

Bioinformatics 3

# V9 – Dynamic Regulation: Petri & Boolean Networks

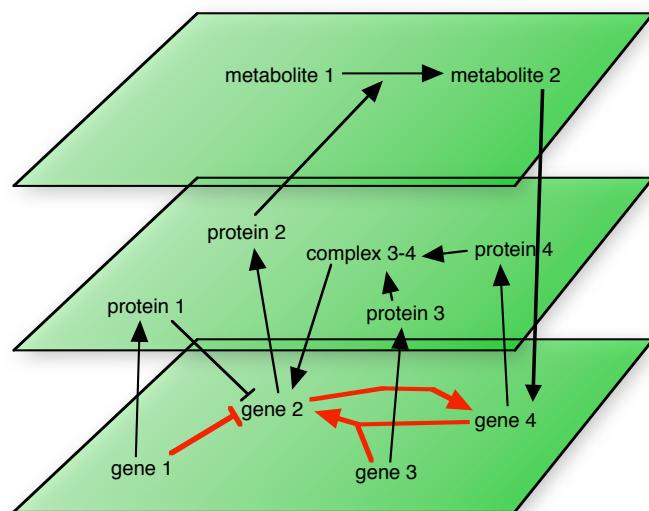
Fri, Nov 18, 2011

# Gene Regulation Networks

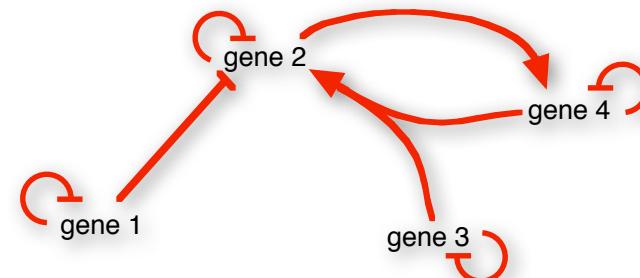
Biological regulation  
via proteins and metabolites

$\Leftrightarrow$

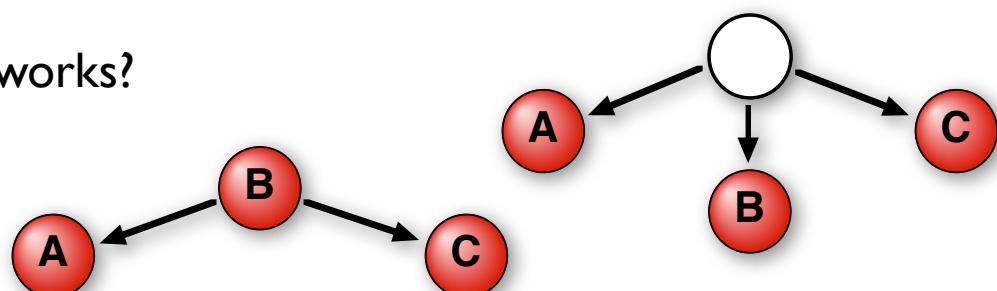
Projected regulatory network



$\Leftrightarrow$

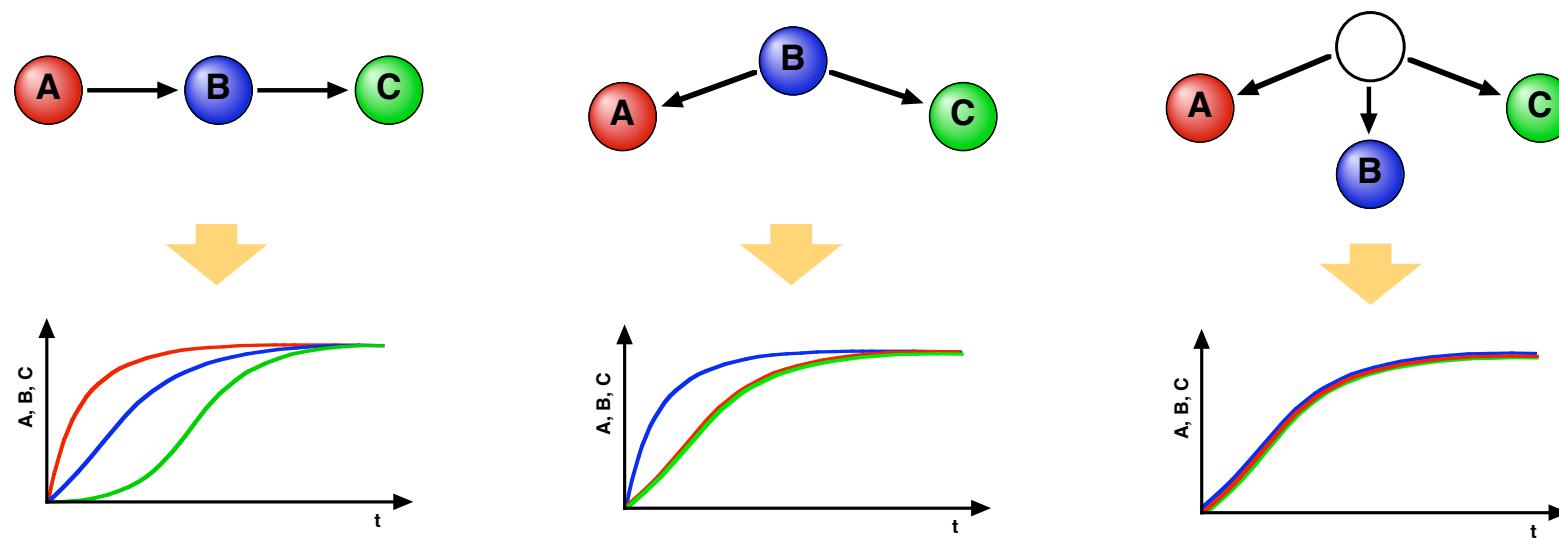


Reconstruction of static networks?



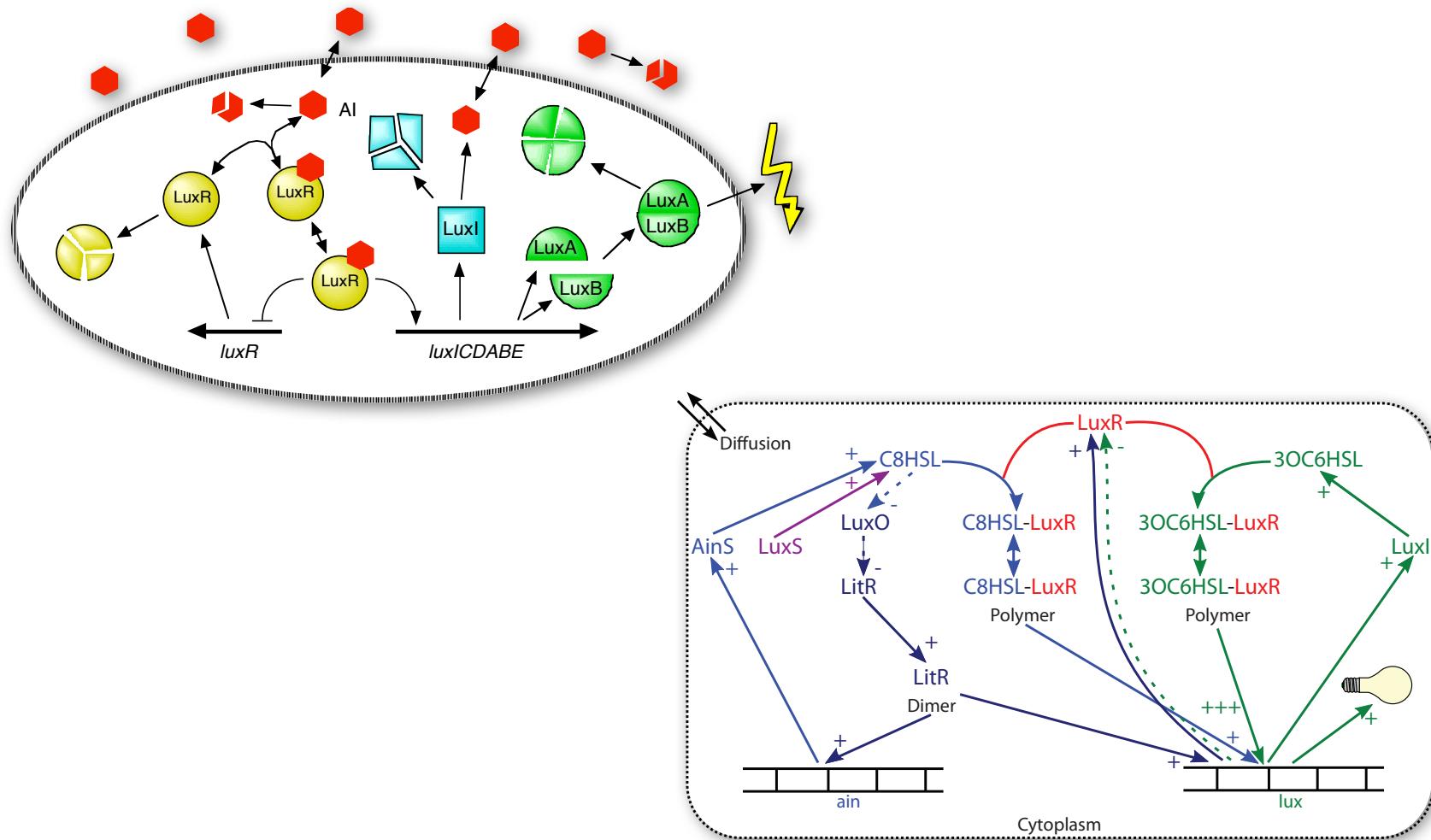
# Dynamic Reconstruction

Different network topologies => different time series



Model large networks efficiently => simplified descriptions  
(processes + numerics)

# QS of *V. fischeri*



# Boolean Networks

"Blackboard explanations" often formulated as **conditional transitions**

- "If LuxI is present, then AI will be produced..."
- "If there is AI and there's no LuxR:AI bound to the genome, then LuxR will be expressed and complexes can form..."
- "If LuxR:AI is bound to the genome, then LuxI is expressed..."

