulate* each other



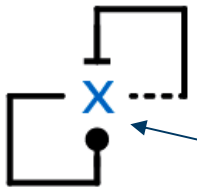
Network Emulation: MI emulates AM

- For *any* rates and initial conditions of **AM**, we can find *some* rates and initial conditions of **MI** such that the (6) trajectories of **MI** retrace those (3) of **AM**:



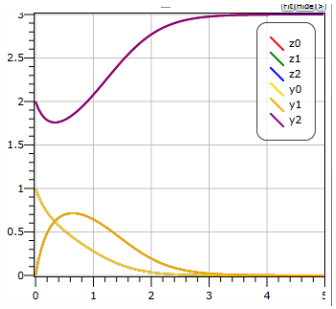
MI

~y,z⇢ x

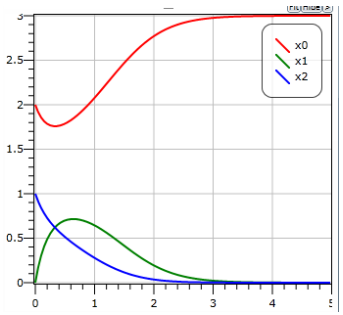


AM

(3 species)



(6 species on 3 trajectories)



(3 species on 3 trajectories)

initialize:
$z = x$
$\sim y = x$

($y_2 = x_0$
$y_1 = x_1$
$y_0 = x_0$)

- How do we find these matching parameters? By a network morphism!

CRN Morphisms

A *CRN morphism* from (S, R) to $(\hat{S}, \hat{R})$
written $m \in (S, R) \to (\hat{S}, \hat{R})$

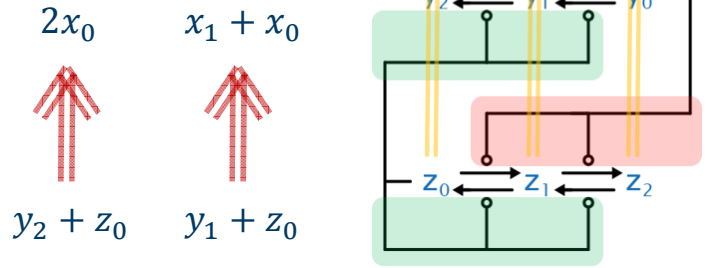
is a pair of maps $m = (m_S, m_R)$

a *species map* $m_S \in S \to \hat{S}$

a *reaction map* $m_R \in R \to \hat{R}$

extended to a *complex map* $m_S \in \mathbb{N}^S \to \mathbb{N}^{\hat{S}}$
linearly: $m_S(\rho)_{\hat{s}} = \Sigma_{s \in m_S^{-1}(\hat{s})} \rho_s$

Mappings (symmetries) between two networks



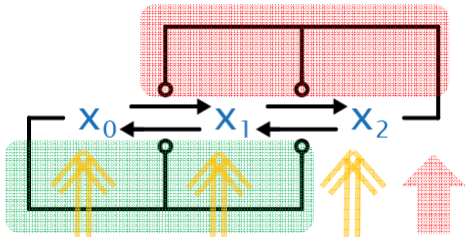
How to check emulations

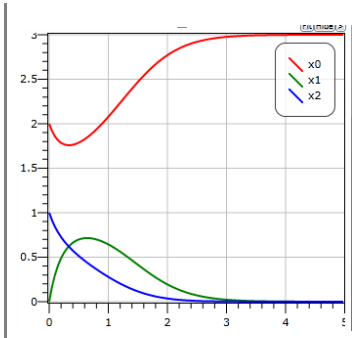
- How do we check a potential emulation morphism for all possible initial conditions of the target?
  - Statically: check conditions on the joint stoichiometric matrices of the two networks under the mapping
- How do we check a potential emulation morphism for all possible rates of the target?
  - Can't; but if one emulation is found, then the rates of the target network can be changed *arbitrarily* and a related emulation will again exist

Network Emulation: MI emulates AM

A mapping of species and reactions

AM



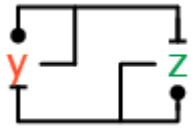


any initial conditions

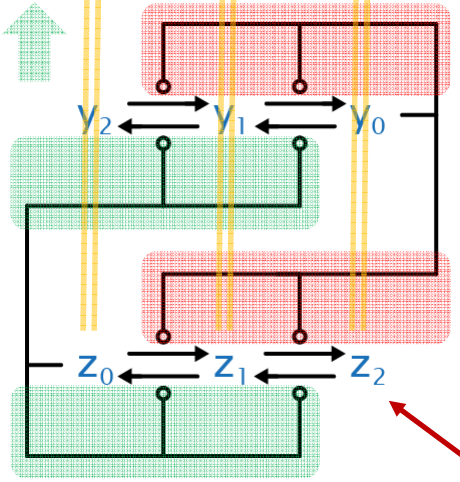
homomorphic mapping

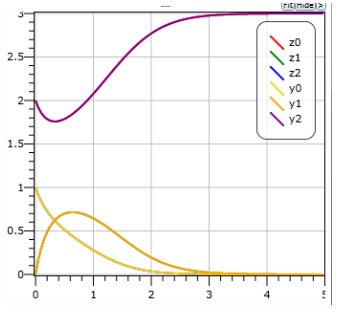
z -> x

~y -> x



MI





initial conditions:

