



HAL
open science

Formal analysis of the Wnt/ β -catenin pathway through statistical model checking

Paolo Ballarini, Emmanuelle Gallet, Pascale Le Gall, Matthieu Manceny

► To cite this version:

Paolo Ballarini, Emmanuelle Gallet, Pascale Le Gall, Matthieu Manceny. Formal analysis of the Wnt/ β -catenin pathway through statistical model checking. International Symposium On Leveraging Applications of Formal Methods, Verification and Validation , Oct 2014, Corfu, Greece. hal-01819866

HAL Id: hal-01819866

<https://hal.science/hal-01819866>

Submitted on 21 Jun 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Formal analysis of the Wnt/ β -catenin pathway through statistical model checking

Paolo Ballarini¹, Emmanuelle Gallet¹, Pascale Le Gall¹, and Matthieu Manceny²

¹ Laboratoire MAS, Ecole Centrale Paris, 92195 Châtenay-Malabry, France
email: {emmanuelle.gallet, pascale.legall, paolo.ballarini}@ecp.fr

² Laboratoire LISITE, ISEP, 28 Rue Notre-Dame-des-Champs 75006 Paris, France
email: matthieu.manceny@isep.fr

Abstract. The Wnt/ β -catenin signalling pathway plays an important role in the proliferation of neural cells, and hence it is the main focus of several research aimed at understanding neurodegenerative pathologies. In this paper we consider a compact model of the basic mechanisms of the Wnt/ β -catenin pathway and we analyse its dynamics by application of an expressive temporal logic formalism, namely the Hybrid Automata Stochastic Logic. This allows us to formally characterise, and effectively assess, sophisticated aspects of the Wnt/ β -catenin pathway dynamics.

Keywords: HASL Model Checking, Stochastic modelling, biological pathways, Wnt/ β -catenin

1 Introduction

Systems Biology [11] is concerned with the development of formalisms for building “realistic” models of biological systems, i.e. models capable of reproducing wet-lab observations. A biological model consists of a set biochemical agents (i.e. species) whose interactions are expressed by a set of *reaction equations*. This leads to either a continuous-deterministic interpretation (i.e. in terms of a system of differential equations), or to a discrete-stochastic interpretation (i.e. in terms of a discrete-state stochastic process).

Stochastic modelling and systems biology. Within the discrete-stochastic semantics realm, which is what we consider in this work, molecular interactions are assumed to be of stochastic nature hence biochemical reactions occur according to probability distributions. In this case what modellers normally do is to generate one (or several) trajectory(ies) through stochastic simulation and observe the evolution of the species (under different model’s configurations) in order to figure out how a given aspect of the model’s dynamics is affected by the various elements of the model (i.e. what species/reactions is responsible for a given observed behaviour). Such an approach has two main advantages: its simplicity and its low computational cost (the runtime for generating a single trajectory or a normally small number of trajectories is very low even for large models). On

the other hand the main disadvantage is that it is little formal, meaning that the modeller must draw conclusions based only on the observation of a single (stochastic) trajectory (or of a trajectory obtained by averaging a normally small number of trajectories).

Stochastic model checking and systems biology. Stochastic model checking [12] (SMC) is a formal technique that allows the modeller to formally express relevant properties in terms of a (stochastic) temporal logic and to assess them against a given stochastic model. This is achieved through an automatic procedure which can either provide an *exact answer* through exhaustive exploration of the model’s state space (i.e. *numerical model checking* [2]) or an *estimated answer* resulting from a finite sampling of the model’s trajectory (i.e. *statistical model checking* [13]). SMC has at least two main advantages with respect to informal approaches: first it provides the modeller with a language for capturing relevant properties formally; second the answer it calculates (e.g. probability that a property is satisfied by the model) are either exact (i.e. they reflect the complete set of possible behaviours of the model) or are accurate estimates (i.e. calculated over a sufficiently large sample of trajectories). The effectiveness of SMC in systems biology applications is demonstrated by an ever increasing number of publications, e.g.[8,10,5].

β -catenin and the WNT pathway. In cellular biology signalling pathways are basic mechanisms responsible for controlling a cell’s life-cycle. Simply speaking a signalling pathway represents a cascade of biochemical reactions which is triggered by a specific signal (i.e. type of molecules) whose presence, normally at the cell membrane, activates the cascade leading to the “transmission” of the signal inside the cell (i.e. cytosol and/or nucleus). In this paper we study a model of the Wnt/ β -catenin pathway, a signalling pathway known to be involved in the pathological degeneration of neuronal cells [14].

Our contribution. In this work we present preliminary results of application of formal analysis, based on the so-called Hybrid Automata Stochastic Logic (HASL) statistical model checking, to a model of the Wnt/ β -catenin pathway presented in [15]. In particular we show how one can define specific HASL formulae for assessing sophisticated characteristics of the Wnt/ β -catenin pathway dynamics. This includes, for example, measuring the temporal location and the amplitude of transient peaks of nuclear β -catenin, exhibited by certain initial conditions, or assessing its oscillatory character resulting from other conditions. If in [15] the analysis of the Wnt/ β -catenin model is simply done through plotting of simulated trajectories, here we move analysis to a higher and more formal level by demonstrating how, through model checking, one gains access to the analysis of sophisticated dynamical aspects of the Wnt/ β -catenin pathway.

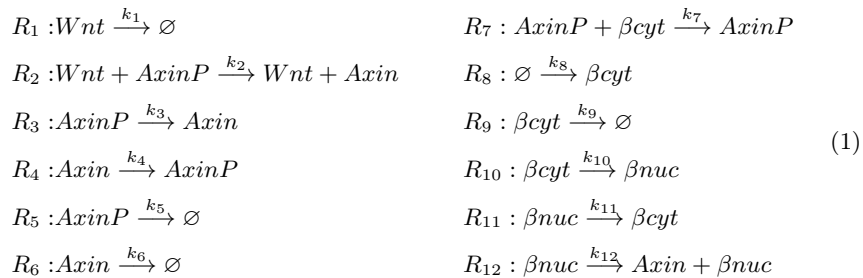
Paper organisation. We introduce the Wnt/ β -catenin mechanism in Section 2 and describe the model presented in [15] which we have used for our analysis. In

Section 3 we give a concise description of the HASL statistical model checking formalism. In Section 4 we present the results obtained by application of HASL model checking to the analysis of Wnt/ β -catenin model. We wrap up the paper with some conclusive remarks and future perspectives in Section 5.

