Fuzzy and Hybrid Modelling of the Caulobacter Crescentus Cell Cycle

George Assaf[a], Fei Liu[b], Monika Heiner[a], Mostafa Herajy[c]

[a]*Brandenburg Technical University (BTU), Cottbus, Germany*
[b]*South China University of Technology, Guangzhou, China*
[c]*Department of Mathematics and Computer Science, Faculty of Science, Port Said University, 42521 - Port Said, Egypt*

Abstract

Studying and analysing the various phases and key proteins of cell cycles is essential for the understanding of cell development and differentiation. To this end, mechanistic models play an important role towards a system level understanding of the interaction between cell cycle components. Many quantitative models of cell cycles have been previously constructed using either stochastic or deterministic approaches. However, cell cycle models are inherently hybrid requiring the full and accurate interplay of the continuous system dynamics and their corresponding discrete events. Moreover, not all required experimental data are usually available when designing in-silico experiments for these scenarios. In this paper, we employ hybrid Petri nets to implement a hybrid model of the Caulobacter crescentus cell cycle. The model handles all required logics of cell cycles in a very elegant way. We then extend this model to support fuzzy kinetics for those parts where sufficient experimental data are not available and thus precise kinetic parameters cannot be estimated. With some of the kinetic parameters being set as fuzzy numbers, the model produces uncertain bands of outputs reflecting different possibilities of an output comprising most likely the correct one.

Email addresses: `george.assaf@b-tu.de` (George Assaf), `feiliu@scut.edu.cn` (Fei Liu), `monika.heiner@b-tu.de` (Monika Heiner), `mherajy@sci.psu.edu.eg` (Mostafa Herajy)
URL: `http://www-dssz.informatik.tu-cottbus.de` (George Assaf)
Corresponding author: feiliu@scut.edu.cn

Keywords:
Hybrid Modelling, Hybrid Petri Nets, Fuzzy Hybrid Petri Nets, Caulobacter Crescentus Cell Cycle, Fuzzy Logic, Systems Biology

1. Introduction

With the advance of modelling techniques of systems biology, hybrid modelling [1, 2, 3, 4], which combines paradigms with different dynamics and semantics, is becoming ubiquitous. The reasons behind the increasing interest in hybrid modelling can be attributed to the increasing number of biological models, where different time scales coexist in the same reaction network and the lack of enough experimental data required to fit all kinetic model parameters [1]. Among the widely studied biological systems, where hybrid modelling plays a role, are the cell cycle dynamics of different organisms [5, 6, 3, 7, 8].

On the one hand, the traditionally preferred approach to construct cell cycle models is the deterministic modelling, where a system of ordinary differential equations (*ODEs*) is constructed by applying one of the well-known kinetic rate laws (e.g. mass action) to reactions of the key proteins as well as their genes and mRNAs [9]. However, stochastic simulation offers a better tool to capture the variability in cell volumes [10] by firing one reaction at a time. Moreover, cell cycle models usually require to trigger certain discrete events during the course of the deterministic simulation (e.g. cell division) and certain processes are naturally better represented continuously (e.g. cell growth) [5]. Thus, it is highly required to implement cell cycle models using a hybrid approach combining both discrete and continuous semantics.

On the other hand, not all kinetic parameters are usually known during model construction and some parts of the network are usually approximated to get a model version of lower resolution with the currently available data. Eventually, a refined (extended) version of the model may be developed when more data are available. For instance, in [11], a computational model of *Caulobacter crescentus stalked cells* was constructed to understand the role of key proteins including

CtrA. However, due to the lack of experimental data, the subnetwork responsible for the proteolysis of CtrA has not been explicitly modelled. In a later version of the model, the reactions responsible for proteolysis of CtrA have been added [12] resulting in a model that can be quantitatively compared with experimental data. To this end, a tool that can efficiently deal with missing or vague model parameters is highly required, whereby it enables modellers to obtain possible results despite missing the exact data [13].

Hybrid Petri nets [14] are excellent tools to model the interplay between model components following discrete or continuous semantics. They have been widely applied to implement many biological systems (e.g. see [4, 5, 15, 3, 16, 17, 18, 19, 20, 21, 22, 23]). As a special class of Petri nets, they provide a graphical and convenient approach to represent and simulate reaction networks that require hybrid (discrete/continuous) semantics. Moreover, hybrid Petri nets can also model the logic of deterministic/stochastic switching by permitting events to occur at random times [4, 16]. Hybrid Petri nets provide a graphical tool to construct and execute (simulate) deterministic/stochastic models and/or deterministic models with a few discrete events. In the former case, reactions are partitioned into deterministic or stochastic reactions and simulated via continuous or stochastic transitions (see e.g. [4, 16]), while in the latter transitions with immediate firing can be employed to simulate instant event firings (e.g. in [21, 22]).

Furthermore, to overcome the lack of kinetic parameters, many approaches have been employed in the literature (see [24] for a few examples). Recently, e.g. in [13, 25, 26, 27], fuzzy logic has been proposed to model biological systems where reactions with unknown/missing kinetic parameters are modelled using fuzzy logic, while the others with known or precise kinetics are represented by *ODEs*, resulting in certain/uncertain hybrid models [25]. In [1], Petri nets that support fuzzy logic have been classified as either basic fuzzy Petri nets, fuzzy quantitative Petri nets, or Petri nets with fuzzy kinetic parameters. Stochastic Petri nets with fuzzy kinetic parameters have been presented in [13] and continuous fuzzy Petri nets in [25].

In this paper, we propose a fuzzy hybrid model of the *Caulobacter crescentus cell cycle* using fuzzy hybrid Petri nets (hybrid Petri nets with fuzzy kinetic parameters). We first implement the *ODE* model of Xu et al. [12] using hybrid Petri nets and then add fuzzy components to the model representing part of the missing semantics. The proposed model exhibits hybrid discrete/continuous semantics as well as hybrid crisp/fuzzy semantics. The model can easily be extended in the future to guide other experiments by adding fuzzy transitions to reactions with unknown kinetic rates.

This paper is organised as follows: in Section 2, we provide some background about hybrid Petri nets, fuzzy logic as well as the dynamics and biological basics of the *Caulobacter crescentus cell cycle*. Afterwards, we give the formal definition and semantics of fuzzy hybrid Petri nets. The hybrid implementation of *Caulobacter crescentus cell cycle* is presented in Section 4, and the fuzzy hybrid model in Section 5. The paper concludes with closing remarks and an outline of future work.

2. Preliminaries

2.1. Hybrid Petri Nets

Hybrid Petri nets ($\mathcal{HPN}$) [14] are extensions of standard place/transition Petri nets [28]. The main advantage of $\mathcal{HPN}$ is that they can easily represent and execute systems where the interplay between discrete and continuous states is highly required to implement the system logic. Many extensions of hybrid Petri nets have been proposed in the literature to accommodate the special requirements of systems modelling [23, 4]. However, all these extensions offer discrete and continuous places as well as discrete and continuous transitions. Discrete places may be used to represent discrete system states, while continuous places could represent fluid quantities of the system that cannot be intuitively handled by just using discrete places (e.g. the amount of water in a tank). Moreover, discrete transitions fire instantaneously at discrete time points, while

continuous transitions fire continuously and could represent a rate of change of continuous system states.

Matsuno et al. [23, 21, 22] have widely employed hybrid Petri nets to model and simulate biological systems by proposing a flexible class of hybrid Petri nets called hybrid functional Petri nets (HFPN). HFPN have been successfully applied afterwards to construct many biological pathways (e.g. see [21, 22, 23]). In [4], we have introduced a special class of hybrid Petri nets ($\mathcal{GHPN}$) that can capture the interplay of stochastic and continuous transitions facilitating the construction and execution of hybrid biological models. Furthermore, a rich set of transitions and arcs has also been included. Some example models are presented in [4, 5, 15, 3, 16, 18, 19, 20]. In [16], we extended $\mathcal{HPN}$ to support colours and thus allowing the construction of configurable models. Table 1 summaries the different nodes and arcs of the hybrid Petri net class presented in [4].

As an example of how $\mathcal{HPN}$ can be adopted to model biological systems, consider the simple $\mathcal{HPN}$ in Figure 1 which represents DNA replication in a cell cycle model. In this network, continuous transitions represent continuous processes such as synthesis, degradation and elongation (i.e., the transitions *syn*, *elongation*, *deg_gene*1, *deg_gene*2, and *deg_gene*3). They fire continuously and infinitely; however their firing rates might be conditioned by discrete places (e.g. the transition *elongation* fires with full rate only, when the place *start_elongation* has at least one token, and it fires with zero rate, when this place is empty). Immediate transitions (i.e. the transitions: *initiation*, *termination*, *elongation_phase*1, *elongation_phase*2, and *elongation_phase*3) fire instantaneously as soon as they get enabled. They are used in this model to represent events that are triggered occasionally.

At the beginning of the execution, the system starts off with an empty marking of the place *init*. The continuous transition *syn* keeps increasing the marking of *init*. When *init* has a value equal to or greater than 0.1 (compare the read arc connecting this transition with *init*), the transition *initiation* is triggered. It adds a token to *start_elongation* and resets *init* to zero. This in turn sets the transition *elongation* at full rate. The transition *elongation*

increases the marking of *elong* continuously, and depending on the level of *elong* one or more phases of DNA elongation could take place (based on the corresponding values of the read arcs). When the value of *elong* reaches one, the immediate transition *termination* will fire and reset *elong* as well as other model places. Inhibitor arcs are employed here to prevent the repetitive firing of immediate transitions.

Figure 2 provides the simulation output of the $\mathcal{HPN}$ in Figure 1. For demonstration purposes, we used hypothetical parameter values to execute the model. However, we will show later in Section 4, how this $\mathcal{HPN}$ can be used as a subnetwork of cell cycle models parametrized by kinetic rate laws and experimentally fitted parameter values.