$z_0 = y_2 = x_0$

$z_1 = y_1 = x_1$

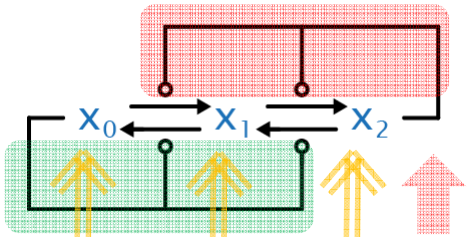
$z_2 = y_0 = x_2$

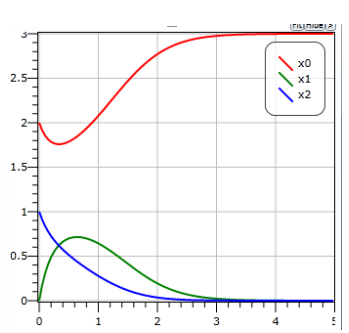
less trivial than you might think:
it need not preserve the out-degree of a node!

Network Emulation: SI emulates AM

A mapping of species and reactions

AM

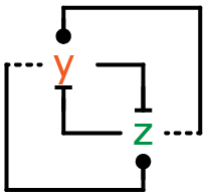




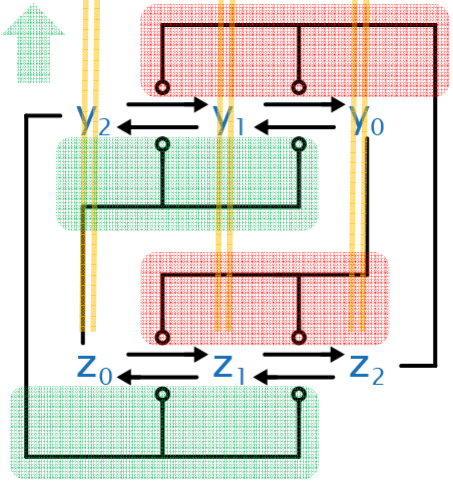
any initial conditions

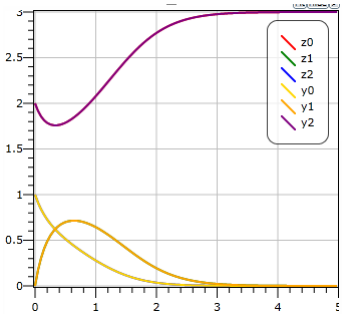
homomorphic mapping

z -> x
~y -> x



SI



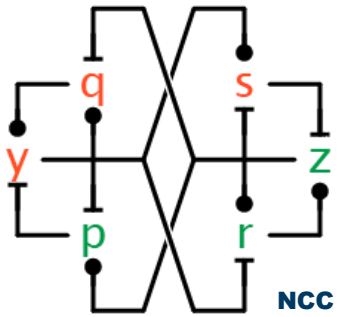


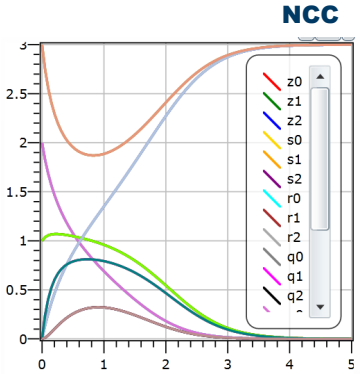
initial conditions:

$z_0 = y_2 = x_0$
$z_1 = y_1 = x_1$
$z_2 = y_0 = x_2$

Network Emulation: NCC emulates MI

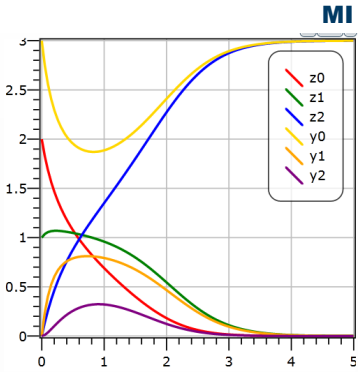
- For *any* rates and initial conditions of MI we can find *some* rates and initial conditions of NCC such that the (18) trajectories of NCC retrace those (6) of MI





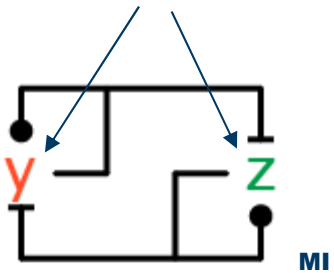
(18 species on 6 trajectories)

z,r,p ⇢ z
y,q,s ⇢ y



(6 species on 6 trajectories)

