

An automata-based formalism for analysing stochastic models of biological networks

Paolo Ballarini



CentraleSupélec

GT Bioss - Journées ASSTABIO - Rennes

22,23 Mars 2018

Context

- ▶ **discrete-state stochastic models (applied to biology)**
 - ▶ continuous-time Markov chains (CTMCs)
 - ▶ more generally **discrete-event stochastic processes (DESPs)**

- ▶ **Performance analysis (desiderata):**
 - ▶ languages for definition of *key performance indicators* (KPIs)
 - ▶ tools for (automatically) assessing KPIs of a given model
 - ▶ in this talk: **hybrid automata** as a means to express KPIs

- ▶ **What kind of performance indicators ?:**
 - ▶ temporal logic reasoning: allow for sophisticated indicators, e.g.:
 - ▶ *what is the population of nucleic β -catenin at time T if LRP5 receptor gets inhibited at least N times within T ?*
 - ▶ *how many mRNA are transcribed within T if gene X stay repressed at most $T' < T$? ...*

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

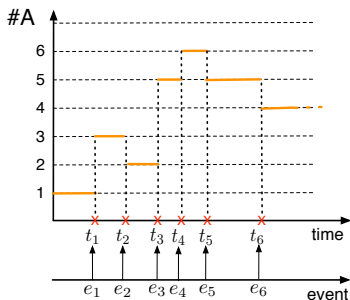
Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

Discrete event stochastic process (DESP)

- ▶ a countable set of states $S = \{s_0, s_1, \dots\}$
- ▶ a fine set of events $E = \{e_1, e_2, \dots, e_n\}$
 - ▶ events occur with a **continuous delay** given by a **distribution of probability**



a DESP trajectory σ_A showing the population of species A

What kind of analysis for a DESP model ?

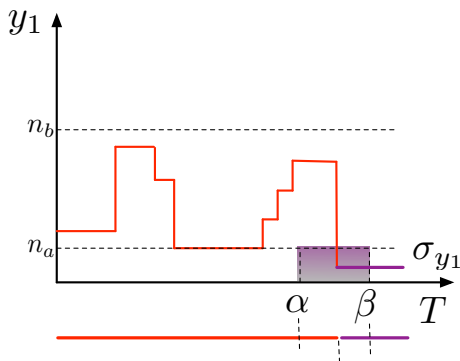
- ▶ **computing state-probability distributions**: probability to be in a given state s
 - ▶ at-time- t (**transient-state distribution**)
 - ▶ in-the-long-run, $t \rightarrow \infty$ (**steady-state distribution**)

- ▶ **computing average values**: e.g. the average population of X at time t
 - ▶ boils down to computing state-probability distributions

- ▶ **computing probability of a path event**
 - ▶ **path event**: set of trajectories fulfilling given **temporal conditions**

Classification of path events

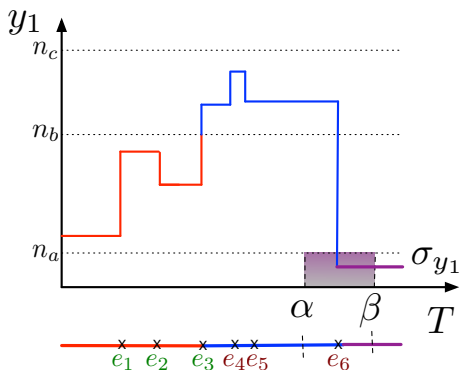
time-bounded state-reachability: reaching a state-region within given delay [Continuous Stochastic Logic, Aziz 2000]



“within time $t \in [\alpha, \beta]$ population y_1 reaches $[0, n_a]$ without leaving $[n_a, n_b]$ ”