2.2. Fuzzy Logic

Fuzzy logic [29] has been introduced to deal with data uncertainty in systems, e.g. biological systems [30, 31, 32, 27]. The core concept of fuzzy logic is the *fuzzy set*. A *fuzzy set* is defined over a universal set $\mathbb{X}$ by its membership function μ which only takes values between the closed interval $[0, 1]$. Thus, it specifies a membership degree for each element belonging to the universal set. Contrary, in crisp sets, the membership function takes only two values $\{0, 1\}$. Compare Figure 3 for an illustrating example.

A fuzzy number is a special case of a convex, normalised fuzzy set whose universal set is given by the set of real numbers. Fuzzy numbers come in different types (shapes), which means that data uncertainty can be expressed in different ways. For example, triangular fuzzy numbers can be determined using three points, i.e.$\tilde{\xi} = (a, b, c)$, $a \leq b \leq c$. Equation 1 gives the membership function of a triangular fuzzy number. Each point could be interpreted differently to describe the uncertainty range of the given data. Another essential term that is associated with fuzzy sets is the α-cut [29]. An α-cut of a fuzzy set at a given membership degree consists of a crisp subset (of the universal set) in which each element has a membership degree greater than or equal to the given α level.

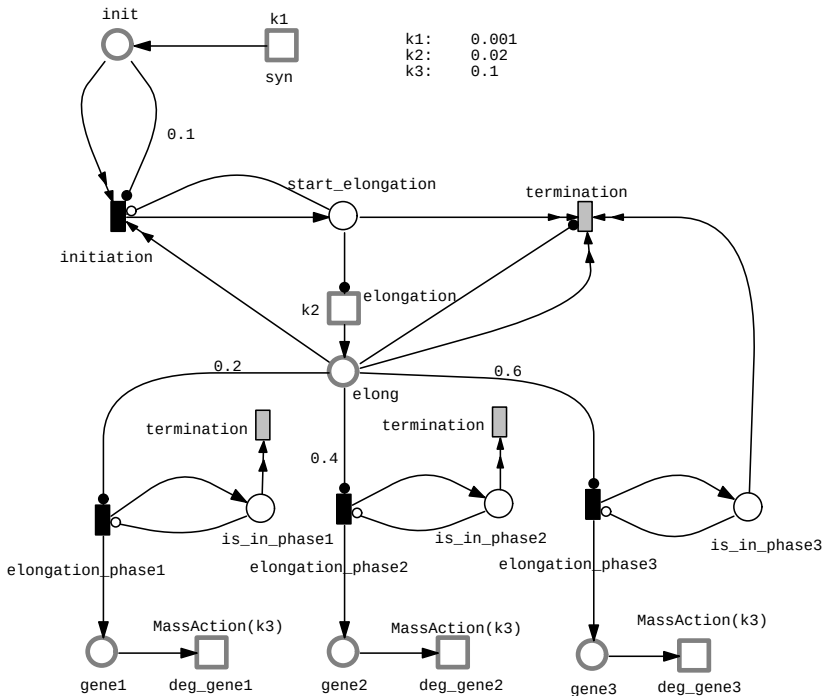
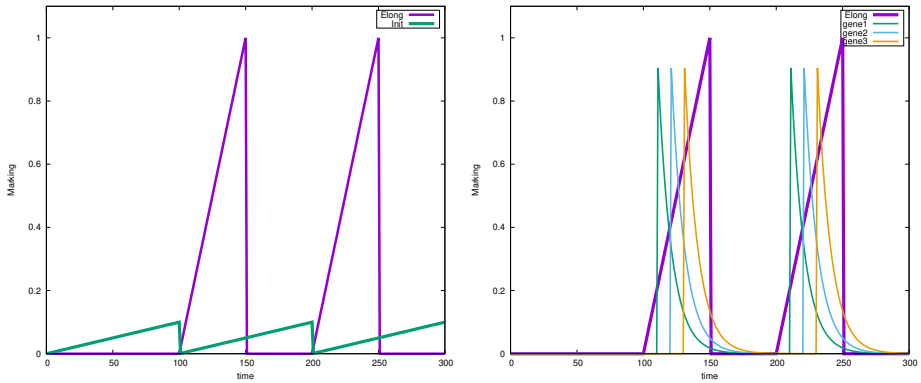


Figure 1: An example of a simple hybrid Petri net representing different events of DNA replication of a cell cycle. The graphical meaning of each node is given in Table 1. The continuous place *init* is increased by a constant rate k_1 via the continuous transition *syn*. When reaching the value of 0.1 (the weight of the read arc), the immediate transition *initiation* triggers the start of DNA elongation by putting a token to the discrete place *start_elongation*. The continuous place *elong* is increased by a constant rate via the continuous transition *elongation*. Depending on the value of the place *elong*, different phases of the elongation are triggered, each is represented by an immediate transition and activates one of the three genes. Eventually, the elongation process is terminated by the firing of the immediate transition *termination* when the value of *elong* reaches one. The simulation dynamics of this $\mathcal{HPN}$ is illustrated in Figure 2. The transition *termination* is a logical transition, resetting upon firing the following places to zero: *start_elongation, elong, is_in_phase1, and is_in_phase 2, is_in_phase3*

$$\mu_{\tilde{\xi}}(x) = \begin{cases} 0 & \text{if } a > x, \\ \frac{x-a}{b-a} & \text{if } a < x \leq b, \\ \frac{c-x}{c-b} & \text{if } b < x \leq c, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$



(a) (b)

Figure 2: Simulation output of the $\mathcal{HPN}$ in Figure 2. (a) The marking of place *init* versus *elong*, and (b) The marking of *elong* versus the places *gene*1, *gene*2, and *gene*3. In (a) the synthesis of *elong* is triggered only when *elong* reaches 0.1. In (b), the marking of *gene*1, *gene*2, and *gene*3 get pulses only when *elong* reaches the corresponding threshold value and then they start to degrade.

Fuzzy analysis techniques are useful for performing uncertainty analysis of a system whose input is uncertain due to, e.g. lack of exact knowledge of some parts of the system. For biological models, kinetic data uncertainty is a common issue. This type of uncertainty usually stems from imprecise knowledge or insufficient understanding of the studied biological system [33]. In such a case, stochastic methods maybe not be of help, thus fuzzy approaches could be excellent tools to tackle this issue. Fuzzy analysis (simulation) produces two kinds of measurements: fuzzy bands and timed membership functions of each output variable of interest. First, an output fuzzy band comprises a set of traces expressing output variability according to input variability. Crucially, an output fuzzy band represents something like the confidence interval in the statistics as the true trace most probably lies in this band. Second, timed membership functions help to extract the most possible truth, where the maximum membership degree appears, i.e. the value corresponding to $\alpha = 1$.

It is worth mentioning that fuzzy Petri nets come with many flavours. For instance, basic fuzzy Petri nets [34, 35] are a special kind of $\mathcal{FPN}$ combining fuzzy

Table 1: Summary of $\mathcal{HPN}$ elements, adopted from [4] and [3].

Class	Element	Graphics	Purpose
Places	Continuous		Represent a continuous variable of the ODE model, or biochemical species.
	Discrete		Denote the current state of an operation, mRNA, or protein.
Transitions	Continuous		Fire continuously over time and hold the rate of change related to two or more continuous places. Represent reactions.
	Stochastic		Fire stochastically in a random manner. Also represent reactions, but not used in this paper.
	Deterministic		Fire after deterministic time delays. Represent some logic of the $\mathcal{HPN}$ model. Not used in this paper.
	Immediate		Fire immediately as soon as they get enabled. Used to represent discrete model events.
	Scheduled		Fire at a specified absolute time point. Not used in this paper
Arcs	Standard		implement the flow of marking between transitions and places and vice versa.
	Read		Implement conditions for a transition to be enabled, but without consuming any marking during the firing.
	Inhibitor		A transition having a preplace connected by an inhibitor arc can only fire if the place marking is less than the arc weight. Used to condition the repeated firing of immediate transitions
	Equal		A transition having a preplace connected by an equal arc can only fire if the place marking is exactly equal to the arc weight. Not used in this paper.
	Modifier		Permit the use of a place in transition rate, but without affecting the enabling of a transition.
	Reset		Reset the corresponding place marking to zero. Used to reset some of the model conditions in response to a discrete event.

logic with standard Petri nets (no time notion) to represent a set of fuzzy rules in the form of the *If-then* and *if-then-else* statement which are useful to address the uncertainty of the model structure, when the system under study suffers from insufficient prior knowledge or measurement data to capture the accurate structure by making use of an engine called fuzzy inference system [36]. Moreover, stochastic [13] and continuous [25] with fuzzy kinetic parameters are special types of fuzzy Petri nets which are useful for modelling and analysing systems that are characterised by their imprecise knowledge of the kinetic rates. Stochastic

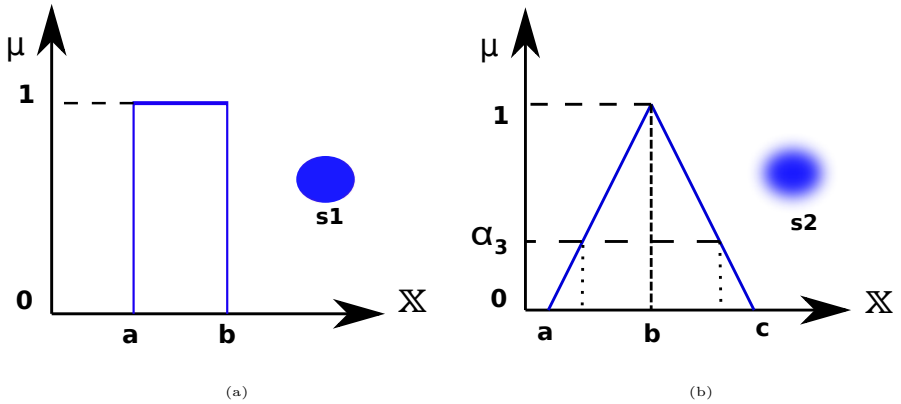


Figure 3: Explanatory examples showing the difference between crisp and fuzzy sets. (a) A crisp kinetic parameter; the species s1 is produced by a reaction whose kinetic parameter is specified by either the value a or the value b. (b) A fuzzy kinetic parameter (fuzzy set); the species s2 is produced by a reaction whose kinetic parameter is specified by an uncertain range of values represented as a triangular fuzzy number which is determined by three real numbers. One α-cut is shown at the level α=3.