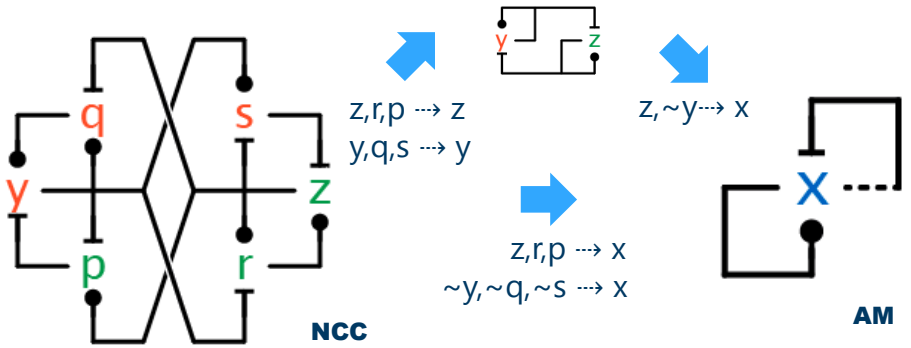
(3 species each)

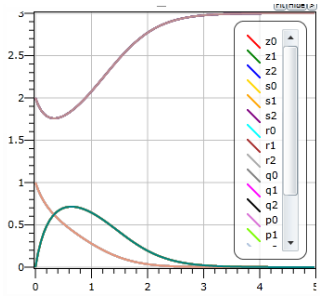


initialize
z,r,p = z
y,q,s = y

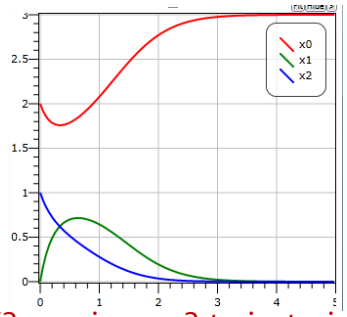
Emulations Compose

- The (18) trajectories **NCC** can *always* retrace those (3) of **AM**





(18 species on 3 trajectories)



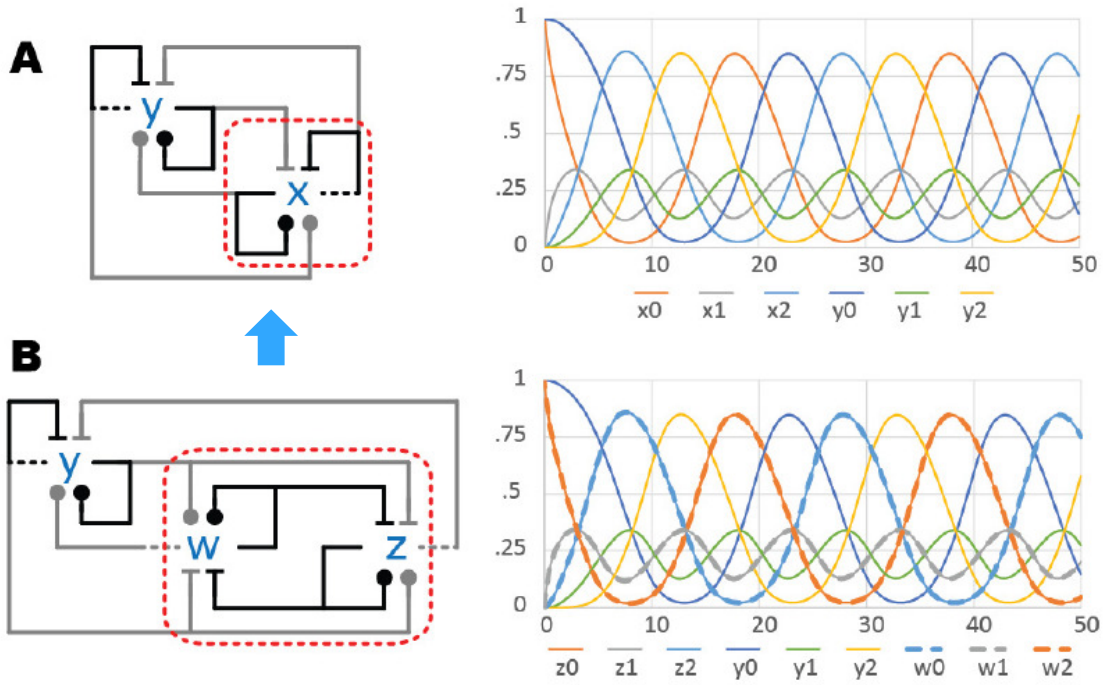
(3 species on 3 trajectories)

The new cell cycle switch can emulate AM *exactly*.

