Formal Methods for Discrete Gene Regulatory Networks

Gilles Bernot

University Nice Sophia Antipolis, I3S laboratory, France

Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Mathematical models in biology: what for ?

Different purposes $\Longrightarrow$ different approaches

- Models as intelligent "Data Base" to store biological knowledge
- *Models as tools for establishing causality chains*
- *Models as design tools for synthetic biology*
- *Models as guidelines for the choice of experiments*

For the 3 last purposes, models can deviate from biological descriptions, while remaining very useful, because they are *dedicated* to the question under consideration.

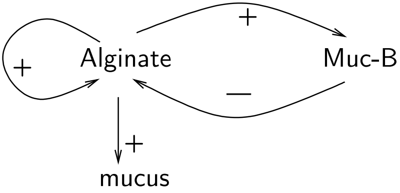
"Kleenex" models...

Static Graph *v.s.* Dynamic Behaviour

Difficulty to predict the result of combined regulations

Difficulty to measure the strength of a given regulation

Example of “competitor” circuits



Multistationarity ?
Homeostasy ?

Many underlying qualitative models: $\approx$ *700 qualitative behaviours*

Mathematical Models and Simulation

1. Rigorously encode sensible knowledge, into ODEs for instance
2. - A few parameters are approximatively known
   - Some parameters are limited to some intervals
   - Many parameters are *a priori* unknown
3. Perform lot of simulations, compare results with known behaviours, and propose some credible values of the unknown parameters which produce robust acceptable behaviours
4. Perform additional simulations reflecting novel situations
5. If they predict interesting behaviours, propose new biological experiments
6. Better tune the model parameters and try to go further

. . . *not my cup of tea* . . .

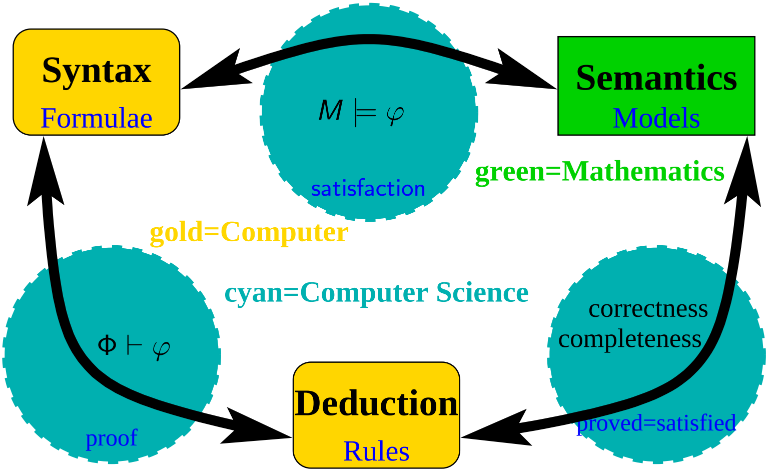
Mathematical Models and Validation

"Brute force" simulations are not the only way to use a computer.
There are computer aided environments which help:

- designing simplified models that can be analytically solved
- avoiding models that can be "tuned" *ad libitum*
- *constraining* models according to experimental data
- *validating* models with a reasonable number of experiments
- defining only models that could be experimentally *refuted*
- proving refutability w.r.t. experimental capabilities

To establish a *methodology* "dry" models ↔ "wet" experiments
one needs to assist reasonning capabilities.

Formal Logic: syntax/semantics/deduction



Syntax

Formulae

$M \models \varphi$

satisfaction

Semantics

Models

green=Mathematics

gold=Computer

cyan=Computer Science

$\Phi \vdash \varphi$

proof

Deduction

Rules

correctness

completeness

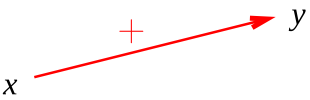
proved=satisfied

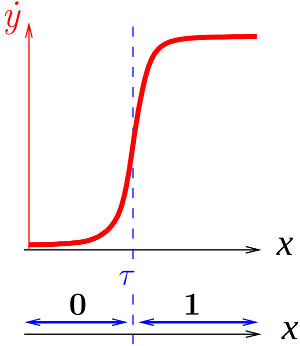
Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Multivalued Regulatory Graphs

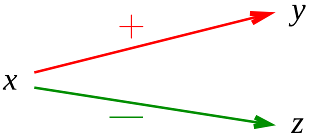
Derivatives are sigmoids w.r.t. the source gene

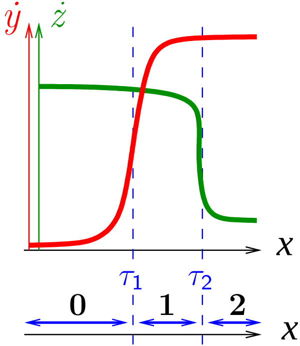




Multivalued Regulatory Graphs

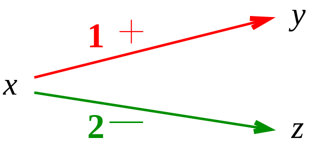
Derivatives are sigmoids w.r.t. the source gene

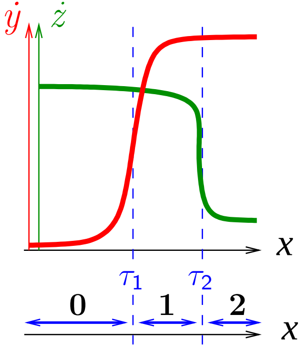




Multivalued Regulatory Graphs

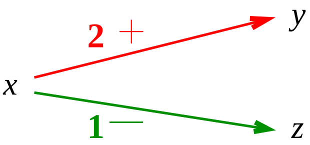
Derivatives are sigmoids
w.r.t. the source gene

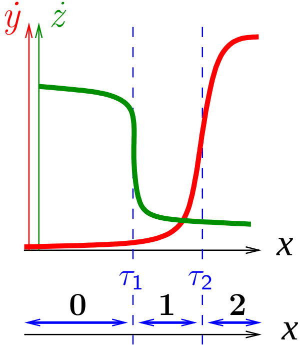




Multivalued Regulatory Graphs

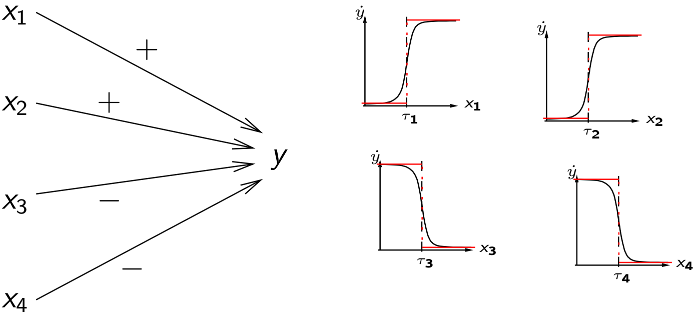
Derivatives are sigmoids w.r.t. the source gene





First simplification: piecewise linear

Approximate sigmoids as step functions:



Presence of an activator = Absence of an inhibitor

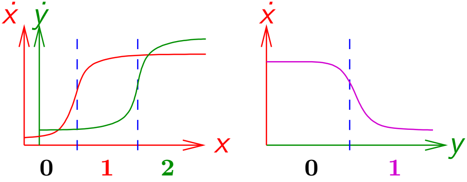
$$\frac{dy}{dt} = k_0 + k_1.\mathbb{1}_{x_1 \geqslant \tau_1} + k_2.\mathbb{1}_{x_2 \geqslant \tau_2} + k_3.\mathbb{1}_{x_3 < \tau_3} + k_4.\mathbb{1}_{x_4 < \tau_4} - \gamma.y$$

Solutions of the form $Ce^{-\gamma t} + \frac{\Sigma \mathbb{1} k_i}{\gamma}$ whose $\lim_{t \to \infty}$ is $\frac{\Sigma \mathbb{1} k_i}{\gamma}$