2 A model of the Wnt/ β -catenin pathway

Neurodegeneration is the process of progressive lost of structure/function of neuronal cells (i.e. neurons) which is at the basis of many neurodegenerative diseases, such as, for example, the Parkinson's disease, Alzheimer's disease and the Amyotrophic lateral sclerosis. Research in this field is particularly focused on the growth of *in vitro* population of neural cells that may potentially be used in replacement therapies for neurodegenerative diseases. Cultivated cells undergo so-called proliferation, a process of successive cell divisions and potential differentiation into neurones and glial cells.

The Wnt/ β -catenin pathway is a signalling pathway known to be involved in the proliferation/differentiation of neural cells. Specific *in vitro* experiments [14] have exhibited a high activity of the Wnt/ β -catenin pathway during the differentiation of ReNcell VM (RVM) cells, i.e. a type of cells derived from the brain of a fetus and that are believed to be an appropriate model for replacement therapies in neurodegenerative pathologies. The activity of the Wnt/ β -catenin is summarised as follows: in absence of extracellular Wnt molecules (normally at cell's membrane), a degradation complex causes the phosphorylation and subsequent destruction of β -catenin located in the cell's cytosol (denoted β_{cyt}); on the other hand in presence of Wnt proteins, the degradation complex is inactivated resulting in accumulation of β_{cyt} . Furthermore from the cytosol β -catenin undergoes a (reversible) relocation to the nucleus (denoted β_{nuc}) wherein it activates the expression of one component of its degradation complex, i.e. the Axin protein. The above described mechanism is captured by a core-version of the Wnt/ β -catenin pathway model presented in [15]. This consists of the twelve biochemical reactions illustrated by equations (1).



The model consists of three basic molecular species: the Axin protein, which can be either in normal ($Axin$) or phosphorylated ($AxinP$) form, the Wnt protein (Wnt) and the β -catenin which can be either located in the cytosol

(β_{cyt}) or in the nucleus (β_{nuc}). Equations (1) account for the following aspects: two reversible events, i.e. the phosphorylation of Axin (reactions R_4 and R_3) and the relocation of β -catenin from/to cytosol/nucleus (reactions R_{10} and R_{11}); the *Wnt* enhanced de-phosphorylation of *Axin* (reaction R_2); the nuclear β -catenin (i.e. β_{nuc}) regulated expression of *Axin* (reaction R_{12}); the phosphorylated *AxinP* enhanced degradation of cytosolic β -catenin (i.e. β_{cyt}) (reaction R_7)³; the constant (DNA regulated) expression of cytosolic β -catenin (i.e. β_{cyt}) (reactions R_8); the degradation of all species i.e. *Wnt* (reactions R_1), Axin in either form (reactions R_5 or R_6) and β_{cyt} (reaction R_9).

In this paper we focus on the discrete-stochastic interpretation of Equations (1), hence species populations are expressed in terms of number of molecules and reactions are of stochastic nature and are assumed to obey the *mass action* law (meaning that a reaction’s rate is proportional to the current population of the reactants, except for R_8 whose rate is constant). With respect to the model configuration we consider two basic sets A and B of parameter values respectively taken from the sets 3 and 4 in [15], and indicated in Table 1.

initial populations (<i>mol.</i>)			rate constants (<i>mol. · min⁻¹</i>)					
par. id	Set A	Set B	par. id	Set A	Set B	par. id	Set A	Set B
n β_{cyt}	11145	12989	k_1	0.6	0.27	k_6	$2.4 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$
n β_{nuc}	4532	5282	k_2	10	20	k_7	$3 \cdot 10^{-4}$	$2.1 \cdot 10^{-4}$
nAxin	144	252	k_3	0.03	0.03	k_8	420	600
nAxinP	125	219	k_4	0.03	0.03	k_9	$1.13 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$
nWnt	1000	1000	k_5	$4.48 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	k_{10}	0.0549	0.0549
			k_{11}	0.135	0.135	k_{12}	$2 \cdot 10^{-4}$	$4 \cdot 10^{-4}$

Table 1. Parameter sets for stochastic interpretation of Wnt/ β -catenin pathway model given by equations (1)

Throughout the remainder of the paper we will analyse the Wnt/ β -catenin model by comparing the dynamics corresponding to the two parameter sets of Table 1. Following [15], we will also consider two variants of the *basic model* (1). The first variant, denoted *Wnt-inject*, represents a single injection of an extra amount (i.e. 1000) of *Wnt* molecules in the system at a fixed delay d_i . The second variant, denoted *Wnt-doped*, represents the presence of a doping mechanism that kicks in at a given delay d_d and then it sustainably generates a fresh Wnt molecule at given frequency (assumed to be exponential distributed with parameter k_d).

³ notice that β_{nuc} dependent *Axin* expression and *AxinP* enhanced β_{cyt} degradation determine, *de facto*, a negative feedback loop between β -catenin and the Axin protein.

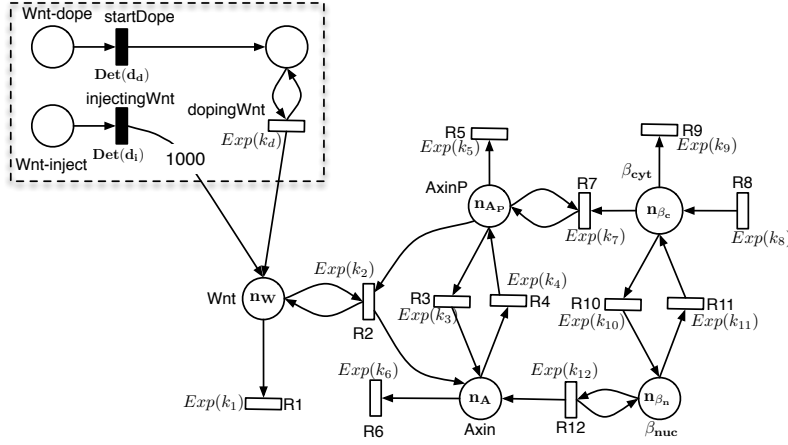


Fig. 1. GSPN model corresponding to Equations (1) of the Wnt/ β -catenin pathway.