**Simplified** mathematical **description** of the dependencies:

Densities of the species	$\Leftrightarrow$	discrete states: on/off, 1/0
Network of dependencies	$\Leftrightarrow$	condition tables
Progress in time	$\Leftrightarrow$	discrete propagation steps

# Boolean Networks II

**State** of the system: described by **vector** of **discrete** values

$$S_i = \{0, 1, 1, 0, 0, 1, \dots\}$$

$$S_i = \{x_1(i), x_2(i), x_3(i), \dots\}$$

fixed number of species with **finite number** of states each

=> finite number of system states

=> periodic trajectories

=> **periodic** sequence of states = **attractor**

=> all states leading to an attractor = **basin of attraction**

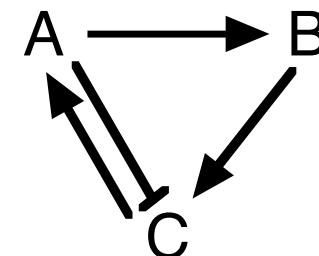
## Propagation:

$$S_{i+1} = \{x_1(i+1), x_2(i+1), x_3(i+1), \dots\}$$

$$x_1(i+1) = f_1(x_1(i), x_2(i), x_3(i), \dots) \quad \text{with } f_i \text{ given by condition tables}$$

# A Small Example

**State vector**  $S = \{A, B, C\}$   $\Rightarrow 8$  possible states



**Conditional evolution:**

A is on if C is on

$A_{i+1}$	$C_i$
0	0
1	1

A activates B

$B_{i+1}$	$A_i$
0	0
1	1

C is on if (B is on && A is off)

$C_{i+1}$	$A_i$	$B_i$
0	0	0
1	0	1
0	1	0
0	1	1

Start from  $\{A, B, C\} = \{1, 0, 0\}$

#	$S_i$	A	B	C
0	$S_0$	1	0	0
1	$S_1$	0	1	0
2	$S_2$	0	0	1
3	$S_3 = S_0$	1	0	0



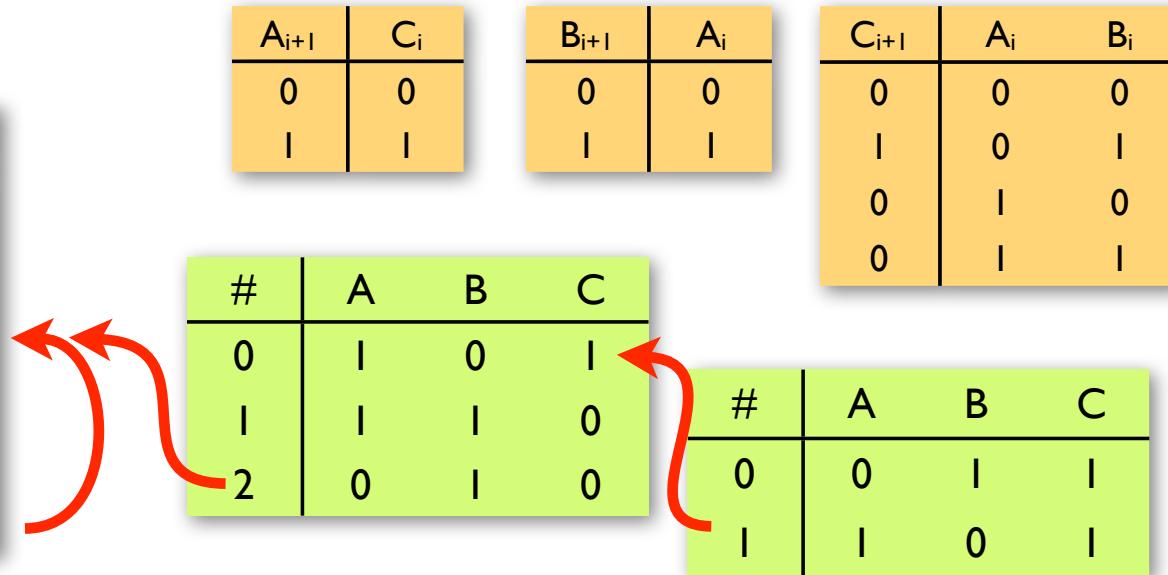
assume that inhibition through A is stronger than activation via B

periodic orbit of length 3

# Test the Other States

Test the other states

#	A	B	C
0			
1			0
2	0		0
3	0	0	
4		0	0
5	0		0

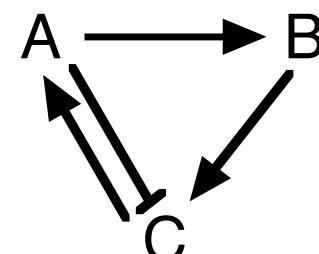


Same attractor as before:

$100 \Rightarrow 010 \Rightarrow 001 \Rightarrow 100$

also reached from:

$110, 111, 101, 011$

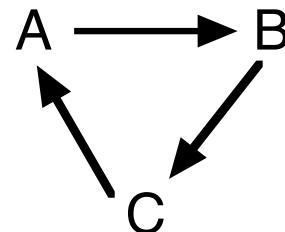


#	A	B	C
0	0	0	0
1	0	0	0



**=> Either all off or stable oszillations**

# A Knock-out Mutant



$A_{i+1}$	$C_i$
0	0
I	I

$B_{i+1}$	$A_i$
0	0
I	I

$C_{i+1}$	$B_i$
0	0
I	I

## Attractors:

#	A	B	C
0	I	0	0
1	0	I	0
2	0	0	I
3	I	0	0

#	A	B	C
0	I	I	0
1	0	I	I
2	I	0	I
3	I	I	0

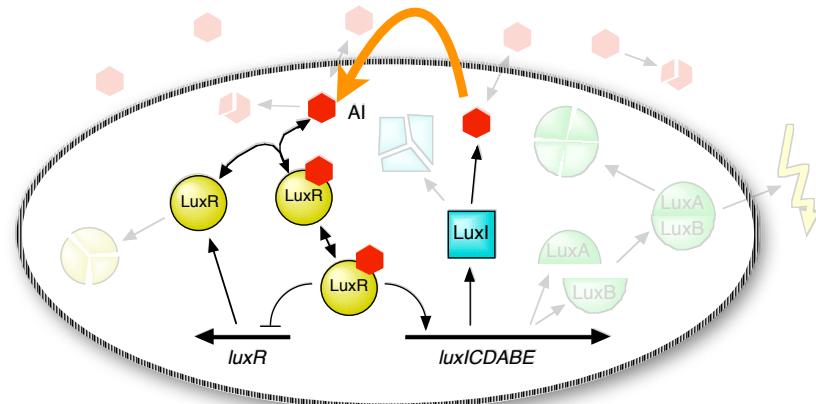
#	A	B	C
0	I	I	I
1	I	I	I

#	A	B	C
0	0	0	0
1	0	0	0

no feedback

=> no stabilization, network just "rotates"

# Boolean Network of QS



**Minimum set** of species:

LuxR, AI, LuxR:AI, LuxR:AI:genome, LuxI

Here: Light signal ( $\text{LuxAB}$ )  $\propto$  LuxI

**Condition tables:** describe the state of a species in the next step given the current states of all relevant species.

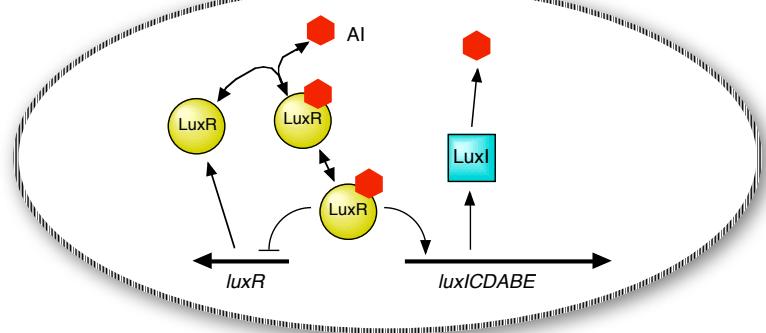
LuxI	LuxR:AI:Genome
0	0
1	1

How does LuxI depend on LuxR:AI:Genome?

LuxR:AI:Genome	LuxR:AI
0	0
1	1

How does LuxR:AI:Genome depend on LuxR:AI?