Classification of path events

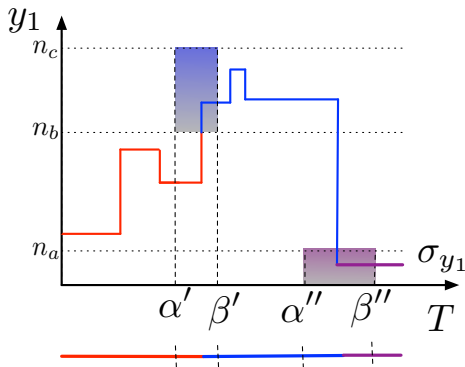
event-conditioned state-reachability: reaching a state by observing only specific events [asCSL, Baier et al., 2003]



“within time $t \in [\alpha, \beta]$ population y_1 reaches $[0, n_a]$ by traversing first $[n_a, n_b]$, where only events of $E_1 \subset E$ are observed, then $[n_b, n_c]$, where only events of $E_2 \subset E$ are observed”

Classification of path events

multiple-bounded state-reachability: several time bounds on a single reachability condition [1-clock time-automaton, CSL^{TA}, Haddad et al. 2007]



“within time $t \in [\alpha', \beta']$ y_1 reaches $[n_b, n_c]$ by traversing $[n_a, n_b]$, then within time $t \in [\alpha'', \beta'']$ it reaches $[0, n_a]$ while staying within $[n_b, n_c]$ ”

Temporal logic for performance evaluation

allow for reasoning (to different extent) about behaviour of a DESP through a combination of:

- ▶ **state-condition**: CSL, pLTL,
- ▶ **state/event-condition**: asCSL, CSL^{TA}
- ▶ **reward-based-condition**: CSRL (to some extent)

Hybrid Automata Specification Language: using of Hybrid Automata to specify path events.

- ▶ very powerful specification language
- ▶ allow for **combining state+event+reward-based conditions for selecting relevant paths**

Outline

Introduction

Discrete event systems and temporal reasoning

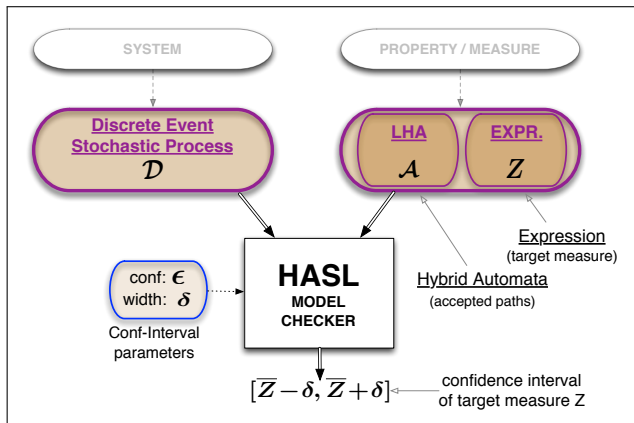
Hybrid Automata Specification Language

Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

HASL-based performance evaluation



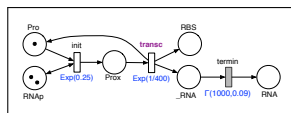
▶ **input:**

- ▶ **model:** a DESP \mathcal{D}
- ▶ **formula:** a pair (\mathcal{A}, Z)

▶ **output:**

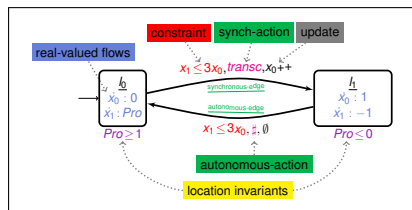
- ▶ 100(1- ϵ)% confidence interval of $E[Z]$

Synchronizing a DESP with a Hybrid Automata



\mathcal{D} : a DESP

×

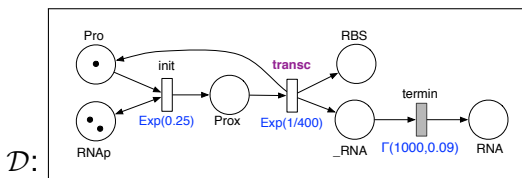


\mathcal{A} : \Rightarrow selects specific paths of \mathcal{D}

HASL model-checking scheme (intuitively):

