On Rule-based Models of Dynamical Systems

François Fages

Project-Team Lifeware: computational systems biology and optimization
http://lifeware.inria.fr/

Institut National de Recherche en Informatique et Automatique
Inria Saclay

Outline of my Talk

1. Chemical reaction networks (as a programming language)
   - Syntax: reaction rules with well-formed kinetics
   - Semantics hierarchy: continuous ODE, stochastic CTMC, discrete PN, Boolean, hypergraphic

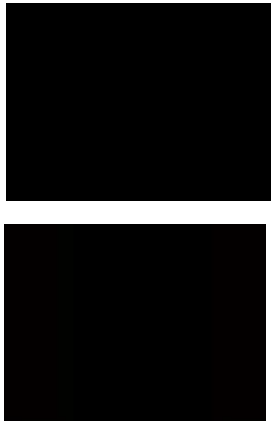
2. Static analyses
   - Structural ODE conservation laws as Petri net place-invariants
   - Relating CRN models by subgraph epimorphisms
   - Necessary conditions for multistationarity as positive circuits in the influence graph

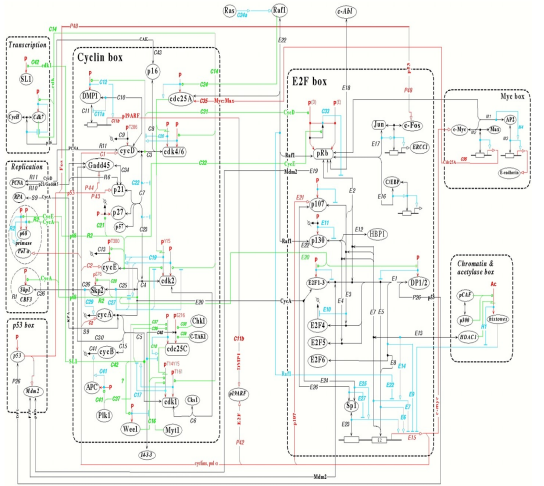
3. Synthesis
   - Computed functions: Turing completeness
   - Compiler of real computable functions in elementary CRNs: normal form theorem
   - Quadratization: complexity and open problems

4. Conclusion on rule-based mathematical modeling

Motivation for this Work: Cells Compute

- Cells process signals
- Regulate their metabolism
- Take decisions such as
  - Replication
  - Differentiation
  - Migration
  - Apoptosis (suicide)
- Control the execution of those processes





Chemical Reaction Networks (CRN)

But what are the programs ?

Analog computation with proteins: gradual concentration levels, continuous time

Church-Turing thesis: there is only one notion of mechanistic computability

- What are the links to Turing machines and digital computation?
- Can we understand, beyond describing, natural CRNs ? (Systems Biology)
- Can we synthetize artificial CRN to implement a function ? (Synthetic Biology)

CRN Syntax

Let $S = \{x_1, \ldots, x_s\}$ be a finite set of molecular species.

Def. A reaction is a quadruple (R, I, P, f), also noted $R \;/\; I \xrightarrow{f} P$
where R (resp. I, P) is a multiset of reactant species (resp. inhibitors, products)
and $f: \mathbb{R}_+^s \to \mathbb{R}_+$ is a rate function (kinetic expression).

- Multisets can be represented by linear expressions (stoichiometric coefficients)
- A reaction catalyst is a molecular species that is both a reactant and a product (can also be an inhibitor).

Def. A reaction (R, I, P, f) is well-formed if

- $f: \mathbb{R}_+^s \to \mathbb{R}_+$ is a partially differentiable function
- $x_i \in R$ if and only if $\frac{\partial f}{\partial x_i}(x) > 0$ for some value $x \in \mathbb{R}_+^s$
- $x_i \in I$ if and only if $\frac{\partial f}{\partial x_i}(x) < 0$ for some value $x \in \mathbb{R}_+^s$.

Def. A reaction is strict if $R(x_i) > 0$ implies $f(x_1, \ldots, x_s) = 0$ whenever $x_i = 0$.

Def. A (strict, well-formed) CRN is a finite set of (strict, well-formed) reactions.

Prop. The ODE associated to a well-formed strict CRN defines a positive system

François Fages, Steven Gay, Sylvain Soliman. Inferring Reaction Systems from Ordinary Differential Equations. *Theoretical Computer Science*, 599:64–78, 2015.

Standard Well-formed Strict CRN Kinetics

mass action law kinetics:

$$\sum_j n_j \times x_j \overset{k \times \prod_j x_j^{n_j}}{\longrightarrow} p$$

Michaelis-Menten kinetics:

$$x \overset{V \times x/(K+x)}{\longrightarrow} y$$

Hill kinetics:

$$x \overset{V \times x^n/(K^n+x^n)}{\longrightarrow} y$$

or negative Hill kinetics:

$$\emptyset/x \overset{V/K^n+x^n}{\longrightarrow} y$$

with rate constants $k, V, K > 0$ *and exponent* $n \geq 1$, *are well-formed and strict.*

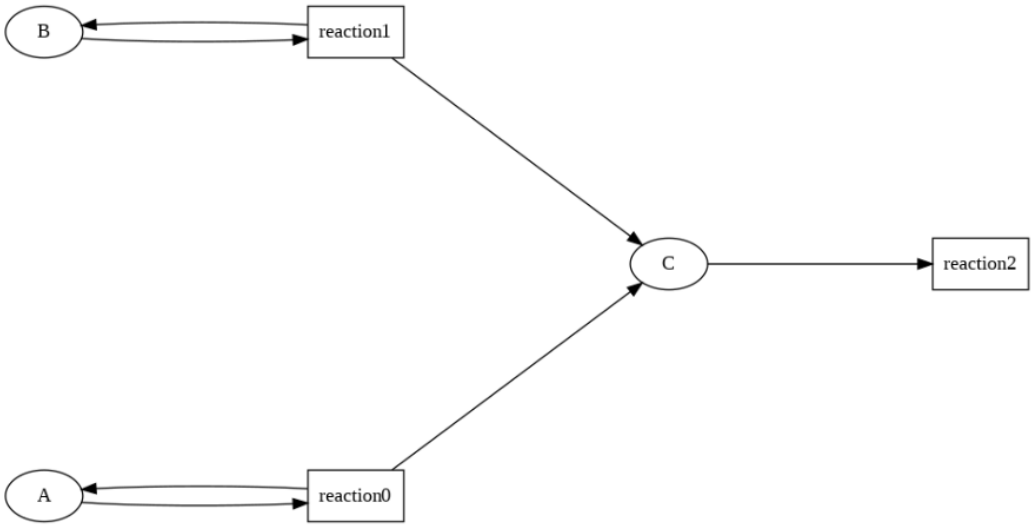
CRN Hypergraph Structure

Standard representation of a hypergraph by a bipartite species/reaction graph.