# Condition Tables for QS II



LuxR	LuxR	AI	LuxR:AI:Genome
	0	0	0
		0	0
	0		0
			0
0	0	0	
		0	
0	0		
0			

LuxR:AI	LuxR	AI	LuxR:AI:Genome
0	0	0	0
0		0	0
	0		0
			0
0	0	0	
0		0	
0	0		

=>

LuxR:AI	LuxR	AI	LuxR:AI:Genome
0	x	0	x
			x
	0		0
0	0		

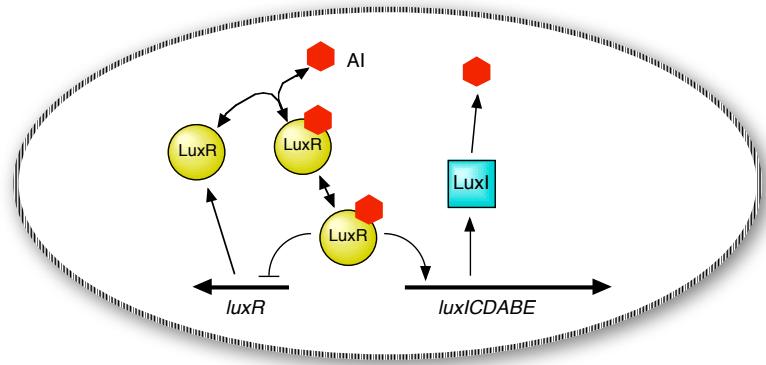
Note: no dissociation

(LuxR:AI:Genome => LuxR:AI + Genome)

only degradation of AI

LuxR:AI:Genome => LuxR + Genome

# Condition tables III



AI	LuxR	AI	LuxI
0	0	0	0
0	1	0	0
1	0	1	0
0	1	1	0
1	0	0	1
1	1	0	1
1	0	1	1
1	1	1	1

=>

AI	LuxR	AI	LuxI
1	x	x	1
0	x	0	0
1	0	1	0
0	1	1	0

# Scanning for Attractors

States of *V. fischeri* QS system mapped onto integers

$$\{\text{LuxR (LR), LuxR:AI (RA), AI, LuxR:AI:Genome (RAG), LuxI (LI)}\} = \{1, 2, 4, 8, 16\}$$

For each **attractor**:

- periodic orbit and its length (period)
- basin of attraction and its relative size (32 states in total)  
=> how likely will the system end in each of the attractors?

**Attractor I:**    orbit: 1                   => period 1  
                          states: 0, 1                => size 2,  $2/32 = 6.25\%$

#	LR	RA	AI	RAG	LI	-	state
0	.	.	.	.	.	-	0
1	X	.	.	.	.	-	1
2	X	.	.	.	.	-	1

=> attractor

# Scanning for Attractors II

**Attractor 2:**    orbit: 3, 9, 17, 5                          => period 4

states: 2, 3, 5, 8, 9, 16, 17    => size 7, 21.9 %

start from state 8:

#	LR	RA	AI	RAG	LI	-	state
0	.	.	.	X	.	-	8
1	.	.	.	.	X	-	16
2	X	.	X	.	.	-	5
3	X	X	.	.	.	-	3
4	X	.	.	X	.	-	9
5	X	.	.	.	X	-	17
6	X	.	X	.	.	-	5

attractor

averaged occupancies in this periodic orbit:

LR	RA	AI	RAG	LI
4/4 = 1	1/4 = 0.25	1/4 = 0.25	1/4 = 0.25	1/4 = 0.25

# Attractors III

**Attractor 3:** period 4, basin of 16 states => 50 %

#	LR	RA	AI	RAG	LI	-	state
0	.	X	X	.	.	-	6
1	.	X	X	X	.	-	14
2	.	.	X	X	X	-	28
3	.	.	X	.	X	-	20

**Attractor 4:** period 4, basin of 4 states => 12.5 %

#	LR	RA	AI	RAG	LI	-	state
0	X	X	X	.	.	-	7
1	X	X	.	X	.	-	11
2	X	.	.	X	X	-	25
3	X	.	X	.	X	-	21

**Attractor 5:** period 2, basin of 3 states => 9.4 %

#	LR	RA	AI	RAG	LI	-	state
0	X	.	X	X	.	-	13
1	.	X	.	.	X	-	18

# Classifying the Attractors

=> Interpret the system's behavior from the properties of the attractors

Attractor	period	basin size	$\langle \text{LuxR} \rangle$	$\langle \text{LuxR:AI} \rangle$	$\langle \text{AI} \rangle$	$\langle \text{LuxR:AI:Gen} \rangle$	$\langle \text{LuxI} \rangle$
1	1	6.25 % (2)	1	0	0	0	0
2	4	21.9% (7)	1	0.25	0.25	0.25	0.25
3	4	50 % (16)	0	0.5	1	0.5	0.5
4	4	12.5 % (4)	1	0.5	0.5	0.5	0.5
5	2	9.4% (3)	0.5	0.5	0.5	0.5	0.5

Three **regimes**:

**dark**: LuxI = 0

free LuxR, no AI

**intermediate**: LuxI = 0.25

free LuxR + little AI

**bright**: LuxI = 0.5

little free LuxR (0.24)  
+ much AI (0.85)

# The Feed-Forward-Loop

External signal determines state of X  
=> response Z for short and long signals X

condition tables:

X	Y	Z	X	Y	Z
0	0	0	0	0	0
1	1	1	1	1	1

X	Y	Z	X	Y	Z
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	1	0	0
1	1	1	1	1	1

Response to signal X(t)

X	Y	Z
0	0	0
-	0	0
0	-	0
0	0	0
-	0	0
0	-	0
0	0	-
0	0	0

X	Y	Z
0	-	0
-	-	0
0	0	0
0	-	0
-	-	0
0	0	0
-	0	0
0	-	0
0	0	-
0	0	0

# The *A. thaliana* Flowering Network



## Model organism in genomics:

- small, convenient to grow
- completely sequenced (2000): 125 Mbp
- easily mutated

also see: Arabidopsis Information Resource (TAIR)  
@ [www.arabidopsis.org/](http://www.arabidopsis.org/)

# Dynamics of the Genetic Regulatory Network for *Arabidopsis thaliana* Flower Morphogenesis

LUIS MENDOZA AND ELENA R. ALVAREZ-BUYLLA\*

*Lab. de Genética Molecular y Evolución, Instituto de Ecología, UNAM, Ap. Postal 70-275,  
Coyoacán, D.F. CP04510, Mexico*

(Received on 26 September 1997, Accepted in revised form on 3 March 1998)

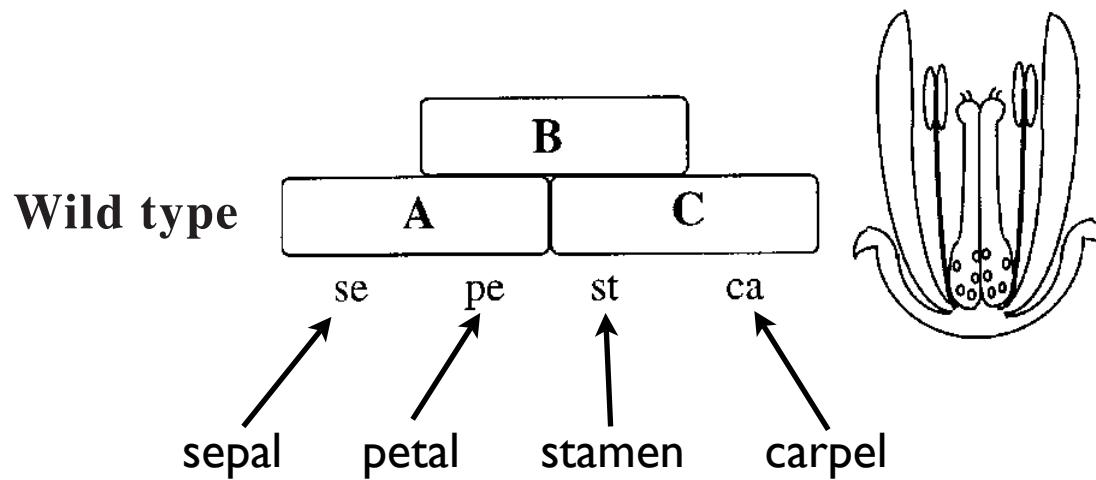
We present a network model and its dynamic analysis for the regulatory relationships among 11 genes that participate in *Arabidopsis thaliana* flower morphogenesis. The topology of the network and the relative strengths of interactions among these genes were based from published genetic and molecular data, mainly relying on mRNA expression patterns under wild type and mutant backgrounds. The network model is made of binary elements and we used a particular dynamic implementation for the network that we call semi-synchronous. Using this method the network reaches six attractors; four of them correspond to observed patterns of gene expression found in the floral organs of *Arabidopsis* (sepals, petals, stamens and carpels) as predicted by the ABC model of flower morphogenesis. The fifth state corresponds to cells that are not competent to flowering, and the sixth attractor predicted by the model is never found in wild-type plants, but it could be induced experimentally. We discuss the biological implications and the potential use of this network modeling approach to integrate functional data of regulatory genes of plant development.