2.1 Stochastic Petri Net model of the Wnt/ β -catenin pathway

The COSMOS [3] model checker which we used for analysing the Wnt/ β -catenin pathway model uses Generalised Stochastic Petri Net (GSPN) [1] as modelling formalism. A GSPN model is a bipartite graph consisting of two classes of nodes, *places* and *transitions*. Places (circle nodes) may contain *tokens* (representing the state of the modelled system) while transitions (bar nodes) indicate how tokens “flow” within the net (encoding the model dynamics). The state of a GSPN consists of a *marking* indicating the distribution of tokens throughout the places (i.e. how many tokens each place contains). A transition is enabled whenever all of its *input places* contains a number of tokens greater than or equal to the multiplicity of the corresponding (input) arc. An enabled transition may *fire* consuming tokens (in a number indicated by the multiplicity⁴ of the corresponding input arcs) from all of its input places and producing tokens (in a number indicated by the multiplicity of the corresponding output arcs) in all of its output places. Transitions can be either *timed* (denoted by thick bars) or *immediate* (denoted by thin bars). Timed transitions are associated to a probability distribution (e.g. Exponential, Uniform, Deterministic, etc). In the context of this paper GSPN places represent biological species (and their marking the molecular population of a species), whereas timed transitions represent chemical reactions. For more details on GSPN we refer the reader to the literature [1].

Figure 1 depicts the GSPN model encoding the Wnt/ β -catenin chemical equations (1). The net contains a place for each species of the Wnt/ β -catenin model and a transition for each reaction. Non filled-in transitions are exponentially distributed (with marking dependent rate) with rate-constant correspond-

⁴ the default multiplicity of an arc is 1 if different is explicitly indicated on the arc

ing to that of either parameter set of Table 1. The sub-net enclosed in dashed line box (top left) has been added in order to add the *Wnt-inject* and *Wnt-dope* behaviour to the basic model. In order to study the behaviour of the *Wnt-inject* (*Wnt-dope*) variant it suffices to add one token in the initial marking of place *Wnt_inject* (*Wnt_dope*). Notice that the black filled-in transitions of the *Wnt-inject*/*Wnt-dope* subnet are associated to deterministic delays.

Observing β_{nuc} dynamics on a single stochastic trajectory. Following [15] we first look at the dynamics of nuclear β -catenin (i.e. β_{nuc}) as observed along a single trajectory simulated over 24 hours (Figure 2 and Figure 3 with the units on x axis being minutes). Figure 2 compares the behavior of β_{nuc} for the two parameter sets (Table 1) of the basic Wnt/ β -catenin model in presence of Wnt (initial population of Wnt set to 1000).

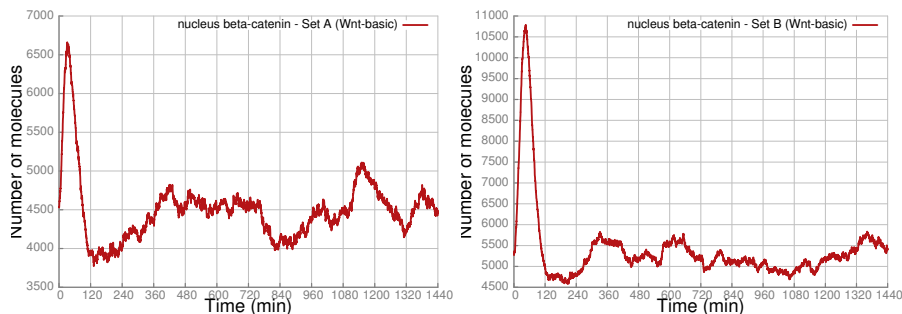


Fig. 2. Dynamics of β_{nuc} along a 24 hours single trajectory of the Wnt/ β -catenin pathway model with parameter set A (left) and set B (right)

The interpretation in this case is quite straightforward: the presence of an initial Wnt signal (i.e. of 1000 molecules) triggers a (delayed) peak in β_{nuc} which however quickly ends due to the steady degradation (and absence of reintegration) of Wnt. Eventually when all Wnt has faded away β_{nuc} noisily converges to a certain level. Figure 3 compares the dynamics of β_{nuc} in presence of *delayed injection* (i.e. *Wnt-inject* model injecting 1000 Wnt molecules at $d_i = 450$ minutes), and in presence of doping (i.e. model *Wnt-dope* with doping starting at time $d_d = 150$ minutes). The effect of delayed injection of 1000 molecules of Wnt at time $d_i = 450$ minutes is highlighted by the presence of the second peak (Figure 3 left). On the other hand the effect of starting a persistent doping of Wnt at time $d_d = 150$ minutes results in an oscillatory, yet rather irregular, behaviour of β_{nuc} (Figure 3 right). In Section 4 we are going to illustrate how to take advantage of the HASL formalism for formally capturing the relevant dynamical characteristics of the above shown trajectories.

