Temporal Constraint Solving for the Analysis of Biological Systems

Aurélien Rizk
Ph.D. defense
–
under supervision of François Fages,
Equipe Projet Contraintes, INRIA Paris-Rocquencourt
–
Ecole doctorale
Sciences mathématiques de Paris-Centre
Université Denis Diderot

Context

Systems biology : **system-level** understanding of biological systems
[Kitano. ICSB 2000]

- Biological systems are **dynamical** systems in time and space
- Need to investigate components characteristics and their **interactions**
- Tools and formalisms required for **modeling and simulation**, **control** and **design** methods.

Temporal logics approach

Temporal logics are general-purpose languages for specifying dynamical properties of discrete transition systems. [Pnueli. FOCS 1977]

- Automatic verification done by **model-checking**
- Model checking succesfully used for the verification of electronic systems and programs, can be efficient on large and complex systems
- Temporal logic adapted to high level specifications, and to **incomplete and imprecise** experimental data obtained in systems biology.

A logical Paradigm for Systems Biology

Biological Model = (Quantitative) State Transition System
Biological Properties = Temporal Logic Formulae
Automatic Validation = Model-checking

Applications of Temporal Logic in Systems Biology:

- **query language of large reaction networks** [Eker et al. PSB 02, Chabrier Fages CMSB 03] and gene regulatory network [Batt et al. Bioinformatics 05]
- **parameter search** in boolean or discrete models [Bernot et al. JTB 04] [Calzone et al. TCSB 06]
- **robustness analysis** [Batt et al. 07]

Biocham : modeling environment based on formal languages for system description and for biological properties

Temporal logics for continuous models

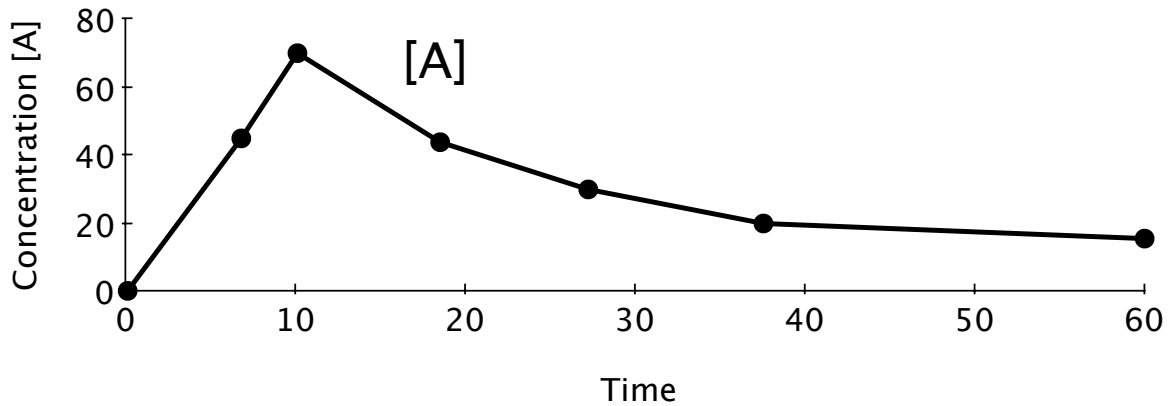
Temporal logics :

- mostly developed for discrete systems
- temporal logics with numerical constraints can deal with continuous time models (ODE or CTMC, hybrid systems)

Abstraction levels in systems biology :

- Presence/absence of molecules (boolean transitions)
- Concentration of molecules (continuous models, rates of reactions)
- Number of molecules (stochastic models, probabilities of reactions)

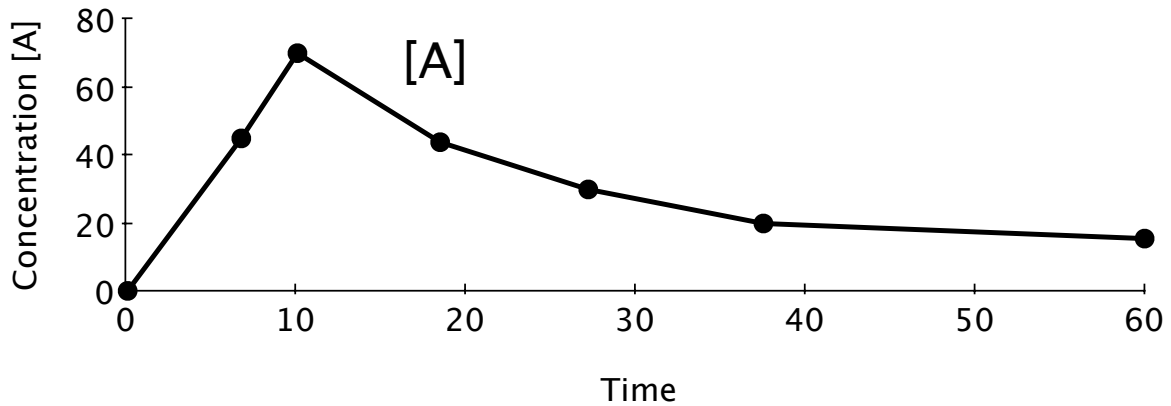
Description of temporal behaviors



Describe as:

- Numerical data time series ([A]=0 at t=0, [A]= 45 at t=7, $\cdots$)

Description of temporal behaviors



Describe as:

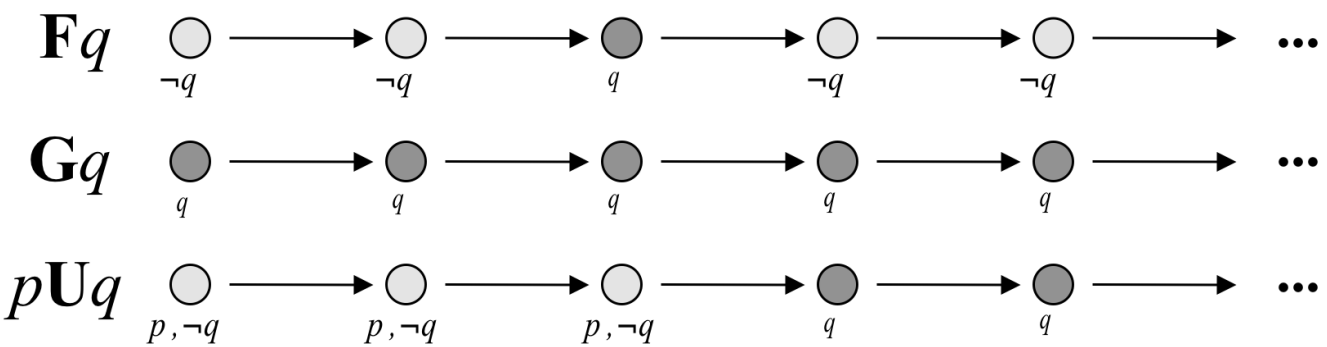
- Numerical data time series ([A]=0 at t=0, [A]= 45 at t=7, $\cdots$)
- [A] rises above threshold 60
- [A] rises above threshold 60 and then remains above 20
- [A] rises then falls (and nothing else happens)
- [A] attains a local maximum of value 70

High level properties adapted to noisy data ... but need **formalization** to be used by a computer program.

