Chemical Reaction Networks as a Modeling and Programming Language

François Fages

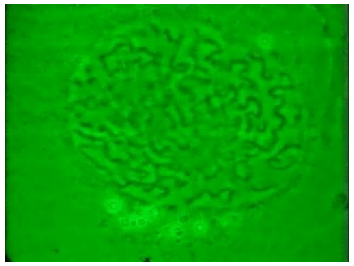
Lifeware group http://lifeware.inria.fr/
on Computational Systems Biology and Optimization

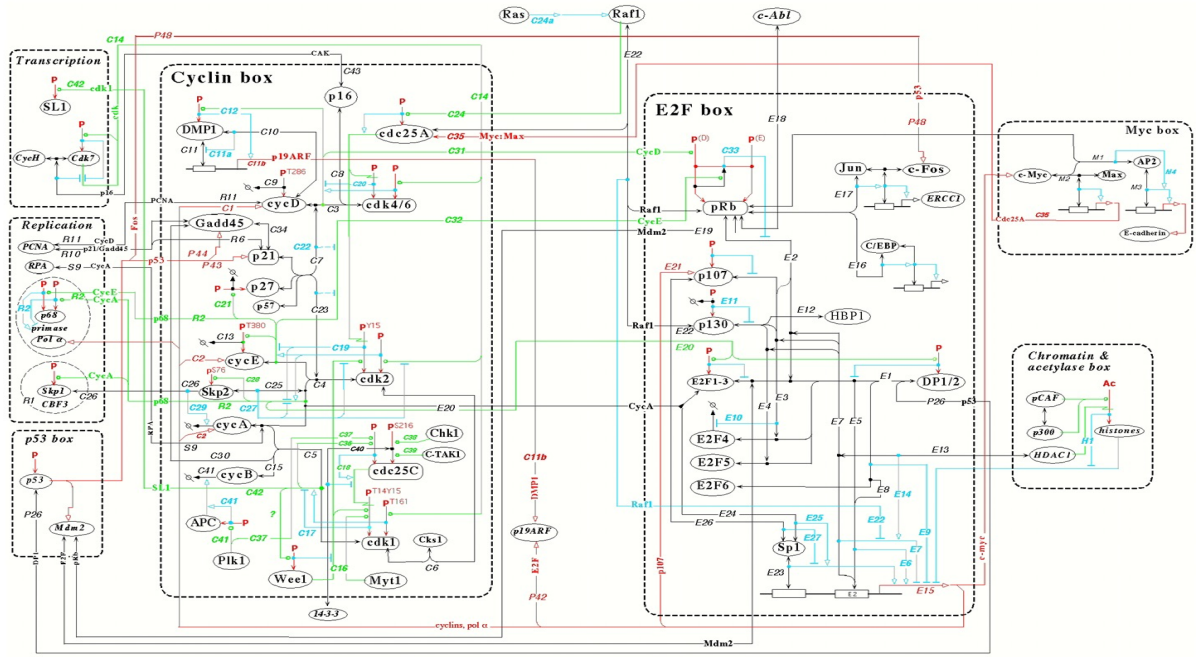
National Institute of Research in Computer Science and Control

Inria Saclay, Palaiseau, France

Lifeware's Motto: Cells Compute

- Cells process information from external signals
  - noise filtering, ultrasensitivity
  - analog-digital conversion
- Make informed decisions
  - metabolism change
  - cell division
  - differentiation
  - migration
  - apoptosis
- Control process execution
  - cell cycle progression, DNA repair
  - homeostasis





What are the programs ?

Chemical Reaction Networks (CRN)

Systems Biology

"Systems Biology aims at systems-level understanding which requires a set of principles and methodologies that links the behaviors of molecules to systems characteristics and functions." Hiroaki Kitano, ICSB 2000

After the end of Human Genome Project (2000)

1. Analyze post-genomic data: RNA, proteins

Data produced with high-throughput technologies

→ Databases GO, KEGG, BioCyc, etc.

Two meanings of "Bioinformatics" :

1. Tools for processing biological data: sequence analysis, phylogenetic trees, structure prediction
2. Science of information processing in cells
   CRN natural program analysis

2. Understand and predict cell processes with protein/RNA/gene networks

→ Modelling & analysis software (CellDesigner, Cytoscape, Copasi, Biocham, Kappa…)

→ Model exchange format for CRN models: Systems Biology Markup Language (SBML)

→ Model repositories: e.g. `biomodels.net` 2000 hand-made models 10000 imported from metabolic maps

→ Simulation of a whole-cell *mycoplasma genitalium* [Karr Covert et al 12]

"Bioinformatics is the study of informatic processes in biotic systems" Ben Hessper, Paulien Hodgeweg 1970

Synthetic Biology

Design and implement new functions in either living cells or artificial devices

Synthetic gene networks added to living cells

- MIT BioBrick standard biological part, IGEM competition since 2004
- Production of an antimalarial drug in engineered yeast (UC Berkeley, SANOFI)
- Biofuels (e.g. from engineered algae)

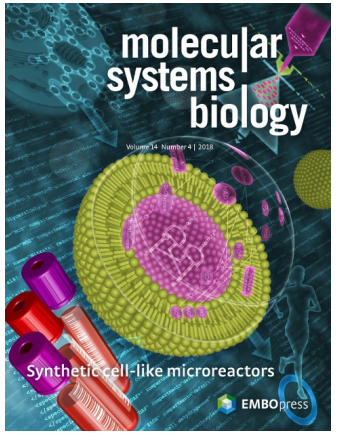
CRN program synthesis
Compilers of high-level functions:
BIOCHAM [F- Le Guludec Bournez Pouly *CMSB* 2017]
CRN++ [Varsik Soloveichik Kurshid *NatComp* 2020]

DNA computing

- Artificial double strand DNA [Phillips Cardelli 2009]
- Turing complete DNA stack programming [Cook Soloveichik Winfree Bruck 2009]

Protein computing

- Cell signal processing, process control [Oishi Klavins 2011, Briat Gupta Khammash 2016]
- Turing complete CRN analog computation [F- Le Guludec Bournez Pouly 2017 *next lecture*]
- Artificial DNA-free micro-reactor diagnosis vesicles [Courbet Amar F- Renard Molina 2018]



Analog Computation with CRN Programs

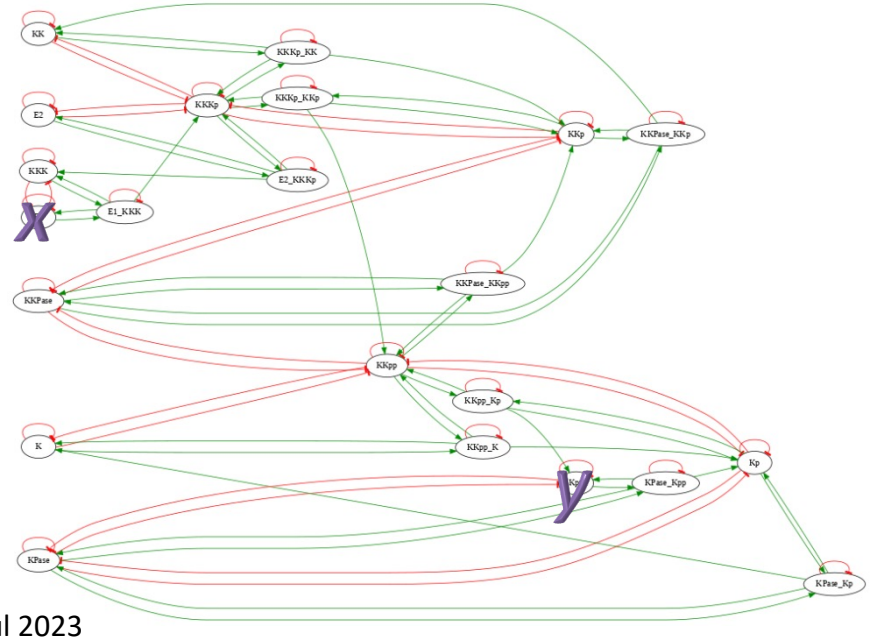
Theorem (Turing-completeness of finite CRNs with ODE semantics) [F- Le Guludec Bournez Pouly CMSB 2017]

Any computable real function (i.e. by a Turing machine with arbitrary requested precision given in input) can be computed by a finite CRN with mass action law kinetics and at most bimolecular reactions.

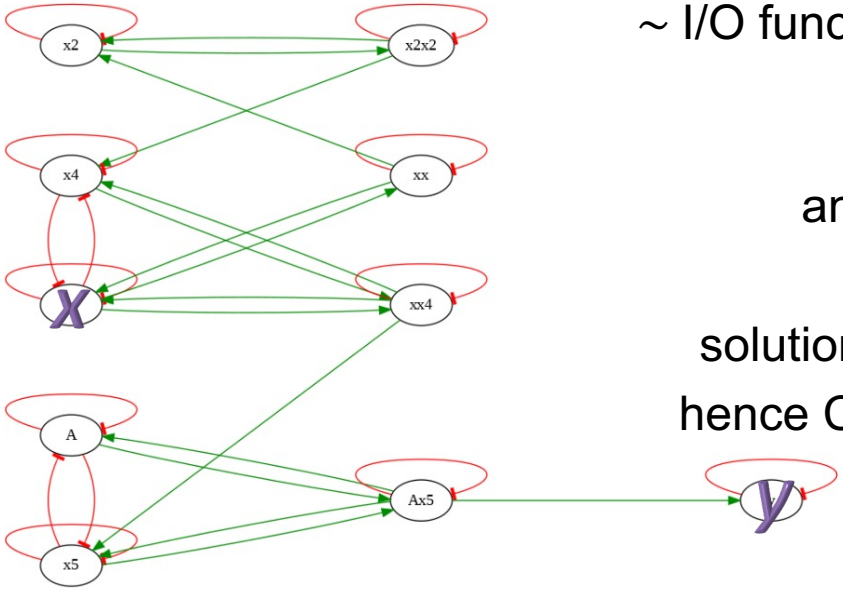
Theorem (Online computation, robust stabilization) [Hemery F- CMSB 2022]

The set of real functions computable online by a CRN is the set of real algebraic functions $P(x, f(x)) = 0$.

Example Natural MAPK CRN structure



Biocham-compiler-generated CRN for I/O function $y = \frac{x^5}{k+x^5}$



~ I/O function of MAPK signaling

[Huang Ferrel PNAS 1996]

stiff sigmoid

$\text{analog}_x\text{-digital}_y$ converter

solution of $(k + x^5)y - x^5 = 0$

hence CRN-computable online

Plan of my Two Lectures

Lecture 1: CRN as a Modelling/Programming Language

1. CRN syntax, hierarchy of semantics and typings by abstract interpretation
2. Model reductions: slow-fast ODE-based decomposition and general graph-theoretic CRN reductions



3. Specifying behaviors in (quantitative) temporal logics: verification, robustness measure, parameter search
4. Case study on coupled modeling of the cell cycle and circadian clock
5. Conclusion on high-level rule-based CRN models versus low-level ODE models

Lecture 2: The cell, an analog chemical computer

1. Turing completeness of continuous CRN over a finite set of abstract molecular species
2. Compiler of mathematical functions in abstract CRNs
3. Comparisons to natural CRNs acquired by natural evolution and artificial evolution
4. Conclusion on engineered design by decomposition and learnt design by evolution

1. CRN Syntax

A CRN is finite set of formal chemical reactions.

A reaction is a triplet (R, P, f) also noted $R \rightarrow_f P$ where

- R is a multiset of reactant species, written with stoichiometric coefficients $\alpha_1 R_1 + \cdots + \alpha_r R_r$
- P is a multiset of product species, written with stoichiometric coefficients $\beta_1 P_1 + \cdots + \beta_r P_r$
- A catalyst is a species that appears as both reactant and product with the same coefficient.
- f is a formal rate function with well-formedness conditions that should be imposed in SBML:

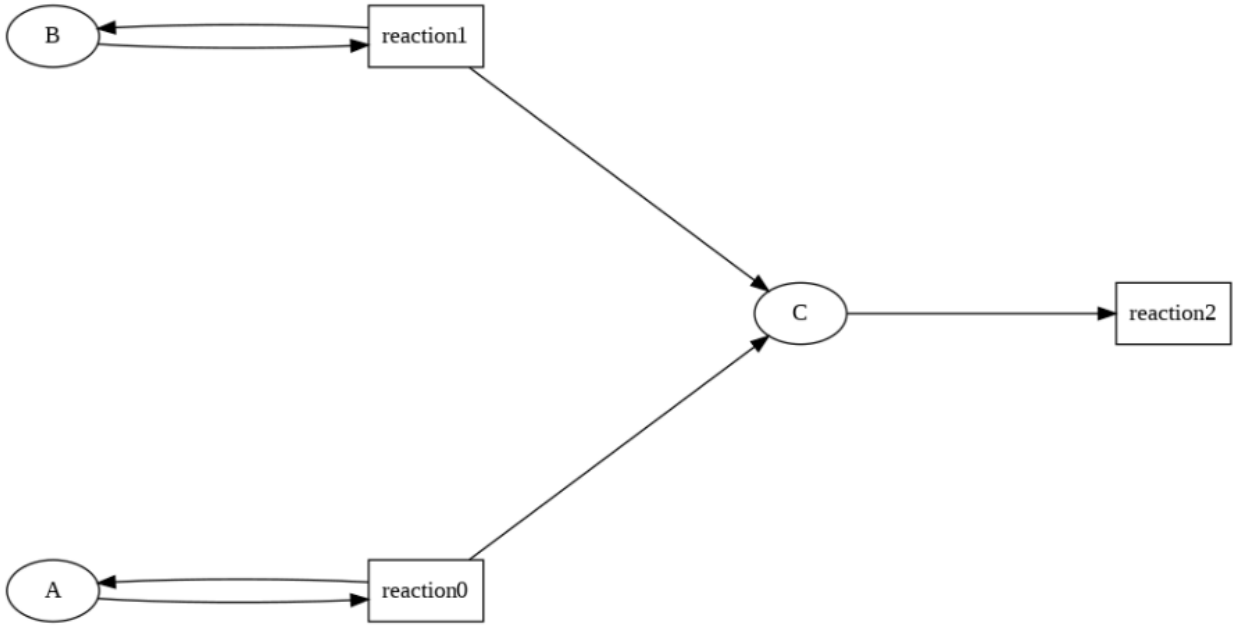
[F- Gay Soliman. Inferring Reaction Systems from Ordinary Differential Equations. *TCS* 2015]

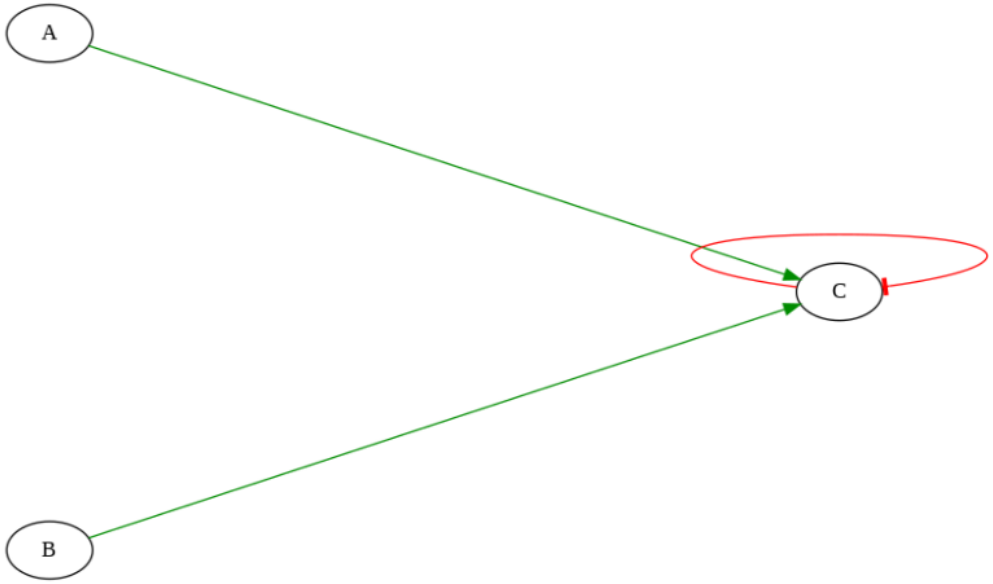
  - $S \in R$ if and only if $\frac{\partial f}{\partial S} \neq 0$ (negative for an inhibitor reactant)
  - $S \in R$, $\frac{\partial f}{\partial M} > 0$ imply $f([S] = 0) = 0$ (ensures positivity)

E.g. Mass action law kinetic function $f = k\, R_1^{\alpha_1} \ldots R_r^{\alpha_r}$, Michaelis-Menten kinetics $\frac{vS}{k+S}$, Hill kinetics $\frac{vS^n}{k+S^n}$

Graphical Representations of CRN

Reaction hypergraph: bipartite graph of species and reactions (Petri net structure, SBGN compatible)



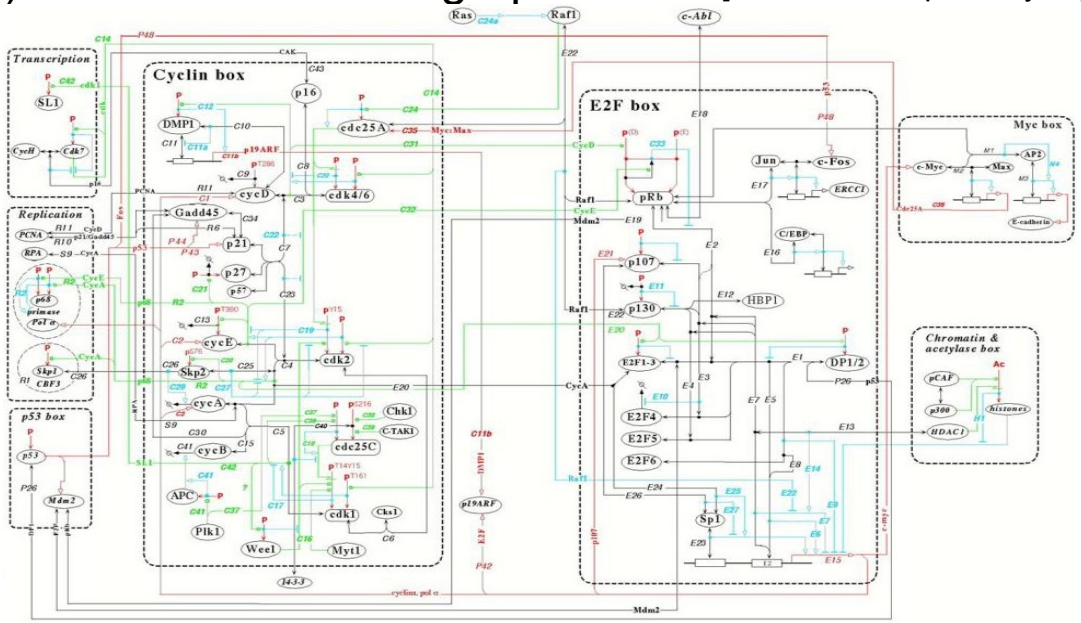


Influence graph: abstraction, graph labeled by influence signs

Model Building: Two Contradictory Perspectives

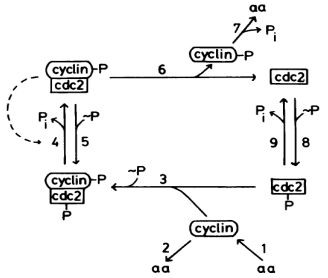
1) Model for representing knowledge: the more detailed the better (do not miss any known information)
2) Model for answering a concrete question: the more abstract the better (get rid of irrelevant information)

1) Generic annotated graph model [Kohn's 99 map cell cycle]



2) Reduced graph models

[Tyson's 91 model cell cycle]



2) Reduced boolean models (and's or or's?)

