La Cellule
un Calculateur Analogique Chimique

François Fages

Project-Team Lifeware

http://lifeware.inria.fr/

Institut National de Recherche en Informatique et Automatique

Inria Saclay – Ile de France

Plan

1. Example of natural MAPK signal processing CRN and its biological function

2. Computable real numbers and functions

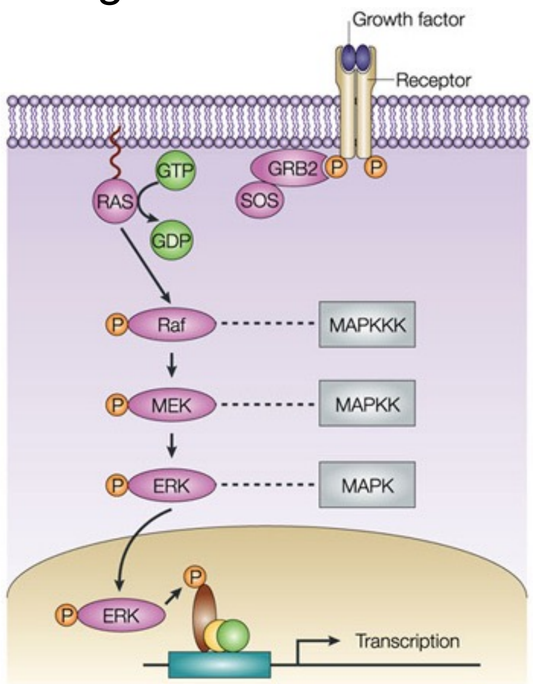
3. Turing-completeness of finite continuous CRNs

4. Compiler of mathematical functions and imperative programs in CRNs
   - Synthesis of oscillators, sigmoids, logical gates
   - Synthesis for on-line computation
   - Design of a differentiation CRN and SEPI-search in BioModels natural CRNs

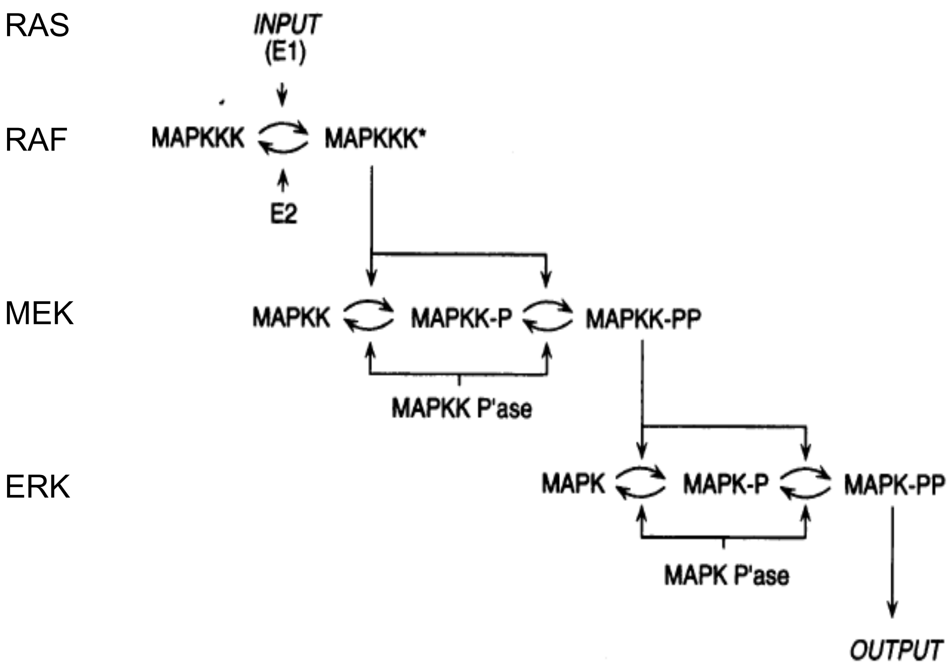
5. Validation of artificial DNA-free RNA-free diagnostic vesicles

1 MAPK Signal Transduction CRN

"Mitogen Activated Protein Kinase" CRN structure: 30 reactions 18 species



Nature Reviews | Molecular Cell Biology



Ubiquitous CRN structure in eukaryotes (yeast, mammals,…)

In several copies in the same cell type with different kinases

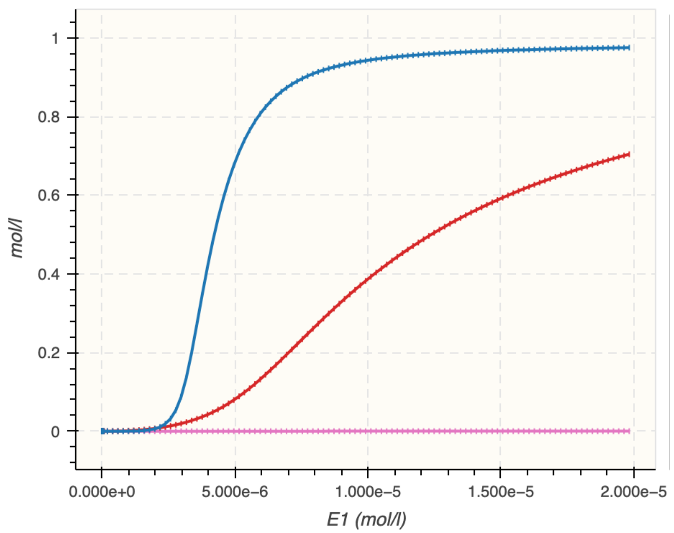
Why that particular CRN structure with 3 levels of 1 or 2 phosphorylations ?

MAPK Function ?

Dose-response diagrams alias bifurcation diagrams alias input/output function

(biologiste) (mathematician) (informatician)

```
dose_response(E1,0,2e-5, show:{KKKp, KKpp, Kpp}).
```



[Huang Ferrel 1996 PNAS]

MAPK responses: Hill functions $\frac{x^n}{c + x^n}$

n ≈ 4.9 at 3rd level mapkpp

n ≈ 1.7 at 2nd level mapkkpp

n = 1 at 1st level mapkkkp (Michaelis-Menten)

- Signal amplification at 2nd level
- Stiff 0/1 response at 3rd level

MAPK CRNs are analog-digital converters in cells

How would you program $\frac{x^5}{c + x^5}$ with biochemical reactions ?

Can we implement any computable function with a finite CRN ?

What does it mean to compute with real number concentrations ?

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

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

Definition. A real function $f:R \rightarrow R$ 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 $f:R \rightarrow R$?
Input encoding e: $\mathcal{L} \rightarrow R$ problem encoding by f: accept w if $f(e(w)) > 2$ 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

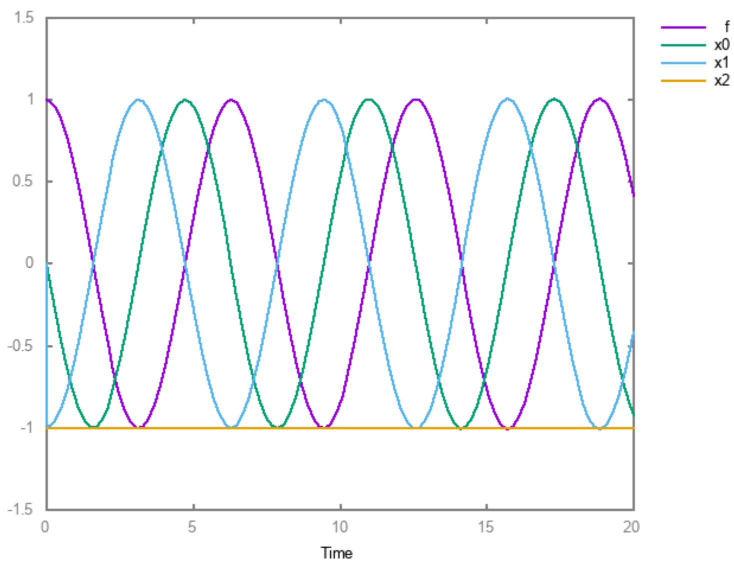
- Intensively used by the U.S. and Japanese armies during world war II
- Electronic versions from late 40s, used until 70s

General Purpose Analog Computer [Shannon 1941]

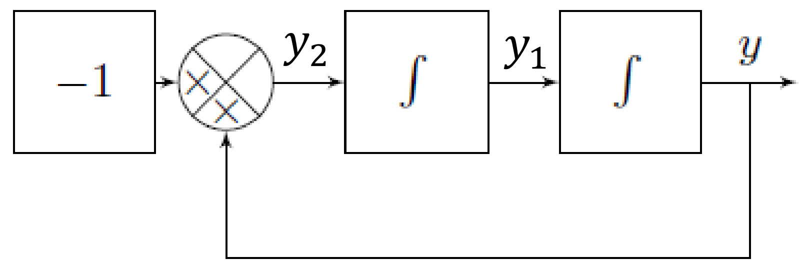
Shannon's formalization of the Differential Analyser by GPAC circuits

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 ?



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

$$y_2 = \frac{dy_1}{dt} = -y = y''$$

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)

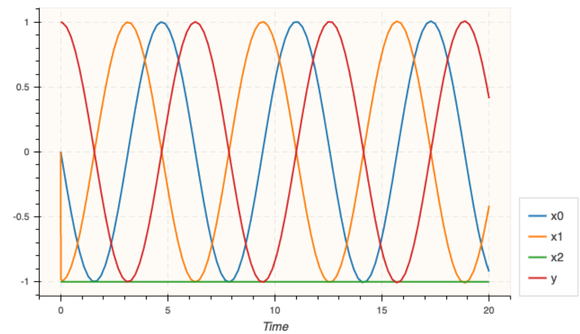
Graça and Costa 2003's formalization of GPAC generated functions

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 $f(t)=y_1(t)$

Example. y=cos(t)



Closure properties: f+g, f-g, f.g, 1/f, ,f ◦g, $\int f$ are GPAC-generable if f, g are.