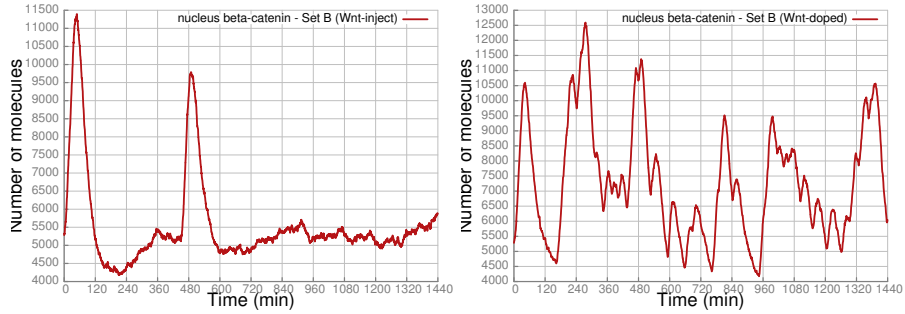


Fig. 3. Dynamics of β_{nuc} along a single trajectory of the Wnt/ β -catenin pathway model with re-injection of Wnt at $t = 450$ minutes (left) and with doping of Wnt started at $t = 150$ minutes (right) and parameter set B

3 HASL statistical model checking

The Hybrid Automata Stochastic Logic (HASL), introduced in [4], extends Deterministic Timed Automata (DTA) logics for addressing Markov chain models [9,7], by employing Linear Hybrid Automata (LHA) as instruments for addressing a general class of stochastic processes, namely that of Discrete Event Stochastic Processes (DESP). An HASL formula $\phi \equiv (\mathcal{A}, Z)$ consists of two elements: 1) \mathcal{A} , a *synchronising* LHA (i.e. an LHA enriched with DESP state and/or event *indicators*) and 2) Z a target expression (see grammar (2)) which expresses the quantity to be evaluated (either a measure of probability or, more generically, any real-valued measure).

Thus given a DESP model \mathcal{D} and a formula $\phi \equiv (\mathcal{A}, Z)$ the HASL model checking procedure employs stochastic simulation to samples trajectories of the synchronised process $\mathcal{D} \times \mathcal{A}$, and then use (only) the paths *selected* by \mathcal{A} (i.e. those paths of $\mathcal{D} \times \mathcal{A}$ that reach an accepting location of \mathcal{A}) for estimating the confidence-interval of the target measure Z . Such a procedure is implemented within the COSMOS [3] model checking framework, a tool which belongs to the fast expanding family of statistical model checkers (e.g. [6,16,17]). For practical reasons in HASL (and in particular within COSMOS) we employ GSPN as high-level language for expressing a DESP. Below we give the definition of DESP and LHA and informally describe the synchronisation process (we refer the reader to [4] for a more formal treatment).

Definition 1 (DESP). A *DESP* is a tuple $\mathcal{D} = \langle S, \pi_0, E, Ind, enabled, delay, choice, target \rangle$ where

- S is a (possibly infinite) set of states,
- $\pi_0 \in dist(S)$ is the initial distribution on states,
- E is a set of events,
- Ind is a set of functions from S to \mathbb{R} called state indicators (including the constant functions),

- $enabled: S \rightarrow 2^E$ are the enabled events in each state with for all $s \in S$, $enabled(s) \neq \emptyset$.
- $delay: S \times E \rightarrow dist(\mathbb{R}^+)$ is a partial function defined for pairs (s, e) such that $s \in S$ and $e \in enabled(s)$.
- $choice: S \times 2^E \times \mathbb{R}^+ \rightarrow dist(E)$ is a partial function defined for tuples (s, E', d) such that $E' \subseteq enabled(s)$ and such that the possible outcomes of the corresponding distribution are restricted to $e \in E'$.
- $target: S \times E \times \mathbb{R}^+ \rightarrow S$ is a partial function describing state changes through events defined for tuples (s, e, d) such that $e \in enabled(s)$.

where $dist(A)$ denotes the set of distributions whose support is A .

Dynamics of a DESP. A configuration of a DESP consists of a triple $(s, \tau, sched)$ with s being the current state, $\tau \in \mathbb{R}^+$ the current time and $sched: E \rightarrow \mathbb{R}^+ \cup \{+\infty\}$ being the function that describes the occurrence time of each scheduled event ($+\infty$ if an event is not yet scheduled). The evolution (i.e simulation) of a DESP \mathcal{D} can be informally summarised in terms of an iterative procedure consisting of the following steps (assuming $(s, \tau, sched)$ is the current configuration of \mathcal{D}): 1) determine the set E_m of events enabled in state s and with minimal delay δ_m ; 2) select the next event to occur $e_{next} \in E_m$ by resolving conflicts (if any) between concurrent events through probabilistic choice according to $choice(s, E_m, \tau)$; 3) determine the new configuration of the process resulting from the occurrence of e_{next} , this in turns consists of three sub-steps: 3a) determine the new state resulting from occurrence of e_{next} , i.e. $s' = target(s, e_{next}, \delta_m)$; 3b) update the current time to account for the delay of occurrence of e_{next} , i.e. $\tau = \tau + \delta_m$; 3c) update the schedule of events according to the newly entered state s' (this implies setting the schedule of no longer enabled events to $+\infty$ as well as determining the schedule of newly enabled events by sampling through the corresponding distribution). The above procedure maps directly on GSPN models, in which case the set of states S corresponds to the set of possible markings of a GSPN, the events E correspond to the (timed) transitions of a GSPN, and the remaining elements (i.e. $delay$, $choice$ and $target$) are determined by the semantics of GSPN (i.e. the so-called *token game*).

Definition 2. A synchronised Linear Hybrid Automaton is a tuple $\mathcal{A} = \langle E, L, \Lambda, I, F, X, flow, \rightarrow \rangle$ where:

- E is a finite alphabet of events;
- L is a finite set of locations;
- $\Lambda: L \rightarrow Prop$ is a location labelling function;
- $I \subseteq L$ is the initial locations;
- $F \subseteq L$ is the final locations;
- $X = (x_1, \dots, x_n)$ is a n -tuple of data variables;
- $flow: L \mapsto Ind^n$ associates an n -tuple of indicators with each location (projection $flow_i$ denotes the flow of change of variable x_i).
- $\rightarrow \subseteq L \times ((Const \times 2^E) \uplus (lConst \times \{\#\})) \times Up \times L$ is the set of edges of the LHA

where \uplus denotes the disjoint union, Const and $|\text{Const}$ denotes the set of possible constraints, respectively left closed constraints, associated with \mathcal{A} (see description below) and Up is the set of possible updates for the variables of \mathcal{A} .