© 1998 Academic Press

# The ABC Model

Coen, Meyerowitz (1991):

three different activities A, B, and C, active in two adjacent whorls,  
mutual inhibition of A and C  
=> combinations determine fate of the tissue



Related genes:

**A:**

*APETALA1 (API)*

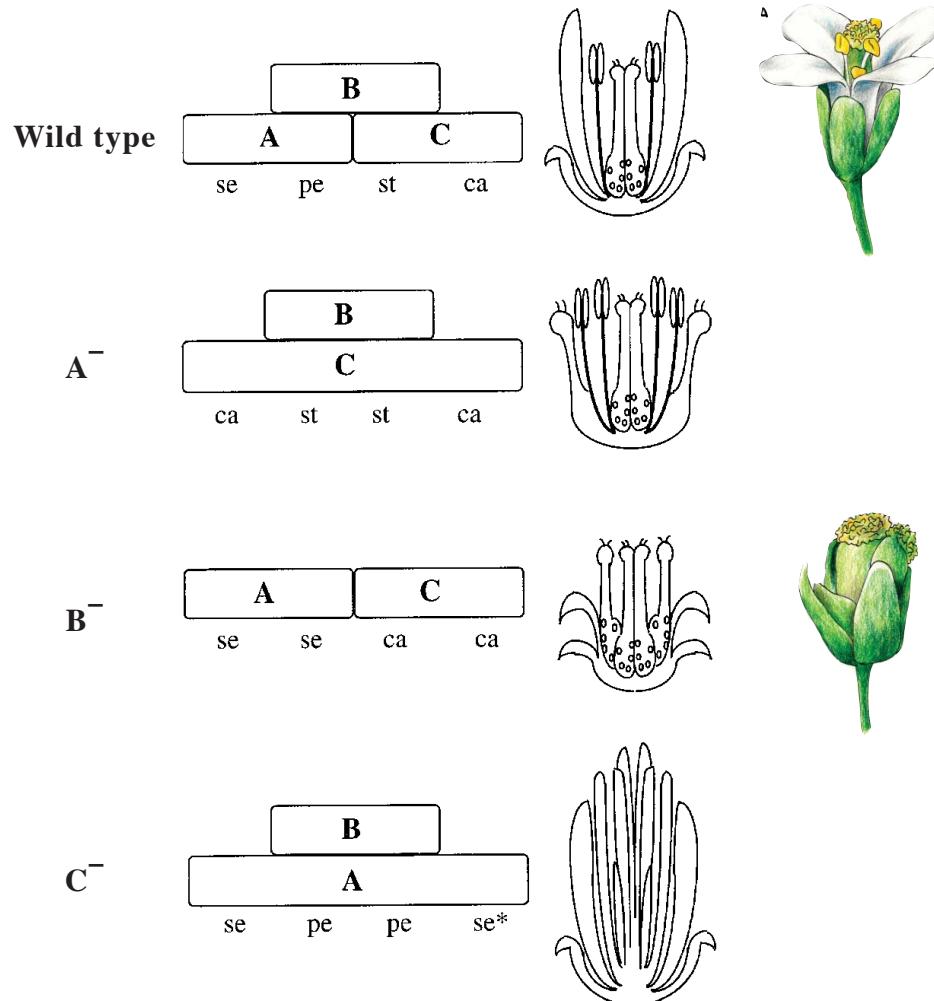
**B:**

*APETALA3 (AP3),  
PISTILATA (PI)*

**C:**

*AGAMOUS (AG)*

# ABC Mutants

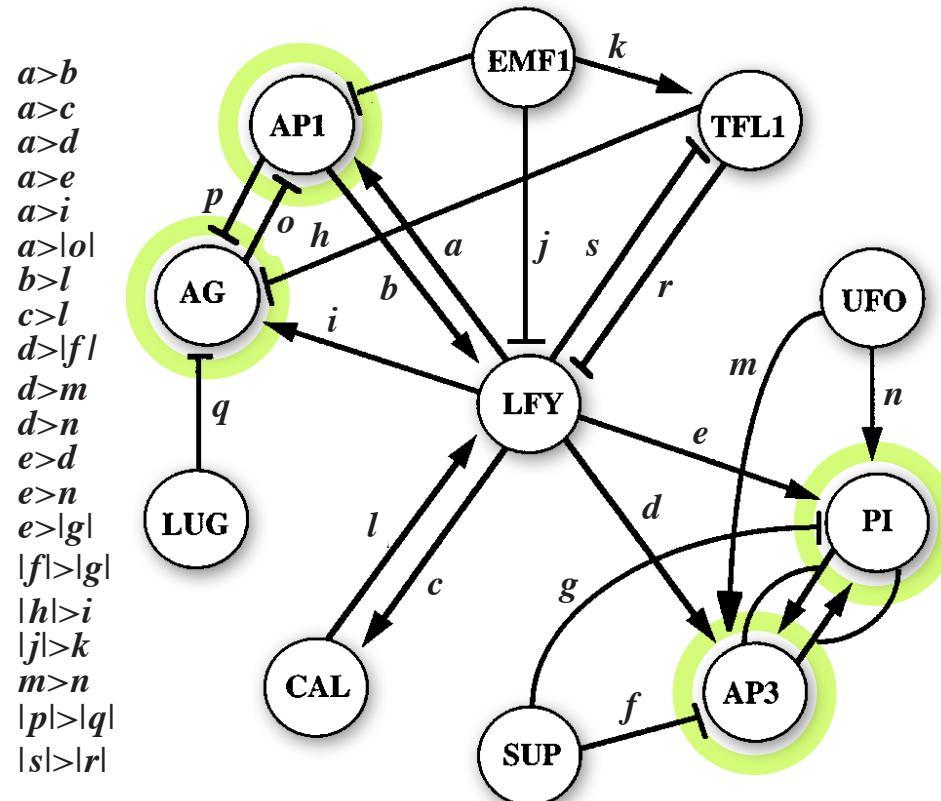


If any of the three functions (activities) is missing, the flowers have different tissue combinations.

se = sepals,  
pe = petals,  
st = stamens,  
ca = carpels,  
se\* = se, pe, pe

# The Network Model

11 genes (including the four ABC genes)



inequalities denote the relative  
weights of the interactions

Mendoza, Alvarez-Buylla, *J. theor Biol.* **193** (1998) 307

# Model Implementation

Here: Boolean model with **weighted** interactions

Propagate state vector  $\mathbf{x} = \{x_1, x_2, \dots, x_{11}\}$  by:

$$x_i(t + 1) = \mathbf{H} \left( \sum_{j=1}^N w_{ij} x_j(t) - \theta_i \right)$$

Heavyside step function:  $\mathbf{H}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases}$

Weights  $w_{ij}$  and threshold  $\theta_i$  are not known exactly  
=> choose integers for simplicity  
=> positive for activation, negative for inhibition

# The Numbers

*EMF1, TFL1, LFY, AP1, CAL, LUG, UFO, BFU, AG, AP3, PI and SUP.*

$$\mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & -2 & 1 & -2 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & -2 \\ 0 & 0 & 4 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \boldsymbol{\theta} = \begin{bmatrix} 0 \\ 0 \\ 3 \\ -1 \\ 1 \\ 0 \\ 0 \\ 1 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Mendoza, Alvarez-Buylla, *J. theor Biol.* **193** (1998) 307

# Synchronous vs. Asynchronous

**Synchronous** propagation (Kauffman (1969)):

=> update all species **simultaneously**

=> biological problem: do all genes respond at exactly the same time?

**Asynchronous** propagation (Thomas (1991)):

=> update one species after the other **in chosen order**

=> order of update may influence dynamic gene activation patterns

**Semi-synchronous** propagation (Mendoza (1998)):

=> split genes in groups:

=> synchronous within group, one group after the other

=> base order of groups upon experimental data (it's still a "choice")

*EMF1, TFL1 => LFY, API, CAL => LUG, UFO, BFU => AG, AP3, PI => SUP*

Mendoza, Alvarez-Buylla, *J. theor Biol.* **193** (1998) 307

# Some Example Patterns

$t=0$	<b>1 0 1 1 1 1 1 1 0 0 1 1</b>
$t=1$	1 0 <b>1 1 1</b> 1 1 1 0 0 1 1
$t=2$	1 0 0 1 1 <b>1 1 1</b> 0 0 1 1
$t=3$	1 0 0 1 1 0 0 0 <b>0 0 1</b> 1
$t=4$	1 0 0 1 1 0 0 0 0 0 <b>1</b>
$t=5$	<b>1 0 0 1 1 0 0 0 0 0 0 0</b>
$t=6$	1 1 <b>0 1 1</b> 0 0 0 0 0 0 0
$t=7$	1 1 0 0 0 <b>0 0 0 0 0 0 0 0</b>

$t=0$	<b>0 1 0 0 0 0 0 0 0 0 0 0</b>
$t=1$	0 0 <b>0 0 0 0 0 0 0 0 0 0</b>
$t=2$	0 0 0 1 0 <b>0 0 0 0 0 0 0</b>

$t=0$	<b>0 0 0 0 0 1 0 1 1 1 0 0</b>
$t=1$	0 0 <b>0 0 0 1 0 1 1 1 0 0</b>
$t=2$	0 0 0 0 0 <b>1 0 1 1 1 0 0</b>
$t=3$	0 0 0 0 0 0 0 <b>1 1 0 0</b>
$t=4$	0 0 0 0 0 0 0 0 <b>1 0 0 0</b>

