A Hybrid Petri Net Model of the Eukaryotic Cell Cycle

Mostafa Herajy and Martin Schwarick

Brandenburg University of Technology at Cottbus,
Computer Science Institute,
Data Structures and Software Dependability,
Postbox 10 13 44, 03044 Cottbus, Germany
http://www-dssz.informatik.tu-cottbus.de/

Abstract. System level understanding of the repetitive cycle of cell growth and division is crucial for disclosing many unexpected principles of biological organisms. The deterministic or stochastic approach are alone not sufficient to study such cell regulation due to the complex reaction network and the existence of reactions with different time scales. Thus, integration of both approaches is necessary to study such biochemical networks. In this paper we present a hybrid Petri net model to study the eukaryotic cell cycle using Generalised Hybrid Petri Nets. The proposed model is intuitively and graphically represented through Petri net primitives. Moreover, it can capture intrinsic and extrinsic noises and deploys stochastic as well as deterministic reactions. Additionally, self-modifying weights are motivated and introduced to Snoopy – a tool for animating and simulating Petri nets.

Keywords: Generalised hybrid Petri nets; hybrid modelling; eukaryotic cell cycle

1 Introduction

The reproduction of eukaryotic cells is controlled by a complex regulatory network of reactions known as cell cycle [17,18,21]. Through it, cells grow, replicate and divide into two daughter cells [12,19]. This regulation cycle consists of four phases: S phase (synthesis) and M phase (mitosis) separated by two gap phases: G1 and G2 [21]. During the S phase, the cell replicates all of its components, while it divides each component more or less evenly between the two daughter cells at the end of the M phase [12]. After the S phase, there is another gap (G2) where the cell ensures that the duplication of DNA has completed and prepares itself for mitosis. Newborn cells are not replicated and located at the G1 gap. Furthermore, the processes of synthesis and mitosis alternate with each other during the reproduction process. Understanding such control cycles is crucial for revealing defects in cell growth which underlies many human diseases (e.g., cancer) [22].

In the eukaryotic cell cycle, the alternation between the S and the M phase as well as the balance of growth and division is governed by the activity of a family

of cyclin-dependent protein kinases (CDK) [12]. Therefore, many computational models have been constructed to study the control system of CDK (e.g., in [1,12,17,18,21]). Some of these models are based on the deterministic approach which represents changes of species concentrations as continuous variables that evolve deterministically and continuously with respect to time. However, such approach does not capture the variability of cell size due to the fluctuation of some species which usually exist in low numbers of molecules [4]. Motivated by this argument, a number of stochastic models have been created and simulated using either a stochastic simulation algorithm (e.g., [12]) or by introducing noise to the model through Langevin equation [20]. However, the stochastic approach is computationally expensive, particularly when the model under study contains reactions of high rates or species of large numbers of molecules.

Similarly, the eukaryotic cell cycle model exhibits high reaction rates of some reactions while some other reactions have low rates. The latter types are responsible for the intrinsic noise due to molecular fluctuations [14]. The existence of reactions of different time scales (fast and slow) suggests the simulation using a hybrid approach. In [14] and [19] two different hybrid approaches are used to simulate the progression of cell cycle.

Correspondingly, Generalised Hybrid Petri Nets ($GHPN_{bio}$) have been introduced, in [10] and [11], to represent and simulate stiff biochemical networks where fast reactions are represented and simulated continuously, while slow reactions are carried out stochastically. $GHPN_{bio}$ provide rich modelling and simulation functionalities by combining all features of Continuous Petri Nets [2] and Extended Stochastic Petri Nets [15], including three types of deterministic transitions. Moreover, the partitioning of the reaction networks can either be done off-line before the simulation starts or on-line while the simulation is in progress. The implementation of $GHPN_{bio}$ is available as part of Snoopy [8] - a tool to design and animate or simulate hierarchical graphs, among them qualitative, stochastic, continuous and hybrid Petri nets. Indeed, the cell cycle model is an ideal case where the majority of $GHPN_{bio}$ features can be demonstrated.

In this paper we present another argument to motivate the hybrid simulation of the cell cycle control system. The cell cycle model contains some components which would be better represented as continuous processes (e.g., volume growth), while other reactions of low rates are vital to represent them as stochastic processes. For instance, Mura and Csikasz-Nagy constructed in [17] a stochastic version of the model in [1] using stochastic Petri nets. However, they got stuck with the problem of representing cell growth processes which evolve continuously and exponentially with respect to time using stochastic Petri net primitives. Indeed cell growth is a typical example where continuous transitions could be used. Moreover, our proposed model is graphically and intuitively represented in terms of Petri nets.

The paper is organised as follows: we start off by a brief introduction of Generalised Hybrid Petri Nets. After that, some related work is pinpointed. Next, we present our hybrid Petri net model of the eukaryotic cell cycle and describe in detail some of its key modelling components. In Section 5 we show

the simulation results produced by Snoopy's hybrid simulation engine. Finally, we sum up with conclusions and outlook.