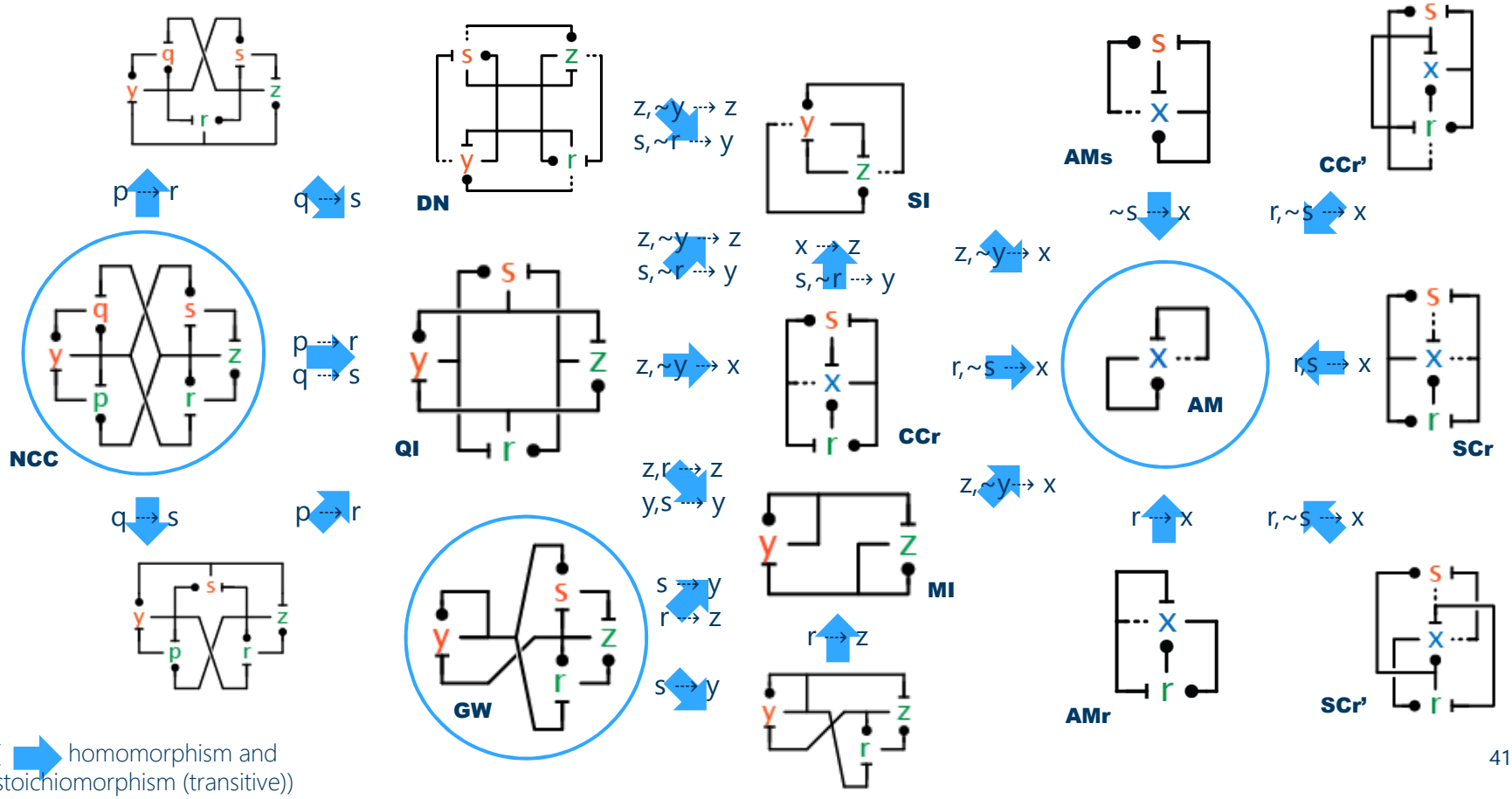
For *any* initial conditions of AM.

And for *any* rates of AM.