Petri nets with fuzzy kinetic parameters comprise basically discrete places and stochastic transitions, for which fuzzy numbers as fuzzy kinetic parameters can be assigned in rate functions. In contrast, continuous Petri nets with fuzzy kinetic parameters comprise continuous places and continuous transitions, for which fuzzy kinetic parameters can be assigned in rate functions. In this research, we make use of fuzzy hybrid nets to capture the uncertainty accompanied with kinetic parameters which are associated with some rate functions (continuous and stochastic ones) of biochemical reaction networks. In Section 3, we sketch the formal definition together with the semantic of fuzzy hybrid Petri nets as they are implemented in our modelling and simulation tool *Snoopy* [37].

2.3. Caulobacter Crescentus Cell Cycle

In many living organisms, a cell is divided into two daughter cells via a process called cell cycle division [11, 12]. Newborn cells may repeat again the same cycles. During a single cycle, there are a number of phases where DNA is synthesised and replicated until eventually the cell is ready for division. In a

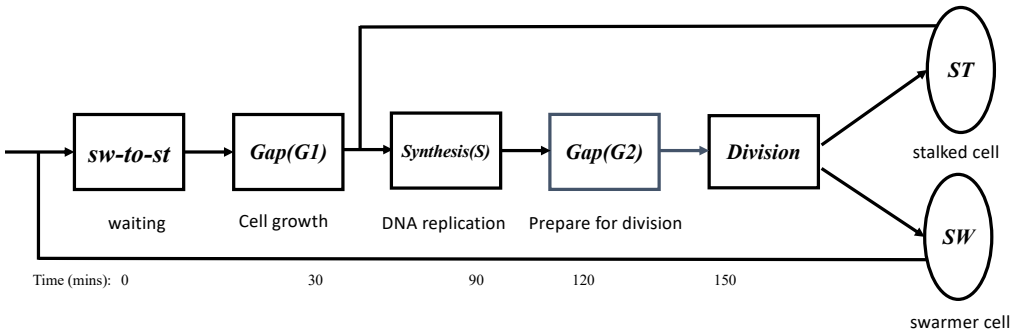


Figure 4: A schematic diagram illustrating the different phases of one cycle in C. crescentus cell cycles. A cell passes through G1, S, and G2 before it divides into two different cells: stalked and swarmer cells. A stalked cell can directly repeat the cycle and locates itself at the S phase, but a swarmer cell requires sometime to grow and enter G1.

typical cell cycle, the four phases are the gap (G1) phase, where a cell is prepared for growth the synthesis (S) phase, where the DNA is replicated, the gap (G2) phase, where the cell is prepared for division, and the mitosis phase M, where the cell is divided into two cells [10].

Caulobacter crescentus (C. crescentus) is an alphaproteobacterium that undergoes robust cell division cycles [12]. Thus they may be a perfect model organism to study the mechanisms and regularity proteins controling different phases of cell cycles in living organisms. They widely occur in fresh water, lakes and streams. C. crescentus undergoes a special cell division cycle where it produces two distinct progenies: stalked and swarmer daughter cells [11]. The stalked cell is directly ready to initiate the new cycle, while the swarmer daughter takes some time to differentiate into a stalked cell before it can repeat the cycle. In C. crescentus the three phases are: gap (G1), synthesis (S), and gap (G2). The time spent at each phase depends on the type of the cell. Figure 4 provides a schematic diagram of the different phases in the C. crescentus cell cycle.

Furthermore, the progression of the C. crescentus cell cycle from one phase to another is controlled by a cell cycle dependent regularity network. To study the temporal dynamics of the key proteins, a cell cycle quantitative model of crescentus has been developed, in [11] which includes the dynamics of key

proteins, e.g. DnaA, GcrA, CtrA, CcrM and SciP. This model has been further extended in [12] to include the subnetwork responsible for proteolysis of CtrA. The model consists of three submodules: the core regulatory network module, the cell cycle-dependent proteolysis of CtrA module and the chromosome replication and methylation module.

All reactions of each module can be classified as either synthesis, degradation, or phosphorylation. The key protein module describes the regularity feedback loops between key proteins. For instance, DnaA activates gcrA, while GcrA regulates ctrA and dnaA. In turn CtrA influences the transcription of dnaA. Please note that throughout of this paper, variables starting with capital letters (e.g. CtrA) represent proteins, while variables starting with small letters denote the corresponding mRNAs of these proteins (e.g. ctrA).

Moreover, a negative feedback loop between DnaA and GcrA exists where DnaA promotes the expression of gcrA, and GcrA inhibits DnaA. CtrA exists in two versions: active (CtrAP) and inactive (CtrA). Active CtrA starts the initiation of Dna replication. Additionally, the phosphorylation as well as the degradation of CtrA is driven by the CckA-dependent pathway. CtrA proteolysis depends on a protease complex, consisting of the protease ClpXP and four other additional adaptors: CpdR, RcdA, PopA, and cdG. The ClpXP level is constant during all the phases of the cell cycle, however it binds with other adaptors forming different complexes. The subnetwork responsible for reactions related to the ClpXP complexes and the four adaptors are forming the second module of the model [12].

The third module of the Xu et al. model is concerned with chromosome replication and methylation. In this module a set of events are fired depending on the concentration of the key proteins. For example, during the transition from the swarmer to the stalked phase, C. crescentus requires high levels of DnaA and low levels of CtrA to initiate DNA replication [12]. Therefore, an event is scheduled to take place, when DnaA become high and CtrA is low in the quantitative model.

The exact details of the different modules as well as the reaction network can

be found in [11, 12]. In this paper, we build a hybrid model based on the ODE one presented in [12]. The model graphically represents all three modules and introduces a further, discrete module to represent the timing events that take place during the model execution.

3. Fuzzy Hybrid Petri nets

Fuzzy hybrid Petri nets ($\mathcal{FHPN}$) are a class of quantitative Petri nets with fuzzy kinetic parameters [37, 31] extending standard hybrid Petri nets by fuzzy kinetic parameters. In this class of $\mathcal{HPN}$, the rate function of a stochastic/continuous transition may enjoy either a crisp kinetic parameter (when the value of the kinetic parameter is precisely known) or a fuzzy kinetic parameter represented as a triangular fuzzy number, when the value of that parameter is not precisely given.

In this section, we present the formal definition and the semantics of $\mathcal{FHPN}$ based on the definition of $\mathcal{HPN}$ previously introduced in [4]. For the graphical representation and typical use cases of each node type, please refer to Table 1.

3.1. Formal Definition

Fuzzy hybrid Petri nets are formally defined by the following tuple $\mathcal{FHPN} = \langle P, T, A, F, V, m_0 \rangle$, where :

- $P = P_{disc} \cup P_{cont}$ is the set of places, with P_{disc} is the set of discrete places and P_{cont} is the set of continuous places.
- $T = T_s \cup T_i \cup T_d \cup T_{sched} \cup T_{cont}$ is the set of transitions, with T_s is the set of stochastic transitions, T_i is the set of immediate transitions, T_d is the set of deterministically delayed transitions, T_{sched} is the set of scheduled transitions, T_{cont} is the set of continuous transitions, and $T^D = T_s \cup T_i \cup T_d \cup T_{sched}$ denotes the set of discrete transitions.
- $P \cap T = \emptyset$.

- $A = A_{disc} \cup A_{cont} \cup A_I \cup A_T \cup A_E \cup A_R \cup A_M$ is the set of directed arcs with $A_{disc} \subseteq (P \times T^D) \cup (T^D \times P)$ defines the set of discrete arcs, $A_{cont} \subseteq (P_{cont} \times T_{cont}) \cup (T_{cont} \times P_{cont})$ defines the set of continuous arcs, $A_T \subseteq (P \times T)$ defines the set of read arcs, $A_I \subseteq (P \times T)$ defines the set of inhibitor arcs, $A_E \subseteq (P_{disc} \times T)$ defines the set of equal arcs, $A_R \subseteq (P \times T^D)$ defines the set of reset arcs, $A_M \subseteq (P \times T)$ defines the set of modifier arcs.

- F is a function

$$F : \begin{cases} A_{cont} \rightarrow, \mathbb{R}^+ \\ A_{disc} \rightarrow \mathbb{N}, \\ A_T \rightarrow \mathbb{R}^+, \\ A_I \rightarrow \mathbb{R}_0^+, \\ A_E \rightarrow \mathbb{N}, \\ A_R \rightarrow \{1\}, \\ A_M \rightarrow \{1\}. \end{cases}$$

  which assigns a positive integer value or a positive rational value as a weight to each arc depending on the arc type. If an arc is not explicitly weighted, we assume a weight of 1.

- V is a set of functions $V = \{g, w, d, f\}$ where :

  1. $g : T_s \rightarrow H_s$ is a function which assigns a stochastic hazard function h_{s_t} to each transition $t_j \in T_s$, whereby $H = \{h_t | h_t : \Gamma^{|\bullet t|} \rightarrow \Gamma, t \in T_s\}$ which means that a kinetic parameter is described by either a fuzzy number or a real (crisp) number in Γ.
  2. $w : T_i \rightarrow H_w$ is a function which assigns a weight function h_w to each immediate transition $t_j \in T_i$, such that $H_w = \{h_{w_t} | h_{w_t} : \mathbb{R}_0^{|\bullet t_j|} \rightarrow \mathbb{R}_0^+, t_j \in T_i\}$ is the set of all weight functions, and $w(t_j) = h_{w_t}, \forall t_j \in T_i$.

