What is a cell cycle checkpoint: The TotemBioNet answer

Déborah Boyenval Gilles Bernot Hélène Collavizza Jean-Paul Comet

Université Côte d'Azur, I3S Laboratory, Sophia Antipolis, France.

CMSB 2020 - Tool Paper

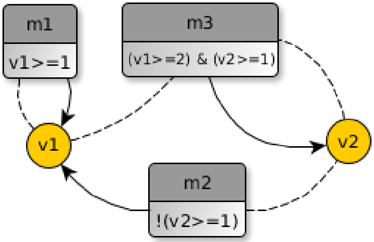
$23^{th} - 25^{th}$ September

TotemBioNet and the qualitative modeling framework

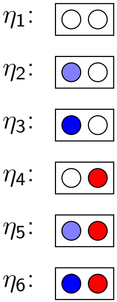
- TotemBioNet automates parameters' identification for René Thomas' discrete modeling framework
- It combines two formal methods: weakest precondition for Hoare logic and model checking for temporal logic
- It computes the *exhaustive* set of Thomas' parameterizations verifying a set of biological properties

René Thomas' syntax: multivaluated regulatory network

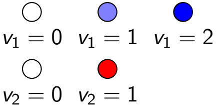
Regulatory graph:



States of the system:



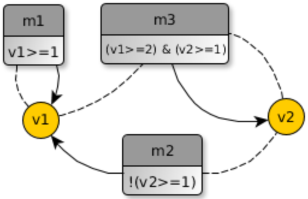
States of variables:



- **m$_1$** $[v_1 \geq 1] \rightarrow v_1$: v_1 activates itself
- **m$_2$** $[\neg(v_2 \geq 1)] \rightarrow v_1$: v_2 inhibits v_1
- **m$_3$** $[(v_1 \geq 2) \wedge (v_2 \geq 1)] \rightarrow v_2$: an activating dimer of v_2

René Thomas' Semantic: asynchronous dynamic

Regulatory graph:



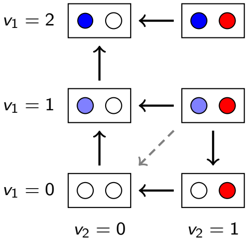
Set of resources:

- ω_{v_1}: m_1, m_2
- ω_{v_2}: m_3

K parameters:

$$K_{v_1,\emptyset} = 0$$
$$K_{v_1,m_1} = 0$$
$$K_{v_1,m_2} = 1$$
$$K_{v_1,m_1,m_2} = 2$$
$$K_{v_2,\emptyset} = 0$$
$$K_{v_2,m_3} = 1$$

Asynchronous transition graph:

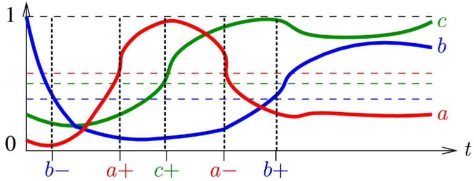


- System dynamics depends on K parameters of the form K_{v,ω_v} where ω_v is a resource of v.
- From a regulatory graph, number of parameterizations : $\prod_v (d^+(v) + 1)^{2^{d^-(v)}}$ where $d^+(v)$ and $d^-(v)$ are resp. the outdegree and indegree of v.

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postconditions: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

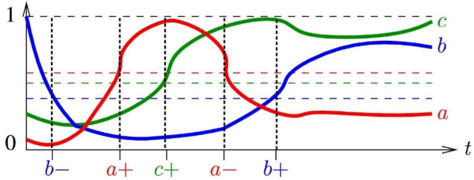
$$H_{ex} : \quad \{a = 0, b = 1, c = 0\} \;\; b- \; ; \; a+ \; ; \; c+ \; ; \; a- \; ; \; b+ \;\; \{a = 0, b = 1, c = 1\}$$

Postcondition noted Q

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postcondition: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

$$H_{ex} : \quad \{a = 0, b = 1, c = 0\} \;\; b- \; ; \; a+ \; ; \; c+ \; ; \; a- \; ; \; b+ \;\; \{a = 0, b = 1, c = 1\}$$

↓

New postcondition noted Q_1 :

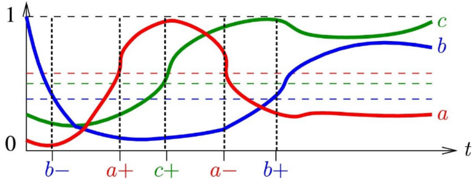
$$\underline{\mathbf{K_{b,\omega} \geq 1}} \wedge a = 0 \wedge b = 0 \wedge c = 1$$