Furthermore \mathcal{A} fulfils the following conditions.

- **c1 (initial determinism):** $\forall l \neq l' \in I, \Lambda(l) \wedge \Lambda(l') \Leftrightarrow \text{false}$. This must hold whatever the interpretation of the indicators occurring in $\Lambda(l)$ and $\Lambda(l')$.
- **c2 (determinism on events):** $\forall E_1, E_2 \subseteq E : E_1 \cap E_2 \neq \emptyset, \forall l, l', l'' \in L$, if $l'' \xrightarrow{\gamma, E_1, U} l$ and $l'' \xrightarrow{\gamma', E_2, U'} l'$ are two distinct transitions, then either $\Lambda(l) \wedge \Lambda(l') \Leftrightarrow \text{false}$ or $\gamma \wedge \gamma' \Leftrightarrow \text{false}$. Again this equivalence must hold whatever the interpretation of the indicators occurring in $\Lambda(l)$, $\Lambda(l')$, γ and γ' .
- **c3 (determinism on \sharp):** $\forall l, l', l'' \in L$, if $l'' \xrightarrow{\gamma, \sharp, U} l$ and $l'' \xrightarrow{\gamma', \sharp, U'} l'$ are two distinct transitions, then either $\Lambda(l) \wedge \Lambda(l') \Leftrightarrow \text{false}$ or $\gamma \wedge \gamma' \Leftrightarrow \text{false}$.
- **c4 (no \sharp -labelled loops):** For all sequences $l_0 \xrightarrow{\gamma_0, E_0, U_0} l_1 \xrightarrow{\gamma_1, E_1, U_1} \dots \xrightarrow{\gamma_{n-1}, E_{n-1}, U_{n-1}} l_n$ such that $l_0 = l_n$, there exists $i \leq n$ such that $E_i \neq \sharp$.

Synchronisation of LHA and DESP. The role of a synchronised LHA \mathcal{A} is to select specific trajectories of a corresponding DESP \mathcal{D} while collecting relevant data (maintained in the LHA variables) along the execution. For the sake of brevity we omit the formal semantics of the product process $\mathcal{D} \times \mathcal{A}$ in this paper, but we provide an intuitive description of it.

A state of the $\mathcal{D} \times \mathcal{A}$ process is described as a triple (s, l, ν) where s is the current state of the DESP, l the current location of the LHA and $\nu : X \rightarrow \mathbb{R}$ the current valuation of the LHA variables. The synchronisation starts from the initial state (s, l, ν) , where s is an initial state of the DESP (i.e. $\pi_0(s) > 0$), l is an initial location of the LHA (i.e. $l \in I$) and the LHA variables are all initial set to zero (i.e. $\nu = 0$)⁵. Notice that, by initial determinism, for every $s \in S$ there is at most one $l \in I$ such that s satisfies $\Lambda(l)$. From the initial state the synchronisation process evolves through transitions where each transition corresponds to traversal of either a synchronised or an autonomous edge of the LHA (notice that because of the determinism constraints of the LHA edges at most only one autonomous or synchronised edge can ever be enabled in any location of the LHA. Furthermore if an autonomous and a synchronised edge are concurrently enabled the autonomous transition is taken first). If in the current location of the LHA (i.e. location l of the current state (s, l, ν) of process $\mathcal{D} \times \mathcal{A}$) there exists an enabled autonomous edge $l \xrightarrow{\gamma, \sharp, U} l'$, then that edge will be traversed leading to a new state (s, l', ν') where the DESP state (s) is unchanged whereas the new location l' and the new variables' valuation ν' might differ from l , respectively ν , as a consequence of the edge traversal. On the other hand if an event e (corresponding to transition $s \xrightarrow{e} s'$) triggered by process \mathcal{D} occurs in

⁵ Notice that because of the “initial-nondeterminism” of LHA there can be at most one initial state for the product process.

state (s, l, ν) , either an enabled synchronous edge $l \xrightarrow{\gamma, E', U} l'$ (with $e \in E'$) exists leading to new state (s', l', ν') of process $\mathcal{D} \times \mathcal{A}$ (from which the synchronised process will proceed) or the system goes to a dedicated rejecting state \perp and the synchronisation halts (indicating rejection of the trace).

HASL expressions. The second component of an HASL formula is an expression related to the automaton. Such an expression, denoted Z , is based on moments of a path random variable Y and defined by the grammar (2).

$$\begin{aligned}
Z &::= c \mid E[Y] \mid Z + Z \mid Z - Z \mid Z \times Z \mid Z/Z \\
Y &::= c \mid Y + Y \mid Y \times Y \mid Y/Y \mid \text{last}(y) \mid \text{min}(y) \mid \text{max}(y) \mid \text{int}(y) \mid \text{avg}(y) \\
y &::= c \mid x \mid y + y \mid y \times y \mid y/y
\end{aligned} \tag{2}$$

Z represents the expectation of an arithmetic expression based on LHA data variables and which uses path operators such as: $\text{last}(y)$ (i.e. the last value of y along a synchronising path), $\text{min}(y)$ ($\text{max}(y)$) the minimum (maximum), value of y along a synchronising path), $\text{int}(y)$ (i.e. the integral over time along a path) and $\text{avg}(y)$ (the average value of y along a path). In recent updates the COSMOS model checker [3] has been enriched with operators for assessing the Probability (Cumulative) Distribution Function (PDF/CDF) of the value that an expression Y takes at the end of a synchronising path. This requires specifying a discretised support of Y through the following syntax: $Z = \text{PDF}(Y, s, l, h)$ which means that the probability of Y to take value in any sub-interval of fixed width s corresponding to the partition of the considered $[l, h]$ support of Y is going to be evaluated (assuming that $[l, h]$ is discretised in $h - l/s$ sub-intervals).