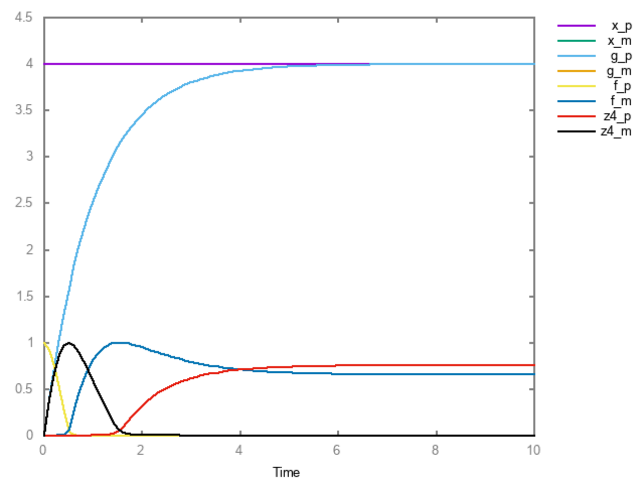
Analytic functions (locally convergent power series) are Turing-computable but some analytic functions are not GPAC-generable [Shannon 41]

- Euler's Gamma function $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ [Hölder1887]
- Riemann's Zeta function $\zeta(x) = \sum_{k=0}^{\infty} \frac{1}{k^x}$ [Hilbert]

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 ! Turing and Shannon

Theorem (analog characterization of Turing computability).

[Bournez Campagnolo Graça Hainry 07 J. of Complexity]

A real function is computable (by Turing machine) iff it is PIVP-computable.

Analog Computation Complexity

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'

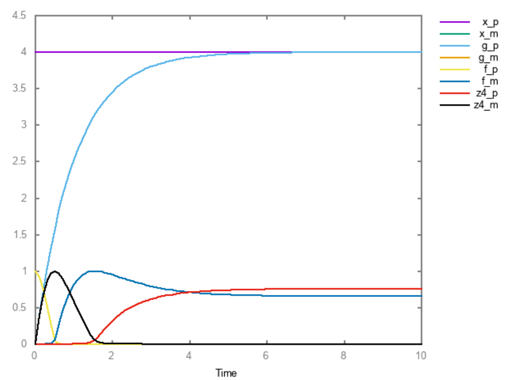
A computational complexity measure should combine time and space-amplitude

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

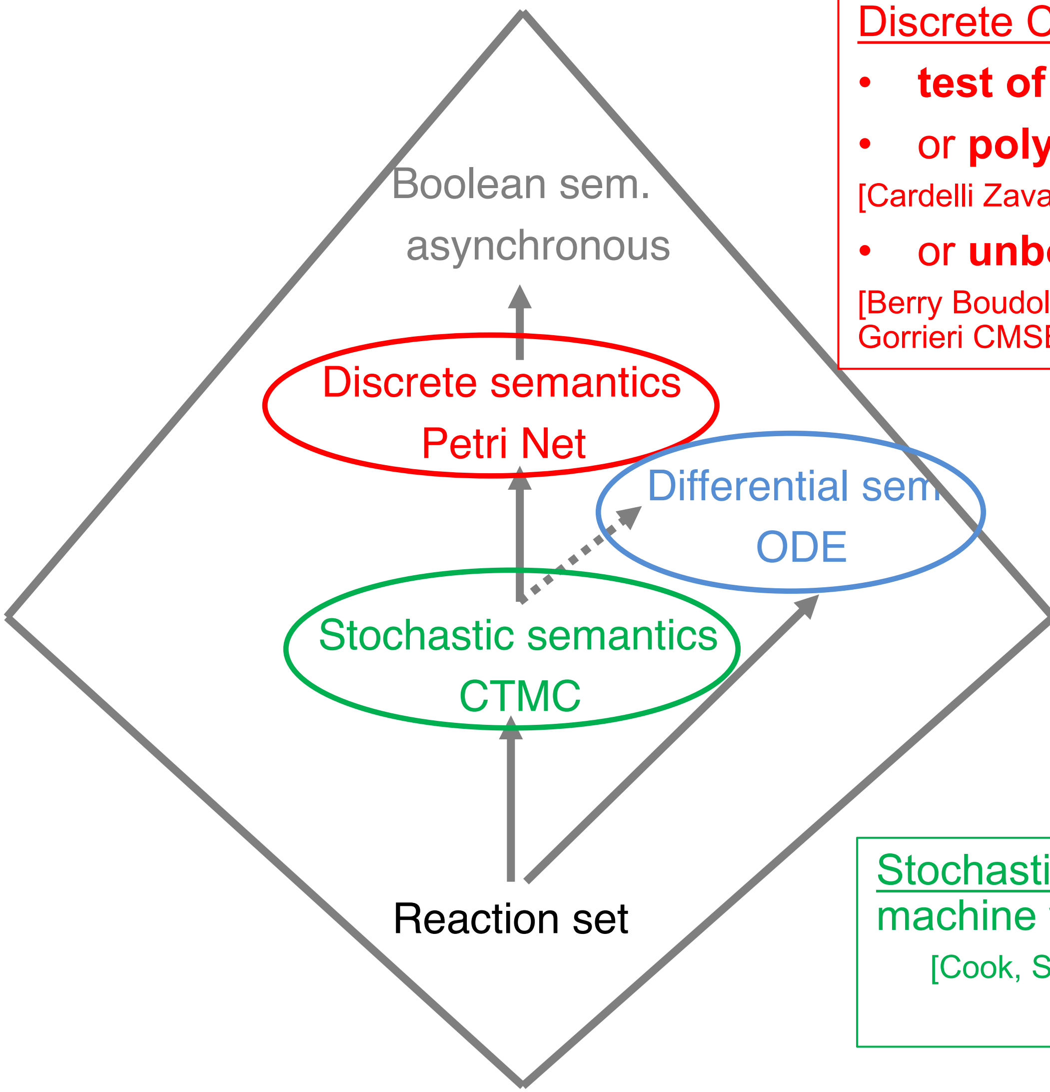
Theorem [Pouly PhD thesis 2015, Bournez Graca Pouly 16 ICALP]

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 !



3. 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?

- 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 ?

[F-, Guillaume Le Guludec , Olivier Bournez , Amaury Pouly. Strong Turing Completeness of Continuous Chemical Reaction Networks and Compilation of Mixed Analog-Digital Programs, *CMSB* 2017]

CRN compilers implemented in Biocham-4 (biochemical abstract machine):

- Compiler of mathematical functions of time, or of some input, or of programs in one abstract CRN
- Compiler for CRN computation of input/output signals on-line [Hemery F- *CMSB* 2022]

Turing Completeness of Continuous CRNs 1/3

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 linear I/O systems)

For a PIVP p[y]

let $\underline{p}_i(y^+_1, y^-_1, \ldots, y^+_n, y^-_n) = p_i[y_i = y^+_i - y^-_i]$ and $\underline{p}_i = \underline{p}^+_i - \underline{p}^-_i$

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

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

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

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

- 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 $\underline{p}^-_i$:

$$M_{i,j}^{+} \xrightarrow{\underline{m}^+_{i,j}} y^+_i + M_{i,j}^{+}$$

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

Turing Completeness of Continuous CRNs 2/3

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.

Complexity?
That algorithm may introduce an exponential number of variables.
The existence of a solution with k variables is proved NP-complete in the non-succinct (matrix) representation [Hemery F. Soliman CMSB 2020]
Conjectured NExp-complete in the succinct (symbolic) representation

Turing Completeness of Continuous CRNs

Theorem [F, 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… CRN programming as “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:

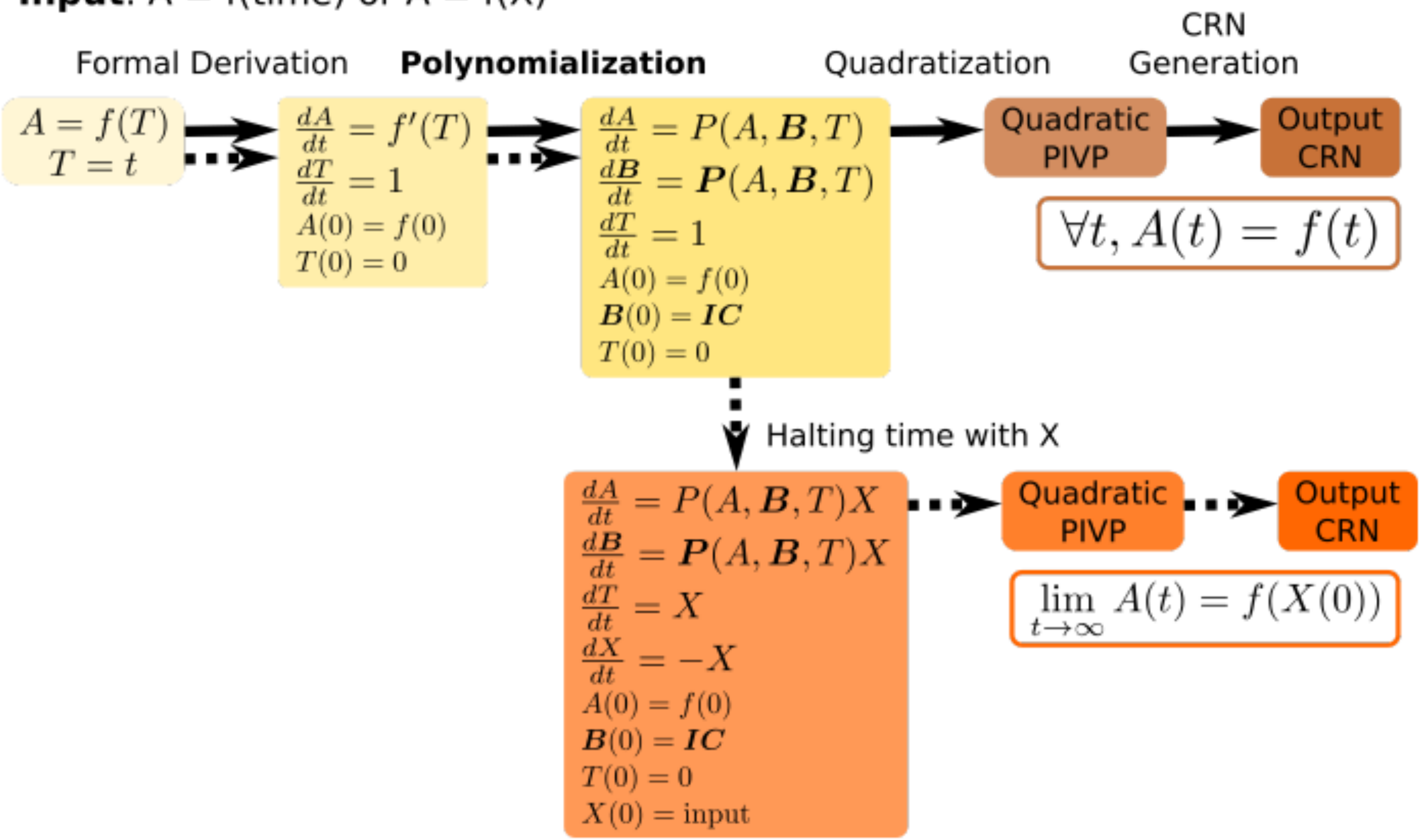
- formal annihilations by complexations (e.g. in a stable inactive complex)
- formal syntheses by modifications (e.g. phosphorylation with kinases)