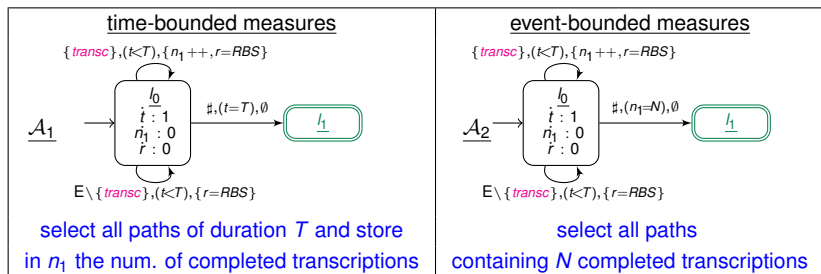
- ▶ a timed-path σ of \mathcal{D} is generated through stochastic simulation
- ▶ \mathcal{A} decides whether σ is accepted or not, meanwhile collecting data about σ
- ▶ collected data is used for estimating relevant measures (wrt accepted paths)

example: selecting paths of gene-exp. toy model

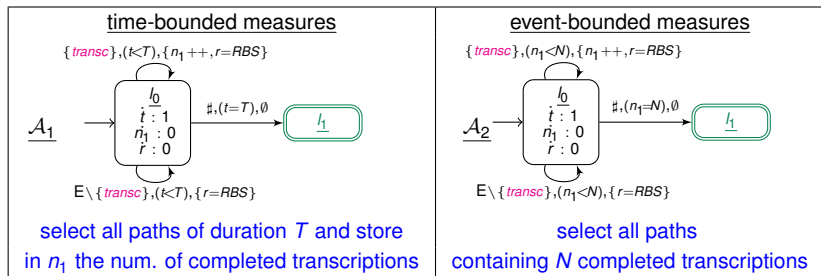


DESP toy-model of gene-expression in Stochastic Petri Net form

Linear Hybrid Automata for selecting paths of DESP \mathcal{D}



specifying the quantity to estimate w.r.t. the selected paths



- ▶ $\phi_1 \equiv (\mathcal{A}_1, E[\text{last}(n_1)])$: avg. number of completed *transcriptions* within T
- ▶ $\phi'_1 \equiv (\mathcal{A}_2, E[\max(r) - \min(r)])$: max. deviation of RBS population within completing N *transcriptions*
- ▶ $\phi_2 \equiv (\mathcal{A}_2, \text{PDF}(\text{last}(t), \text{step}, \text{start}, \text{end}))$: PDF of delay for completing N *transcriptions* calculated by histogram over support $[\text{start}, \text{end}]$ discretised into *step-wide* sub-intervals

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

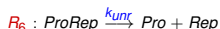
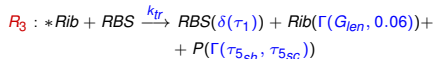
Biological case studies

Performance analysis of gene-expression

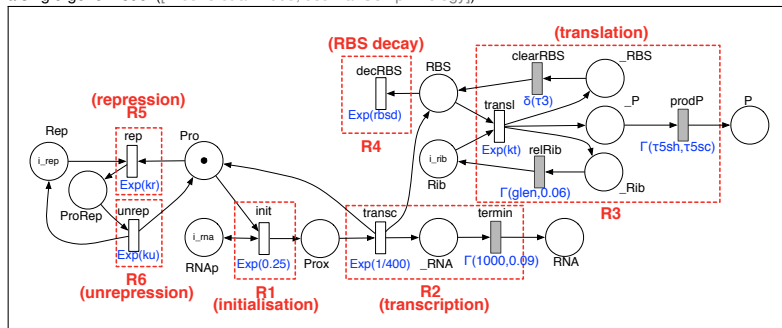
Performance analysis of stochastic oscillators