2) Reduced Petri net models (discrete levels)

2) Reduced stochastic CTMC models (intrinsic noise, time)

2) Reduced ODE models (timing)

[Tyson's 91 model cell cycle]

$$
\begin{aligned}
d[C2]/dt &= k_6[M] - k_8[\sim P][C2] + k_9[CP] \\
d[CP]/dt &= -k_3[CP][Y] + k_8[\sim P][C2] - k_9[CP] \\
d[pM]/dt &= k_3[CP][Y] - [pM]F([M]) + k_5[\sim P][M] \\
d[M]/dt &= [pM]F([M]) - k_5[\sim P][M] - k_6[M] \\
d[Y]/dt &= k_1[aa] - k_2[Y] - k_3[CP][Y] \\
d[YP]/dt &= k_6[M] - k_7[YP]
\end{aligned}
$$

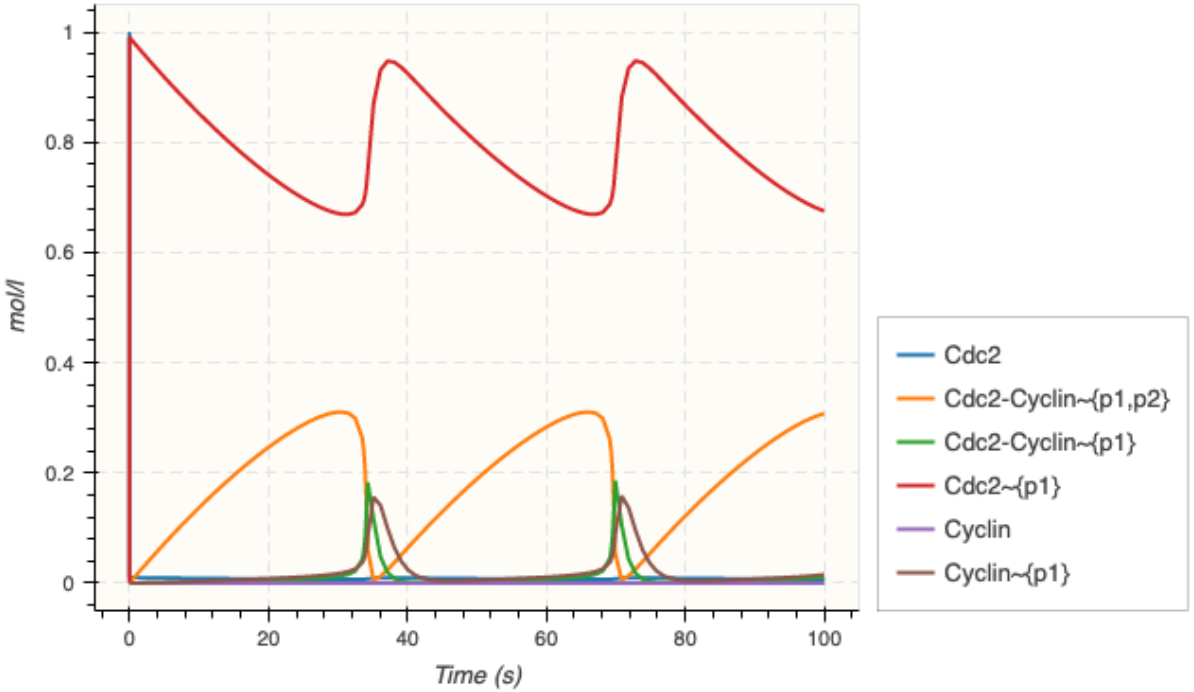
Requires general notions of CRN structure reductions and CRN dynamics abstractions

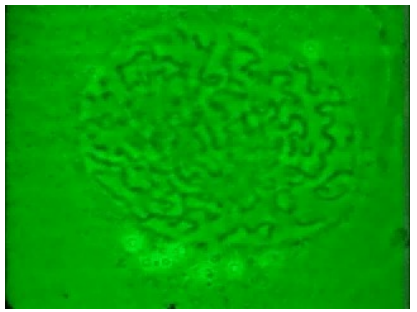
CRN Semantics $A + B \xrightarrow{k.A.B} C$

Differential semantics: concentrations, continuous time evolution

Ordinary differential equations (ODE) $\frac{dA}{dt} = -k.A.B \quad \frac{dB}{dt} = -k.A.B \quad \frac{dC}{dt} = k.A.B$

ODE simulation of Tyson's 1991 model of the cell cycle:





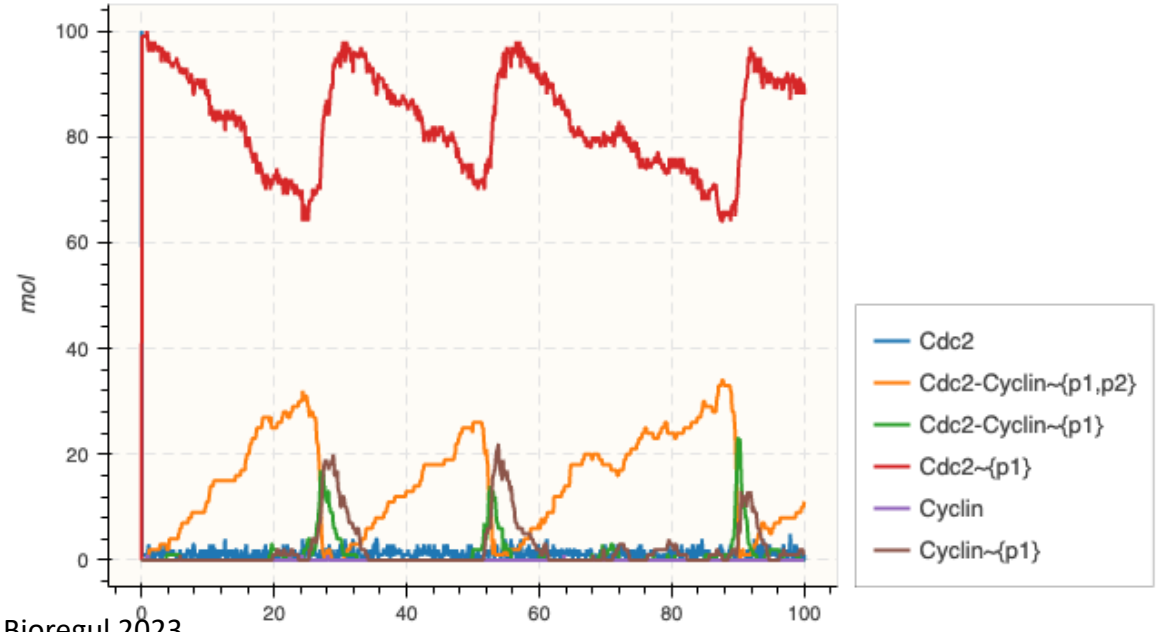
CRN Semantics $A + B \xrightarrow{k.A.B} C$

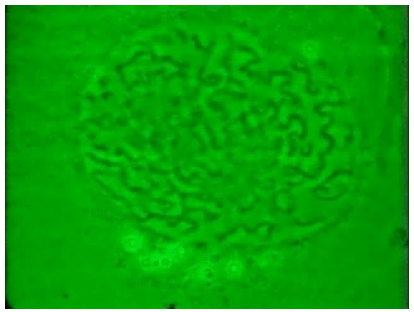
Differential semantics: concentrations, continuous time evolution

Ordinary differential equations (ODE) $\frac{dA}{dt} = -k.A.B \quad \frac{dB}{dt} = -k.A.B \quad \frac{dC}{dt} = k.A.B$

Stochastic semantics: numbers of molecules, probability and time of transition (intrinsic noise)

Continuous Time Markov Chain (CTMC) $A , B \xrightarrow{p(S_i),\, t(Si)} C{+}{+}, A{-}{-}, B{-}{-}$





CRN Semantics $A + B \xrightarrow{k.A.B} C$

Differential semantics: concentrations, continuous time evolution

Ordinary differential equations (ODE) $\frac{dA}{dt} = -k.A.B$ $\frac{dB}{dt} = -k.A.B$ $\frac{dC}{dt} = k.A.B$

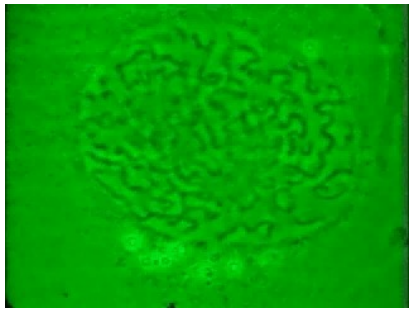
Stochastic semantics: numbers of molecules, probability and time of transition (intrinsic noise)

Continuous Time Markov Chain (CTMC) $A , B \xrightarrow{p(S_i),\, t(Si)} C{+}{+}, A{-}{-}, B{-}{-}$

Petri net semantics: numbers of molecules $A , B \rightarrow C{+}{+}, A{-}{-}, B{-}{-}$

Multiset rewriting

Structural invariants for ODEs



CRN Semantics $A + B \xrightarrow{k.A.B} C$

Differential semantics: concentrations, continuous time evolution

Ordinary differential equations (ODE) $\frac{dA}{dt} = -k.A.B$ $\frac{dB}{dt} = -k.A.B$ $\frac{dC}{dt} = k.A.B$

Stochastic semantics: numbers of molecules, probability and time of transition (intrinsic noise)

Continuous Time Markov Chain (CTMC) $A , B \xrightarrow{p(S_i),\, t(Si)} C{+}{+}, A{-}{-}, B{-}{-}$

Petri net semantics: numbers of molecules $A , B \rightarrow C{+}{+}, A{-}{-}, B{-}{-}$

Multiset rewriting

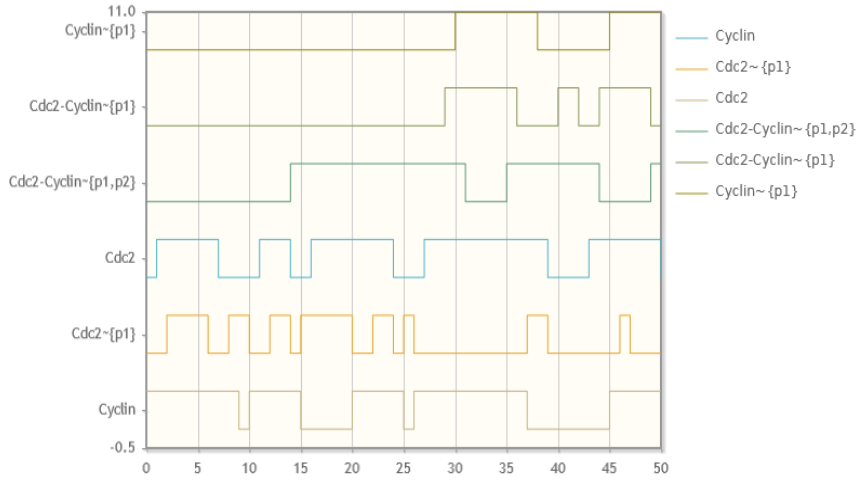
Structural invariants for ODEs

Boolean semantics: presence/absence $A \wedge B \rightarrow C \wedge \neg A \wedge \neg B$

Asynchronous transition system $A \wedge B \rightarrow C \wedge A \wedge \neg B$

$A \wedge B \rightarrow C \wedge \neg A \wedge B$

$A \wedge B \rightarrow C \wedge A \wedge B$



CRN Multiple Semantics
Example of Lotka-Volterra Dynamics

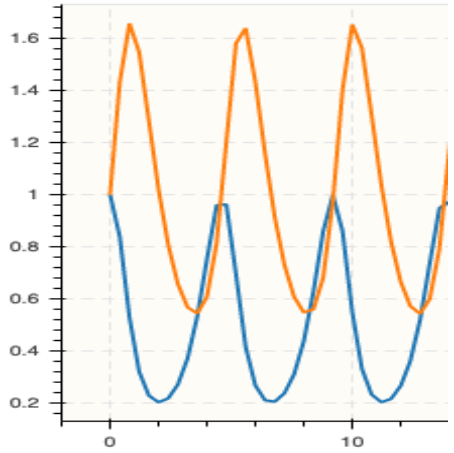
CRN: reaction rules with kinetics

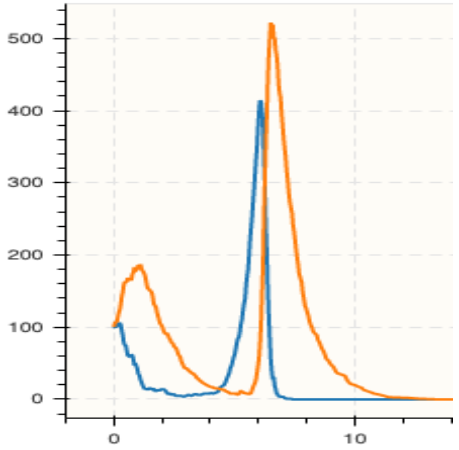
```
MA(k1) for A+B=>2*B
MA(k2) for A=>2*A
MA(k3) for B=>_
```

- ODE semantics:

$$\frac{dB}{dt} = k1 * A * B - k3 * B$$
$$\frac{dA}{dt} = k2 * A - k1 * A * B$$

sustained osscillations





- Stochastic semantics (continuous time Markov chain):

almost sure extinction of the predator

- Boolean semantics:

```
list_stable_states.
```

```
[A-0,B-0]
[A-1,B-0]
```

```
generate_ctl_not.
```

```
reachable(stable(A))
reachable(stable(notA))
reachable(stable(notB))
reachable(steady(B))
```

Model Dynamics Abstractions

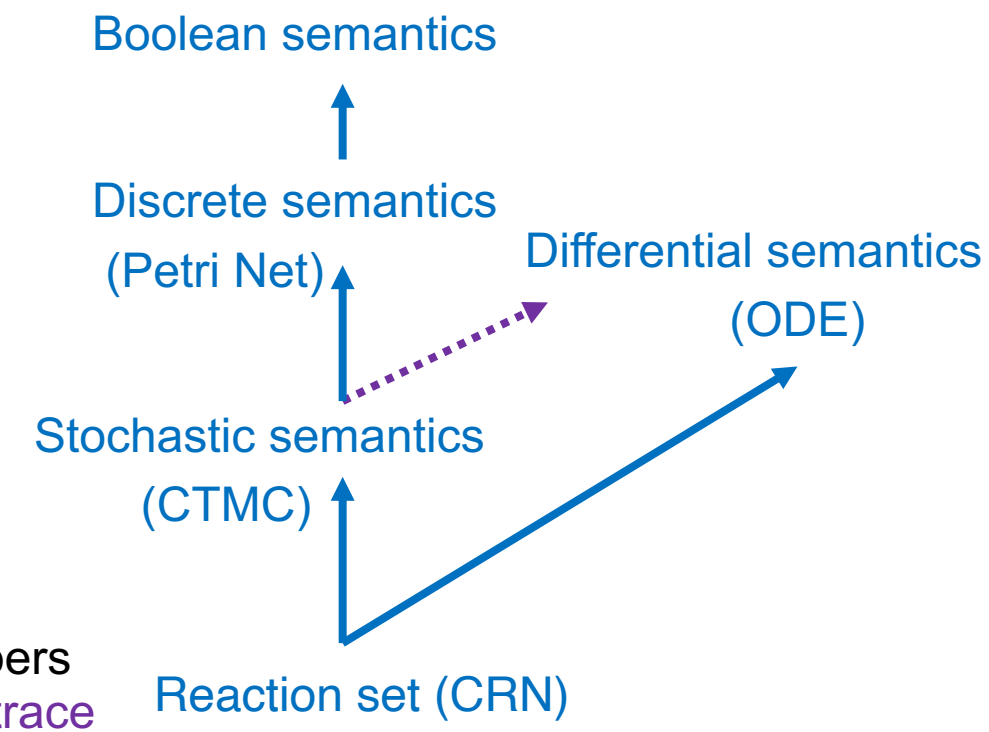
Theory of **Abstract Interpretation** for computer programming [Cousot Cousot *POPL* 1977]
applied here to the CRN programming language to define

- a hierarchy of CRN semantics
- various CRN typings (making SBML annotations formal)

Theorem (abstract interpretation →) Galois connections between the syntactical, stochastic CTMC, Petri net and Boolean transition semantics [F- Soliman *TCS* 2008]

If a behavior is not possible in the Boolean semantics (verifiable by model-checking) it is not possible in the stochastic semantics for any reaction rates.

Boolean model behaviors may correspond to rare events.



Theorem (approximation ⇢) When volume and molecule numbers tend to infinity the ODE trace approximates the mean stochastic trace at all time points [Kurtz 1978, 1992]

Computation Domains as Lattices

A computation domain is a lattice $\mathcal{D}(\sqsubseteq, \sqcup, \sqcap)$ i.e. a poset

- partially ordered by $\sqsubseteq$ approximation ordering (precision loss ordering)
- existence of least upper bound lub $x \sqcup y$ and glb $x \sqcap y$

Complete lattice $\mathcal{D}(\sqsubseteq, \bot, \top, \sqcup, \sqcap)$ if any subset has lub and glb
E.g. power-set domain $\mathcal{P}(\mathcal{S})(\subseteq, \emptyset, \mathcal{S}, \cup, \cap)$ ordered by set inclusion
$\emptyset$: least element, no approximation, $\mathcal{S}$: universal element, greatest approximation

Example (set of possible results or traces of a program execution)

Answer sets of integers $\mathcal{P}(\mathbb{Z})(\subseteq, \emptyset, \mathbb{Z}, \cup, \cap)$.
Answer sets of signs $\mathcal{P}(\{+,-\})(\subseteq, \emptyset, \{+,-\}, \cup, \cap)$.
Integer vector state transition traces $\mathcal{P}((\mathbb{N}^n)^\omega)(\subseteq, \emptyset, (\mathbb{N}^n)^\omega, \cup, \cap)$: Petri Net traces
Integer vector state transition traces $\mathcal{P}((\mathbb{N}^n)^2)(\subseteq, \emptyset, (\mathbb{N}^n)^2, \cup, \cap)$: Petri Net transitions
Boolean traces $\mathcal{P}((\{0,1\}^n)^\omega)(\subseteq, \emptyset, (\{0,1\}^n)^\omega, \cup, \cap)$ Boolean CRN traces
Boolean transitions $\mathcal{P}((\{0,1\}^n)^2)(\subseteq, \emptyset, (\{0,1\}^n)^2, \cup, \cap)$ Boolean CRN transitions

CRN Syntactical Domain

Given a finite set $\mathcal{M}$ of molecule names, the universe of all possible reactions is the set

$\mathcal{R} = \{f \texttt{ for } S => S' \mid \quad S \text{ and } S' \text{ are multisets over } \mathcal{M} \text{ and } f \text{ a formal rate function}$

Definition

The syntactic domain $\mathcal{C}_\mathcal{R}$ of reaction systems over $\mathcal{M}$ is the poset $(\mathcal{P}(\mathcal{R}), \subseteq)$ with $\bot = \emptyset$ the empty model and $\top = \mathcal{R}$ the universal model.

Remark: Monotonicity of non-deterministic trace semantics $\alpha(R)$: more reactions give more (stochastic, Petri net or Boolean) traces, $R \subseteq R' \Rightarrow \alpha(R) \subseteq \alpha(R')$

Remark: Learning CRNs from observed traces can try to

- start from the universal model (most abstract CRN)
- keep only the reactions supported by the data (minimal support CRN)

Domain of Petri Net Transitions

Definition

A discrete (Petri Net) state is a vector of non-negative integers in $\mathbb{N}^{|\mathcal{M}|}$.
The domain of discrete (Petri Net) transitions is $\mathcal{D}_{\mathcal{D}} = (\mathcal{P}(\mathbb{N}^{|\mathcal{M}|} \times \mathbb{N}^{|\mathcal{M}|}), \subseteq)$.

Remark: Discrete states and reactant/product multisets have the same mathematical structure: $|\mathcal{M}|$-dimensional integer vectors.

Remark: In a given discrete state S, the rate function f of a reaction gets a value $f_r(S) \in \mathbb{R}^+$ called the weight, intensity or propensity of the reaction (probability after normalisation).

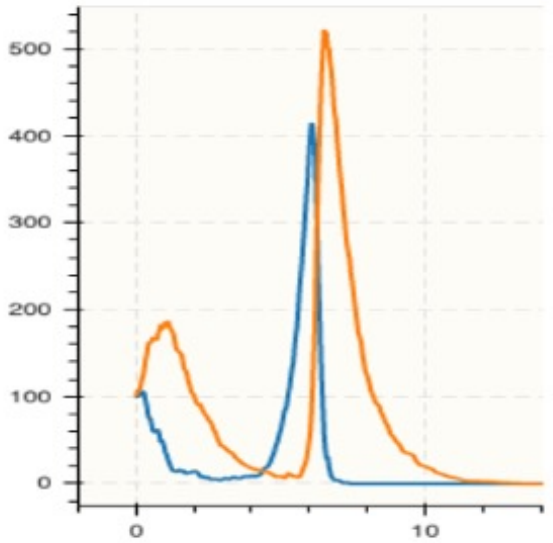
Remark: For a given volume V_k of the location where a compound x_k resides, a concentration C_k for a molecule is translated into a number of molecules $N_k = \lfloor C_k \times V_k \times N_A \rfloor$, where N_A is Avogadro's number, or in practice a smaller conversion factor ($N_A = 100$ by default in Biocham)

Stochastic Transitions and Gillespie's Stochastic Simulation Algorithm

Definition

The universe $\mathcal{S}$ of stochastic transitions over $\mathcal{M}$ is the set of triplets (S, S', τ) where propensity $\tau \in \mathbb{R}^+$ and S, S' are discrete states over $\mathcal{M}$.
The domain of stochastic transitions is $\mathcal{D}_{\mathcal{S}} = (\mathcal{P}(\mathcal{S}), \subseteq)$.

1. Initialize state S and time t
2. Let $f = \sum_r f_r(S)$ total reaction propensities in state S
3. Let r_1, r_2 be two independent uniform(0,1) random numbers
4. Let $\delta t = -ln(r_1)/f$ be the next reaction time
   (exponential distribution of mean $1/f$)
5. Choose reaction r such that $\sum_{k=1}^{r-1} f_k < r_2 f \leq \sum_{k=1}^{r} f_k$
   i.e. reaction r with probability $f_r(S)/f$
6. Update $S := S - S_r + S'_r$ and $t := t + \delta t$ and loop in 2.



What is an Abstraction Formally ?

In what sense can we formally relate, for a given CRN

- the set of its reactions
- the set of its Stochastic transitions
- the set of its Petri net transitions
- the set of its boolean transitions

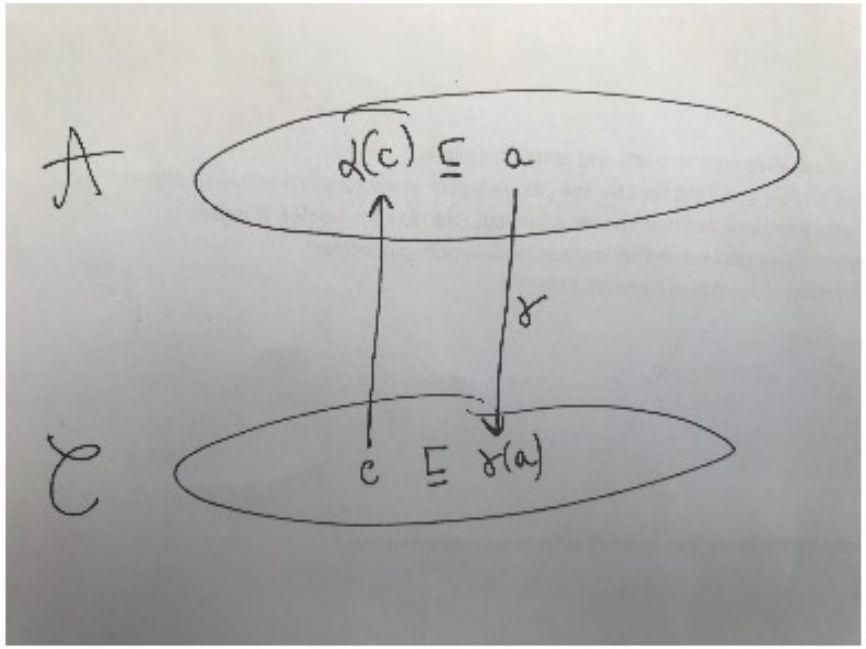
as successive increasing abstractions of the CRN dynamics ?

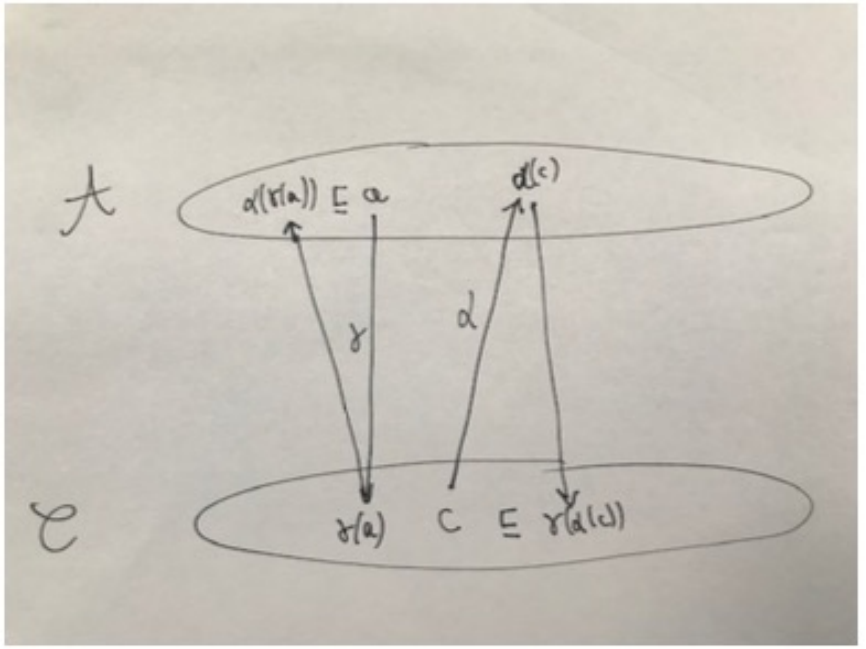
Theory of Abstractions as Galois Connections

Definition

A Galois connection $\mathcal{C} \underset{\gamma}{\overset{\alpha}{\rightleftarrows}} \mathcal{A}$ *between two lattices* $\mathcal{C}$ *and* $\mathcal{A}$ *is a pair of abstraction* $\alpha : \mathcal{C} \to \mathcal{A}$ *and concretization* $\gamma : \mathcal{A} \to \mathcal{C}$ *adjoint functions:*

$$\forall c \in \mathcal{C}, \forall a \in \mathcal{A} : \quad c \sqsubseteq_{\mathcal{C}} \gamma(a) \Leftrightarrow \alpha(c) \sqsubseteq_{\mathcal{A}} a$$





Theory of Abstractions as Galois Connections

Definition

A Galois connection $\mathcal{C} \underset{\gamma}{\overset{\alpha}{\leftrightarrows}} \mathcal{A}$ *between two lattices* $\mathcal{C}$ *and* $\mathcal{A}$ *is a pair of abstraction* $\alpha : \mathcal{C} \to \mathcal{A}$ *and concretization* $\gamma : \mathcal{A} \to \mathcal{C}$ *adjoint functions:* $\forall c \in \mathcal{C}, \forall a \in \mathcal{A}$
$c \sqsubseteq_{\mathcal{C}} \gamma(a) \Leftrightarrow \alpha(c) \sqsubseteq_{\mathcal{A}} a$.