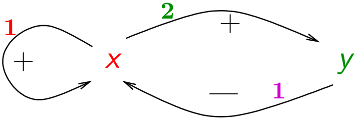
As many such equations as genes in the interaction graph

In each hypercube, all the trajectories have a unique *attractive point*, which can be outside de hypercube

Discrete Gene Networks (Thomas & Snoussi)



In each state,
a variable v tries to
go toward the interval
numbered $K_{v,\omega}$:
the one containing $\frac{\Sigma \mathbb{1} k_i}{\gamma}$



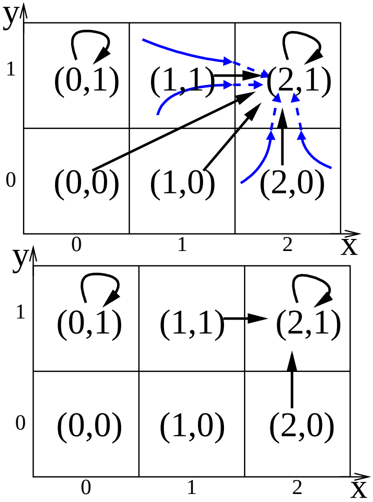
No help :	K_x	K_y
x helps :	$K_{x,x}$	$K_{y,x}$
Absent y helps :	$K_{x,\overline{y}}$	
Both :	$K_{x,x\overline{y}}$	

(x,y)	*Focal Point*
(0,0)	$(K_{x,\overline{y}}, K_y)$
(0,1)	(K_x, K_y)
(1,0)	$(K_{x,x\overline{y}}, K_y)$
(1,1)	$(K_{x,x}, K_y)$
(2,0)	$(K_{x,x\overline{y}}, K_{y,x})$
(2,1)	$(K_{x,x}, K_{y,x})$

Presence of an activator = Absence of an inhibitor = **A resource**

State Graphs

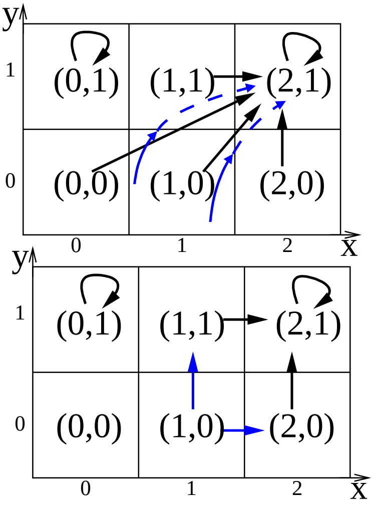
(x,y)	*Focal Point*
(0,0)	$(K_{x,\overline{y}}, K_y)$=(2,1)
(0,1)	(K_x, K_y)=(0,1)
(1,0)	$(K_{x,x\overline{y}}, K_y)$=(2,1)
(1,1)	$(K_{x,x}, K_y)$=(2,1)
(2,0)	$(K_{x,x\overline{y}}, K_{y,x})$=(2,1)
(2,1)	$(K_{x,x}, K_{y,x})$=(2,1)



State Graphs

(x,y)	*Focal Point*
$(0,0)$	$(K_{x,\overline{y}}, K_y)=(2,1)$
$(0,1)$	$(K_x, K_y)=(0,1)$
$(1,0)$	$(K_{x,x\overline{y}}, K_y)=(2,1)$
$(1,1)$	$(K_{x,x}, K_y)=(2,1)$
$(2,0)$	$(K_{x,x\overline{y}}, K_{y,x})=(2,1)$
$(2,1)$	$(K_{x,x}, K_{y,x})=(2,1)$

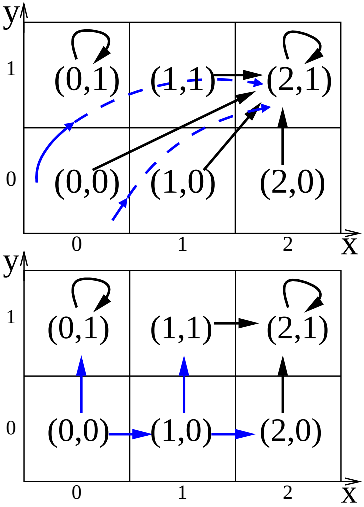
“desynchronization” ⟶



State Graphs

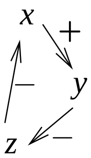
(x,y)	*Focal Point*
(0,0)	$(K_{x,\overline{y}},\, K_y)$=(2,1)
(0,1)	$(K_x,\, K_y)$=(0,1)
(1,0)	$(K_{x,x\overline{y}},\, K_y)$=(2,1)
(1,1)	$(K_{x,x},\, K_y)$=(2,1)
(2,0)	$(K_{x,x\overline{y}},\, K_{y,x})$=(2,1)
(2,1)	$(K_{x,x},\, K_{y,x})$=(2,1)

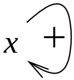
"desynchronization" ⟶
by **units** of Manhattan distance



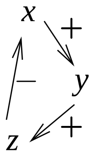
Multistationarity vs. positive cycles

- A cycle in the interaction graph is *positive* if it contains an *even* number of inhibitions
- ***Theorem:*** *if the state graph exhibits several attraction basins then there is at least one positive cycle in the interaction graph*
- Was a conjecture from the 70's to 2004; proved by Adrien Richard (and by Christophe Soulé for the continuous case)

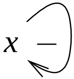




Oscillations vs. negative cycles



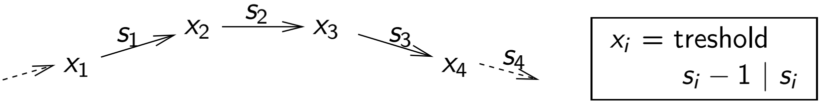
- A cycle in the interaction graph is *negative* if it contains a *odd* number of inhibitions
- ***Thomas conjecture:*** *if the state graph exhibits an homeostasy (stable oscillations) then there is at least one negative cycle in the interaction graph*
- Was a conjecture from the 70's to ≈2010. ***Counter-examples*** have been found (A. Richard, J.-P. Comet, P. Ruet)



Nonetheless it remains a very useful tip in practice when modelling biological examples!

Caracteristic state of a cycle

Helps characterizing the saddle point (resp. center of the oscillations) of the behaviour "driven" by a positive (resp. negative) cycle.



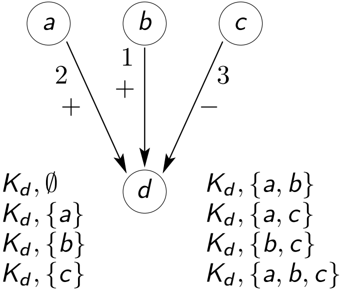
Whatever the sign of $x_i \to x_{i+1}$, for some set of resources ω one should have $K_{x_{i+1},\omega} < s_{i+1} \leqslant K_{x_{i+1},\omega x_i}$, all along the cycle

but it remains a heuristic, at least for negative cycles. . .

Thomas parameters: exponential number

2^i parameters
where i is the in-degree of the gene

$\prod_{genes} (o+1)^{2^i}$ possible parameter values
where o is the out degree of each gene



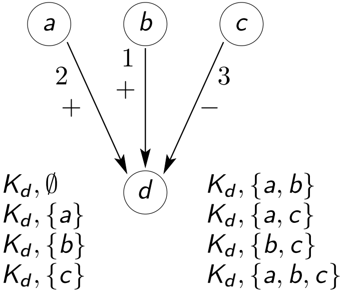
$K_d, \emptyset$
$K_d, \{a\}$
$K_d, \{b\}$
$K_d, \{c\}$

$K_d, \{a, b\}$
$K_d, \{a, c\}$
$K_d, \{b, c\}$
$K_d, \{a, b, c\}$

Yeast≈7000 genes   Human≈25000 genes   Rice≈40000 genes

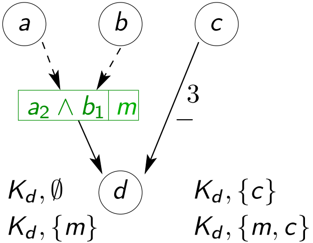
Multiplexes: encode cooperation knowledge

“Proteins of *a* and *b* form a complex before acting on *d*. . .”