$B \xrightarrow{k.B} B + C$

$C \xrightarrow{k.C} \emptyset$

$A \xrightarrow{k.A} A + C$



CRN Semantics

One given CRN $\{(R_r, I_r, P_r, f_r)\}_{r \in C}$ can be interpreted in a hierarchy of semantics :

- Continuous interpretation by ordinary differential equations (ODE) in explicit form $x \in \mathbb{R}_+^S$

$$\frac{dx_i}{dt} = \sum_{r \in C} (P_r(x_i) - R_r(x_i)).\, f_r(x)$$

- Stochastic interpretation by continuous-time Markov chain (CTMC) $x \in \mathbb{N}_+^S$

$$x \xrightarrow{x \geq R_r,\ p = \frac{f_r(x)}{\sum f_{r'}(x)},\ \tau = Exp(\frac{1}{\sum f_{r'}(x)})} x - R_r + P_r$$

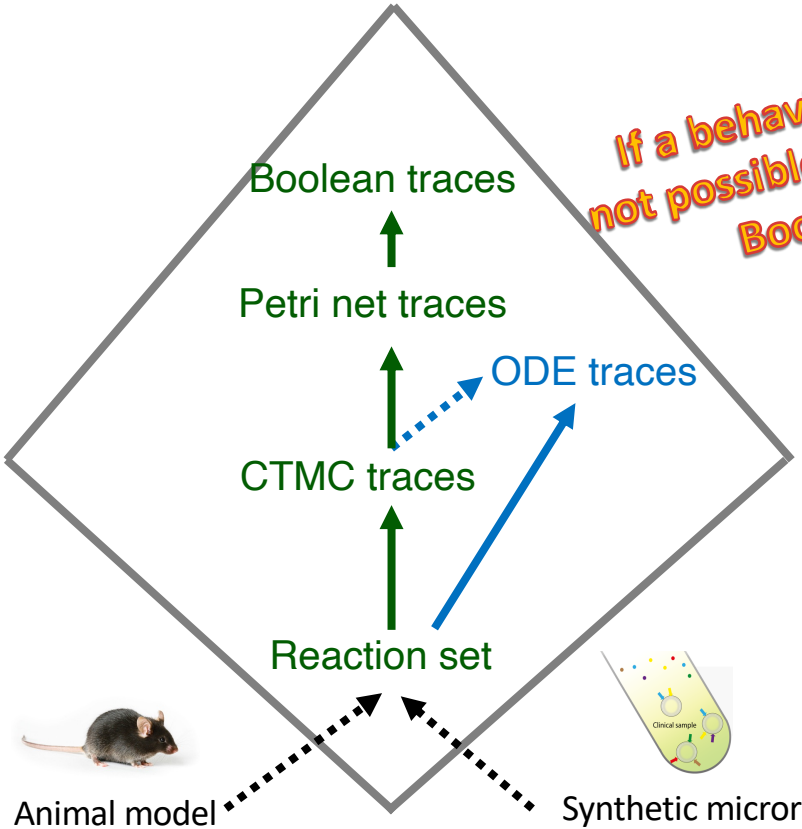
- Rate-independent non-deterministic discrete interpretation by Petri Net (PN) $x \in \mathbb{N}_+^S$

$$x \xrightarrow{x \geq R_r} x - R_r + P_r$$

- Rate-independent non-deterministic asynchronous Boolean state transition interpretation $x \in \mathcal{B}_+^S$

$$x \xrightarrow{x \geq \overline{R}_r,\ x < \overline{P}_r} x' \quad \text{with} \quad (x \wedge \neg \overline{R}_r) \vee \overline{P}_r \leq x' \leq x \vee \overline{P}_r$$

Hierarchy of Trace Semantics



If a behavior is not possible in the Boolean semantics, it is not possible in the stochastic semantics for any reaction rates. Boolean model-checking can exhibit rare events

Thm. (abstract interpretation ↑) Galois connections between the domains of syntactical, stochastic, Petri net and Boolean trace semantics

Fages, Soliman. Abstract Interpretation and Types for Systems Biology. *Theoretical Computer Science* 403(1):52–70, 2008.

Thm. (approximation ⋮) For large numbers of molecules the ODE semantics approximates the *mean* stochastic behavior
[Gillespie 1971 Kurtz 1978, 1992]

Limit theorem
Not the whole story

2. Static Analyses

2.1 Petri Net Place Invariants as Structural ODE Conservation Laws

Def. In an ODE system in explicit form, a conservation law is a function $g(x_1, \ldots, x_k)$ over a subset of variables $\{x_1, \ldots, x_k\}$ such that $\frac{dg(x_1,\ldots,x_k)}{dt} = 0$.

Def. In a Petri Net with integer incidence matrix V, a place-invariant is a non empty multiset of places A such that $V.A = 0$, i.e. the number of tokens in A remains the same whatever transitions are made.

Prop. The place-invariants of a CRN are rate-independent linear conservation laws of the ODE.

E.g. $\{E, SE\}$ and $\{S, SE, P\}$ are the place-invariants of the Michaelis-Menten CRN $S + E \leftrightarrow SE \rightarrow E + P$
They provide linear conservation laws that can be used to eliminate 2 variables, e.g. E and P
since $\forall t \quad E(t) + SE(t) = E(0) + SE(0), \quad S(t) + SE(t) + P(t) = S(0) + SE(0) + P(0)$.

Efficiently implemented in Biocham by a Constraint Logic Program over the integers.

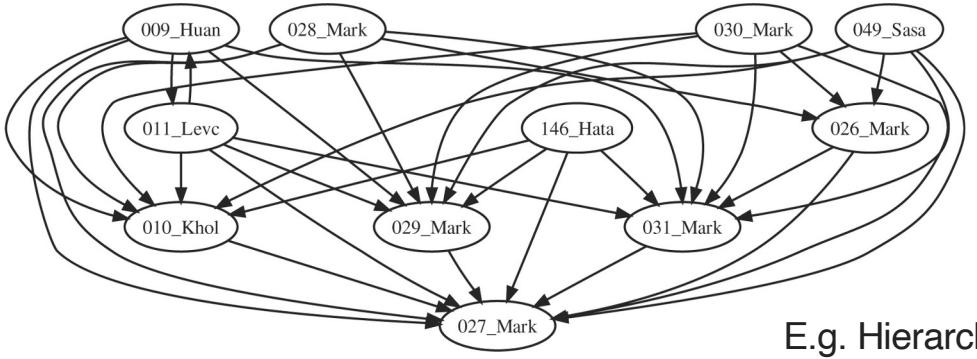
Sylvain Soliman. Invariants and Other Structural Properties of Biochemical Models as a Constraint Satisfaction Problem. *Algorithms for Molecular Biology*, 7(15), 2012.