1. duality: $(\mathcal{A}, \sqsupseteq_{\mathcal{A}}) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathcal{C}, \sqsupseteq_{\mathcal{C}})$ is also a Galois connection
2. $\gamma \circ \alpha$ is extensive (precision lost by abstraction)
   $\forall c \in \mathcal{C}\ c \sqsubseteq_{\mathcal{C}} \gamma \circ \alpha(c)$ take $a = \alpha(c)$ in the definition
3. $\alpha \circ \gamma$ is contracting (precision gained by concretization!)
   $\forall a \in \mathcal{A}\ \alpha \circ \gamma(a) \sqsubseteq_{\mathcal{A}} a$ take $c = \gamma(a)$ in the definition
4. α and γ are monotonic
   $c \sqsubseteq_{\mathcal{C}} d \Rightarrow c \sqsubseteq_{\mathcal{C}} \gamma \circ \alpha(d)$ by 2. $\Rightarrow \alpha(c) \sqsubseteq_{\mathcal{A}} \alpha(d)$ by def.
   $a \sqsubseteq_{\mathcal{A}} b \Rightarrow \alpha \circ \gamma(a) \sqsubseteq_{\mathcal{A}} b$ by 3. $\Rightarrow \gamma(a) \sqsubseteq_{\mathcal{C}} \gamma(b)$ by def.

Pointwise Galois Connections between Powersets

Between powersets, any mapping defined pointwise is monotonic and defines a Galois connection abstraction.

Corollary

Let $\mathcal{C}$ and $\mathcal{A}$ be two sets, and $\alpha : \mathcal{P}(\mathcal{C}) \longrightarrow \mathcal{P}(\mathcal{A})$ be a function such that $\alpha(c) = \bigcup_{e \in c} \alpha(\{e\})$.

Then the function $\gamma(a) = \bigcup \alpha^{-1}(\downarrow a)$ forms a Galois connection $\mathcal{P}(\mathcal{C}) \underset{\gamma}{\overset{\alpha}{\rightleftarrows}} \mathcal{P}(\mathcal{A})$ between $(\mathcal{P}(\mathcal{C}), \subseteq)$ and $(\mathcal{P}(\mathcal{A}), \subseteq)$.

Proof.

α is monotonic since $c \subseteq d$ implies $\bigcup_{e \in c} \alpha(\{e\}) \subseteq \bigcup_{e' \in d} \alpha(\{e'\})$.
The previous proposition on $\gamma(a)$ concludes the proof. □

Abstraction: Syntactic CRN Domain $\rightarrow$ Stochastic Trace Domain

Definition

Let $\alpha_{\mathcal{RS}} : \mathcal{C}_{\mathcal{R}} \rightarrow \mathcal{D}_{\mathcal{S}}$ *associate the set of transition triplets with propensities* $\{(S, S', f_r(S)) \mid S \in \mathcal{S},\ (f_r \texttt{ for } S_r \Rightarrow S'_r) \in R,\ S \geq S_r,\ S' = S - S_r + S'_r\}$

Proposition

Let $\gamma_{\mathcal{RS}}(s) = \bigcup \alpha_{\mathcal{RS}}{}^{-1}(\downarrow s)$. $\mathcal{C}_{\mathcal{R}} \underset{\gamma_{\mathcal{RS}}}{\overset{\alpha_{\mathcal{RS}}}{\longleftrightarrow}} \mathcal{D}_{\mathcal{S}}$ *is a Galois connection.*

Remark: $\alpha_{\mathcal{RS}}$ is not surjective. Because of the properties of CRN stochastic traces, e.g.

Abstraction: Syntactic CRN Domain $\rightarrow$ Stochastic Trace Domain

Definition

Let $\alpha_{\mathcal{RS}} : \mathcal{C}_{\mathcal{R}} \rightarrow \mathcal{D}_{\mathcal{S}}$ *associate the set of transition triplets with propensities* $\{(S, S', f_r(S)) \mid S \in \mathcal{S},\ (f_r \texttt{ for } S_r \Rightarrow S'_r) \in R,\ S \geq S_r,\ S' = S - S_r + S'_r\}$

Proposition

Let $\gamma_{\mathcal{RS}}(s) = \bigcup \alpha_{\mathcal{RS}}{}^{-1}(\downarrow s)$. $\mathcal{C}_{\mathcal{R}} \underset{\gamma_{\mathcal{RS}}}{\overset{\alpha_{\mathcal{RS}}}{\longleftrightarrow}} \mathcal{D}_{\mathcal{S}}$ *is a Galois connection.*

Remark: $\alpha_{\mathcal{RS}}$ is not surjective. Because of the properties of CRN stochastic traces, e.g. *monotonicity* $(S_1, S, f(S_1)) \in \alpha_{\mathcal{RS}}(R) \Rightarrow (S_2, S + S_2 - S_1, f(S_2)) \in \alpha_{\mathcal{RS}}(R)$ for $S_1 \leq S_2$. The information gained by $\alpha \circ \gamma$ is the elimination of incomplete transitions.

Abstraction: Syntactic CRN Domain $\rightarrow$ Stochastic Trace Domain

Definition

Let $\alpha_{\mathcal{RS}} : \mathcal{C}_{\mathcal{R}} \rightarrow \mathcal{D}_{\mathcal{S}}$ *associate the set of transition triplets with propensities* $\{(S, S', f_r(S)) \mid S \in \mathcal{S},\ (f_r \texttt{ for } S_r \Rightarrow S'_r) \in R,\ S \geq S_r,\ S' = S - S_r + S'_r\}$

Proposition

Let $\gamma_{\mathcal{RS}}(s) = \bigcup \alpha_{\mathcal{RS}}{}^{-1}(\downarrow s)$. $\mathcal{C}_{\mathcal{R}} \underset{\gamma_{\mathcal{RS}}}{\overset{\alpha_{\mathcal{RS}}}{\longleftrightarrow}} \mathcal{D}_{\mathcal{S}}$ *is a Galois connection.*

Remark: $\alpha_{\mathcal{RS}}$ is not surjective. Because of the properties of CRN stochastic traces, e.g. *monotonicity* $(S_1, S, f(S_1)) \in \alpha_{\mathcal{RS}}(R) \Rightarrow (S_2, S + S_2 - S_1, f(S_2)) \in \alpha_{\mathcal{RS}}(R)$ for $S_1 \leq S_2$. The information gained by $\alpha \circ \gamma$ is the elimination of incomplete transitions.
Remark: $\alpha_{\mathcal{RS}}$ is not injective. The same set of stochastic transitions is obtained with $CRN1 = \{\texttt{e for A=>B}\}$ and $CRN2 = CRN1 \cup \{\texttt{e for 2*A=>A+B}\}$. $\gamma \circ \alpha$ is not the identity. The information lost is the absence of redundant reactions $CRN1 \subsetneq CRN2$.

Abstraction: Stochastic Trace $\rightarrow$ Petri Net Trace

Trivial forgetful functor abstraction: just forget about propensities.

Proposition

Let $\alpha_{\mathcal{SD}} : \mathcal{D}_{\mathcal{S}} \rightarrow \mathcal{D}_{\mathcal{D}}$ be the function associating to a set of stochastic transitions the discrete transitions obtained by projection on the two first components, and $\gamma_{\mathcal{SD}}(d) = \bigcup \alpha_{\mathcal{SD}}{}^{-1}(\downarrow d)$.
$\mathcal{D}_{\mathcal{S}} \underset{\gamma_{\mathcal{SD}}}{\overset{\alpha_{\mathcal{SD}}}{\leftrightarrows}} \mathcal{D}_{\mathcal{D}}$ is a Galois connection.

Remark: $\alpha_{\mathcal{SD}}$ is surjective, but not $\alpha_{\mathcal{SD}} \circ \alpha_{\mathcal{RS}}$ by previous prop.

$\alpha_{\mathcal{SD}}$ is not injective as the transition rates are simply forgotten.

Abstraction: Petri Net Trace → Boolean Trace

Let a boolean state be a vector of booleans of dimension $|\mathcal{M}|$ indicating the absence/presence of each molecule in the state.

Definition

The universe $\mathcal{B}$ of boolean transitions is the set of pairs of boolean states. The domain of boolean transitions is $\mathcal{D}_\mathcal{B} = (\mathcal{P}(\mathcal{B}), \subseteq)$.

Let $\alpha_{\mathcal{NB}} : \mathbb{N}^{|\mathcal{M}|} \to \mathbb{B}^{|\mathcal{M}|}$ be the zero/non-zero abstraction $\alpha_{\mathcal{NB}}(v) = (v \neq 0)$ and its pointwise extension from discrete states to boolean states.

Proposition

Let $\alpha_{\mathcal{DB}} : \mathcal{D}_\mathcal{D} \to \mathcal{D}_\mathcal{B}$ be the set extension of $\alpha_{\mathcal{NB}}$. Let $\gamma_{\mathcal{DB}}(b) = \bigcup \alpha_{\mathcal{DB}}^{-1}(\downarrow b)$.
$\mathcal{D}_\mathcal{D} \underset{\gamma_{\mathcal{DB}}}{\overset{\alpha_{\mathcal{DB}}}{\leftrightarrows}} \mathcal{D}_\mathcal{B}$ is a Galois connection.

Remark: $\alpha_{\mathcal{DB}}$ is surjective but obviously not injective: gets rid of molecule numbers.

Abstraction: Syntactic CRN → Biocham Boolean Trace Semantics

$$\alpha_{RB}(\{R \to P\}) = \{(S, S') \in \mathcal{B} \mid S \geq \alpha_{\mathcal{NB}}(R) \\ ((S \wedge \neg\alpha_{\mathcal{NB}}(R)) \vee \alpha_{\mathcal{NB}}(P)) \leq S' \leq (S \vee \alpha_{\mathcal{NB}}(P))\}$$

Boolean transitions obtained with possible consumption or not of the reactants.

Remark: Biocham differs from standard boolean PN semantics which consumes all reactants. For instance, the reaction `A+B=>C+D` gives rise to 4 transitions:

- $S \wedge A \wedge B \longrightarrow S \wedge \neg A \wedge \neg B \wedge C \wedge D$
- $S \wedge A \wedge B \longrightarrow S \wedge \neg A \wedge B \wedge C \wedge D$
- $S \wedge A \wedge B \longrightarrow S \wedge A \wedge \neg B \wedge C \wedge D$
- $S \wedge A \wedge B \longrightarrow S \wedge A \wedge B \wedge C \wedge D$

Question: which transitions should be associated for the *boolean threshold abstraction* $\alpha_\theta(v) = (v > \theta)$ instead of zero/non-zero abstraction $\alpha(v) = (v \neq 0)$?

Answer: add transitions for production or not of C, D. Reaction synchrony lost: equivalent to influence model with generalized asynchronous semantics.

Theorem

For any reaction system R, $\alpha_{\mathcal{DB}}(\alpha_{\mathcal{SD}}(\alpha_{\mathcal{RS}}(R))) \subseteq \alpha_{RB}(R)$.

Type Checking/Inference by Abstract Interpretation

A type system for C is a Galois connection $\mathcal{C} \to_\alpha \mathcal{A}$.

The type inference problem is

Input a concrete element $x \in \mathcal{C}$, e.g. a CRN

Output its typing $\alpha(x)$, e.g. parameter dimensions.

The type checking problem is,

Input $x \in \mathcal{C}$ with a typing $y \in \mathcal{A}$

Output whether $x \sqsubseteq_\mathcal{C} \gamma(y)$
(correct typing)
or equivalently $\alpha(x) \sqsubseteq_\mathcal{A} y$
(compatible with inferred type)

In linear time for abstractions computable reaction per reaction.

Dimensional Analysis: Typing of Kinetic Parameters

The dimension of parameters (in terms of time and volume)
can be inferred from the rate functions of the reactions
by accumulation of dimension constraints
as a typing abstraction:

```
biocham: v*A/(k+A) for A => B.
biocham: parameter(k = 1, v = 1).
biocham: list_dimensions.
v has dimension time^(-1).volume^(0)
k has dimension time^(0).volume^(-1)
```

Typing constraints used:
$dim(k) = volume^{-1} \ \wedge \ dim(v).volume^{-1}/volume^{-1} = time^{-1}$

Typing CRNs by their Differential Influence Graph (DIG)

```
k1 for _ => A
k2*[A] for A => _
k3*[A]*[B] for A + B => C
```

$$\dot{x_A} = k1 - k2 * x_A - k3 * x_A * x_B$$
$$\dot{x_B} = -k3 * x_A * x_B$$
$$\dot{x_C} = k3 * x_A * x_B$$

Definition

The differential semantics of a reaction model

$$R = \{f_i \texttt{ for } l_i \texttt{ => } r_i\}_{i=1,\ldots,n}$$

is the ODE system $dx_k/dt = \dot{x_k} = \sum_{i=1}^{n} v_i(x_k) * f_i$
where $v_i = r_i - l_i$ *is the stoichiometric change vector of reaction* i.

Positive and negative influences can be defined by
the Jacobian matrix $J_{ij} = \partial \dot{x_i}/\partial x_j$

Typing CRNs by their Differential Influence Graph (DIG)

$A \overset{+}{\to} B$ if positive influence of molecule A on molecule B

$A \overset{-}{\to} B$ if negative influence of molecule A on molecule B

Definition

The differential influence graph (DIG) of a reaction model R is the graph of molecules with two kinds of edges:

$$\alpha_{\mathcal{JI}}(R) = \{A \overset{+}{\to} B \mid \partial \dot{x_B}/\partial x_A > 0 \text{ at some point in } \mathbb{R}^n_+\} \cup \{A \overset{-}{\to} B \mid \partial \dot{x_B}/\partial x_A < 0 \text{ at some point in } \mathbb{R}^n_+\}$$

```
k1 for _ => A
k2*[A] for A => _
k3*[A]*[B] for A + B => C
```

$$\dot{x_A} = k1 - k2 * x_A - k3 * x_A * x_B$$
$$\dot{x_B} = -k3 * x_A * x_B$$
$$\dot{x_C} = k3 * x_A * x_B$$

$\text{DIG} = \{A \overset{-}{\to} A,\ B \overset{-}{\to} A,\ A \overset{-}{\to} B,\ B \overset{-}{\to} B,\ A \overset{+}{\to} C,\ B \overset{+}{\to} C\}$

Typing CRNs by their Stoichiometric Influence Graph (SIG)

Definition

The stoichiometric influence graph *(SIG) of a reaction model R is defined by*

$$\alpha_{\mathcal{RI}}(R) = \{A \xrightarrow{+} B \mid \exists(f_i \text{ for } I_i \Rightarrow r_i) \in R, I_i(A) > 0 \text{ and } v_i(B) > 0\} \cup \{A \xrightarrow{-} B \mid \exists(f_i \text{ for } I_i \Rightarrow r_i) \in R, I_i(A) > 0 \text{ and } v_i(B) < 0\}$$

$\alpha_{\mathcal{RI}}(\{\texttt{B => B+A}\}) = \{\texttt{B} \xrightarrow{+} \texttt{A}\}$

$\alpha_{\mathcal{RI}}(\{\texttt{A + B => B \_}\}) = \{\texttt{B} \xrightarrow{-} \texttt{A}, \texttt{A} \xrightarrow{-} \texttt{A}\}$

$\alpha_{\mathcal{RI}}(\{\texttt{A + C => C+ B }\}) = \{\texttt{C} \xrightarrow{-} \texttt{A}, \texttt{A} \xrightarrow{-} \texttt{A}, \texttt{A} \xrightarrow{+} \texttt{B}, \texttt{C} \xrightarrow{+} \texttt{B}\}$

$\alpha_{\mathcal{RI}}(\{\texttt{A + B => C}\}) = \{\texttt{A} \xrightarrow{+} \texttt{C}, \texttt{B} \xrightarrow{+} \texttt{C}, \texttt{A} \xrightarrow{-} \texttt{B}, \texttt{B} \xrightarrow{-} \texttt{A}, \texttt{A} \xrightarrow{-} \texttt{A}, \texttt{B} \xrightarrow{-} \texttt{B}, \}$

Proposition

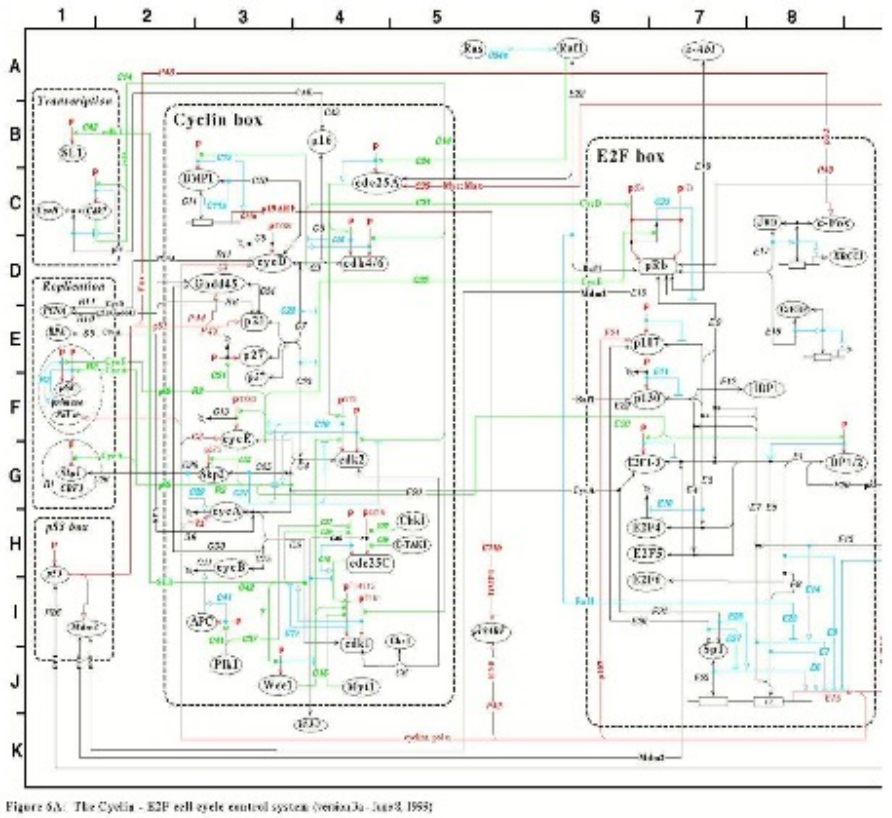
The SIG of n reaction rules is computable in $O(n)$ time

The SIG of Kohn's Map of the Mammalian Cell Cycle

Reaction model:
500 variables
800 reaction rules

Stoichiometric Influence Graph:
computed in *0.2 sec.*
1231 positive influences
1089 negative influences

No tuple (A,B) with both A $\xrightarrow{+}$B and A $\xrightarrow{-}$B.



DIG⊆SIG

Theorem

For any reaction model R with increasing kinetics, the DIG is a subgraph of the SIG: $\alpha_{\mathcal{JI}}(R) \subseteq \alpha_{\mathcal{RI}}(R)$.

Proof.

If $(A \overset{+}{\to} B) \in \alpha_{\mathcal{JI}}(R)$ then $\partial \dot{x_B}/\partial x_A > 0$ at some point,
hence there exists a term of the form $v_i(B) * f_i$ with $\partial f_i/\partial x_A$ of the same sign as $v_i(B)$.
If $v_i(B) > 0$, then $\partial f_i/\partial x_A > 0$ and $l_i(A) > 0$ as f_i is increasing
so $(A \overset{+}{\to} B) \in \alpha_{\mathcal{RI}}(R)$.
If on the contrary $v_i(B) < 0$, then $\partial f_i/\partial x_A < 0$, contradiction.
The proof is symmetrical for $(A \overset{-}{\to} B)$. □

DIG≠ SIG for $\{k_1 * A \texttt{ for } A => \_, \quad k_2 * A \texttt{ for } A => 2 * A\}$
$\dot{x_A} = (k_2 - k_1) * x_A$ can be made positive, null or negative.

DIG=SIG Condition

Theorem

Let R be a reaction model with well-formed increasing kinetics and where no molecule is at the same time an activator and an inhibitor of the same target molecule, then $\alpha_{\mathcal{RI}}(R) = \alpha_{\mathcal{JI}}(R)$.

Corollary

The DIG of a CRN is independent of the kinetic expressions as long as they are well-formed increasing, and there is no positive−negative influence pairs in the SIG.

Corollary

The DIG of a CRN with n reactions with well-formed increasing kinetics is computable in time $O(n)$ if there is no positive−negative pair in the SIG.