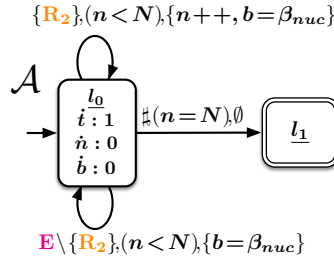


Fig. 4. Simple example of LHA that synchronises with the Wnt/ β -catenin GSPN model of Figure 1: the automaton selects paths containing N occurrences of reaction R_2 .

Example Figure 4 shows a simple example of LHA that synchronises with the Wnt/ β -catenin GSPN model of Figure 1. It uses three data variables: a clock t

(storing the simulation time), a counter n (counting the number of occurrences of reactions R_2) and a variable b which keeps track of the population of β_{nuc} . In the initial location l_0 the clock variable t grows with constant flow $\dot{t}=1$, whereas n and b flows is null. On occurrence of R_2 the top synchronising self-loop edge on l_0 is traversed hence n is incremented whereas on occurrence of any other reaction the bottom self-loop on l_0 is traversed, hence n is not updated. On the hand b is updated with the current value of β_{nuc} on occurrence of any reaction. As soon as N occurrences of R_2 have been observed the autonomous edge $l_0 \rightarrow l_1$ is traversed and synchronization halts (reaching of accepting location l_1). Below few examples of complete HASL formulae composed with the LHA of Figure 4.

- $\phi_1 \equiv (\mathcal{A}, E[last(t)])$: representing the average time for observing N occurrences of R_2 .
- $\phi_2 \equiv (\mathcal{A}, E[max(b)])$: representing the maximum population reached by β_{nuc} within the first N occurrences of R_2 .
- $\phi_3 \equiv (\mathcal{A}, PDF(last(t), 0.1, 0, 10))$: representing the PDF of the delay for observing N occurrences of R_2 (computed over the interval $[0, 10]$ with a discretisation step of 0.1)

4 Model analysis through HASL formulae

In order to analyse the dynamics of the Wnt/ β -catenin model we define a number of HASL formulae dedicated to capturing specific dynamical aspects of the GSPN model in Figure 1.

4.1 Measuring the maximal peaks of β_{nuc} resulting from an unsustained Wnt signal.

Both the *Wnt-basic* and *Wnt-inject* models are designed to study the behaviour of the Wnt/ β -catenin pathway in presence of an unsustained Wnt signal: i.e. a given amount of initial Wnt signal is present in the system but is steadily being consumed (reaction R_1) without being reintegrated (*Wnt-basic*) or being reintegrated once after a delay d_i (*Wnt-inject*). The effect of a non-reintegrated Wnt signal results is the production of a single peak of β_{nuc} (Figure 2) whereas a single, delayed, reintegration of 1000 Wnt molecules produces a second, shifted peak (Figure 3 left) in the population of β_{nuc} . We introduce some HASL formulae for formally measuring the time location and the amplitude of such β_{nuc} *transient* peaks. Observe that the analysis of a stochastic model through observation of a single simulated trajectory (as proposed in [15]) is in general little informative and even more so in this case as repeated trajectories of the wntb model exhibit a rather large variance. In the light a more formal approach is vital to obtain a meaningful analysis.

Automaton \mathcal{A}_{peaks} . The LHA in Figure 5 is conceived for locating the maximal and minimal peaks along an alternating trace of a given observed species, in

this case β_{nuc} . The automaton uses a number of data variables (Table 2) and is dependent on two configuration aspects: the setting of a parameter δ (the chosen noise level, see below) and the partition of the event set $E = E_{+\beta_{nuc}} \cup E_{-\beta_{nuc}} \cup E_{=\beta_{nuc}}$ where $E_{+\beta_{nuc}}$, $E_{-\beta_{nuc}}$ and $E_{=\beta_{nuc}}$ are the events yielding respectively: an increase of β_{nuc} , a decrease of β_{nuc} and having no effect on β_{nuc} population. Specifically, for model (1) we have: $E_{+\beta_{nuc}} = \{R_{10}\}$, $E_{-\beta_{nuc}} = \{R_{11}\}$ and $E_{=\beta_{nuc}} = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8, R_9, R_{12}\}$.

\mathcal{A}_{peaks} consists of an initial location **start**, a final location **end**, and 4 intermediate locations (**Min**, **Inc**, **Max** and **Dec**) where the actual analysis of the synchronised trajectory takes place. From **start** (on entering of which the initial amount of β_{nuc} is stored in x) the processing of the simulated trajectory leads either to **Min** or **Max** depending if the an increase (decrease) of β_{nuc} above (below) the chosen level of noise δ is observed (i.e. \mathcal{A}_{peaks} copes fine both in the case that the observed species initially increases or decreases).

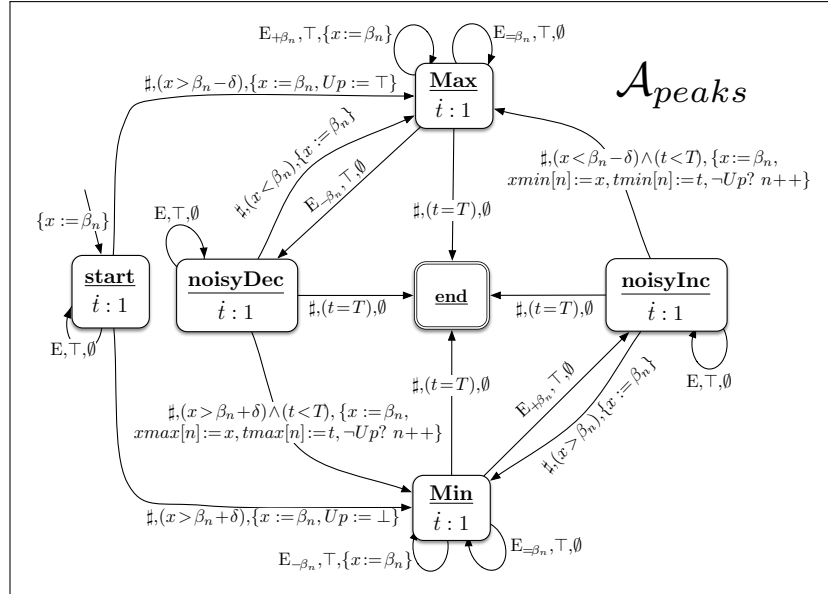


Fig. 5. $\mathcal{A}_{\beta_peaks}$: an LHA for locating the maximal peaks (up to noise level δ) of β_n .