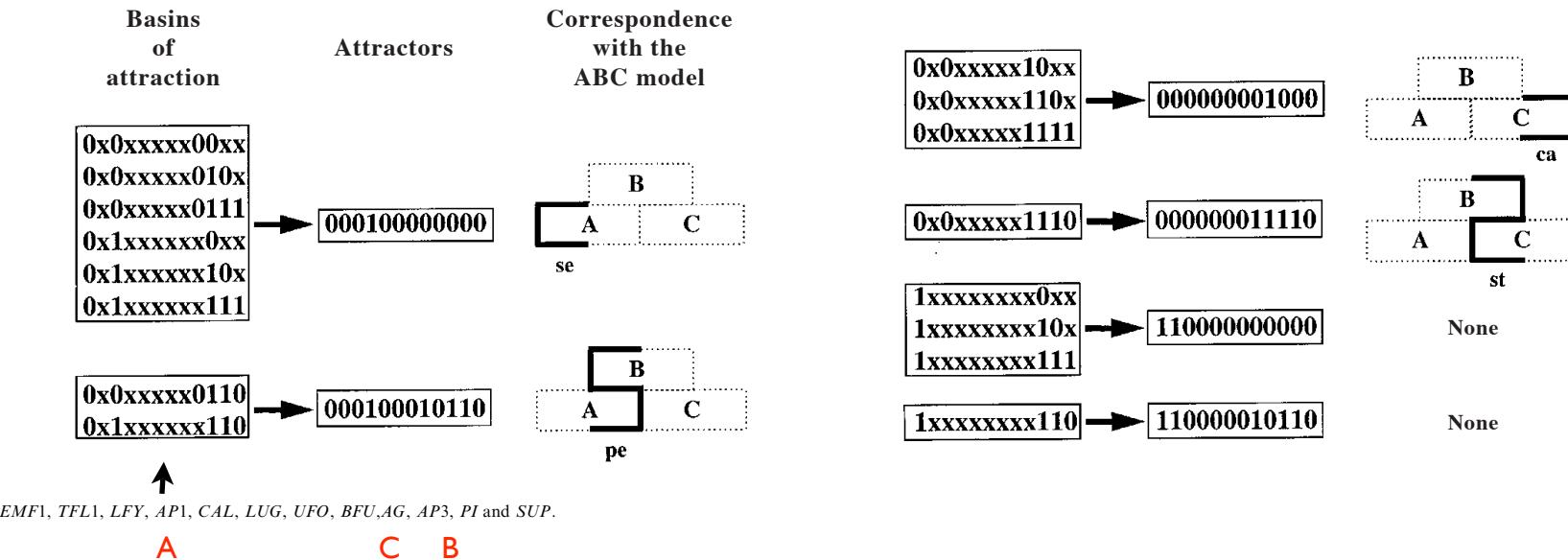
$t=0$	<b>1 1 1 1 1 1 0 0 1 1 0</b>
$t=1$	1 0 <b>1 1 1</b> 1 1 0 0 1 1 0
$t=2$	1 0 0 1 1 <b>1 1 0</b> 0 1 1 0
$t=3$	1 0 0 1 1 0 0 1 <b>0 1 1</b> 0
$t=4$	1 0 0 1 1 0 0 1 0 1 1 <b>0</b>
$t=5$	<b>1 0 0 1 1 0 0 1 0 1 1 0</b>
$t=6$	1 1 <b>0 1 1</b> 0 0 1 0 1 1 0
$t=7$	1 1 0 0 0 <b>0 0 1</b> 0 1 1 0

$t=0$	<b>0 1 0 0 0 1 0 1 1 1 1 0</b>
$t=1$	0 0 <b>0 0 0 1 0 1 1 1 1 0</b>
$t=2$	0 0 0 0 0 <b>1 0 1 1 1 1 0</b>
$t=3$	0 0 0 0 0 0 0 <b>1 1 1 0</b>

$t=0$	<b>0 0 0 0 0 0 1 0 0 1 1 0</b>
$t=1$	0 0 <b>0 0 0 0 0 1 0 0 1 1 0</b>
$t=2$	0 0 0 1 0 <b>0 1 0 0 1 1 0</b>
$t=3$	0 0 0 1 0 0 0 <b>1 0 1 1 0</b>

**Exhaustive** search: start from **all**  $2^{12} = 4096$  possible **initial** states,  
run for  $t = 200$  steps  
=> **six stationary** patterns (attractors of size 1)

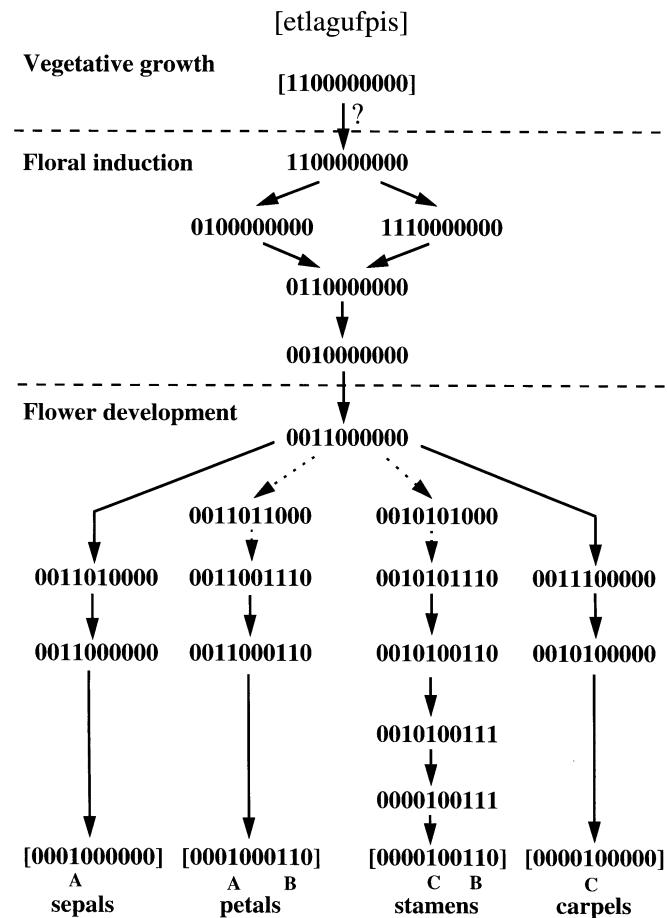
# The Attractors



From gene **activation patterns** in the attractors:

- => identify the **four floral tissue types** of the ABC model
- => one attractor with floral **inhibitors** EMF1, TFL1  
(characteristic for cells that are not part of the flowers)
- => one yet **unidentified** state

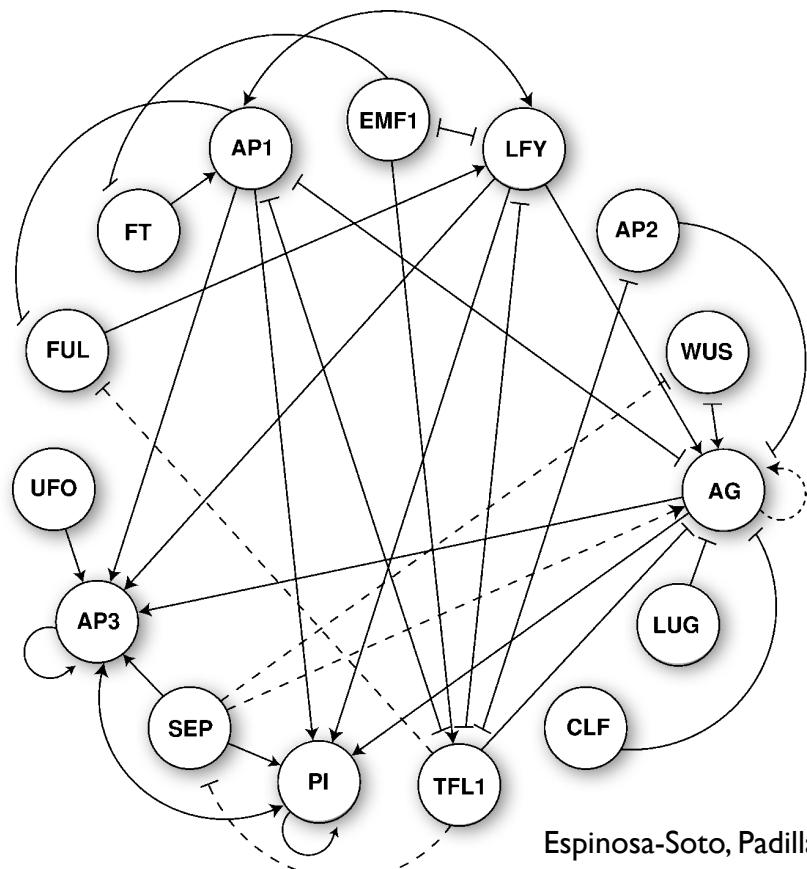
# Possible Pathways



Note: the model does not include temporal and spatial information required to predict where and when which genes are activated or repressed ("signals")  
=> these pathways are a "proposal" only

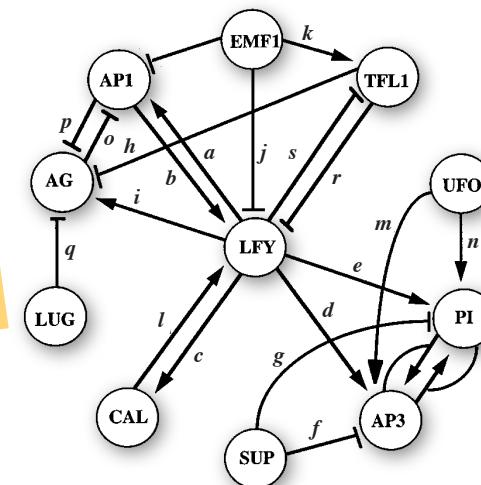
# Evolution of Networks

A few years later: additional genes and predicted interactions (---)



Espinosa-Soto, Padilla-Longoria, Alvarez-Buylla,  
*The Plant Cell* **16** (2004) 2923

2004  
1998

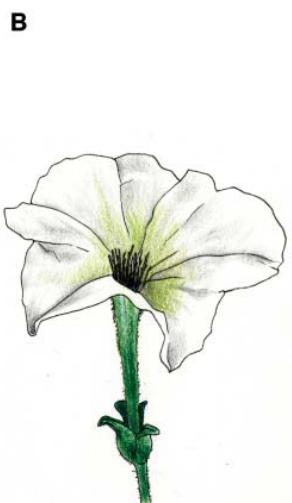


Mendoza, Alvarez-Buylla,  
*J. theor Biol.* **193** (1998) 307

# Predictions for Petunia



From *A. thaliana* predict/  
understand green petals  
mutant phenotype for petunia.