Formalize temporal properties in Linear Time Logic (LTL)

Linear Time Logic add **temporal operators** to usual logical operators $(\neg, \wedge, \vee, \rightarrow)$:

- $\mathbf{F}q$ (*finally*) : q is true at some time point in the future;
- $\mathbf{G}q$ (*globally*) : q is true at all time points in the future;
- $p\mathbf{U}q$ (*until*) : p is true until q becomes true.
- $\mathbf{X}q$ (*next*) : q is true at the next time point;



LTL($\mathbb{R}$) examples

- [A] rises above threshold 60:
  F([A]>60)
- [A] rises above threshold 60 and then remains above 20:
  F([A]>60 $\wedge$ **G**([A]> 20))
- [A] rises then falls (and nothing else happens):
  (d[A]/dt>0) **U** (**G** (d[A]/dt<0))
- [A] attains a local maximum of value 70:
  F([A] $<$ 70 $\wedge$ X([A] $=$ 70 $\wedge$ X([A] $<$ 70)))
- Numerical data time series:
  F(Time=0$\wedge$[A]=0 $\wedge$ **F**(.... $\wedge$ **F**(Time=45$\wedge$[A]=7)...))

Problem

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

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

$\longrightarrow$ need for a continuous degree of satisfaction of temporal logic formulae

How far is the system from verifying the specification ?

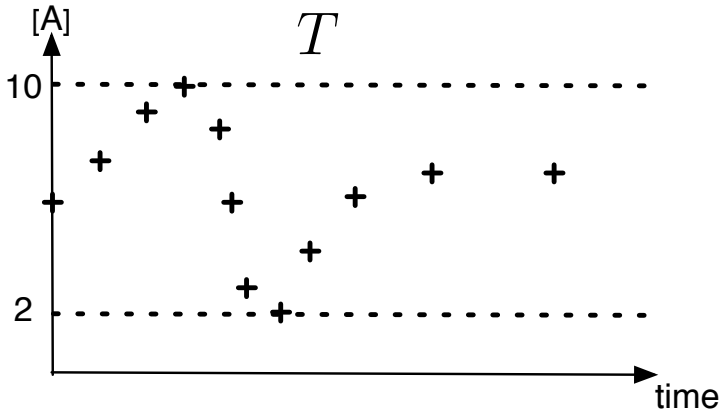
Thesis contributions

- generalization of model-checking to **temporal constraint solving** which enables definition of **continuous evaluation** of temporal logic formulas
- show how continuous valuation can be used for **quantitative analysis** in systems biology and that it enables parameter search in high dimension and robustness analysis of temporal properties
- implementation of methods in Biocham modeling environment
- application to biological problems

Outline

Validity domain of free variables in LTL($\mathbb{R}$) formulae

Evaluation of temporal logic formulae on numerical traces



$LTL(\mathbb{R})$

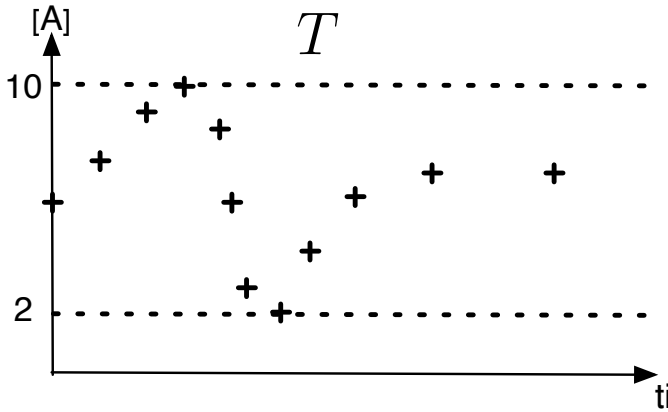
Φ=F([A]≥7
∧F([A]≤0))

Model-checking

the formula is false

Validity domain of free variables in LTL($\mathbb{R}$) formulae

Evaluation of temporal logic formulae on numerical traces



$LTL(\mathbb{R})$

Φ=F([A]≥7 ∧F([A]≤0))

Model-checking

the formula is false

$QFLTL(\mathbb{R})$

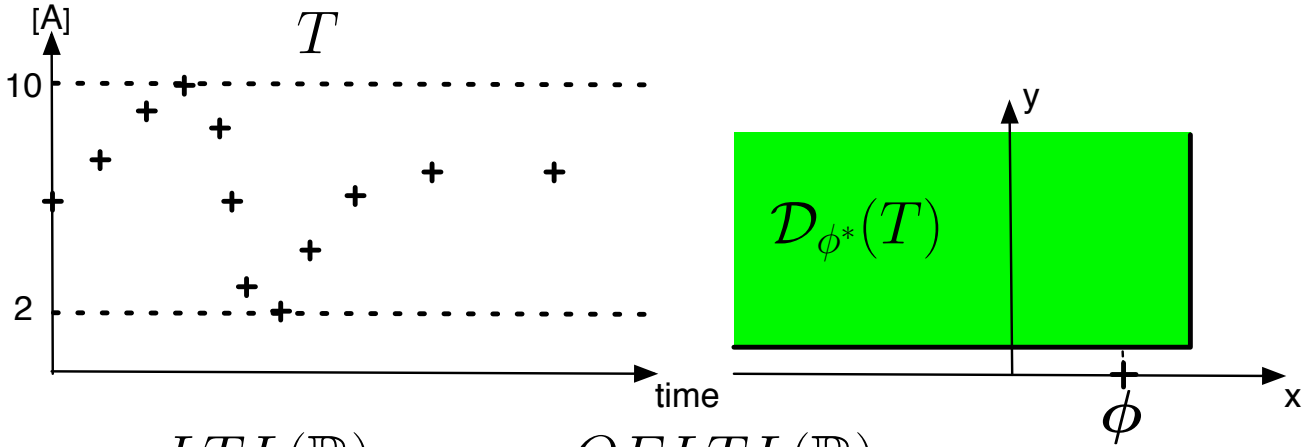
Φ*=F([A]≥x ∧F([A]≤y))

Constraint solving

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

Validity domain of free variables in LTL($\mathbb{R}$) formulae

Evaluation of temporal logic formulae on numerical traces



$LTL(\mathbb{R})$

Φ=F([A]≥7 ∧F([A]≤0))

Model-checking

the formula is false

$QFLTL(\mathbb{R})$

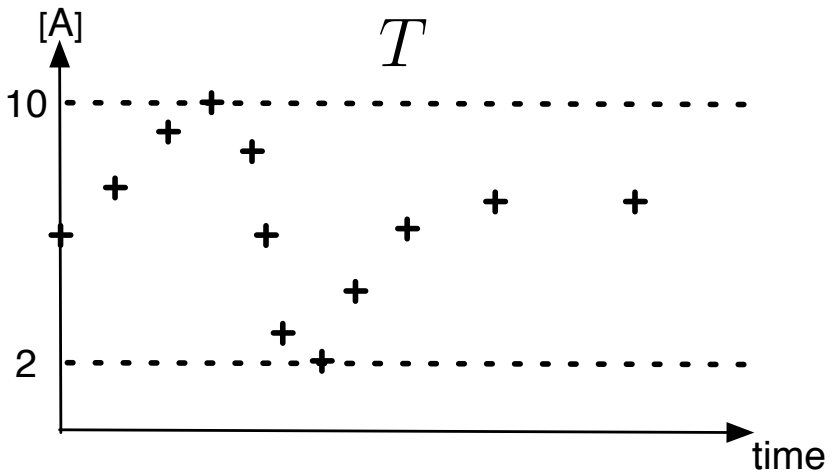
Φ*=F([A]≥x ∧F([A]≤y))

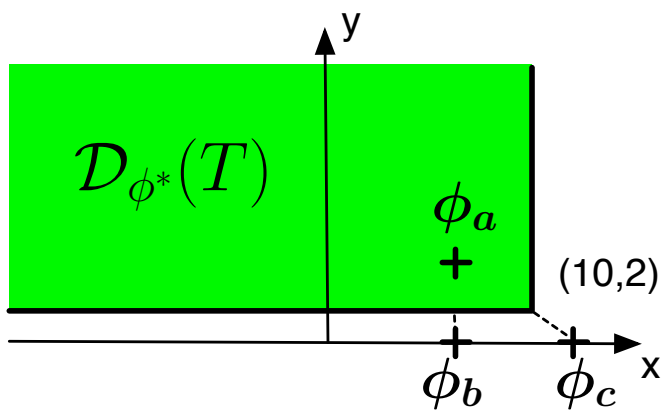
Constraint solving

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

Validity domain $\mathcal{D}_{\phi^*}(T)$: set of values of the variables in a LTL($\mathbb{R}$) formula making it true on a given trace T.

