Quantitative Bounded Model-Checking in Linear Time Logic FO-LTL($\mathbb{R}_{lin}$)

MPRI C2-19

Biochemical Programming

François Fages
Inria Saclay, Lifeware team, France

8 February 2019

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

1. Closed formulae
   - Syntax and semantics on a trace
   - Verification algorithm, parameter search by scanning
2. Constraints with variables
   - Syntax and semantics by validity domains
   - Constraint Solving algorithm for trace analysis
3. Continuous satisfaction degree
   - Parameter optimization by evolutionary algorithm
   - Robustness measure

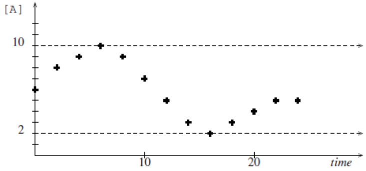
F. Fages, P. Traynard. Temporal Logic Modeling of Dynamical Behaviors: First-Order Patterns and Solvers. In Logical Modeling of Biological Systems, pages 291-323. John Wiley Sons, Inc., 2014.

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):28272839, 2011.

F. Fages, A. Rizk. On Temporal Logic Constraint Solving for the Analysis of Numerical Data Time series. Theoretical Computer Science, 408(1):5565, 2008.

Linear Time Logic LTL($\mathbb{R}$) over Traces

Trace (experiment or simulation):



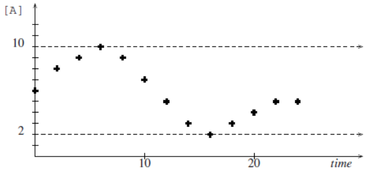
State variables: time, concentration $[A]$, derivative $d[A]/dt$.

Atomic propositions: arithmetic expressions over state variables

Temporal operators: **X**, **F**, **G**, **U**, **R**

Linear Time Logic LTL($\mathbb{R}$) over Traces

Trace (experiment or simulation):



State variables: time, concentration $[A]$, derivative $d[A]/dt$.

Atomic propositions: arithmetic expressions over state variables

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: $\mathbf{FG}([A] > 0.2)$

Local maximum reachability: $\mathbf{F}(d[A]/dt \geq 0 \wedge \mathbf{X} d[A]/dt \leq 0)$

Oscillations *oscil(A,n)* if at least n derivative sign changes

Curve fitting

$\mathbf{F}(Time = 1 \wedge [M] = 0.05 \wedge \mathbf{F}(Time = 2 \wedge [M] = 0.12 \wedge [M] = 0.12 \wedge \mathbf{F}(Time = 3 \wedge [M] = 0.25)))$

Semantics of LTL($\mathbb{R}$) over finite traces

Completion of finite traces with an infinite loop on the last state.

Semantics of LTL($\mathbb{R}$) over finite traces

Completion of finite traces with an infinite loop on the last state.

$\pi \models \phi$ for a proposition ϕ if ϕ holds in the first state of π
$\pi \models \mathbf{X}\phi$ if $\pi^1 \models \phi$
$\pi \models \mathbf{F}\phi$ if $\exists k \geq 0\ \pi^k \models \phi$
$\pi \models \mathbf{G}\phi$ if $\forall k \geq 0\ \pi^k \models \phi$
$\pi \models \phi\ \mathbf{U}\ \psi$ if $\exists k \geq 0\ \pi^k \models \psi \wedge \forall j < k\ \pi^j \models \phi$
$\pi \models \phi\ \mathbf{R}\ \psi$ if $\forall k \geq 0\ \pi^k \models \psi \vee \exists j < k\ \pi^j \models \phi$
ϕ releases ψ if ψ is always true or until ϕ becomes true

Semantics of LTL($\mathbb{R}$) over finite traces

Completion of finite traces with an infinite loop on the last state.

$\pi \models \phi$ for a proposition ϕ if ϕ holds in the first state of π
$\pi \models \mathbf{X}\phi$ if $\pi^1 \models \phi$
$\pi \models \mathbf{F}\phi$ if $\exists k \geq 0\ \pi^k \models \phi$
$\pi \models \mathbf{G}\phi$ if $\forall k \geq 0\ \pi^k \models \phi$
$\pi \models \phi\ \mathbf{U}\ \psi$ if $\exists k \geq 0\ \pi^k \models \psi \wedge \forall j < k\ \pi^j \models \phi$
$\pi \models \phi\ \mathbf{R}\ \psi$ if $\forall k \geq 0\ \pi^k \models \psi \vee \exists j < k\ \pi^j \models \phi$
ϕ releases ψ if ψ is always true or until ϕ becomes true

Duality: $\neg(\phi\ \mathbf{U}\ \psi) = (\neg\phi\ \mathbf{R}\ \neg\psi)$, $\neg\ \mathbf{F}\ \phi =\ \mathbf{G}\ \neg\phi$, $\neg\mathbf{X}\phi = \mathbf{X}\neg\phi$,

Expressiveness: $\mathbf{G}\phi = \mathit{false}\ \mathbf{R}\ \phi$, $\mathbf{F}\phi = \mathit{true}\ \mathbf{U}\ \phi$,

Semantics of LTL($\mathbb{R}$) over finite traces

Completion of finite traces with an infinite loop on the last state.