2.2 Relating CRN Models by Graph Matching

SBML: markup language, exchange format for CRN models

Biomodels: repository of models of biological and biomedical systems (with typically 10 to 10^3 variables)
flat list of thousands of models in SBML with reference to publications and various annotations.

Hierarchy of CRN models related by graph morphisms (purely structural concept)
= Metamodel of models at different levels of details



E.g. Hierarchy of MAPK signalling CRN models in Biomodels

Steven Gay, Sylvain Soliman, François Fages. A Graphical Method for Reducing and Relating Models in Systems Biology. *Bioinformatics*, 26(18):i575–i581, 2010.

Subgraph Epimorphisms

Def. A graph epimorphism from graph $G = (V, A)$ to $G' = (V', A')$ is a surjective function $f : V \to V'$ s.t.

- for all $u, v \in V$, if $(u, v) \in A$, then $(f(u), f(v)) \in A'$ (graph homomorphism),
- and for all $(u', v') \in A'$, there exists $(u, v) \in A$ such that $f(u) = u'$ and $f(v) = v'$ (surjectivity on arcs).

Def. A subgraph epimorphism from G is graph epimorphism from a subgraph induced by a subset $U \subseteq V$

Thm. $G \xrightarrow[SEPI]{} G'$ iff G' is isomorphic to a graph obtained from G by a sequence of graph operations to

- delete species (and incoming/outgoing arcs)
- delete reactions (and incoming/outgoing arcs)
- merge species (and their incoming and outgoing arcs)
- merge reactions (and their incoming and outgoing arcs).

Thm. The existence of a SEPI between two graphs is NP-complete.

Proof. By reduction of SAT.

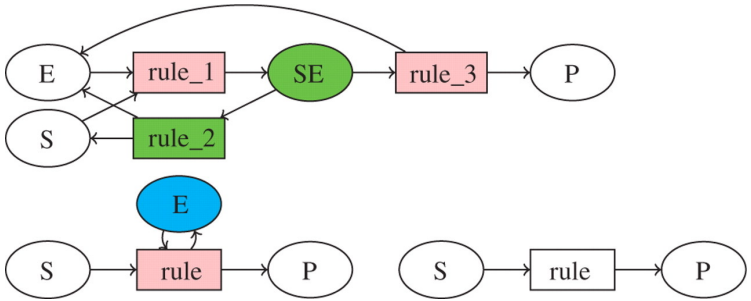
Implemented in Biocham using a Constraint Logic Program (or a SAT solver).

Steven Gay, François Fages, Thierry Martinez, Sylvain Soliman, Christine Solnon. On the subgraph Epimorphism Problem. *Discrete Applied Mathematics*, 162:214–228, 2014.

Example of Michaelis-Menten Reduction

Unlike subgraph isomorphisms, SEPI captures Michaelis Menten CRN reductions from

from 3 reactions with mass action kinetics $k_1.E.S,\ k_2.SE, k_3.SE$ with the enzyme E and complex SE



to 1 reaction $S \rightarrow P$ without the enzyme

with Michaelian kinetics $\frac{V_M\ S}{K_M\ +\ S}$ where $V_M = k_3 E_0$ and $K_M = \frac{k_2+k_3}{k_1}$

Justified by quasi-steady state approximation on SE if $E << S$ (or by quasi-equilibrium if $k_3 \ll k_2$ with $K_M = \frac{k_2}{k_1}$)

Assuming $\frac{d\ ES}{dt} = 0$ we get $\frac{dP}{dt} = -\frac{dS}{dt} = \frac{V_M\ S}{K_M\ +\ S}$ and can show preservation of time scales [Segel 84]

More precise proof then Tikhonov theorem of perturbation theory ? Non standard analysis ? Transseries ?

Conjecture: compositionality of QSSA/QE reductions, reduction to automaton of reduced dynamics.

2.3 Influence Graph of a CRN

Def. The differential influence graph (DIG) of a CRN is the graph of signs of the Jacobian matrix:

$$\{A \to^{+} B \mid \partial \dot{x}_B / \partial x_A > 0 \text{ for some value } x \in R_+^s\}$$
$$\cup \{A \to^{-} B \mid \partial \dot{x}_B / \partial x_A < 0 \text{ for some value } x \in R_+^s\}$$

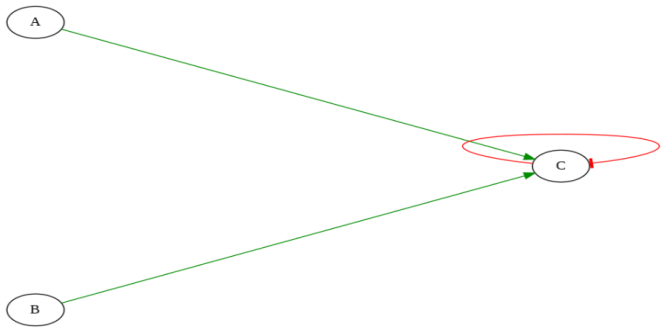
Def. The syntactical influence graph (SIG) of a CRN is the graph with signed arcs

$$\{A \to^{+} B \mid \exists (R_i, I_i, P_i, f_i)\ (R_i(A) > 0 \text{ and } P_i(B) - R_i(B) > 0) \text{ or } (I_i(A) > 0 \text{ and } P_i(B) - R_i(B) < 0)\}$$
$$\cup \{A \to^{-} B \mid \exists (R_i, I_i, P_i, f_i)\ (R_i(A) > 0 \text{ and } P_i(B) - R_i(B) < 0) \text{ or } (I_i(A) > 0 \text{ and } P_i(B) - R_i(B) > 0)\}$$

E.g. $A \xrightarrow{k.A} A + C$

$C \xrightarrow{k.C} \emptyset$

$B \xrightarrow{k.B} B + C$



Prop. The DIG of a well-formed CRN is included in its SIG, and equal if the SIG contains no conflict pair.

Positive Circuits as Necessary Condition for Multistationarity

Thm. A necessary condition for non-zero multistationarity in the ODE semantics of a CRN is the existence of a positive circuit in the reaction-labelled DIG of the CRN

- Using at most once each reaction
- Not both forward and backward reactions of a reversible reaction
- Not all species involved in a conservation law
- And in every DIG obtained by reversing the arcs targeting some technical subsets of species
- And in every DIG obtained by rewiring the arcs targeting any permutation of some subsets of species.

Sylvain Soliman. A stronger necessary condition for the multistationarity of chemical reaction networks. *Bulletin of Mathematical Biology*, 75(11):2289–2303, 2013.

Implemented in Biocham by graph rewriting algorithms.

Adrien Baudier, François Fages, Sylvain Soliman. Graphical Requirements for Multistationarity in Reaction Networks and their Verification in BioModels. *Journal of Theoretical Biology*, 459:79–89, 2018.