Emulations are Modular

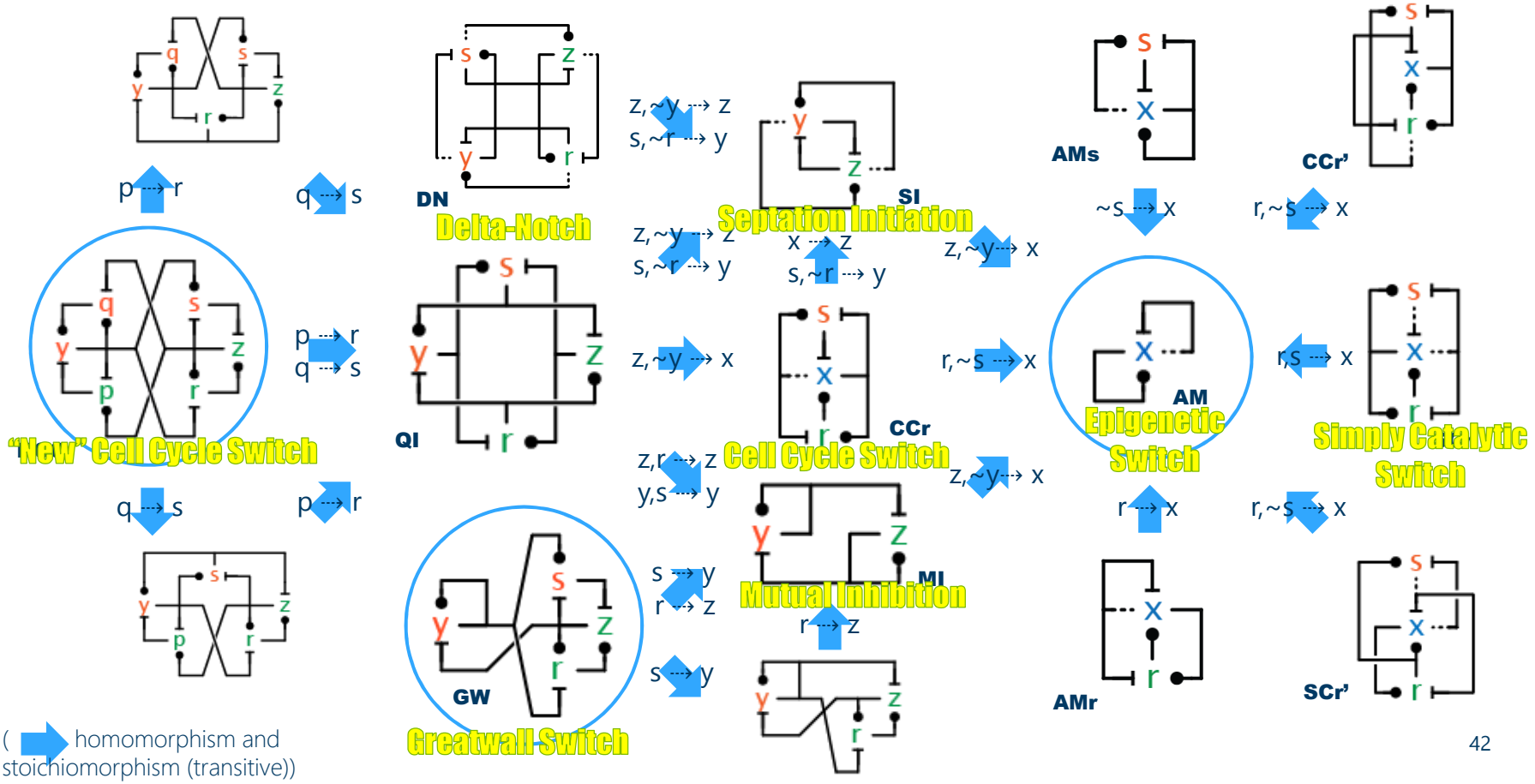


Approximate Majority Emulation Zoo



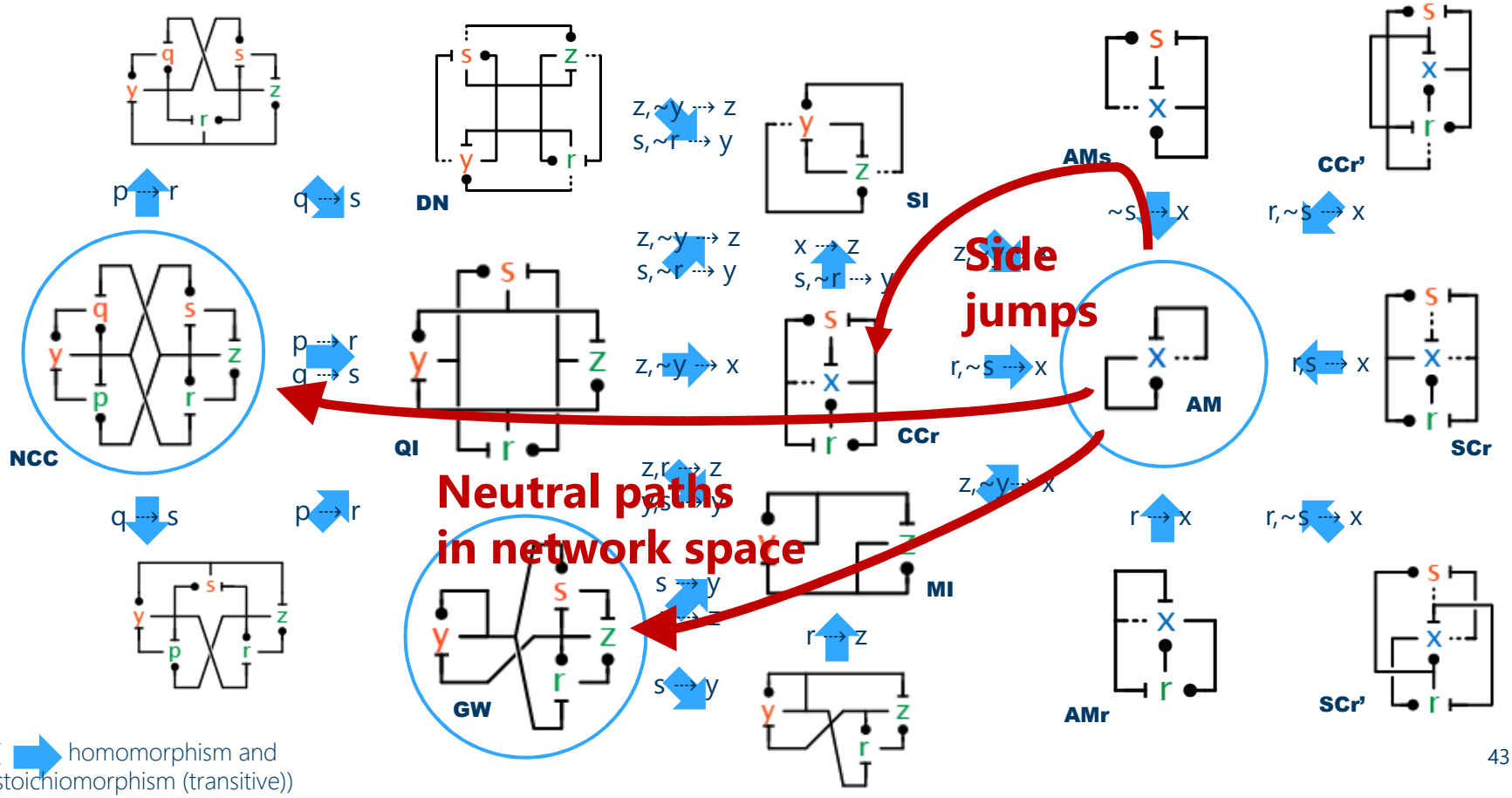
(homomorphism and stoichiomorphism (transitive))