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

- SEPI search + fitting kinetics by enzyme concentration tuning
- Robustness w.r.t. parameter perturbations (extrinsic variability)
- Robustness w.r.t. stochastic simulations (intrinsic variability)

4. Compiler of Mathematical Functions in CRN

Input: A = f(time) or A = f(X)



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.

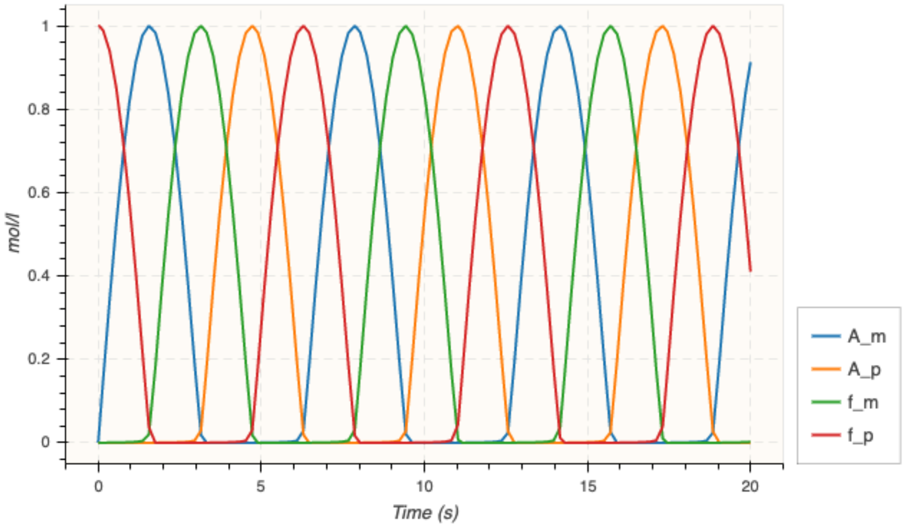
MA(1.0) for A_m=>A_m+f_m.

MA(1.0) for f_m=>A_p+f_m.

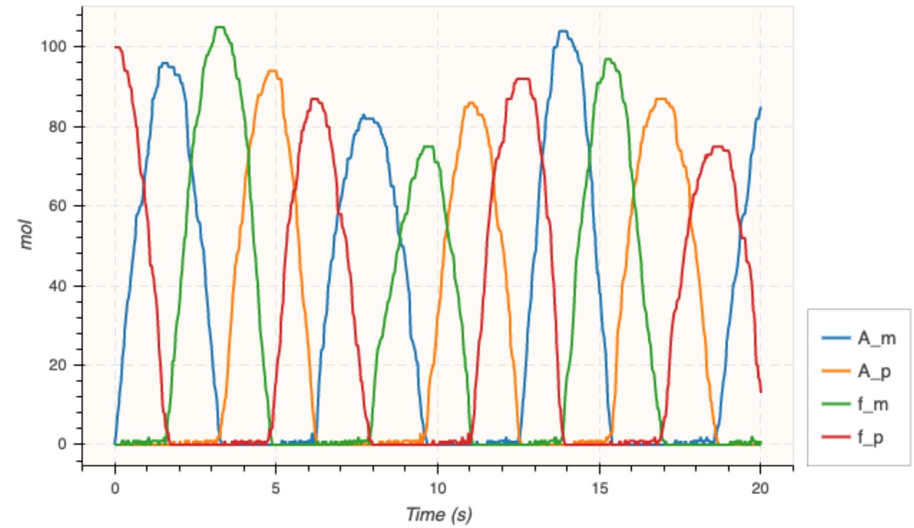
MA(1.0) for f_p=>A_m+f_p.

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

ODE simulation (<u>design</u>)



Stochastic simulation (<u>test</u>)



Compiling Cosine(input)

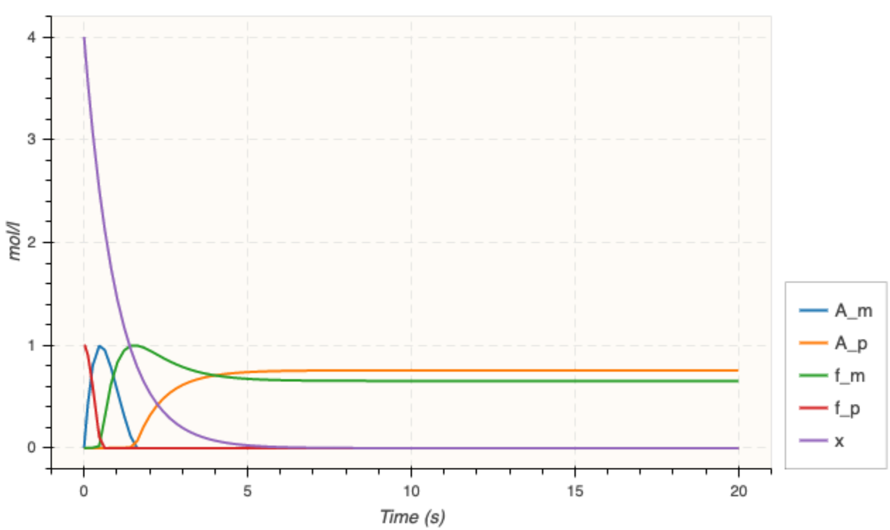
```
biocham: parameter(input=4).
biocham: compile_from_expression(cos,x,f).
initial_state(f_p=1, x=input).
MA(fast) for f_m+f_p=>_.
MA(fast) for A_m+A_p=>_.
MA(1.0) for A_p+x=>A_p+f_p+x.
MA(1.0) for A_m+x=>A_m+f_m+x.
MA(1.0) for f_m+x=>A_p+f_m+x.
MA(1.0) for f_p+x=>A_m+f_p+x.
MA(1.0) for x=>_.
```

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

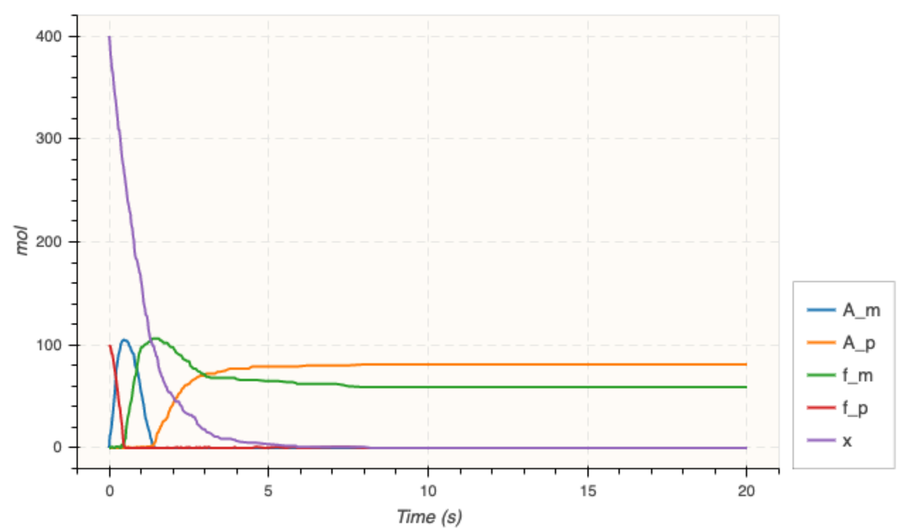
$$g'(t) = x - g(t)$$

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

ODE simulation (<u>design</u>)



Stochastic simulation (<u>test</u>)