2. CRN Reductions

1. Example of Michaelis-Menten reduction
   - Conservation laws
   - Quasi-steady state approximation

2. Slow-fast ODE system decompositions
   - Tikhonov theorem

3. General notion of CRN reduction by subgraph epimorphism (SEPI)
   - Graph-theoretic model reduction operations
   - NP-completeness of SEPI detection
   - Automatic reconstruction of model hierarchies in BioModels.net
   - Comparison between synthetic CRNs and natural CRNs

Michaelis-Menten CRN

An enzyme E binds to a substrate S to catalyze the formation of product P:

$E+S \xrightarrow{k1} C \xrightarrow{k3} E+P$

$E+S \xleftarrow{k2} C$

Mass action law kinetics ODE:

$dE/dt = -k1.E.S+(k2+k3).C$

$dS/dt = -k1.E.S+k2.C$

$dC/dt = k1.E.S-(k2+k3).C$

$dP/dt = k3.C$

Two conservation laws (species s.t. $\Sigma^n_{i=1}Mi$ = *constant* as $\Sigma^n_{i=1}$ dMi/dt = 0, also Petri net place invariant)

$E+C=E_0+C_0, \quad S+C+P=S_0+C_0+P_0$

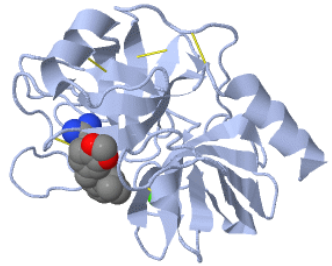
One can eliminate two variables $E= E_0+C_0-C$ and P and get the equivalent ODE system with **C_0 S_0 fixed**

$dS/dt = -k1.(E_0+C_0-C).S+k2.C$ (bad practice to deposit SBML model with invariant eliminated)

$dC/dt = k1.(E_0+C_0).S-(k1.S+k2+k3).C$

Let us further assume $C_0=0$, $P_0=0$

Michaelis-Menten Slow/Fast Time Scales

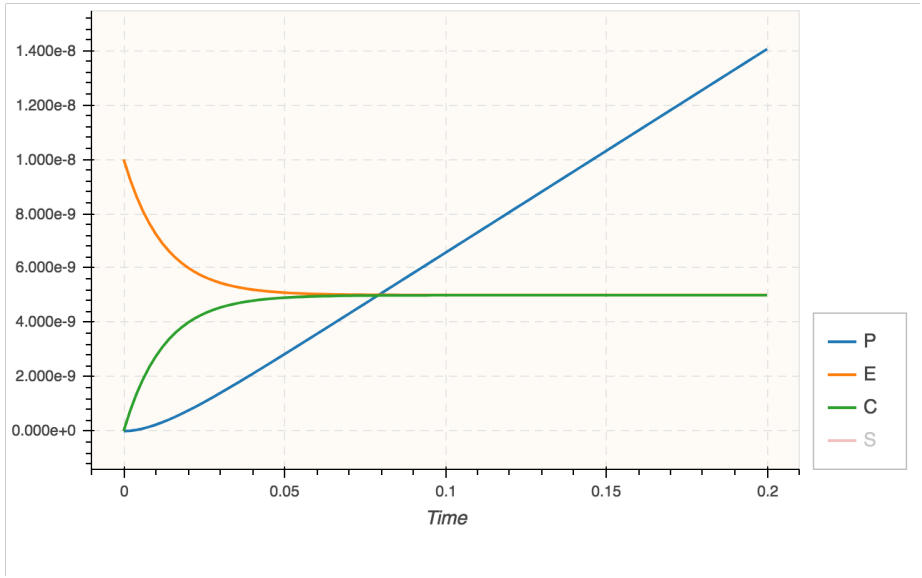


Hydrolysis of benzoyl-L-arginine ethyl ester by **trypsin** (protein of 223 amino acids)

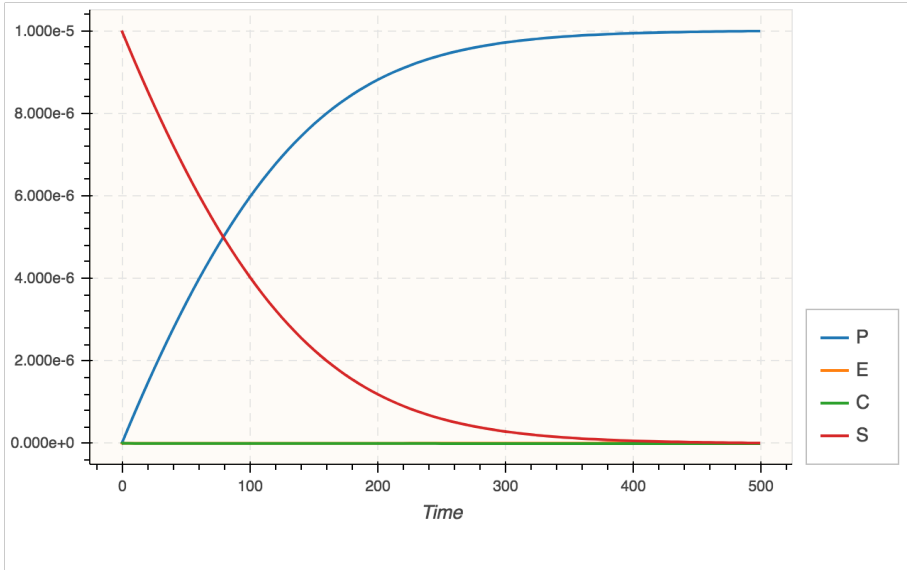
```
present(E,1e-8). present(S,1e-5).
parameter(k1=4e6, k2=25, k3=15).
k1*E*S for E+S => C.
k2*C for C => E+S.
k3*C  for C => E+P.
```

E << S

Complex formation 5e-9 in **0.1s**



Product formation 1e-5 in **400s**



Quasi-Steady State Approximation (QSSA)

After short time assume $dC/dt \simeq 0 \simeq k1E_0S-(k1S+k2+k3)C$

Then $C = k1E_0S/(k1S+k2+k3)$

$= E_0S/(S+(k2+k3)/k1)$

$= E_0S/(K_m+S)$ with $K_m=(k2+k3)/k1$

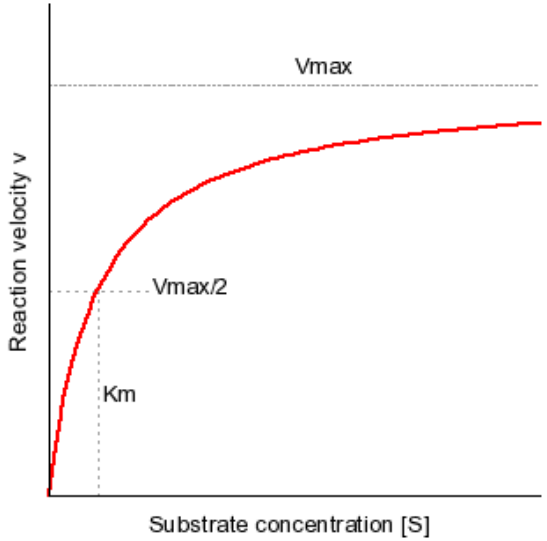
K_m is substrate concentration with half maximum velocity

We get $dP/dt = -dS/dt = -k1(E_0-C)S+k2C$

$= -k1E_0S + (k1S+k2)\,E_0\,S\,/\,(K_m+S)$

$= V_m\,S\,/\,(K_m+S)$ where $V_m= k3E_0$

V_m is maximum velocity at saturing substrate concentration



Michaelis-Menten kinetics: `Vm S/(Km+S) for S => P`

Leonor Michaelis and Maude Menten 1913

Victor Henry (X) 1903

C and E are eliminated but sometimes E is re-injected as a slow variable…

`k3*E*S /(Km+S) for S+E => E+P`

Slow-Fast ODE Decomposition and Reduction to Slow Dynamics

Tikhonov theorem.

Consider an ODE system defined for $(X, Y) \in R^n \times R^m$

$dX/dt = f(X, Y)$

$\epsilon\, dY/dt = g(X, Y)$ with $\epsilon \ll 1$

such that for $X \in U$ $g(X, Y) = 0 \iff Y = G(X)$ with $(X, Y) \in W$

and $G(X)$ is an asymptotically stable fixed point for the fast subsystem

then,

for any initial condition $(X0, Y0)$ s.t. $Y0$ belongs to the basin of attraction of $G(X0)$ for the fast subsystem,

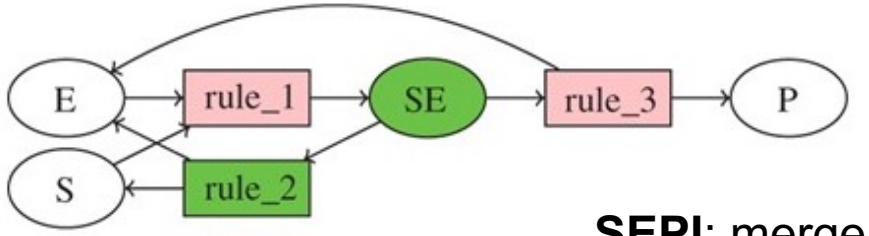
the solution $(X(t), Y(t))$ tends to $(x(t), G(x(t)))$, when $\epsilon \to 0$,

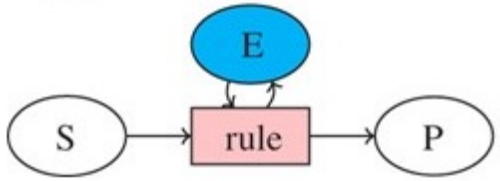
where $x(t)$ is the solution of the slow subsystem $dx/dt = f(x, G(x))$ with initial condition $X0$.

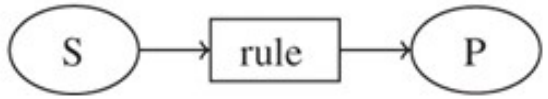
- Limit theorem with no bound on the error.
- Solution Y=G(X) may be difficult to express and stability difficult to prove.
- Different decompositions for different regimes lead to an automaton of reduced ODE systems with gluing pb

Model Reductions on the CRN Hypergraph Structure by Subgraph Epimorphisms [Gay F- Soliman ECCB 2010]

Reaction hypergraph represented by bipartite graph of species and reactions.







SEPI: merge r1 r3 delete r2 SE

$\{rule_1 \rightarrow rule,\ rule_3 \rightarrow rule,\ SE \rightarrow \perp,\ rule_2 \rightarrow \perp\}$

SEPI, SISO: delete E

$\{E \rightarrow \perp\}$

Model reduction by 4 graph edition operations:

- Delete (irrelevant) species nodes
- Delete (neglectable) reaction nodes

) SISO: subgraph isomorphism

- Merge (similar) species nodes
- Merge (chain) reaction nodes

) EPI: epimorphism

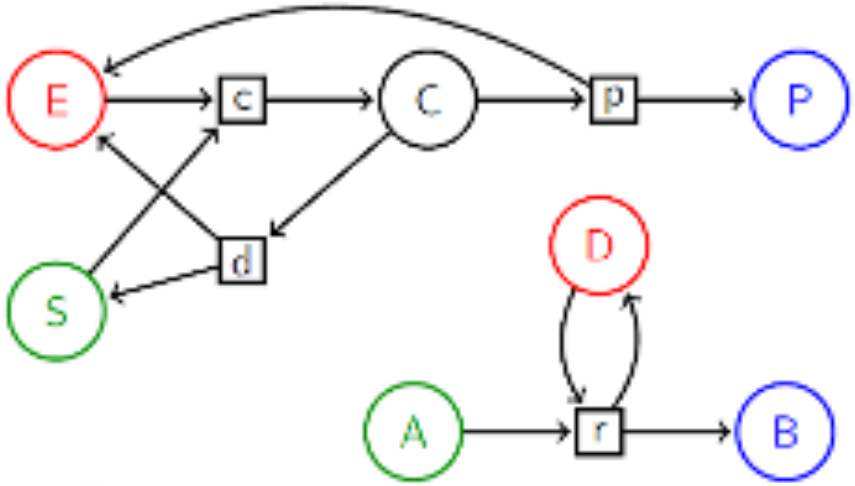
) SEPI: subgraph epimorphism

A graph morphism from G to G' associates G' nodes to G nodes and preserves G arcs

Theorem. *Let G = (S, R, A) and G' = (S', R', A') be two reaction graphs.*
There exists a subgraph epimorphism μ *from G to G' if and only if there exists a finite sequence of* delete *and* merge *operations that, when applied to G, yield a graph isomorphic to G'.*

Detecting the Existence of a SEPI between CRN Graphs

	E	$\to$	D
	S	$\to$	A
	P	$\to$	B
merge	c	$\to$	r
merge	p	$\to$	r
delete	d	$\to$	$\bot$
delete	C	$\to$	$\bot$



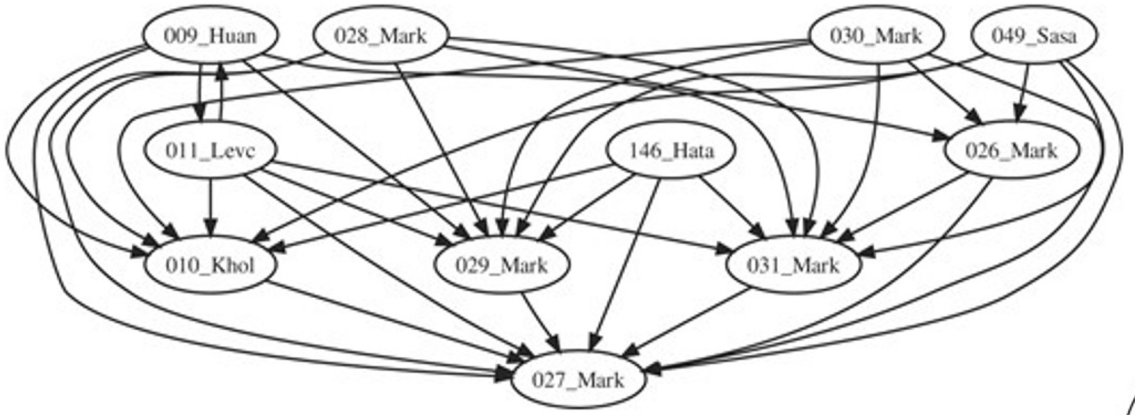
Theorem. [Gay Martinez F- Soliman Solnon DAM 2014] *Let G = (S, R, A) and G' = (S', R', A') be two reaction graphs. Deciding the existence of a* subgraph epimorphism μ *from G to G' is NP-complete.*

- Implemented in Biocham using Constraint Logic Program or SAT solver.
- Some timeouts in BioModels for models above 100 species

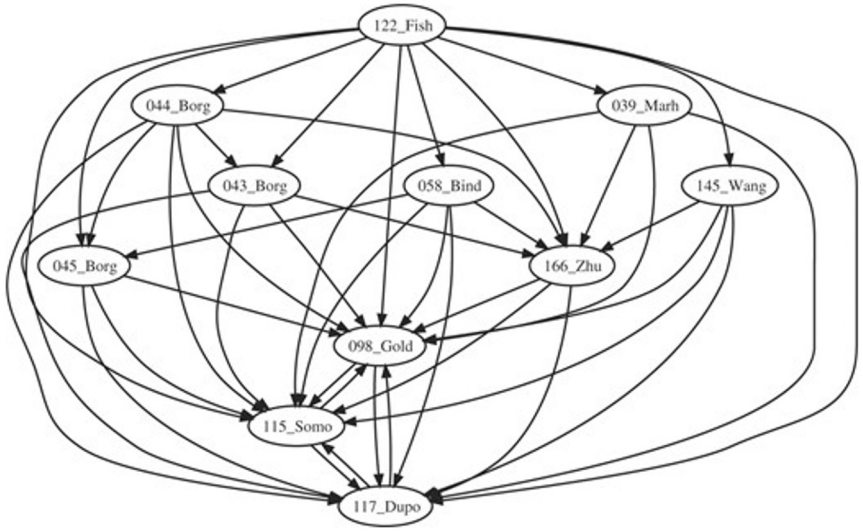
- Non-unique sepi-glb(G, G') nor sepi-lub(G,G') which combines details of G and G' for free!
- No good algorithm to restrict the merge operation to neighboring species or reactions (as in graph minors)

SEPI-detection of Metamodels in BioModels

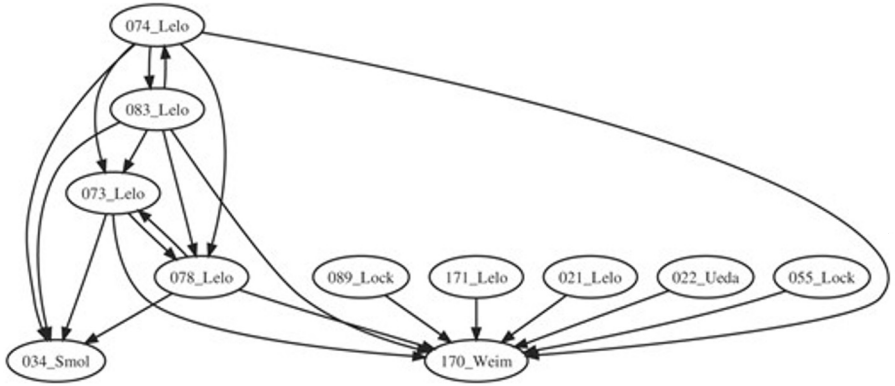
MAPK metamodel:



Calcium oscillation metamodel:

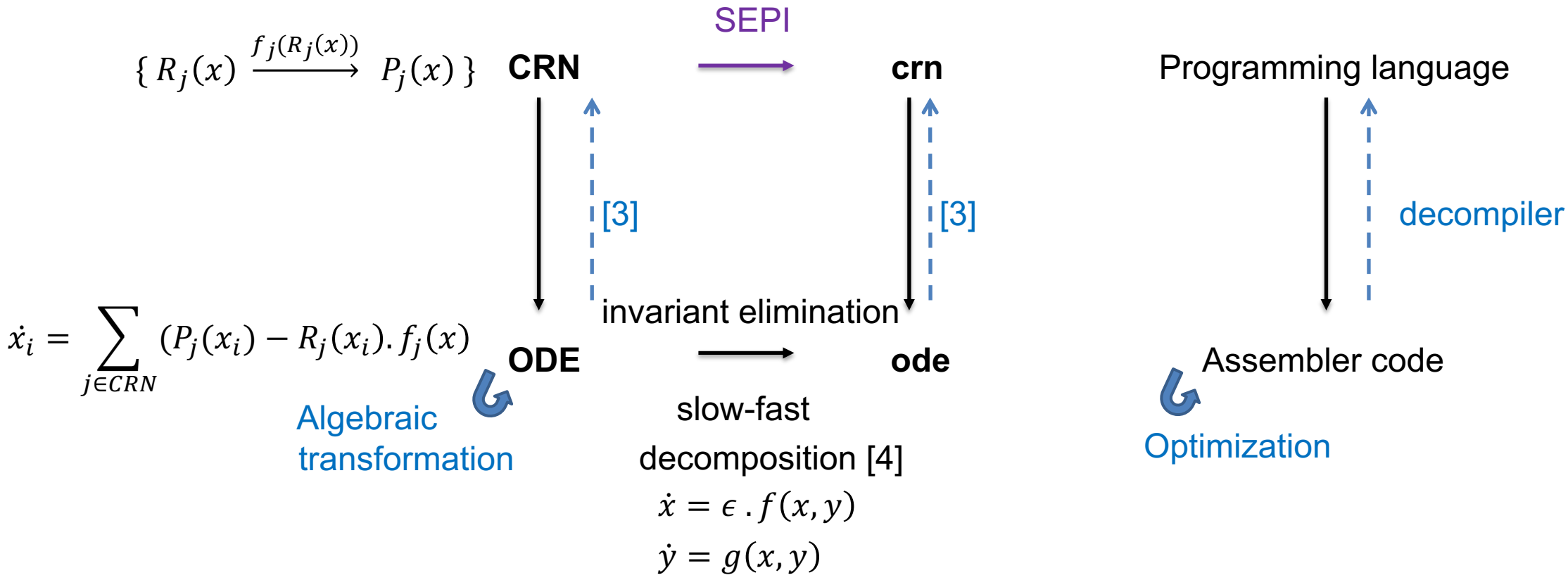


Circadian clock metamodel:



Also used to compare synthetic CRN to natural CRNs (next evening lecture) [F- Le Guludec Bournez Pouly CMSB 2017 Hemery F- CMSB 2023 ?]

Model Reduction Methods for CRN versus ODE Models



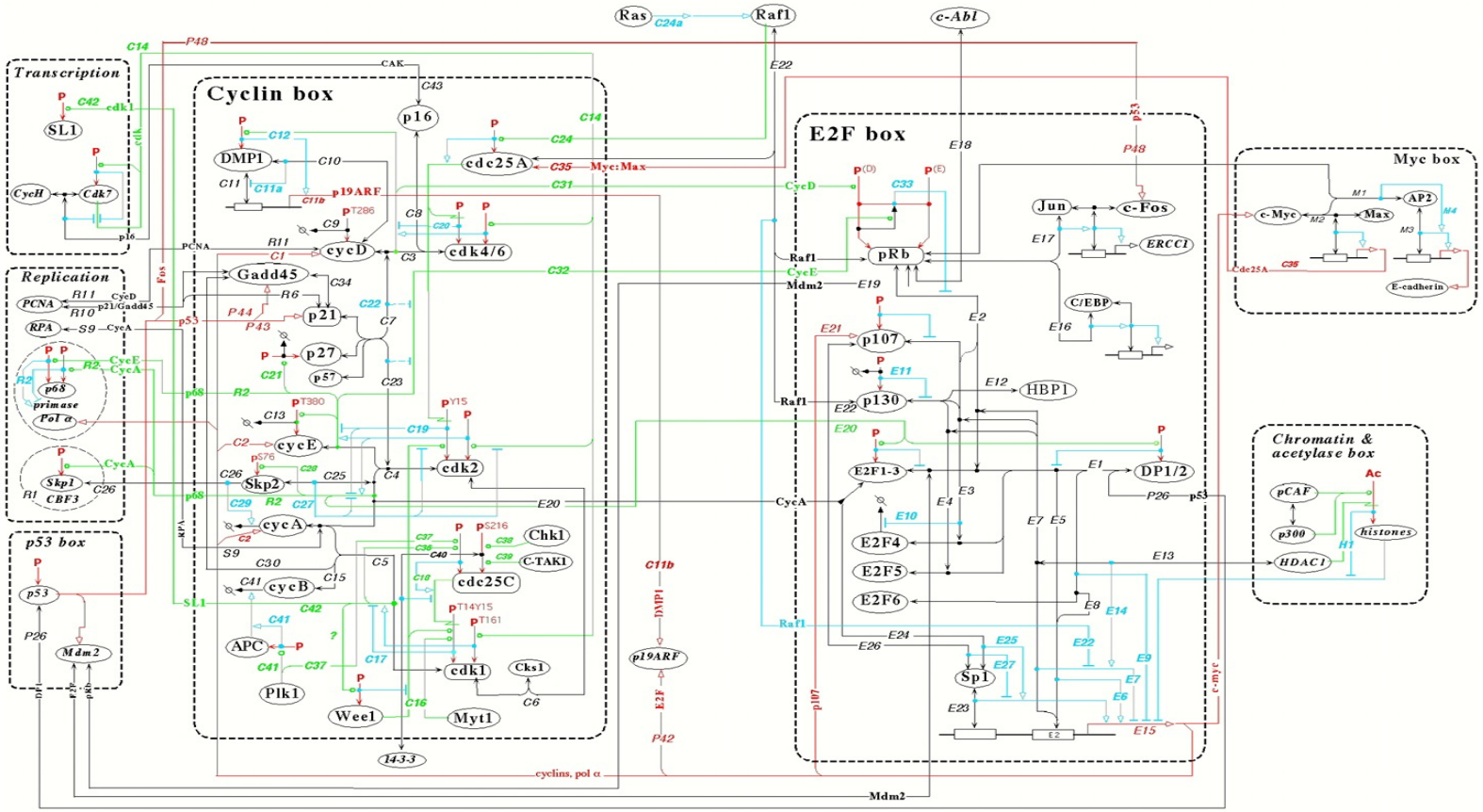
[3] François Fages, Steven Gay, Sylvain Soliman. Inferring Reaction Systems from Ordinary Differential Equations. *TCS* 2015.
[4] Sylvain Soliman, François Fages, Ovidiu Radulescu. A constraint solving approach to model reduction by tropical equilibration. *AMB*, 2014.

Coffee ?

4. Temporal Logics as Specification Language of CRN Behaviors

How to query the possible transitions of Kohn's map (1999) of the cell cycle ?