Evaluation on Biomodels

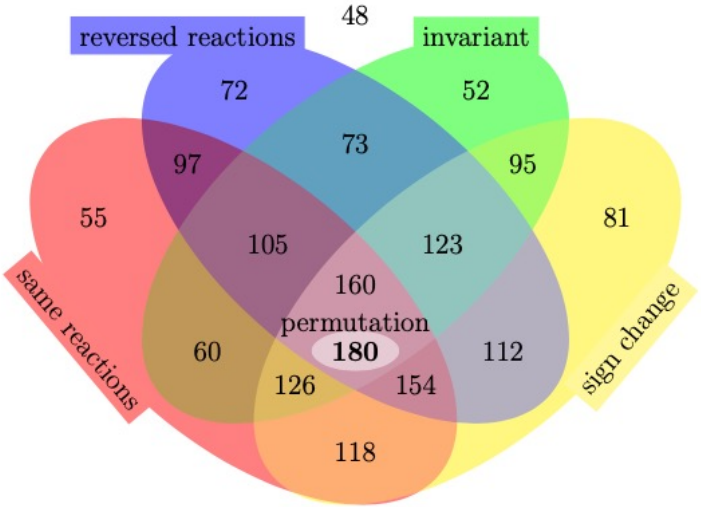


Figure 6: Number of models among the 506 curated reaction models of BioModels for which multistationarity can be ruled out by using respectively original Thomas's positive circuit condition, Cor. 2.3 (no same reactions), 2.4 (no reversed reactions), 2.5 (no invariant) and 2.6 (sign change), plus 2.7 (permutation).

r	Jacobian Method [13] model (1)	Graphical Method (Alg 2 & 4) (1)	(2)	(3)
1	4	0.3	1.6	0.6
2	75	0.5	2	1
3	44	1	2	1
4	81	1	3	1
5	191	1	4	1
6	256	1	4	1
7	444	1	5	1
8	795	2	5	1
9	1169	2	6	2
10	2195	2	6	2
11	3998	2	6	2
12	7696	2	7	2
13	15180	2	7	2
14	32180	3	7	2
15	67740	3	7	2
16	171700	3	8	2
17	1199000	4	8	2
50	×	12	17	4
100	×	26	40	6
500	×	343	549	34
1000	×	1200	1874	98

Table 2: Execution times given in milliseconds for the analysis of the r-site phosphorylation system of [34], first as reported in [13] for the Jacobian method using symbolic computation, then obtained with our graphical algorithm on the same model and on two variants concerning the writing of the dephosphorylation and phosphorylation reactions.

3. CRN Synthesis

Computable Real Numbers and Functions

Classical definitions of computable analysis based on Turing machines

Definition. A real number *r* is computable if there exists a Turing machine with

Input: precision $p \in \mathbb{N}$

Output: rational number $q \in \mathbb{Q}$ with $|r-q| < 2^{-p}$

Examples. Rational numbers, limits of computable Cauchy sequences π, e, …

Definition. A real function $f:R \to R$ is computable if there exists a Turing machine that computes $f(x)$ with an oracle for x.

Examples. Polynomials, trigonometric functions, analytic functions…

Counter-examples. x=0, $\lceil x \rceil$ are not computable (undecidable on x=0.000…) discontinuous functions are not computable

Decision problem $w \in \mathcal{L}$: analog encoding by a real function $f:R \to R$?

Input encoding e: $\mathcal{L} \to \mathbb{R}$ problem encoding by f: accept w if f(e(w)) >1 reject if <-1

Analog Computer? Differential Analyzer [Bush 1931]

Underlying principles: Lord Kelvin, 1876

First ever built: Vannevar Bush, MIT, 1931

Applications: from gunfire control up to aircraft design

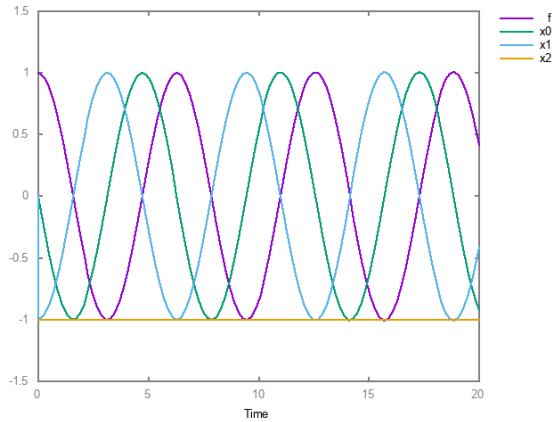
- Intensively used by the U.S. and Japanese armies during world war II
- Electronic versions from late 40s, used until 70s

General Purpose Analog Computer [Shannon 1941]

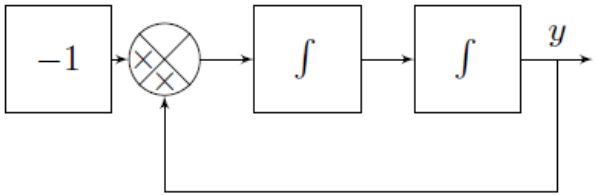
Shannon's formalization of the Differential Analyser by GPAC circuits

A time function if GPAC-generated if it is the output of some unit of a GPAC circuit built from:

1. Constant unit
2. Sum unit
3. Product unit
4. Integral $\int y\,dx$ unit (dt by default)



What does this GPAC circuit compute ?



$$y_1 = \frac{dy}{dt}$$

$$\frac{dy_1}{dt} = -y = y''$$

if $y(0) = 1, y_1(0) = 0$

$y(t) = cos(t) \quad y_1(t) = sin(t)$

CRN Implementation of GPAC Units

Mass action law kinetics reaction network with output concentration stabilizing on the result of the operation applied to the input concentrations

Positive constant units: molecular concentrations

Product unit $z = x.y$

$$x + y \xrightarrow{k.x.y} x + y + z$$
$$z \xrightarrow{k.z} \_$$
$$\frac{dz}{dt} = k(xy - z)$$
$$= 0 \text{ when } z = x.y$$

Sum unit $z = x + y$

$$x \xrightarrow{k.x} x + z$$
$$y \xrightarrow{k.y} y + z$$
$$z \xrightarrow{k.z} \_$$
$$\frac{dz}{dt} = k(x + y - z)$$
$$= 0 \text{ when } z = x + y$$

Time integral $z = \int x\,dt$ unit

$$x \xrightarrow{x} x + z$$
$$\frac{dz}{dt} = x$$
$$z = \int_0^T x\,dt$$

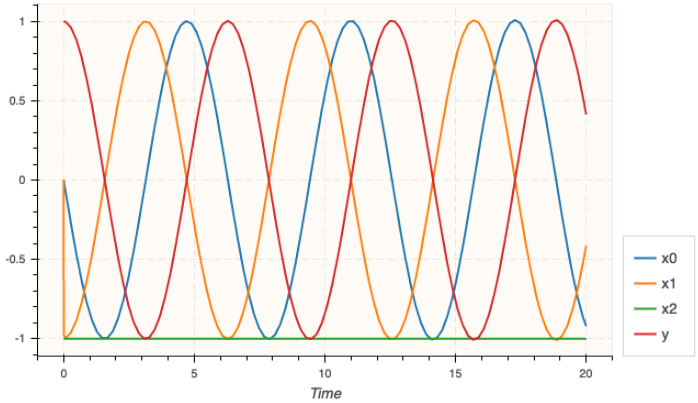
Polynomial ODE Initial Value Problems (PIVP)