Demo Synthetic Oscillators and Sigmoids

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

CRN synthesis for generating cosine(time)+1

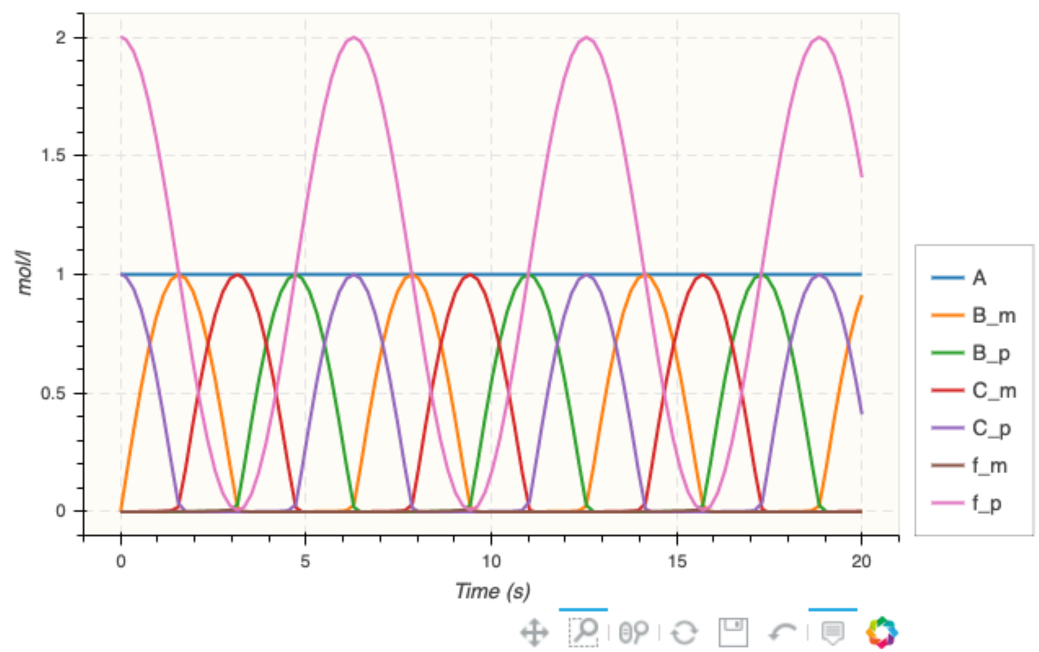
In [19]:
```
compile_from_expression(cos+1, f).
```

Out[19]:

In [20]:
```
list_model.
```

Out[20]:
```
MA(fast) for f_m+f_p=>_.
MA(fast) for C_m+C_p=>_.
MA(fast) for B_m+B_p=>_.
MA(1.0) for B_p=>B_p+C_p+f_p.
MA(1.0) for B_m=>B_m+C_m+f_m.
MA(1.0) for C_m=>B_p+C_m.
MA(1.0) for C_p=>B_m+C_p.
initial_state(f_p=2).
initial_state(C_p=1).
initial_state(A=1).
parameter(
  fast = 1000
).
```

In [21]:
```
numerical_simulation. plot.
```



TD Chemical Arithmetic

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

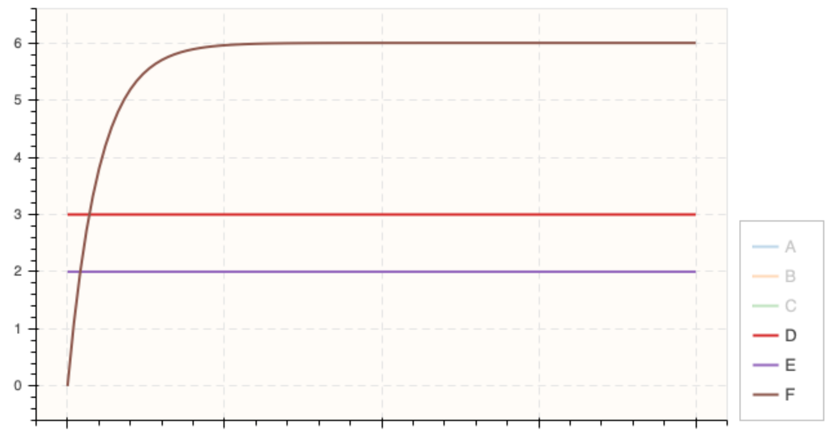
Product F = E * F

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

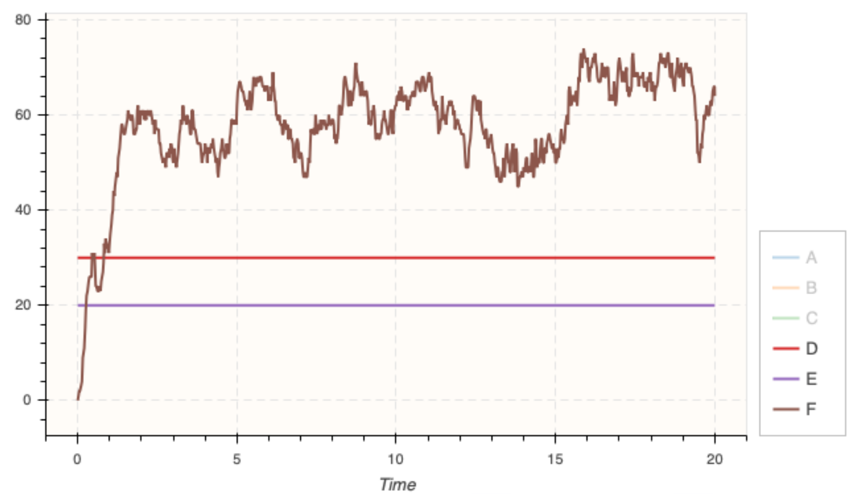
In [12]:
```
F => _.
```

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.
```



Sequentiality and Iteration

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

1. Asynchronous (precondition) CRN programming

[Huang Jiang Huang Cheng 2012 ICCAD]

[Huang Huang Chiang Jiang Fages 2013 IWBDA]

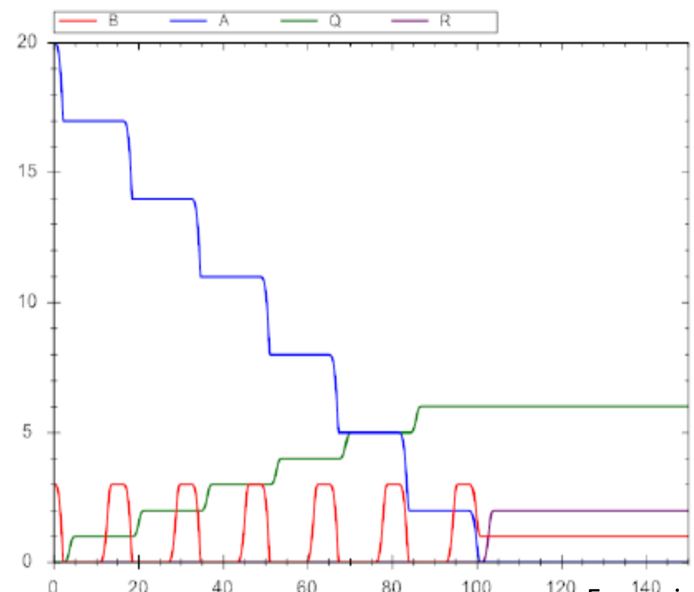
many species and reactions

2. Synchronous (clock) CRN programming

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

many reactions with the clock

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



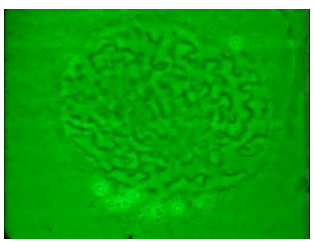
The Program of Life: Cell Division Cycle

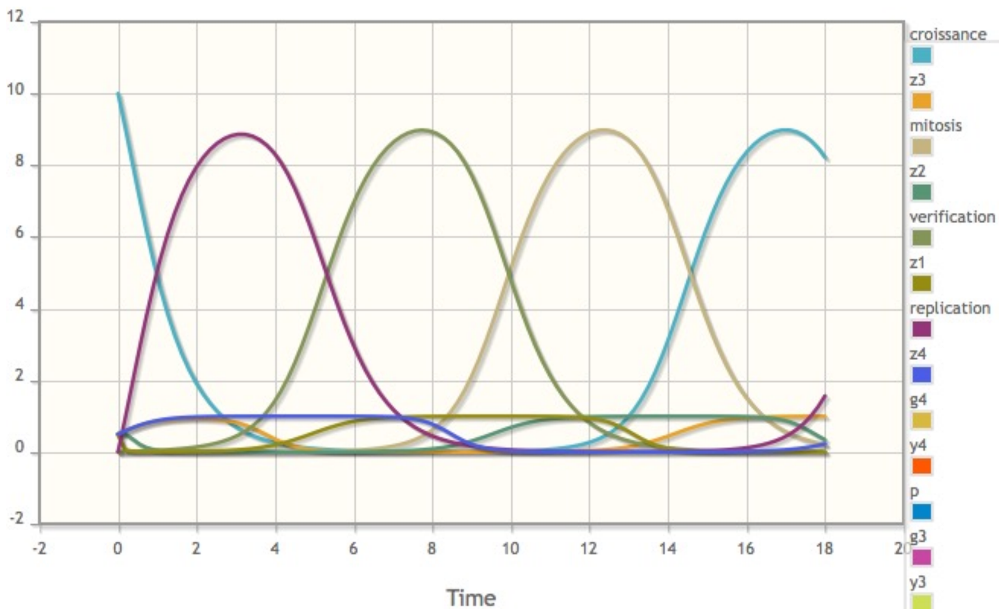
```
while true {grow; duplicate; verify/repair; separate}
```

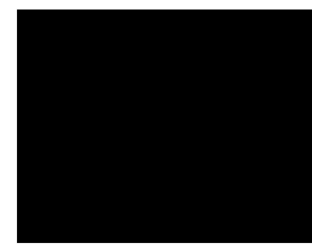
→ compilation of sequentiality and loops with program control variables

→ 50 reactions

→ 13 variables







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

On-Line Computation [Hemery F- CMSB 2022]

A CRN over m inputs X, 1 output y and n auxiliary Z, stabilizes $f : I \subset \mathbb{R}_+^m \mapsto \mathbb{R}_+$, over the domain $D \subset \mathbb{R}_+^{m+1+n}$ if:

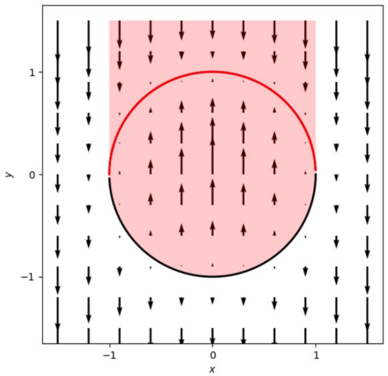
1. $\forall X^0 \in I, \{(X, y, Z) \in D | X = X^0\}$ is of plain dimension $n + 1$,
2. In the differential semantic with pinned input species X and initial conditions $X^0, y^0, Z^0 \in D$: $\lim_{t \to \infty} y(t) = f(X^0)$

Theorem. The set of functions that can be stabilized by a CRN with mass action law kinetics is the set of algebraic functions
(i.e. solution of some polynomial equation P(x, f(x))=0)

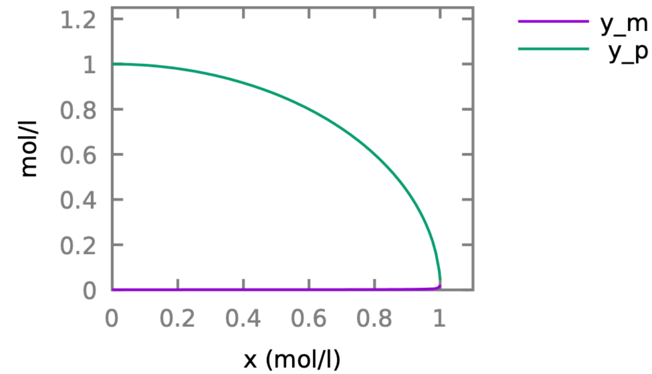
Circle Curve and Bring Radical