$\pi \models \phi$ for a proposition ϕ if ϕ holds in the first state of π
$\pi \models \mathbf{X}\phi$ if $\pi^1 \models \phi$
$\pi \models \mathbf{F}\phi$ if $\exists k \geq 0\ \pi^k \models \phi$
$\pi \models \mathbf{G}\phi$ if $\forall k \geq 0\ \pi^k \models \phi$
$\pi \models \phi\ \mathbf{U}\ \psi$ if $\exists k \geq 0\ \pi^k \models \psi \wedge \forall j < k\ \pi^j \models \phi$
$\pi \models \phi\ \mathbf{R}\ \psi$ if $\forall k \geq 0\ \pi^k \models \psi \vee \exists j < k\ \pi^j \models \phi$
ϕ releases ψ if ψ is always true or until ϕ becomes true

Duality: $\neg(\phi\ \mathbf{U}\ \psi) = (\neg\phi\ \mathbf{R}\ \neg\psi)$, $\neg\ \mathbf{F}\ \phi = \ \mathbf{G}\ \neg\phi$, $\neg\mathbf{X}\phi = \mathbf{X}\neg\phi$,

Expressiveness: $\mathbf{G}\phi = \mathit{false}\ \mathbf{R}\ \phi$, $\mathbf{F}\phi = \mathit{true}\ \mathbf{U}\ \phi$,

Negation free formulae: expressed with $\wedge$, $\vee$, **X**, **F**, **G**, **U**, **R** with negations eliminated down to atomic propositions.

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

Input: A finite trace π and a LTL($\mathbb{R}$) formula ϕ
Output: whether or not $\pi \models \phi$

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

Input: A finite trace π and a LTL($\mathbb{R}$) formula ϕ
Output: whether or not $\pi \models \phi$

1. Complete the trace with a loop on the last state

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

Input: A finite trace π and a 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:
    - Label the states with the atomic propositions that are true,

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

Input: A finite trace π and a 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:
   - Label the states with the atomic propositions that are true,
   - Add $\mathbf{X}\phi$ to the immediate predecessors of states labeled by ϕ,

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

Input: A finite trace π and a 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:
   - Label the states with the atomic propositions that are true,
   - Add $\mathbf{X}\phi$ to the immediate predecessors of states labeled by ϕ,
   - Add ϕ **U** ψ to the predecessors of states labelled by ψ while they satisfy ϕ,

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

Input: A finite trace π and a 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:
   - Label the states with the atomic propositions that are true,
   - Add $\mathbf{X}\phi$ to the immediate predecessors of states labeled by ϕ,
   - Add ϕ $\mathbf{U}$ ψ to the predecessors of states labelled by ψ while they satisfy ϕ,
   - Add ϕ $\mathbf{R}$ ψ to the last state if it is labelled by ψ, to the states labelled by ϕ and ψ, and to their predecessors while ψ holds

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

Input: A finite trace π and a 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:
   - Label the states with the atomic propositions that are true,
   - Add $\mathbf{X}\phi$ to the immediate predecessors of states labeled by ϕ,
   - Add ϕ $\mathbf{U}$ ψ to the predecessors of states labelled by ψ while they satisfy ϕ,
   - Add ϕ $\mathbf{R}$ ψ to the last state if it is labelled by ψ, to the states labelled by ϕ and ψ, and to their predecessors while ψ holds
3. Return true if the initial state is labelled by ϕ, and false otherwise

Parameter Search by Scanning

input: a reaction systems $R(\boldsymbol{k})$ with n parameters $\boldsymbol{k}$ given with range $[\underline{k}_i, \overline{k}_i]$, step size s_i and an LTL($\mathbb{R}$) formula ϕ
output: parameter values $\boldsymbol{v}$ such that $\pi(\boldsymbol{v}) \models \phi$ where $\pi(\boldsymbol{v})$ is a simulation trace of $R(\boldsymbol{v})$ or fail

Parameter Search by Scanning

input: a reaction systems $R(\boldsymbol{k})$ with n parameters $\boldsymbol{k}$ given with range $[\underline{k}_i, \overline{k}_i]$, step size s_i and an LTL($\mathbb{R}$) formula ϕ
output: parameter values $\boldsymbol{v}$ such that $\pi(\boldsymbol{v}) \models \phi$ where $\pi(\boldsymbol{v})$ is a simulation trace of $R(\boldsymbol{v})$ or fail

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

Parameter Search by Scanning

input: a reaction systems $R(\boldsymbol{k})$ with n parameters $\boldsymbol{k}$ given with range $[\underline{k}_i, \overline{k}_i]$, step size s_i and an LTL($\mathbb{R}$) formula ϕ
output: parameter values $\boldsymbol{v}$ such that $\pi(\boldsymbol{v}) \models \phi$ where $\pi(\boldsymbol{v})$ is a simulation trace of $R(\boldsymbol{v})$ or fail

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 $O(s_1 * \ldots s_n)$

Parameter Search by Scanning

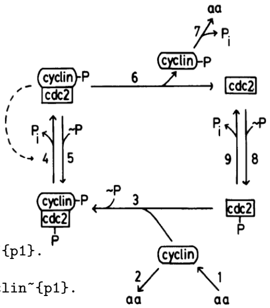
input: a reaction systems $R(\boldsymbol{k})$ with n parameters $\boldsymbol{k}$ given with range $[\underline{k}_i, \overline{k}_i]$, step size s_i and an LTL($\mathbb{R}$) formula ϕ
output: parameter values $\boldsymbol{v}$ such that $\pi(\boldsymbol{v}) \models \phi$ where $\pi(\boldsymbol{v})$ is a simulation trace of $R(\boldsymbol{v})$ or fail