(A genetically modified Hoare logic, Bernot et al., 2019)

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postcondition: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

$$H_{ex} : \{a = 0, b = 1, c = 0\} \; b- \; ; \; a+ \; ; \; c+ \; ; \; a- \; ; \; b+ \; \{a = 0, b = 1, c = 1\}$$

↓

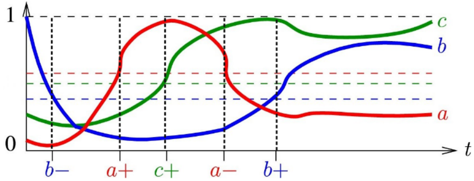
New postcondition noted Q_2 :

$$(\mathbf{K_{b,\omega} \geq 1}) \wedge (\mathbf{K_{a,\omega} < 1}) \wedge a = 1 \wedge b = 0 \wedge c = 1$$

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postcondition: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

$$H_{ex}: \quad \{a=0, b=1, c=0\} \ \ b- \ ; \ a+ \ ; \ c+ \ ; \ a- \ ; \ b+ \ \ \{a=0, b=1, c=1\}$$

↓

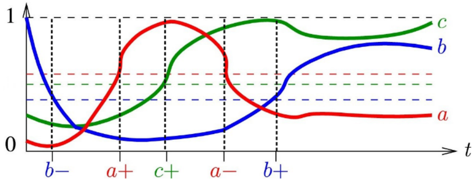
New postcondition noted Q_2 :

$$\dots \qquad \underline{(\mathbf{K_{b,\omega}} \geq \mathbf{1}) \wedge (\mathbf{K_{a,\omega}} < \mathbf{1})} \wedge a=1 \wedge b=0 \wedge c=1$$

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postcondition: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

$$H_{ex}: \quad \{a=0, b=1, c=0\} \quad b- \; ; \; a+ \; ; \; c+ \; ; \; a- \; ; \; b+ \quad \{a=0, b=1, c=1\}$$

$\downarrow$

Q_5

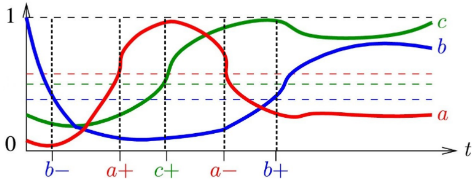
Weakest Precondition (WP)

$$\underline{(\mathbf{K}_{\mathbf{b},\omega} \geq \mathbf{1}) \wedge (\mathbf{K}_{\mathbf{a},\omega} < \mathbf{1}) \wedge (\mathbf{K}_{\mathbf{c},\omega} \geq \mathbf{1}) \wedge (\mathbf{K}_{\mathbf{a},\omega} \geq \mathbf{1}) \wedge (\mathbf{K}_{\mathbf{b},\omega} < \mathbf{1})} \wedge a=0 \wedge b=1 \wedge c=0$$

The genetically modified Hoare logic

Hoare triple noted H: {Pre} Path {Post}

- Precondition: a=0, b=1, c=0
- Path: b-; a+; c+; a-; b+
- Postcondition: a=0, b=1, c=1



Normalised expression profiles from a biological experiment

Path: $b-; a+; \exists\,((\mathbf{a}+; \mathbf{c}+), (\mathbf{c}+; \mathbf{a}+)); a-; b+$

↓

Disjunctive WP

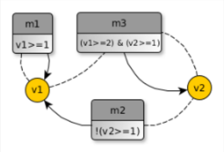
Path: $b-; a+; \forall\,((\mathbf{a}+; \mathbf{c}+), (\mathbf{c}+; \mathbf{a}+)); a-; b+$

↓

Conjunctive WP

TotemBioNet workflow

Input.smb



Regulatory Graph
(.graphml)