Gene-expr. with stochastic delayed dynamics

gene expression: the process through which proteins are synthesized from a sequence of DNA.



a single-gene model ([Ribeiro *et al.* 2006, Journal Comp. Biology]):



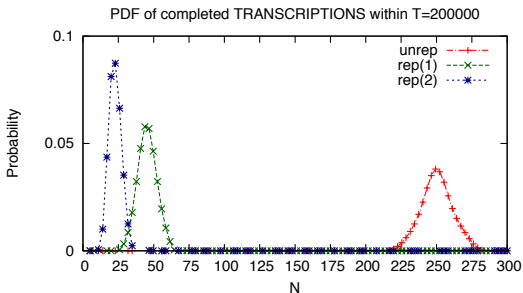
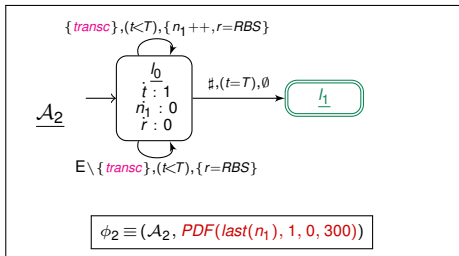
1- measuring PDF of completed transcriptions

Model settings:

- ▶ unrep: no repressor
- ▶ rep(1): one mol. of repressor
- ▶ rep(2): two mol. of repressor

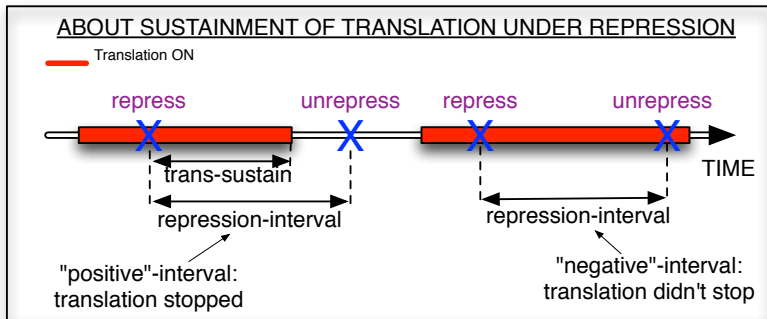
LHA settings:

- ▶ N : completed transcriptions



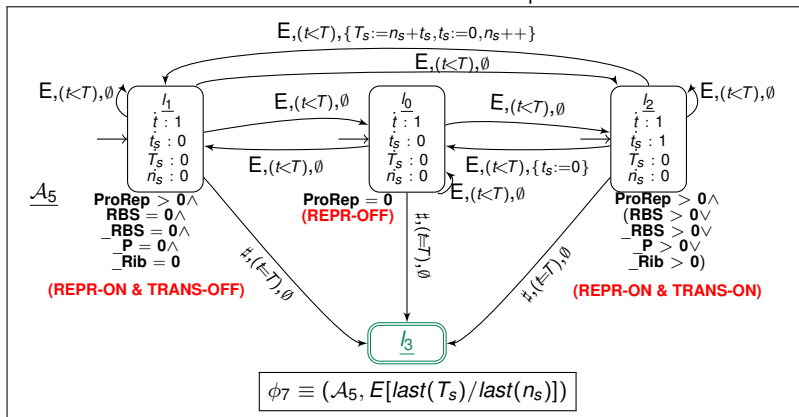
2- measuring sustainment of *translation*

sustainment of translation : *"how long does it take for an on-going translation to halt since repression kicks in (given that translation halts)"*



an LHA for measuring sustainment of translation

sustainment of translation under repression



legend of used variables:

t : total time

t_s : sustain time (inter-repression)