Graça and Costa 2003's formalization of GPAC generated functions

Definition. A real time function $f:R_+ \to R$ is PIVP-generable iff there exist a vector of polynomials $p \in R^n[R^n]$ and of initial values $y(0) \in R^n$

and a solution function $y:R_+ \to R^n$ such that $y'(t)=p(y(t))$ and $\mathbf{f(t)=y_1(t)}$

Example. y=cos(t)



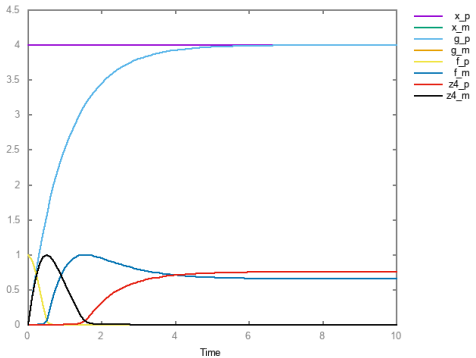
Closure properties:

f+g, f-g, f.g, 1/f, ,f ◦g, y s.t. y′ =f(y) are GPAC-generable if f, g are.

PIVP-Computable Function f(x)

Definition. [Graça Costa 03 J. Complexity] A real function f:R→R is PIVP-computable if there exists vectors of polynomials $p \in R^n[R^n]$ and $q \in R^n[R]$ and a function y: $R^n \to R^n$ such that $y'(t)=p(y(t))$, $\mathbf{y(0)=q(x)}$ and $\mathbf{|y_1(t)-f(x)|<y_2(t)}$ with $y_2(t) \geq 0$ decreasing for t>1 and $\lim_{t\to\infty} y_2(t) = 0$

Example. y=cos(4)



Reconciles Digital and Analog Computation !

Theorem (analog characterization of Turing computability). [Bournez Campagnolo Graça Hainry 07 J. Complex]
A real function is computable (by Turing machine) iff it is PIVP-computable.

Analog characterization of Ptime

Time in ODE is a bad measure of complexity

- Exponential speedup by changing time variable $t' = e^t$
- But price to pay in the amplitude of t'

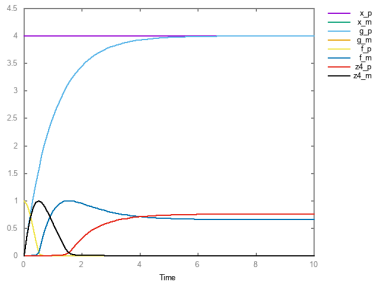
A computational complexity measure should combine time and space-amplitude

- length in the n dimensions of the trajectory to compute the result

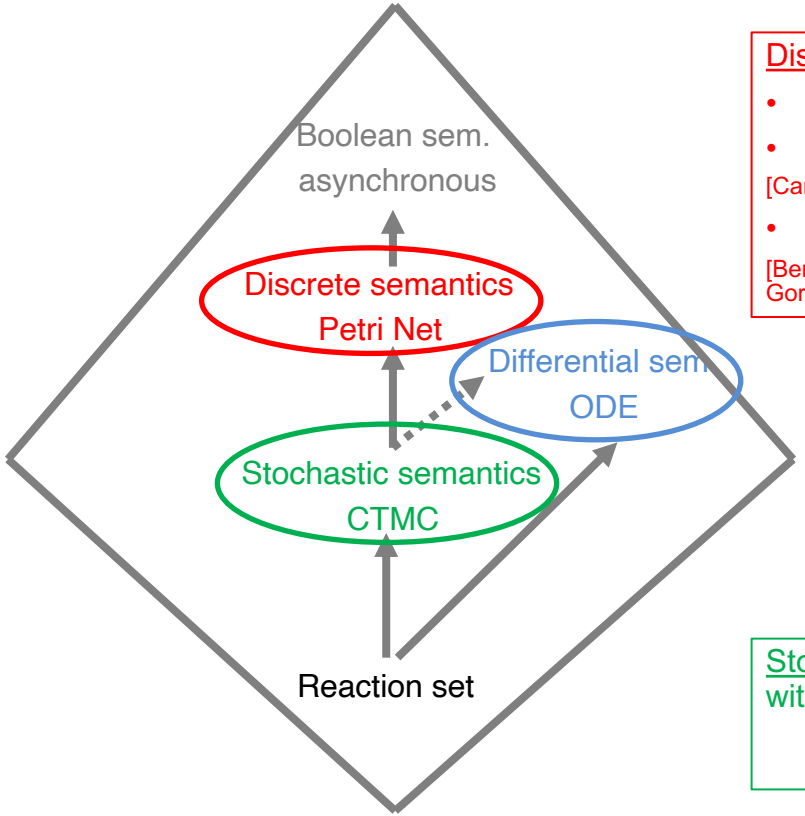
Theorem [Pouly PhD thesis 2015, Bournez Graca Pouly 16 ICALP]

A real function is computable in P iff it is PIVP-computable with a trajectory of polynomial length (i.e. polynomial time and polynomial amplitude)

Reconciles Digital and Analog Polynomial Time Complexity !



Turing-Completeness of CRNs



Discrete CRN: **Not Turing complete** without

- test of absence (Petri net inhibitor arc)
- or polymerisation reactions

[Cardelli Zavatero MSCS 2010, Cook et al 2009]

- or unbounded membranes

[Berry Boudol CHAM 1994, Paun Rozenberg TCS 2002, Busi Gorrieri CMSB 2005]

DIfferential CRN: for each function *for each input there exists a circuit* computing the result [Magnosco 1997 Phys. Rev., Helmfelt Weinberger PNAS 1991]

Turing completeness? for each function *there exists a circuit* computing the result *for each input*

Stochastic CRN: Simulation of a Turing machine with a **small probability of error**

[Cook, Soloveichik, Winfree, Bruck 2009]

Turing Completeness of Continuous CRN

- Consider mass action law kinetics
  - polynomial ODEs
  - PIVP computation of input/output function
- Molecular concentration are positive real values
  - Restriction to positive dynamical systems
- Elementary reactions with at most two reactants
  - Restriction to PIVP of degree at most 2

Turing Completeness of Continuous CRNs

Lemma (positive systems) Any PIVP-computable function can be encoded by a PIVP of double dimension on R^+, preserving polynomial length complexity.