3. $d : T_d \cup T_{sched} \rightarrow \mathbb{R}_0^+$, is a function which assigns constant time to each deterministically delayed and three real values to each scheduled transition representing the beginning of the firing interval, the repetition value, and the end of the firing interval; respectively.
4. $f : T_{cont} \rightarrow H_c$ is a function which assigns a rate function h_c to each continuous transition $t_j \in T_{cont}$, whereby $H = \{h_t | h_t : \Gamma^{|\bullet t|} \rightarrow \Gamma, t \in T_{cont}\}$ which means that a kinetic parameter is described by either a fuzzy number or a real (crisp) number in Γ.

Please note that the notation Γ denotes the set of all fuzzy numbers including real numbers in $\mathbb{R}_0^+$, as a triangular fuzzy number can be seen as a real number by assigning to its points the same value, i.e. (a=b=c). The notations $^\bullet t$ and $t^\bullet$ denote the set of pre-places and the set of post-places of the transition t, respectively.

- $m_0 = m_{cont} \cup m_{disc}$ is the initial marking for both the continuous and discrete places, whereby $m_{cont} \in \mathbb{R}^{+}{}_0^{|P_{cont}|}$, $m_{disc} \in \mathbb{N}_0^{|P_{disc}|}$.

Here the notation $\mathbb{N}$ denotes the set of positive integer numbers, while the notations $\mathbb{R}^+$ and $\mathbb{R}_0^+$ denote the sets of non-negative real numbers and non-negative real numbers including zero, respectively.

3.2. Semantics

Fuzzification is combined with hybrid Petri nets by following the Zadeh's extension principle [29], according to which a fuzzy kinetic parameter can be represented as a union of its α-cut, typically spread over the continuous interval $[0, 1]$. Each α-cut comprises a number (specified by the user) of crisp values (or samples). When an $\mathcal{FHPN}$ model contains fuzzy kinetic parameters, a sampling strategy has to be utilised for the purpose of performing sampling over all the involved fuzzy kinetic parameters. Then, hybrid simulation has to be performed on each sample/sample combination. Please note that the sampling strategy has to be chosen carefully, as it determines the number of required samples and

Algorithm 1 Fuzzy Hybrid Simulation

Input: $\mathcal{FHPN}$ model with K fuzzy kinetic parameters, J number of alpha levels (α-cuts) and number of samples per each level; Sampling strategy.

Output: Fuzzy bands of all variables together with their membership functions over time.

```
for each α level α_j, j= 0, 1 , .. J − 1 do
    for each fuzzy kinetic parameter k do
        compute α-cut set using j and k;
        perform sampling using the chosen sampling strategy;
    end for
    for each sample combination S among all fuzzy kinetic parameters do
        perform hybrid simulation using Algorithm 2;
    end for
end for
for each place p_i, i= 0, 1 , .. M do
    construct fuzzy band using all hybrid simulation traces of p_i;
    compose membership functions of p_i at each time point;
end for
```

thus the total number of simulations that have to be performed [31]. As a result, a (fuzzy) band of simulation traces is constructed for each output of interest. These fuzzy bands illustrate how the inputs of uncertain kinetic parameters affect the outputs. Moreover, membership functions over time (which we call timed membership functions) are reconstructed which precisely describe the associated uncertainties at a certain time point. For more insight on how fuzzy simulation algorithms work, please consult, e.g. [32, 27]. Algorithm 1 sketches the required steps for fuzzy hybrid simulation.

To construct the membership function and the fuzzy band, Algorithm 1 requires the simulation of the $\mathcal{HPN}$ at each sampling point. To simplify the discussion, in this paper we consider $\mathcal{HPN}$ models which contain continuous and immediate transitions only as required by the C. crescentus cell cycle. For the

general case, when a model contains stochastic as well as continuous transitions, the semantics is given in details in [4].

The semantics of continuous transitions and continuous places are given by a system of ODEs, see formula (2) [38].

$$\frac{dp_i}{dt} = \sum_{\forall t_j \in {}^{\bullet}p_i} F(t_j, P_i) \cdot v_j - \sum_{\forall t_j \in p_i^{\bullet}} F(p_i, t_j) \cdot v_j \tag{2}$$

where $F(t_j, P_i)$ is the arc weight connecting the transition t_j with the place p_i, likewise $F(p_i, t_j)$, V_j is the maximal firing rate of the transition t_j, and ${}^{\bullet}p_i$, and $p_i^{\bullet}$ denote the pre- and post-transition of the place p_i, respectively.

In this version of $\mathcal{HPN}$, continuous transitions are considered to be enabled all the time; however, their firing rates can change depending on the transition's pre-places. Since we consider continuous transitions where transition rate functions follow certain biological patterns, negative marking of pre-places is not possible [38]. For instance, when all the pre-places of a continuous transition have positive marking, the corresponding transition will fire with a rate greater than zero. However, when any of its pre-places have zero values, the firing rate will be zero, too. The main advantage of this semantics is that we need to consider just one system of ODEs during the whole simulation of the $\mathcal{HPN}$ model which is appropriate for simulating biological models where hundred of thousands of places could exist.

Contrary, immediate transitions fire instantly in a discrete fashion. If any of the immediate transitions are enabled, the numerical integration of the system of ODEs is interrupted to fire the immediate transitions. If more than one immediate transitions are enabled which are in conflict, the conflict is resolved via the weight value assigned to each immediate transition. For example, in Figure 1, the transitions *initiation* and *termination* could be enabled at the same time, if the firing rate of transition *syn* is high. However, we asign a higher weight to *termination* to resolve the conflict. Algorithm 2 summarises the steps involved in simulating an $\mathcal{HPN}$ model that contains continuous and immediate transitions only.

Algorithm 2 Hybrid Simulation

Input: $\mathcal{HPN}$ model with initial state $m(\tau_0)$, simulation interval $[\tau_0, \tau_{end}]$.

Output: Hybrid simulation traces over time.

```
construct a system of ODEs according to (2).
let τ = τ0, m(τ)= m(τ0);
update output trace by the current state m(τ);
while τ ≤ τend do
    initialize the ODE solver with the current marking m(τ);
    let d = the time step of the next output point;
    solve the system of ODEs till τ + d or an immediate transition is enabled;
    set τ to the time of the ODE solver
    check and fire all enabled immediate transitions;
    if τ + d is reached then
        record the simulation output at this time point;
    end if
end while
```

4. Hybrid Model of Caulobacter Crescentus Cell Cycle

In this section, we discuss in more details a hybrid implementation of the C. crescentus cell cycle based on the ODE model presented in [12]. The model consists of four modules and takes advantage of the discrete/continuous semantics of hybrid Petri nets, where discrete places and transitions implement triggered events of the cell cycles, while continuous places and transitions reproduce the ODE semantics.

4.1. Model Construction

Figure 5 provides the complete HPN model. The model consists of four modules: (1) a discrete module, (2) chromosome replication and methylation, (3) key regulatory protein module, and (4) cycle-dependent proteolysis of CtrA. In what follows, we discuss each part in more details.

4.1.1. Discrete Representation of Cell Cycle Events

In [11] a set of events have been proposed to guide the model dynamics during the deterministic simulation. These include the start and end of processes like DNA initiation and elongation. In our version of the model, we adopt a set of discrete places and immediate transitions to implement these events. This discrete part of the model is similar in logic to the simple hybrid Petri net example given in Figure 1. However, transition rates and arc weights are set based on the biological information presented in [11, 12].

This part of the model consists of four main events: replication start signal (the transition *init_replication*), ccrM replication (the transition *replicate_ccrM*), ctrA replication (the transition *replicate_ctrA*) and elongation termination (the transition *terminate_elongation*). Moreover, read arcs are used to satisfy the required preconditions of each transition, standard and reset arcs are used to implement the outcome of transition firing, and inhibitor arcs are meant to prevent the repeated firing of immediate transitions during the time of a single cell cycle.

The process of replication initiation takes place whenever the marking of Ini becomes greater than or equal to 0.05. The value of Ini is increased (continuously) via the synthesis process represented by the continuous transition *syn_ini*. The rate of this transition is a hill function of the concentration of the places $CtrAP$, $hCori$, and $DnaA$. Once *init_replication* is fired, it signals the start of the elongation process by putting the value of 0.05 in the continuous place *Elong*. It also adds a token to *Zring_state* and a value of 1.05 to DNA. The value of Ini is reset after the firing to account for the next event of the cell cycle.

Next, the transition *replicate_ccrM* is enabled when the value of *Elong* is greater than or equal to 0.2. The value of *Elong* is continuously increased by the transition *syn_Elong*. Once *replicate_ccrM* is fired, it will add a token to $hccrM$ and another token to *ccrM_rep_state*. The latter is used to prevent the repeated firing of *replicate_ccrM* as it is still enabled with respect to the value of *Elong*. Similarly, the transition *replicate_ctrA* becomes enabled as soon as

the value of *Elong* reaches 0.375. When it fires, it adds one token to *hctrA* and another one to *ctrA_rep_state*.

Eventually, when the value of *Elong* is further increased and reaches one, the transition *terminate_elongation* is fired, which resets the values of *Elong* as well as other discrete places so that it prepares the model for the next phase of the cell cycle. Figure 6a depicts the dynamics of *Elong* and *Ini* during two cycles of cell division.

4.1.2. Chromosome Replication and Methylation

The chromosome replication and methylation module is directly linked to the discrete module. The main components are the DNA variables (*hctrA*, *hccrM*, and *hCori*) as well as the *Ini* and *Elon* places that are related continuous processes. Each of these DNA places gets one marking whenever the corresponding transition fires (one times per one cell cycle). These DNAs are employed during the synthesis of the mRNAs of the key proteins (please see next module). The marking of these places are slowly consumed via the continuous transitions linked to them which represent biological degradations. Figure 6 shows the dynamics of the DNAs during the execution of the first two cell cycles. Please note that the place *Count* represents the number of chromosomes and is used to signal the separation of chromosomes. *Count* is increased from 1 to 2 when the *init_replication* fires and decreased back later when the *reset_zring_dna* is fired. Without increasing *Count*, the rate of *syn_Elong* will not be increased in a rate permitting the different elongation phases to take place.