```
stabilize_expression(x^2 + y^2 - 1, y, [x = 0, y=1]).
```

- Circle top curve:



$$\begin{aligned} y_m + y_p &\xrightarrow{\text{fast}} \varnothing, \\ \varnothing &\rightarrow y_p, \\ 2 \cdot y_p &\rightarrow y_p, \\ 2 \cdot x &\rightarrow y_m + 2 \cdot x, \\ 2 \cdot y_m &\rightarrow 3 \cdot y_m. \end{aligned} \tag{10}$$



- Bring radical:

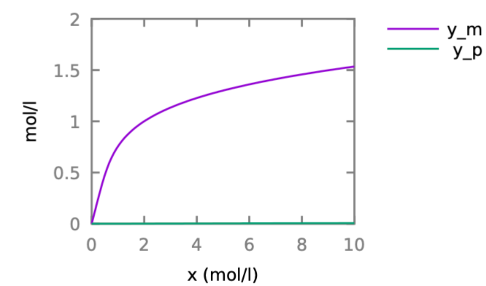
```
stabilize_expression(y^5 + y + x, y, [x = 2, y=1])
```

algebraic function

with no algebraic expression

but CRN expression …

$$\begin{array}{cc} y_m + y_p \xrightarrow{\text{fast}} \varnothing, & y2_m + y2_p \xrightarrow{\text{fast}} \varnothing, \\ y3_m + y3_p \xrightarrow{\text{fast}} \varnothing, & y_p \rightarrow \varnothing, \\ y2_m + y3_p \rightarrow y2_m + y3_p + y_p, & y2_p + y3_m \rightarrow y2_p + y3_m + y_p, \\ x \rightarrow x + y_m, & y_m \rightarrow \varnothing, \\ y2_p + y3_p \rightarrow y2_p + y3_p + y_m, & y2_m + y3_m \rightarrow y2_m + y3_m + y_m, \\ 2 \cdot y_p \rightarrow y2_p + 2 \cdot y_p, & 2 \cdot y_m \rightarrow y2_p + 2 \cdot y_m, \\ y2_p \rightarrow \varnothing, & y2_m \rightarrow \varnothing, \\ y2_p + y_p \rightarrow y2_p + y3_p + y_p, & y2_m + y_m \rightarrow y2_m + y3_p + y_m, \\ y3_p \rightarrow \varnothing, & y2_p + y_m \rightarrow y2_p + y3_m + y_m, \\ y2_m + y_p \rightarrow y2_m + y3_m + y_p, & y3_m \rightarrow \varnothing. \end{array}$$



A new kind of science
S. Wolfram

Hermite, C. Sur la résolution de l'équation du cinquème degré. Comptes Rendus de l'Académie des Sciences, 1858.

Logical Gates

Assuming concentrations in [0, 1]

And: C = A /\ B

[C] = min([A],[B]) `A+B => C` (destructive on A, B, rate-independent)

or

[C]=[A]*[B] $\frac{dC}{dt} = A * B - C$ (non-destructive stabilizing)

```
MA(k) for A+B => A+B+C
MA(k) for C => _
```
(any rate constant k but the same for both reactions)

Or: C = A \/ B

[C]=[A]+[B]-[A]*[B] $\frac{dC}{dt} = A + B - A * B - C$ (non-destructive stabilizing)

```
MA(k) for A => A+C
MA(k) for B => B+C
k*A*B for A+B+C => A+B
```
(not well-formed, should use C+ C-)

```
MA(k) for C => _
```

Not: C = ¬ A

[C]=1-[A] $\frac{dC}{dt} = 1 - A - C$

```
k for - => C
k*A for A+C => A
```
(not well-formed, should use C+ C-)