Proof. Encode $y_i \in R$ by y^-_i $y^+_i \in R^+$ such that $\mathbf{y_i = y^+_i - y^-_i}$ (dual-rail encoding of [Hars Toth 79] used in [Oishi Klavins 2011] for encoding linear I/O systems)

For a PIVP $p[y_1, \ldots, y_n]$ let $\underline{p}_i(y^+_1, y^-_1, \ldots, y^+_n, y^-_n) = p_i[y = y^+_i - y^-_i]$

Let $\underline{p}_i = \underline{p}^+_i - \underline{p}^-_i$ where $\underline{p}^+_i$, $\underline{p}^-_i$ are positive coefficient polynomials.

Let us consider the positive PIVP defined by

$$y^+_i{}' = \underline{p}^+_i - f_i\, y^+_i\, y^-_i \qquad y^+_i(0) = \max(0, y_i(0))$$

$$y^-_i{}' = \underline{p}^-_i - f_i\, y^+_i\, y^-_i \qquad y^-_i(0) = \max(0, -y_i(0))$$

where f_i is chosen large enough such that $f_i\, y^+_i\, y^-_i \geqq \max(\underline{p}^+_i, \underline{p}^-_i)$

At any time time we have $y_i(t) = y^+_i(t) - y^-_i(t)$.

- Fast annihilation reactions: $y^+_i + y^-_i \xrightarrow{f_i} \_$
- n-ary catalytic synthesis reactions for each monomial $\underline{m}^+_{i,j}$ in $\underline{p}^+_i$, $\underline{m}^-_{i,j}$ in $\underline{p}^-_i$:

$$M_{i,j}{}^+ \xrightarrow{\underline{m}^+_{i,j}} y^+_i + M_{i,j}{}^+ \quad \text{and} \quad M_{i,j}{}^- \xrightarrow{\underline{m}^-_{i,j}} y^+_i + Mi_{,j}{}^-$$

Turing Completeness of Continuous CRNs

Lemma (quadratic systems) [Carothers Parker Sochacki Warne 2005]
Any PIVP can be encoded by a PIVP of degree ≤ 2.

Proof. Introduce variable $v_{i1,\ldots,in}$ for each possible monomial $y_1^{i1}\ldots y_n^{in}$
We have $y_1 = v_{1,0\ldots,0}$, $y_2 = v_{0,1,0\ldots,0}$,... y'_i is of degree one in $v_{i1,\ldots,in}$
$v'_{i1,\ldots,in} = \sum_{k=1}^{n} i_k v_{i_1,\ldots,i_{k-1},\ldots,i_n} y'_k$ is of degree at most 2. Trade high dimension for low degrees.

That algorithm may introduce an exponential number of variables.

Thm. Deciding the existence of this kind of quadratization with k variables is NP-complete in the non-succinct (matrix) representation (in NExp with succinct symbolic representation).

Proof. By reduction of the feedback vertex set problem.

Mathieu Hemery, François Fages, Sylvain Soliman. On the Complexity of Quadratization for Polynomial Differential Equations. In *CMSB'20: Proceedings of the eighteenth international conference on Computational Methods in Systems Biology*, *Lecture Notes in Computer Science*. Springer-Verlag, 2020.

Optimal Quadratizations

Suboptimal quadriatization implemented in Biocham using a SAT solver and solution preserving heuristics.

Mathieu Hemery, François Fages, Sylvain Soliman. On the Complexity of Quadratization for Polynomial Differential Equations. In *CMSB'20: Proceedings of the eighteenth international conference on Computational Methods in Systems Biology*, *Lecture Notes in Computer Science*. Springer-Verlag, 2020.

Optimal branch-and-bound algorithm for monomial quadratization

Andrey Bychkov and Gleb Pogudin. Optimal monomial quadratization for ode systems. In Proceedings of the IWOCA 2021 - 32nd International Workshop on Combinatorial Algorithms, July 2021.

Existence of non-monomial quadratizations of smaller dimension

Alauddin, F.: Quadratization of ODEs: monomial vs. non-monomial. SIAM Undergraduate Res. Online **14** (2021).

Could the existence of a quadratization of dimension k be undecidable? Similarly to

Matiyasevich 1971 Hilbert 10th problem, solving of Diophantine Equations, is undecidable,

Turing Completeness of Continuous CRNs

Theorem [Fages, Le Guludec, Bournez, Pouly CMSB 2017]

Any computable function over the reals can be computed by a continuous CRN over a finite set of molecular species (no polymerization, no compartments)

In this view, the (protein) concentrations are the information carriers.

The programs of a cell are implicitly defined by the set of all possible reactions

- with the proteins encoded in its genome
- and the chemicals of the environment.

Program change is determined by gene expression which can be seen as a (digital) metaprogram

- No artificial construct (no polymers)
- Compatible with natural cells: making of programming a “natural science” !

Normal Form Theorem

Theorem (abstract CRN normal form)

A real function is computable if and only if it is computable by a system of elementary reactions of the form