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 $O(s_1 * \dots s_n)$

Continuous optimization procedure ? need a continuous satisfaction degree (fitness function) for LTL(R) formulae

Yeast Cell Cycle Control [Tyson 91]



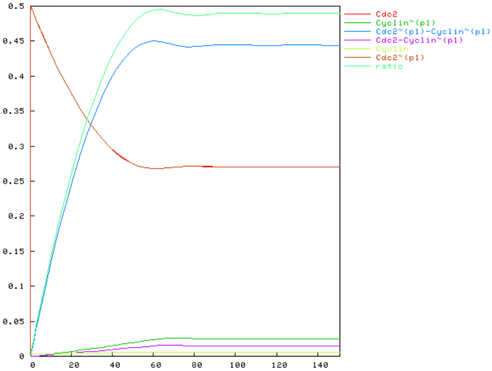

```
k1 for _ => Cyclin.

k2*[Cyclin] for Cyclin => _.

k8*[Cdc2] for Cdc2 => Cdc2~{p1}.

k9*[Cdc2~{p1}] for Cdc2~{p1} => Cdc2.

k3*[Cyclin]*[Cdc2~{p1}] for Cyclin+Cdc2~{p1} => Cdc2~{p1}-Cyclin~{p1}.

k4p*[Cdc2~{p1}-Cyclin~{p1}] for Cdc2~{p1}-Cyclin~{p1} => Cdc2-Cyclin~{p1}.

k4*[Cdc2-Cyclin~{p1}]^2*[Cdc2~{p1}-Cyclin~{p1}] for
   Cdc2~{p1}-Cyclin~{p1} =[Cdc2-Cyclin~{p1}]=> Cdc2-Cyclin~{p1}.

k5*[Cdc2-Cyclin~{p1}] for Cdc2-Cyclin~{p1} => Cdc2~{p1}-Cyclin~{p1}.

k6*[Cdc2-Cyclin~{p1}] for Cdc2-Cyclin~{p1} => Cdc2+Cyclin~{p1}.

k7*[Cyclin~{p1}] for Cyclin~{p1} => _.

parameter(k1,0.015). parameter(k2,0.015). parameter(k3,200).
parameter(k4p,0.018). parameter(k4,180). parameter(k5,0).
parameter(k6,1). parameter(k7,0.6). parameter(k8,100).parameter(k9,100).

present(Cdc2,1).
```

Learning Oscillations

```
biocham: search_parameters([k3,k4],[(0,200),(0,200)],20,
oscil(Cdc2-Cyclin~{p1},3),150).
```

First values found :

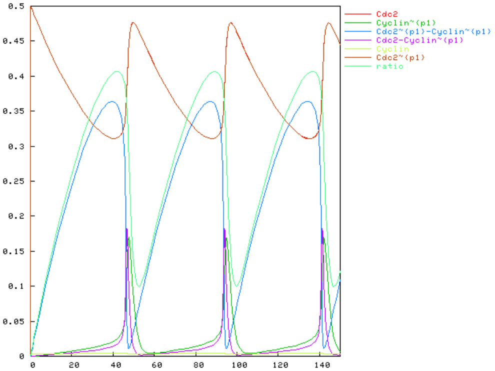
```
parameter(k3,10).

parameter(k4,70).
```



Learning Oscillations

```
biocham: search_parameters([k3,k4],[(0,200),(0,200)],20,
oscil(Cdc2-Cyclin~{p1},3) & F([Cdc2-Cyclin~{p1}]>0.15), 150).
```

```
First values found :

parameter(k3,10).

parameter(k4,120).
```

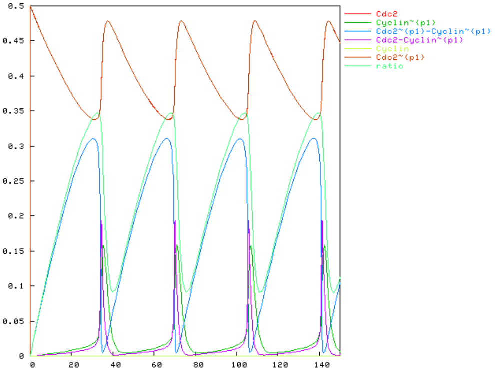


Learning Oscillations

```
biocham: search_parameters([k3,k4],[(0,200),(0,200)],20,
period(Cdc2-Cyclin~{p1},35), 150).
```

First values found:

```
parameter(k3,10).

parameter(k4,280).
```



True/False valuation of temporal logic formulae

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
- sensitivity analyses

True/False valuation of temporal logic formulae

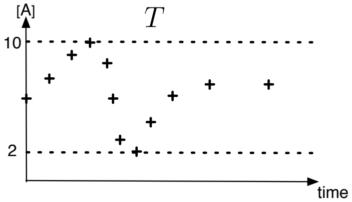
The **True/False** valuation of temporal logic formulae is **not well adapted** to several problems :

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

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

How far is the system from verifying the specification ?