2 Related Work

Mura and Csikasz-Nagy constructed in [17] a stochastic Petri net model based on the work of [1] to study the effect of noise on cell cycle progression. However, some components could not intuitively be represented using SPN primitives only (e.g., cell growth). Moreover, their model is based on phenomenological rate laws (e.g., Michaelis-Menten) which do not work well with stochastic simulation algorithms [12]. Sabouri-Ghomi et al. [18], and Kar et al. [12], asserted that applying Gillespie's stochastic simulation algorithm [3] directly to phenomenological rate laws might produce incorrect results. Therefore, they unpacked the phenomenological deterministic model of Tyson-Novak [21] in terms of elementary mass action kinetics. The Tyson-Novak model is based on a bistable switch between the complex CycB-Cdk1 (denoted by variable X) and the complex Cdh1-APC (denoted by the variable Y). CycB-Cdk1 phosphorylates Cdh1-APC and free Cdh1-APC catalyses the degradation of CycB-Cdk1. To model a complete cell cycle, Kar et al. [12] unpacked the effect of Cdc20 and Cdc14 which are lumped in the variable Z in the Tyson-Novak model. High activity of CycB-Cdk1 promotes the synthesis of Cdc20 which activates Cdc14. Finally the dephosphorylated Cdc14 activates Cdh1-APC. The Kar et al. model accounts for both intrinsic and extrinsic noises. Intrinsic noise is due to the fluctuation of species of low numbers of molecules, while extrinsic noise is due to the unequal division of the cell between the two daughter cells [12].

In [19], a hybrid model which combines ordinary differential equations (ODEs) and discrete Boolean networks has been constructed to adapt quantitative as well as qualitative parts in the same model. The latter approach requires less knowledge about realistic kinetic rate constants. Liu et al. [14] simulate the stochastic model of [12] by statically partition the model reactions into slow and fast ones. However, as they have reported in their paper, the resulting hybrid model requires more simulation time than the original stochastic one. Therefore, they have (re)packed fast reactions in terms of phenomenological rate lows. The model presented in this paper differs from the Liu et al. one in the intuitive graphical representation and execution time of the hybrid model.

In this paper a hybrid Petri net model of the eukaryotic cell cycle is presented. The model is hybrid in the sense that it combines continuous, stochastic and immediate transitions to represent deterministic, stochastic and control components. Our main goal is to show how such class of models are intuitively represented and executed using hybrid Petri net primitives. Using Snoopy's simulator, it can be simulated either deterministically, stochastically or in a hybrid way.

3 Generalised Hybrid Petri Nets

To model stiff biochemical networks, $GHPN_{bio}$ [11] combine both stochastic and continuous elements in one and the same model. Indeed, continuous and stochastic Petri nets complement each other. The fluctuation and discreteness can be conveniently modelled using stochastic simulation and at the same time, the computationally expensive parts can be simulated deterministically using ODE solvers. Modelling and simulation of stiff biochemical networks are outstanding functionalities that $GHPN_{bio}$ provide for systems biology.

Generally speaking, biochemical systems can involve reactions from more than one type of biological networks, for example gene regulation, metabolic pathways or signal transduction pathways. Incorporating reactions which belong to distinct (biological) networks, tends to result into stiff systems. This follows from the fact that gene regulation networks species may contain a few number of molecules, while metabolic networks species may contain a large number of molecules [13].

In the rest of this section, we will give a brief introduction of $GHPN_{bio}$ in terms of the graphical representation of its elements as well as the firing rules and connectivity between continuous and stochastic net parts.

3.1 Elements

The $GHPN_{bio}$ elements are classified into three categories: places, transitions and arcs.

$GHPN_{bio}$ offer two types of places: discrete and continuous. Discrete places (single line circle) hold non-negative integer numbers which represent e.g., the number of molecules of a given species (tokens in Petri net notions). On the other hand, continuous places - which are represented by shaded line circle - hold non-negative real numbers which represent the concentration of a certain species.

Furthermore, $GHPN_{bio}$ offer five transition types: stochastic, immediate, deterministically delayed, scheduled, and continuous transitions [7]. Stochastic transitions which are drawn in Snoopy as a square, fire with an exponentially distributed random delay. The user can specify a set of firing rate functions, which determine the random firing delay. The transitions' pre-places can be used to define the firing rate functions of stochastic transitions. Immediate transitions (black bar) fire with zero delay, and have always highest priority in the case of conflicts with other transitions. They may carry weights which specify the relative firing frequency in the case of conflicts between immediate transitions. Deterministically delayed transitions (represented as black squares) fire after a specified constant time delay. Scheduled transitions (grey squares) fire at user-specified absolute time points. Continuous transitions (shaded line square) fire continuously in the same way like in continuous Petri nets. Their semantics are governed by ODEs which define the change in the transitions' pre- and post-places. More details about the biochemical interpretation of deterministically delayed, scheduled, and immediate transitions can be found in [9] and [15]. To