+

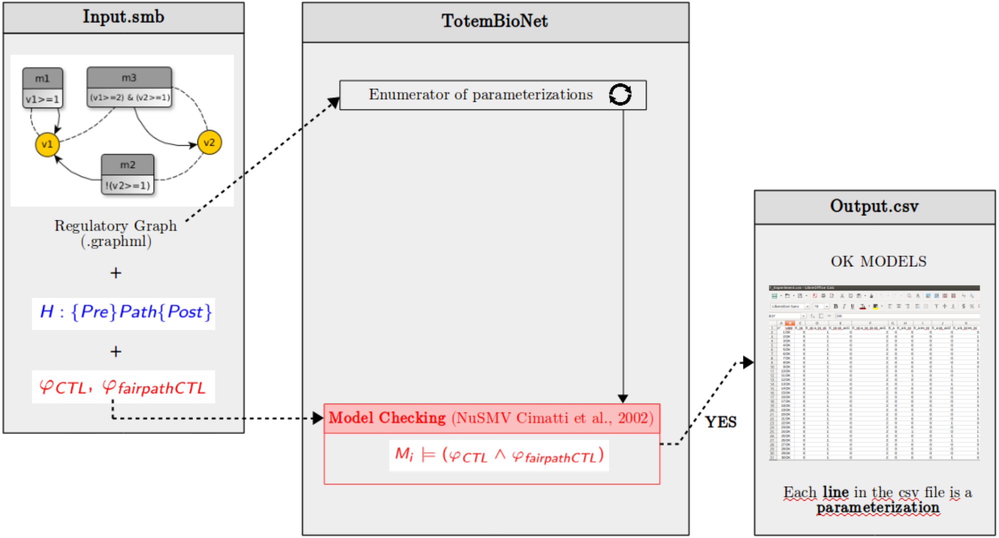
$H : \{Pre\} Path \{Post\}$

+

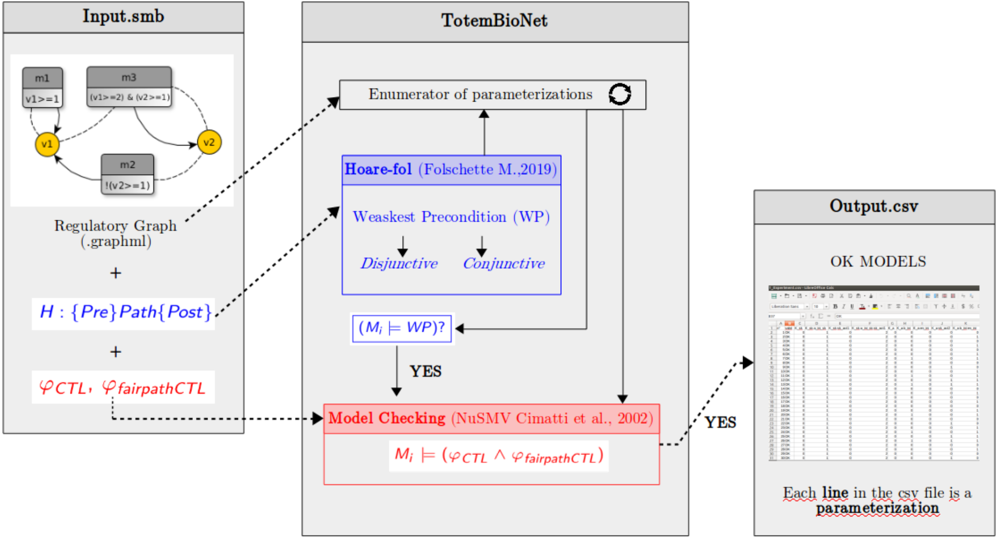
$\varphi_{CTL}, \varphi_{fairpathCTL}$

TotemBioNet

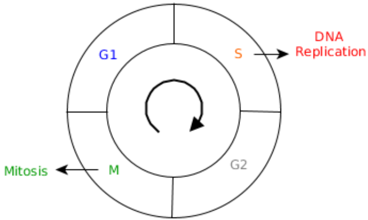
TotemBioNet workflow



TotemBioNet workflow

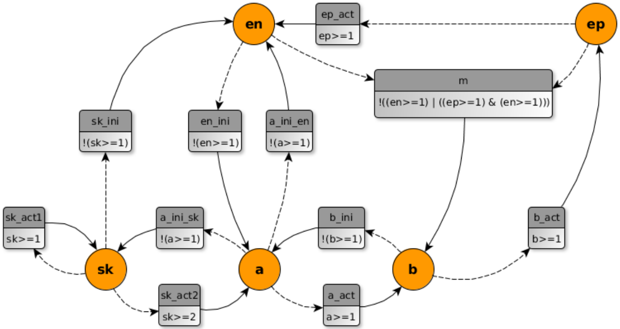


A multivalued mammalian cell cycle with 5 abstract variables



- ***sk*** : CycE/Cdk2
- ***a*** : CycA/Cdk1
- ***b*** : CycB/Cdk1
- ***en*** : APC-cdh1, Wee1, p21, p27
- ***ep*** : APC-cdc20

Behaegel et al., JBCB 2016



DEMO 1 DEMO 2

Graph made with yEd : `https://www.yworks.com/products/yed`

Verification of the cell cycle with TotemBioNet

Cell cycle and phases (G1,S,G2,M):

$$\mathbf{H_{init}} : \{\ G1_{init}\ \}\ \begin{matrix} sk+; sk+; en-; \\ a+; sk-; sk-; b+; \\ a-; ep+; \\ en+; b-; ep-; \end{matrix}\ \{\ G1_{init}\ \}$$

A globally cyclic behaviour:

$$\varphi_{cyclic} \equiv G1_{init} \Rightarrow AX(AF(G1_{init}))$$

with $G1_{init}$ the state $sk = 0, ep = 0, a = 0, b = 0, en = 1$.