Once in location **Min** (**Max**) the behaviour of the automaton depends on the type of observed event. If an event $e \in E_{+\beta_{nuc}}$ ($e \in E_{-\beta_{nuc}}$) is observed then location **noisyInc** (**noisyDec**) is entered indicating that β_{nuc} has increased (decreased) although the increase (decrease) has not (yet) exceeded δ (with respect to the most recent detected minimum (maximum) previously stored in x). On the other hand if while in **Min** (**Max**) an event $e \in E_{-A}$ ($e \in E_{+\beta_{nuc}}$) is observed, then this means that the current value of β_{nuc} went below (above) the previously detected minimum (maximum) hence x must be updated with the

newly found (potential) minimum (maximum) $x := \beta_{nuc}$. Finally an occurrence of any event $e \in E_{\beta_{nuc}}$ while in **Min** or **Max** is simply ignored. From **noisyInc** (**noisyDec**) the processing of input trace may lead back to **Min** (**Max**) if β_{nuc} re-decreases (re-increases) below (above) x (hence requiring an update $x := \beta_{nuc}$) or it may lead to **Max** (**Min**) as soon as β_{nuc} has increased (decreased) above (below) the noise level (i.e. $x > \beta_{nuc} - \delta$). The *autonomous* edge **noisyInc** \rightarrow **Max** (**noisyDec** \rightarrow **Min**) is traversed as soon as the value stored in x corresponds to an actual minimum (maximum) along the processed β_{nuc} trace (i.e. this is the case when the current value of β_{nuc} gets δ molecules far away from that stored in x). Hence when traversing **noisyInc** \rightarrow **Max** (**noisyDec** \rightarrow **Min**) we are sure that x contains a minimum (maximum) thus its value and its occurrence time are stored in the n^{th} element of the array $xmin[n] := x$ ($xmax[n] := x$), respectively $tmin[n] := t$ ($tmax[n] := t$). The processing terminates (entering of **End** from any other location) as soon as the simulation time is $t = T$ at which point all detected maxima and minima are stored in \mathcal{A}_{peaks} data variables.

Data variables		
name	domain	description
t	$\mathbb{R}_{\geq 0}$	time elapsed since beginning measure
n	\mathbb{N}	counter of detected local maxima/minima
up	bool	boolean flag indicating whether measuring started with detection of a max or min
x	\mathbb{N}	(overloaded) variable storing most recent detected maximum/minimum
$xmax(xmin)$	$\mathbb{N}^{\mathbb{N}}$	array of detected maxima (minima)
$tmax(tmin)$	$\mathbb{R}^{\mathbb{N}}$	array of detected occurrence time of maxima (minima)

Table 2. The data variables of automata \mathcal{A}_{peaks} of Figure 5 for locating the peaks of a noisy oscillatory traces

HASL formulae for measuring the effects of unsustained Wnt signal. Based on automaton \mathcal{A}_{peaks} we define the following HASL formulae:

- $\phi_{xmax} \equiv (\mathcal{A}_{peaks}, E[last(xmax[1])])$: the average value of the first β_{nuc} maximal peak.
- $\phi_{tmax} \equiv (\mathcal{A}_{peaks}, E[last(tmax[1])])$: the average value of the occurrence time of first β_{nuc} maximal peak.
- $\phi_{PDFmax} \equiv (\mathcal{A}_{peaks}, PDF(last(tmax[1]), 1, 30, 80))$: the PDF of the occurrence time of first β_{nuc} maximal peak (computed over the interval [30, 80] with a discretisation step 1)

Measuring the incidence of Wnt decay rate on β_{nuc} peaks. The decay speed of the Wnt signal affects the dynamics of β_{nuc} (the temporal location and height

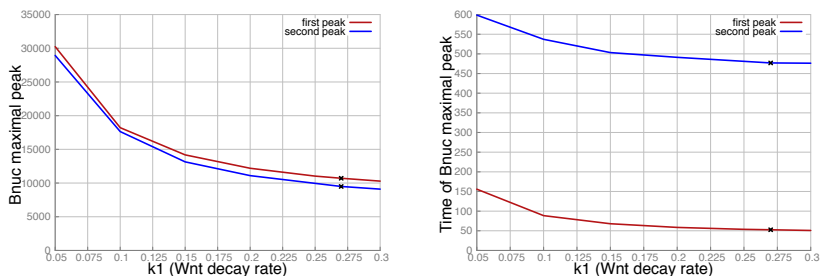


Fig. 6. Average value (left) and occurrence time (right) of the first and second β_{nuc} peak for the *Wnt-inject* model in function of Wnt decay rate (highlighted points correspond to k_1 original value as in Table 1 set B).

of β_{nuc} peaks). We performed a number of experiments aimed at addressing this aspect, specifically we assessed ϕ_{xmax} and ϕ_{tmax} against different instances of the Wnt/ β -catenin model with delayed Wnt re-injection (i.e. the *Wnt-inject* model) where each instance corresponds to a different value of k_1 (the decay rate Wnt). Figure 6 displays the results concerning the evaluation of the first and second peak of β_{nuc} . They indicate that both the average height (left) and the average occurrence time (right) of the first and second peaks of β_{nuc} decrease as the Wnt decay rate k_1 increases⁶. Figure 6 left also shows that the second peak of β_{nuc} (induced by Wnt re-injection) has, on average, a smaller amplitude than the first one and with a roughly constant difference of about 10% less between the two except for a $k_1 = 0.1$ for which the first and second peak's amplitude differs of about 5%.