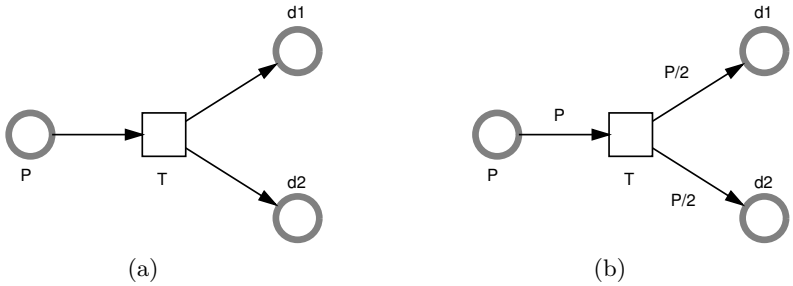


Fig. 1. Self-modifying weight illustrated by a simple biological example. (a) cell division cannot be modelled (b) cell division can intuitively be modelled.

simplify the presentation, we occasionally refer to stochastic, immediate, deterministically delayed or scheduled transitions as discrete transitions.

The connection between those two types of nodes (places and transitions) takes place using a set of different arcs (edges). $GHPN_{bio}$ offer six types of arcs: standard, inhibitor, read, equal, reset and modifier arcs. Standard arcs connect transitions with places or vice versa. They can be discrete, i.e., carry non-negative integer-valued weights (stoichiometry in the biochemical context), or continuous i.e., carry non-negative real-valued weights. In addition to their influence on the enabling of transitions, they affect also the place marking when a transition fires by adding (removing) tokens from the transition's post-places (pre-places). For more details see [10].

To support the special modelling requirements of some biological models (e.g., cell cycle model), arc weights are allowed to be a pre-place of a transition [23] or even a function which is defined in terms of the transition's pre-places [16].

Consider the following simple biological example. When a cell divides the mass between two daughter cells, each daughter takes approximately half of the mass. This example cannot be modelled using standard Petri nets as shown in Figure 1a. In Figure 1b, Using self-modifying weight; the ongoing arc of the transition "t" has weight equal to the marking of the place "P", while each of the two outgoing arcs has weight equal to the half marking of place "P".

Motivated by the case study of this paper, self-modifying weights are introduced to all arc types supported by Snoopy (standard, read, inhibitor, and equal arc). For more detail see Section 4.2.

Extended arcs like inhibitor, read, equal, reset, and modifier arcs can only be used to connect places to transitions, but not vice versa. A transition connected with an inhibitor arc is enabled if the marking of the pre-place is less than the arc weight. Contrary, a transition connected with a read arc is enabled if the marking of the pre-place is greater than or equal to the arc weight. Similarly, a transition connected using an equal arc is enabled if the marking of the pre-place is equal to the arc weight.

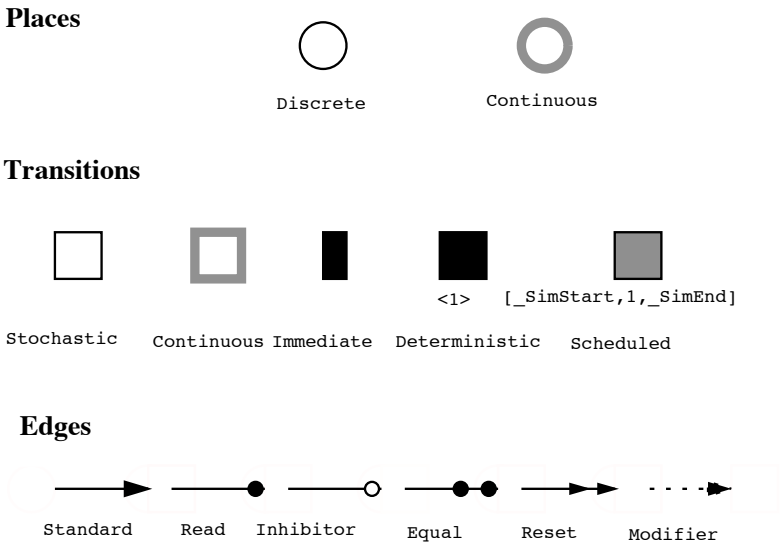


Fig. 2. Graphical representation of the $GHPN_{bio}$ elements. Places are classified as discrete and continuous, transitions as continuous, stochastic, immediate, deterministically delayed, and scheduled, and edges as standard, inhibitor, read, equal, reset, and modifier.

The other two remaining arcs do not affect the enabling of transitions. A reset arc is used to reset a place marking to zero when the corresponding transition fires. Modifier arcs permit to include any place in the transitions' rate functions and simultaneously preserve the net structure restriction.

The connection rules and their underlying formal semantics are discussed in more details below. Figure 2 provides a graphical illustration of all elements. Although this graphical notation is the default one, they can be easily customised using the Petri nets editing tool, Snoopy.