By model-checking ! [Chabrier F- *CMSB* 2003, Bernot Comet Peres Richard *CMSB* 2003, Lincoln et al *PCB* 2002]



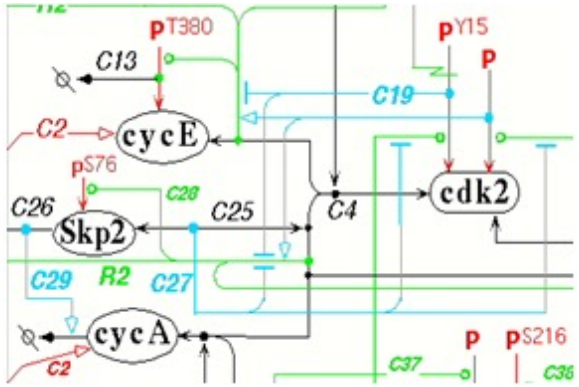
Transcription of Kohn's Map in Reaction System

Detail of the complexation of cdk2 with cycA and cycE :
Total:

→ 165 proteins and genes

→ 532 variables

→ 732 reactions



No kinetics

Boolean state transition semantics:

- Asynchronous: selection of one reaction firing at a time: `A+B => C`
- Non-deterministic: selection of one Boolean transition for that reaction:

$$A \wedge B \rightarrow C \wedge \neg A \wedge \neg B$$
$$A \wedge B \rightarrow C \wedge A \wedge \neg B$$
$$A \wedge B \rightarrow C \wedge \neg A \wedge B$$
$$A \wedge B \rightarrow C \wedge A \wedge B$$

- zero/non-zero abstraction of the stochastic/Petri net transitions (first course)

Symbolic Representation of Boolean Transition Systems

How to represent a transition system over

- 532 boolean variables ?
- $2^{532} \sim 10^{177}$ boolean states ?? >>10^{80} the number of atoms in the observable universe
- $2^{2^{532}}$ sets of boolean states ???

Represent a set of states by a Boolean constraint over n Boolean variables:

- *False*: empty set
- *True*: full set of 2^{532} states
- *A*: set of 2^{531} states where A is present
- $A \vee \neg B$: set of 3.2^{530} states with either A present or B absent
- Of course some (bad) sets require formulae of exponential size

Represent a transition relation by a Boolean constraint over $2.n$ variables

- $R(A_1, \ldots, A_N, A'_1, \ldots, A'_n)$ disjunction of the relation associated to each reaction
- Reaction `A+B => C` transition relation over set of species {A,B, C, D, E} ?

  $A \wedge B \wedge C' \wedge (D' = D) \wedge (E' = E)$

  i.e. C appears (C' is true) A and B may disappear, D, E remain unchanged

Kohn's Map Model-Checking

BIOCHAM NuSMV symbolic model-checker time in seconds [Chabrier Fages CMSB 2003]

Initial state G2	Query:	Time:
	Compiling of the set of initial states and transition system	29s
Reachability of G1 phase	**EF** CycE	2s
Reachability of G1 phase	**EF** CycD	1.9s
Checkpoint for mitosis complex	¬**E (**¬Cdc25Nterm **U** Cdk1Thr161-CycB	2.2s
Oscillations CycA	**EG ((EF** ¬ CycA**)** ∧ **(EF** CycA**))**	31.8s
Oscillations CycB	**EG ((EF** ¬ CycB**)** ∧ **(EF** CycB**))** false ! (omission of CycB synthesis in Kohn's map)	6s

Computation Tree Logic CTL*

Infinite computation pathways.

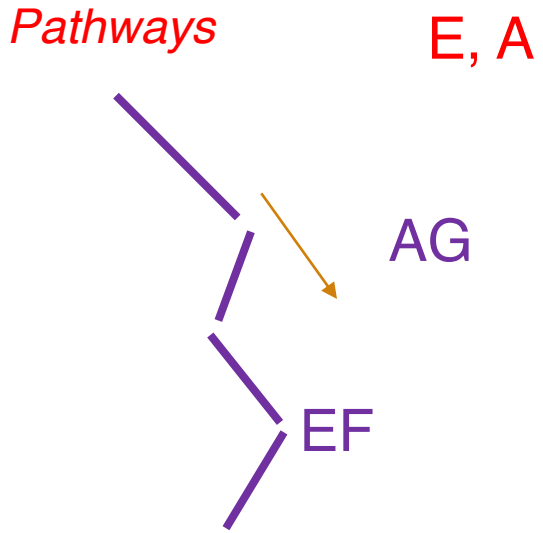
Propositional logic with **modal operators** for qualifying when (in logical time future) and where (on which computation path) a Boolean proposition is true.

Introduced for **program verification** and **program synthesis** in [Pnueli 1977]

\ Pathways Time	E Exists path	A All paths
X next	EX(ϕ)	AX(ϕ)
F finally	EF(ϕ) ***reachability***	AF(ϕ) ***liveness***
G globally	EG(ϕ)	AG(ϕ) ***safety***
U until	E (ϕ_1 U ϕ_2)	A (ϕ_1 U ϕ_2)

F,G,U

Time



Kripke Semantics of CTL*

A Kripke structure K=(S,R) is a set S of states with a total relation $R \subseteq SxS$

The **truth of a formula ϕ in a state s or on a path π** of K is defined by:

$\pi \models \phi$ for a state formula ϕ if $s \models \phi$ where **s is the initial state of π**

$s \models \phi$ if ϕ is a propositional formula true in s

$s \models \mathbf{E}\ \phi$ if there is a path π starting from s such that $\pi \models \phi$	*Exists path*
$s \models \mathbf{A}\ \phi$ if for every path π starting from s such that $\pi \models \phi$	*For all paths*
$\pi \models \mathbf{X}\ \phi$ if $\pi^1 \models \phi$ where π^1 is the suffix of π without its first state	*At next time point*
$\pi \models \mathbf{F}\ \phi$ if $\exists\ k \geq 0$ such that $\pi^k \models \phi$ where π^k is the k^{th} suffix of π	*At some time point*
$\pi \models \mathbf{G}\ \phi$ if $\forall\ k \geq 0$, $\pi^k \models \phi$	*At all time points*
$\pi \models \phi_1\ \mathbf{U}\ \phi_2$ if $\exists\ k \geq 0\ (\ \pi^k \models \phi_2 \wedge \forall\ j < k\ \pi^j \models \phi_1\ )$	*Until*
$\pi \models \phi_1\ \mathbf{R}\ \phi_2$ if $\forall\ k \geq 0\ (\ \pi^k \models \phi_2 \vee \exists\ j < k\ \pi^j \models \phi_1\ )$	*Release*

Duality: $\neg\ \mathbf{E}\phi = \mathbf{A} \neg\ \phi$, $\neg\ \mathbf{F}\phi = \mathbf{G} \neg\ \phi$, $\neg\ \mathbf{X}\phi = \mathbf{X} \neg\ \phi$, $\neg\ (\phi_1\ \mathbf{U}\ \phi_2) = \neg\ \phi_1\ \mathbf{R} \neg\ \phi_2$

CTL Fragment of CTL*

In CTL fragment, each temporal operator must be preceded by a path quantifier

Basis of three operators: **EX, EG, EU**

- **EF** ϕ = **E**(true **U** ϕ) $\quad s \models \mathbf{EF}\ \phi$ if $\exists\ \pi$ from s $\exists\ k \geq 0\ \ \pi^k \models \phi$
- **AX** $\phi = \neg\ \mathbf{EX} \neg\ \phi$ $\quad s \models \mathbf{AX}\ \phi$ if $\forall\ \pi$ from s $\ \pi^1 \models \phi$
- **AF** $\phi = \neg\ \mathbf{EG} \neg\ \phi$ $\quad s \models \mathbf{AF}\ \phi$ if $\forall\ \pi$ from s $\exists\ k \geq 0\ \ \pi^k \models \phi$
- **AG** $\phi = \neg\ \mathbf{EF} \neg\ \phi$ $\quad s \models \mathbf{AG}\ \phi$ if $\forall\ \pi$ from s $\forall\ k \geq 0, \pi^k \models \phi$
- Etc…

Any CTL formula is thus a state formula and can be identified to the set of states that satisfy it

$$\phi \simeq \{s \in S : s \models \phi\}$$ [Emerson 90]

Example in metabolism: EF(product) = {metabolites : present metabolites $\models$ EF(product) }

a symbolic model-checker returns a set (or the sets) of metabolites sufficient for the production

Biochemical Reachability Properties in CTL (from some initial state)

Initial state = initial biological conditions = molecules present / absent (/ undetermined)

- Can the cell produce some protein P (from initial state) ?
  - **EF(**P**)** ≜ **reachable(**P**)**
- Can the cell produce P, Q and not R?
  - **reachable(**P^Q^¬R**)**

About *pathways*:

- Can the cell reach a given set s of states while passing by another set of states s_2?
  - **EF(**s_2^**EF**s**)**
- Is it possible to produce P without Q before ?
  - **E(**¬Q **U** P**)**
- If not, this gives a *phenomenological non-causal notion of checkpoint*
  - ¬**E(**¬s_2 **U** s**)** ≜ **checkpoint(**s_2,s**)**

Cum hoc sed non propter
Correlation is not causality

Biochemical Reachability Properties in CTL (from some initial state)

- Is a given set of states `s` a stable state set *(infinite loop with no escaping possibility)*?
  - **`stable(s)`** $\triangleq$ **`AG(s)`**
- Is `s` a steady state *(infinite loop with escaping possibility)* ?
  - **`steady(s)`** $\triangleq$ **`EG(s)`**

- Can the cell reach a given stable state `s`?
  - **`reachable(stable(s))`**

  alternance of path quantifiers **`EF AG`** ϕ *(not expressible in LTL)*

- Must the cell reach a given stable state `s`?
  - **`AF(stable(s))`**

- What are the stable states?
  - *Not expressible in CTL.*

  needs to combine CTL with enumeration, see Biocham `generate_ctl(stable(s))`

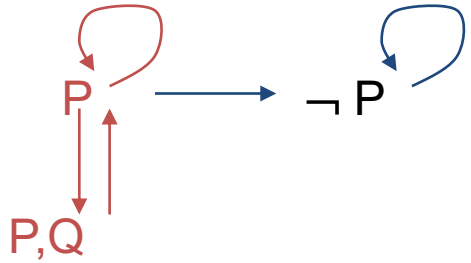
Oscillation Properties in CTL*/CTL

CTL*: **EG((F** ¬P**)** ^ **(F** P**))** expresses possibility of oscillation but is not in CTL

CTL: **EG((EF** ¬P**)** ^ **(EF** P**))** provides a (weaker) necessary condition for oscillation

- *not sufficient condition* for oscillations without fairness:

- also with weak fairness (no rule stays continuously fireable without being fired)



- Needs strong fairness (no rule is infinitely often fireable without being fired)

Basic CTL Model-Checking Algorithm

Dynamic programming algorithm for computing the set of states satisfying a CTL formula:

$\{s \in K : s \models \phi\}$ in a *finite* Kripke structure K.

Represent K explicitly by the finite graph of all state transitions

and iteratively label the nodes with the *subformulas of* ϕ that are true in that node:

- Add proposition ϕ to the states satisfying ϕ
- Add **EX** ϕ to the immediate predecessors of the states labeled by ϕ
- Add **EF** ϕ to all the predecessors of the states labeled by ϕ
- Add **E**($\phi 1$ **U** $\phi 2$) to the predecessor states of $\phi 2$ while they satisfy $\phi 1$
- Add **EG** ϕ to the states of the subgraph satisfying ϕ which are on a path leading to a non trivial (i.e. containing at least one edge) strongly connected component.

Space and time in $O(|K|*|\phi|)$,

CTL model-checking is Ptime-complete in non-succinct (explicit) representation of K

Example

Apply the previous basic model-checking algorithm to show

$P \rightarrow \neg P \models \mathbf{EG((EF}\ \neg P\mathbf{)\ \hat{}\ (EF}\ P\mathbf{))}$

Subformulae:

P	1	0
¬ P	0	1
EF P	1	0
EF ¬P	1	1
(EF ¬P**) ^ (EF** P**)**	1	0
EG((EF ¬P**) ^ (EF** P**))**	1	0

Symbolic CTL Model-Checking Algorithm

Represent **a set of states** by a symbolic **boolean constraint c(V)** over **state variables V**

e.g. $p \vee \neg q$ represents the set of all states where p is present and q absent

Represent the **transition relation** by a **boolean constraint r(V,V')** on twice state variables

e.g. the constraint $p \vee (\neg p \wedge \neg p')$ represents the transition graph $p \rightarrow \neg p$

Represent CTL operators by **state constraint transformers**

e.g. $[EX(c(V))] = \exists V'\ r(V,V') \wedge c(V') \triangleq ex(c(V))$

constraint of having one immediate successor r(V,V') satisfying c(V')

e.g. $[AX(c(V))] = \forall V'\ r(V,V') \Rightarrow c(V') \triangleq ax(c(V))$

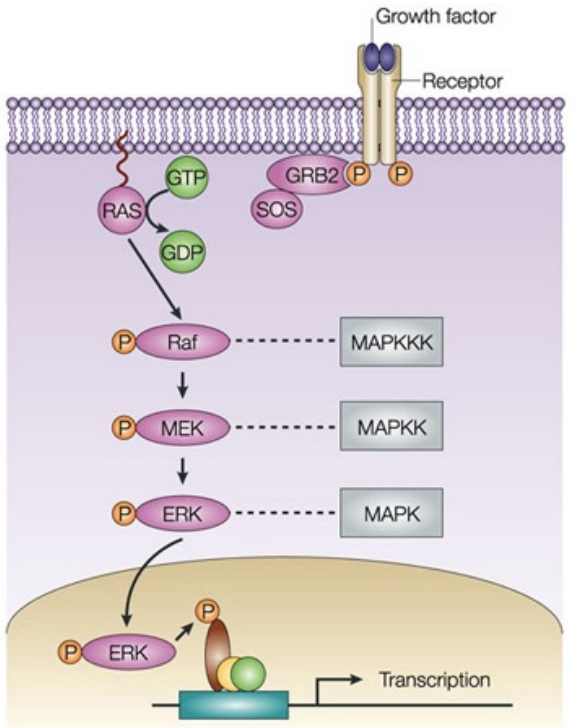
constraint of having all immediate successors r(V,V') satisfying c(V')

Returns *Boolean state constraints* to satisfy an input CTL formula

TP Chemical Signalling

http://lifeware.inria.fr/biocham4/online/notebooks/C2-19-Biochemical-Programming/22ctl.ipynb

```
In [24]: expand_ctl(checkpoint(A, B)).

Out[24]: not EU(not A,B)

In [25]: expand_ctl(checkpoint2(A, B)).

Out[25]: not B/\EF(B)/\not EU(not A,B)

In [26]: check_ctl(query:checkpoint2(KKpp,Kpp)).

Out[26]: Trace:
         present({E1, KKK, E2, KK, KKPase, KKpp, K, KPase, Kpp}).

         checkpoint2(KKpp,Kpp) is false

In [27]: check_ctl(query:checkpoint(KKpp,Kpp), boolean_initial_states: some).

Out[27]: Trace:
         present({E1, KKK, E2, KK, KKPase, KKpp, K, KPase}).

         checkpoint(KKpp,Kpp) is true
```



Logical Paradigm for Systems Biology

Use of model-checking algorithms [Lincoln et al. 2002] [Chabrier Fages 2003] [Bernot et al. 2004]…

Biological process model = State Transition System K

Biological property = Temporal Logic Formula φ

Model validation = model-checking: K, s $\models$? φ

Model reduction = model-checking: K'?$\subset$K K', s $\models \varphi$

Static experiment design = model-checking: K, s? $\models \varphi$

Model behaviors = enumeration of true formulae: K, s $\models \varphi$?

Model Inference, dyn. exp. design = constraint solving: K?, s? $\models \varphi$

Generalizations to quantitative temporal logics

- FO-LTL(R_{lin}) [Rizk, Batt, F, Soliman 09] STL [Donze Maler 12] parameter search, robustness
- SAT modulo ODE [Gao Clarke 2012] formal verification on parameter range
- CRN synthesis: K?, s? $\models$ reachable(stable(y $\approx \frac{x^4}{c+x^4}$)

First-Order Linear Time Logic FO-LTL($\mathbb{R}_{lin}$)

1. FO-LTL($\mathbb{R}$) closed formulae
   - Syntax and semantics on a trace
   - Model-checking algorithm, parameter search by scanning
2. FO-LTL($\mathbb{R}_{lin}$) constraints with free variables
   - Syntax and validity domain semantics on a trace
   - Constraint Solving algorithm
3. FO-LTL($\mathbb{R}_{lin}$) continuous satisfaction degree
   - Parameter optimization by evolutionary algorithm
   - Robustness and parameter sensitivity estimation

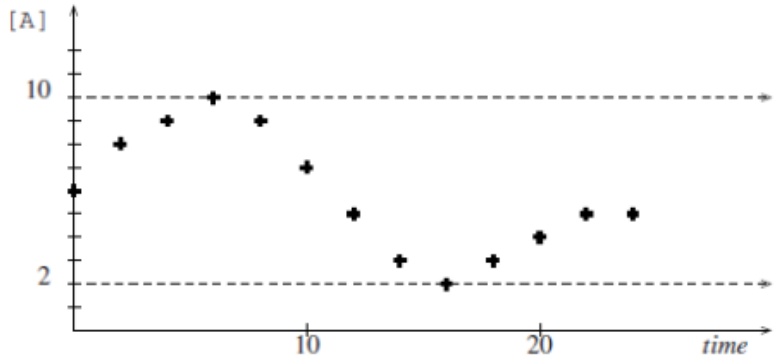
F. Fages, P. Traynard. Temporal Logic Modeling of Dynamical Behaviors: First-Order Patterns and Solvers. In Logical Modeling of Biological Systems, pages 291-323. John Wiley Sons, Inc., 2014.

A. Rizk, G. Batt, F. Fages, S. Soliman. Continuous Valuations of Temporal Logic Specifications with applications to Parameter Optimization and Robustness Measures. Theoretical Computer Science, 412(26):2827–2839, 2011.

Aurélien Rizk, Grégory Batt, François Fages, Sylvain Soliman. A general computational method for robustness analysis with applications to synthetic gene networks. Bioinformatics, 12(25):il69–il78, 2009.

F. Fages, A. Rizk. On Temporal Logic Constraint Solving for the Analysis of Numerical Data Time series. Theoretical Computer Science, 408(1):55–65, 2008.

Interpretation of FO-LTL($\mathbb{R}$) Formulae over Finite Traces



Trace (experiment or simulation):

State variables: time, concentrations A.
Closed arithmetic propositions over state variables (no free variable)
Temporal operators: **X**, **F**, **G**, **U**, **R**

Minimum threshold reachability: $\mathbf{F}([A] > 0.2)$
Minimum threshold stability: $\mathbf{G}([A] > 0.2)$
Reachability of stable state: $\mathbf{FG}([A] > 0.2)$
Curve fitting:
$\mathbf{F}(Time == 1 \wedge [M] == 0.05 \wedge \ \mathbf{F}(Time == 2 \wedge [M] == 0.12 \wedge [M] == 0.12 \wedge \ \ldots)))$

FO-LTL($\mathbb{R}$) Verification Algorithm

Input: A finite trace π and a FO-LTL($\mathbb{R}$) formula ϕ

Output: whether or not $\pi \models \phi$

1. Complete the trace with a loop on the last state
2. Iteratively label the states with the sub-formulae of ϕ that are true:
   - Add state proposition labels to the states where they are true
   - Label $\mathbf{X}\phi$ the immediate predecessor of any state labeled by ϕ,
   - Label ϕ **U** ψ the predecessors of any state labelled by ψ while they satisfy ϕ,
   - Label ϕ **R** ψ
     - the last state if it is labelled by ψ,
     - the states labelled by ϕ and ψ
     - their successors,
     - and their predecessors while ψ holds
3. Return true if the initial state is labelled by ϕ, and false otherwise

Parameter Scanning for Satisfying an FO-LTL($\mathbb{R}$) Formula

input: a parametric CRN $R(\boldsymbol{k})$ with
n kinetic parameters $\boldsymbol{k}$ given with range $[\underline{k}_i, \overline{k}_i]$, step size s_i
and an FO-LTL($\mathbb{R}$) formula ϕ to satisfy over time horizon T

output: (fail or) parameter values $\boldsymbol{v}$ such that
$\pi(\boldsymbol{v}) \models \phi$ where $\pi(\boldsymbol{v})$ is a simulation trace of $R(\boldsymbol{v})$ up to time T

1. Scan the parameter value space $\Pi_1^n[\underline{k}_i, \overline{k}_i]$ with a fixed step size s_i for each parameter k_i
2. Test whether $\pi(\boldsymbol{v}) \models \phi$ by model checking
3. Return the first value set $\boldsymbol{v}$ which satisfies f

Exponential complexity in the number n of parameters $O((\frac{\overline{k}-\underline{k}}{s})^n)$

From "Generate and Test" to Guided Search

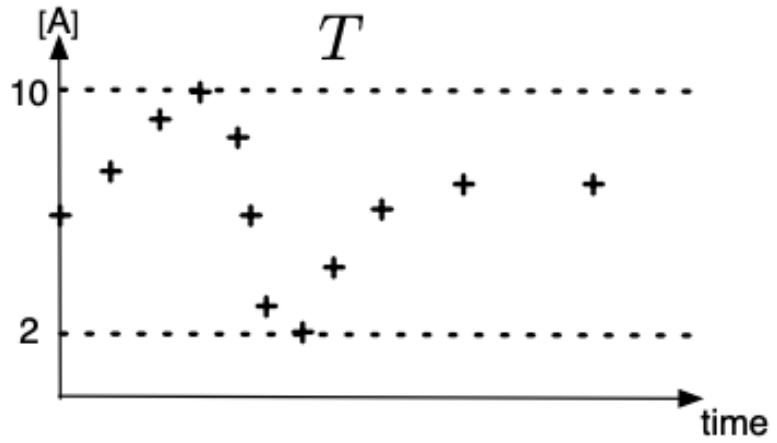
The **True/False** valuation of temporal logic formulae is **not well adapted** to several problems :

- parameter search, optimization and control of continuous models
- quantitative estimation of robustness
- parameter sensitivity analyses

$\rightarrow$ one would like to use the temporal logic formulae to guide the search
$\rightarrow$ need for a continuous degree of satisfaction of temporal logic formulae

How far is the system from verifying the specification ?