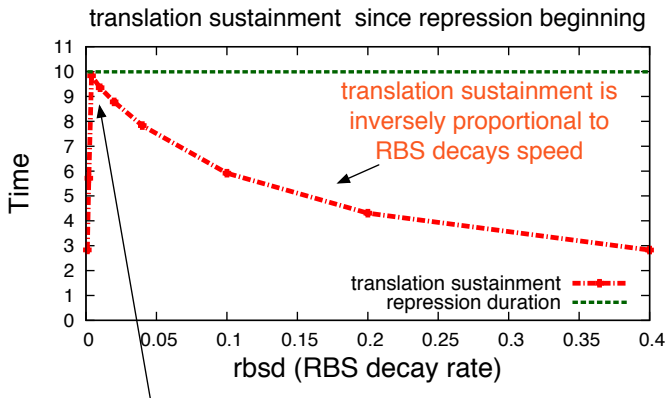
T_s : total sustain time

n_s : num. of repress-intervals where translation halted

measuring the sustainment of gene translation

experiment settings:

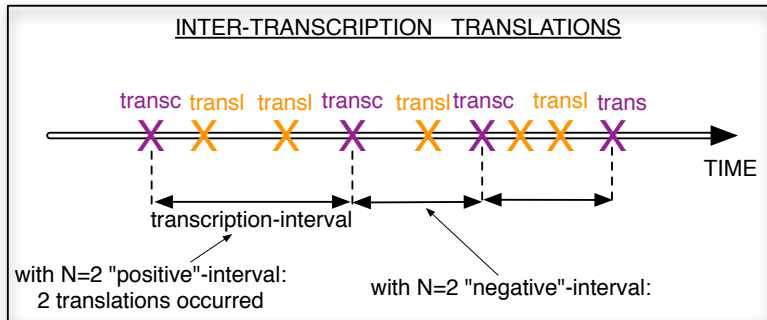
- ▶ constant repression duration i.e. $\text{unrepress} \sim \delta(10)$
- ▶ varying RBS decay rate



cross point (~ 0.01): RBS decays so slowly that translations never stop for the majority of repression-intervals

3- Probability of inter-transcription translations

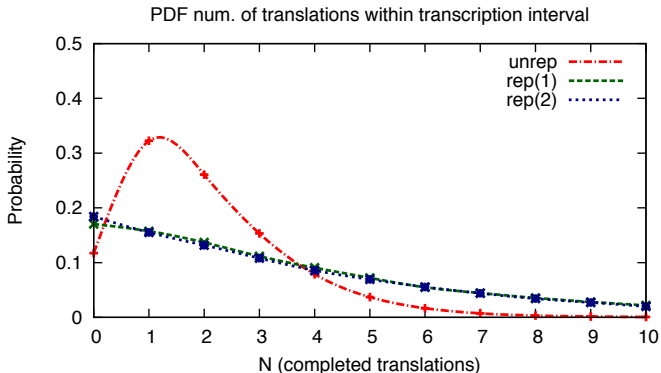
"with what probability N translations will complete in between two consecutive transcription events?"



measuring PDF of inter-transcription translations

experiment settings:

- ▶ repressed vs unrepressed model



- ▶ unrepressed model: probability density with maximum around $N \sim 1.2$
- ▶ repressed model: PDFs asymptotically decreasing (independent of the level of repression)

Outline

Introduction

Discrete event systems and temporal reasoning

Hybrid Automata Specification Language

Biological case studies

Performance analysis of gene-expression

Performance analysis of stochastic oscillators

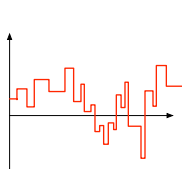
the problem: formal analysis of oscillations

Q1: can we find a temporal logic characterisation of oscillations for a stochastic model? **Yes, through automata specifications**

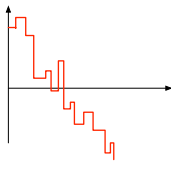
Q2: can we use it to decide whether a stochastic model oscillates ? **Yes for CTMC [Spieler 2014]**

Q3: can we use it to measure the characteristics (i.e. period, amplitude) of an oscillator ? **to different extend with either timed or hybrid automata**