3.2 Connection Rules

A critical question arises when considering the combination of discrete and continuous elements: how are these two different parts connected with each other? Figure 3 provides a graphical illustration of how the connection between different elements of $GHPN_{bio}$ takes place.

Firstly, we will consider the connection between continuous transitions and the other elements of $GHPN_{bio}$. Continuous transitions can be connected with continuous places in both directions using continuous arcs (i.e., arc with real-valued weight). This means that continuous places can be pre- or post-places of continuous transitions. These connections typically represent deterministic biological interactions.

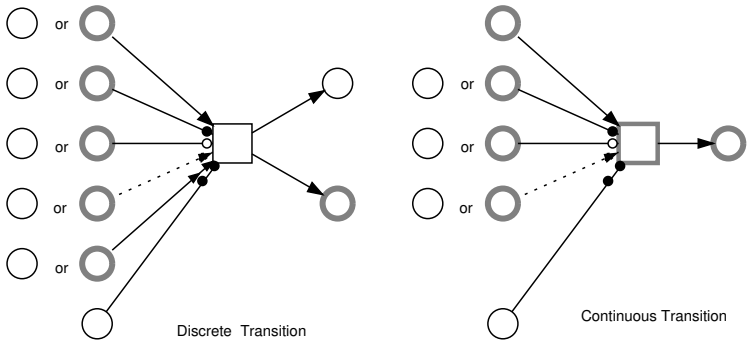


Fig. 3. Possible connections between $GHPN_{bio}$ elements. The restrictions are: discrete places can not be connected with continuous transitions using standard arcs, continuous places can not be tested with equal arcs, and continuous transitions can not use reset arcs.

Continuous transitions can also be connected with discrete places, but only by one of the extended arcs (inhibitor, read, equal, and modifier). This type of connection allows a link between discrete and continuous parts of the biochemical model.

Discrete places are not allowed to be connected with continuous transitions using standard arcs, because the firing of continuous transitions is governed by ODEs which require real values in the pre- and post-places. Hence, this cannot take place in the discrete world.

Secondly, discrete transitions can be connected with discrete or continuous places in both directions using standard arcs. However, the arc's weight needs to be considered. The connection between discrete transitions and discrete places takes place using arcs with non-negative integer numbers, while the connection between continuous places and discrete transitions is weighted by non-negative real numbers. The general rule to determine the weight type of arcs is the type of the connected place.

4 The Model

Figure 4 shows the hybrid Petri net model based on the previous one introduced by Kar et al. in [12]. Proteins, genes and mRNAs are represented by places. Transitions represent reactions. We use the same kinetic parameters and initial values. For the sake of compactness will not repeat them again here. Initial markings are shown inside the places. Moreover, we use Snoopy's logical node features to simplify connections between different nodes. For example, place X and Y are involved in many reactions which decreases the network's readability. We repeat them multiple times with same names to keep the model understandable

(logical places). Likewise the transition "divide" (logical transitions). Furthermore, the increase of cell volume size is intuitively represented using a continuous transition with a rate $\mu \cdot V$, where μ is the growth factor and V is the cellular volume.

The model contains three different transition types: continuous, stochastic, and immediate. Continuous transitions simulate the corresponding reactions deterministically, while stochastic transitions carry them out stochastically. The latter transitions are responsible for molecular fluctuations. Immediate transitions monitor the model evolution and perform the division when the free number of molecules of Cdh1_APC reaches a certain threshold ($\hat{Y} = Y + YX + XY$).

In the sequel we present in more detail some of the model's key components and the corresponding $GHPN_{bio}$ representations.

4.1 Decision to Perform Division

When the number of molecules of $\hat{Y}$ becomes greater than a certain threshold (in our case 1200), the cell can divide the mass and other components between the two daughter cells. In Figure 5, this process is represented by an immediate transition "check" with the weight $\hat{Y} > threshold$. Recall that immediate transition weights determine the firing frequencies of immediate transitions in case of conflicts. A weight of zero means that a transition cannot fire at all. Therefore, when the transition "check" has weight greater than zero, it adds a token to the place "ready_to_divide" which signals the transition "divide" to carry out the division. To give the transition "divide" a chance to fire before re-checking the value of $\hat{Y}$, an inhibitor arc is used to constrain this case.

An interesting characteristic of the model is the division process. Although the division can take place when the value of $\hat{Y}$ is greater than a certain threshold, it does not do that all the times. For example, at the beginning of the simulation, the initial value of $\hat{Y}$ satisfies the dividing criterion. However; the cell should not divide because it is still at G1 phase which means that it has to replicate before it can divide. We model these cases by adding a new immediate transition which detects the critical value of $\hat{Y}$, before checking for division. Therefore the transition "critical" monitors the value of $\hat{Y}$. When the value of $\hat{Y}$ goes below a certain threshold, it enables the division process.