Model-Checking Generalized to Temporal Constraint Solving



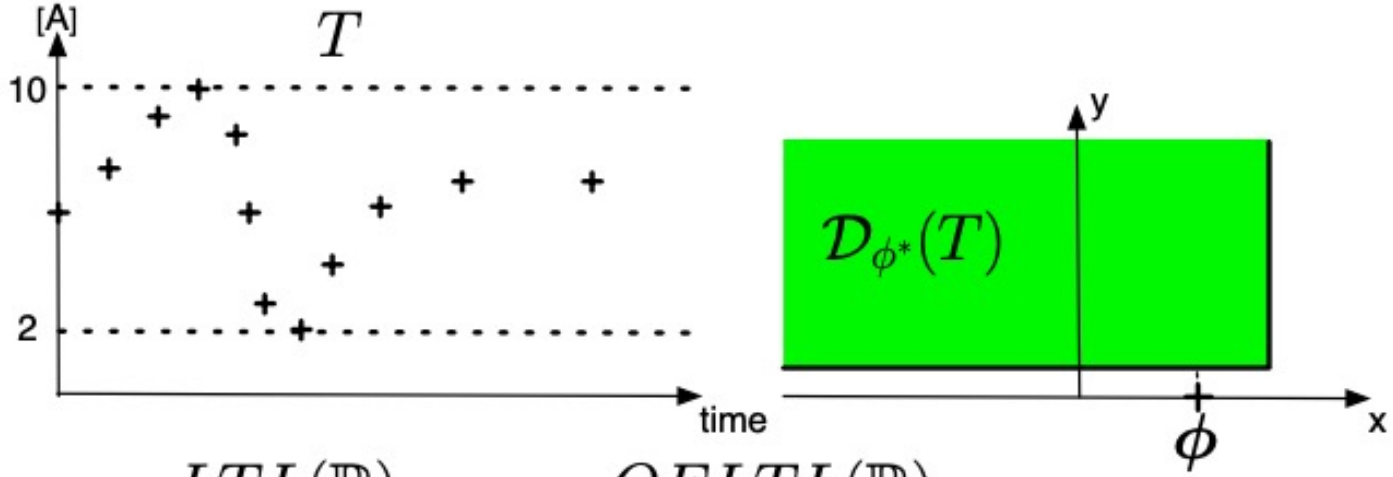
$$LTL(\mathbb{R})$$

Φ=F([A]≥7
∧F([A]≤0))

Model-checking

the formula is false

Model-Checking Generalized to Temporal Constraint Solving



$LTL(\mathbb{R})$

Φ=F([A]≥7 ∧F([A]≤0))

Model-checking

the formula is false vd=2 **sd=1/3**

$QFLTL(\mathbb{R})$

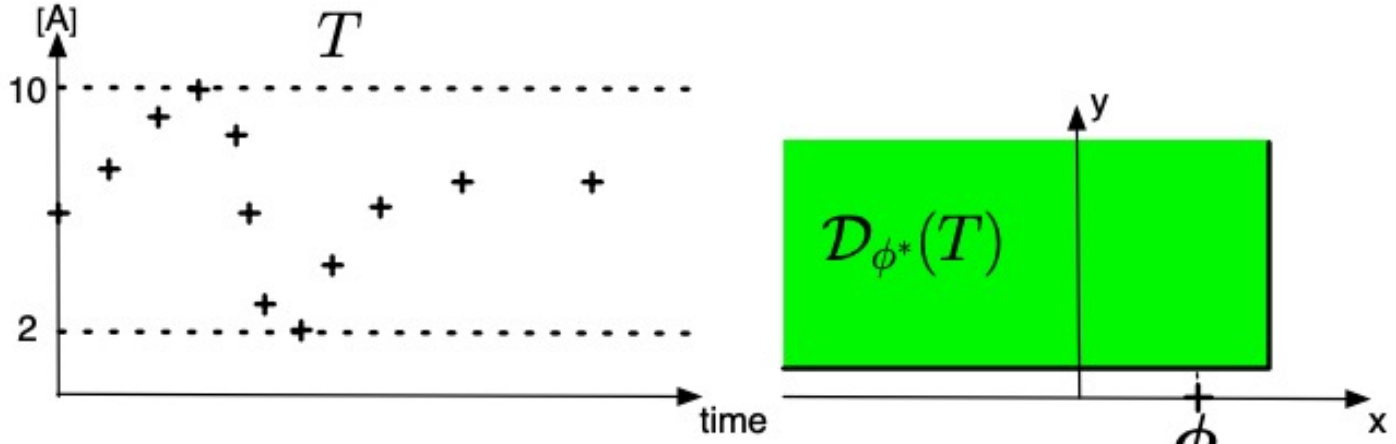
Φ*=F([A]≥x ∧F([A]≤y))

Constraint solving

the formula is true for any x≤10 ∧ y≥2

validity domain

Model-Checking Generalized to Temporal Constraint Solving



$LTL(\mathbb{R})$ | $QFLTL(\mathbb{R})$

Φ=F([A]≥7 ∧F([A]≤0))

Φ*=F([A]≥x ∧F([A]≤y))

Model-checking

the formula is false vd=2 **sd=1/3**

Constraint solving

the formula is true for any x≤10 ∧ y≥2

validity domain

Violation degree $vd(T,\phi) = \text{distance}(D_{\phi^*}(T), val(\phi))$ to objective values in ϕ

Continuous satisfaction degree $sd(T,\phi) = \frac{1}{1+vd(T,\phi)} \in [0,1]$

FO-LTL($\mathbb{R}_{lin}$) Constraints with Free Variables

- Free variables $x, y, \ldots$
- Linear constraints as atomic propositions
- Logical quantifiers $\forall x \; \exists y$
- Temporal operators: $\mathbf{X}$, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{U}$, $\mathbb{R}$

maximum(A,x): $\mathbf{G}(A <= x) \wedge \mathbf{F}(A >= x)$

local_maximum(A,x): $\mathbf{F}(A < x \wedge \mathbf{X}(A >= x \wedge \mathbf{X}(A <= x)))$

decrease(A,x): $A >= x \wedge \mathbf{X}(A <= x)$

decrease(A): $\exists x \; A >= x \wedge \mathbf{X}(A <= x)$

peak(A,x,t)): $A < x \wedge \mathbf{X}(A >= x \wedge \mathbf{X}(A <= x) \wedge \mathit{Time} = t)$

Validity Domains of Free Variables

FO-LTL($\mathbb{R}_{lin}$) formula $\phi(\boldsymbol{y})$ with free variables $\boldsymbol{y}$

The *validity domain* of $\phi(\boldsymbol{y})$ on a finite trace T is the set of values $\boldsymbol{x}$ where $\phi(\boldsymbol{x})$ holds: $\mathcal{D}_{T,\phi(\boldsymbol{y})} = \{\boldsymbol{x} \in \mathbb{R}^v \mid T \models \phi(\boldsymbol{x})\}$

Linear constraints over $\mathbb{R}$ define polyhedra

Validity domains $\mathcal{D}_{T,\phi(\boldsymbol{y})}$ are finite unions of polyhedra

- polyhedra for linear constraints,
- intersection for conjunction and **G**
- union for disjunction and **F**
- complementation for negation,
- projection for $\exists$

$\rightarrow$ Finite unions of polyhedra

BIOCHAM uses the Parma Polyhedral Library PPL

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{T,\phi}$ of the free variables of ϕ on a trace $T = (s_0, \ldots, s_n)$ is the vector $\mathcal{D}_{s_0,\phi}$ of least domains satisfying

$\mathcal{D}_{s_i,c(\boldsymbol{x})} = \{\boldsymbol{v} \in \mathbb{R}^k \mid s_i \models c[\boldsymbol{v}/\boldsymbol{x}]\}$ for a constraint $c(\boldsymbol{x})$,

$\mathcal{D}_{s_i,\phi\wedge\psi} = \mathcal{D}_{s_i,\phi} \cap \mathcal{D}_{s_i,\psi}$, and $\mathcal{D}_{s_i,\phi\vee\psi} = \mathcal{D}_{s_i,\phi} \cup \mathcal{D}_{s_i,\psi}$,

$\mathcal{D}_{s_i,\neg\phi} = \complement\, \mathcal{D}_{s_i,\phi}$,

$\mathcal{D}_{s_i,\exists x\phi} = \Pi_x \mathcal{D}_{s_i,\phi}$, and $\mathcal{D}_{s_i,\forall x\phi} = \mathcal{D}_{s_i,\neg\exists x\neg\phi}$,

$\mathcal{D}_{s_i,\mathbf{X}\phi} = \mathcal{D}_{s_{i+1},\phi}$ if $i < n$, and $\mathcal{D}_{s_n,\mathbf{X}\phi} = \mathcal{D}_{s_n,\phi}$,

$\mathcal{D}_{s_i,\mathbf{F}\phi} = \bigcup_{j=i}^{n} \mathcal{D}_{s_j,\phi}$, and $\mathcal{D}_{s_i,\mathbf{G}\phi} = \bigcap_{j=i}^{n} \mathcal{D}_{s_j,\phi}$,

$\mathcal{D}_{s_i,\phi\ \mathbf{U}\ \psi} = \bigcup_{j=i}^{n} (\mathcal{D}_{s_j,\psi} \cap \bigcap_{k=i}^{j-1} \mathcal{D}_{s_k,\phi})$.

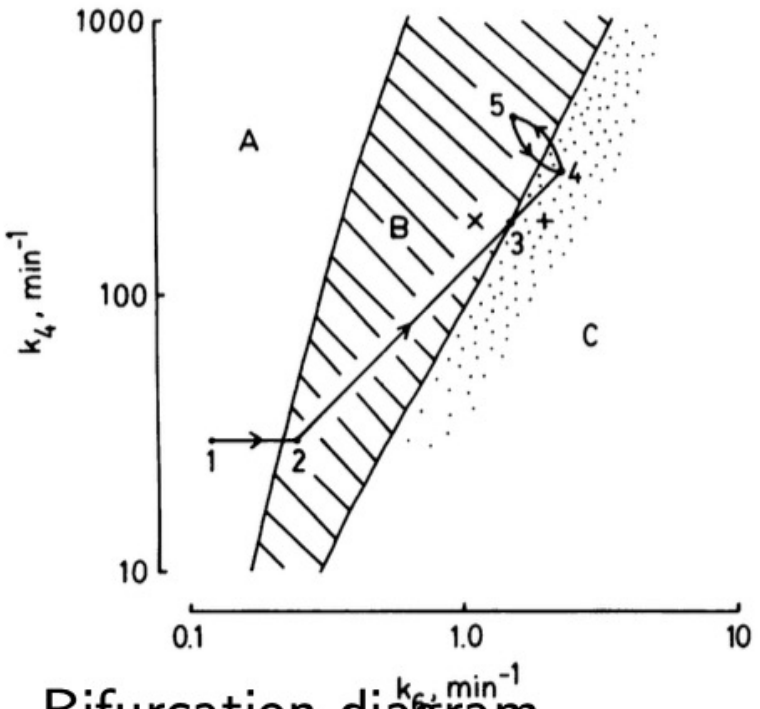
where $\complement$ is set complement over domains,
Π_x is domain projection out of x, restoring domain $\mathbb{R}$ for x.

Continuous Satisfaction Landscape of FO-LTL($\mathbb{R}_{lin}$) Objective

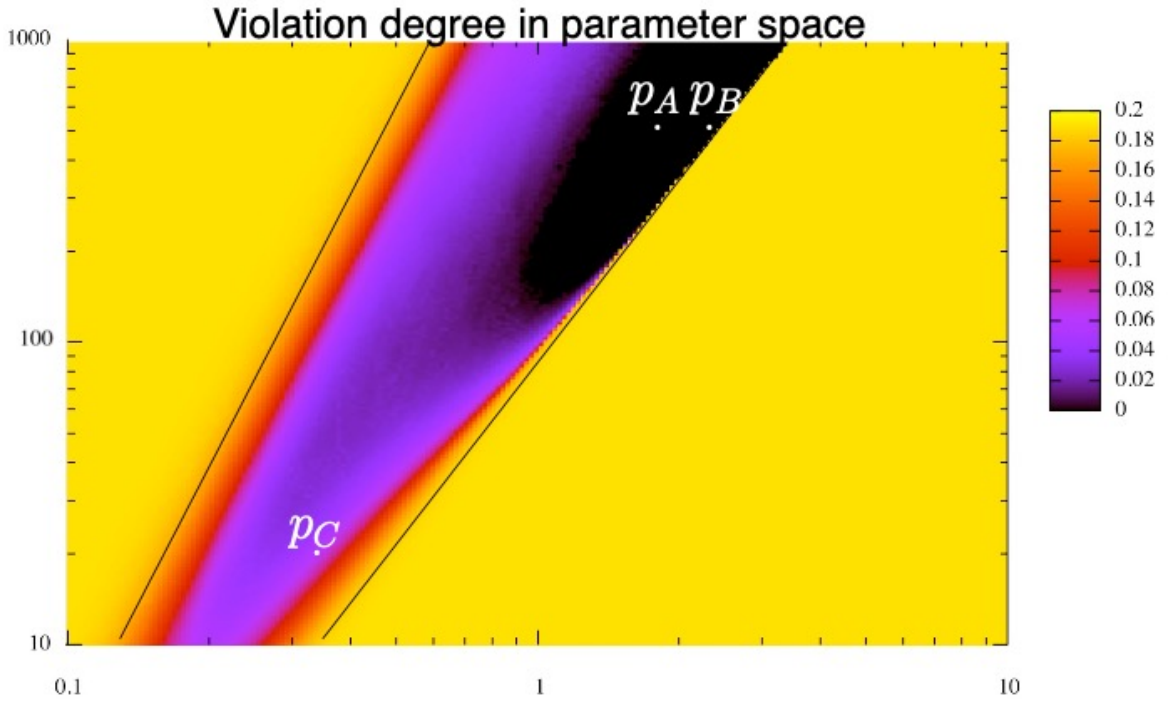
Example with :

- yeast cell cycle model [Tyson PNAS 91]
- oscillation of at least 0.3

ϕ^*: **F**([A]≥x) ∧ **F**([A]≤y); amplitude x-y≥0.3



Bifurcation diagram

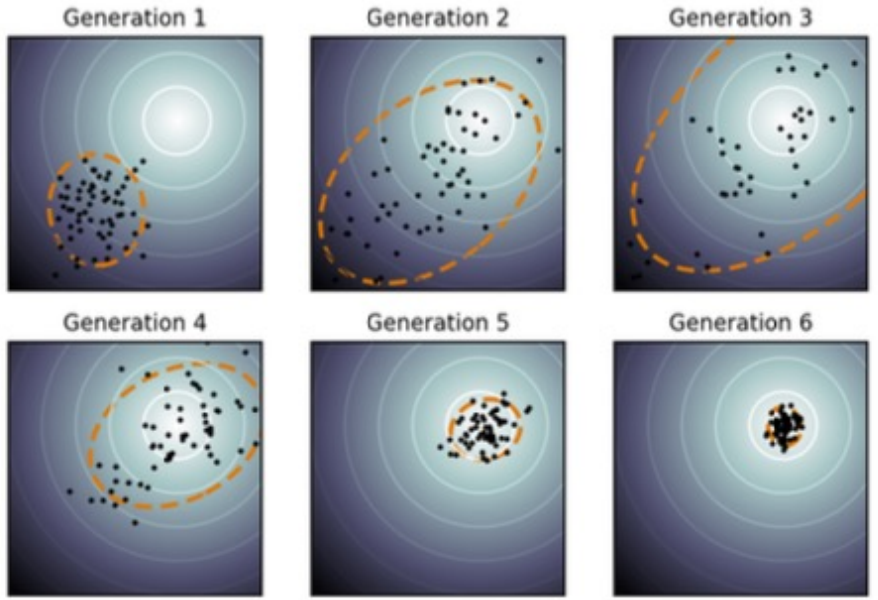


FO-LTL($\mathbb{R}_{lin}$) satisfaction diagram

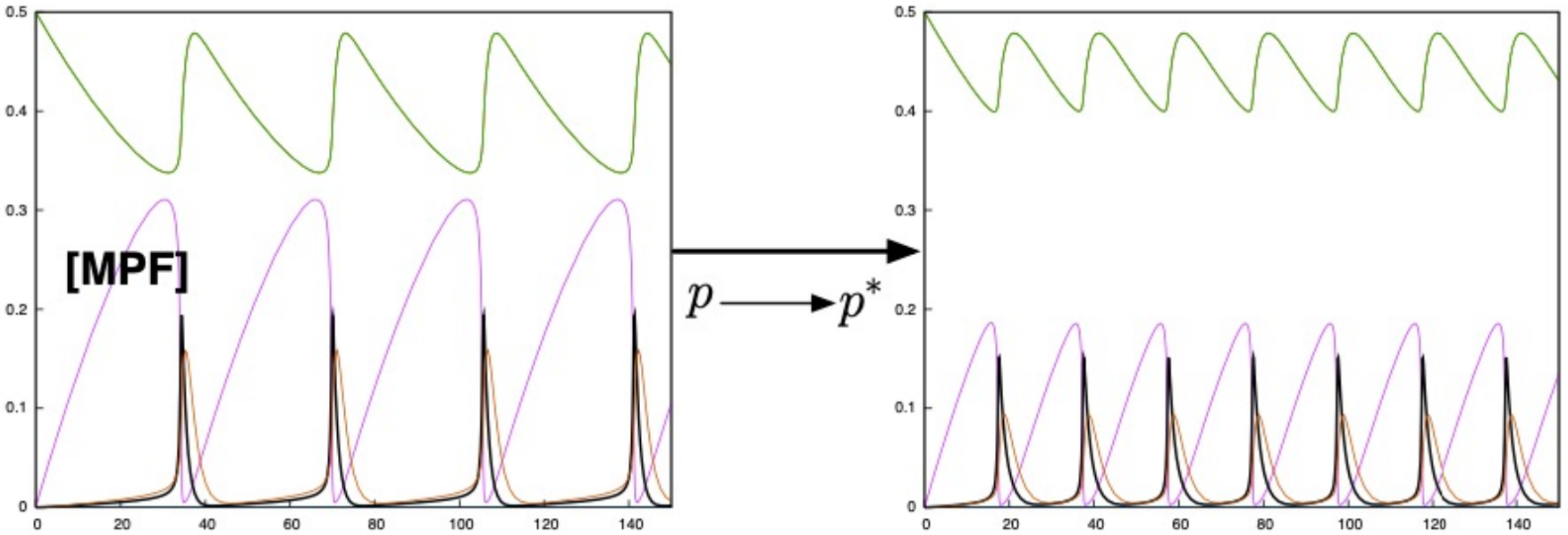
Covariance Matrix Adaptation Evolutionary Strategy

CMA-ES maximizes a black box fitness function (here $sd(\phi)$) in continuous domain (here parameter values) [Hansen Osermeier 01]

1. probabilistic neighborhood: multivariate normal distribution
2. estimation of covariance matrix by sampling (e.g. 50 best parameter set points from 100 random points simulations)
3. distribution (ellipsoid) update according to covariance matrix



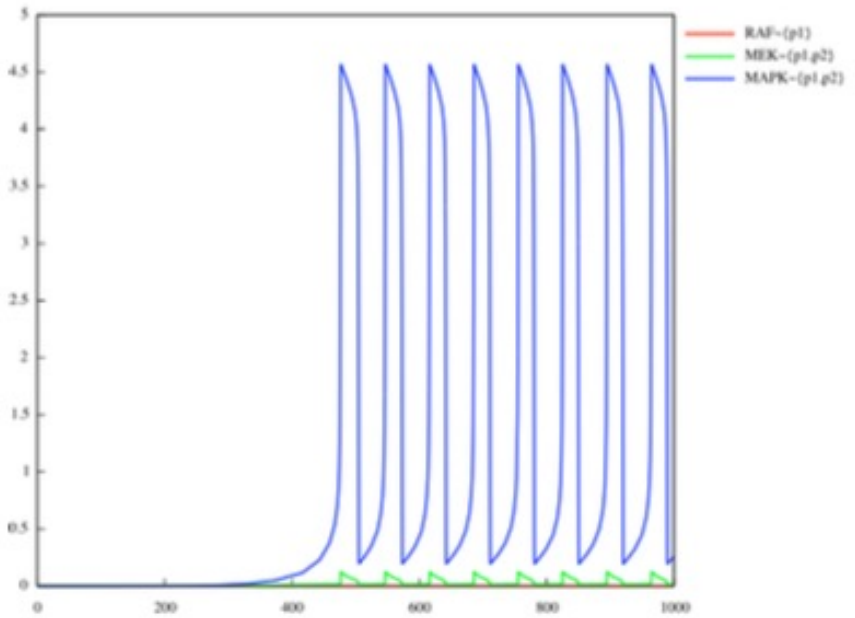
Parameter Optimization with Period Objective



- Pb : find values of 8 parameters such that period is 20
  formula:**F**(local_max(MPF) ∧Time=t1∧ **F**(local_max(MPF) ∧Time=t2)) ∧ z=t2-t1
  objective pseudo period: z=20
- Solution found after 60s (200 calls to the fitness function)

Oscillations in MAPK signal transduction cascade

- **MAPK** signaling model [Huang Ferrel PNAS 96]



- **search for oscillations** in **37 dimensions** (30 parameters and 7 initial conditions) Solution found after 3 min (200 calls to the fitness function)
- No negative feedback in the **reaction graph**, but negative circuit in the **influence graph** (necessary condition)

Robustness Measure Definition

Robustness defined with respect to :

- a biological system
- a functionality property D_a
- a set P of perturbations
- General notion of robustness proposed in [Kitano MSB 07]:

$$\mathcal{R}_{a,P} = \int_{p \in P} D_a(p)\ prob(p)\ dp$$

- Computational measure of robustness w.r.t. FO-LTL($\mathbb{R}_{\text{lin}}$) by sampling:

$$\mathcal{R}_{\phi,P} = \sum_{p \in P} sd(T(p), \phi)\ prob(p)$$

where $T(p)$ is the trace obtained by numerical integration of the ODE for perturbation p

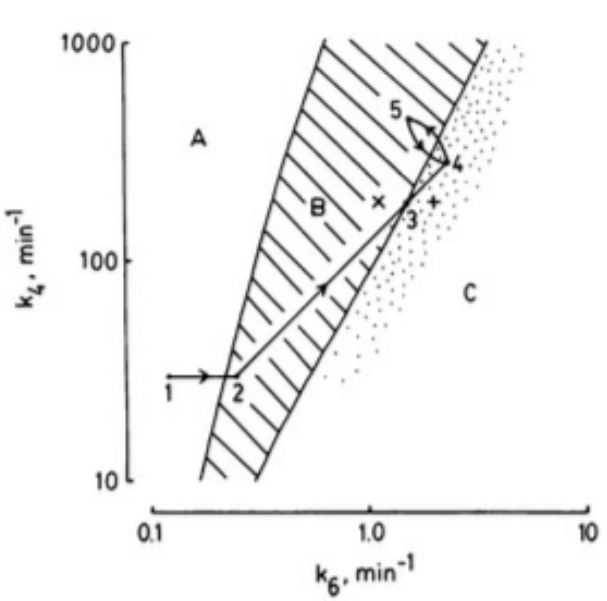
Robustness measure w.r.t parameter perturbations (extrinsic noise)

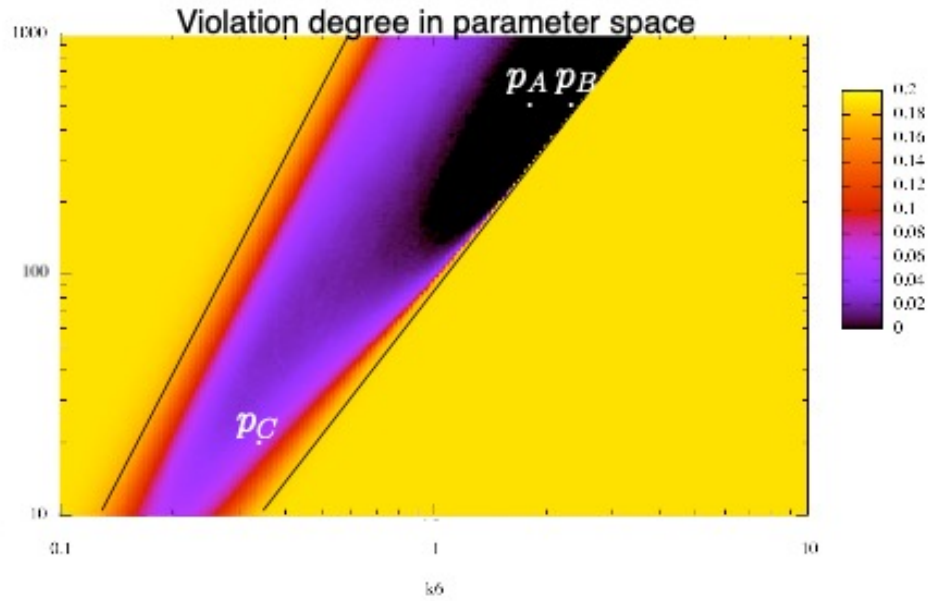
Example in cell cycle model [Tyson PNAS 91]