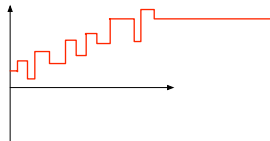
- ▶ oscillation: the repetitive variation (in time), of some measure about a central value



oscillating trace



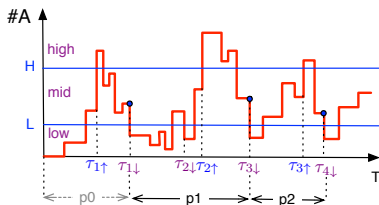
divergent trace



convergent trace

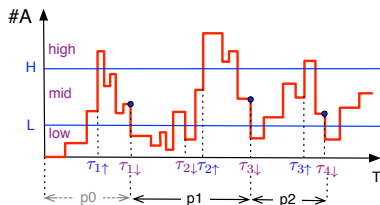
measuring the period: noisy periodicity [Spieler 2009,2014]

- ▶ given two thresholds $L < H$, a function σ_A is **noisy periodic** iff it (perpetually) crosses the intervals $low[0, L)$, $mid[L, H)$, $low[H, +\infty)$.



- ▶ **noisy period realisation**: delay between two consecutive *mid-low* crossing points interleaved by a visit to the *high* region

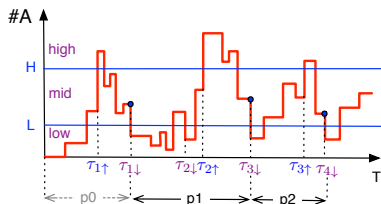
Formal characterisation of noisy period (1)



- ▶ $\tau_{j\downarrow}$ (resp. $\tau_{j\uparrow}$): the j^{th} *mid-to-low* (resp. *mid-to-high*) crossing point
- ▶ $T_{\downarrow} = \cup_j \tau_{j\downarrow}$ (resp. $T_{\uparrow} = \cup_j \tau_{j\uparrow}$) set of all *low-crossing points*
- ▶ $T_{k\downarrow} \subseteq T_{\downarrow}$: k -th sequence of contiguous *low-crossing points* not interleaved by any *high-crossing point*

$$T_{1\downarrow} = \{\tau_{1\downarrow}, \tau_{2\downarrow}\}, T_{2\downarrow} = \{\tau_{3\downarrow}\}, \dots$$

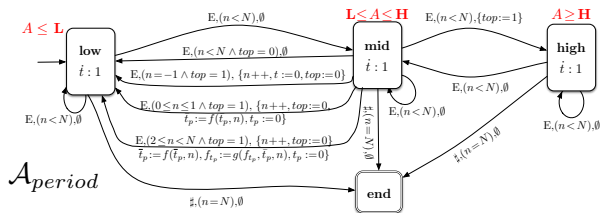
Formal characterisation of noisy period (2)



For σ_A a noisy periodic trajectory we define:

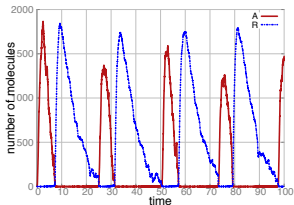
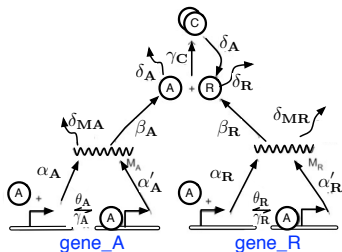
- ▶ $t_{p_k} = \min(T_{(k+1)\downarrow}) - \min(T_{k\downarrow})$ (k^{th} noisy period realisation)
- ▶ $\bar{t}_p(n) = \frac{1}{n} \sum_{k=1}^n t_{p_k}$ (average period estimator)
- ▶ $s_{t_p}^2(n) = \frac{1}{n} \sum_{k=1}^n (t_{p_k} - \bar{t}_p(n))^2$ (period variance estimator)