_ => z or x => x+z or x+y => x+y+z

plus annihilation reactions x+y => _ all with mass action law kinetics

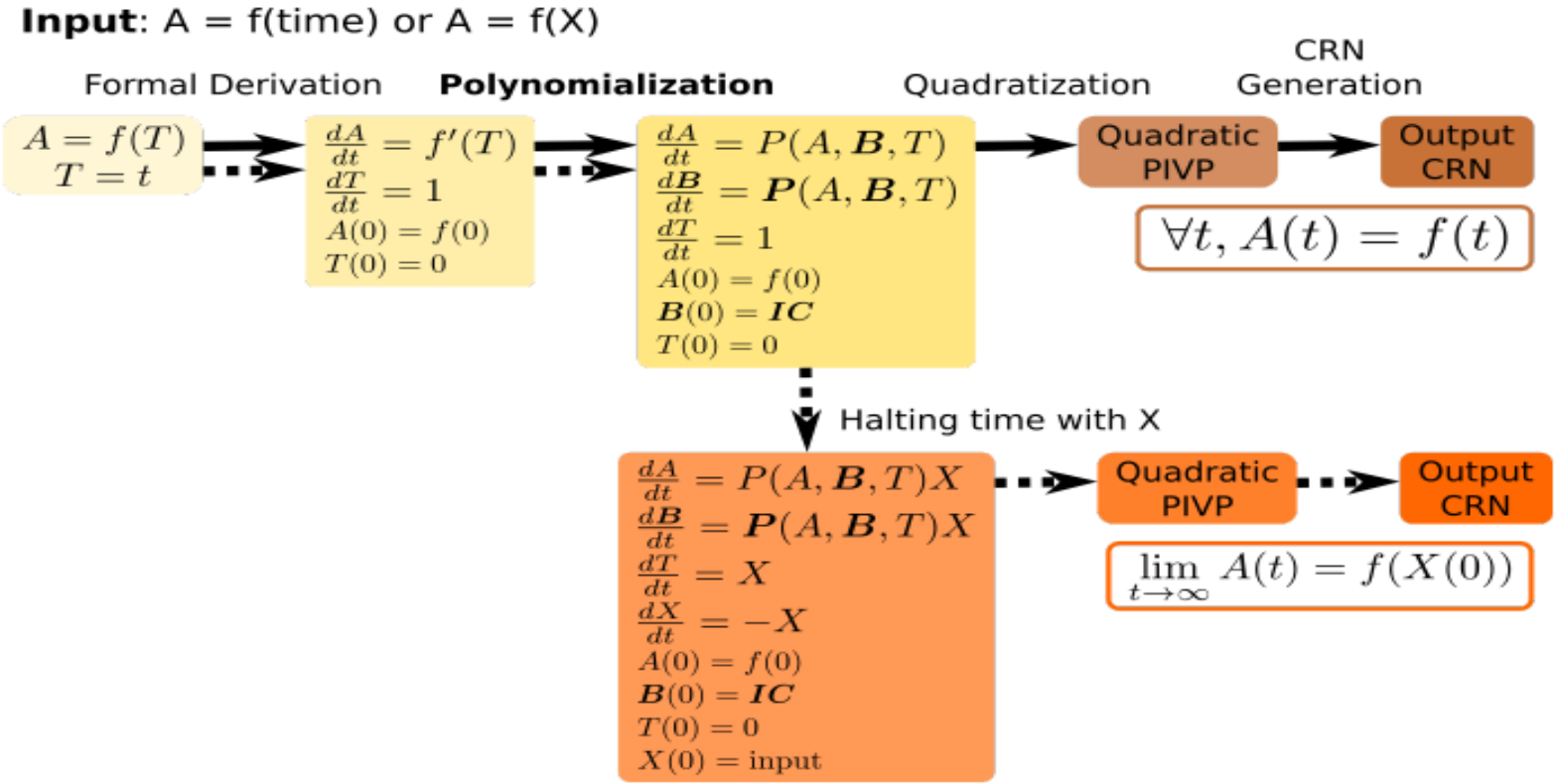
Realistic CRN:

- formal annihilations by complexations (e.g. in a stable inactive complex)
- formal syntheses by modifications (e.g. phosphorylation with kinases)

Concrete CRN: search mapping with real enzymes (e.g. Brenda database)

- Easier for CRN with rate independence property (ensured by graphical conditions [Degrand Fages Soliman CMSB 2020])
- Robustness w.r.t. parameter perturbations (extrinsic variability)
- Robustness w.r.t. stochastic simulations (intrinsic variability)

Compiler of Real Functions in Elementary CRNs

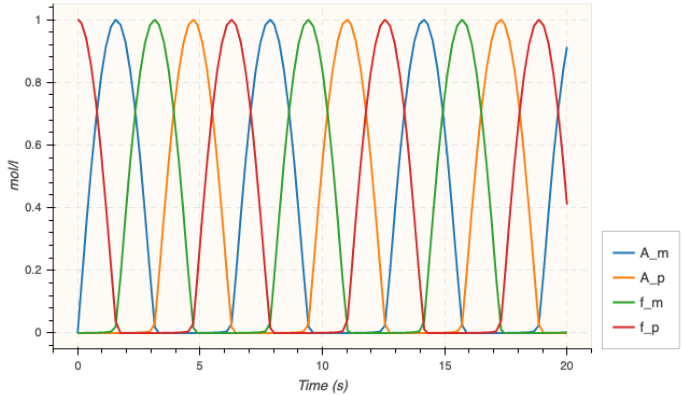


Compiling Cosine(time)

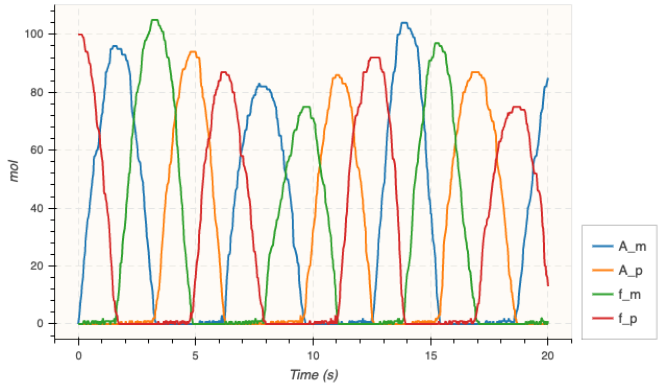
```
biocham: compile_from_expression(cos,time,f).
initial_state(f_p=1).
MA(fast) for f_m+f_p=>_.
MA(fast) for A_m+A_p=>_.
MA(1.0) for A_p=>A_p+f_p.
MA(1.0) for A_m=>A_m+f_m.
MA(1.0) for f_m=>A_p+f_m.
MA(1.0) for f_p=>A_m+f_p.
```

$$\frac{dA\_m}{dt} = f\_p - fast * A\_m * A\_p$$
$$\frac{dA\_p}{dt} = f\_m - fast * A\_m * A\_p$$
$$\frac{df\_m}{dt} = A\_m - fast * f\_m * f\_p$$
$$\frac{df\_p}{dt} = A\_p - fast * f\_m * f\_p$$

ODE simulation (design)



Stochastic simulation (test)



Compiling Cosine(input)

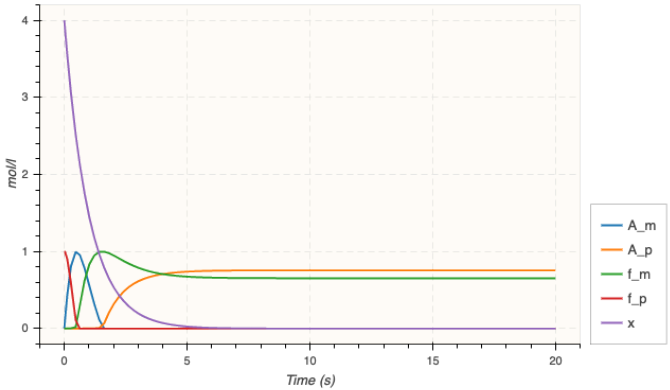
```
biocham: parameter(input=4).
biocham: compile_from_expression(cos,x,f).
initial_state(f_p=1, x=input).
MA(fast) for f_m+f_p=>_.
MA(fast) for A_m+A_p=>_.
MA(1.0) for A_p+x=>A_p+f_p+x.
MA(1.0) for A_m+x=>A_m+f_m+x.
MA(1.0) for f_m+x=>A_p+f_m+x.
MA(1.0) for f_p+x=>A_m+f_p+x.
MA(1.0) for x=>_.
```