Experiment	Hoare triple	$\|H\|^1$	Temporal logic formula	$\|S\|^3$	Computation Time (s)[2]
1 (DEMO)	H_{init}	676	φ_{cyclic}	609	6.1

[1] $|H|$: number of parameterizations satisfying the Hoare triple, $|S|$ satisfying both Hoare triple and formulas
[2] Performed on an Intel Core i7-8650U processor, 1.90GHz, 8 cores.

Verification of an hypothesis about cell cycle phases with TotemBioNet

$$\mathbf{H_{perm}} : \left\{ \; G1_{init} \; \right\} \quad \begin{array}{c} Forall((sk+;\, sk+;\, en-), (sk+;\, en-;\, sk+), (en-;\, sk+;\, sk+)); \\ Forall((a+;\, sk-;\, sk-;\, b+), (a+;\, sk-;\, b+;\, sk-), (a+;\, b+;\, sk-;\, sk-), (sk-;\, a+;\, sk-;\, b+), \\ (sk-;\, a+;\, b+;\, sk-), (b+;\, a+;\, sk-;\, sk-), (sk-;\, sk-;\, a+;\, b+), (sk-;\, b+;\, a+;\, sk-), \\ (b+;\, sk-;\, a+;\, sk-), (sk-;\, sk-;\, b+;\, a+), (sk-;\, b+;\, sk-;\, a+), (b+;\, sk-;\, sk-;\, a+)) \\ Forall((ep+;\, a-), (a-;\, ep+)); \\ Forall((en+;\, b-;\, ep-), (en+;\, ep-;\, b-), (ep-;\, b-;\, en+), \\ (ep-;\, en+;\, b-), (b-;\, en+;\, ep-), (b-;\, ep-;\, en+)); \end{array} \quad \left\{ \; G1_{init} \; \right\}$$

Exp	Hoare triple	$\lvert H \rvert$[3]	Temporal logic formula	$\lvert S \rvert$[5]	Computation Time (s)
2 (DEMO)	H_{perm}	0	φ_{cyclic}	0	0.24

[3] $|H|$: number of parameterizations satisfying the Hoare triple, $|S|$ satisfying both Hoare triple and formulas

Verification of an hypothesis about cell cycle phases with TotemBioNet

$\mathbf{H_{perm}}$: $\{\ G1_{init}\ \}$

$$Forall((sk+;\ sk+;\ en-),(sk+;\ en-;\ sk+),(en-;\ sk+;\ sk+));$$
$$Forall((a+;\ sk-;\ sk-;\ b+),(a+;\ sk-;\ b+;\ sk-),(a+;\ b+;\ sk-;\ sk-),(sk-;\ a+;\ sk-;\ b+),$$
$$(sk-;\ a+;\ b+;\ sk-),(b+;\ a+;\ sk-;\ sk-),(sk-;\ sk-;\ a+;\ b+),(sk-;\ b+;\ a+;\ sk-),$$
$$(b+;\ sk-;\ a+;\ sk-),(sk-;\ sk-;\ b+;\ a+),(sk-;\ b+;\ sk-;\ a+),(b+;\ sk-;\ sk-;\ a+))$$
$$Forall((ep+;\ a-),(a-;\ ep+));$$
$$Forall((en+;\ b-;\ ep-),(en+;\ ep-;\ b-),(ep-;\ b-;\ en+),$$
$$(ep-;\ en+;\ b-),(b-;\ en+;\ ep-),(b-;\ ep-;\ en+));$$

$\{\ G1_{init}\ \}$

$\mathbf{H_{permG1}}$: $\{\ G1_{init}\ \}$

$$\underline{Forall}((sk+;\ sk+;\ en-),(sk+;\ en-;\ sk+),(en-;\ sk+;\ sk+));$$
$$a+;\ \underline{Forall}((sk-;\ sk-;\ b+),(sk-;\ b+;\ sk-),(b+;\ sk-;\ sk-));$$
$$\underline{Forall}((ep+;\ a-),(a-;\ ep+));$$
$$en+;\ b-;\ ep-;$$