Hybrid automata for noisy period detection

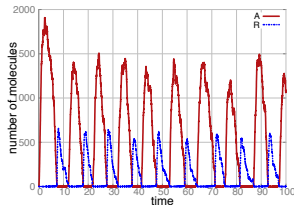


- ▶ detect noisy-periodic traces of observed species A (stops at N^{th} period)
- ▶ variables: t (total duration of periods) n (counter of periods) top (boolean flag) t_p (duration of last detected period); μ_{t_p} (avg. period until last detected); $s_{t_p}^2$ (period variance until last detected)
- ▶ output measures:
 - ▶ $Z_1 \equiv E[\text{last}(\bar{t}_p)]$ (average period duration)
 - ▶ $Z_2 \equiv PDF(\bar{t}_p, s, l, h)$ (PDF of the average period duration)
 - ▶ $Z_3 \equiv E[\text{last}(s_{t_p}^2)]$ (variance of period duration)

Circadian-clock oscillator [Vilar *et al.* 2002]



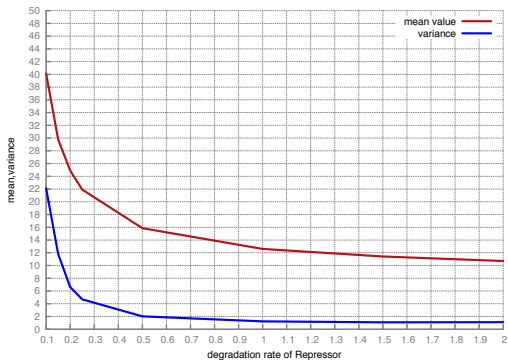
$\delta_R = 0.2$
original repressor degradation



$\delta_R = 2$
10x faster repressor degradation

mean/variance of oscillation period measured with

\mathcal{A}_{period}

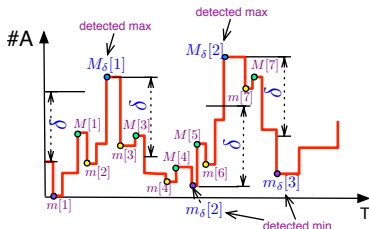


- ▶ HASL specifications: $\phi_1 : (\mathcal{A}_{period}, E[\text{last}(\bar{t}_p)])$ $\phi_2 : (\mathcal{A}_{period}, E[\text{last}(s_{t_p}^2)])$
- ▶ experiment setting: num. of observed periods (per trace) $N=100$, $L=1$, $H=1000$ partition
- ▶ **outcome**: slowing repressor degradation increases the period duration, but makes oscillations more “stochastic” (larger variance of period duration)

Measuring the amplitude: noisy alternance

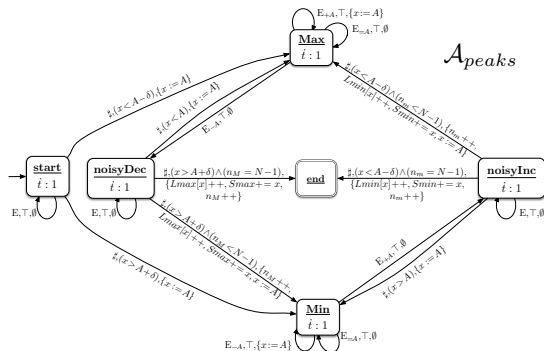
given a noise value $\delta > 0$, a function σ_A is **δ alternating** if it contains infinitely many δ -separated local minima (and equivalently δ -separated local maxima).