PIVP that generates f(g(t)) with $\lim_{t-\infty} g(t) = x$

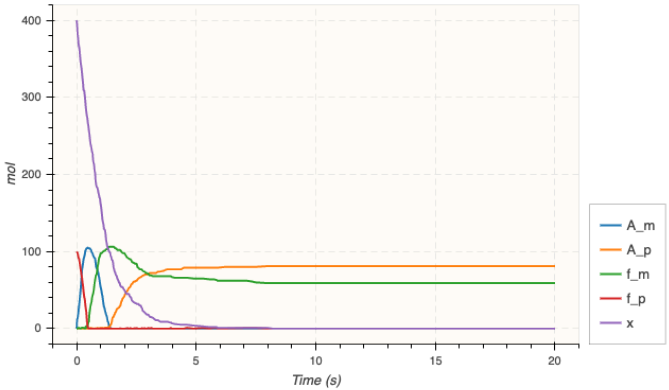
$$g'(t) = x - g(t)$$

$$g(t) = x + (x0 - x)e^{-t}$$

ODE simulation (design)



Stochastic simulation (test)



Sequentiality and Iteration

```
Division(A, B)
  begin
  01    while A ≥ B
  02        A := A − B
  03        Q := Q + 1
  04    R := A
  end
```

Main Reactions		Preconditions
01	while $[A] \geq [B]$	
02	$(A + B \to D)$	$\neg G_\theta$
03	$C \to Q + E$	$A_\theta \wedge \neg B_\theta$
04	$D \to F$	$\neg C_\theta$
05	$E \to G$	$\neg D_\theta$
06	$F \to B$	$\neg E_\theta$
07	$G \to C$	$\neg F_\theta$
08	$D \to R$	$\neg A_\theta$

1. Asynchronous (precondition) CRN programming

[Huang Jiang Huang Cheng 2012 ICCAD]

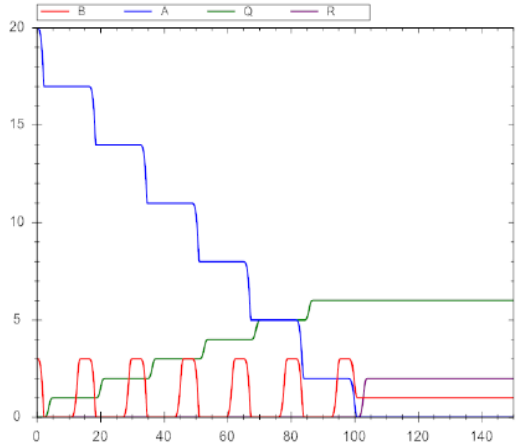
[Huang Huang Chiang Jiang Fages 2013 IWBDA]

many species and reactions

2. Synchronous (clock) CRN programming

[Vasic, David Soloveichik, Sarfraz Khurshid 2018 CRN++]

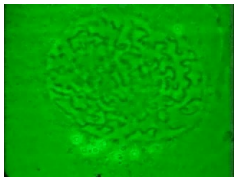
many reactions with the clock

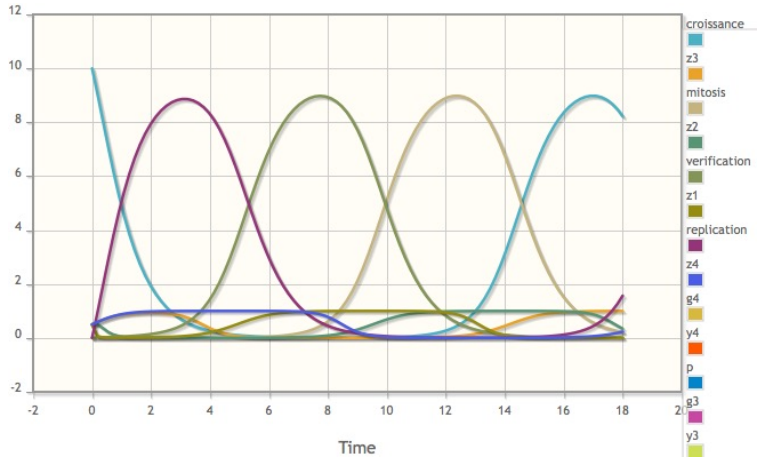


Cell Division Cycle Program

```
while true {growing; replication; verification; mitosis}
```

→ compilation of sequentiality and loops with program control variables
→ 50 reactions
→ 13 variables





Cyclins D, E, A, B appear as necessary markers for implementing sequentiality

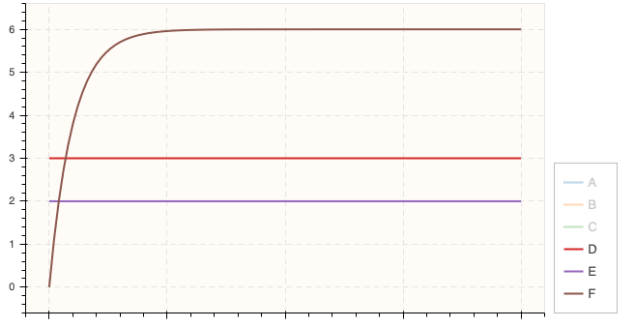
TD Chemical Arithmetic

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

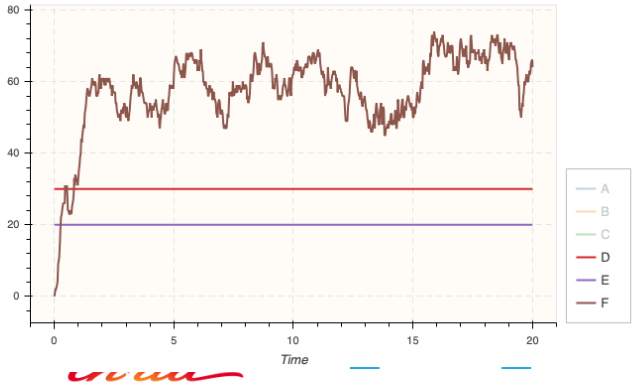
Product F = E * F

```
In [11]: D+E => D+E+F.

In [12]: F => _.

In [13]: present(D,d). parameter(d=3).
         present(E,e). parameter(e=2).
```

```
In [17]: option(show:{D,E,F}).
         numerical_simulation. plot.
```




```
In [19]: option(method:ssa, stochastic_conversion: 10).
         numerical_simulation. plot.
```



Wrapup

- A rule-based model has an explicit graphical structure
- It is higher-level than a flat ODE model similarly to assembly code
- It can be compared to other rule-based models by graph reductions to form a hierarchy of models at different levels of details

- It can be interpreted in a hierarchy of semantics ODE, CTMC, PN, Bool according to the question
- Graphical analyses provide efficient necessary or sufficient conditions for several dynamical properties

- Turing-completeness: any computable real function can be computed by a finite CRN
- Robust online computations characterized as the set of real algebraic functions
- Compiler of real functions in elementary CRNs though symbolic transformation steps
- Importance of the quadratization problem

- Open problem for restricting to a catalogue of reactions along the compilation pipeline.
- Practical applications for the design of biosensors