Violation and satisfaction degree of an LTL($\mathbb{R}$) formula





$\phi^*(x,y) = F([A] \geq x \wedge F([A] \leq y))$

ϕ_a = F([A]≥6 ∧ F([A]≤5))

ϕ_b = F([A]≥7 ∧ F([A]≤0))

ϕ_c = F([A]≥12 ∧ F([A]≤0))

$\phi^*_{(6,5)}$

$\phi^*_{(7,0)}$

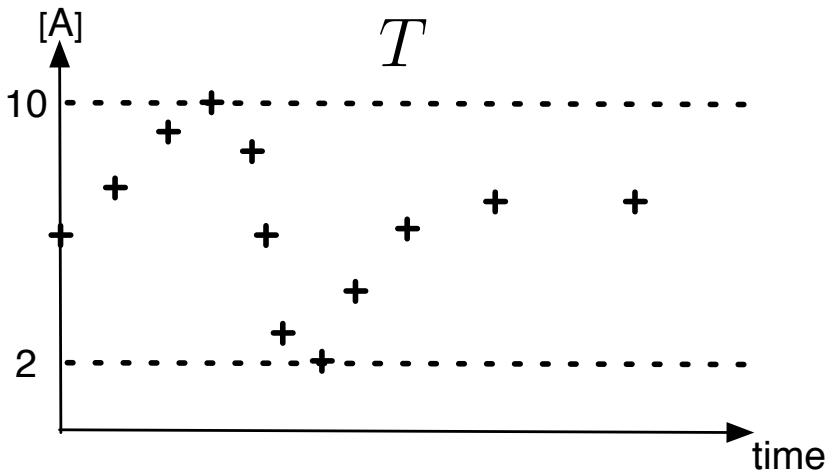
$\phi^*_{(12,0)}$

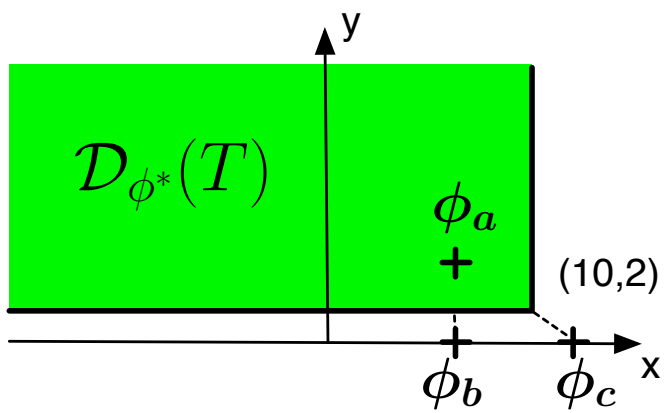
Violation and satisfaction degree of an LTL($\mathbb{R}$) formula

Definition of violation degree $vd(T,\phi)$ and satisfaction degree $sd(T,\phi)$

In the variable space of ϕ^*, original formula ϕ is single point $var(\phi)$.

$vd(T,\phi) = min_{v \in D_{\phi^*}(T)} d(v, var(\phi))$ $\quad$ $sd(T,\phi) = \frac{1}{1+vd(T,\phi)} \in [0,1]$





$\phi^*(x,y) = F([A] \geq x \wedge F([A] \leq y))$

ϕ_a = F([A]≥6 ∧ F([A]≤5))

ϕ_b = F([A]≥7 ∧ F([A]≤0))

ϕ_c = F([A]≥12 ∧ F([A]≤0))

$\phi^*_{(6,5)}$	vd=0	sd=1	(✓)
$\phi^*_{(7,0)}$	vd=2	sd=0.33	(×)
$\phi^*_{(12,0)}$	vd=2√2	sd=0.26	(×)

Computation of validity domain $\mathcal{D}_{\phi^*}(T)$

Algorithm (Computation by induction on ϕ subformulae)

- $\mathcal{D}_{s_i,\alpha} = \{\mathbf{v} \in \mathbb{R}^k \mid s_i \models \alpha[\mathbf{v}/\mathbf{x}]\}$ *for an atomic proposition* α,
- $\mathcal{D}_{s_i,\phi\wedge\psi} = \mathcal{D}_{s_i,\phi} \cap \mathcal{D}_{s_i,\psi}$,

- $\mathcal{D}_{s_i,\mathbf{F}\phi} = \cup_{j\geq i}\mathcal{D}_{s_j,\phi}$,

Computation of validity domain $\mathcal{D}_{\phi^*}(T)$

Algorithm (Computation by induction on ϕ subformulae)

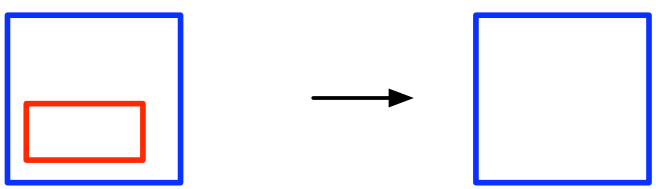
- $\mathcal{D}_{T,\phi} = \mathcal{D}_{s_0,\phi}$,
- $\mathcal{D}_{s_i,\alpha} = \{\mathbf{v} \in \mathbb{R}^k \mid s_i \models \alpha[\mathbf{v}/\mathbf{x}]\}$ *for an atomic proposition* α,
- $\mathcal{D}_{s_i,\phi\wedge\psi} = \mathcal{D}_{s_i,\phi} \cap \mathcal{D}_{s_i,\psi}$,
- $\mathcal{D}_{s_i,\phi\vee\psi} = \mathcal{D}_{s_i,\phi} \cup \mathcal{D}_{s_i,\psi}$,
- $\mathcal{D}_{s_i,\mathbf{X}\phi} = \mathcal{D}_{s_{i+1},\phi}$,
- $\mathcal{D}_{s_i,\mathbf{F}\phi} = \cup_{j\geq i}\mathcal{D}_{s_j,\phi}$,
- $\mathcal{D}_{s_i,\mathbf{G}\phi} = \cap_{j\geq i}\mathcal{D}_{s_j,\phi}$,
- $\mathcal{D}_{s_i,\phi\mathbf{U}\psi} = \cup_{j\geq i}(\mathcal{D}_{s_j,\psi} \cap \cap_{k\in[i,j-1]}\mathcal{D}_{s_k,\phi})$.

$\longrightarrow$ computation done by finite unions and intersections of domains.

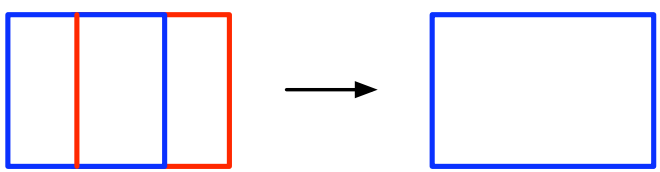
Algorithm computing $\mathcal{D}_{\phi^*}(T)$ [Fages Rizk TCS 08] implemented in Biocham

Validity domain representation

- when at most one variable per atomic formula : domain is a finite union of **orthotopes**
- when linear constraints on variables : domain is a finite union of **polyhedra**
- intersections and unions of domains computed with the **Parma Polyhedra Library**
- domain simplification rules :



Omega-reduction



Pairwise merging

Temporal constraint solving algorithm

Algorithm implemented in Biocham in gnu-prolog.