Model-Checking Generalized to Constraint Solving



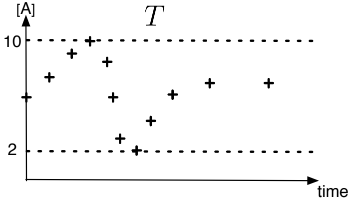
$LTL(\mathbb{R})$

$\Phi = F([A] \geq 7 \wedge F([A] \leq 0))$

Model-checking

the formula is false

Model-Checking Generalized to Constraint Solving



$LTL(\mathbb{R})$

Φ=F([A]≥7
∧F([A]≤0))

Model-checking

the formula is false

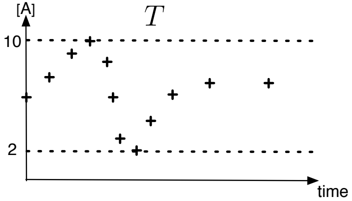
$QFLTL(\mathbb{R})$

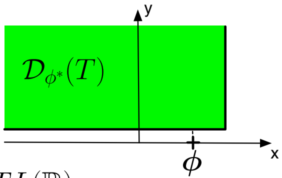
Φ*=F([A]≥x
∧F([A]≤y))

Constraint solving

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

Model-Checking Generalized to Constraint Solving

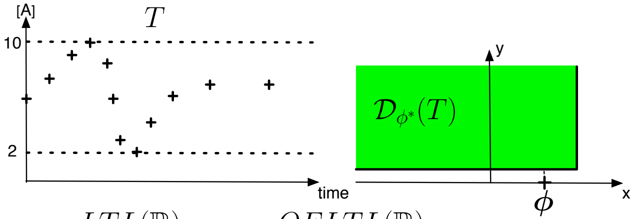




$LTL(\mathbb{R})$	$QFLTL(\mathbb{R})$
Φ=F([A]≥7 ∧F([A]≤0))	Φ*=F([A]≥x ∧F([A]≤y))
Model-checking → the formula is false	**Constraint solving** → the formula is true for any x≤10 ∧ y≥2

Validity domain $\mathcal{D}_{\phi^*}(T)$ of **free variables** in ϕ^*

Model-Checking Generalized to 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 $\mathcal{D}_{\phi^*}(T)$ of **free variables** in ϕ^*

Violation degree $vd(T,\phi) = \text{distance}(val(\phi), D_{\phi^*}(T))$

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

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

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

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

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

Minimum value m: $\mathbf{G}([A] \geq m)$
Minimum amplitude a: $\exists v\ \mathbf{F}([A] \leq v) \wedge \mathbf{F}([A] \geq v + a)$

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

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

Minimum value m: $\mathbf{G}([A] \geq m)$
Minimum amplitude a: $\exists v \; \mathbf{F}([A] \leq v) \wedge \mathbf{F}([A] \geq v + a)$

Crossing at time t: $\mathbf{F}([A] > [B] \wedge \mathbf{X}([A] \leq [B] \wedge Time = t))$
Timing constraints
$\mathbf{G}(Time \leq t1 \Rightarrow [A] < 1 \wedge Time \geq t2 \Rightarrow [A] > 10) \wedge t2 - t1 < 60$

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

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

Minimum value m: $\mathbf{G}([A] \geq m)$
Minimum amplitude a: $\exists v\ \mathbf{F}([A] \leq v) \wedge \mathbf{F}([A] \geq v + a)$

Crossing at time t: $\mathbf{F}([A] > [B] \wedge \mathbf{X}([A] \leq [B] \wedge Time = t))$
Timing constraints
$\mathbf{G}(Time \leq t1 \Rightarrow [A] < 1 \wedge Time \geq t2 \Rightarrow [A] > 10) \wedge t2 - t1 < 60$

Oscillations *oscil(A,n)* if at least n derivative sign changes

Validity Domains of Free Variables

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

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

Validity Domains of Free Variables

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

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

For linear constraints over $\mathbb{R}$, validity domains can be represented as finite unions of polyhedra

- polyhedra for conjunctions,
- union for disjunction,
- complementation for negation,
- projection for $\exists$

BIOCHAM uses the Parma Polyhedral Library PPL

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ of the free variables of ϕ on a trace $T = (s_0, ..., s_n)$ 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 $c(\boldsymbol{x})$,

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\dots,s_n),\phi}$ of the free variables of ϕ on a trace $T = (s_0, \dots, s_n)$ 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 $\mathcal{D}_{s_i,\phi\vee\psi} = \mathcal{D}_{s_i,\phi} \cup \mathcal{D}_{s_i,\psi}$,

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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 $\mathcal{D}_{s_i,\phi\vee\psi} = \mathcal{D}_{s_i,\phi} \cup \mathcal{D}_{s_i,\psi}$,

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

where $\complement$ is the set complement operator over domains, and

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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 $\mathcal{D}_{s_i,\forall x\phi} = \mathcal{D}_{s_i,\neg\exists x\neg\phi}$,

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

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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 $\mathcal{D}_{s_n,\mathbf{X}\phi} = \mathcal{D}_{s_n,\phi}$,

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

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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 $\mathcal{D}_{s_i,\mathbf{G}\phi} = \bigcap_{j=i}^{n} \mathcal{D}_{s_j,\phi}$,

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

Inductive Definition of Validity Domains

The *validity domain* $\mathcal{D}_{(s_0,\ldots,s_n),\phi}$ 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 the set complement operator over domains, and Π_x is the domain projection operator out of x, restoring domain $\mathbb{R}$ for x.

Complexity with Bound Constraints only FO-LTL($\mathbb{R}_{box}$)

Bound constraints $x \geq b$, $x \leq b$ define boxes $\mathcal{R}_i \in \mathbb{R}^v$. Let the size of a union of boxes $\mathcal{D}$ be the least k s.t.$\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{R}_i$.