4.1.3. Key Regularity Proteins Module

The cell cycles of the C. crescentus are mainly controlled by a set of key proteins. These proteins are represented in our model by the continuous places: *DnaA*, *CcrM*, *GcrA*, *SciP*, and *CtrA*. Each protein is systematically linked with a synthesis and degradation process. The synthesis of all proteins are modelled by mass action kinetics in terms of the corresponding mRNA and a synthesis rate constant. For example, the synthesis of *DnaA* is written as

$ks\_DnaA * dnaA$, where ks_DnaA is a rate constant and dnaA is the mRNA of the $DnaA$ protein. The synthesis process does not consume the mRNA value. Therefore, a modifier arc is used here to better reflect this semantics. Figure 7 provides the temporal dynamics of the key proteins.

The mRNAs of the key proteins are also having synthesis and degradation transitions associated with them. However, some of them make use of their DNA places during the synthesis (e.g., $dnaA$, $ccrM$, and $ctrA$) which link this module and the chromosome replication and methylation one. Please note that the synthesis of ccrM is delayed compared to the rest of mRNAs. Therefore, $IccrM$ was introduced in the original models in [11, 12]. Moreover, mRNA synthesis also includes the concentration of the key proteins forming a kind of feedback loops with them.

A special key protein of the network is $CtrA$. In addition to the synthesis and degradation, it exists in two forms: the active and inactive versions. The latter is modelled by the place $CtrAP$. $CtrA$ can be converted into $CtrAP$ via phosphorylation, while $CtrAP$ is converted back to $CtrA$ via dephosphorylation.

4.1.4. Cycle-dependent Proteolysis of CtrA

The degradation of $CtrAP$ is controlled by a cycle-dependent proteolysis network introduced in [12]. The main component of this network comprises the different forms of $ClpXP$. $ClpXP$ binds with $CpdR$ and forms the complexes $Complex1$, $Complex2$ and $Complex3$. The binding reactions are reversible reactions. $RcdA$ and cdG also take part in some binding processes.

Active PleD (PleDP) induces the degradation of cdG. According to the biological background presented in [12], PleC takes part in the dephosphorylation of PleD. However, the dynamics of PleC is not well known. Therefore, in [12], PleC is fitted according to experimental data using a trigonometric function of the simulation time. To do the same in our HPN, we have used the continuous place t to represent the current simulation time and reset it every 150 minutes (time of a single cycle) by the immediate transition $reset\_time$. Setting the rate of the continuous transition $timer\_trans$ to produce a rate of change of one to

the place t. Hence the value of this place will represent the simulation time at each numerical integration. While this semantics is not common for all biological models, we can see that HPN can also handle non-common logic, often necessary for biological modelling.

4.2. Model Simulation

The model is constructed and executed using Snoopy's hybrid simulator [39, 15]. Different hybrid algorithms are implemented in Snoopy [40, 41]. However, the model presented in this paper does not require the full hybrid interplay algorithm implemented in [4], since only a few discrete events are fired in each cell cycle. Therefore, runtime performance is not an issue for our model. The HPN model is included in the Supplementary Material 1 and the steps how to execute the model is summarised in Supplementary Material 2. The simulation traces are exported as *csv* files to plot the various figures in this paper.

5. Fuzzy Modelling of Caulobacter Crescentus Cell Cycle

In this section, we adopt the Caulobacter crescentus cell cycle model to demonstrate the tools provided by fuzzy hybrid Petri nets to tackle data uncertainty. Namely, we generate fuzzy bands and fuzzy membership functions for places of the HPN that are affected by input uncertainty of the kinetic parameters.

The *Caulobacter Crescentus Stalked Cells* model is hampered by kinetic data (parameters) uncertainty. The effect of kinetic data uncertainty can be obviously seen in this model in, e.g. , the species *cdG* and *RcdA* where the experimental data never perfectly fit with the traces obtained by simulating the model given in [12]. The uncertainties in the experimental data may come from different sources, such as measurement errors, operational errors by operators or environmental noise. Such uncertain experimental data cannot yield precise ODE models, therefore usually resulting in a large discrepancy between fitted models and underlying (unknown) accurate models. In this case, uncertainty

analysis methods should be applied to model the system with such uncertain kinetic data, and the resulting uncertain model can be in fact as a set of model instances, which may incorporate the real system behaviour. The fuzzy approach offers a good possibility to explore this issue using its membership functions and uncertain bands. The uncertain band produced by a fuzzy model gives the upper and lower bound of a model output, as well as the in-between curves, between which the true behaviour of the system may hide.

Fuzzification of the Caulobacter crescentus cell cycle model is performed by permitting some kinetic parameters to be fuzzy ones represented as triangular fuzzy numbers. We take the species *cdG* and *RcdA* as an example to illustrate the fuzzy approach. For the purpose of exploring the effect of uncertainty on the species *cdG*, and *RcdA*, we modelled the following kinetic parameters as fuzzy kinetic parameters: *ks_cdG* with (0.004,0.005,0.01), *kd_cdG* with $(0.07, 0.8, 1.3)$, *ks_RcdA* with (0.04,0.1642,0.2), *kd_RcdA* with (0.1,0.2323,0.5) and *Jd_RcdA* with (0.05,2.0,5.75). Please note that these kinetic parameters have been chosen as fuzzy ones as they are used by synthesis and/or degradation transitions of the corresponding species.

Figure 10 gives the fuzzy band of the species *cdG* (Figure 10a) compromising all traces corresponding to each parameter value combination taken over the considered fuzzy kinetic parameters. *Figure 10b* presents the membership function of the variable *cdG* at time 180, giving the pessimistic value, the most possible value and the optimistic value of the concentration at that time point.

Of course, we can obtain the membership function at each time point to check and locate the bounds of an output at each time point. In contrast, the corresponding ODE model or other so-called precise models only give one trace for one initial setting, and such a trace may not be the true trace and we never know where the true trace stays. However, the fuzzy model can accommodate and locate the true trace with its uncertain bands, which never can be achieved with precise modelling methods. For example, the true trace of the variable *cdG* (Figure 10) (close to experimental data) lies in the given fuzzy band (Figure 10a). Other species like *RcdA* also show the same effect.

Figure 11 gives the fuzzy band of the variable *RcdA* and the corresponding membership function at time 180. The discussion for the variable *cdG* applies equally to the variable *RcdA*. Figure 12 gives the membership functions of the two species at two different points of simulation time.

Besides, we also draw the 3D plots (fence plots) of the timed membership functions of both variables *cdG* and *RcdA* in Figure 13, which allows us to check the pessimistic, most possible and optimistic value of the concentrations of *cdG* and *RcdA* at each time point.

The fuzzy hybrid Petri net model together with the required steps to perform fuzzy simulation are provided in Supplementary Material S3.

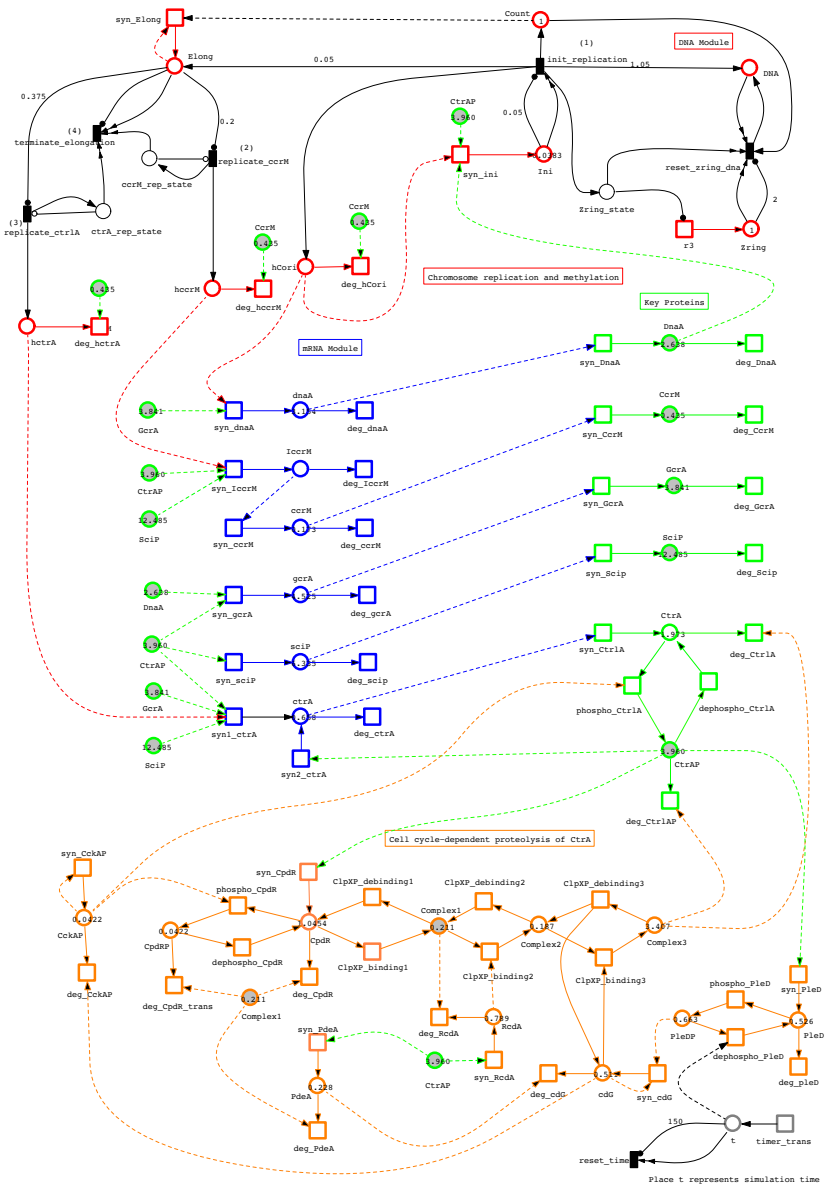
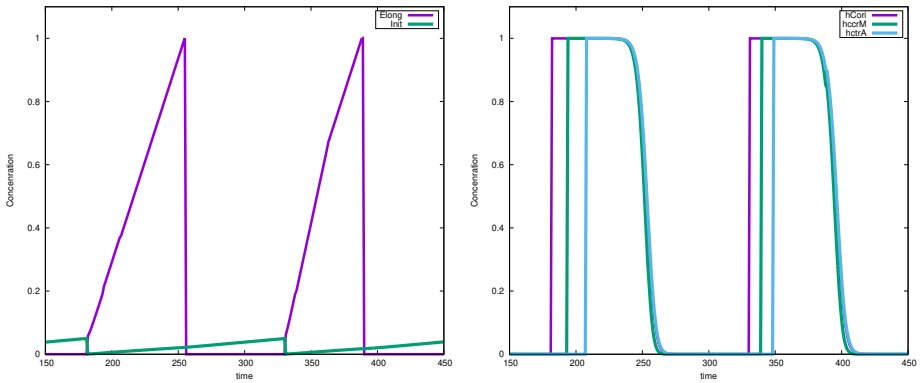