Strong completeness

The temporal constraint solving algorithm is correct and complete : a valuation $\vec{v}$ makes ϕ true at time t_i, $T, t_i \models_{LTL} (\phi(\vec{v}))$, if and only if $\vec{v}$ is in the computed domain of ϕ at t_i, $\vec{v} \in \mathcal{D}_\phi(t_i)$.

Complexity

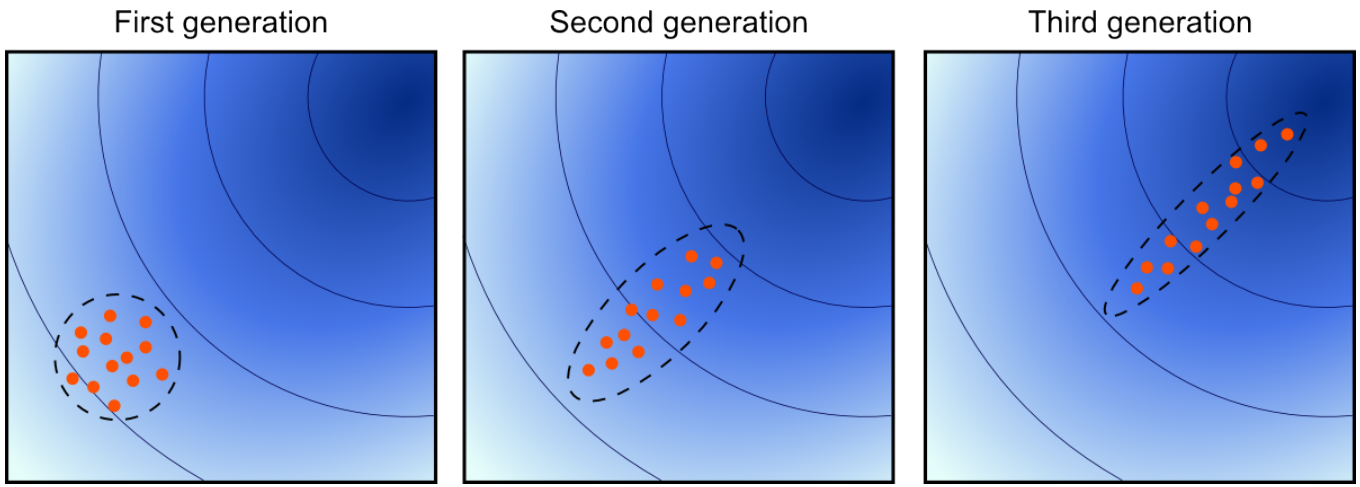
Validity domain of size f formula containing k variables on length n trace at most (i) $(nf)^{2k}$ when at most one variable is present per atomic formula and (ii) 2^{nf} otherwise

Temporal constraint solving algorithm for Computation Tree Logic (CTL) in [Fages Rizk CP 09]

Using violation degree as cost function [Rizk et al. TCS 09]

- Use existing optimization toolbox for kinetic parameter search using violation degree as cost function
- Use state-of-the-art Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [Hansen Osermeier 01, Hansen 08]
- CMA-ES minimizes an objective function in continuous domain in a black box scenario :

- CMA-ES uses a probabilistic neighborhood and updates information in covariance matrix at each move



Robustness Measure Definition [Rizk et al. ISMB 09]

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]:

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

Robustness Measure Definition [Rizk et al. ISMB 09]

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]:

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

- Our computational measure of robustness w.r.t. LTL($\mathbb{R}$) spec: Given an ODE model with initial conditions, a TL formula ϕ and a set of perturbations P (on initial conditions or parameters),

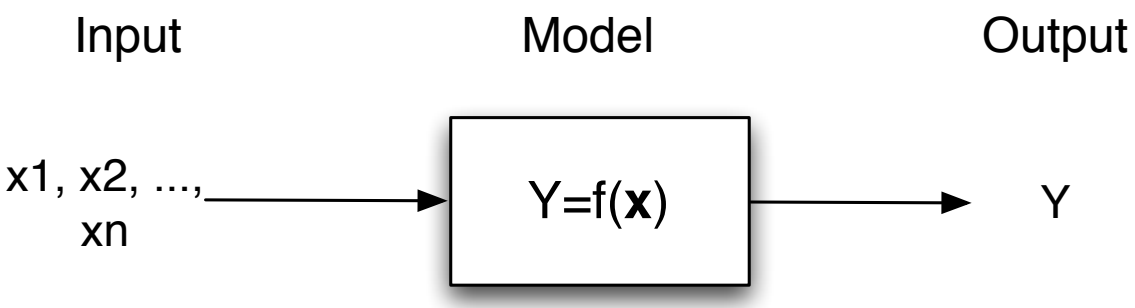
$$\mathcal{R}_{\phi,P} = \int_{p \in P} sd(T(p), \phi)\ prob(p)\ dp$$

$\longrightarrow$ evaluate mean behavior of a system subject to noise, compare robustness of different designs, use robustness as optimization objective

Sensitivity analysis

Sensitivity analysis

Study how the **variation in the output** can be apportioned to different **sources of variation**. [Saltelli 2000]



Instantiate with satisfaction degree of LTL formulas as output and biological model parameters (kinetic, initial conditions) as input

Global sensitivity analysis

Global SA (*Morris screening, Sobol variance based methods*) : accounts for the whole range of possible parameter variation, sensitivity of individual parameters evaluated while varying all other parameters as well.

Sensitivity indices

Evaluate quantitative impact of factors on output by estimating :

$$S_i = \frac{V(E[Y|X_i])}{V(Y)}$$

Application of continuous satisfaction degree

Continuous satisfaction degree benefits :

- parameter optimization : **efficient** search (compared to boolean evaluation) with respect to **flexible** high level specifications (compared to curve-fitting)

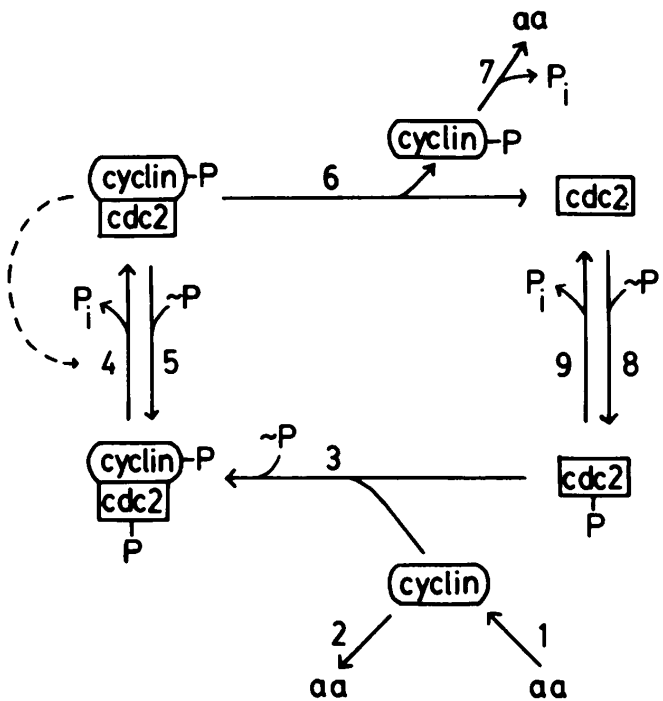
- robustness and sensitivity analysis : **generic** computational method with respect to high level specifications