$K_d, \emptyset$
$K_d, \{a\}$
$K_d, \{b\}$
$K_d, \{c\}$

$K_d, \{a, b\}$
$K_d, \{a, c\}$
$K_d, \{b, c\}$
$K_d, \{a, b, c\}$



$K_d, \emptyset$
$K_d, \{m\}$

$K_d, \{c\}$
$K_d, \{m, c\}$

multiplex name $= m$

multiplex formula $\equiv a_2 \wedge b_1$

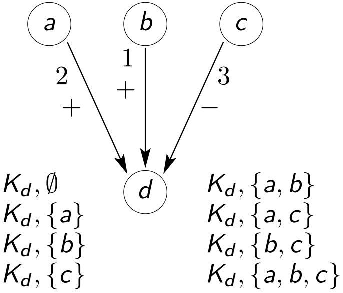
abbreviation:

$v_i \equiv (v \geq i)$

8 → 4 parameters

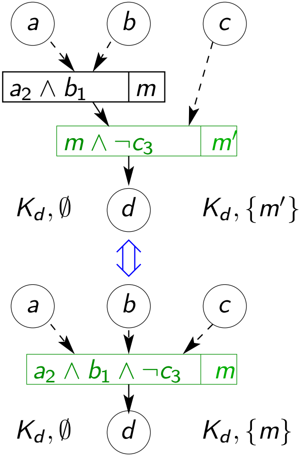
Any propositional formula + remove sign

“. . . and c inhibits d whatever a or b”



$K_d, \emptyset$
$K_d, \{a\}$
$K_d, \{b\}$
$K_d, \{c\}$

$K_d, \{a, b\}$
$K_d, \{a, c\}$
$K_d, \{b, c\}$
$K_d, \{a, b, c\}$



8 ⟶ 2 parameters,

$(\mathbf{o}+\mathbf{1})^{\mathbf{8}} \rightarrow (\mathbf{o}+\mathbf{1})^{\mathbf{2}}$ **parameterizations**

The main problem

Exhaustively identify the sets of (integer) parameters that cope with known behaviours from biological experiments

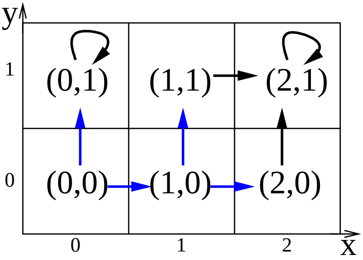
Solution = perform reverse engineering *via* **formal logic**

- 2003: enumeration + CTL + model checking (Bernot,Comet,Pérès,Richard)
- 2005: path derivatives + model checking (Batt,De Jong)
- 2005: PROLOG with constraints (Trilling,Corblin,Fanchon)
- 2007: symbolic execution + LTL (Mateus,Le Gall,Comet)
- 2011: traces + enumeration + CTL + model checking (Siebert,Bockmayr)
- 2015: genetically modified Hoare logic + constraint solving (Bernot,Comet,Roux,Khalis,Richard)

Menu

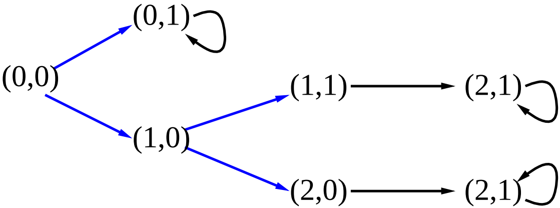
1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Time has a tree structure. . .



As many possible state graphs as possible parameter sets. . . (huge number)

. . . from each initial state:



CTL = Computation Tree Logic

Atoms = comparaisons : (x=2) (y>0) ...

Logical connectives: $(\varphi_1 \wedge \varphi_2)$ $(\varphi_1 \implies \varphi_2)$ $\cdots$

Temporal modalities: made of 2 characters

first character	second character
A = for **A**ll path choices	X = ne**X**t state
	F = for some **F**uture state
E = there **E**xist a choice	G = for all future states (**G**lobally)
	U = **U**ntil

$AX(y = 1)$: the concentration level of y belongs to the interval 1 in all states directly following the considered initial state.

$EG(x = 0)$: there exists at least one path from the considered initial state where x always belongs to its lower interval.

Temporal Connectives of CTL

neXt state:

$EX\varphi$: φ can be satisfied in a next state

$AX\varphi$: φ is always satisfied in the next states

eventually in the Future:

$EF\varphi$: φ can be satisfied in the future

$AF\varphi$: φ will be satisfied at some state in the future

Globally:

$EG\varphi$: φ can be an invariant in the future

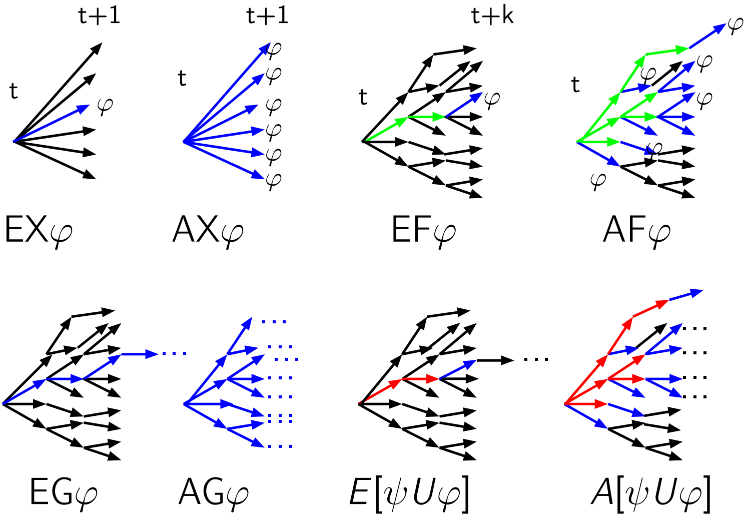
$AG\varphi$: φ is necessarilly an invariant in the future

Until:

$E[\psi U\varphi]$: there exist a path where ψ is satisfied until a state where φ is satisfied

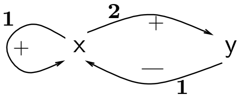
$A[\psi U\varphi]$: ψ is always satisfied until some state where φ is satisfied

Semantics of Temporal Connectives



(*after* → : φ , *after* → : ψ)

CTL to encode Biological Properties



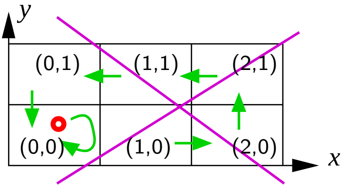
Common properties:

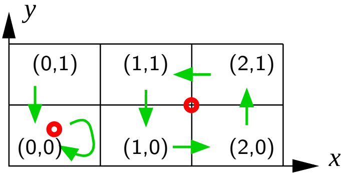
"functionality" of a sub-graph

Special role of "feedback loops"

– positive: *multistationnarity* (even number of —)

– negative: *homeostasy* (odd number of —)





Characteristic properties: $\begin{cases}(x=2)\Longrightarrow AG(\neg(x=0))\\(x=0)\Longrightarrow AG(\neg(x=2))\end{cases}$

They express *"the positive feedback loop is functional"*

(satisfaction of these formulas relies on the parameters $K_{\dots}$)

Model Checking

- Efficiently computes all the states of a state graph which satisfy a given formula: $\{\ \eta\ \mid\ M\ \models_\eta\ \varphi\ \}$.
- Efficiently select the models which globally satisfy a given formula.