Approximate Majority Emulation Zoo



(homomorphism and stoichiomorphism (transitive))

Approximate Majority Emulation Zoo



(⇢ homomorphism and stoichiomorphism (transitive))

Nature likes a good algorithm

First part

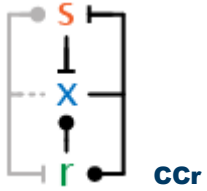
Approximate

"default" rates and initial conditions

Second part

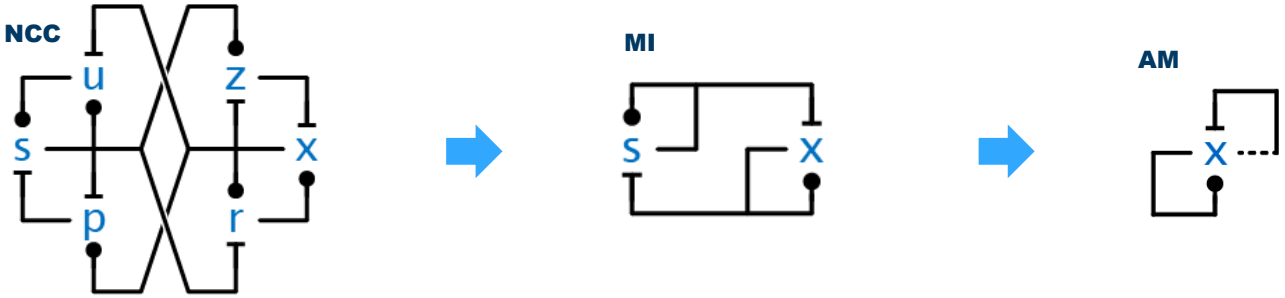
Exact

any rates and initial conditions



These additional feedbacks *do exist* in real cell cycles (via indirections)

The cell cycle switch *can* exactly emulate AM



Other work

Characterization of Emulation

Emulation Theorem: If $m \in (S, R) \to (\hat{S}, \hat{R})$ is a CRN reactant morphism and stoichiomorphism then it is a CRN emulation