Applications to biological problems

- Yeast cell cycle model : **oscillatory behavior analysis** in parameter space
- Optimization of cancer therapeutic schedule : **control problem**
- Synthetic biology in *E. Coli* : **design optimization**

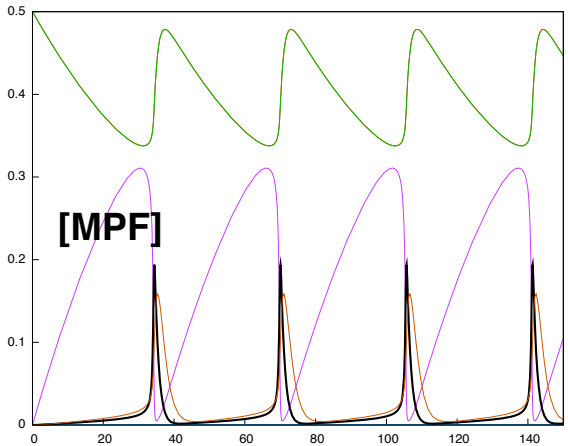
Yeast cell cycle model [Tyson PNAS 91]

- ODE model of the yeast cell cycle (6 variables, 8 kinetic parameters)
- models Cdc2 and Cyclin interactions, exhibits sustained oscillations



$\longrightarrow$ is it possible to change oscillations characteristics ? is robustness the same everywhere in kinetic parameter space ?

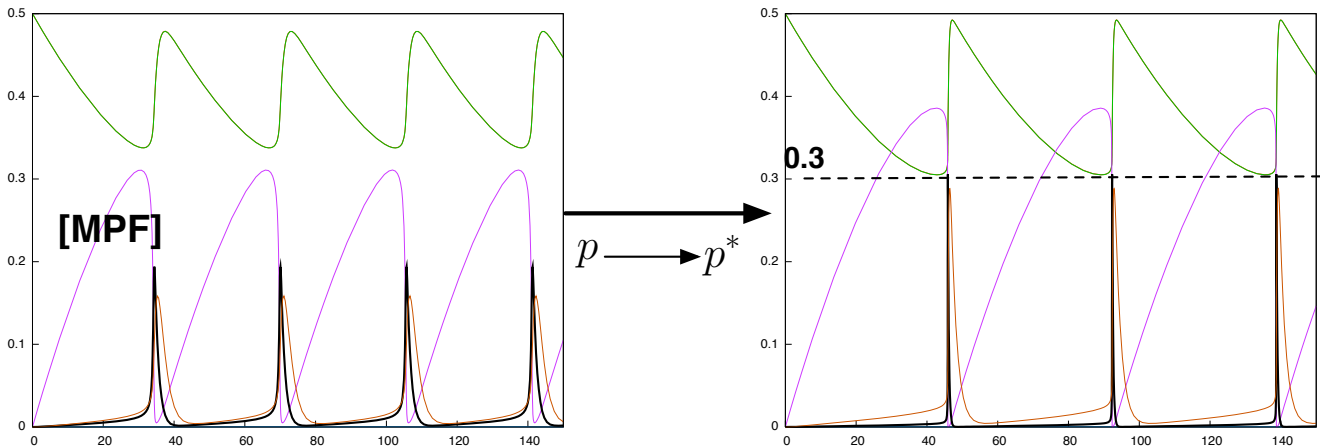
Learning kinetic parameter values from LTL specifications



- Pb : find values of 8 parameters such that amplitude is ≥ 0.3
  ϕ^*: **F**([MPF]>x $\wedge$ **F**([MPF]<y)) $\wedge$ x-y $>$ z

  amplitude x-y
  goal : z $= 0.3$

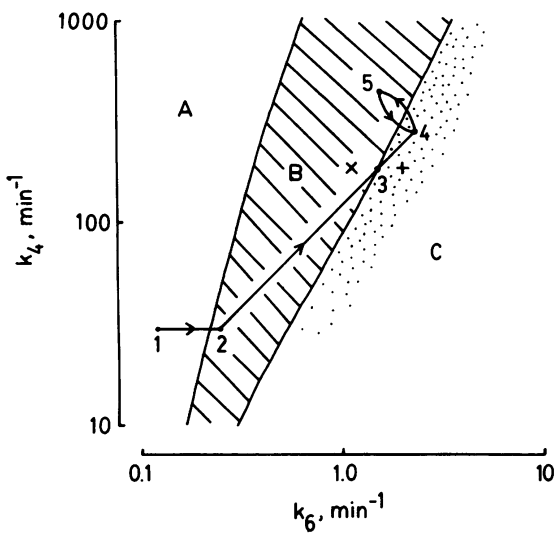
Learning kinetic parameter values from LTL specifications



- Pb : find values of 8 parameters such that amplitude is ≥ 0.3
  ϕ^*: **F**([MPF]>x $\wedge$ **F**([MPF]<y)) $\wedge$ x-y $>$ z

  amplitude x-y
  goal : z $= 0.3$
- $\longrightarrow$ solution found after 30s (100 calls to the fitness function)

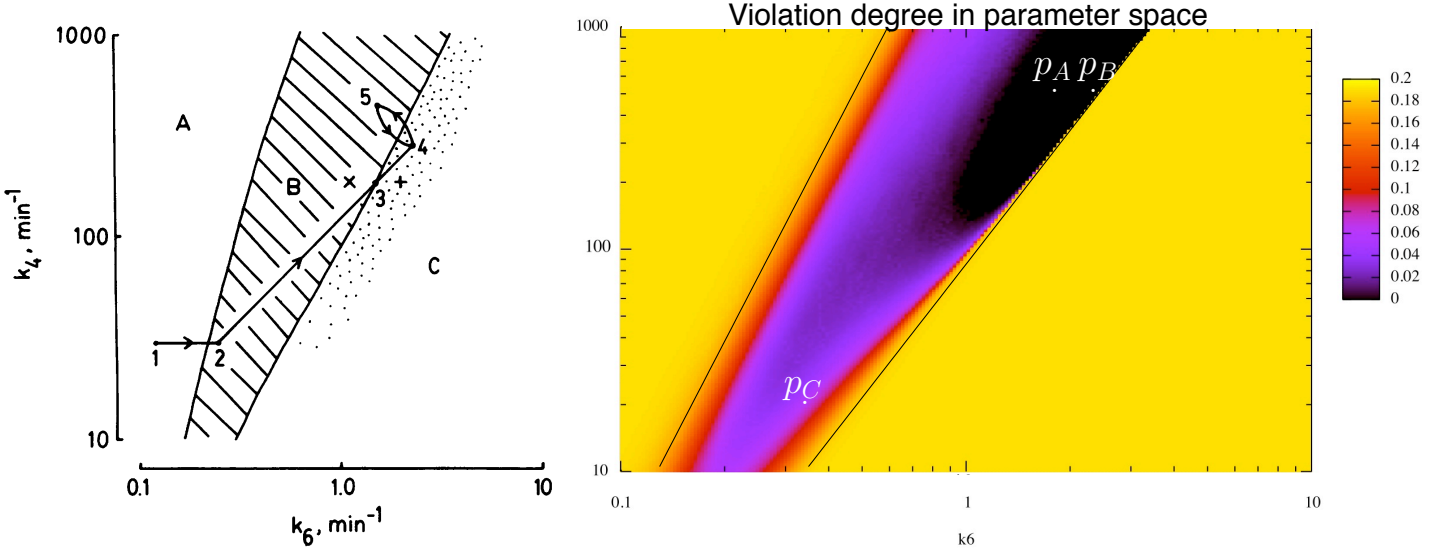
Robustness analysis w.r.t parameter perturbations