Intensively used:

- to find the set of **all** possible discrete parameter values
- to check models under construction w.r.t. **known behaviours** (one often gets an empty set of parameter values!)
- and to prove the **consistency** of a biological **hypothesis**

Model Checking for CTL

Computes all the states of a discrete state graph that satisfy a given formula: $\{ \eta \mid M \models_\eta \varphi \}$.

Idea 1: work on the state graph instead of the path trees.

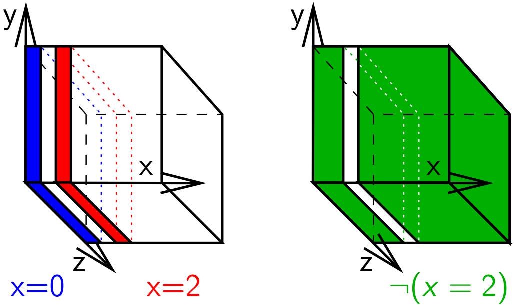
Idea 2: check first the atoms of φ and then check the connectives of φ with a bottom-up computation strategy.

Idea 3: (computational optimization) group some cases together using BDDs (Binary Decision Diagrams).

Example: $(x = 0) \implies AG(\neg(x = 2))$

Obsession: *travel the state graph as less as possible*

$(x = 0) \implies AG(\neg(x = 2))$



and $AG(\neg(x = 2))$?

. . . one should **travel** **<u>all</u>** the paths from any green box and check if successive boxes are green: *too many boxes to visit*.

Trick: $AG(\neg(x = 2))$ is equivalent to $\neg EF(x = 2)$
start from the red boxes and follow the transitions backward.

Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

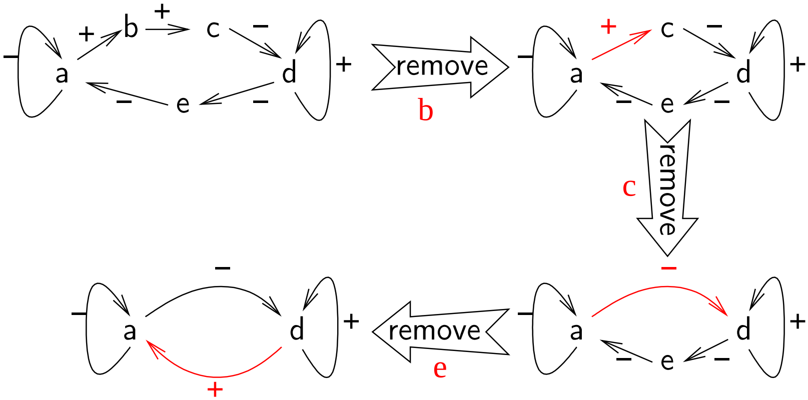
Simplifications driven by the hypothesis

Biologists spend money and time for experiments because they have a **hypothesis** φ in mind that they want to test...

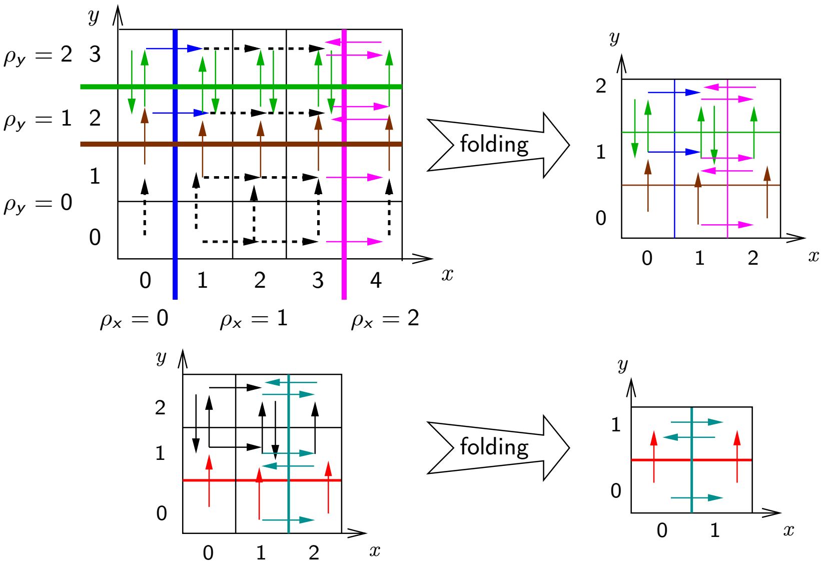
... Successive simplified views of the studied biological object and of the hypothesis:

$$\begin{array}{c}\text{Model}\\ M_1\\ \text{satisfies}\\ \varphi_1\end{array} \Longleftrightarrow \begin{array}{c}\text{Model}\\ M_2\\ \text{satisfies}\\ \varphi_2\end{array} \Longleftrightarrow \begin{array}{c}\text{Model}\\ M_3\\ \text{satisfies}\\ \varphi_3\end{array} \Longleftrightarrow \dots$$

Simplifications *via* gene removing (Naldi)

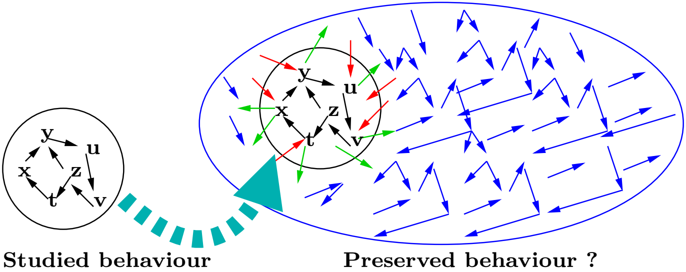


Simplifications *via* level folding



Simplifications *via* subgraphs

Embeddings of Regulatory Networks:



Necessary and sufficient condition on the *local* dynamics of the “input frontier”

. . . *Also* fusion of genes, *etc.*

Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Standard Hoare logic: *swap(x,y)*

```
aux := x  ;
x   := y  ;
y   := aux
```

$\rightarrow$ triple "$\{P\}program\{Q\}$"

precondition P, postcondition Q

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \wedge (y = y_0)\}$

```
aux := x  ;
x   := y  ;
y   := aux
```

$\{(y = x_0) \wedge (x = y_0)\}$

→ "P $\Longrightarrow$ (weakest precondition)" ?

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \land (y = y_0)\}$

```
aux := x  ;
x   := y  ;
y   := aux
```

$\longleftarrow \{(aux = x_0) \land (x = y_0)\}$

$\{(y = x_0) \land (x = y_0)\}$

→ backward proof strategy

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \wedge (y = y_0)\}$

aux := x ;

← $\{(aux = x_0) \wedge (y = y_0)\}$

x := y ;

← $\{(aux = x_0) \wedge (x = y_0)\}$

y := aux

$\{(y = x_0) \wedge (x = y_0)\}$

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \wedge (y = y_0)\}$

$\longleftarrow \{(x = x_0) \wedge (y = y_0)\}$

```
aux := x ;
```

$\longleftarrow \{(aux = x_0) \wedge (y = y_0)\}$

```
x   := y   ;
```

$\longleftarrow \{(aux = x_0) \wedge (x = y_0)\}$

```
y   := aux
```

$\{(y = x_0) \wedge (x = y_0)\}$

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \wedge (y = y_0)\}$

$\{(x = x_0) \wedge (y = y_0)\}$

aux := x ;

$\{(aux = x_0) \wedge (y = y_0)\}$

x := y ;

$\{(aux = x_0) \wedge (x = y_0)\}$

y := aux

$\{(y = x_0) \wedge (x = y_0)\}$

$$\frac{}{\{Q[v \leftarrow expr]\}\ \ v := expr\ \ \{Q\}}\ :=$$

$$\frac{\{P\}p_1\{Q'\} \qquad \{Q'\}p_2\{Q\}}{\{P\}p_1 ; p_2\{Q\}}\ ;$$

Standard Hoare logic: *swap(x,y)*

$\{(x = x_0) \land (y = y_0)\}$

$\longleftarrow \{(x = x_0) \land (y = y_0)\}$

aux := x ;