Complexity with Bound Constraints only FO-LTL($\mathbb{R}_{box}$)

Bound constraints $x \geq b$, $x \leq b$ define boxes $\mathcal{R}_i \in \mathbb{R}^v$. Let the size of a union of boxes $\mathcal{D}$ be the least k s.t.$\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{R}_i$.

Proposition (complexity of the validity domain)

The validity domain of an FO-LTL($\mathbb{R}_{box}$) formula of size f on v variables on a trace of length n is a union of boxes of size $O((nf)^{2v})$.

Complexity with Bound Constraints only FO-LTL($\mathbb{R}_{box}$)

Bound constraints $x \geq b$, $x \leq b$ define boxes $\mathcal{R}_i \in \mathbb{R}^v$. Let the size of a union of boxes $\mathcal{D}$ be the least k s.t.$\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{R}_i$.

Proposition (complexity of the validity domain)

The validity domain of an FO-LTL($\mathbb{R}_{box}$) formula of size f on v variables on a trace of length n is a union of boxes of size $O((nf)^{2v})$.

Proof.

One bound constraint produces at most n bounds. We have at most nf bounds, $O((nf)^2)$ intervals and $O((nf)^{2v})$ boxes. □

Complexity with Bound Constraints only FO-LTL($\mathbb{R}_{box}$)

Bound constraints $x \geq b$, $x \leq b$ define boxes $\mathcal{R}_i \in \mathbb{R}^v$. Let the size of a union of boxes $\mathcal{D}$ be the least k s.t.$\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{R}_i$.

Proposition (complexity of the validity domain)

The validity domain of an FO-LTL($\mathbb{R}_{box}$) formula of size f on v variables on a trace of length n is a union of boxes of size $O((nf)^{2v})$.

Proof.

One bound constraint produces at most n bounds. We have at most nf bounds, $O((nf)^2)$ intervals and $O((nf)^{2v})$ boxes. □

Ex. $\mathbf{F}([A] + 1 = u \vee \cdots \vee [A] + f = u)$ can create $O(nf)$ values for u.

Complexity with Bound Constraints only FO-LTL($\mathbb{R}_{box}$)

Bound constraints $x \geq b$, $x \leq b$ define boxes $\mathcal{R}_i \in \mathbb{R}^v$. Let the size of a union of boxes $\mathcal{D}$ be the least k s.t.$\mathcal{D} = \bigcup_{i=1}^{k} \mathcal{R}_i$.

Proposition (complexity of the validity domain)

The validity domain of an FO-LTL($\mathbb{R}_{box}$) formula of size f on v variables on a trace of length n is a union of boxes of size $O((nf)^{2v})$.

Proof.

One bound constraint produces at most n bounds. We have at most nf bounds, $O((nf)^2)$ intervals and $O((nf)^{2v})$ boxes. □

Ex. $\mathbf{F}([A]+1 = u \vee \cdots \vee [A]+f = u)$ can create $O(nf)$ values for u.
$\mathbf{F}([A_1]+1 = X_1 \vee ... \vee [A_1]+f = X_1) \wedge \ldots \wedge \mathbf{F}([A_v]+1 = X_v \vee ... \vee [A_v]+f = X_v)$ has a validity domain of $O((nf)^v)$ points.

Yeast Cell Cycle Control [Tyson 91]

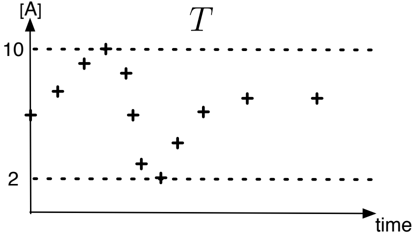
```
domains(G(([MPF]>=v)&([MPF]=<v+a))).
v + a >= 0.837557, v =< 0.0016107
Time elapsed : 48 ms

domains(maxAmpl([MPF],[a])).
a >= 0.835946
Time elapsed : 0 ms

domains(distanceSuccPeaks(MPF,d)).
d = 23.3555
|
d = 23.1196
|
d = 23.0935
|
d = 23.119
Time elapsed : 330 ms
```

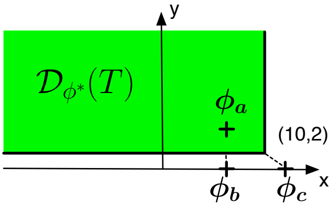
Violation degree of an LTL formula

Violation degree $vd(T,\phi)$ and satisfaction degree $sd(T,\phi)$

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

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





ϕ^*(x,y)= F([A]≥x ∧ F([A]≤y))

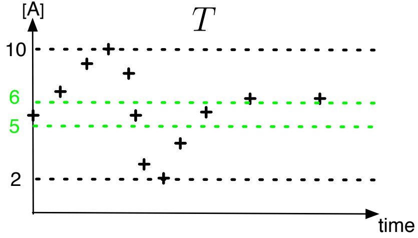
ϕ_a = F([A]≥6 ∧ F([A]≤5))	$\phi^*_{(6,5)}$	*vd*=0	(✓)
ϕ_b = F([A]≥6 ∧ F([A]≤0))	$\phi^*_{(6,0)}$	*vd*=2	(✗)
ϕ_c = F([A]≥12 ∧ F([A]≤0))	$\phi^*_{(12,0)}$	*vd*=2√2	(✗)

