Logical and incremental formalization of cell cycle checkpoints

Déborah Boyenval[1,2], Gilles Bernot[1], Jean-Paul Comet[1] and Franck Delaunay[2]

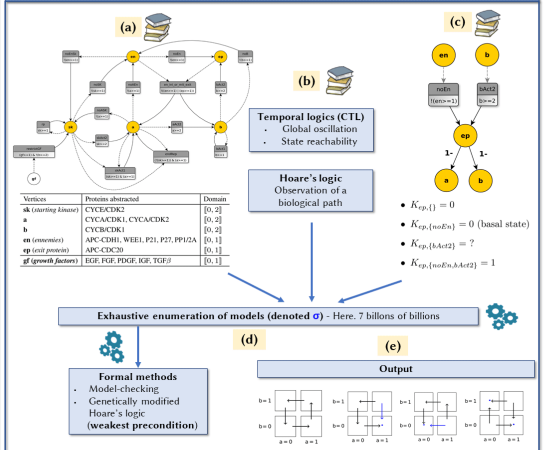
Université Côte d'Azur. 1- I3S laboratory, Sophia Antipolis 2- iBV, Nice

{Deborah.Boyenval, Gilles.Bernot, Jean-Paul.Comet, Franck.Delaunay}@univ-cotedazur.fr

Checkpoints ensure the integrity of DNA during the cell cycle, which is a succession of molecular and cellular events leading to the division of a mother cell into two genetically identical daughter cells. The DNA is first duplicated (S-phase), then equally distributed between the opposing poles of the cell, which finally divides into two independent daughter cells (M-phase). Two additional phases depict the process of cell preparation before S and M phases: respectively the G1 and G2 phases. Many modeling studies investigating checkpoints are based on ODE systems while the checkpoint concept itself is fundamentally discrete. So far to our knowledge, very few qualitative models have yet attempted to formally define this **discrete concept**, as the notion of discrete cell cycle phase is still fuzzy from a formal point of view.

1 Logical modeling of dynamic biological behaviours : the mammalian cell cycle case

TotemBioNet: Automated identification of logical models [1]



Vertices	Proteins abstracted	Domain
sk (*starting kinase*)	CYCE/CDK2	[0, 2]
a	CYCA/CDK1, CYCA/CDK2	[0, 2]
b	CYCB/CDK1	[0, 2]
en (*ennemies*)	APC-CDH1, WEE1, P21, P27, PP1/2A	[0, 1]
ep (*exit protein*)	APC-CDC20	[0, 1]
gf (*growth factors*)	EGF, FGF, PDGF, IGF, TGFβ	[0, 1]

(a) Our cell cycle interaction graph, where the nature of each regulation is encoded in **propositional logic**, following the **René Thomas' formalism** [2].

(b) Dynamical knowledge can be encoded either in **Hoare's logic** [3], **CTL** [2], or
(c) by directly constraining the value of the parameters $\mathbf{K_{v,\omega}}$

The family of parameters $\mathbf{K_{v,\omega}}$ allows us to build an asynchronous transition graph

(d) A **logical model** refers to a substitution of all symbols $\mathbf{K_{v,\omega}}$

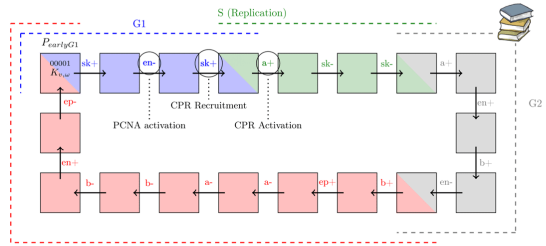
(e) TotemBioNet efficiently combines **model-checking** and **genetically modified Hoare** logic to exhaustively identify the **set of models** that satisfy the dynamic properties.

The canonical cell cycle defined using Hoare's logic

$$H_c \equiv \{P_{earlyG1}\}\ \underbrace{sk+;en-;sk+;}_{G1}\ \underbrace{a+;sk-;sk-;}_{S}\ \underbrace{a+;en+;b+;en-;}_{G2}\ \underbrace{b+;ep+;a-;a-;b-;b-;ep-;en+}_{M}\ \{P_{earlyG1}\}$$

- $P_{earlyG1}$ is the **precondition** (the initial state of the system)
- ... and the **postcondition** (the final state)
- Between: A succession of events where each event is of the form **v**+ or **v**-.

$P_{earlyG1}$ define a state where *sk=0, a=0, b=0, ep=0 and en=1* (**00001**).