Robustness analysis w.r.t parameter perturbations

- cell cycle model [Tyson PNAS 91]
- oscillation of at least 0.2

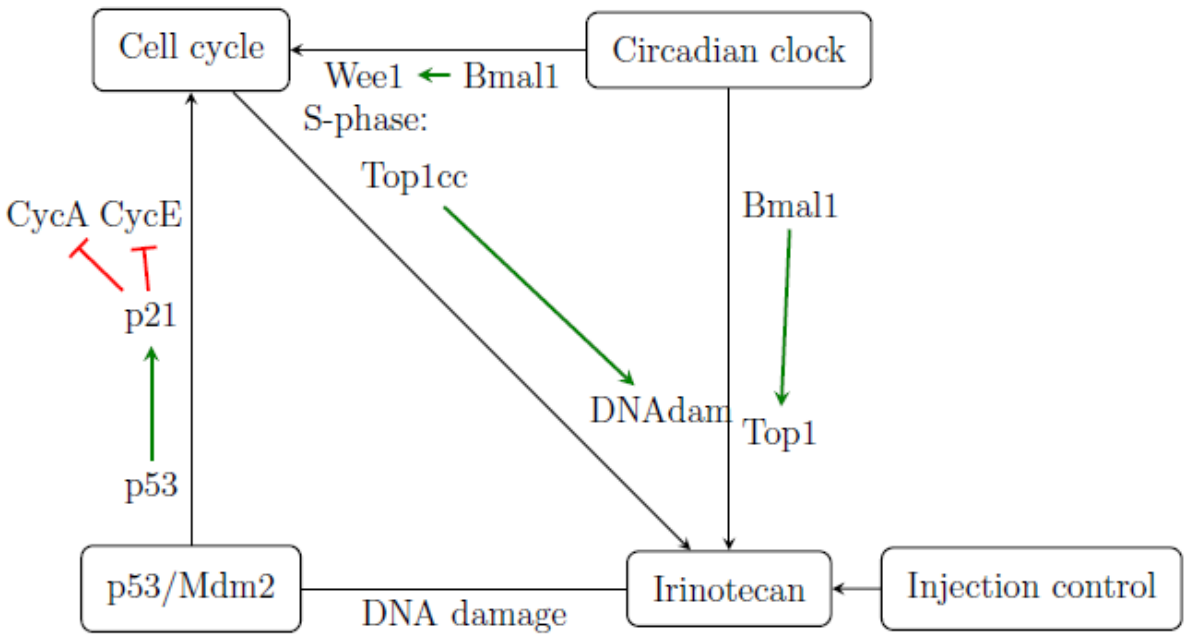
  ϕ^*: F([MPF]>x $\wedge$ F([MPF]<y)) $\wedge$ x-y > z; amplitude z=0.2
- parameters normally distributed, $\mu = p_{ref}$, CV=0.2



$\mathcal{R}_{\phi,p_A} = 0.991,, \mathcal{R}_{\phi,p_B} = 0.917, \mathcal{R}_{\phi,p_C} = 0.932$

Coupled Models of Cell Cycle, Circadian Clock, DNA repair

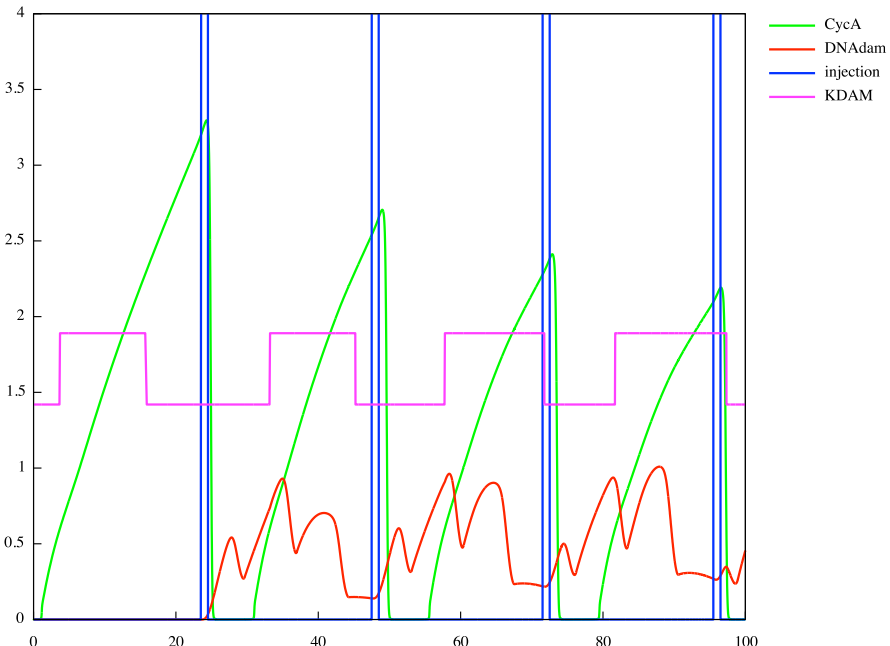
- Context of colorectal cancer chronotherapies (collab. INSERM France, EU Tempo, coord. F. Lévi INSERM Villejuif France)
- Coupled model of the cell cycle [Tyson Novak 04] and the circadian clock [Leloup Goldbeter 99] with DNA repair system p53/Mdm2 and effect of irinotecan anticancer drug



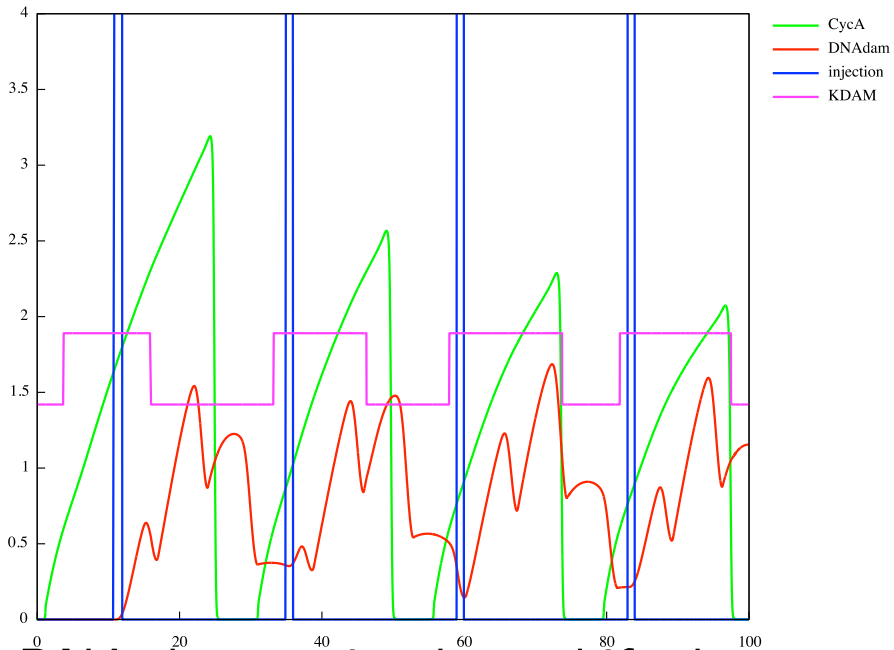
→ coupled models built by finding kinetic parameter values such that period of cell cycle entrained by circadian clock