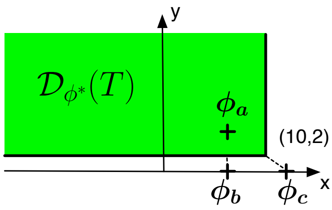
Violation degree of an LTL formula

Violation degree $vd(T,\phi)$ and satisfaction degree $sd(T,\phi)$

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

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





ϕ^*(x,y)= F([A]≥x ∧ F([A]≤y))

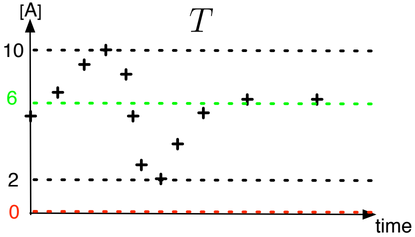
ϕ_a = F([A]≥6 ∧ F([A]≤5))	ϕ^*(6,5)	*vd*=0	(✓)
ϕ_b = F([A]≥6 ∧ F([A]≤0))	ϕ^*(6,0)	*vd*=2	(✗)
ϕ_c = F([A]≥12 ∧ F([A]≤0))	ϕ^*(12,0)	*vd*=$2\sqrt{2}$	(✗)

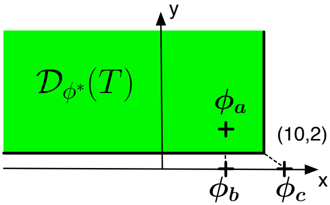
Violation degree of an LTL formula

Violation degree $vd(T, \phi)$ and satisfaction degree $sd(T, \phi)$

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

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





ϕ^*(x,y)= F([A]≥x ∧ F([A]≤y))

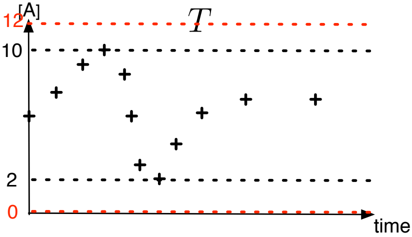
ϕ_a = F([A]≥6 ∧ F([A]≤5))	ϕ^*(6,5)	*vd*=0	(✓)
ϕ_b = F([A]≥6 ∧ F([A]≤0))	ϕ^*(6,0)	*vd*=2	(✗)
ϕ_c = F([A]≥12 ∧ F([A]≤0))	ϕ^*(12,0)	*vd*=2√2	(✗)

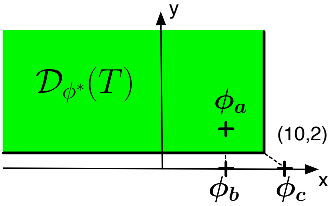
Violation degree of an LTL formula

Violation degree $vd(T,\phi)$ and satisfaction degree $sd(T,\phi)$

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

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





ϕ^*(x,y)= F([A]≥x ∧ F([A]≤y))

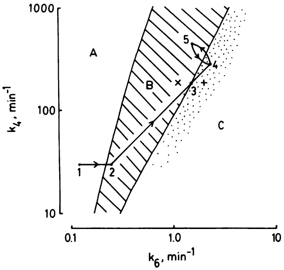
ϕ_a = F([A]≥6 ∧ F([A]≤5))	ϕ^*(6,5)	vd=0	(✓)
ϕ_b = F([A]≥6 ∧ F([A]≤0))	ϕ^*(6,0)	vd=2	(✗)
ϕ_c = F([A]≥12 ∧ F([A]≤0))	ϕ^*(12,0)	vd=2√2	(✗)

LTL Continuous Satisfaction Landscape

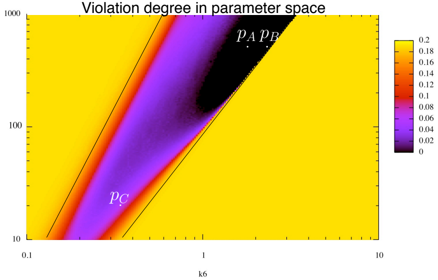
Example with :

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

$$\phi^*:\ \mathbf{F}(\ [A]\geq x) \wedge \mathbf{F}([A]\leq y);\ \text{amplitude } x\text{-}y\geq 0.3$$



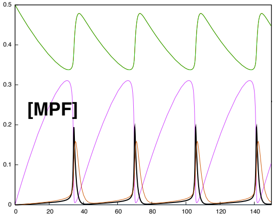
Bifurcation diagram



LTL satisfaction diagram

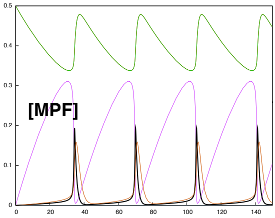
Searching kinetic parameter values from LTL specifications

- simple model of the yeast cell cycle from [Tyson PNAS 91]
- models Cdc2 and Cyclin interactions (6 variables, 8 kinetic parameters)



Searching kinetic parameter values from LTL specifications

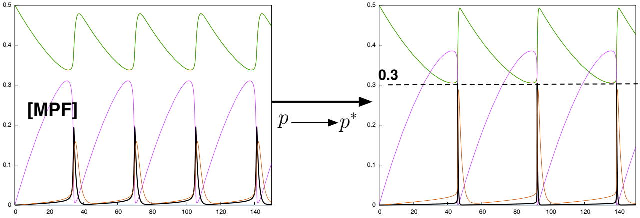
- simple model of the yeast cell cycle from [Tyson PNAS 91]
- models Cdc2 and Cyclin interactions (6 variables, 8 kinetic parameters)