Figure 5: The hybrid Petri net representation of the C. crescentus cell cycle. Colours denote different modules with black colour denoting the discrete module, red color denoting chromosome replication and methylation, green color denoting key proteins, and orange color denoting cell cycle-dependent proteolysis. Places with same names and grey color are logical nodes used to simplify the connections. The graphical notations of nodes and arcs are explained in Table 1.



(a) (b)

Figure 6: Simulation results of the discrete and chromosome model components (two cycles). In (a), the marking of the place *Elong* starts to increase, once the value of Ini reached the threshold value (0.05), and (b) the places $hctrA$, $hccrM$ and $hCori$ each get a pulse, once their corresponding elongation event is triggered. Afterwards, they are constantly degraded over time.

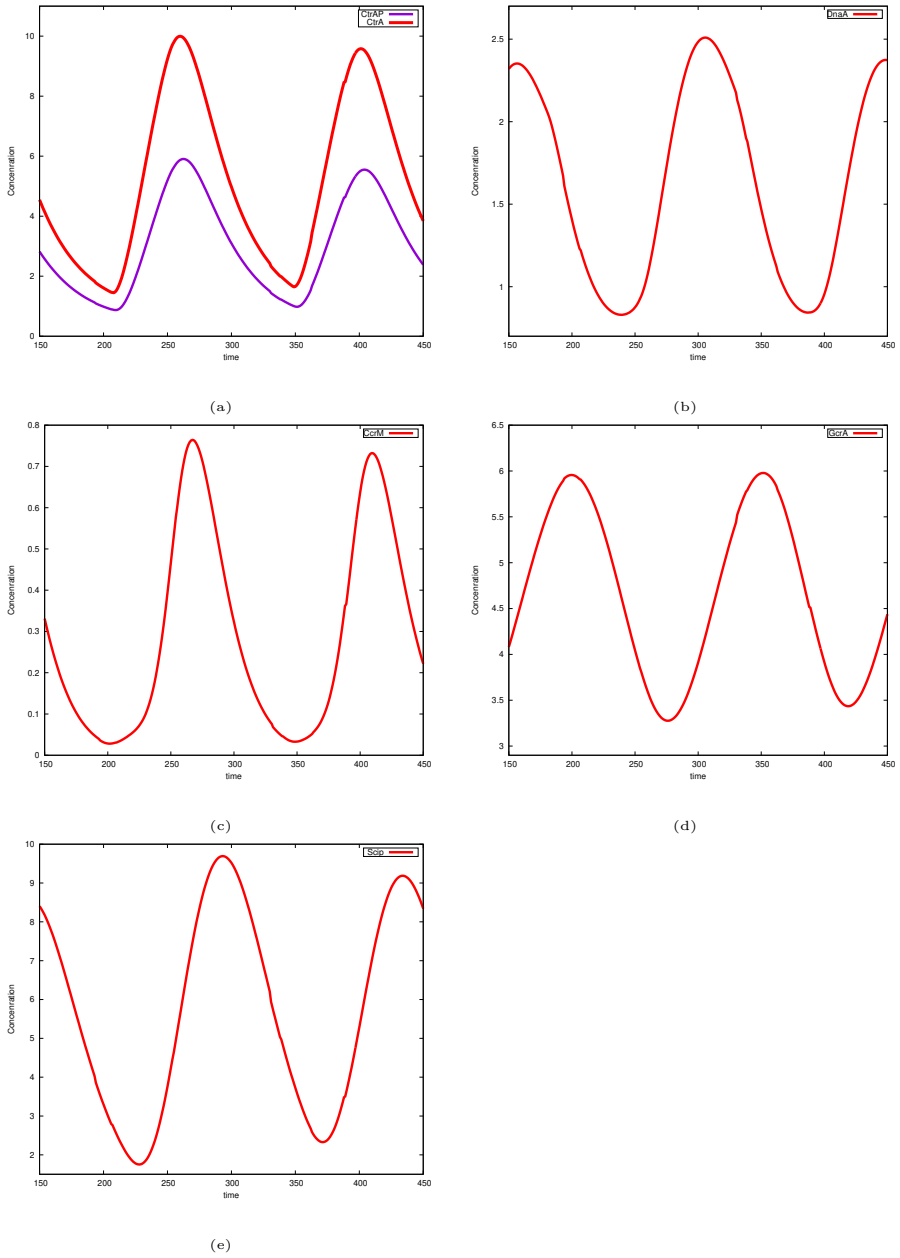


Figure 7: Simulation results of key proteins over two cell cycles: (a) CtrA (active and inactive versions), (b) DnaA, (c) CcrM, (d) GcrM and (d) Scip. The hybrid model reproduces the dynamics of the key proteins as given in [12].

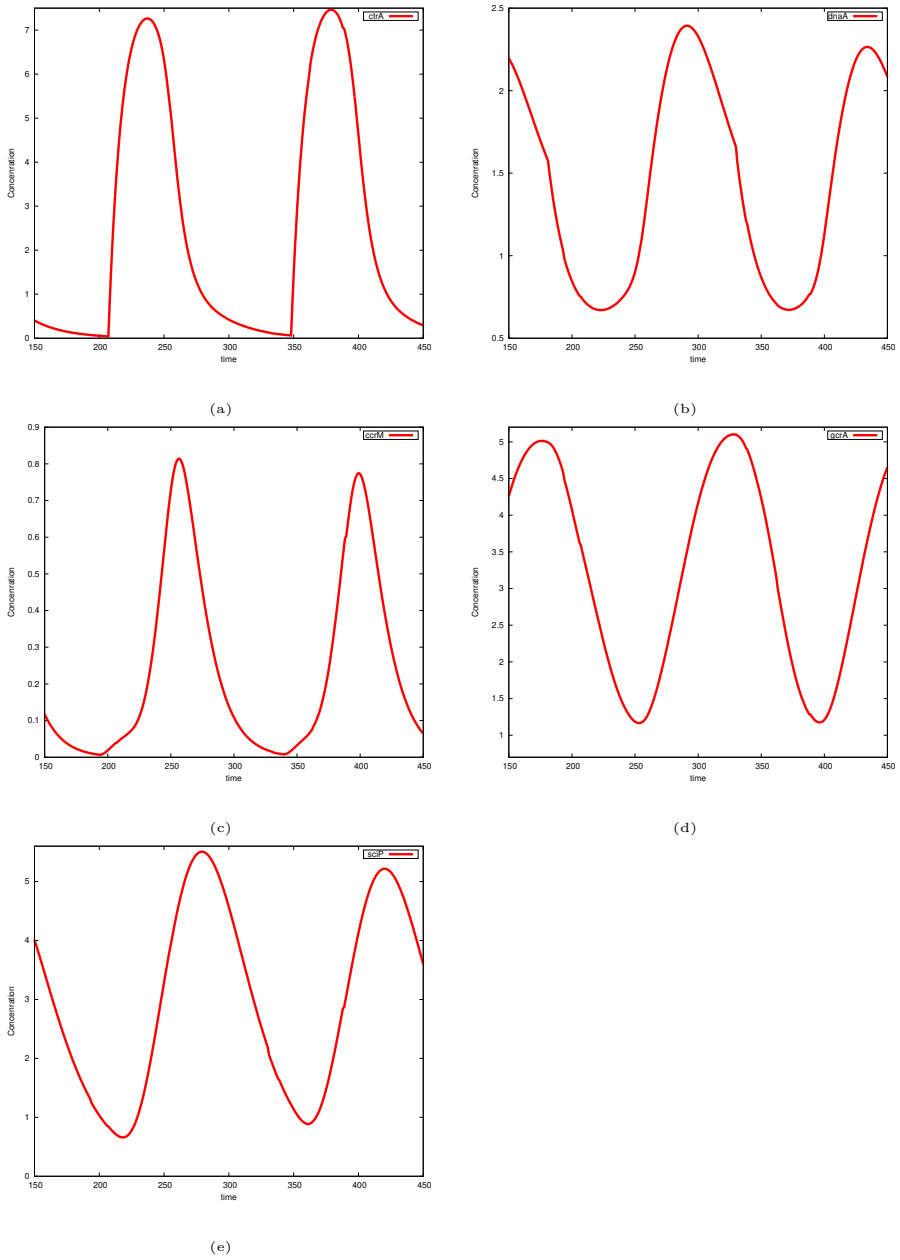


Figure 8: Simulation results of mRNAs of two cell cycles: (a) ctrA, (b) dnaA, (c) ccrM, (d) ccrM and (e) scip. The model also reproduces the dynamics of this part according to the results presented in [12].