Espinosa-Soto, Padilla-Longoria, Alvarez-Buylla,  
*The Plant Cell* 16 (2004) 2923

# What is it Worth?

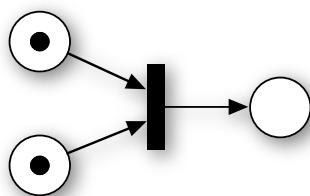
Generally: => quality of the **results** depends on the quality of the **model**  
=> quality of the model depends on the quality of the **assumptions**

**Assumptions** for the Boolean network description:

- (• subset of the species considered => reduced system state space)
- only discrete density levels => dynamic balances lost,  
• conditional yes–no causality reduced to oscillations  
=> no continuous processes
- discretized propagation steps => timing of concurrent paths?

"You get what you pay for"

# Petri-Nets

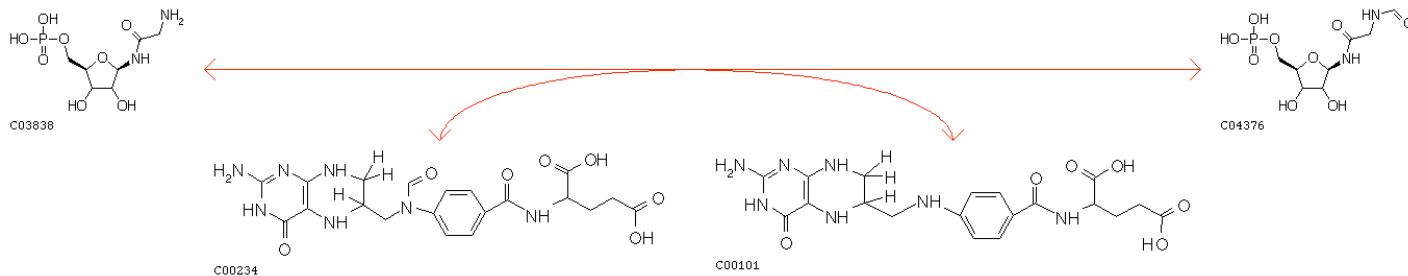


Bipartite **graph** of

- places
- transitions
- directed weighted arcs

} two types of nodes

Metabolic reaction:



place  
=  
metabolite

transition  
=  
enzyme

weighted arc  
=  
stoichiometries

# Petri Nets: More Accurate

**Places:** have a capacity ( $1 \dots \infty$ )

=> max. number of tokens (default:  $\infty$ )

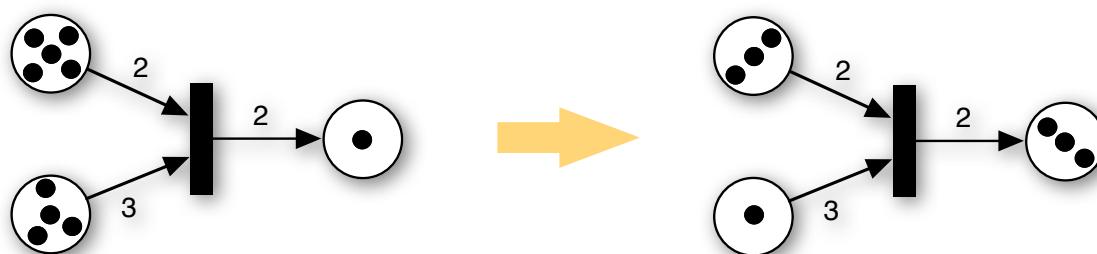
**Arcs:** have costs ( $1 \dots \infty$ )

=> number of tokens that are consumed/produced (default: 1)

**Transitions:** can fire, when the conditions are fulfilled

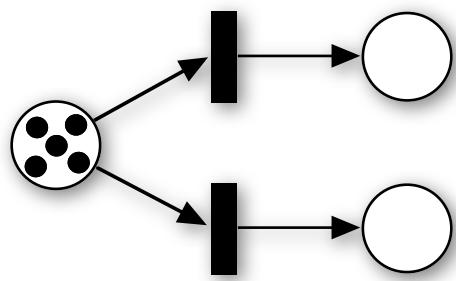
=> enough tokens on the in-places:  $\geq$  costs for in-arcs

=> enough remaining capacity on the out-places:  $\geq$  costs for out-arcs



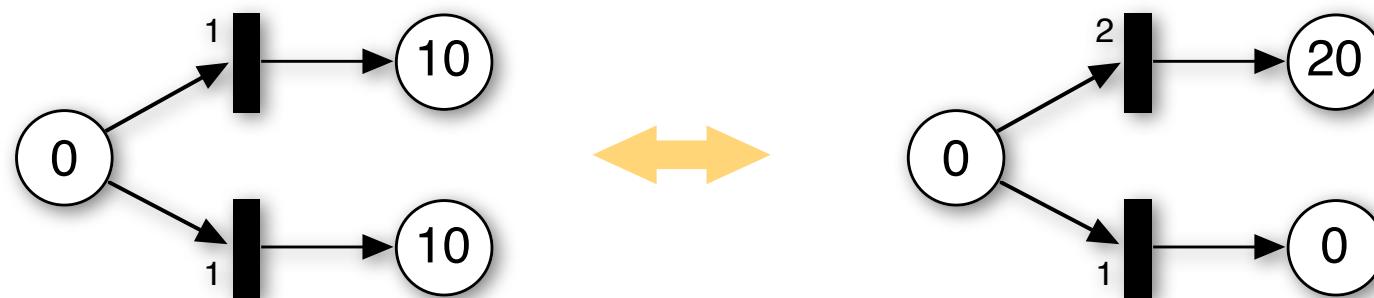
**Marking** = state of the network = numbers of tokens on the places

# Multiple Possibilities

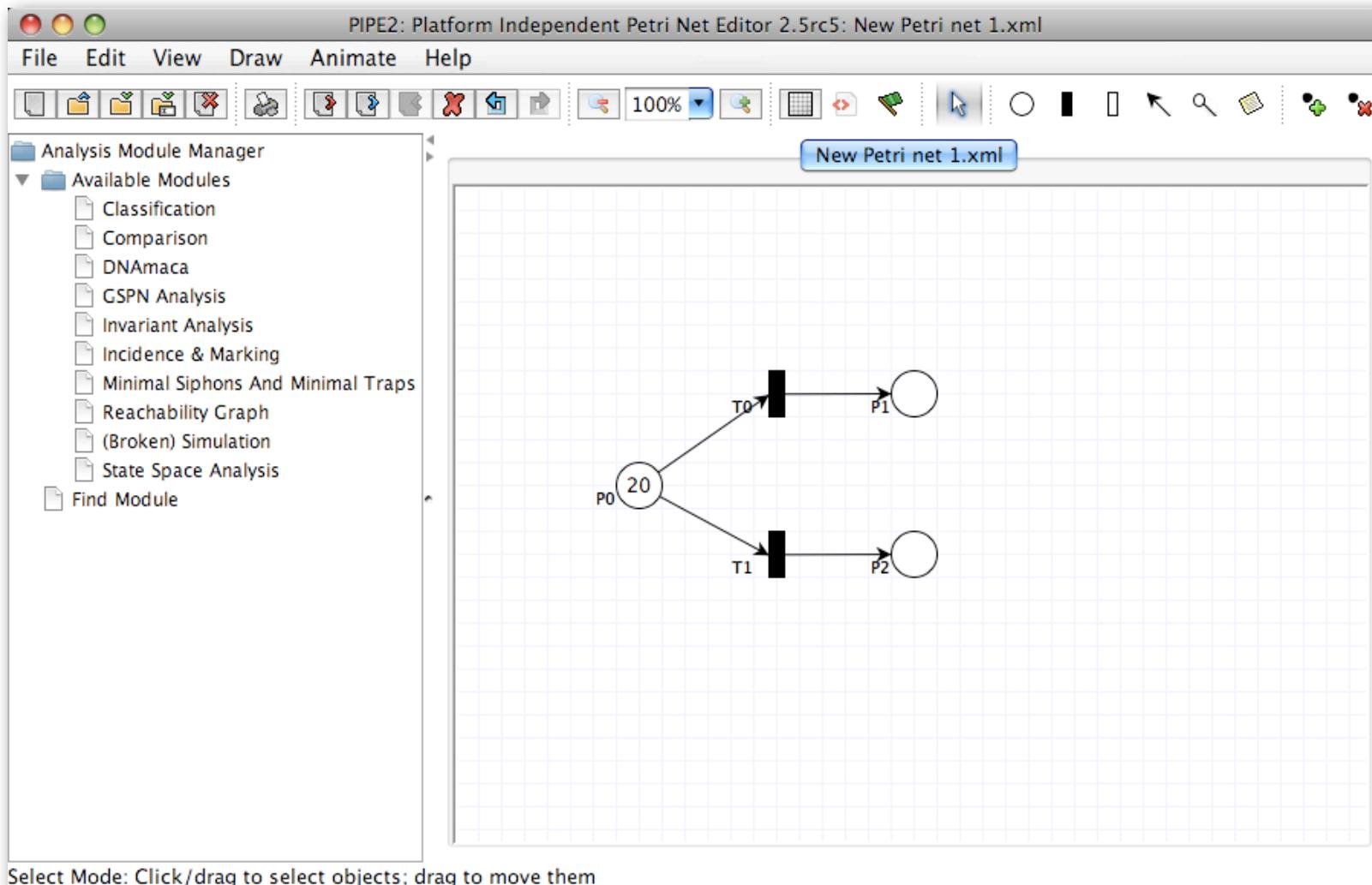


When **multiple transitions** may fire:

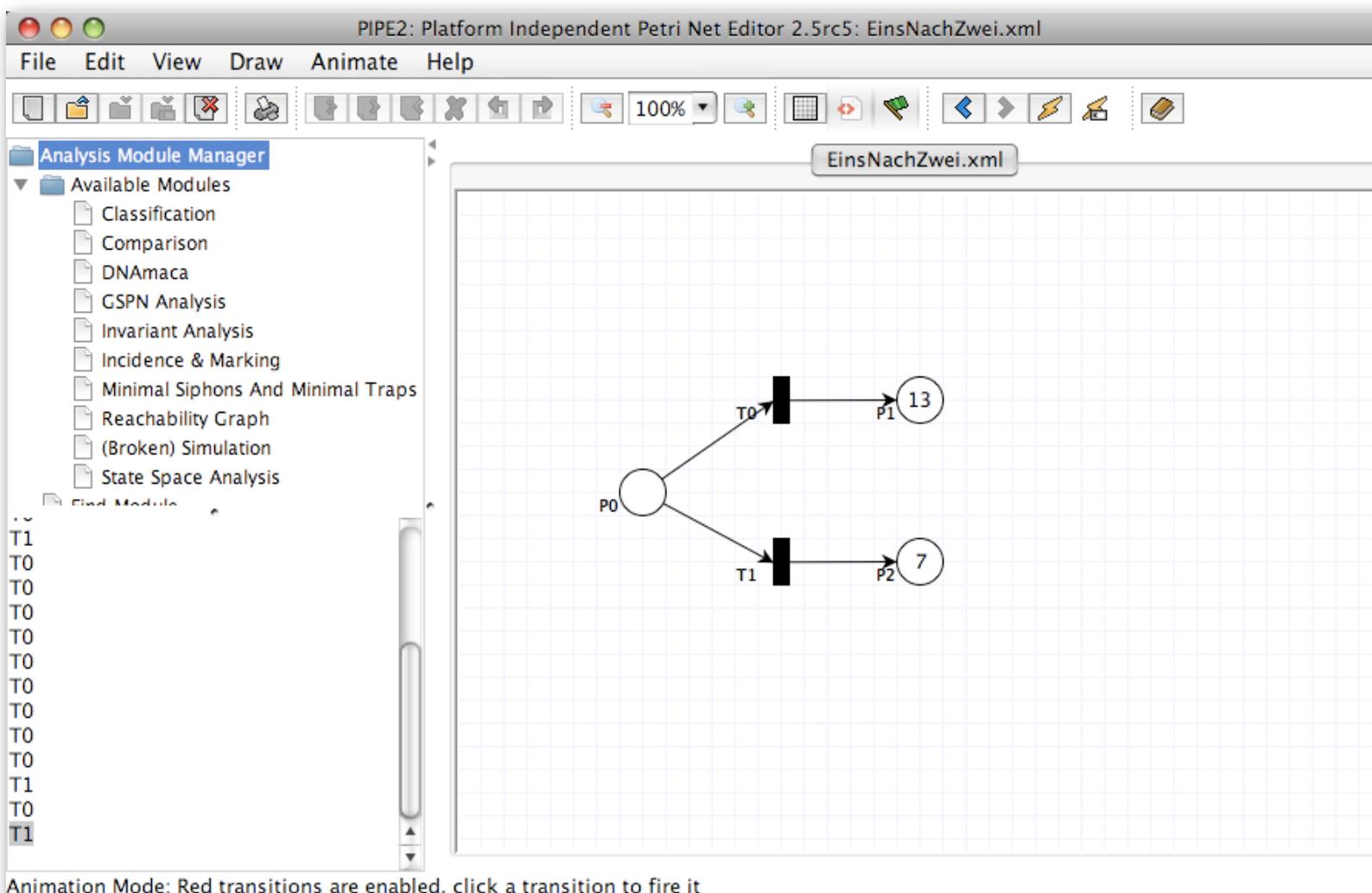
- all are equal  
=> choose one randomly
- if priorities are defined  
=> transition with highest priority fires



# Platform Independent Petri Net Editor



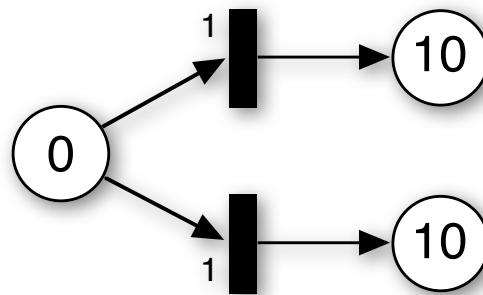
# "Token Game"



# Token Spread

Token Game = stochastic simulation

Run	P1	P2
1	10	10
2	15	5
3	11	9
4	9	11
5	13	7
6	7	13
7	7	13
8	5	15
9	9	11
10	8	12
<hr/>		
$\langle N \rangle$		9.4
$\sigma$		2.8
<hr/>		
		10.6
		2.8

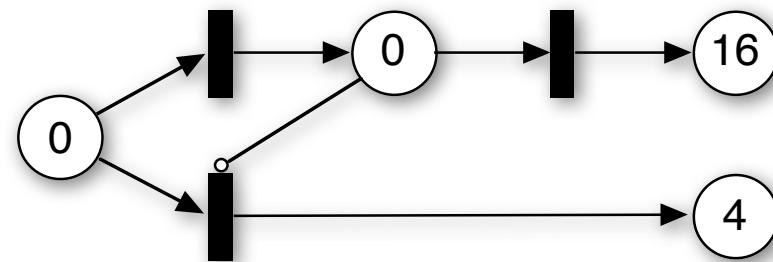
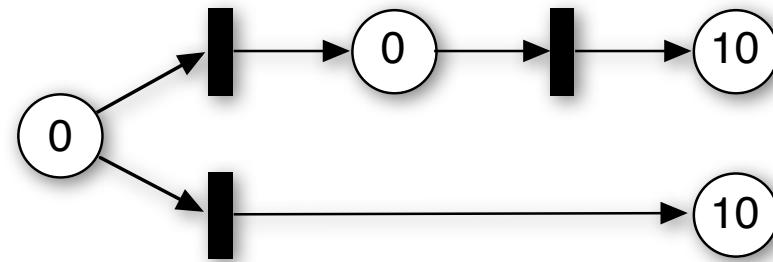


for comparison:  
expected from Poisson distribution

$$\lambda = 10$$

$$\sigma = \lambda^{1/2} \approx 3.2$$

# Inhibition



# Time Consuming Transitions

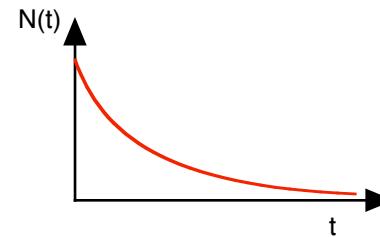
Until now: every transition was instantaneous

**SPN** (Stochastic Petri Net):

Each transitions takes some time – exponentially distributed waiting times

$$\frac{dN}{dt} = -k N \quad \Rightarrow \quad N(t) = N(0) e^{-kt}$$

=> survival times distributed exponentially



**GSPN** (Generalized Stochastic Petri Net):

Time-consuming and instantaneous transitions

**DSPN** (Deterministic Stochastic Petri Net):

Waiting times are fixed or exponentially distributed

General Petri nets: all types of transitions may occur

# Analysis

"Token Game" simulations

- => phenomenologically: what happens, does the model work?
- => stationary states? (we're stuck...)
- => periodic orbits?
- => relative probabilities of certain states?