- Pb : find values of 8 parameters such that amplitude is ≥ 0.3
  ϕ^*: **F**($[A] \geq v$) $\wedge$ **F**($[A] \leq y$)
  amplitude z=x-y
  goal : $z = 0.3$

Searching kinetic parameter values from LTL specifications

- simple model of the yeast cell cycle from [Tyson PNAS 91]
- models Cdc2 and Cyclin interactions (6 variables, 8 kinetic parameters)



- Pb : find values of 8 parameters such that amplitude is $\geq$ 0.3
  ϕ^*: **F**([A]$\geq$v) $\wedge$ **F**([A]$\leq$y)
  amplitude z=x-y
  goal : z = 0.3
- Solution found after 30s (100 calls to the fitness function)

Covariance Matrix Adaptation Evolutionary Strategy

CMA-ES maximizes a black box fitness function (here $sd(\phi)$) in continuous domain ($\boldsymbol{k}$) [Hansen Osermeier 01, Hansen 08]

Covariance Matrix Adaptation Evolutionary Strategy

CMA-ES maximizes a black box fitness function (here $sd(\phi)$) in continuous domain ($\boldsymbol{k}$) [Hansen Osermeier 01, Hansen 08]

- probabilistic neighborhood: multivariate normal distribution

Covariance Matrix Adaptation Evolutionary Strategy

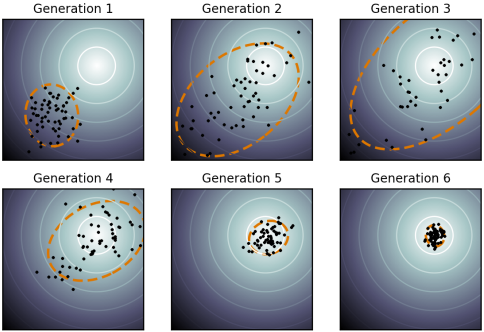
CMA-ES maximizes a black box fitness function (here $sd(\phi)$) in continuous domain ($\boldsymbol{k}$) [Hansen Osermeier 01, Hansen 08]

- probabilistic neighborhood: multivariate normal distribution
- evaluation of covariance matrix by sampling (e.g. 50 best of 100 simulations)

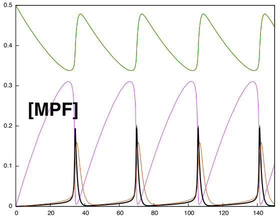
Covariance Matrix Adaptation Evolutionary Strategy

CMA-ES maximizes a black box fitness function (here $sd(\phi)$) in continuous domain ($\boldsymbol{k}$) [Hansen Osermeier 01, Hansen 08]

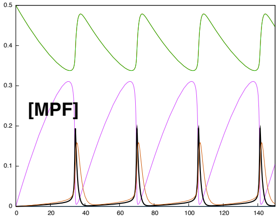
- probabilistic neighborhood: multivariate normal distribution
- evaluation of covariance matrix by sampling (e.g. 50 best of 100 simulations)
- move and distribution update according to covariance matrix



Searching Parameter Values from Period Constraints in LTL

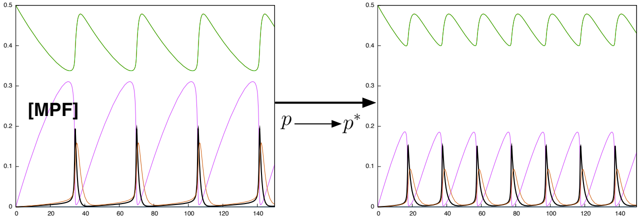


Searching Parameter Values from Period Constraints in LTL



- Pb : find values of 8 parameters such that period is 20
  ϕ^*:**F**(MPF$_{localmaximum}$ $\wedge$Time=t1$\wedge$ **F**(MPF$_{localmaximum}$ $\wedge$Time=t2))
  (with MPF$_{localmaximum}$: d([MPF])/dt>0 $\wedge$ **X**(d([MPF])/dt$\leq$0))
  period z=t2-t1
  goal z=20

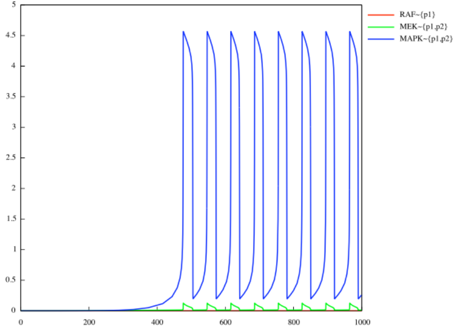
Searching Parameter Values from Period Constraints in LTL



- Pb : find values of 8 parameters such that period is 20
  ϕ^*:**F**(MPF$_{localmaximum}$ $\wedge$Time=t1$\wedge$ **F**(MPF$_{localmaximum}$ $\wedge$Time=t2))
  (with MPF$_{localmaximum}$: d([MPF])/dt>0 $\wedge$ **X**(d([MPF])/dt$\leq$0))
  period z=t2-t1
  goal z=20
- Solution found after 60s (200 calls to the fitness function)