The H_c Hoare triple from a semantic point of view - Each edge joining two states is built using a parameter $\mathbf{K_{v,\omega}}$.

$$wp(H_c) \equiv$$

$$sk=0 \wedge a=0 \wedge b=0 \wedge ep=0 \wedge en=1$$

$$(K_{sk,\{noASK,restrictGF\}} > 0) \wedge (K_{en,\{noAEn,noB\}} < 1) \wedge (K_{sk,\{noASK,rp,noEnSk,restrictGF\}} < 1) \wedge$$
$$(K_{a,\{sk:Act2,EnEP\}} > 0) \wedge (K_{sk,\{rp,noEnSk,restrictGF\}} < 2) \wedge (K_{sk,\{rp,noEnSk,restrictGF\}} < 1) \wedge$$
$$K_{a,\{endRep,EnEp\}} > 1) \wedge (K_{en,\{noSK,noB\}} > 0) \wedge (K_{b,\{aAct2\}} > 0) \wedge (K_{en,\{noSK\}} < 1) \wedge$$
$$(K_{b,\{EnEp,bAct1,aAct2\}} > 1) \wedge (K_{ep,\{noEnEp,bAct2\}} > 0) \wedge (K_{a,\{endRep\}} < 2) \wedge (K_{a,\{endRep\}} < 1) \wedge$$
$$(K_{b,\{bAct1\}} < 2) \wedge (K_{b,\{bAct1\}} < 1) \wedge (K_{en,\{noAEn,noSK,noB\}} > 0) \wedge (K_{ep,\{\}} < 1)$$

2 Yet another cell cycle model but what is a cell cycle checkpoint?

The TotemBioNet tool identifies the set of models σ compatible with the canonical cell cycle and some of its major properties expressed in CTL. **32 models are identified**:

How many of them satisfies the discrete notion of checkpoint?

A checkpoint ensures that any event **Y** that **can start** a phase does not occur before the completion of any event **X** that **can end** the previous phase.

The generic predicate *checkpoint*(π_i,π_{i+1})

$$\boldsymbol{checkpoint}(\pi_i,\pi_{i+1}) \iff \forall \pi_i \in [G1,S,G2,M],\ \exists \sigma \mid\ \forall X,\forall Y,\ \boldsymbol{canEnd}_\sigma(X,\pi_i) \wedge \boldsymbol{canStart}_\sigma(Y,\pi_{i+1}) \implies \boldsymbol{isRequired}_\sigma(X,Y)$$

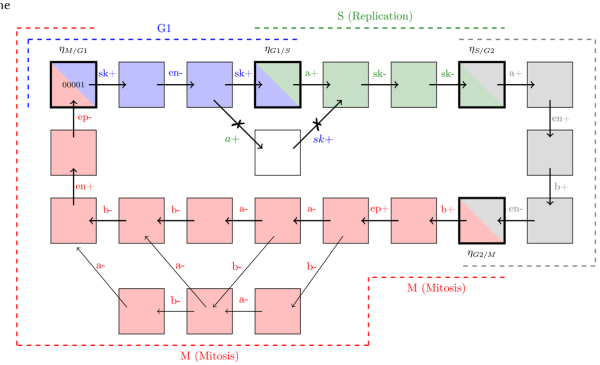
A single ordering of p has been defined.
However, current knowledge cannot ensure that there is only one possible order.

$$\boldsymbol{canEnd}_\sigma(\boldsymbol{X},\pi_i) \iff \exists\ p' \in \boldsymbol{permutations}(p) \mid (\sigma(\boldsymbol{wp}(\{P_\pi\}p'\{Q_\pi\})) \wedge X = \boldsymbol{last}(p'))$$
$$\boldsymbol{canStart}_\sigma(\boldsymbol{Y},\pi_i) \iff \exists\ p' \in \boldsymbol{permutations}(p) \mid (\sigma(\boldsymbol{wp}(\{P_\pi\}p'\{Q_\pi\})) \wedge Y = \boldsymbol{first}(p'))$$

- $\mathbf{P_\pi}$ (resp. $\mathbf{Q_\pi}$): the unique initial (resp. final) state of a phase π_i
- **p**: the set of events to be completed within a phase π_i

$\boldsymbol{\sigma(K_{v,\omega})} = \mathbf{1}$ if the parameter $K_{v,\omega}$=1 in the considered logic model σ.

The predicate ***wp*** queries TotemBioNet to:
(a) construct the weakest **p**recondition for all orderings of events in the phase π_i
(b) identify **permutations** for which the weakest **p**recondition is satisfiable



$Q_{\pi i}$ is called the checkpoint state and is denoted $\eta_{\pi i/\pi i+1}$.

$$\boldsymbol{isRequired}_\sigma(X,Y) \iff (\sigma(K_{y,\omega_{beforeX}}) - \eta_{beforeX}) \times (\sigma(K_{y,\omega_{afterX}}) - \eta_{afterX}) \leq 0$$

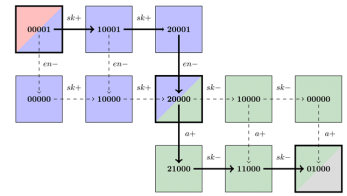
The ***isRequired*** clause is false for G1/S if the state $\eta_{G1/S}$ is bypassed to reach S from G1.