Measuring the PDF of occurrence time of β_{nuc} peaks. Figure 7 displays the PDF of the first (left) and second (right) peak of β_{nuc} obtained by evaluation of ϕ_{PDFmax} against the *Wnt-inject* model (parameter set B). Both PDF curves exhibit a slight long-tail character with the majority of points being to the right of the maximum likely occurrence time.

5 Conclusion

We have presented a formal study of a stochastic model of the Wnt/ β -catenin pathway, a biological mechanism with a relevant role in controlling the life-cycle of neuronal embryonal cells. This model has been previously considered [15] however it was analysed only *informally*, i.e. through observation of simulated trajectories. By means of a powerful formalism (i.e. HASL model checking) we formally characterised and accurately assessed a number of relevant aspects

⁶ results computed with confidence level 99% and interval-width of 1% of the estimated measure.

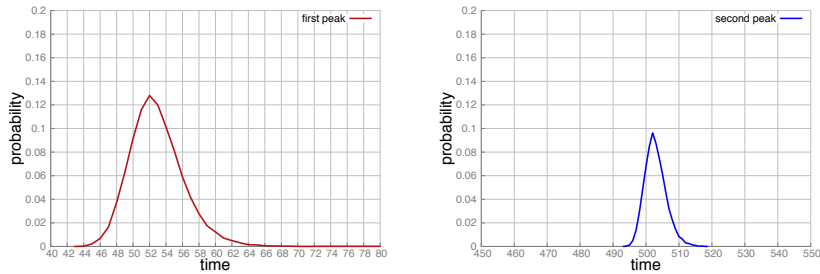


Fig. 7. The PDF of the time of occurrence of the first β_{nuc} peak (left) and second peak (right) for the *Wnt-inject* model (with re-injection of 1000 Wnt molecules at $t = 450$ minutes).

of the Wnt/ β -catenin dynamics. In particular in this work we have focused on studying the effects induced on nuclear β -catenin (a basic element of the Wnt/ β -catenin pathway) by the presence of a degrading (possibly reintegrated) Wnt signal. That included measuring of the average value and the PDF of the occurrence time and the amplitude of the β_{nuc} peaks resulting from a transitory Wnt signal. We plan to evolve this preliminary study in several directions, including the formal analysis of the effects induced by a sustained Wnt signal, as well as the analysis of the dynamics of β_{nuc} over a population of asynchronously evolving cells.

References

1. M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. John Wiley & Sons, 1995.
2. C. Baier, B. Haverkort, H. Hermanns, and J.-P. Katoen. Model-checking algorithms for CTMCs. *IEEE Trans. on Software Eng.*, 29(6), 2003.
3. P. Ballarini, H. Djafri, M. DufLOT, S. Haddad, and N. Pekergin. COSMOS: a statistical model checker for the hybrid automata stochastic logic. In *Proceedings of the 8th International Conference on Quantitative Evaluation of Systems (QEST'11)*, pages 143–144. IEEE Computer Society Press, sep. 2011.
4. P. Ballarini, H. Djafri, M. DufLOT, S. Haddad, and N. Pekergin. HASL: an expressive language for statistical verification of stochastic models. In *Proc. Valuetools*, 2011.
5. Paolo Ballarini, Jarno Mäkelä, and Andre S. Ribeiro. Expressive statistical model checking of genetic networks with delayed stochastic dynamics. In David Gilbert and Monika Heiner, editors, *CMSB*, volume 7605 of *Lecture Notes in Computer Science*, pages 29–48. Springer, 2012.
6. Benoit Boyer, Kevin Corre, Axel Legay, and Sean Sedwards. Plasma-lab: A flexible, distributable statistical model checking library. In Kaustubh R. Joshi, Markus Siegle, Mariëlle Stoelinga, and Pedro R. D’Argenio, editors, *QEST*, volume 8054 of *Lecture Notes in Computer Science*, pages 160–164. Springer, 2013.

7. T. Chen, T. Han, J.-P. Katoen, and A. Mereacre. Quantitative model checking of CTMC against timed automata specifications. In *Proc. LICS'09*, 2009.
8. Alexandre David, Kim Guldstrand Larsen, Axel Legay, Marius Mikucionis, Danny Bøgsted Poulsen, and Sean Sedwards. Runtime verification of biological systems. In *ISoLA (1)*, pages 388–404, 2012.
9. S. Donatelli, S. Haddad, and J. Sproston. Model checking timed and stochastic properties with CSL^{TA} . *IEEE Trans. on Software Eng.*, 35, 2009.
10. J. Heath, M. Kwiatkowska, G. Norman, D. Parker, and O. Tymchyshyn. Probabilistic model checking of complex biological pathways. *Theoretical Computer Science*, 319(3):239–257, 2008.
11. H. Kitano. *Foundations of Systems Biology*. MIT Press, 2002.
12. M. Kwiatkowska, G. Norman, and D. Parker. Stochastic model checking. In *Formal Methods for the Design of Computer, Communication and Software Systems: Performance Evaluation*, volume 4486 of *LNCS*, pages 220–270. Springer, 2007.
13. Axel Legay and Benoît Delahaye. Statistical model checking : An overview. *CoRR*, abs/1005.1327, 2010.
14. Orianne Mazemondet, Rayk Hubner, Jana Frahm, Dirk Koczan, Benjamin M Bader, Dieter G Weiss, Adelinde M Uhrmacher, Moritz J Frech, Arndt Rolfs, and Jiankai Luo. Quantitative and kinetic profile of wnt/ β -catenin signaling components during human neural progenitor cell differentiation. *Cell Mol Biol Lett*, 2011.
15. Orianne Mazemondet, Mathias John, Stefan Leye, Arndt Rolfs, and Adelinde M Uhrmacher. Elucidating the sources of β -catenin dynamics in human neural progenitor cells. *PLOS-One*, 7(8):1–12, 2012.
16. K. Sen, M. Viswanathan, and G. Agha. VESTA: A statistical model-checker and analyzer for probabilistic systems. In *Proc. QEST'05*, 2005.
17. Håkan LS Younes. Ymer: A statistical model checker. In *Computer Aided Verification*, pages 429–433. Springer, 2005.