4.2 Cell Division and Self-modifying Weights

When a cell divides, it divides all of its components more-or-less evenly between two daughter cells. This is another ideal case to demonstrate self-modifying weights [23]. In Figure 5, when the transition "divide" fires, it removes all of the current marking of the place V and adds $V/2$ to it. To permit uneven division of the cell volume and other components, arc weights can be a function which operates on the current place marking [16]. However, we restrict the places used in arc weights to the transitions' pre-places to maintain the Petri net structure.

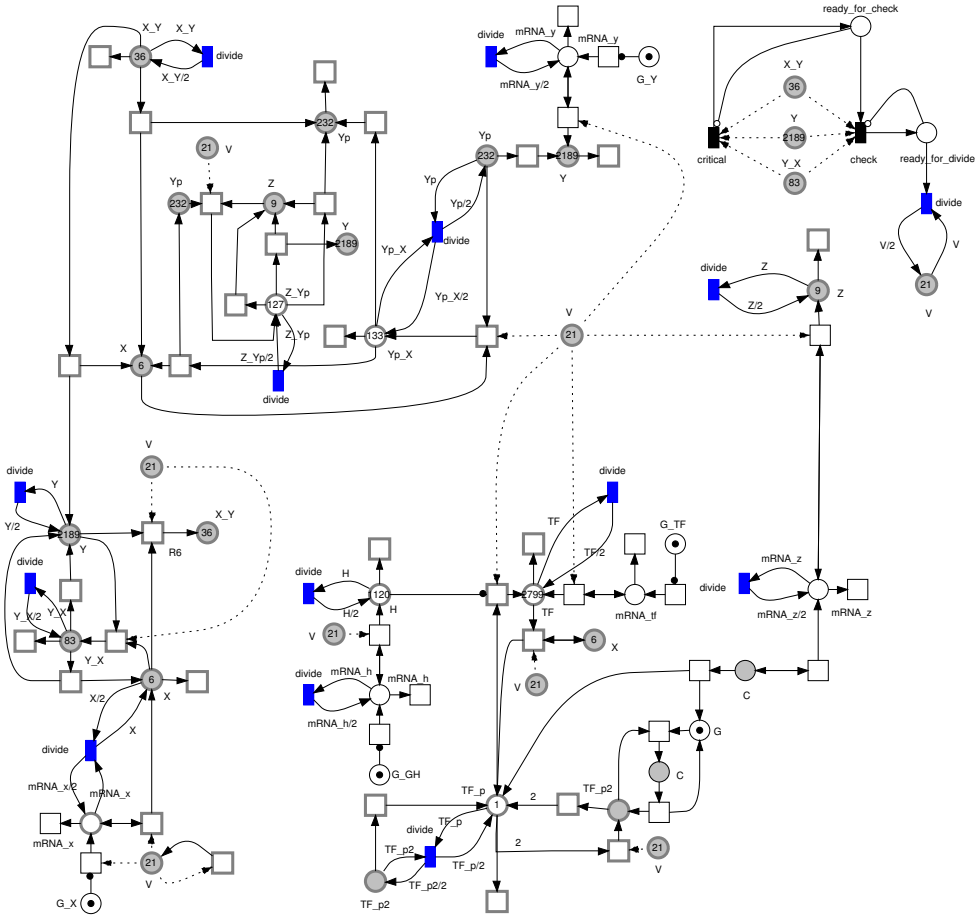


Fig. 4. A Generalised Hybrid Petri Nets representation of the eukaryotic cell cycle. The model employs different types of transitions: continuous, stochastic and immediate. All reactions affecting m_RNAs are represented and simulated stochastically. Repetitive nodes (places and transitions) with same names are logical nodes. When the transition "divide" fires, it divides the current place marking more or less equally. The type of division (equal, or unequal) depends on the outgoing arc weight and its effect takes place using self-modifying weight.

4.3 Transition Partitioning

The model in Figure 4 contains transitions which fire at different rates. For instance, transition "R3", as illustrated in Figure 6a, fires more frequently than "R1". Slow transitions should be simulated stochastically to account for molecular fluctuations, while fast transitions need to be simulated continuously to