```
MA(k) for C => _
```

C = A and B

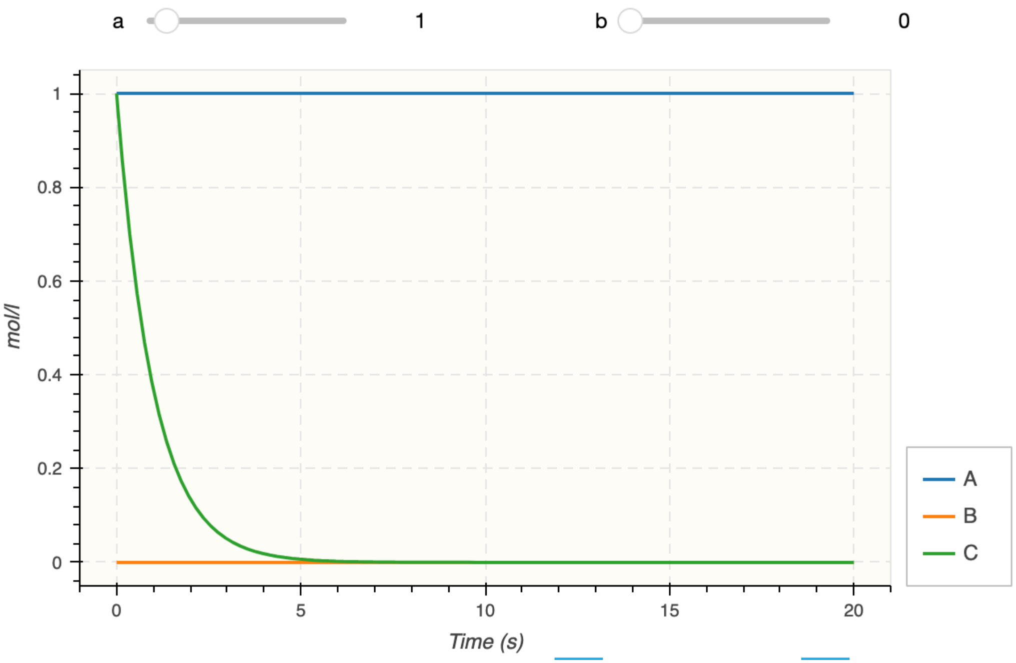
stabilize C = A * B

In [1]:
```
stabilize_expression(C-A*B, C, [A=1, B=1, C=1]).
```
Out[1]:

In [2]:
```
list_model.
```
Out[2]:
```
MA(1.0) for A+B=>A+B+C.
MA(1.0) for C=>_.
initial_state(C=1).
initial_state(A=1).
initial_state(B=1).
```

In [3]:
```
parameter(a=1, b=1).
present(A, a). present(B, b).
```

In [5]:
```
%slider a b
```

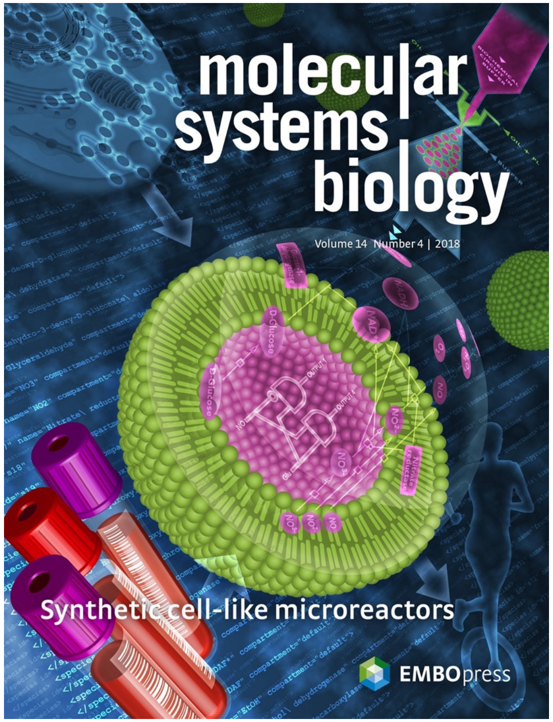


5. Validation in Artificial DNA-free RNA-free diagnostic vesicles

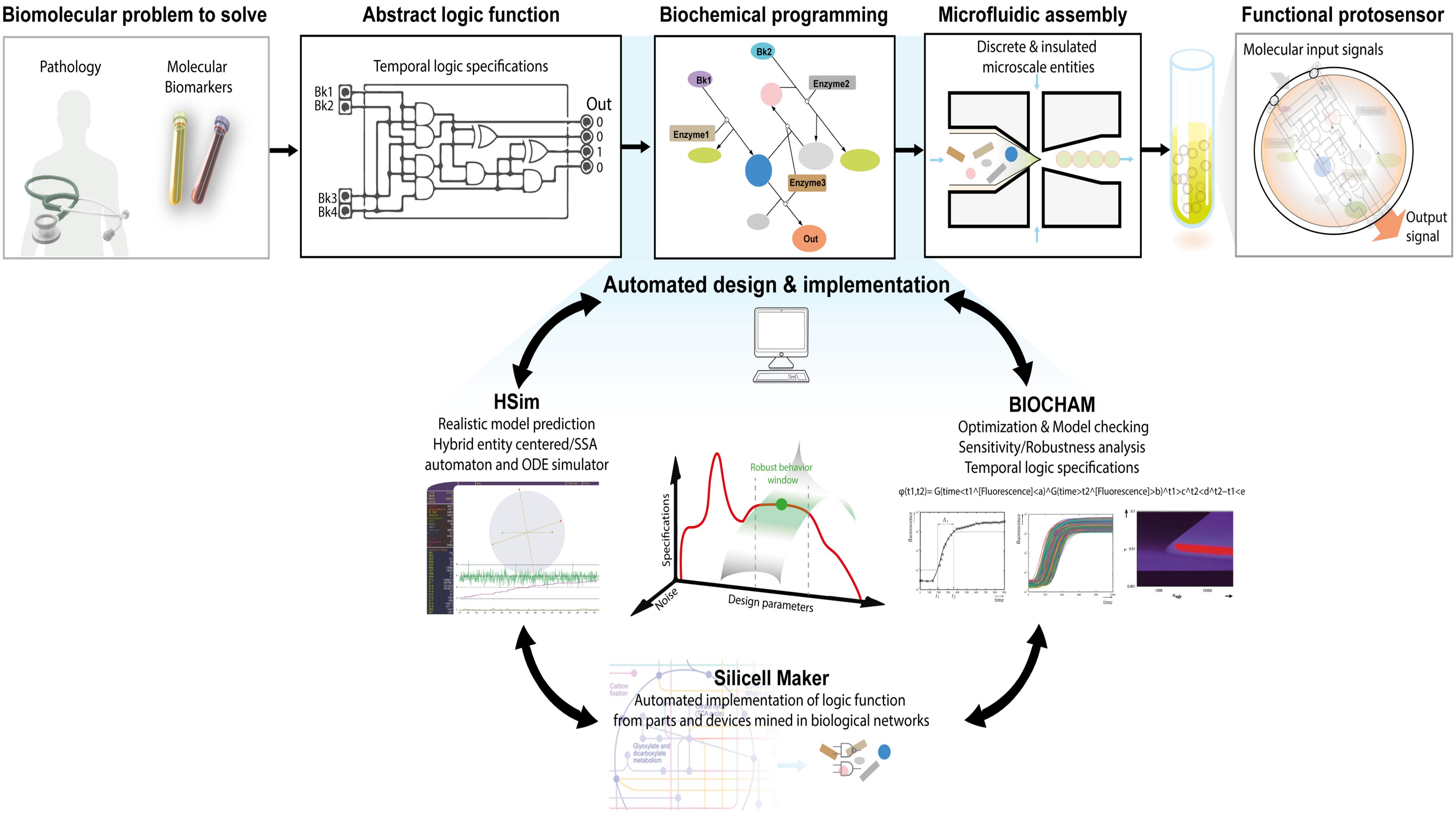
Computer-Aided Biochemical Programming of Synthetic Micro-reactors as Diagnostic Devices

Alexis Courbet [1], Patrick Amar [2], F- [3], Eric Renard [4], Franck Molina [1]

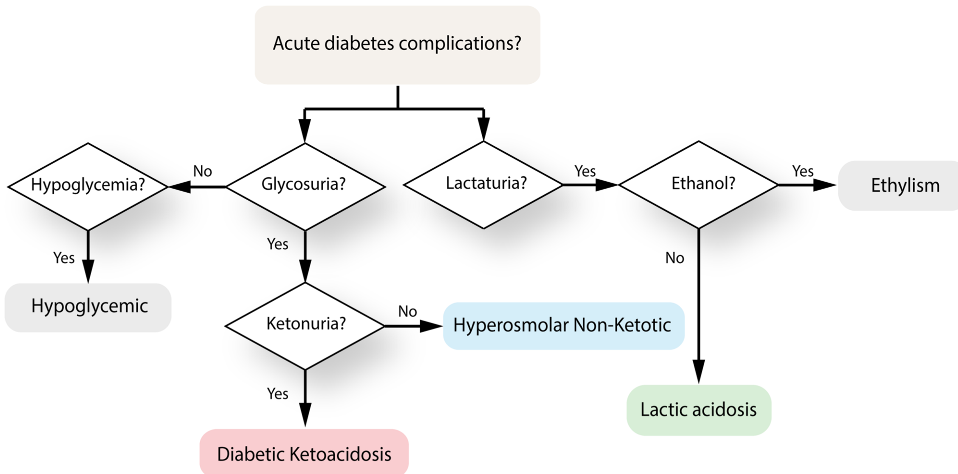
[1] Sys2diag UMR9005 CNRS/ALCEDIAG, Montpellier
[2] LRI, Université Paris Sud - UMR CNRS 8623, Orsay
[3] Inria Saclay IdF, Palaiseau
[4] INSERM 1411, Montpellier University Hospital

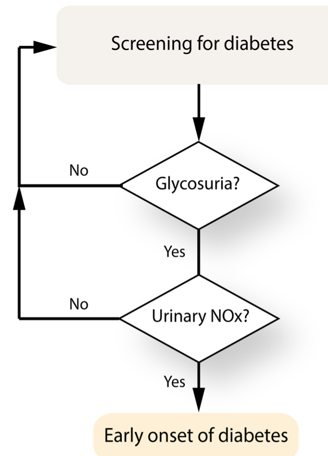


Protosensor CRN Design Workflow

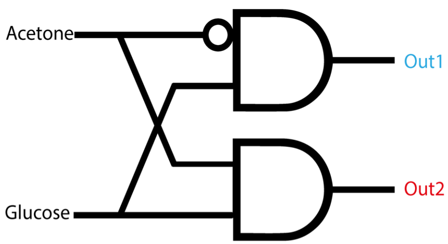


Comas Differential Diagnostic Algorithm



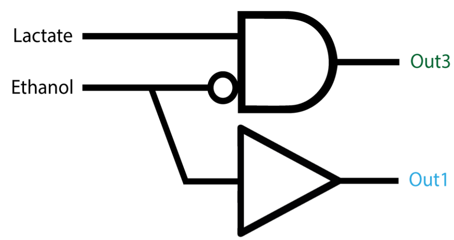


GluONe



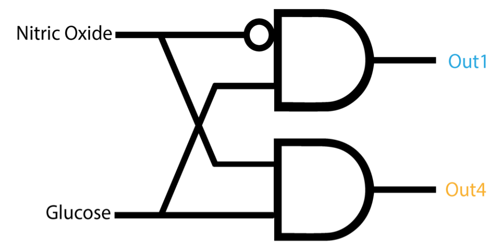
Glucose	Acetone	Out2	Out1
0	0	0	0
1	0	0	1
0	1	0	0
1	1	1	0

LacOH



Lactate	EtOH	Out3	Out1
0	0	0	0
1	0	1	0
0	1	0	1
1	1	0	1

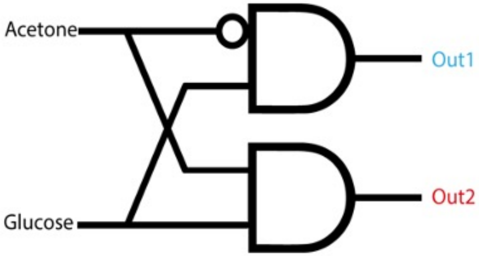
GluNOx



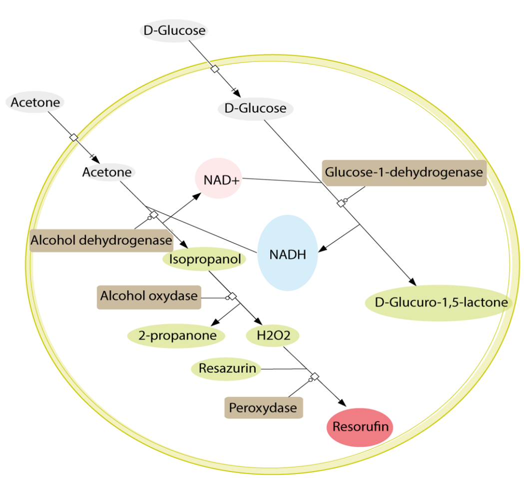
NOx	Glucose	Out4	Out1
0	0	0	0
1	0	0	0
0	1	0	1
1	1	1	0

Reactions for Implementing Logical Gates

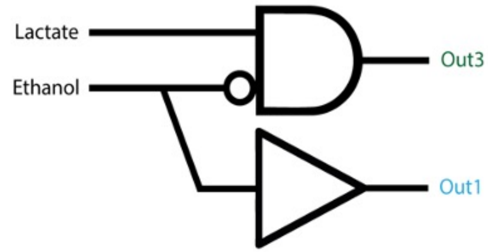
GluONe



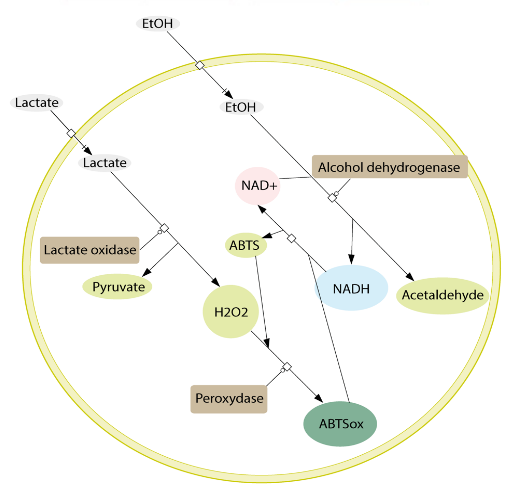
Glucose	Acetone	Out2	Out1
0	0	0	0
1	0	0	1
0	1	0	0
1	1	1	0



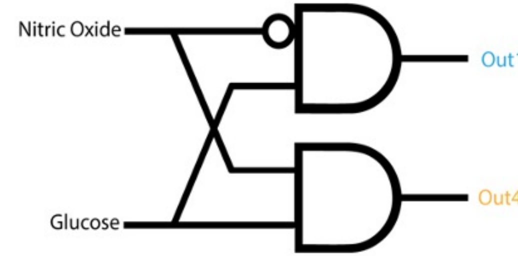
LacOH



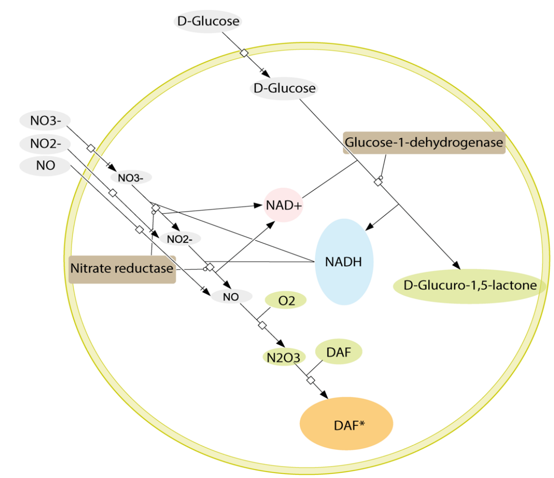
Lactate	EtOH	Out3	Out1
0	0	0	0
1	0	1	0
0	1	0	1
1	1	0	1



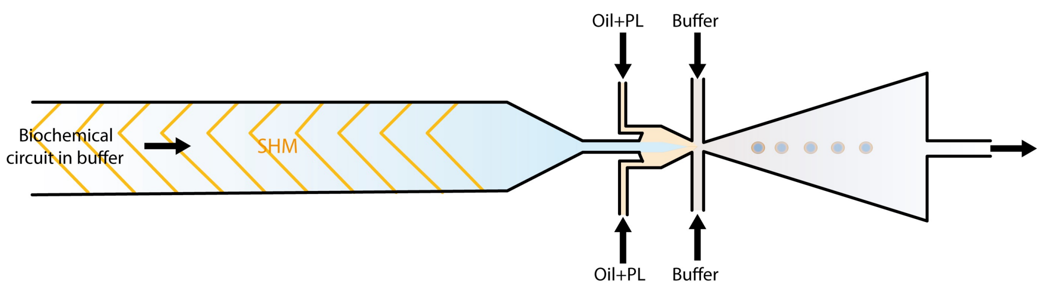
GluNOx

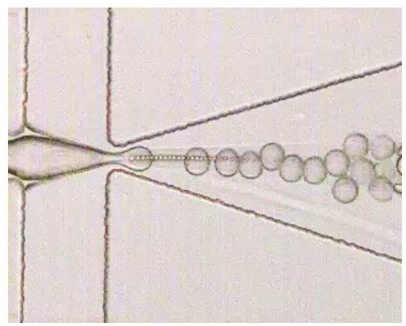


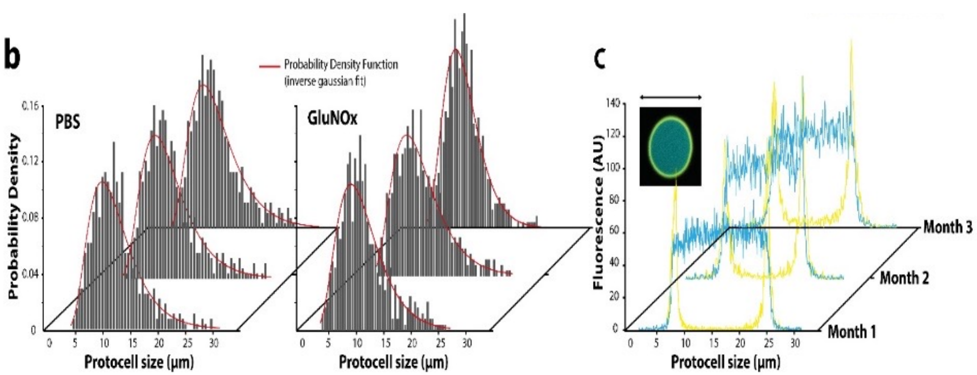
NOx	Glucose	Out4	Out1
0	0	0	0
1	0	0	0
0	1	0	1
1	1	1	0

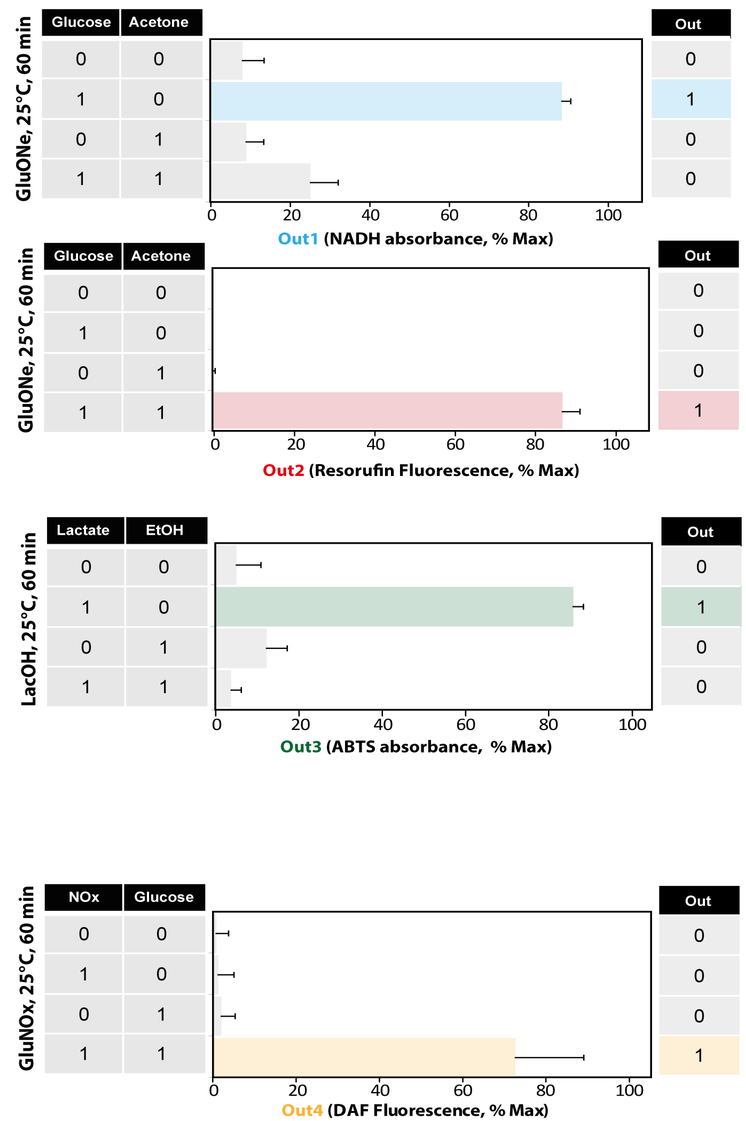


Microfluidic Assembly and Validation in Human Urine









Glucose	Acetone	Out
0	0	0
1	0	1
0	1	0
1	1	0

Glucose	Acetone	Out
0	0	0
1	0	0