Optimize therapeutic schedule

Optimize therapeutic schedule by maximizing satisfaction degree of :
G([DNA damage]<v1) in healthy cells
F([DNA damage]>v2) in phase shifted cells



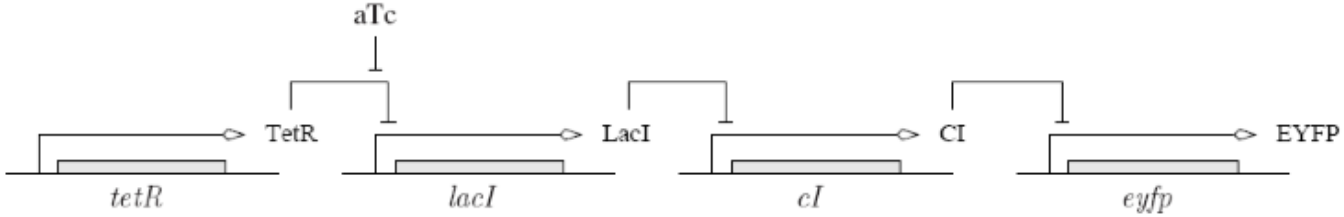
Optimized schedule for minimum DNA damage in healthy cells



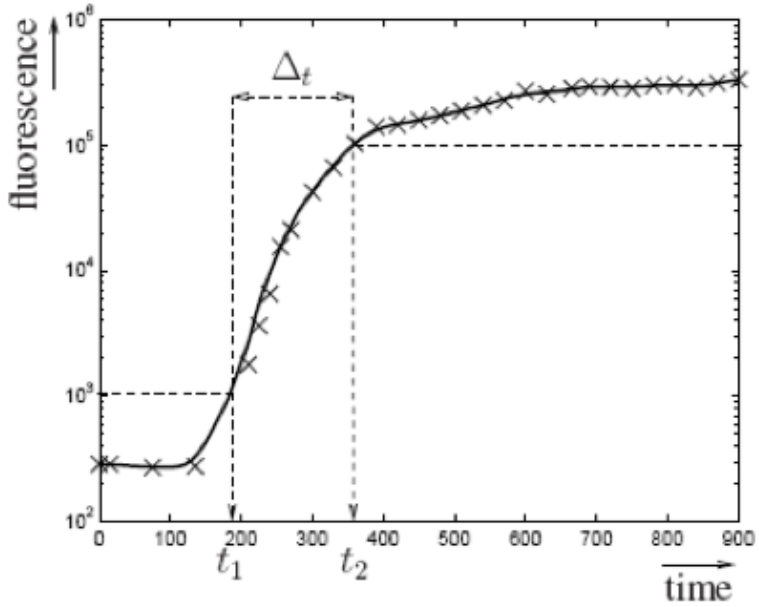
DNA damage in phase-shifted cells

Application to Synthetic Biology in *E. Coli* [Rizk et al. ISMB 09]

Cascade of transcriptional inhibitions implemented in *E.coli* [Weiss et al PNAS 05]



The output protein EYFP is controlled by the small input molecule aTc



The system is well-timed if EYFP remains below 10^3 for at least 150 min., then exceeds 10^5 after at most 450 min., and switches from low to high levels in less than 150 min.

Specifying the expected behavior in LTL($\mathbb{R}$)

The timing specifications can be formalized in temporal logic as follows:

$$\begin{aligned}\phi(t_1, t_2) = \quad & \mathbf{G}(\mathit{time} < t_1 \rightarrow [\mathtt{EYFP}] < 10^3) \\ \wedge \quad & \mathbf{G}(\mathit{time} > t_2 \rightarrow [\mathtt{EYFP}] > 10^5) \\ \wedge \quad & t_1 > 150 \wedge t_2 < 450 \wedge t_2 - t_1 < 150\end{aligned}$$

which is abstracted into

$$\begin{aligned}\phi(t_1, t_2, b_1, b_2, b_3) = \quad & \mathbf{G}(\mathit{time} < t_1 \rightarrow [\mathtt{EYFP}] < 10^3) \\ \wedge \quad & \mathbf{G}(\mathit{time} > t_2 \rightarrow [\mathtt{EYFP}] > 10^5) \\ \wedge \quad & t_1 > b1 \wedge t_2 < b_2 \wedge t_2 - t_1 < b_3\end{aligned}$$

for computing validity domains for b_1, b_2, b_3
with the objective $b_1 = 150, b_2 = 450, b_3 = 150$ for computing the satisfaction degree in a given trace.

ODE model and perturbation model

- ODE model with Hill functions :

$$\dot{x}_{tetR} = \kappa_{tetR} - \gamma_{tetR}\, x_{tetR} \qquad \dot{x}_{lacI} = \kappa_{lacI} \frac{\theta_{tetR}^{\eta_{tetR}}}{\theta_{tetR}^{\eta_{tetR}} + x_{tetR}^{\eta_{tetR}} \frac{\theta_{aTc}^{\eta_{aTc}}}{\theta_{aTc}^{\eta_{aTc}} + u_{aTc}^{\eta_{aTc}}}} - \gamma_{lacI}\, x_{lacI}$$

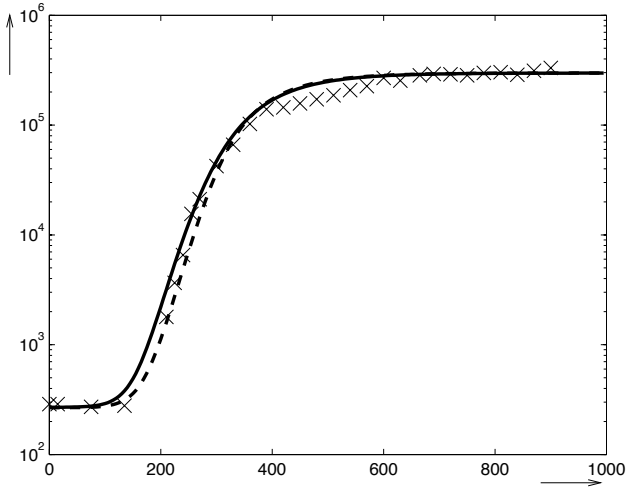
$$\dot{u}_{aTc} = 0$$

$$\dot{x}_{cI} = \kappa_{cI} \frac{\theta_{lacI}^{\eta_{lacI}}}{\theta_{lacI}^{\eta_{lacI}} + x_{lacI}^{\eta_{lacI}}} - \gamma_{cI}\, x_{cI} \qquad \dot{x}_{eyfp} = \kappa_{eyfp} \frac{\theta_{cI}^{\eta_{cI}}}{\theta_{cI}^{\eta_{cI}} + x_{cI}^{\eta_{cI}}} - \gamma_{eyfp}\, x_{eyfp}$$