$\longleftarrow \{(aux = x_0) \land (y = y_0)\}$

x := y ;

$\longleftarrow \{(aux = x_0) \land (x = y_0)\}$

y := aux

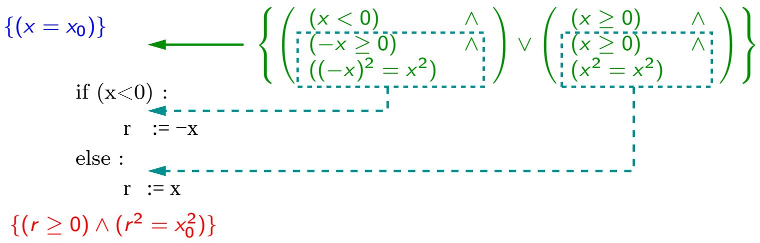
$\{(y = x_0) \land (x = y_0)\}$

$$\frac{}{\{Q[v \leftarrow expr]\} \quad v := expr \quad \{Q\}} \; :=$$

$$\frac{\{P\}p_1\{Q'\} \qquad \{Q'\}p_2\{Q\}}{\{P\}p_1 ; p_2\{Q\}} \; ;$$

$$\frac{\dfrac{\dfrac{}{\{Q_3\}a_1\{Q_2\}} := \qquad \dfrac{}{\{Q_2\}a_2\{Q_1\}} :=}{\{P\}a_1 ; a_2\{Q_1\}} ; \qquad \dfrac{}{\{Q_1\}a_3\{Q\}} :=}{\{P\}a_1 ; a_2 ; a_3\{Q\}} \; ;$$

Standard Hoare logic: *abs(x)*



$$\{(x = x_0)\} \longleftarrow \left\{ \begin{pmatrix} (x<0) & \wedge \\ (-x \geq 0) & \wedge \\ ((-x)^2 = x^2) & \end{pmatrix} \vee \begin{pmatrix} (x \geq 0) & \wedge \\ (x \geq 0) & \wedge \\ (x^2 = x^2) & \end{pmatrix} \right\}$$

```
if (x<0) :
        r   := −x
else :
        r  := x
```

$$\{(r \geq 0) \wedge (r^2 = x_0^2)\}$$

$$\frac{\{Q_1\}p_1\{Q\} \qquad\qquad \{Q_2\}p_2\{Q\}}{\{(e \wedge Q_1) \vee (\neg e \wedge Q_2)\}\ \texttt{if}\ e\ \texttt{then}\ p_1\ \texttt{else}\ p_2\ \{Q\}}\ \textbf{if}$$

Also:

While loop: $$\frac{\{e{\wedge}I\}p\{I\} \qquad (\neg e{\wedge}I){\Longrightarrow}Q}{\{I\}\mathit{while}\ e\ \mathit{with}\ I\ \mathit{do}\ p\{Q\}}$$

Empty program: $$\frac{P{\Longrightarrow}Q}{\{P\}\varepsilon\{Q\}}$$ use sparingly: loses *weakest* precondition!

Assertion language (Pre/Post)

Terms: v gene | $n \in \mathbb{N}$ | $K_{v,\{\dots\}}$ parameter symbols | $+$ | $-$

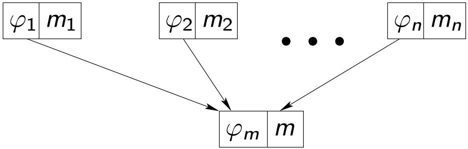
atoms: $t \geqslant t'$ | $t < t'$ | $t = t'$ | $\dots$

Connectives: $\neg \mid \wedge \mid \vee \mid \Longrightarrow$

Example:

$$(a \leqslant 3 \;\wedge\; d+1 < K_{d,\{m,c\}}) \;\vee\; (K_{d,\{c\}} < K_{d,\{m,c\}} \;\wedge\; c \geqslant 3)$$

From multiplexes to assertions: flatening



$\overline{\varphi_m} \;\equiv\; \varphi_m[m_i \leftarrow \varphi_i]$ for all i and recursively

Assertions that formalize Thomas'framework

ω is the set of ressources of v:

$$\Phi_v^{\omega} \quad \equiv \quad \Big(\bigwedge_{m \,\in\, \omega} \overline{\varphi_m} \Big) \wedge \Big(\bigwedge_{m \,\in\, G^{-1}(v) \setminus \omega} \neg \overline{\varphi_m} \Big)$$

v can increase:

$$\Phi_v^{+} \quad \equiv \quad \bigwedge_{\omega \subset G^{-1}(v)} (\Phi_v^{\omega} \Longrightarrow K_{v,\omega} > v)$$

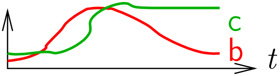
v can decrease:

$$\Phi_v^{-} \quad \equiv \quad \bigwedge_{\omega \subset G^{-1}(v)} (\Phi_v^{\omega} \Longrightarrow K_{v,\omega} < v)$$

Trace specifications

- $x+ \mid x- \mid x := n \mid \mathit{assert}(\varphi)$
- $p_1 ; p_2 ; \cdots ; p_n$
- $\mathit{if}\ \varphi\ \mathit{then}\ p_1\ \mathit{else}\ p_2$
- $\mathit{while}\ \varphi\ \mathit{with}\ \psi\ \mathit{do}\ p$
- $\forall(p_1, p_2, \cdots, p_n)$
- $\exists(p_1, p_2, \cdots, p_n)$

Examples:



- $b+; c+; b-$
- $\exists(b+, b-, c+, c-, \varepsilon)$
- $\mathit{while}\ (b < 2)\ \mathit{with}\ (c > 0)$
  $\mathit{do}\ \exists(b+, b-, \forall((c-; a-), c+))\ \mathit{od};$
  $b-$

Genetic, a la Hoare, inference rules

Incrementation rule: $\dfrac{}{\{\ \Phi_v^+\ \wedge\ Q[v\leftarrow v+1]\ \}\ v+\ \{Q\}}$

Decrementation rule: $\dfrac{}{\{\ \Phi_v^-\ \wedge\ Q[v\leftarrow v-1]\ \}\ v-\ \{Q\}}$

Assertion rule: $\dfrac{}{\{\ \varphi\wedge Q\ \}\ \mathit{assert}(\varphi)\ \{Q\}}$

Universal quantifier rule: $\dfrac{\{P_1\}p_1\{Q\} \qquad \{P_2\}p_2\{Q\}}{\{P_1\wedge P_2\}\ \forall(p_1,p_2)\ \{Q\}}$

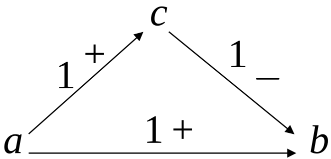
Existential quantifier rule: $\dfrac{\{P_1\}p_1\{Q\} \qquad \{P_2\}p_2\{Q\}}{\{P_1\vee P_2\}\ \exists(p_1,p_2)\ \{Q\}}$

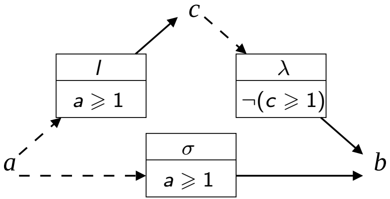
Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Example: Feedforward “loop”

Uri Alon most frequent gene network patterns





Behaviour of b after switching a from off to on ?