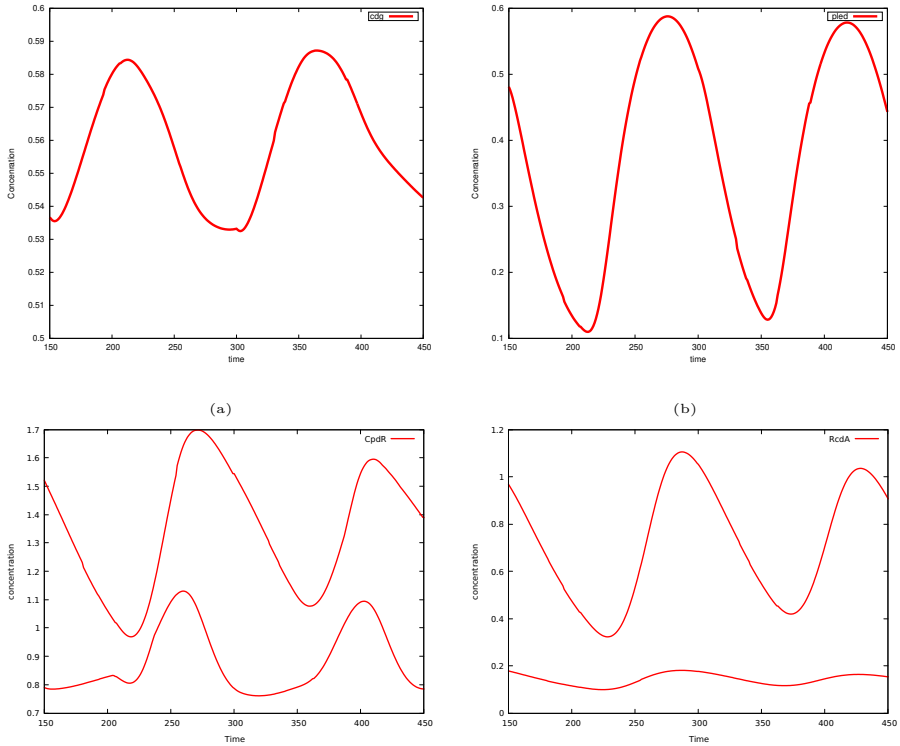


Figure 9: Simulation results of key proteins taking part in cycle-dependent proteolysis of CtrA.

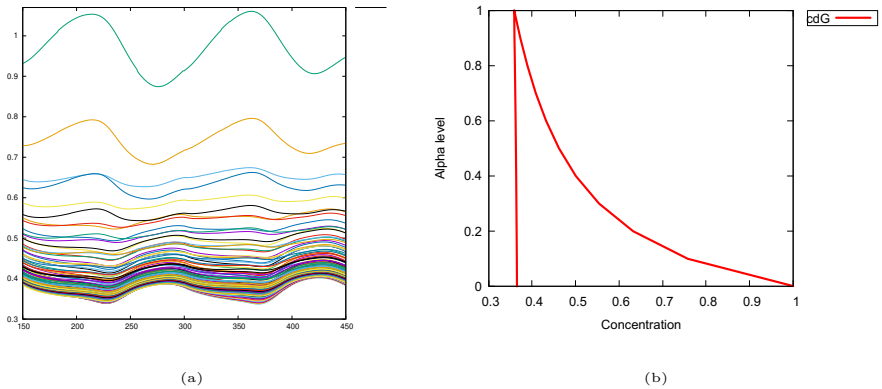
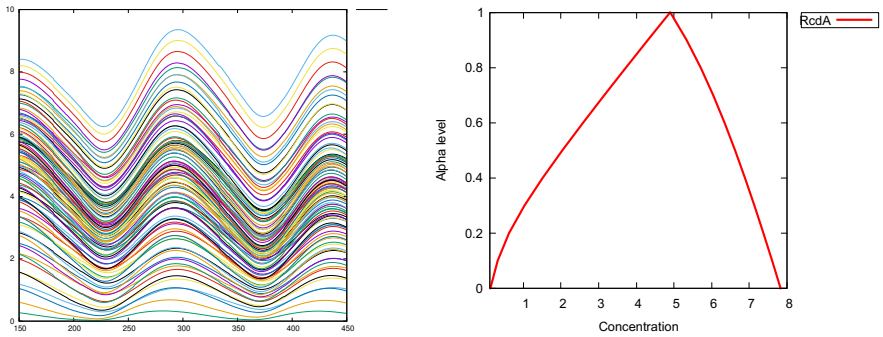
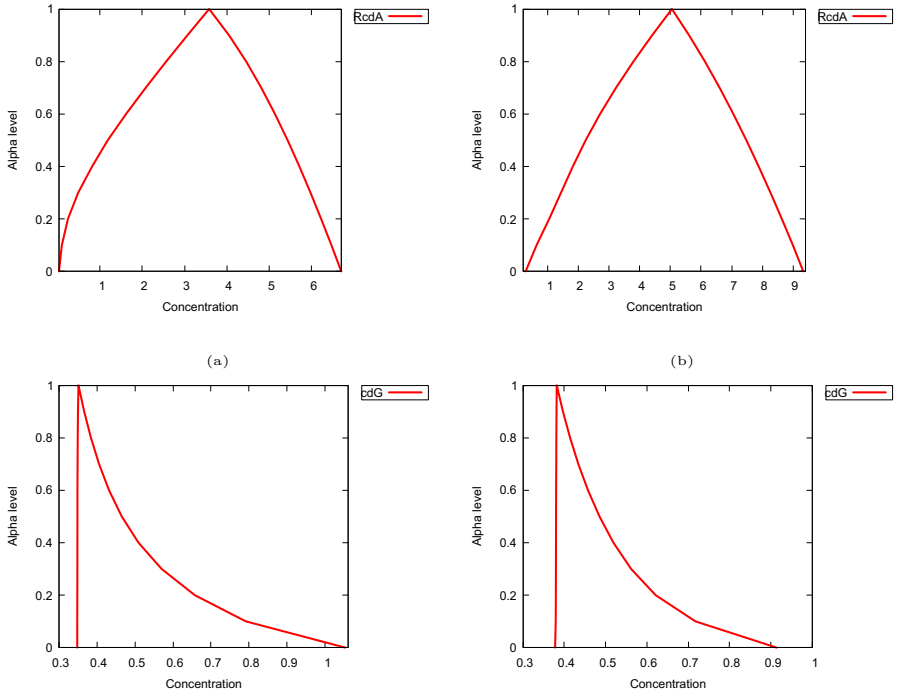


Figure 10: Fuzzy simulation traces of *cdG*: (a) a band of simulation traces describing the variability of the input kinetic parameter and (b) corresponding membership function at time 180.



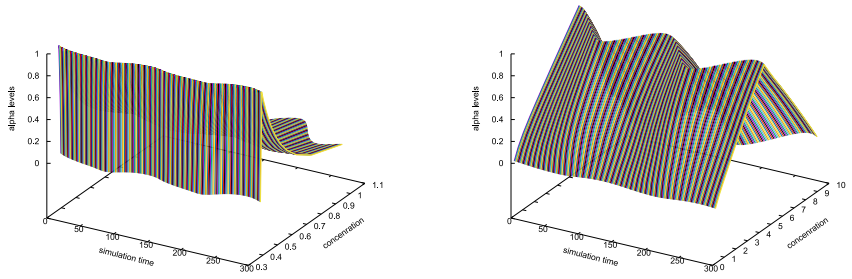
(a) (b)

Figure 11: Fuzzy simulation traces of *RcdA*: (a) a band of simulation traces describing the variability of the input kinetic parameter and (b) corresponding membership function at time 180.



(a) (b) (c) (d)

Figure 12: Membership functions of the places *RcdA* and *cdG*: (a), (d) at time 60 and (b), (c) at time 150.



(a) (b)

Figure 13: A fence plot of the membership functions over time of two cell cycles for the places: *RcdA* and *cdG*. Unlike the membership function representation in Figure 12, the fence plot shows how the membership function evolves over time. Thus it provides a better feedback about the model behaviour.

6. Conclusion and Future Work

In this paper, we have presented a hybrid Petri net model of the Caulobacter Crescentus Cell Cycle, which has exactly the same behaviour as the ODE model published in [12]. This model has then been extended to support fuzzy kinetic parameters in transitions where not enough experimental data is available to fit these kinetic rate constants. Hybrid Petri nets and fuzzy logic provide together a new tool to systems biologists to construct and then simulate biological systems at different scales and resolutions. The presented work can be extended in different ways. First, while we have presented a hybrid continuous/discrete model, stochastic transitions with and without fuzzy kinetic parameters have not been used in this paper. A similar idea can be applied in the future to construct fuzzy hybrid models that integrate continuous as well as stochastic logics. Second, the output band presented in Section 5 provides a valuable tool to show the different possibilities of the model results when the component of the corresponding fuzzy number is adjusted. This fuzzy band can be further combined with sensitivity analysis for providing a better feedback to the modeller.

It is worth mentioning that all ideas presented in this paper equally apply to coloured fuzzy hybrid Petri nets, see e.g. [31].

7. Acknowledgments

George Assaf is funded by the Katholischer Akademischer Auslaender-Dienst (KAAD) scholarship. Fei Liu is funded by National Natural Science Foundation of China (62273153). Mostafa Herajy is funded by a full scholarship from the Ministry of Higher Education of the Arab Republic of Egypt.

References

[1] F. Liu, M. Heiner, D. Gilbert, Hybrid modelling of biological systems: current progress and future prospects, Briefings in Bioinformatics 23 (3), bbac081. doi:10.1093/bib/bbac081.

[2] K. Takahashi, K. Kaizu, B. Hu, M. Tomita, A multi-algorithm, multi-timescale method for cell simulation, Bioinformatics 20 (4) (2004) 538–546. doi:10.1093/bioinformatics/btg442.

[3] M. Herajy, F. Liu, M. Heiner, Efficient modelling of yeast cell cycles based on multisite phosphorylation using coloured hybrid Petri nets with marking-dependent arc weights, Nonlinear Analysis: Hybrid Systems 27 (2018) 191 – 212. doi:10.1016/j.nahs.2017.09.002.