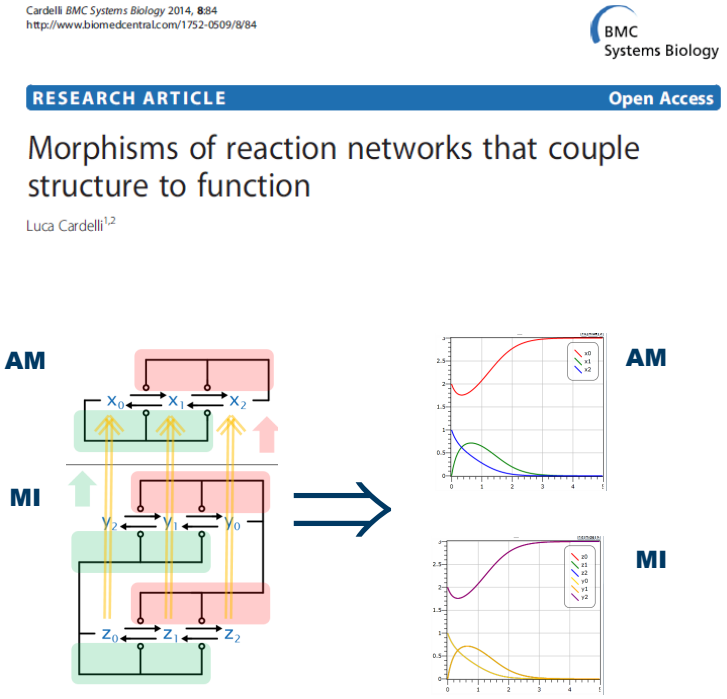
reactant morphism $\boldsymbol{m_S}^{\mathbf{T}} \cdot \boldsymbol{\rho} = \hat{\boldsymbol{\rho}} \cdot \boldsymbol{m_R}^{\mathbf{T}}$ preserve enough network structure

stoichiomorphism $\boldsymbol{\varphi} \cdot \boldsymbol{m_R} = \boldsymbol{m_S} \cdot \hat{\boldsymbol{\varphi}}$ preserve enough chemical stoichiometry

$\Downarrow$

emulation $\forall \hat{\boldsymbol{v}}.\ F(\hat{\boldsymbol{v}} \circ m_S) = \hat{F}(\hat{\boldsymbol{v}}) \circ m_S$ preserve derivatives

F is the *differential system* of (S, R), given by the law of mass action, $\hat{\boldsymbol{v}}$ is a state of $(\hat{S}, \hat{R})$. $\boldsymbol{\varphi}$ is the stoichiometric matrix and $\boldsymbol{\rho}$ is the related reactant matrix. $\boldsymbol{m_S}$ and $\boldsymbol{m_R}$ are the characteristic 0-1 matrices of the morphism maps m_S (on species) and m_R (on reactions). $-^{\mathbf{T}}$ is transpose.
Homomorphism implies reactant morphism.



Model Reduction

- Efficient algorithms to:
  - Discover emulation morphisms
  - Find reduced networks
  - Compute quotient CRNs

Forward and Backward Bisimulations for Chemical Reaction Networks

Luca Cardelli[1], Mirco Tribastone[2], Max Tschaikowski[3], and Andrea Vandin[4]

1 Microsoft Research & University of Oxford, UK
luca@microsoft.com
2-4 University of Southampton, UK
{m.tribastone,m.tschaikowski,a.vandin}@soton.ac.uk

Aggregation reduction

Emulation reduction

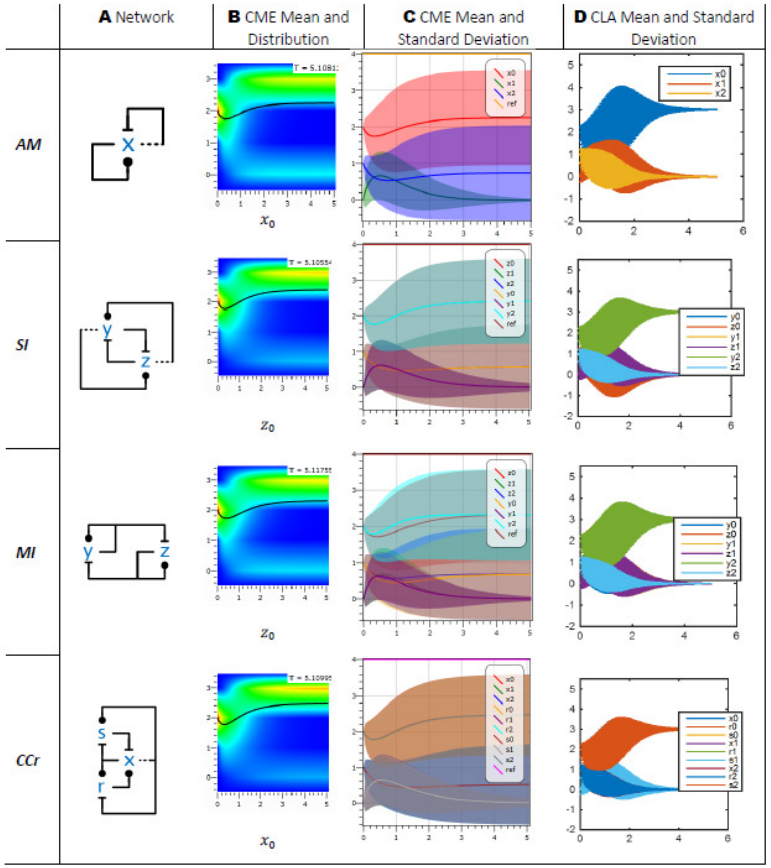
Original model			*Forward reduction*				*Backward reduction*			
Id	$\|R\|$	$\|S\|$	*Red.(s)*	$\|R\|$	$\|S\|$	*Speed-up*	*Red.(s)*	$\|R\|$	$\|S\|$	*Speed-up*
M1	3538944	262146	4.61E+4	990	222	—	7.65E+4	2708	222	—
M2	786432	65538	1.92E+3	720	167	—	3.68E+3	1950	167	—
M3	172032	16386	8.15E+1	504	122	1.16E+3	1.77E+2	1348	122	5.34E+2
M4	48	18	1.00E–3	24	12	1.00E+0	2.00E–3	45	12	1.00E+0
M5	194054	14531	3.72E+1	142165	10855	1.03E+0	1.32E+3	93033	6634	1.03E+0
M6	187468	10734	3.07E+1	57508	3744	1.92E+1	2.71E+2	144473	5575	3.53E+0
M7	32776	2506	1.26E+0	16481	1281	6.23E+0	1.66E+1	32776	2506	x
M8	41233	2562	1.12E+0	33075	1897	1.12E+0	1.89E+1	41233	2562	x
M9	5033	471	1.91E–1	4068	345	1.04E+0	4.35E–1	5033	471	x
M10	5797	796	1.61E–1	4210	503	1.47E+0	7.37E–1	5797	796	x
M11	5832	730	3.89E–1	1296	217	1.32E+1	6.00E–1	2434	217	7.55E+0
M12	487	85	2.00E–3	264	56	1.88E+0	6.00E–3	426	56	1.31E+0
M13	24	18	1.20E–2	24	18	x	7.00E–3	6	3	1.00E+0