Simple off→on→off behaviour of b with the help of c:

$$\{(a = 1 \ \wedge \ b = 0 \ \wedge \ c = 0)\} \ b+ \ ; \ c+ \ ; \ b- \ \ \{b = 0\}$$

possible if and only if: $K_{b,\{\sigma,\lambda\}} = 1 \ \wedge \ K_{c,\{l\}} = 1 \ \wedge \ K_{b,\{\sigma\}} = 0$

Feedforward example (continued)

off→on→off behaviour of b without the help of c:

$$\{(a = 1 \ \wedge \ b = 0 \ \wedge \ c = 0)\}\ b+ \ ; \ b- \ \ \{b = 0\}$$

$$\left\{\begin{array}{l} b = 0 \\ ((c \geqslant 1) \wedge (a < 1)) \Longrightarrow ((K_b = 1) \wedge (K_b = 0)) \\ ((c \geqslant 1) \wedge (a \geqslant 1)) \Longrightarrow ((K_{b,\sigma} = 1) \wedge (K_{b,\sigma} = 0)) \\ ((c < 1) \wedge (a < 1)) \Longrightarrow ((K_{b,\lambda} = 1) \wedge (K_{b,\lambda} = 0)) \\ ((c < 1) \wedge (a \geqslant 1)) \Longrightarrow ((K_{b,\sigma\lambda} = 1) \wedge (K_{b,\sigma\lambda} = 0)) \end{array}\right\} \textbf{not satisfiable!}$$

Feedforward example (continued)

Although $b+; c+; b-$ is possible, if c becomes “on” <u>before</u> b, then b will never be able to get “on”

Proof by refutation:

$$\left\{ \begin{array}{l} a = 1 \ \wedge \ b = 0 \ \wedge \ c = 1 \ \wedge \\ K_{b,\sigma\lambda} = 1 \ \wedge \ K_{c,l} = 1 \ \wedge \ K_{b,\sigma} = 0 \end{array} \right\}$$

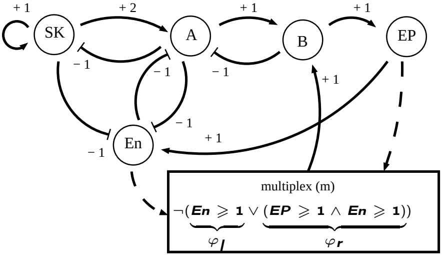
$$\mathit{while}\ b < 1\ \mathit{with}\ I\ \mathit{do}\ \exists(b+, b-, c+, c-)$$

$$\{\ b = 1\ \}$$

the triple is inconsistent, whatever the loop invariant I !

Cell cycle in mammals

- A 22 gene model reduced to 5 variables using multiplexes


+ 1
SK
+ 2
A
+ 1
B
+ 1
EP
− 1
− 1
− 1
+ 1
− 1
+ 1
En
− 1
multiplex (m)
$\neg(\underbrace{\boldsymbol{En} \geqslant \mathbf{1}}_{\varphi_l} \vee \underbrace{(\boldsymbol{EP} \geqslant \mathbf{1} \wedge \boldsymbol{En} \geqslant \mathbf{1})}_{\varphi_r})$

SK = Cyclin E/Cdk2, Cyclin H/Cdk7
A = Cyclin A/Cdk1
B = Cyclin B/Cdk1
En = APC^{G1}, CKI (p21, p27), Wee1

EP = APC^{M}, Phosphatases

- 48 states, 26 parameters, 339 738 624 possible valuations, 12 trace specifications and a few temporal properties

Cell cycle in mammals (continued)

- 13 parameters have been entirely identified (50%) and only 8192 valuations remain possible according to the generated constraints (0.002%)
- Additional reachability constraints (e.g. endoreplication and quiescent phase) have been necessary, on an extended *hybrid* extension of the Thomas' framework, to identify (almost) all parameters
- This initial Hoare logic identification step was crucial: it gave us the sign of the derivatives in all the (reachable) states

Correctness, Completeness and Decidability

- If there is a proof tree for $\{P\}p\{Q\}$ then for each initial state satisfying P, there are traces in the gene network that realize the trace specification p, and for all of them, if terminating, they satisfy Q at the end.
- If for each initial state satisfying P there are traces that realize p in the gene network and if they all satisfy Q at the end, then there exists a proof tree for $\{P\}p\{Q\}$.
- There is a simple algorithm to compute, for each Q, the minimal loop invariant I such that $\{I\}$*while e with I do* $p\{Q\}$. (However well chosen slightly non minimal invariants can considerably simplify the proof tree. . .)

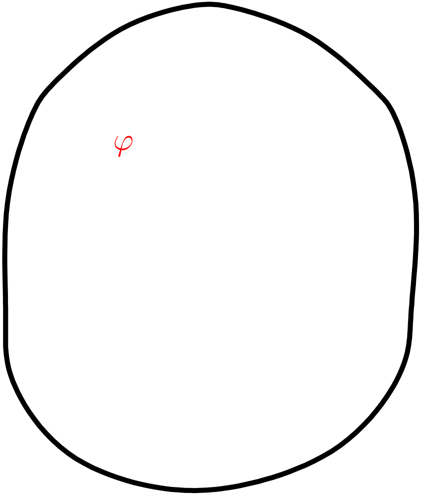
Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Generation of biological experiments (1)

Set of all the formulas:

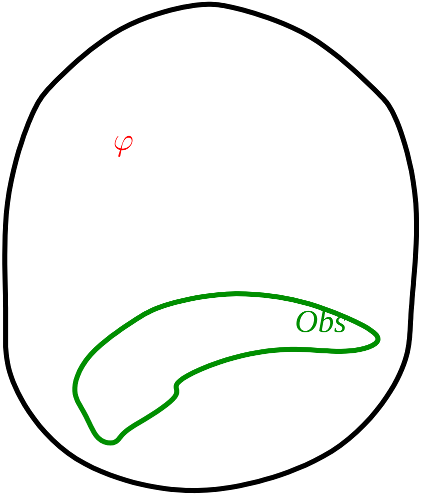
φ = hypothesis



Generation of biological experiments (2)

Set of all the formulas:

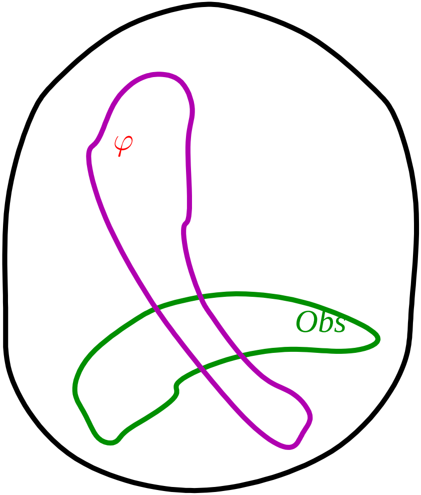
φ = hypothesis
Obs = possible experiments



Generation of biological experiments (3)

Set of all the formulas:

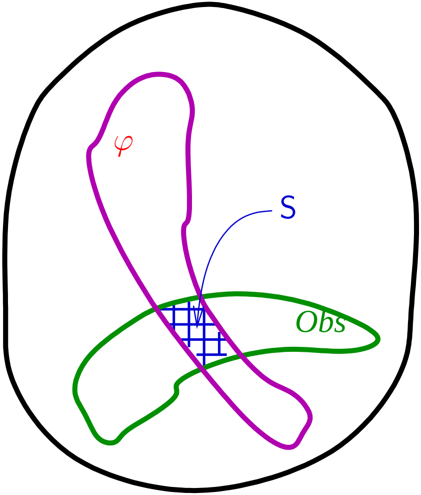
φ = hypothesis
Obs = possible experiments
$Th(\varphi) = \varphi$ inferences



Generation of biological experiments (4)

Set of all the formulas:

φ = hypothesis
Obs = possible experiments
$Th(\varphi)$ = φ inferences
S = sensible experiments



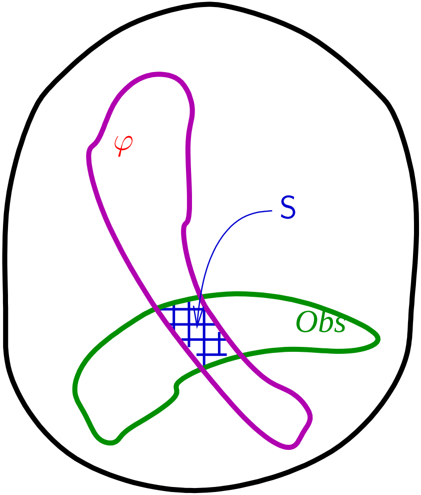
Generation of biological experiments (5)