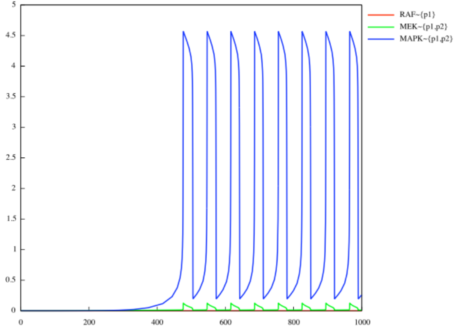
Oscillations in MAPK signal transduction cascade

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



Oscillations in MAPK signal transduction cascade

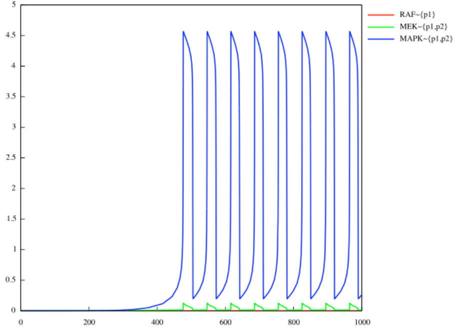
- **MAPK** signaling model [Huang Ferrel PNAS 96]



- **search for oscillations** in **37 dimensions** (30 parameters and 7 initial conditions)
  Solution found after 3 min (200 calls to the fitness function)
  Oscillations already observed by simulation [Qiao et al. 07]

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)
  Oscillations already observed by simulation [Qiao et al. 07]
- No negative feedback in the **reaction graph**, but negative circuits in the **influence graph**

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]:

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

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. LTL($\mathbb{R}$) spec:

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

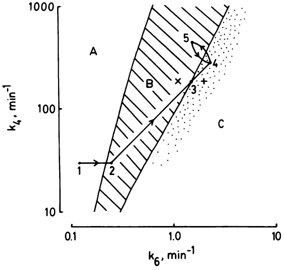
where $T(p)$ is the trace obtained by numerical integration of the ODE for perturbation p

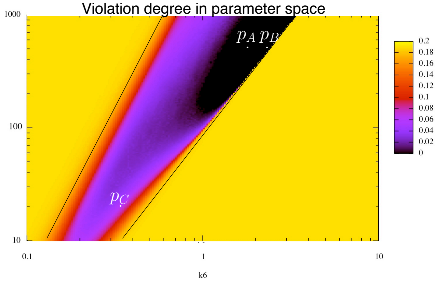
Robustness analysis w.r.t parameter perturbations

Example with :

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

  ϕ^*: **F**([A]≥x) ∧ **F**([A]≤y) ; amplitude x-y≥0.2
- parameters normally distributed, $\mu = p_{ref}$, CV=0.2



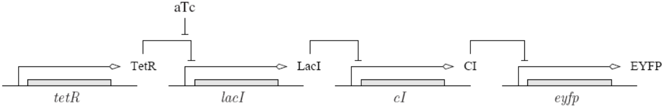


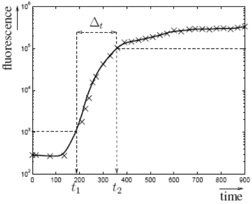
$$\mathcal{R}_{\phi,p_A} = 0.83,\ \mathcal{R}_{\phi,p_B} = 0.43,\ \mathcal{R}_{\phi,p_C} = 0.49$$

Application to Synthetic Biology in *E. Coli*

Cascade of transcriptional inhibitions added to *E.coli* [Weiss 05 pnas]

input small molecule aTc **output** protein EYFP





Specification: EYFP has to remain below 10^3 for at least 150mn 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 FO-LTL($\mathbb{R}$)

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

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

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

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

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

which is abstracted into

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

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.

Improving robustness

Variance-based global sensitivity indices $S_i = \frac{Var(E(R|P_i))}{Var(R)} \in [0, 1]$

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

degradation factor γ has the strongest impact on the cascade.

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

degradation factor γ has the strongest impact on the cascade.

aTc variations have a very low impact

Improving 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 %	$S_{\kappa_{cl},\gamma}$	6.2 %
$S_{\kappa_{cl}}$	6.1 %	$S_{\kappa^0_{cl},\gamma}$	5.0 %
$S_{\kappa^0_{lacl}}$	3.3 %	$S_{\kappa^0_{cl},\kappa_{eyfp}}$	2.8 %
$S_{\kappa^0_{cl}}$	2.0 %	$S_{\kappa_{cl},\kappa_{eyfp}}$	1.8 %
$S_{\kappa_{lacl}}$	1.5 %	$S_{\kappa^0_{eyfp},\gamma}$	1.5 %
$S_{\kappa^0_{eyfp}}$	0.9 %	$S_{\kappa^0_{cl},\kappa_{cl}}$	1.1 %
$S_{u_{aTc}}$	0.4 %	$S_{\kappa^0_{cl},\kappa_{lacl}}$	0.5 %
total first order	40.7 %	total second order	31.2 %

degradation factor γ has the strongest impact on the cascade.

aTc variations have a very low impact

sensitivity to regulated κ_{eyfp} EYFP production more important than basal κ^0_{eyfp}

Improving 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.

aTc variations have a very low impact

sensitivity to regulated κ_{eyfp} EYFP production more important than basal κ^0_{eyfp}

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}).

TD9 Protocell optimization

- FO-LTL(Rlin) trace analysis
- FO-LTL(Rlin) quantitative model-checking
- sensitivity
- robustness
- parameter search