- ▶ **goal**: to locate the peaks of oscillation
- ▶ **local max** : the maximum value distanced at least δ from previous **local min**



- ▶ remark: we don't impose any specific $L < H$ thresholds

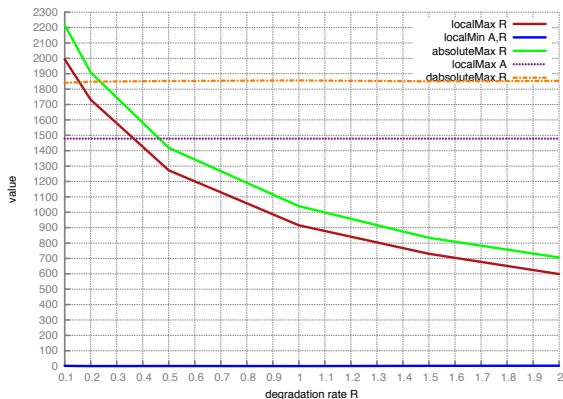
Hybrid automata for detecting peaks of δ alternating traces



- ▶ detect local maxima and minima for observed species A (stops at N^{th} maxima)
- ▶ requires: *i*) partitioning event set ($E = E_{A-} \cup E_{A-} \cup E_{A+}$); *ii*) setting δ par.
- ▶ variables: $Lmax$ (detected local max); $Lmin$ (detected local min)
- ▶ output measures:

- ▶ $Z_{max} \equiv E[\text{last}(Smax)/n_M]$ (average height of δ -separated maxima)
- ▶ $Z_{min} \equiv E[\text{last}(Smin)/n_m]$ (average height of δ -separated minima)

Circ. clock max/min peaks measured with \mathcal{A}_{peaks}



- ▶ HASL measures: $(\mathcal{A}_{peaks}, E[\text{last}(S_{max})/n_M])$, $(\mathcal{A}_{peaks}, E[\text{last}(S_{min})/n_m])$
- ▶ degradation of R reduces maximal peaks of R but has no effects on A 's
- ▶ num. of observed local Max $N=100$ per trace
- ▶ $\delta=10\%$ of absolute max of observed species

Message to bring home ...

- ▶ **hybrid automata** are powerful means for expressing **performance indicators of biological models**
 - ▶ Example: **need to measure oscillations \implies you can !**
- ▶ **tool support:** COSMOS statistical model-checker
(<https://www.lacl.fr/~barbot/cosmos/>)
- ▶ perspectives:
 - ▶ **HASL driven parameter estimation:** devise the parameters of a DESP model so to maximise an HASL measure
 - ▶ **improving HASL runtime:** resorting to approximated stochastic simulation techniques

Acknowledgments

- ▶ Benoit Barbot, Marie Dufлот, Serge Haddad, Nihal Pekergin

BIBLIOGRAPHY

- [BDDHP15]:** P. Ballarini, B. Barbot, M. Duflot, S. Haddad, N. Pekergin. "*HASL: A new approach for performance evaluation and model checking from concepts to experimentation*" Performance Evaluation 2015.
- [Bal15]:** P. Ballarini, "*Analysing oscillatory trends of discrete-state stochastic processes through HASL statistical model checking*". STTT Journal, 2015.
- [BD15]:** P. Ballarini, M. Duflot, "*Applications of an expressive statistical model checking approach to the analysis of genetic circuits.*". Theoretical Computer Science, 2015.
- [BMR12]:** P. Ballarini, J. Mäkelä, A.S. Ribeiro "*Expressive Statistical Model Checking of Genetic Networks with Delayed Stochastic Dynamics*" Proceeding of 8th Int. Conference CMSB'12
- [BDDHP11c]:** P. Ballarini, H. Djafri, M. Duflot, S. Haddad, N. Pekergin. "*COSMOS: a Statistical Model Checker for Hybrid Automata Stochastic Logic*" Proceeding of 8th Int. Conference QEST'11
- [BDDHP11b]:** P. Ballarini, H. Djafri, M. Duflot, S. Haddad, N. Pekergin. "*Petri Nets Compositional Modeling and Verification of Flexible Manufacturing Systems*" Proceeding of Int. Conference CASE'11
- [BDDHP12]:** E. G. Amparore, P. Ballarini, M. Beccuti, S. Donatelli, G. Franceschinis "*Expressing and computing passage time measures of GSPN models with HASL*" Proceeding of 8th Int. Conference ATPN'13