Set of all the formulas:

φ = hypothesis
Obs = possible experiments
$Th(\varphi)$ = φ inferences
S = sensible experiments

Refutability:

$$\mathsf{S} \Longrightarrow \varphi \ ?$$



Generation of biological experiments

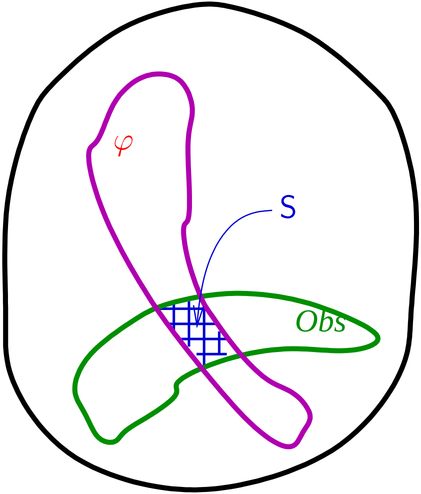
Set of all the formulas:

φ = hypothesis
Obs = possible experiments
$Th(\varphi)$ = φ inferences
S = sensible experiments

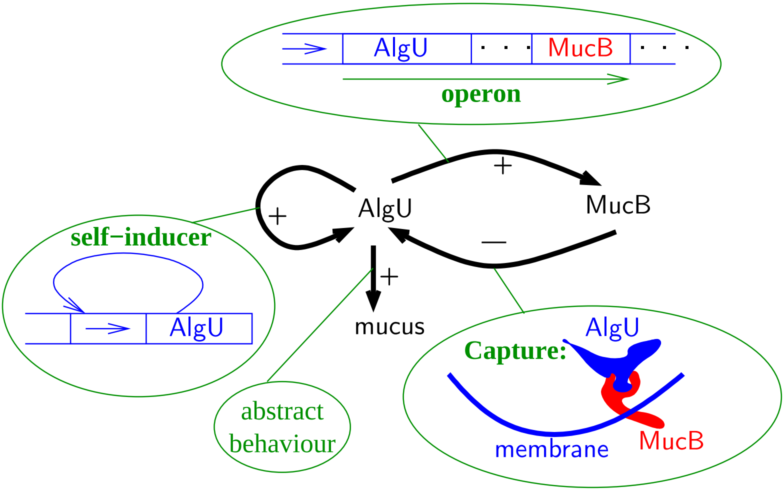
Refutability:

$$S \Longrightarrow \varphi \ ?$$

Best refutations:
Choice of experiments in S ?
. . . optimisations

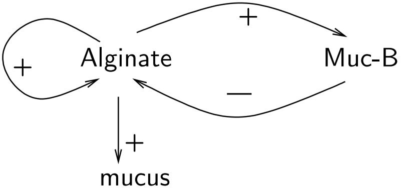


Example: Mucus Production in *P. aeruginosa*



How to validate a multistationnarity

$\mathcal{M}$: *(unknown thresholds)*



$$\Phi : \begin{cases} (Alginate = 2) \Longrightarrow AG(Alginate = 2) & (hypothesis) \\ (Alginate = 0) \Longrightarrow AG(Alginate < 2) & (knowledge) \end{cases}$$

Assume that only *mucus* can be observed:

Lemma: $AG(Alginate = 2) \Longleftrightarrow AFAG(mucus = 1)$

(... formal proof by computer ...)

→ $\boxed{\text{To validate: } (Alginate = 2) \Longrightarrow AFAG(mucus = 1)}$

$(Alginate = 2) \Longrightarrow AFAG(mucus = 1)$

Karl Popper:
to validate = to try to refute
thus A=false is useless
experiments must begin with a pulse

$A \Longrightarrow B$	*true*	*false*
true	true	false
false	true	true

The pulse forces the bacteria to reach the initial state $Alginate = 2$.
If the state is not directly controlable we need to prove lemmas:

$$(something\ reachable) \Longrightarrow (Alginate = 2)$$

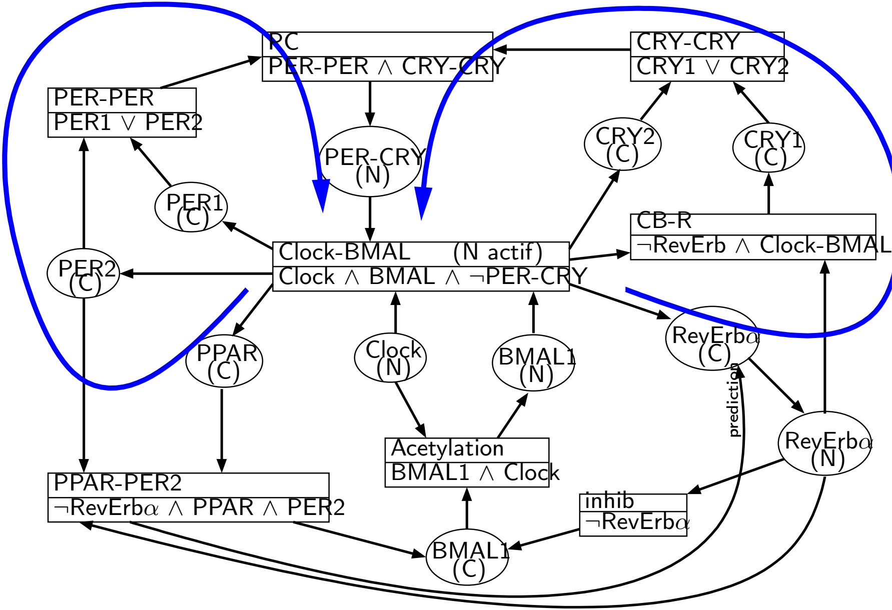
General form of a test:

$$(something\ \underline{reachable}) \Longrightarrow (something\ \underline{observable})$$

Menu

1. Formal logic and dynamic models for biology
2. Discrete models for gene networks according to René Thomas
3. Gene networks and temporal logic
4. Models as mediums for checking biological hypotheses
5. Genetically modified Hoare logic, and examples
6. Extracting interesting experiments from models
7. Complex vs. complicated. . .

Circadian clock interaction graph



The target question

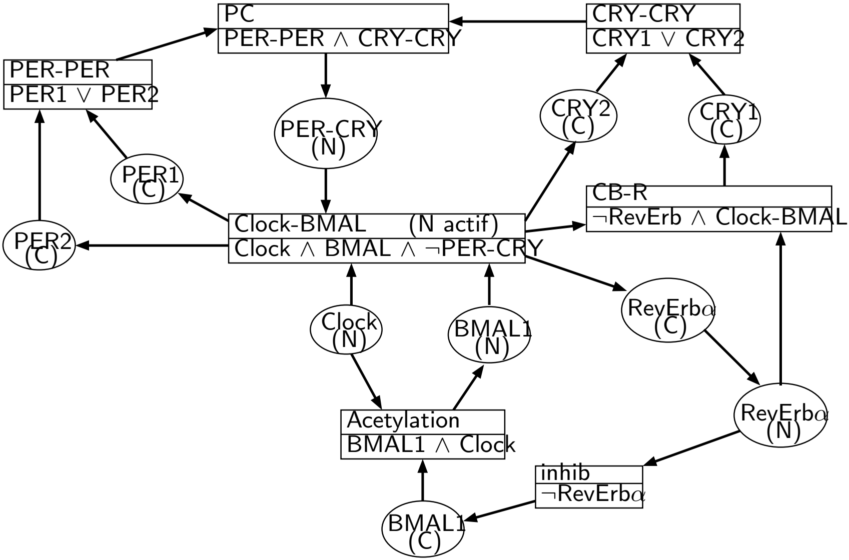
Impact of the day length on the persistence of the circadian circle ?