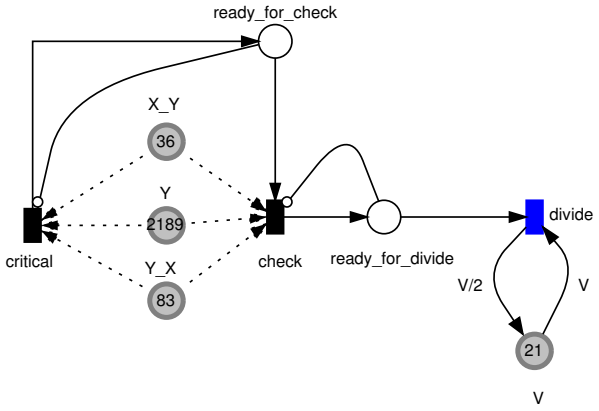


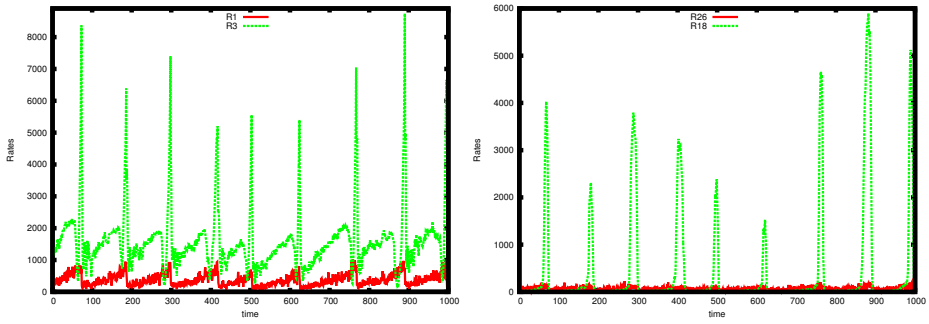
Fig. 5. A sub-net for modelling the decision of the division process. The transition "critical" monitors the value of $\hat{Y}$ and adds a token to "ready_for_check" when $\hat{Y} < 300$. Later, when the value of $\hat{Y}$ increases and becomes greater than a threshold (1200), the transition "check" fires and adds a token to "ready_for_divide" which signals the transition "divide" to perform the division. Inhibitor arcs are used as a check point for the sequence of events: critical → check → divide.

increase the numerical efficiency. Indeed, the latter types consume the majority of computational resources.

In this model, transitions are partitioned statically before the simulation starts. The transition type is decided by executing a single run and analyse the results as in Figure 6. Increasing (decreasing) the accuracy of the model results involves converting more continuous (stochastic) transitions into stochastic (continuous) ones. Similarly, controlling the speed of the model simulation will require the opposite procedure.

Another approach to do the partitioning is to perform it dynamically during the simulation. Using this technique, a transition changes its type from stochastic to continuous or vice versa according to the current firing rate. $GHPN_{bio}$ provide the user with a trade-off between efficiency and accuracy by permitting the user to specify two thresholds: $a_{0_{min}}$ and $a_{0_{max}}$, the minimum and maximum cumulative propensity, respectively. Moreover, two other thresholds are required to perform dynamic partitioning: the place marking threshold and the transition rate threshold. The former is used to ensure that species concentrations are large enough to be simulated continuously, while the latter is used to partition transitions into fast and slow based on their rates. For a transition to be simulated continuously its rate has to exceed the rate threshold and the marking of all its pre-place must be greater than the marking threshold.

Nevertheless, in both cases cell growth has to be represented and simulated continuously. Using off-line partitioning, this can be easily told to the simulator by drawing a continuous transition. However, in the case of dynamic partitioning;



(a) (b)

Fig. 6. Example of different transition firing rates. (a) transition "R3: X+Y → Y_X" fires more frequently than transition "R1: mRNA_x → mRNA_x+X" and (b) transition "R18: H → H+TF" fires much more than "R26: mRNA_tf→ mRNA_tf +TF".

the transition rate threshold should be set less than the expected rate of cell growth.

5 Simulation Results

In this section, we compare the simulation results of the following scenarios: when all of the reactions are simulated stochastically, when reactions related only to mRNAs are simulated stochastically and when all reactions are simulated continuously. In all cases cell growth is simulated using a continuous transition. Figure 7 shows the results of the three approaches using species with low numbers of molecules (mRNA_x and mRNA_z). Since reactions related to mRNAs are simulated stochastically in hybrid and stochastic simulations, their results are close to each other.

Figure 8 shows time course simulation results of proteins X and Y. In hybrid and stochastic simulations, X and Y are affected with fluctuations of mRNAs. while in continuous one there is no such effect.

Figure 9 compares continuous and hybrid simulation results of the volume size (V). Using continuous simulation, cell divides all the time equal and the model produces no variability in its volume size. The hybrid simulation shows variability in the volume size because species of low numbers of molecules (e.g., mRNAs) are simulated stochastically which account for the molecular fluctuations and therefore, they are responsible for the intrinsic noise [12].

As a conclusion, the hybrid simulation approach can reproduce the results of the stochastic approach. However, substantially amount of simulation time could be saved. Fortunately, the resulting system of ordinary differential equations (ODEs) of this model is not stiff (for more details, see [5] and [6]). Therefore, an explicit ODE solver can be used to increase the performance of the hybrid simulation engine in connection with the stochastic simulation algorithms (SSA).