$\{\ G1_{init}\ \}$

Exp	Hoare triple	$\|H\|$[4]	Temporal logic formula	$\|S\|$[5]	Computation Time (s)
2 (DEMO)	H_{perm}	0	φ_{cyclic}	0	0.24
3 (DEMO)	H_{permG1}	260	φ_{cyclic}	240	2.4

[4] $|H|$: number of parameterizations satisfying the Hoare triple, $|S|$ satisfying both Hoare triple and formulas

Verification of cell cycle checkpoints with TotemBioNet

$$\mathbf{H_{permG1}} : \{\ G1_{init}\ \} \begin{array}{c} \underline{Forall}((sk+;sk+;en-),(sk+;en-;sk+),(en-;sk+;sk+)); \\ a+;\underline{Forall}((sk-;sk-;b+),(sk-;b+;sk-),(b+;sk-;sk-)); \\ \underline{Forall}((ep+;a-),(a-;ep+)); \\ en+;b-;ep-; \end{array} \{\ G1_{init}\ \}$$

$$\varphi_{G1/S} \equiv \left(\ G1_{init}\ \right) \Rightarrow \neg \left(\begin{array}{l} EX(\mathbf{a{=}1} \wedge EX(sk=1 \wedge EX(en=0 \wedge EX(sk=2)))) \\ \vee\ EX(sk=1 \wedge EX(\mathbf{a{=}1} \wedge EX(en=0 \wedge EX(sk=2)))) \\ \vee\ EX(sk=1 \wedge EX(en=0 \wedge EX(\mathbf{a{=}1} \wedge EX(sk=2)))) \\ \vee\ EX(\mathbf{a{=}1} \wedge EX(en=0 \wedge EX(sk=1 \wedge EX(sk=2)))) \\ \vee\ EX(sk=1 \wedge EX(\mathbf{a{=}1} \wedge EX(sk=1 \wedge EX(en=0)))) \\ \vee\ EX(sk=1 \wedge EX(sk=2 \wedge EX(\mathbf{a{=}1} \wedge EX(en=0)))) \\ \vee\ EX(\mathbf{a{=}1} \wedge EX(sk=1 \wedge EX(sk=2 \wedge EX(en=0)))) \\ \vee\ EX(en=0 \wedge EX(\mathbf{a{=}1} \wedge EX(sk=1 \wedge EX(sk=2)))) \\ \vee\ EX(en=0 \wedge EX(sk=1 \wedge EX(\mathbf{a{=}1} \wedge EX(sk=2)))) \end{array} \right)$$

- with $G1_{init}$: $sk=0, ep=0, a=0, b=0, en=1.$
- $EX(\mathbf{a{=}1} \wedge EX(sk=1 \wedge EX(en=0 \wedge EX(sk=2))))$ is equivalent to the path $a+;sk+;en-;sk+$

Verification of cell cycle checkpoints with TotemBioNet

Exp	Hoare triple	$\lvert H \rvert$	Temporal logic formula	$\lvert S \rvert$	Computation Time (s)
3	H_{permG1}	260	$\varphi_{cyclic} \wedge \varphi_{G2/M} \wedge \varphi_{G1/S}$	**28**	2.9

TotemBioNet's features in a nutshell

WHAT: automates parameters' identification using two formal methods: Hoare logic and fair path/CTL combined with model-checking

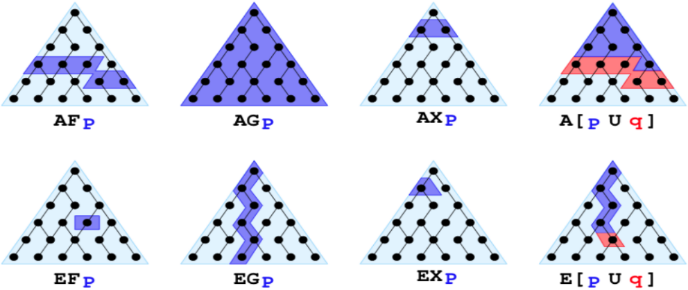
PURPOSE: formalization biological knowledge and their quick verification

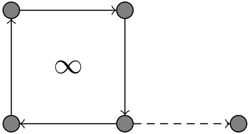
WHERE: `https://gitlab.com/totembionet/totembionet`[5]

IMPROVEMENTS: incremental analysis of parameterizations + a Jupyter notebook.

[5] only on Linux and Mac

CTL and fair-path CTL





- *p* and *q* two properties
- Temporal modalities made up of 2 letters : a *quantifier* and a *temporal operator*
- **Quantifiers**: A,E, **Temporal operators**: F,G,X,U

https://www.uio.no/studier/emner/matnat/ifi/INF5140/v17/slides/clt-antonio.pdf