"State space analysis"

- => average number of tokens, distributions, throughputs
- => reachability of markings (states)
  - => liveness
  - => deadlocks, traps, siphons

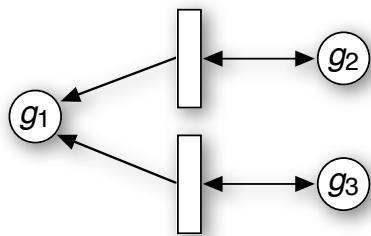
# Petri Nets for Gene Regulation

To encode the dependencies of gene regulation we need:

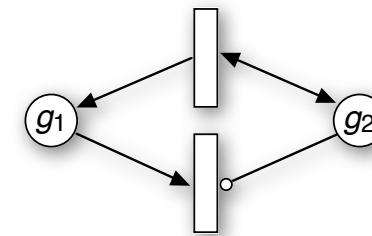
**activation, inhibition, logical and, logical or**

- transcription factors are not consumed => read arcs
- encode on/off states => capacity constraints on the places

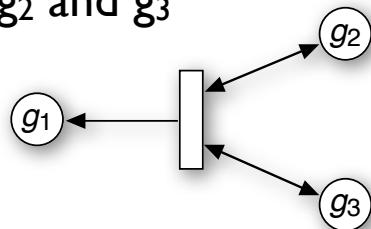
$g_1 = g_2 \text{ or } g_3$



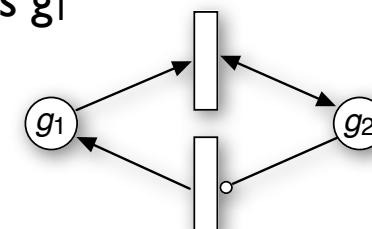
$g_2$  activates  $g_1$



$g_1 = g_2 \text{ and } g_3$



$g_2$  inhibits  $g_1$

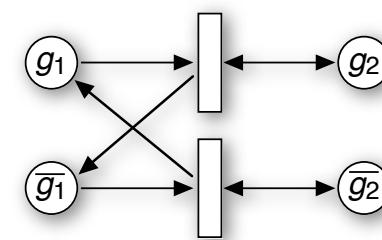
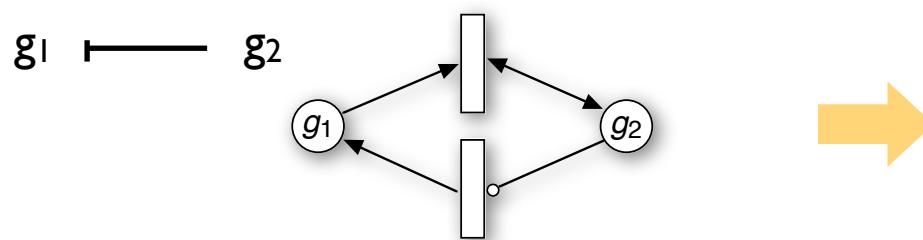
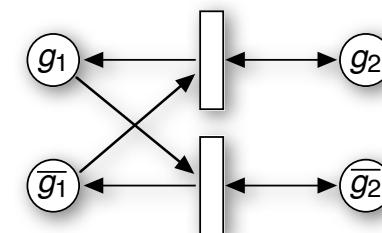
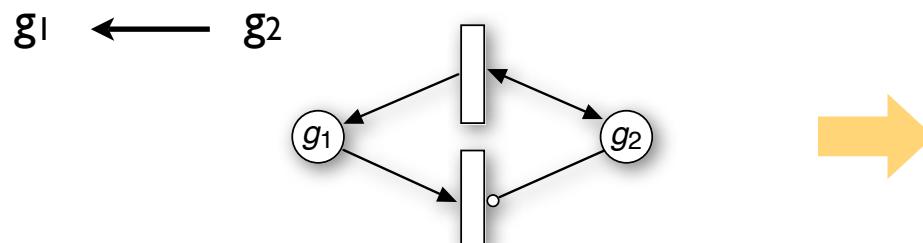


# Boolean Regulatory Petri Nets

Introduce complementary places: tokens on  $g_1$  plus on  $\bar{g}_1 = 1$

=> capacity constraints fulfilled automatically (when initial markings are okay)

=> no inhibitory arcs required



# Reverse Engineering Networks

Problem: "Find **the** network that explains the biological processes!"

=> usually too ambitious

Experiments: co-expression data

=> co-regulation of different genes (correlation or direct interaction?)

=> time-series of individual genes

Strategies: "Find **all** networks that are compatible with the experiments"

=> combinatorial explosion, usually too many candidates

=> does not work...

"Find **one** network that is compatible with the experiments"

=> solvable task, but how good is this network?

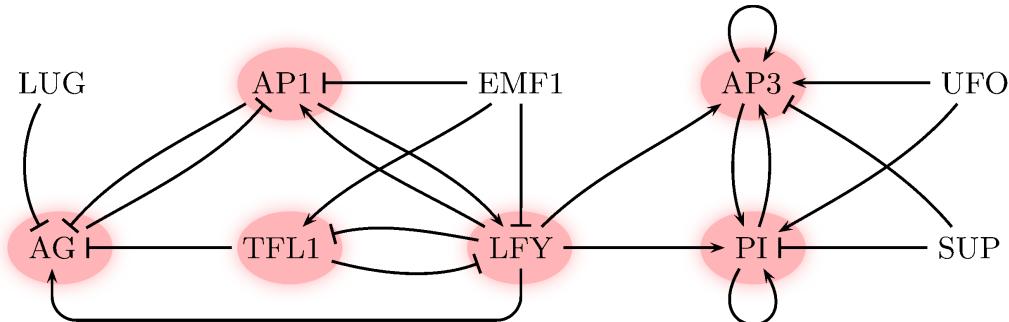
=> does not work...

"Find **some** networks that are compatible with the experiments"

=> algorithms exist, need heuristics (experience) to assess coverage

=> does work...

# Flowering in *Arabidopsis*



Minimal model of flower  
morphogenesis in *A. thaliana*  
=> only "red" genes

Identify steady states of different parts of the flower  
=> find **dead** markings

$$\begin{bmatrix} M_d^1(T) = 0 \\ M_d^1(L) = 0 \\ M_d^1(A) = 1 \\ M_d^1(G) = 0 \\ M_d^1(P) = 0 \\ M_d^1(I) = 0 \end{bmatrix} \quad \begin{bmatrix} M_d^2(T) = 0 \\ M_d^2(L) = 0 \\ M_d^2(A) = 1 \\ M_d^2(G) = 0 \\ M_d^2(P) = 1 \\ M_d^2(I) = 1 \end{bmatrix} \quad \begin{bmatrix} M_d^3(T) = 0 \\ M_d^3(L) = 0 \\ M_d^3(A) = 0 \\ M_d^3(G) = 1 \\ M_d^3(P) = 0 \\ M_d^3(I) = 0 \end{bmatrix} \quad \begin{bmatrix} M_d^4(T) = 0 \\ M_d^4(L) = 0 \\ M_d^4(A) = 0 \\ M_d^4(G) = 1 \\ M_d^4(P) = 1 \\ M_d^4(I) = 1 \end{bmatrix}$$

Sepals

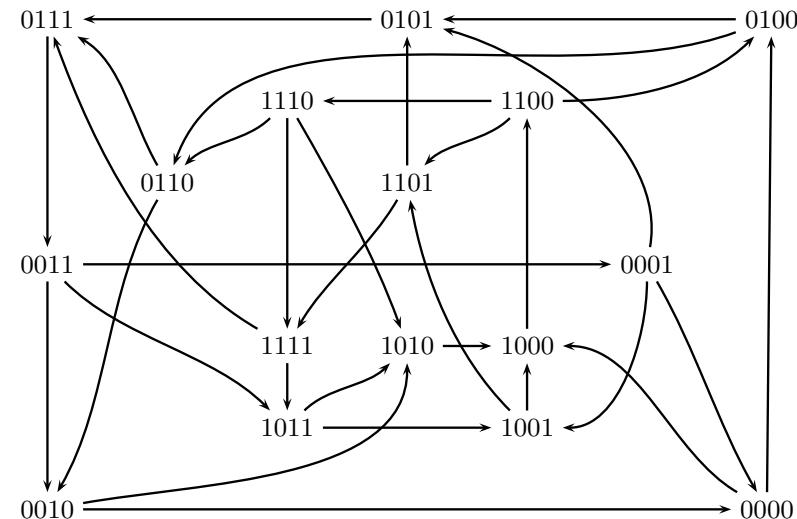
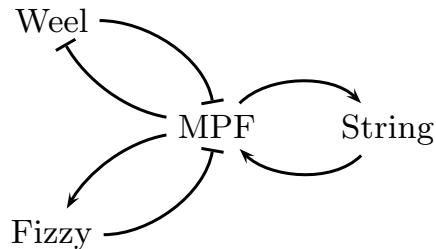
Petals

Carpels

Stamens

# Drosophila Cell Cycle

Minimal regulation network for the first cell cycles during *D. melanogaster* embryonic development  
(MPF = Mitosis Promoting Factor)



Asynchronous graph of all possible states  
(and transitions) — MFWS

Does the model reproduce **oszillations**?

=> prove that the system is **deadlock-free**

(evaluate conditions that any of the transitions cannot fire any more)

Note on a mutation: when MPF inhibits Fizzy => dead markings => no stable oszillations

# Summary

**Today:** simplified modelling of gene regulation networks

- Boolean Networks
  - genes are on/off, propagation via condition tables
  - => direct implementation of experimentally found dependencies
  - => no real-time information
  - => steady states (attractors) — network reconstruction — mutations
- Petri nets
  - places, transitions, and arcs (plus capacities)
  - => more general, more analysis tools, but more complex
  - => can include real-time dynamics (via time-consuming transitions)

**Next lectures:** Prof. Helms

- network reconstruction
- metabolic networks, static and dynamic