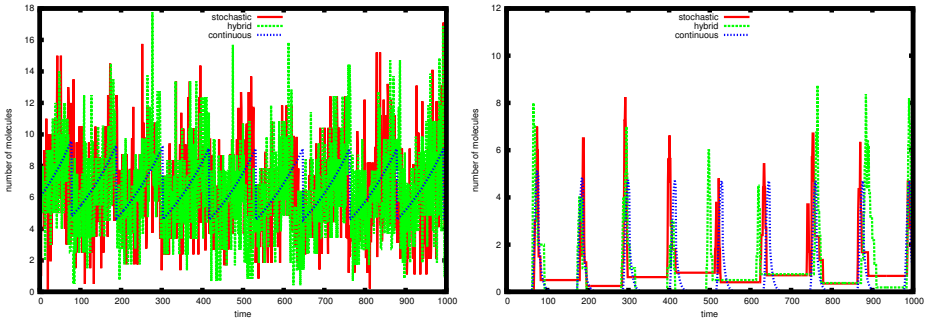


Fig. 7. Time course result of the model in Figure 4 using Snoopy simulator (a) mRNA_x and (b) mRNA_z.

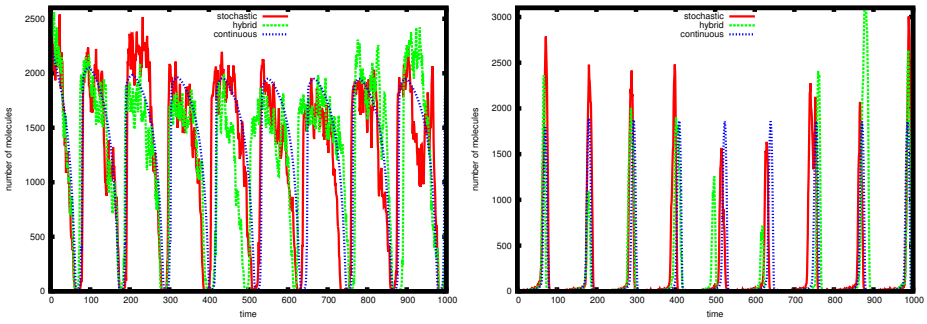


Fig. 8. Time course result of species with large number of molecules; (a) Y and (b) X.

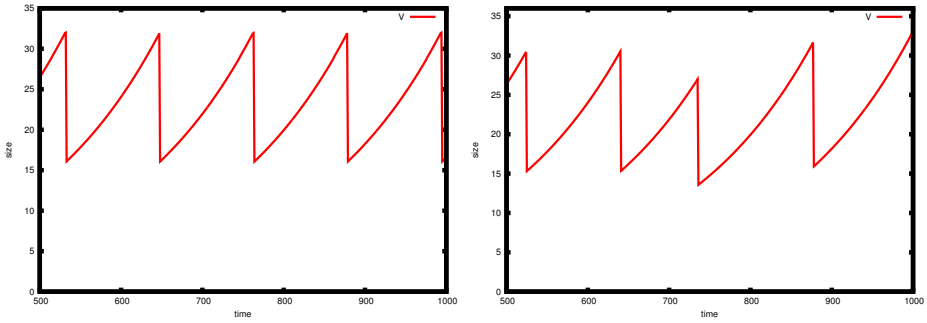


Fig. 9. Continuous and hybrid simulation results for the cellular volume (V); (a) continuous result and (b) hybrid simulation result.

6 Conclusions and Outlook

In this paper we have presented a hybrid Petri net model of the eukaryotic cell cycle. The model can be executed using either continuous, stochastic or hybrid simulators. It employs continuous, stochastic and immediate transitions to intuitively represent the entire model logic.

The model is implemented using Snoopy which is available free of charge at http://www-dssz.informatik.tu-cottbus.de/snoopy.html.

Self-modifying weight is a new added feature to Snoopy which is currently not available in the official Snoopy release.

From the simulation results we notice that hybrid simulation produces results close to the stochastic one while simulation efficiency could be preserved. Indeed, the reactions of this model could easily be separated into slow and fast reactions, which makes it an ideal case study for hybrid simulation algorithms.

Self-modifying arcs are of paramount importance to model such biological cases since they provide a direct tool to program some biological phenomenon (e.g., cell division). Therefore, we intend to add more functionalities into this direction to permit more user-defined operators depending on transition's pre-places.

So far the partitioning of the reactions into stochastic and deterministic ones is carried out using a heuristic approach (see Section 4.3). However, as it has been risen during the review process; a better justification for the partitioning could be performed. For instance, the fast processes can be regarded as processes that could be described by quasi (or pseudo)-steady state approach, assuming that they reach equilibrium rapidly. In other words, they could be better described by setting the corresponding ODE to zero and solving for the fast variables. In contrast, continuous dynamics could be seen as more appropriate for abundant molecules whose concentration display a small coefficient of variation, and stochastic dynamics for those molecules evolving at low copy number.

The presented model could be viewed as a sub-net in a bigger network of reactions (e.g., modelling budding yeast cell cycle or Fission yeast cells). Snoopy's hierarchical nodes might simplify such task as they provide an easy tool to insert a sub net in a bigger one.