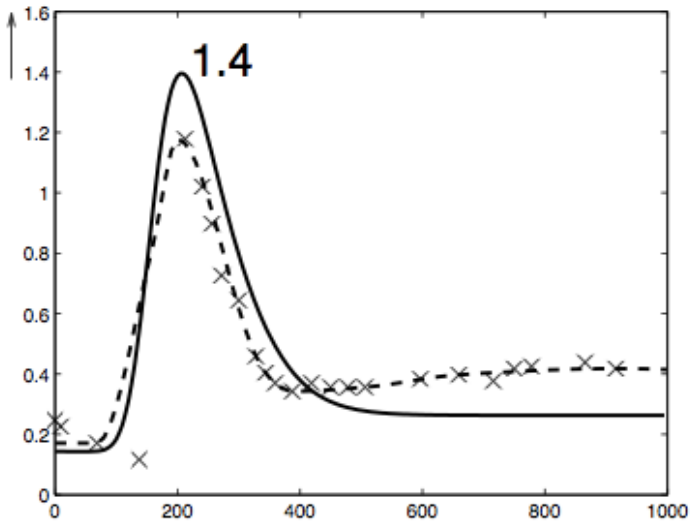
- Perturbation model : (log-)normally distributed parameters

$$\dot{x} = f(x,q), \text{with } q \sim LogN(p, \sigma^2 p)$$

Improving local robustness



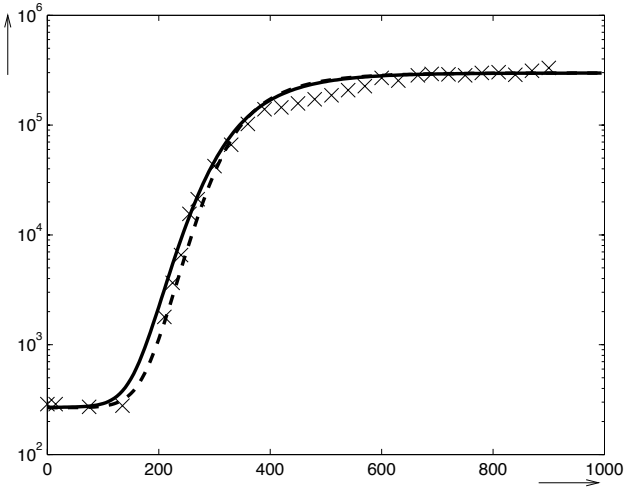
Average behavior of the system (5000 simulations)



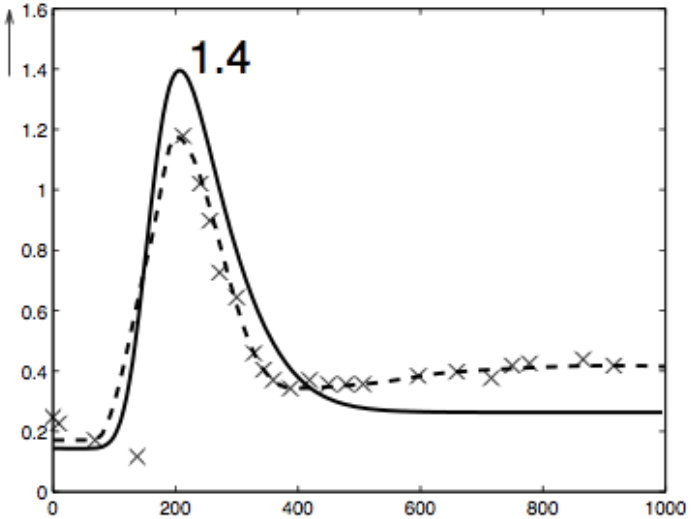
Coefficient of variation over time

- robustness = 0.9

Improving local robustness

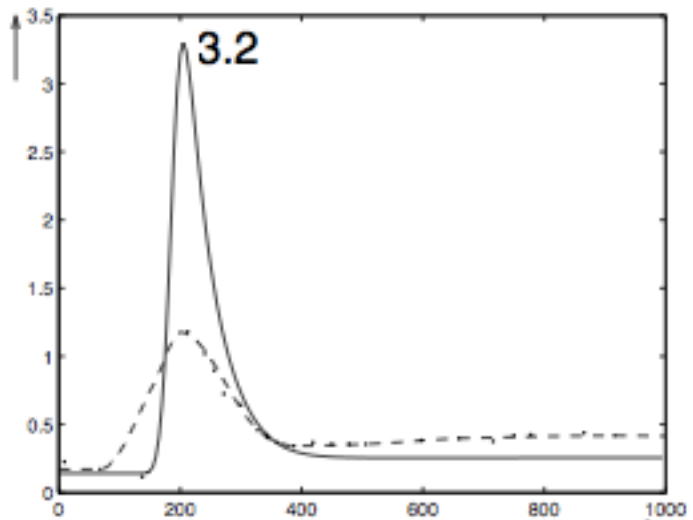


Average behavior of the system (5000 simulations)



Coefficient of variation over time

- robustness = 0.9
- robustness as criterion objective → optimized parameters found s.t robustness = 1



Coefficient of variation in the optimized system

Parameter contribution on global robustness

Variance-based global sensitivity indices

$$S_i = \frac{Var(E(R|P_i))}{Var(R)} \in [0,1]$$

S_γ	20.2 %	$S_{\kappa_{eyfp},\gamma}$	8.7 %
$S_{\kappa_{eyfp}}$	7.4 %	$S_{\kappa_{cI},\gamma}$	6.2 %
$S_{\kappa_{cI}}$	6.1 %	$S_{\kappa^0_{cI},\gamma}$	5.0 %
$S_{\kappa^0_{lacI}}$	3.3 %	$S_{\kappa^0_{cI},\kappa_{eyfp}}$	2.8 %
$S_{\kappa^0_{cI}}$	2.0 %	$S_{\kappa_{cI},\kappa_{eyfp}}$	1.8 %
$S_{\kappa_{lacI}}$	1.5 %	$S_{\kappa^0_{eyfp},\gamma}$	1.5 %
$S_{\kappa^0_{eyfp}}$	0.9 %	$S_{\kappa^0_{cI},\kappa_{cI}}$	1.1 %
$S_{u_{aTc}}$	0.4 %	$S_{\kappa^0_{cI},\kappa_{lacI}}$	0.5 %
total first order	40.7 %	total second order	31.2 %

- degradation factor γ has the strongest impact on the cascade.
- the basal production of EYFP is due to an incomplete repression of the promoter by CI (high effect of κ_{cI}) rather than a constitutive leakage of the promoter (low effect of κ^0_{eyfp}).

Other applications and scalability

Scalability :

- fitness function computation = **numerical simulation** + **satisfaction domain** (satisfaction degree computation)
- Computational cost of violation degree :
  - curve fitting, box specification : less than 10% of required time for numerical simulation
  - oscillation amplitude : 300%
  - oscillation period : 100%
- parameter search parallelized with MPI, efficient on 100-1000 cores depending on problem

Other applications :

- kinetic parameter search in multiple conditions in FSH signaling network (joint work with Eric Reiter, INRA Tours) (40 unknown parameters, solution found in few hours on 32 cores)
- find oscillations in MAPK cascade (37 unknown parameters found in few minutes on 1 core)
- iGEM competition (PARIS team 2007)
- analysis of temporal experimental data [Fages Rizk CMSB 07]

Conclusion

Definition of a continuous degree of satisfaction of LTL($\mathbb{R}$) formulae which can be computed by LTL($\mathbb{R}$) constraint solving algorithm

Continuous satisfaction degree enables :

- measuring the satisfaction of high level specifications
- efficient parameter optimization w.r.t. temporal specification
- measuring and optimizing the robustness of a model w.r.t temporal logic specifications
- sensitivity analysis w.r.t. temporal specification

Related work :

- probabilistic/statistical model checking [Kwiatkowska et al. SIGMETRICS 08, Clarke et al. CMSB 08]
- alternative quantitative interpretation of TL [Fainekos and Pappas FORMATS 07]

Perspectives

- reduce temporal constraint solving computation time with **trace simplification**
- define **formula patterns** in natural language for easier use by non specialists [Monteiro et al, Bioinformatics 2008]
- evaluation of parameter search on larger models with rich biological data (e.g. Chen et al. cell cycle model validation w.r.t. 130 mutants) using parameter search parallelization
- use optimization methods providing **sets of valid parameters**
- develop methods to propose **network structure** modifications when parameter search fails using sensitivity analysis