0	1	0
1	1	1

Lactate	EtOH	Out
0	0	0
1	0	1
0	1	0
1	1	0

NOx	Glucose	Out
0	0	0
1	0	0
0	1	0
1	1	1

Doctor in the Cell

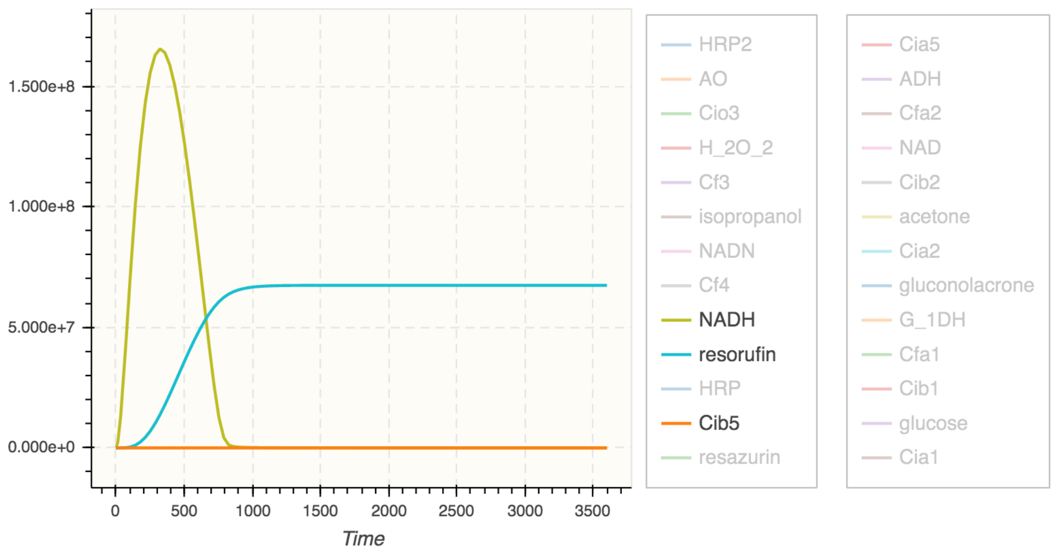
http://lifeware.inria.fr/biocham4/online/

In [9]:
```
%slider a b c d e
```

a 9.20788e+07 b 175012

c 2.2326e+06 d 3e+08

e 3e+08



In [14]:
```
seed(0). search_parameters(F(Time<T /\ G(resorufin>1e7)),
    [0 <=  a<= 1e9, 0 <= b <= 1e9, 0 <= c <= 1e9],
    [T -> 200]).
```

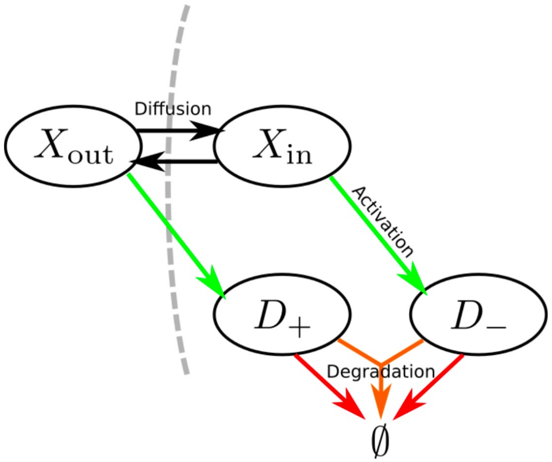
Out[14]:
```
 Time: 2.116 s
Stopping reason: Fitness: function value -9.91e-01 <= stopFitness (1.00e-04)
Best satisfaction degree: 112.773262
[0] parameter(a=560308913.9562368)
[1] parameter(b=165571523.66216615)
[2] parameter(c=296073304.37417626)
```

Differentiation CRN [Hemery F- CMSB 2023?]

The derivative of a computable function may be not computable.

The derivative of an input signal cannot be computed in arbitrary precision but can be approximated online with some delay

Max Whitby, Luca Cardelli, Marta Kwiatkowska, Luca Laurenti, Mirco Tribastone, and Max Tschaikowski. Pid control of biochemical reaction networks. *IEEE Transactions on Automatic Control*, 67(2):1023–1030, 2021.



$$X_{\mathrm{out}} \xrightarrow{k_{\mathrm{diff}}} X_{\mathrm{in}} \qquad X_{\mathrm{in}} \xrightarrow{k_{\mathrm{diff}}} X_{\mathrm{out}}$$

$$X_{\mathrm{out}} \xrightarrow{k.k_{\mathrm{diff}}} X_{\mathrm{out}} + D_+ \qquad X_{\mathrm{in}} \xrightarrow{k.k_{\mathrm{diff}}} X_{\mathrm{in}} + D_-$$

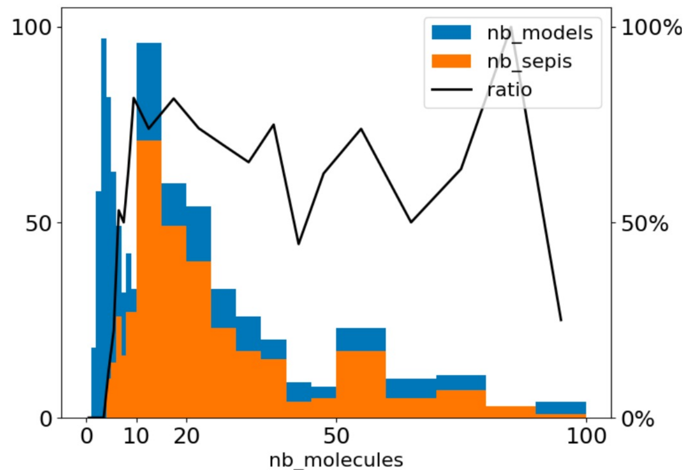
$$D_+ \xrightarrow{k} \emptyset \qquad D_- \xrightarrow{k} \emptyset$$

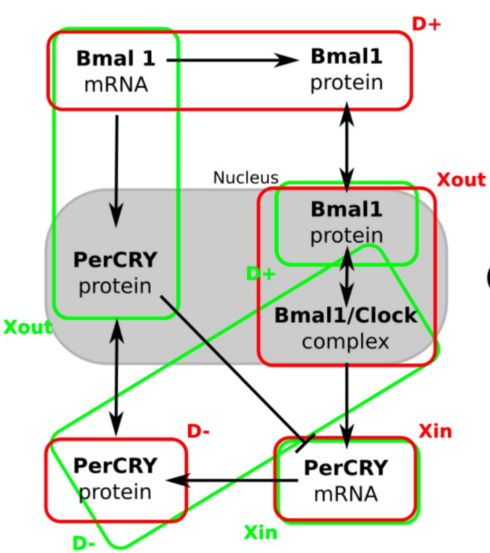
$$D_+ + D_- \xrightarrow{\mathrm{fast}} \emptyset$$

Estimation of the error can be used in some cases to correct its use in a

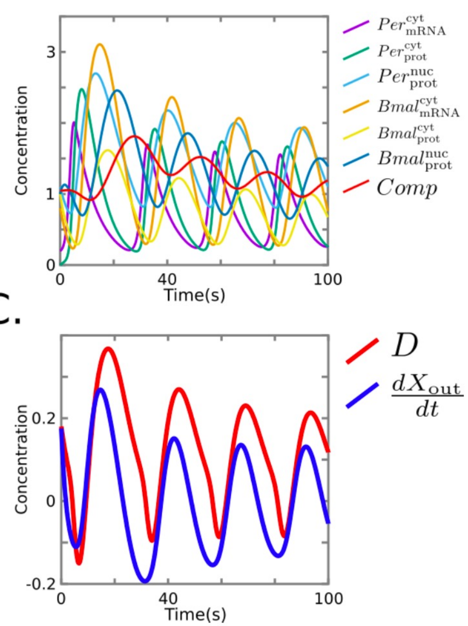
$$X_{\mathrm{in}}(t) = X_{\mathrm{out}}\,(t-\epsilon) + o(\epsilon^2) \qquad \epsilon = \frac{1}{k_{\mathrm{diff}}}$$

SEPI Search of the Derivative CRN in BioModels





C.



Model ID	# Species	# reactions	Topic
0005	6	19	Cell cycle
0010	8	20	MAPK cascade oscillations
0012	6	28	Repressilator
0021	10	60	Circadian clock
0022	10	74	Circadian clock
0034	9	47	Circadian clock
0035	9	30	Circadian clock