7 Acknowledgements

The authors acknowledge the comments of Monika Heiner during the implementation of $GHPN_{bio}$ and the preparation of this manuscript.

References

1. Chen, K., Calzone, L., Csikasz-Nagy, A., Cross, F., Novak, B., Tyson, J.: Integrative analysis of cell cycle control in budding yeast. Mol. Biol. Cel 5(8), 3841–3862 (2004)

2. Gilbert, D., Heiner, M.: From Petri Nets to Differential Equations - An Integrative Approach for Biochemical Network Analysis. In: Donatelli, S., Thiagarajan, P. (eds.) Petri Nets and Other Models of Concurrency - ICATPN 2006, Lecture Notes in Computer Science, vol. 4024, pp. 181–200. Springer Berlin / Heidelberg (2006)
3. Gillespie, D.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J. Comput. Phys. 22(4), 403 – 434 (1976)
4. Gillespie, D.: Stochastic simulation of chemical kinetics. Annual review of physical chemistry 58(1), 35–55 (2007)
5. Hairer, E., Nørsett, S., Wanner, G.: Solving Ordinary Differential Equations I: Non-stiff Problems, Springer Series in Comput. Mathematics, vol. 8. Springer-Verlag, second edn. (1993)
6. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, Springer Series in Comput. Mathematics, vol. 14. Springer-Verlag, second edn. (1996)
7. Heiner, M., Gilbert, D., Donaldson, R.: Petri nets for systems and synthetic biology. In: Formal Methods for Computational Systems Biology, Lecture Notes in Computer Science, vol. 5016, pp. 215–264. Springer Berlin / Heidelberg (2008)
8. Heiner, M., Herajy, M., Liu, F., Rohr, C., Schwarick, M.: Snoopy – a unifying Petri net tool. In: proc. of 33rd International Conference on Application and Theory of Petri Nets and Concurrency (2012)
9. Heiner, M., Lehrack, S., Gilbert, D., Marwan, W.: Extended stochastic Petri nets for model-based design of wetlab experiments. In: Transactions on Computational Systems Biology XI, pp. 138–163. Springer, Berlin, Heidelberg (2009)
10. Herajy, M., Heiner, M.: Hybrid representation and simulation of stiff biochemical networks. Nonlinear Analysis: Hybrid Systems (2012), to appear
11. Herajy, M., Heiner, M.: Hybrid representation and simulation of stiff biochemical networks through generalised hybrid Petri nets. Tech. Rep. 02/2011, Brandenburg University of Technology Cottbus, Dept. of CS (2011)
12. Kar, S., Baumann, W.T., Paul, M.R., Tyson, J.J.: 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), 6471–6476 (2009)
13. Kiehl, T., Mattheyses, R., Simmons, M.: Hybrid simulation of cellular behavior. Bioinformatics 20, 316–322 (2004)
14. Liu, Z., Pu, Y., Li, F., Shaffer, C., Hoops, S., Tyson, J., Cao, Y.: Hybrid modeling and simulation of stochastic effects on progression through the eukaryotic cell cycle. J. Chem. Phys 136(34105) (2012)
15. Marwan, W., Rohr, C., Heiner, M.: Petri nets in Snoopy: A unifying framework for the graphical display, computational modelling, and simulation of bacterial regulatory networks, Methods in Molecular Biology, vol. 804, chap. 21, pp. 409–437. Humana Press (2012)
16. Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., Miyano, S.: Biopathways representation and simulation on hybrid functional Petri net. In silico biology 3(3) (2003)
17. Mura, I., Csikász-Nagy, A.: Stochastic Petri net extension of a yeast cell cycle model. Journal of Theoretical Biology 254(4), 850 – 860 (2008)
18. Sabouri-Ghomi, M., Ciliberto, A., Kar, S., Novak, B., Tyson, J.J.: Antagonism and bistability in protein interaction networks. Journal of Theoretical Biology 250(1), 209 – 218 (2008)
19. Singhania, R., Sramkoski, R.M., Jacobberger, J.W., Tyson, J.J.: A hybrid model of mammalian cell cycle regulation. PLoS Comput Biol 7(2), e1001077 (02 2011)

20. Steuer, R.: Effects of stochasticity in models of the cell cycle: from quantized cycle times to noise-induced oscillations. Journal of Theoretical Biology 228(3), 293 – 301 (2004)
21. Tyson, J., Novak, B.: Regulation of the eukaryotic cell cycle: Molecular antagonism, hysteresis, and irreversible transitions. Journal of Theoretical Biology 210(2), 249 – 263 (2001)
22. Tyson, J., Novak, B.: A Systems Biology View of the Cell Cycle Control Mechanisms. Elsevier, San Diego, CA, (2011)
23. Valk, R.: Self-modifying nets, a natural extension of Petri nets. In: Proceedings of the Fifth Colloquium on Automata, Languages and Programming. pp. 464–476. Springer-Verlag, London, UK, (1978)