From the BioNetGen database

Stochastic Switches

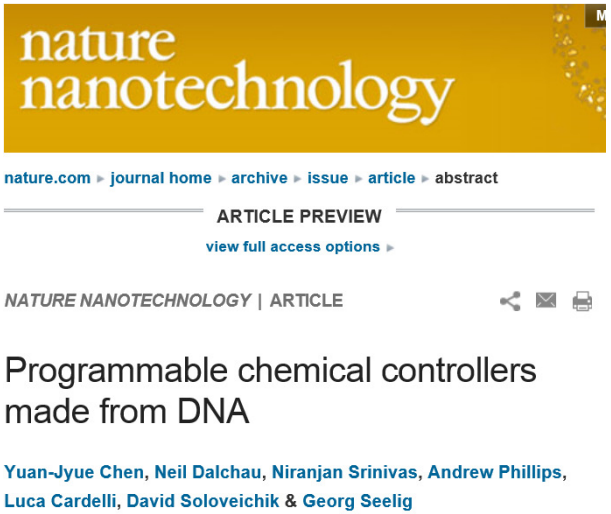
- Disentangle the contribution of complexity to stochasticity
  - Compare network noise on the baseline of deterministic emulation, across networks of different size and structure

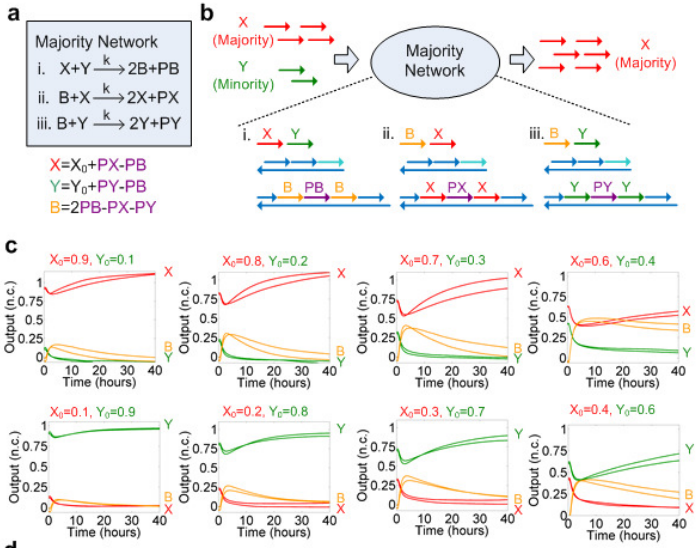
Luca Cardelli, Attila Csikász-Nagy, Neil Dalchau,
Mirco Tribastone, Max Tschaikowski



Synthetic Implementation of AM

- We produced a chemical implementation of AM using DNA gates
- I.e., a 'synthetic reimplementation' of the central cell-cycle switch.





Conclusions

Networks are Algorithms

- They are *methods* for achieving a function
  - We need to understand how these methods relate to each other
  - In addition to how and how well they implement function
  - Algorithms can be obfuscated, and nature can obfuscate networks (to what end?)
- Network emulation can be checked *statically*
  - By stoichiometric/reaction-rate (*structural*) properties
  - That is, no need to compare ODE (*functional*) properties
  - For *any* initial conditions and rates of (one of) the networks
- Efficient algorithms can find emulations
  - Automatic model reduction of large networks