[4] M. Herajy, M. Heiner, Hybrid representation and simulation of stiff biochemical networks, J. Nonlinear Analysis: Hybrid Systems 6 (4) (2012) 942–959. doi:10.1016/j.nahs.2012.05.004.

[5] M. Herajy, M. Schwarick, M. Heiner, Hybrid Petri Nets for Modelling the Eukaryotic Cell Cycle, Springer Berlin Heidelberg, 2013, Ch. ToPNoC, pp. 123–141. doi:10.1007/978-3-642-40465-8_7.

[6] D. Barik, W. T. Baumann, M. R. Paul, B. Novak, J. J. Tyson, A model of yeast cell-cycle regulation based on multisite phosphorylation, Molecular Systems Biology 6 (1) (2010) 405. doi:10.1038/msb.2010.55.

[7] Z. Liu, Y. Pu, F. Li, C. A. Shaffer, S. Hoops, J. J. Tyson, Y. Cao, Hybrid modeling and simulation of stochastic effects on progression through the eukaryotic cell cycle, The Journal of Chemical Physics 136 (3) (2012) 034105. doi:10.1063/1.3677190.

[8] M. Ahmadian, J. J. Tyson, J. Peccoud, Y. Cao, A hybrid stochastic model of the budding yeast cell cycle, npj Systems Biology and Applications 6 (1). doi:10.1038/s41540-020-0126-z.

[9] J. Tyson, B. Novak, Regulation of the eukaryotic cell cycle: Molecular antagonism, hysteresis, and irreversible transitions, Journal of Theoretical Biology 210 (2) (2001) 249 – 263. doi:10.1006/jtbi.2001.2293.

[10] S. Kar, W. T. Baumann, M. R. Paul, J. J. Tyson, Exploring the roles of noise in the eukaryotic cell cycle., Proceedings of the National Academy

of Sciences of the United States of America 106 (16) (2009) 6471–6476. doi:10.1073/pnas.0810034106.

[11] S. Li, P. Brazhnik, B. Sobral, J. J. Tyson, A quantitative study of the division cycle of caulobacter crescentus stalked cells, PLOS Computational Biology 4 (1) (2008) 1–19. doi:doi.org/10.1371/journal.pcbi.0040009.

[12] C. Xu, H. Hollis, M. Dai, X. Yao, L. T. Watson, Y. Cao, M. Chen, Modeling the temporal dynamics of master regulators and CtrA proteolysis in Caulobacter crescentus cell cycle, PLOS Computational Biology 18 (1) (2022) 1–25. doi:10.1371/journal.pcbi.1009847.

[13] F. Liu, M. Heiner, M. Yang, Fuzzy stochastic Petri nets for modeling biological systems with uncertain kinetic parameters, PLOS ONE 11 (2) (2016) 1–19. doi:10.1371/journal.pone.0149674.

[14] R. David, H. Alla, Discrete, Continuous, and Hybrid Petri Nets, Springer, 2010.

[15] M. Herajy, F. Liu, C. Rohr, M. Heiner, Snoopy's hybrid simulator: A tool to construct and simulate hybrid biological models, BMC Systems Biology 11 (1) (2017) 71. doi:10.1186/s12918-017-0449-6.

[16] M. Herajy, F. Liu, C. Rohr, M. Heiner, Coloured hybrid Petri nets: An adaptable modelling approach for multi-scale biological networks, Computational Biology and Chemistry 76 (2018) 87–100. doi:10.1016/j.compbiolchem.2018.05.023.

[17] H. Matsuno, M. Nagasaki, S. Miyano, Hybrid Petri net based modeling for biological pathway simulation, Natural Computing 10 (3) (2011) 1099–1120. doi:10.1007/s11047-009-9164-6.

[18] A. Ismail, M. Herajy, E. Atlam, M. Heiner, A Graphical Approach for Hybrid Simulation of 3D Diffusion Bio-models via Coloured Hybrid Petri Nets, Modelling and Simulation in Engineering 2020, article ID 4715172, 14 pages. doi:10.1155/2020/4715172.

[19] A. Ismail, M. Herajy, M. Heiner, A graphical approach for the hybrid modelling of intracellular calcium dynamics based on coloured hybrid Petri nets, in: P. Zuliani, P. Liò (Eds.), Automated Reasoning for Systems Biology and Medicine, Vol. 30 of Computational Biology, Springer, Cham, 2019, Ch. 13, pp. 349–367, first online: 12. June 2019. doi:10.1007/978-3-030-17297-8_13.

[20] D. Gilbert, M. Heiner, L. Ghanbar, J. Chodak, Spatial quorum sensing modelling using coloured hybrid Petri nets and simulative model checking, BMC Bioinformatics 20 (supplement 4), accepted: July, 26 2018; published: April 2019. doi:10.1186/s12859-019-2690-z.

[21] M. Matsui, S. Fujita, S. Suzuki, H. Matsuno, S. Miyano, Simulated cell division processes of the xenopus cell cycle pathway by genomic object net, Journal of Integrative Bioinformatics (2004) 0001doi:10.1515/jib-2004-3.

[22] S. Fujita, M. Matsui, H. Matsuno, S. Miyano, Modeling and simulation of fission yeast cell cycle on hybrid functional Petri net, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E87-A (11) (2004) 2919–2927.

[23] H. Matsuno, Y. Tanaka, H. Aoshima, A. Doi, M. Matsui, S. Miyano, Biopathways representation and simulation on hybrid functional Petri net, In silico biology 3 (3).

[24] B. J. Daigle, M. K. Roh, L. R. Petzold, J. Niemi, Accelerated maximum likelihood parameter estimation for stochastic biochemical systems, BMC Bioinformatics 13 (1) (2012) 68. doi:10.1186/1471-2105-13-68.

[25] F. Liu, S. Chen, M. Heiner, H. Song, Modeling biological systems with uncertain kinetic data using fuzzy continuous Petri nets, BMC Systems Biology 12 (4) (2018) 42. doi:10.1186/s12918-018-0568-8.

[26] F. Liu, W. Sun, M. Heiner, D. Gilbert, Hybrid modelling of biological

systems using fuzzy continuous Petri nets, Briefings in Bioinformatics 22 (1) (2019) 438–450. doi:10.1093/bib/bbz114.

[27] G. Assaf, Fuzzy coloured Petri nets for modelling biological systems with uncertain kinetic parameters , Ph.D. thesis, BTU Cottbus, Dep. of CS (December 2022). doi:10.26127/BTUOpen-5851.

[28] W. Reisig, Understanding Petri Nets, Springer, 2013. doi:10.1007/978-3-642-33278-4.

[29] L. Zadeh, Fuzzy sets, Information and Control 8 (1965) 338–353.

[30] F. Liu, M. Heiner, D. Gilbert, Fuzzy Petri nets for modelling of uncertain biological systems, Briefings in Bioinformatics 21 (1) (2018) 198–210. doi:10.1093/bib/bby118.

[31] G. Assaf, M. Heiner, F. Liu, Colouring fuzziness for systems biology, Theoretical Computer Science 875 (2021) 52–64. doi:10.1016/j.tcs.2021.04.011.

[32] G. Assaf, M. Heiner, F. Liu, Coloured fuzzy Petri nets for modelling and analysing membrane systems, Biosystems 212 (2022) 104592. doi:10.1016/j.biosystems.2021.104592.

[33] S. Marino, I. B. Hogue, C. J. Ray, D. E. Kirschner, A methodology for performing global uncertainty and sensitivity analysis in systems biology, Journal of Theoretical Biology 254 (1) (2008) 178–196. doi:10.1016/j.jtbi.2008.04.011.

[34] V. Shen, Knowledge representation using high-level fuzzy Petri nets, IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans 36 (6) (2006) 1220–1227. doi:10.1109/TSMCA.2006.878968.

[35] H.-C. Liu, J.-X. You, Z. Li, G. Tian, Fuzzy Petri nets for knowledge representation and reasoning: A literature review, Engineering Applications

of Artificial Intelligence 60 (2017) 45–56. doi:https://doi.org/10.1016/j.engappai.2017.01.012.

[36] S. Mahapatra, S. K. Nanda, B. Panigrahy, A cascaded fuzzy inference system for indian river water quality prediction, Advances in Engineering Software 42 (10) (2011) 787–796. doi:10.1016/j.advengsoft.2011.05.018.

[37] G. Assaf, M. Heiner, F. Liu, Biochemical reaction networks with fuzzy kinetic parameters in Snoopy, in: L. Bortolussi, G. Sanguinetti (Eds.), Proc. CMSB 2019, Vol. 11773 of LNCS/LNBI, Springer, 2019, pp. 302–307. doi:10.1007/978-3-030-31304-3_17.

[38] M. Herajy, M. Heiner, Adaptive and bio-semantics of continuous Petri nets: Choosing the appropriate interpretation, Fundamenta Informaticae 160 (1-2) (2018) 53–80. doi:Haseltine0210.3233/FI-2018-1674.

[39] M. Heiner, M. Herajy, F. Liu, C. Rohr, M. Schwarick, Snoopy – a unifying Petri net tool, in: Proc. PETRI NETS 2012, Vol. 7347 of LNCS, Springer, 2012, pp. 398–407.

[40] M. Herajy, M. Heiner, Accelerated simulation of hybrid biological models with quasi-disjoint deterministic and stochastic subnets, in: E. Cinquemani, A. Donzé (Eds.), Hybrid Systems Biology: 5th International Workshop, HSB 2016, LNBI, Springer, 2016, pp. 20–38. doi:10.1007/978-3-319-47151-8_2.

[41] E. Haseltine, J. Rawlings, Approximate Simulation of Coupled Fast and Slow Reactions for Stochastic Chemical Kinetics, J. Chem. Phys. 117 (15) (2002) 6959–6969. doi:10.1063/1.1505860.