0041	10	32	Creatine kinase
0059	5	44	Calcium oscillation
0065	8	38	Operon lactose
0067	7	34	Circadian clock
0069	10	36	Bistable switch
0080	10	20	Inhibition of adenylate cyclase
0082	10	20	Inhibition of adenylate cyclase
0084	8	16	ERK Cascade
0099	7	28	Spontaneous Oscillations
0101	6	26	Signal Processing in TGF-β
0107	9	66	Cell cycle
0108	9	37	Superoxide dismutase overexpression
0112	10	24	Smad signalling
0116	6	34	MAPK cascade (crosstalk)
0170	7	35	Circadian clock
0171	10	60	Circadian clock
0181	6	36	Cell cycle
0185	8	39	Circadian clock
0193	8	18	Amplification and inhibition in MCC assembly
0198	9	24	Activation of guanylate cyclase by nitric oxide
0199	8	20	Catalysis and regulation in nitric-oxide synthase
0202	7	43	Calcium oscillation
0206	9	37	Circadian clock
0213	6	32	Folate pathway
0215	6	28	Regulatory Tcell
0216	5	34	Circadian clock
0221	8	40	Tricarboxylic acid cycle
0222	8	40	Tricarboxylic acid cycle
0228	9	51	Cell cycle
0229	7	28	Circadian clock
0232	7	29	Tricarboxylic acid cycle
0240	6	31	DegU transcriptional regulator
0245	6	41	Aerobic metabolism in yeast
0251	9	42	MAPK & cell fate decision
0257	8	38	Self-maintaining Metabolism
0262	9	27	AkT Signalling
0263	9	27	AkT Signalling
0269	9	42	Hormonal crosstalk in plant
0275	4	21	Bistable switch
0289	4	24	Regulatory Tcell
0290	4	24	Regulatory Tcell
0296	5	24	Ecological oscillator

Wrap-up

- Binary reaction systems over a finite set of molecules (without polymerization) are Turing-complete under the differential semantics
  - PIVP definition of computable function
  - Notion of computational complexity as trajectory length of stabilizing PIVPs

- CRN compilers [Biocham v4]
  - Input: Function specification (elementary, algebraic curve, imperative program)
  - Output: system of elementary reactions with mass action law kinetics
  - Exact characterization of the result for an ideal fluid implementation
  - Possibility to compare to natural CRNs using SEPI graph matcing

- Real implementation in artificial vesicles [Molina's lab CNRS-Alcen]

- Alternative CRN design by evolution/learning:
  - Artificial evolution of CRNs [Degrand Hemery F 2019]
  - Nature algorithms for learning [Valliant 2013 book]

CRN ⟷ Function

↻

Mutations