ϕ^*: $\mathbf{F}([A] \geq x) \wedge \mathbf{F}([A] \leq y) \wedge z \geq x - y$

amplitude objective $z = 0.2$

parameters normally distributed with coefficent of variation 0.2

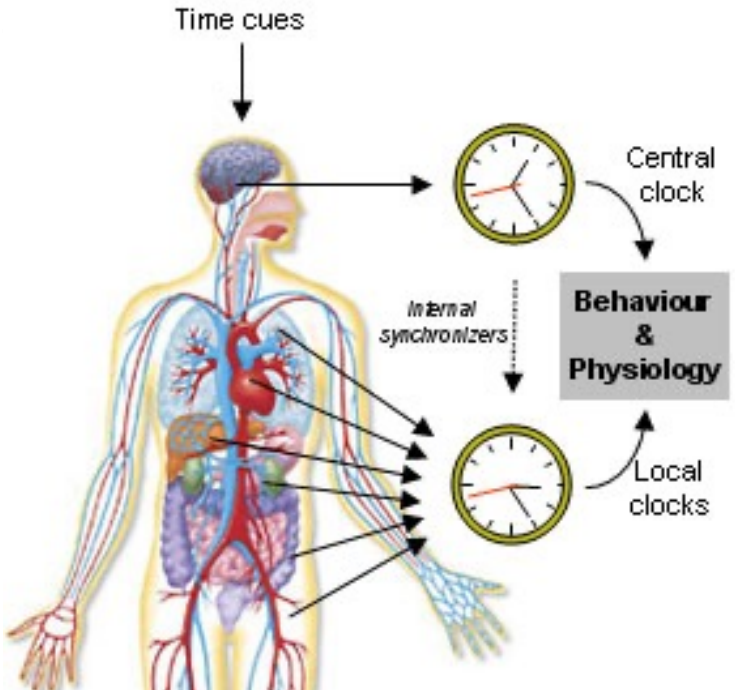


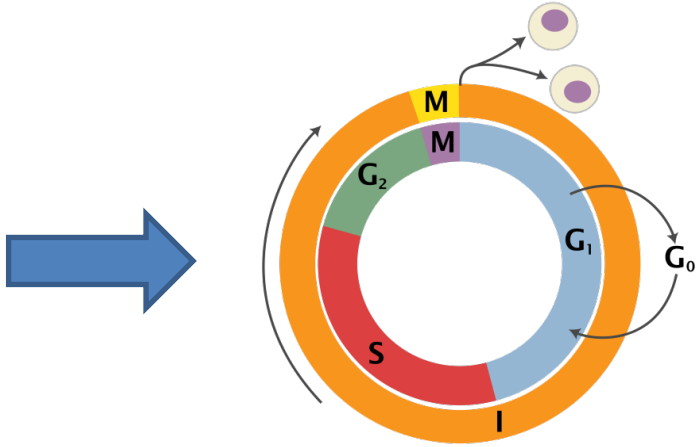


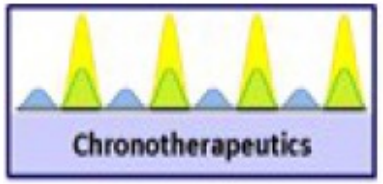
$$\mathcal{R}_{\phi,p_A} = 0.83,\ \mathcal{R}_{\phi,p_B} = 0.43,\ \mathcal{R}_{\phi,p_C} = 0.49$$

4. Case study on coupled modeling of the cell cycle and circadian clock

- Time gating for mitosis by effects of clock genes on cell cycle genes
  inhibition of Wee1 synthesis by Clock-Bmal1 [Matsuo et al 2003]
- Model-based predictions on conditions of entrainment [Calzone Soliman 2006] and period doubling (24h, 48h) phenomena [Gerard Goldbeter 2012]
- also repression of c-Myc by Clock-Bmal1 and inhibition of p21 by Reverb-α







Mormont MC, Levi F.
Cancer chronotherapy: principles, applications, and perspectives.
Cancer, 2003.

Formal Behavior Specification in Temporal Logic

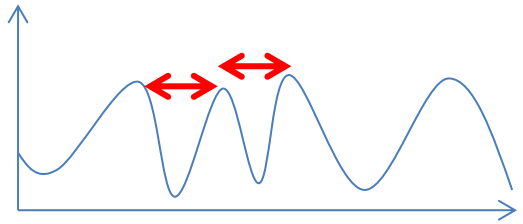
- Linear Time Temporal logic (LTL) extends classical logic with time operators **X**: next, **F**: finally, **G**: globally, **U**: until
  - Reachability of a stable set of states **FG**(s)

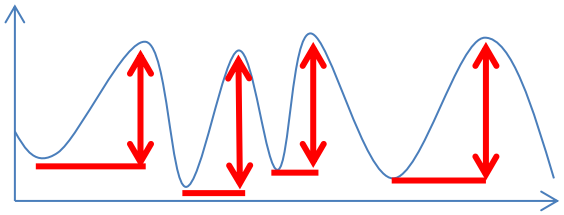
- First-order LTL with linear constraints, FO-LTL(R_{lin}), express quantitative properties about concentrations:

  - Reachability of threshold **F**(x>c)
  - Maximum value **G**(x<v)

  - Distance between successive peaks
  - Amplitude of next peak

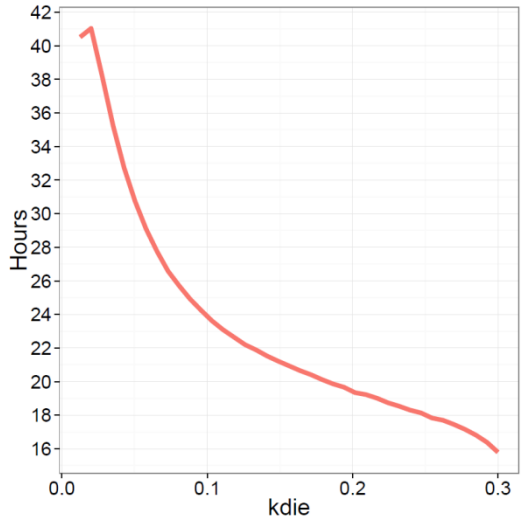
  - Period constraints
  - Phase constraints …



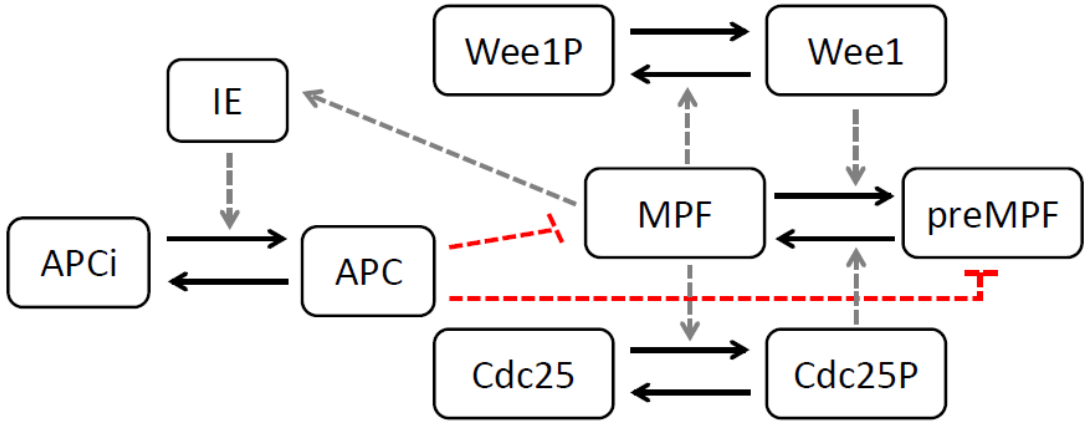


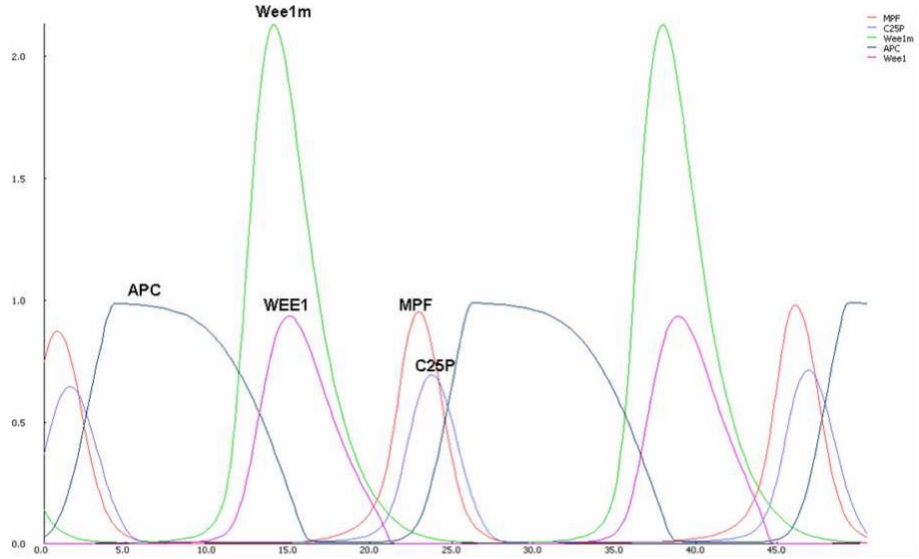
Cell Cycle Model [Qu-McLellan-Weiss Model 2003]

- Focus on G2/M phase
- 10 molecular species including Wee1
- 31 kinetic parameters



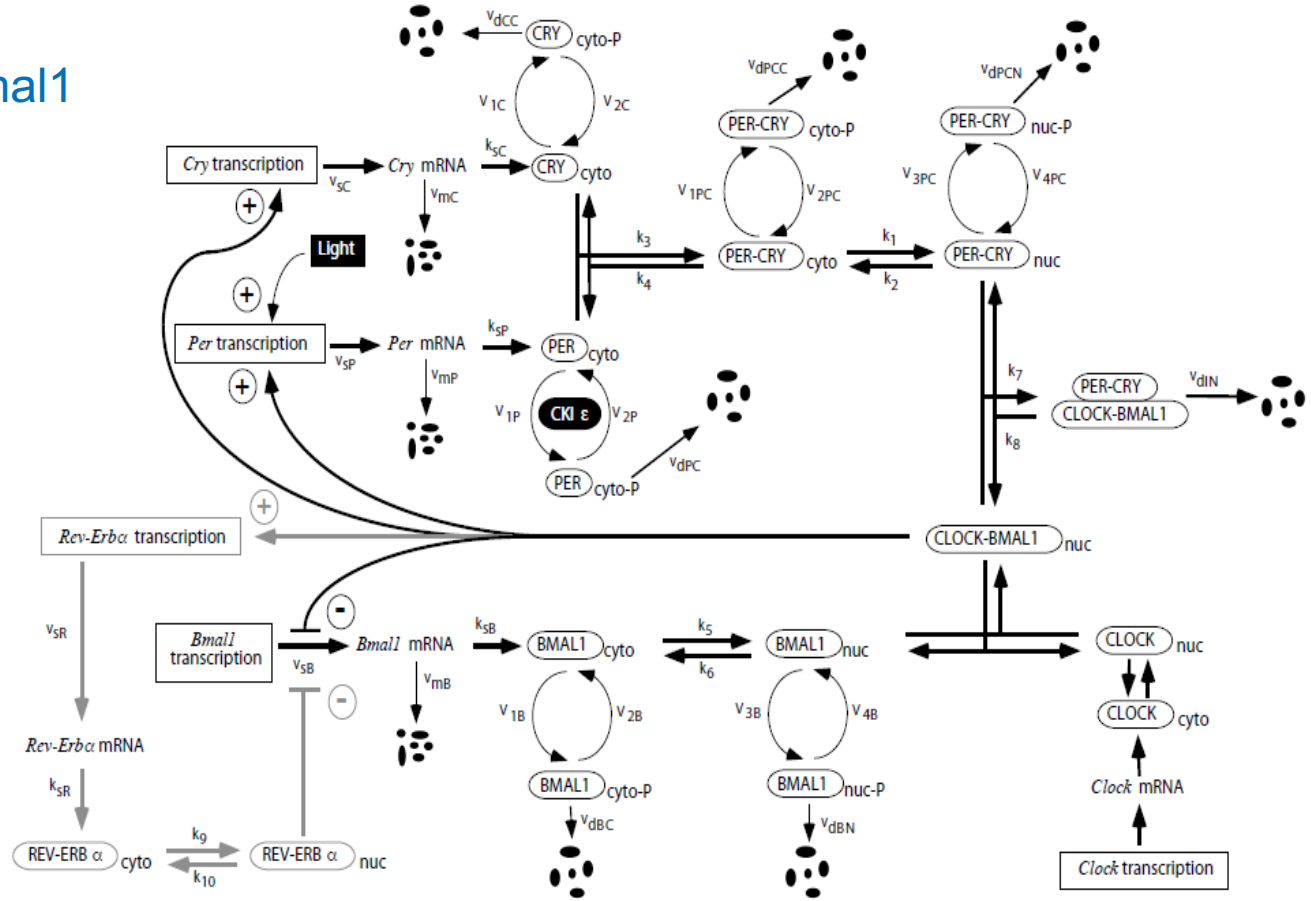
Variation of the cell cycle free period by *kdie* degradation rate constant (important in growing G1 phase)



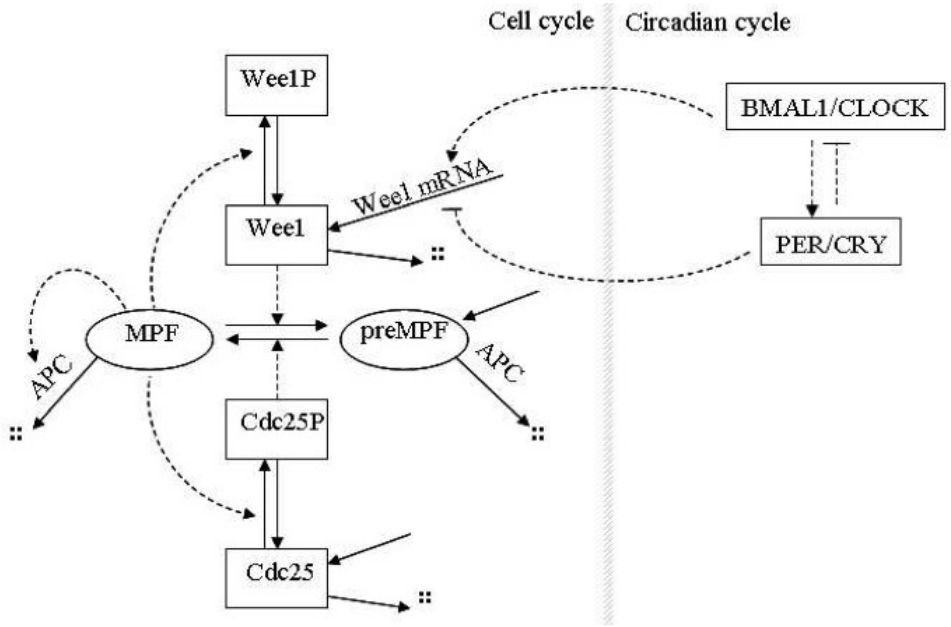


Circadian Clock Model [Leloup Goldbeter 03]

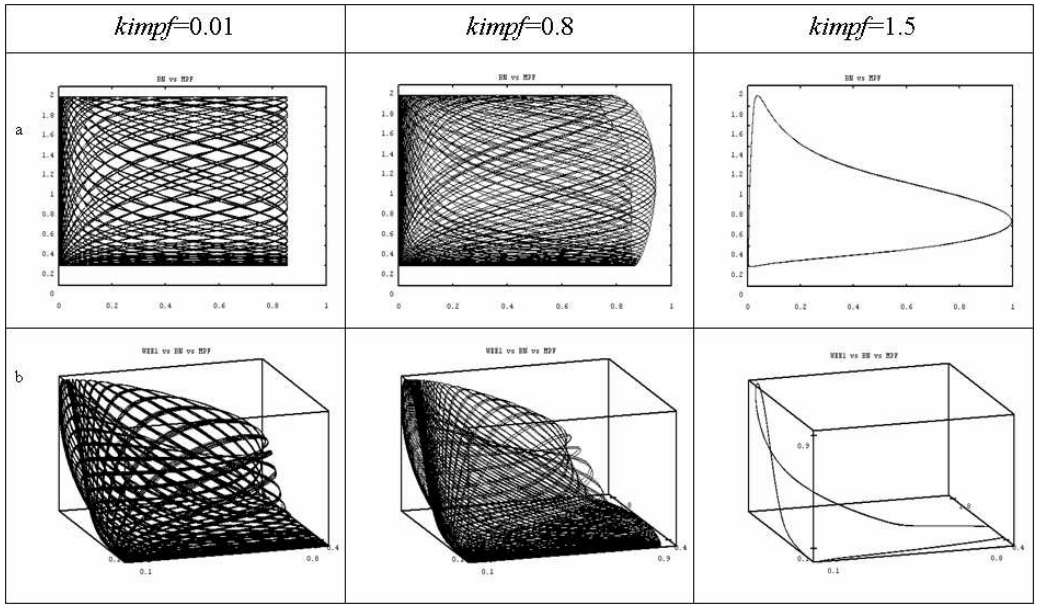
- 19 species, 70 parameters
- 4 genes: Per, Cry, Rev-erb α, Bmal1
- 2 negative feedback loops:
  - Per-Cry
  - Rev-erb α



Coupled Circadian Clock Model → Cell Cycle [Calzone Soliman 2006]



Time gating of mitosis hypothesis [Matsuo et al 2003]



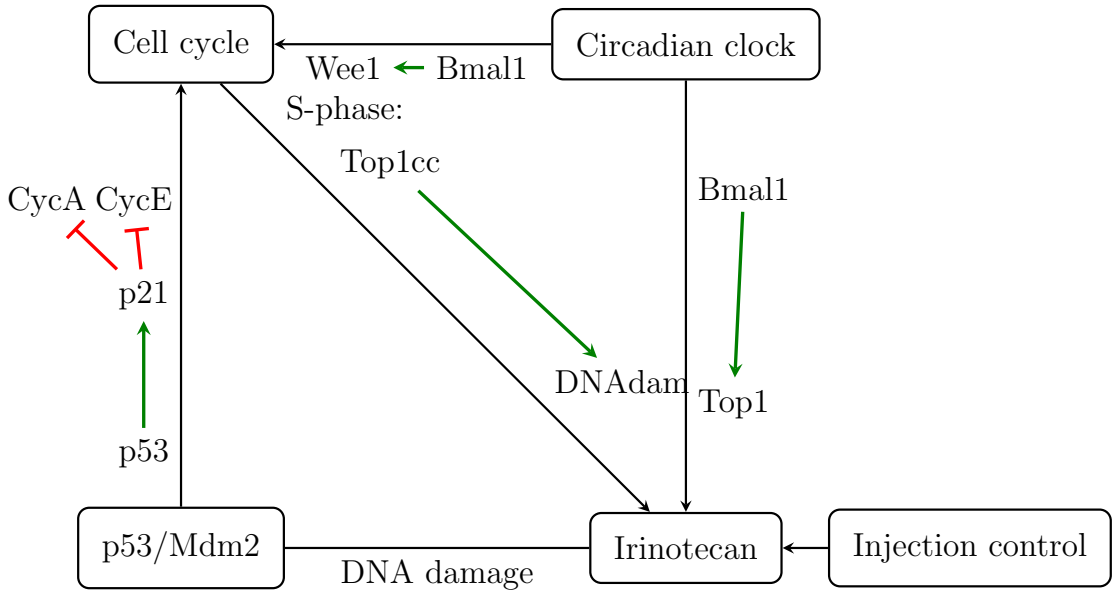
Entrainment conditions on parameter values

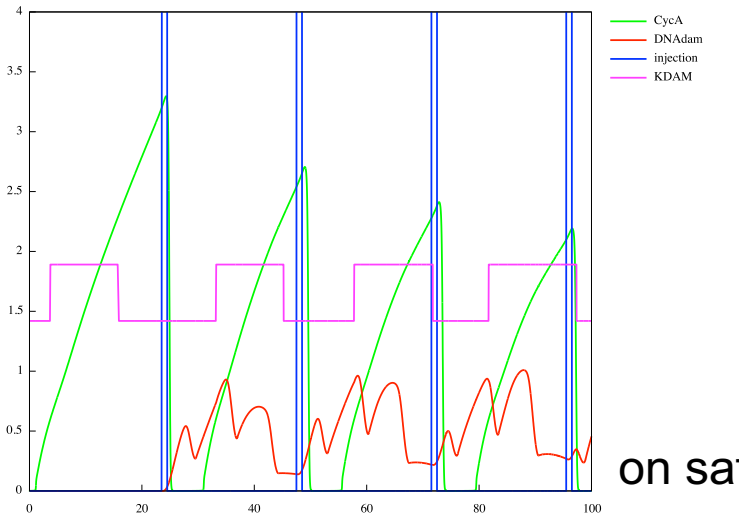
Coupling synthesis reaction of Wee1activated by Bmal1 repressed by Per-Cry:

```
(ksweemp+ksweem*[Bmal1])/(Kweem+kwpcn*[PC]) for _ => mWee1
```

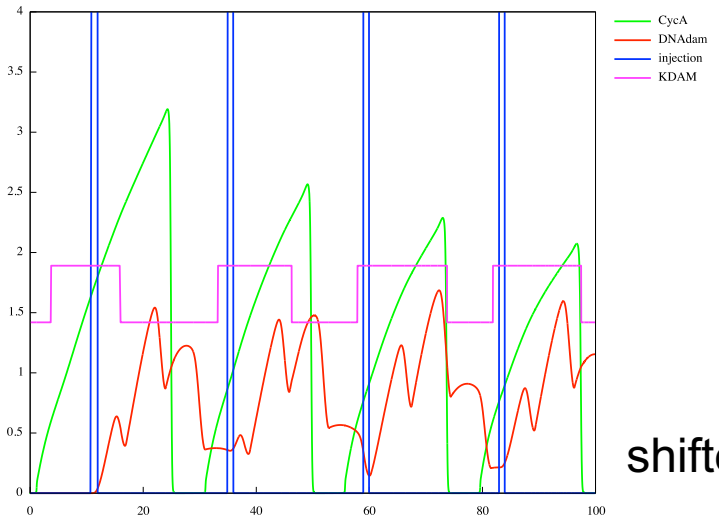
Irinotecan Exposure Chronotherapy Optimisation [De Maria F- Soliman *TCS* 2011]