3 Calculation of the solutions X and Y – Definition of a standard phase

SWI Prolog

π_i	Canonical path (p)	Initial state (P_π)	canStart(p,Y)	canEnd(p,X)
G1	{sk+,en-,sk+}	sk=0,ep=0,a=0,b=0,en=1	Y=[sk+]	X=[sk+]
S	{a+,sk-,sk-}	sk=2,ep=0,a=0,b=0,en=0	Y=[a+]	X=[sk-]
G2	{a+,en+,b+,en-}	sk=0,ep=0,a=1,b=0,en=0	Y=[a+]	X=[en-]
M	{b+,ep+,a-,a-,b-,b-,en+,ep-}	sk=0,ep=0,a=2,b=1,en=0	Y=[b+]	X=[a-,b-,en+,ep-]

G1 and S **hyper-pavements** and their biologically feasible paths



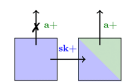
- The modelled biological knowledge admits several orders of events within the M phase (33 paths)
- The M phase is poorly constrained by biological knowledge
- A **standard phase** is defined by the set of states traversed by the set of feasible permutations of p from Pπi to Qπi

Calculation of the solutions of the predicate *checkpoint*(π_i,π_{i+1})

SWI Prolog

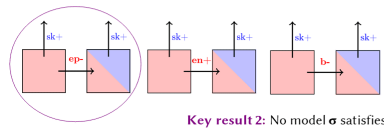
Checkpoint	Evaluation	$\lvert\sigma \models checkpoint(\pi_i,\pi_{i+1})\rvert$
G1/S	True	16/32
S/G2	True	32/32
G2/M	True	32/32
M/G1	False	0/32

Computation of the satisfiability of the checkpoint(πi,πi+1) predicate on the 32 remaining models



$$isRequired(sk+,a+) \equiv ((\sigma(K_{a,\{EnEp\}}) - 0) \times (\sigma(K_{a,\{EnEp,skAct2\}}) - 0)) \leq 0$$

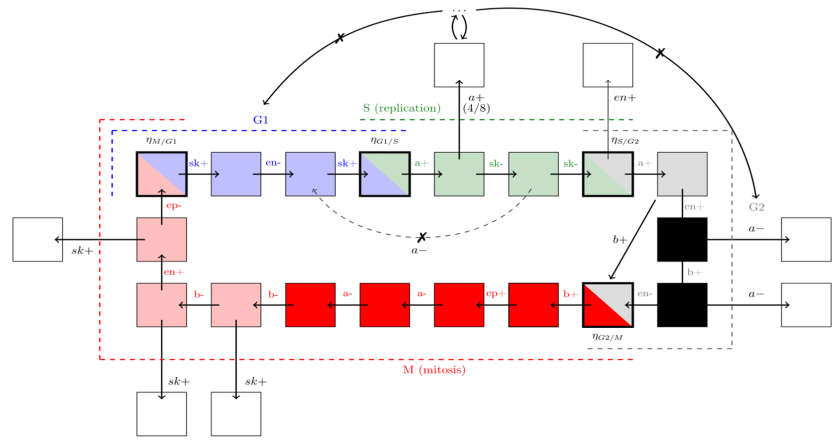
Key result 1: 16 models **σ** satisfy the predicate isRequired for G1/S
Only models for which $K_{a,[EnEp]} = 0$ are solutions of the G1/S checkpoint.



$$isRequired(ep-,sk+) \equiv ((\sigma(K_{sk,\{noASk,restrictGF\}}) - 0) \times (\sigma(K_{sk,\{noASk,restrictGF\}}) - 0)) \leq 0$$

Key result 2: No model **σ** satisfies the predicate isRequired for M/G1 - *The example of one couple (X,Y)*

4 Prospects: Mitosis exit checkpoint and nonstandard checkpoints formalization



[1] Déborah Boyenval, Gilles Bernot, Hélène Collavizza, and Jean-Paul Comet. What is a cell cycle checkpoint? the TotemBioNet answer. *In 18th International Conference on Computational Methods in Systems Biology (CMSB 2020)*, 2020.
[2] Gilles Bernot, Jean-Paul Comet, Adrien Richard, and Janine Guespin. Application of formal methods to biological regulatory networks: Extending Thomas' asynchronous logical approach with temporal logic. *Journal of theoretical biology, 229:339–47*, 09 2004.
[3] Gilles Bernot, Jean-Paul Comet, Zohra Khalis, Adrien Richard, and Olivier Roux. A genetically modified hoare logic. *Theoretical Computer Science, 765*, 06 2015.