$\Longrightarrow$ framework with time delays:

- mainly replace the integer $K_{x,\omega}$ by real numbers $C_{x,\omega,n}$, called *celerities*, where n is the current state of x
- notice that $C_{x,\omega,n} > 0$ if $K_{x,\omega} > n$ and a few other logical properties
- extension of temporal logic with delays: $AF_{[t_1,t_2]}$ and so on

Decidability is lost but the identification of parameters remains "almost" automatic with such constant speeds $C_{x,\omega,n}$ (constraint solving on intervals)

Fold levels and remove PPAR



PC
PER-PER ∧ CRY-CRY

CRY-CRY
CRY1 ∨ CRY2

PER-PER
PER1 ∨ PER2

PER-CRY
(N)

CRY2
(C)

CRY1
(C)

PER1
(C)

CB-R
¬RevErb ∧ Clock-BMAL

PER2
(C)

Clock-BMAL (N actif)
Clock ∧ BMAL ∧ ¬PER-CRY

Clock
(N)

BMAL1
(N)

RevErbα
(C)

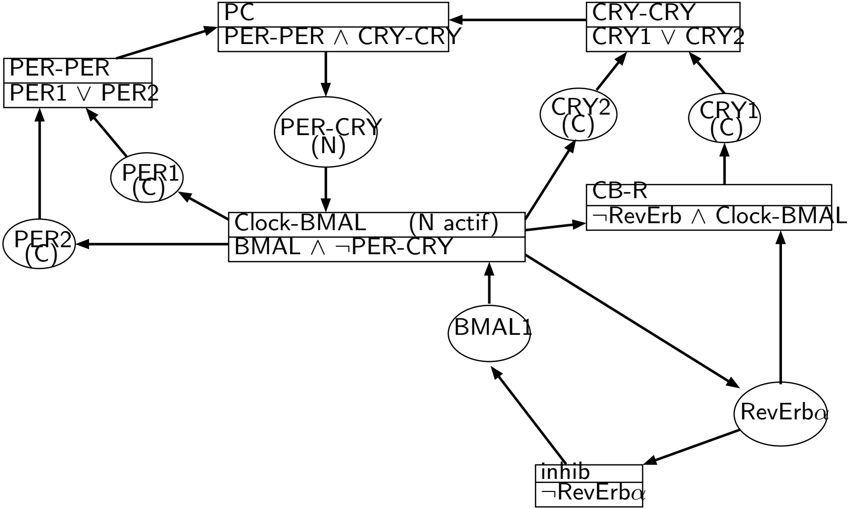
Acetylation
BMAL1 ∧ Clock

RevErbα
(N)

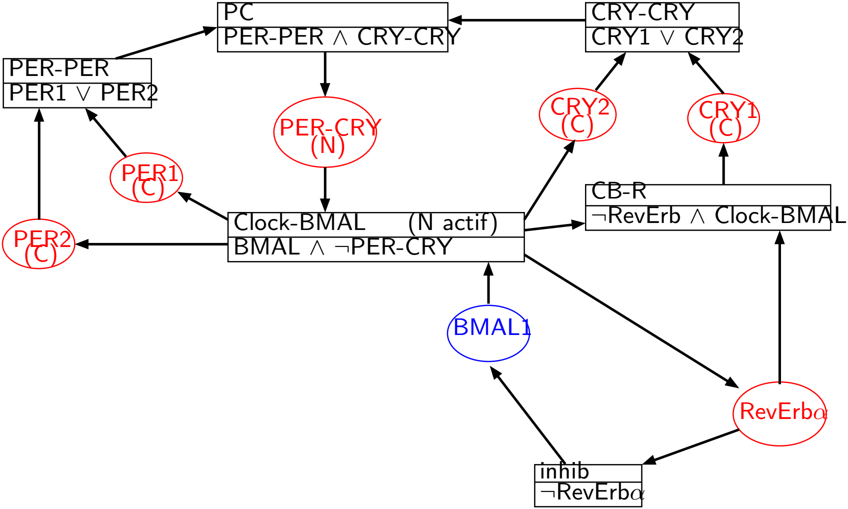
inhib
¬RevErbα

BMAL1
(C)

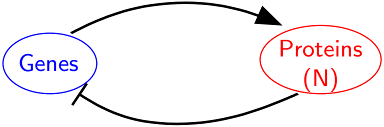
Remove Clock and “tunnel” pathways



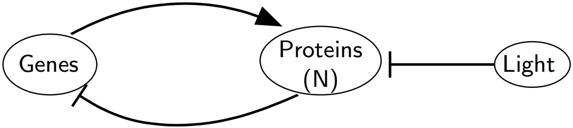
Separate inhibitors/activators of Clock-BMAL



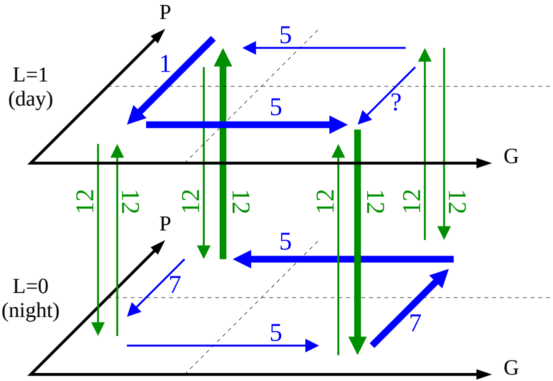
Fusion of all inhibitors



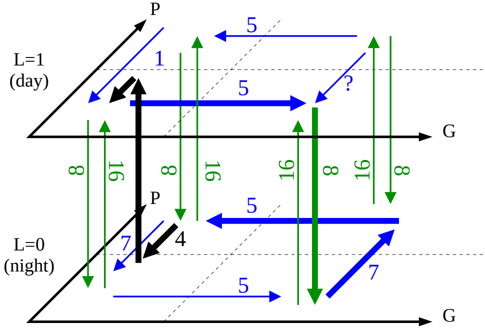
and Light prevents PER-CRY to enter the nucleus:



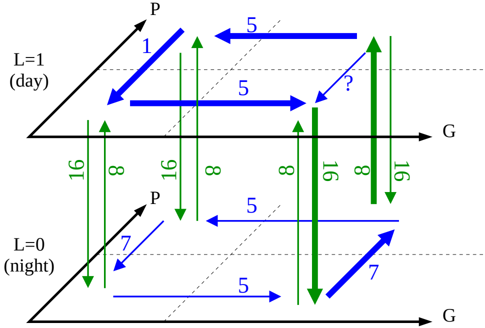
12 hours model



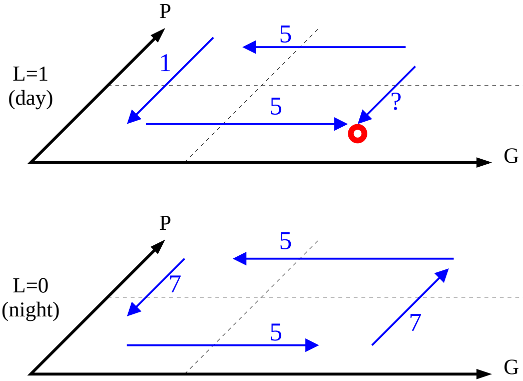
Winter model



Summer model



Jet lag + training



Take Home Messages

Make explicit the hypotheses that motivate the biologist

A far as possible formalize them to get a computer aided approach

Behavioural *properties* are as much important as *models*

Mathematical models are not reality: let's use this freedom !
(several views of a same biological object)

Modelling is significant only with respect to the considered experimental *reachability* and *observability* (for refutability)

Formal proofs can suggest wet experiments

"Kleenex" models help understanding main behaviours

Specialized qualitative approaches can make complex models simple