Coupled cycle-clock-p53Mdm2-Irinotecan model





on safe cell



shifted phase

Optimal control of drug exposure

- max pulses satisfying always DNAdam<1 on safe cells
- and DNA damage>1 on phase shifted cells

Unexpected Acceleration of the Clock at high FBS in NIH3T3 Fibroblasts

Time series data in individual mice fibroblasts [Feillet Delaunay 2012]

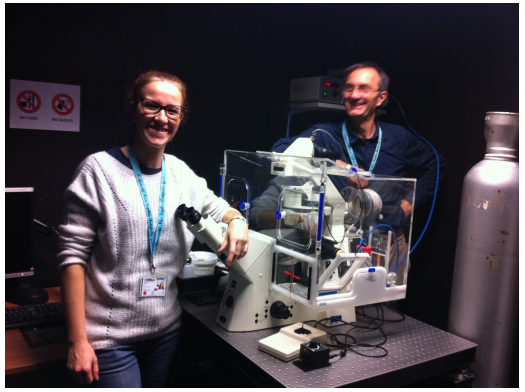
Fluorescent markers of the cell cycle and the circadian clock (RevErbα)

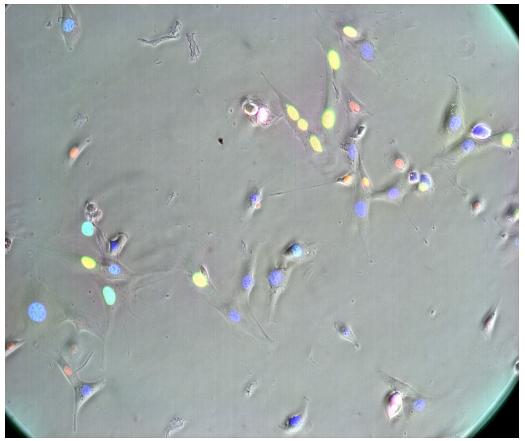
Medium with various concentrations of serum (FBS)

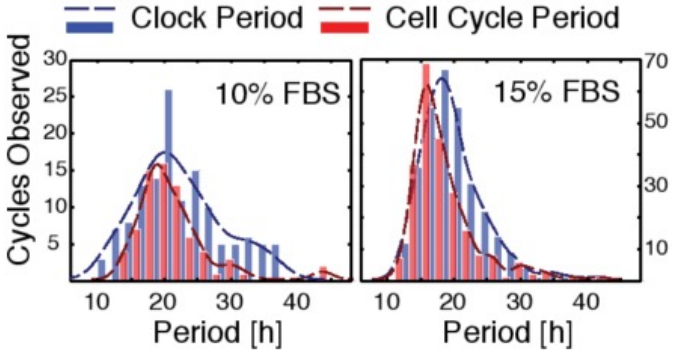
- FBS modulates the cell cycle frequency
- No observed time gating for mitosis
- But observed acceleration of the circadian clock
  in fastly dividing cells ! and not in confluent cells (24h)

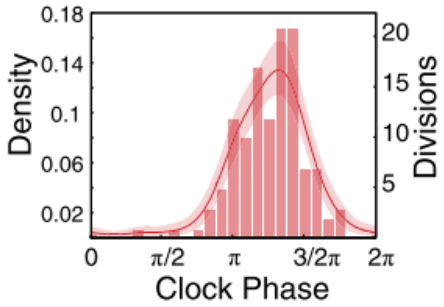
FBS 10% → Cell cycle 22h → Circadian clock 22h, phase 7h

FBS 15% → Cell cycle 19h → Circadian clock 18h, phase 7h





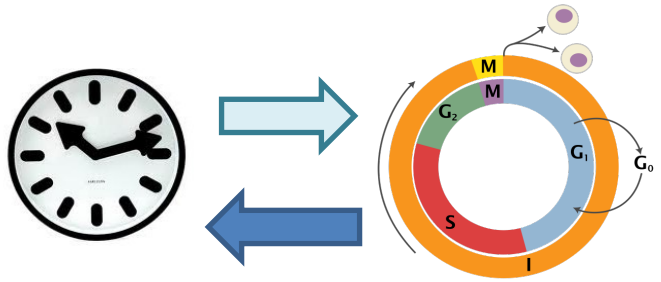




Statistical model 1:1, 5:4, 3:2 phase locking [Feillet et al Delaunay Rand PNAS 2014]

Mechanistic Model for Reverse Effect Cell Cycle → Clock

[Traynard, Feillet, Soliman, Delaunay, F., *Biosystems* 2016]



Hypothesis 1: Uniform inhibition of gene transcription during mitosis

- Entrainment in period
- No entrainment in phase

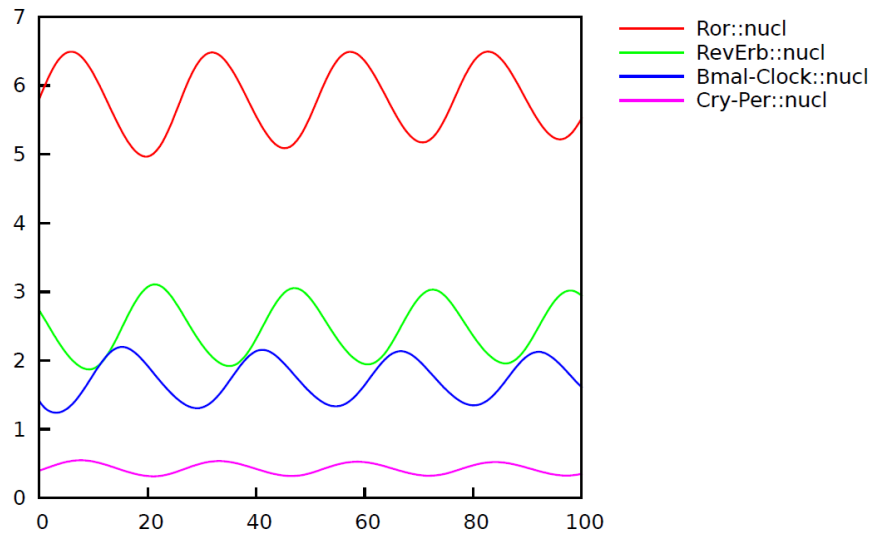
Hypothesis 2: Selective regulation of clock genes during mitosis

- Entrainment in period and phase fitted to experimental data
- Prediction of Reverb up-regulation or of Bmal1 down-regulation during mitosis
- Proposal of experiment at FBS 5% to discriminate between the two

Relogio-Herzel Model of the Circadian Clock (2011)

- 20 species, 71 parameters
- 60 parameters fitted to liver cell data
  - amplitude, period and phase data
- Per, Cry, Reverb, Ror, Bmal genes

Relógio, A., Westermark, P. O., Wallach, T., Schellenberg, K., Kramer, A., & Herzel, H. (2011). Tuning the mammalian circadian clock: robust synergy of two loops. PLoS Computational Biology.



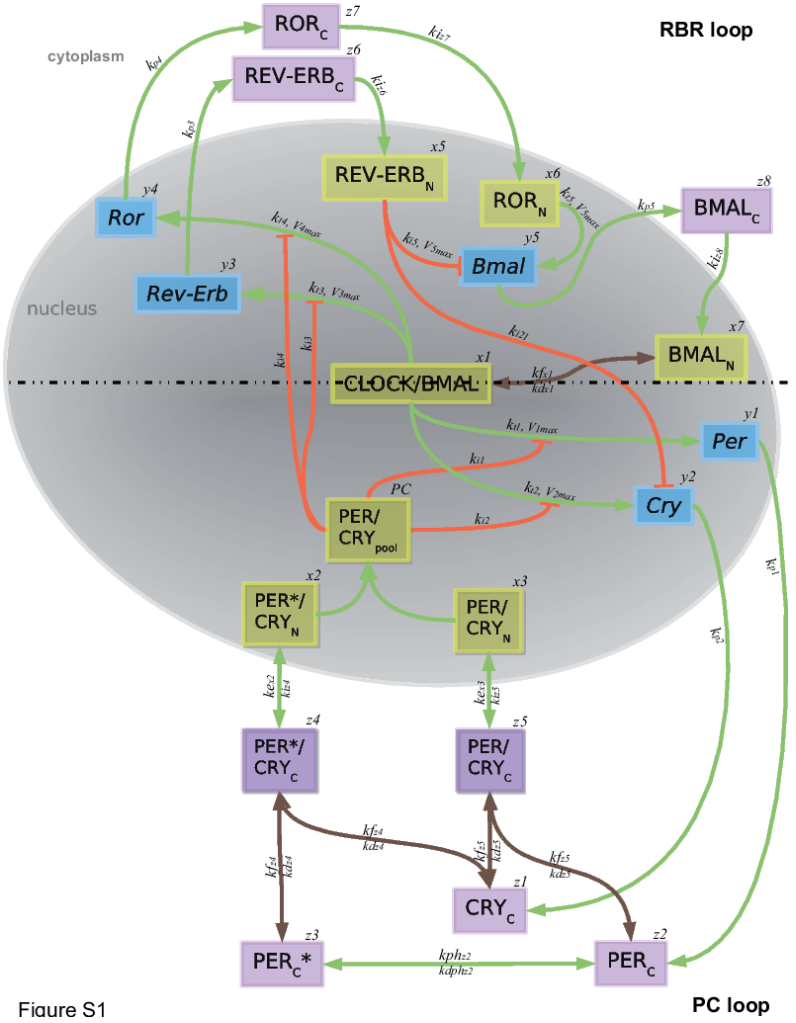
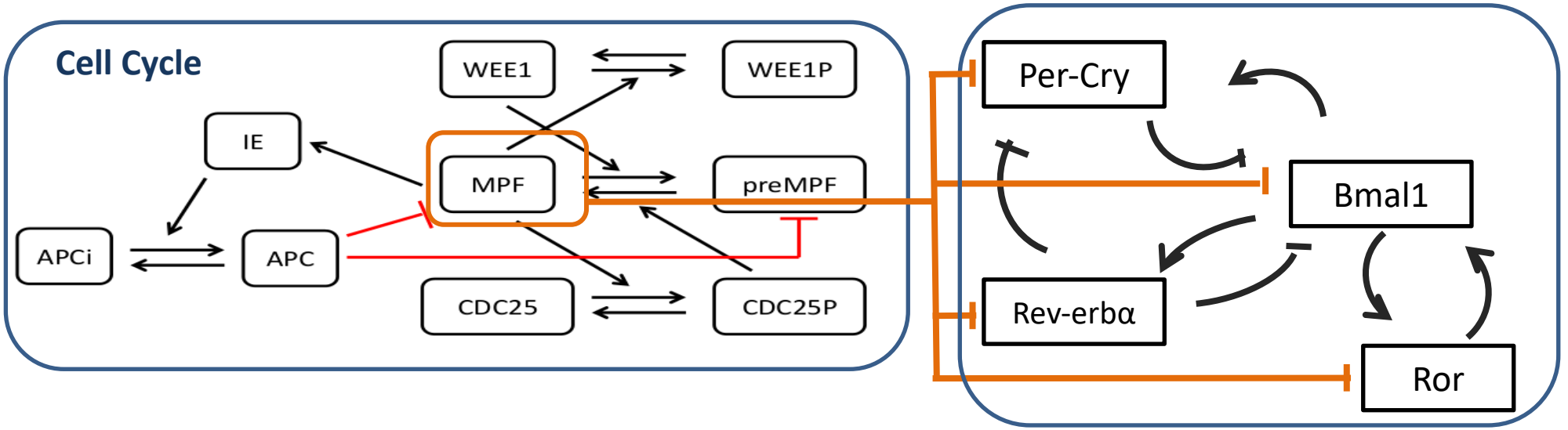
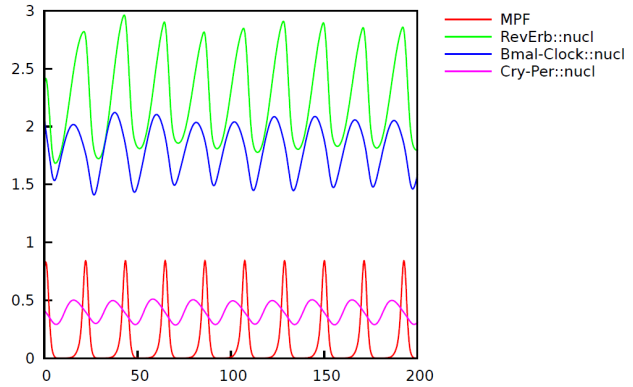


Figure S1

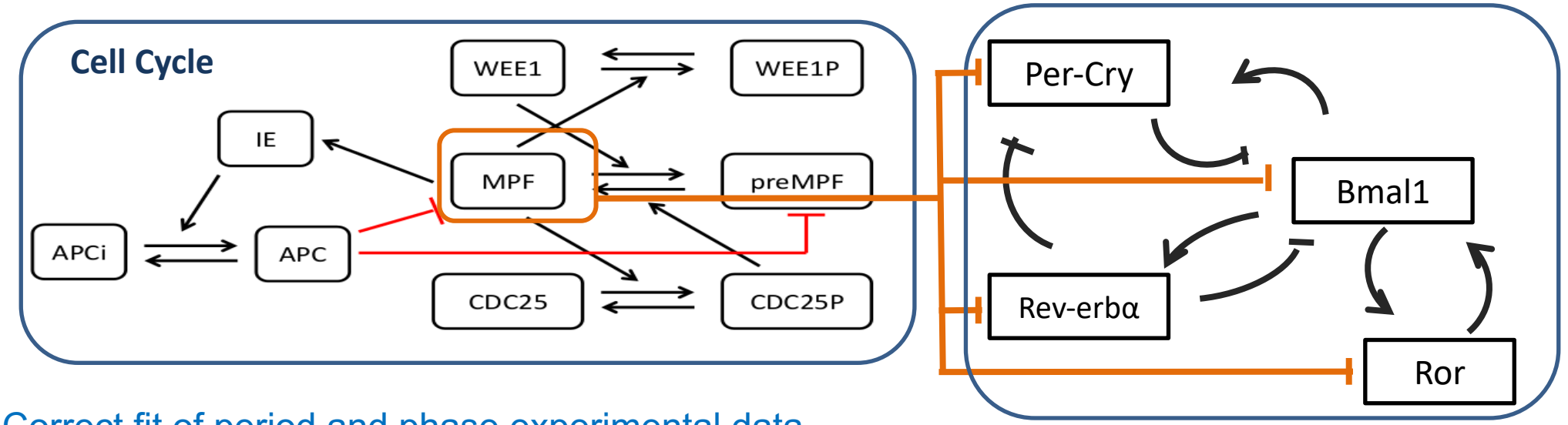
Hyp. 1: Uniform Inhibition of Transcription during Mitosis [Kang et al. 2008]



- Correct acceleration of both the cell cycle and the circadian clock
- But impossible to fit the observed phase shift
  between cell division time and RevErb peak
  - Experimental phase: 7h
  - Model phase: 18h



Hyp. 2: Selective Regulation of Clock Genes during Mitosis



Correct fit of period and phase experimental data
playing with only coupling strength regulation parameters

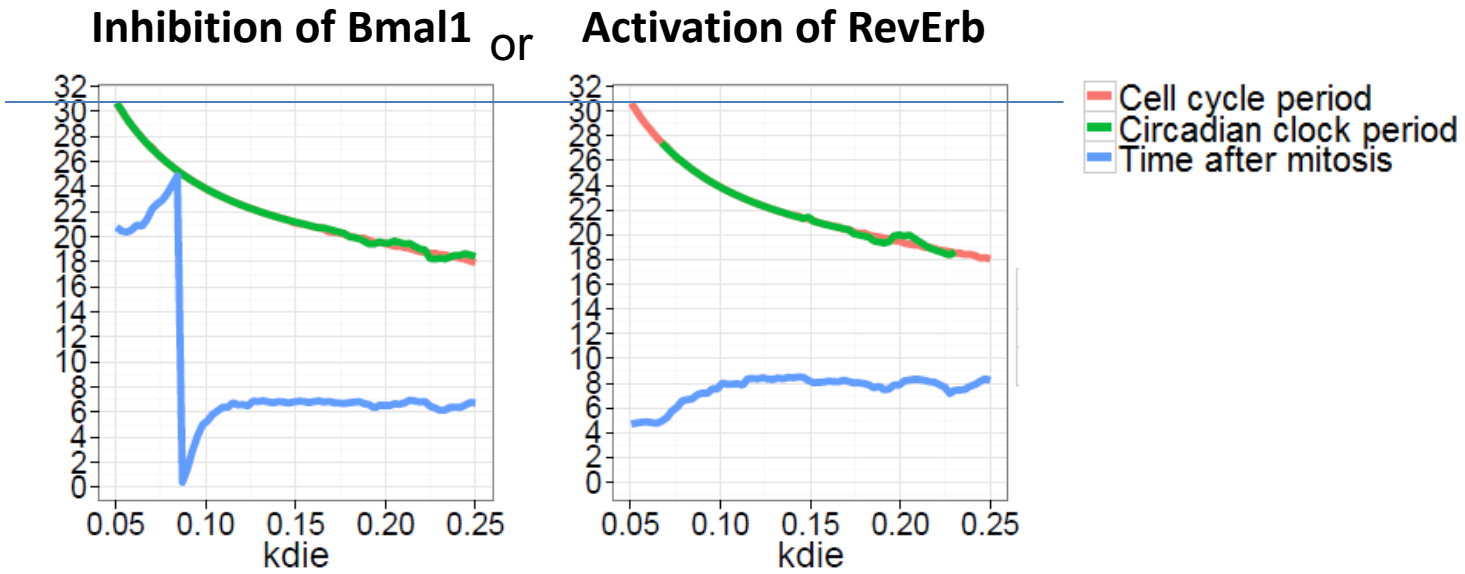
Two sets of parameter values fit the data:

- either down-regulation of Bmal1
- or up-regulation of RevErbα during mitosis

Parameters	First set	Second set
Synthesis coefficient for *Per*	0.66	2.40
Synthesis coefficient for *Cry*	2.30	0.67
Synthesis coefficient for *RevErb-α*	1.04	1.92
Synthesis coefficient for *Ror*	2.1	1.51
Synthesis coefficient for *Bmal1*	0	0.78
Duration	2.97h	2.81h

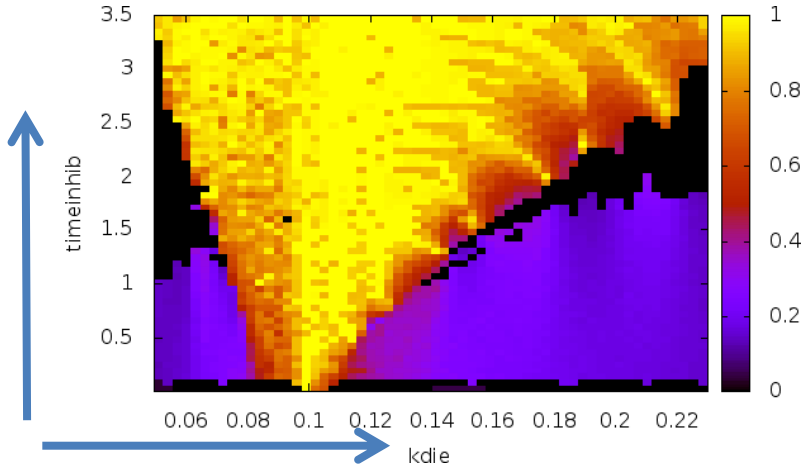
Hypothesis 2: Results and Predictions

Results:



Prediction: different behaviors for a slow cell cycle (5% FBS)

Stronger control of the clock by the divisions

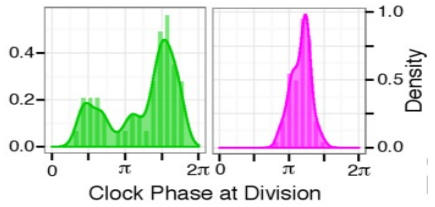


Score for the property:
The cell cycle and the circadian clock have the **same period**

Model Predictions for Treatment by Dexamethasone

Dexamethasone synchronize cellular clocks, but complex dynamics observed

Medium	Clock period	Division period	Mean delay
FBS 10%	24.2 h ± 0.5 h	20.1 h ± 0.94 h	10.7 h
FBS 20%	21.25 h ±0.36 h	19.5 h ±0.42 h	8.3 h
	29 h±1.05 h	16.05 h±0.48 h	6h/12h/22h

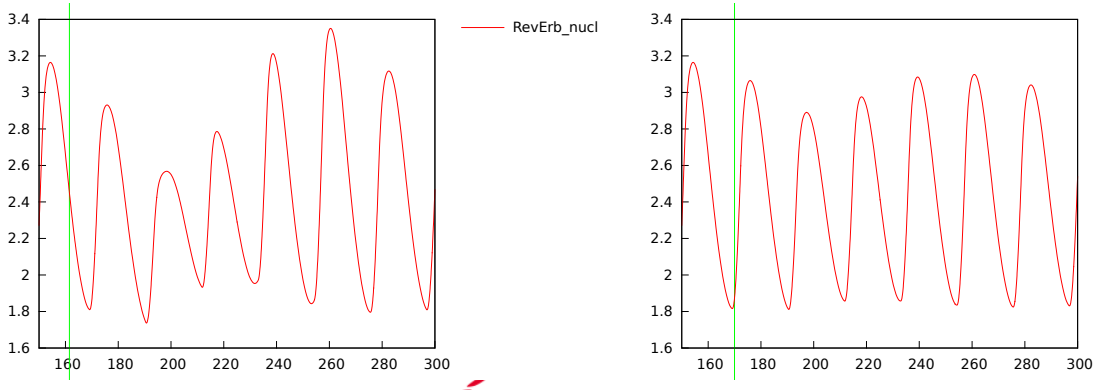


- Interpreted as 5:4 and 1:1 locking modes for 10% FBS and 3:2 and 1:1 for 15%

[C. Feillet et al. Phase locking and multiple oscillating attractors for the coupled mammalian clock and cell cycle., PNAS 2014]

- In our model, Dex pulse is modeled by induction of a high level of Per.
  - Clock perturbation varies according to the time T of the pulse
  - Stabilization of the clock may occur well after the 70h of observed data…

peak-peak distance in
[18.8, 22.7] with T=162h
[20.9, 21.7] with T=170h



Conclusion on CRN Modeling and Programming Language

Programming theory of biological processes can provide efficient

- Model building methods (modular, updates, testing, continuous integration, GitHub …)
- Model analysis methods (graph theoretic, abstract interpretation, model-checking, …)
- Before using classical mathematical analysis methods

High-level CRN modeling/programming language

- Hierarchy of semantics ODE, CTMC, Petri Net, Boolean
- Explicit graph structure allowing for efficient analyses
  - Model comparison in the large by subgraph epimorphism SEPI
  - Graphical conditions for ODE conservation laws (P-invariants), extreme fluxes (T-invariants), rate-independence
  - Graphical requirements for multistationarity in CRNs [Baudier F- Soliman. Journal of Theoretical Biology, 459:79–89, 2018]
- Used in  BIOCHAM modeling platform
  - together with temporal logic language to specify desired behaviors, verify and optimize them

Next lecture: abstract CRN synthesis to implement input/output functions and comparison to natural CRNs