Automatic Parameterization of the Purine Metabolism Pathway through Discrete Event-based Simulation

Simone Caligola[1], Tommaso Carlucci[2], Franco Fummi[1], Carlo Laudanna[2], Gabriela Constantin[2], Nicola Bombieri[1], Rosalba Giugno[1]
[1]*Department of Computer Science*, [2]*Department of Medicine*, *University of Verona*, Verona, Italy

***Abstract*—Stochastic Petri Nets (SPN) are recognized as one of the standard formalisms to model metabolic networks. They allow incorporating randomness in the model and taking into account possible fluctuations and noise due to molecule interactions in the environment. Even though some frameworks have been proposed to implement and simulate SPN (e.g., Snoopy, Monalisa), they do not allow for automatic model parameterization, which is a crucial task to identify the network configurations that lead the model to satisfy certain biological properties. We present a framework to synthesize the SPN model of a metabolic network into executable code that can be simulated through a discrete event-based simulator. The framework allows the user to formally define the network properties to be observed and to automatically extrapolate, through Assertion-based Verification (ABV), the parameter configurations that lead the network to satisfy such properties. We applied the framework to model the purine metabolism and to reproduce the metabolomics data obtained from naive lymphocytes and autoreactive T cells implicated in the induction of experimental autoimmune disorders. We show system parameterization extrapolated by the framework to reproduce the experimental results and to simulate the model under different conditions.**

***Index Terms*—Stochastic Petri Net, Metabolic Networks, Electronic Design Automation, T cells, Autoimmunity.**

INTRODUCTION

Petri nets (PN) are an effective formalism used to model metabolic networks. PN have been widely applied in Systems Biology because they provide an intuitive graphical representation, well-founded mathematical properties (e.g., place/transition invariants) for qualitative analysis, and extensions to perform dynamic system simulation. Place invariants (p-invariants) and transition invariants (t-invariants) were used, for example, to validate important biological processes such as apoptosis [1] and to understand the processes at the basis of the iron homeostasis [2].

PNs have been also adopted to perform quantitative analysis of biological systems. In particular, stochastic Petri nets (SPN) have gained popularity in Systems Biology because, unlike ordinary differential equations (ODEs), they support the inherent uncertainty of the biological processes [3]. For example, SPN have been succefully used to obtain new insights in the development of hepatic granuloma throughout the course of infection [4], and to model signal transduction pathways in the process of angiogenesis [5].

Software applications have been developed to model and simulate biological systems through SPN. The best known are Snoopy [6] and Monalisa [7]. They allow simulating complex biochemical systems including metabolic pathways, signal transduction pathways, and gene expression networks. A recurrent issue of these software applications is related to the concept of *parameterization*. Very often, due to the lack of quantitative information, it is necessary to explore the *solution space* of the parameters to identify which network configurations lead the model to satisfy certain biological properties. This process requires the manual tuning of unknown parameters to obtain model behaviors matching the biological knowledge. Since the solution space to explore grows exponentially with the network size, such a manual parametrization task becomes prohibitive when applied to realistically large networks. In this work we propose a framework that applies languages, techniques, and tools well established in the field of electronic design automation (EDA) to model and simulate metabolic networks. Since biological systems and electronics systems share several characteristics like concurrency, reactivity, abstraction levels as well as issues like reverse engineering and design space exploration [8], we show how the proposed EDA-based framework can introduce automation and flexibility to model, simulate, and to help the analysis of metabolic networks. EDA techniques have been applied to study qualitatively the leukocyte integrin-activation signalling network through the use of boolean networks [9]. However, the study of metabolic networks dynamics requires to take into account the flux of metabolites in a quantitative way. In this paper we present a new framework that, starting from a PN model of a metabolic network, generates executable code that implements SPN to simulate the network dynamics suitable for discrete event-based simulation. The user defines network properties through a formal specification language, which are synthesized and integrated into the network code. The framework then applies Assertion-based Verification (ABV) combined to an automatic parameter generation based on a genetic algorithm to extrapolate the parameter configurations that satisfy the defined network properties.

We applied the framework to model and simulate the purine metabolism and to reproduce the metabolomics data obtained from naive lymphocytes and autoreactive T cells implicated in the induction of experimental autoimmune disorders. We show that the framework automatically extrapolates system parametrizations that reproduce the experimental results and that allow simulating the model under different conditions.

I. BACKGROUND ON STOCHASTIC PETRI NETS

SPN is a class of Petri nets in which at every transition k of the network state is associated a delay τ_k, which is determined by a random variable. Formally, a SPN is a five-tuple $SPN = \{P, T, F, M_0, \Lambda\}$ where P is the set of the places, T the set of transitions, $F \subset (P \times T) \cup (T \times P)$ is the set of relations between places and transitions, M_0 is the initial configuration of the net (marking), and Λ is the

G.C. was supported by the European Research Council (ERC) grants IMMUNOALZHEIMER (nr. 695714, ERC advanced grant) and IMPEDE (nr. 693606, ERC Proof of Concept grant) (G.C.)

set of exponentially distributed firing rates λ_k associated with the transitions. The firing rates are defined trough *propensity functions* (hazard) a_k that have the pre-places of the transition k as domain, and it gives the probability that a reaction will occur in the next infinitesimal time interval.

To simulate biochemical systems, specific types of propensity functions are used, such as *mass-action propensity functions* and *Michelis-Menten propensity functions*. A well-established method to perform a simulation of a SPN is the Gillespie's algorithm, called Stochastic Simulation Algorithm (SSA). This method is a Monte Carlo procedure that calculates a possible trajectory of the system simulating one chemical reaction at each step, and that chooses the firing time τ. Gillespie proposed two equivalent variants of his algorithm: the Direct Method (DM) [10], and the First Reaction Method (FRM) [11]. In general, DM is more efficient in terms of computational time and space while FRM is well suited for a concurrent and parallel implementation [12]. Our methodology is based on the FRM variant.

II. METHODS

A. Discrete Event-based Simulation of Stochastic Petri Nets through an EDA Framework

The proposed methodology simulates the system under analysis through SPN. Our methodology implements an adapted version of the FRM algorithm in which the reaction delay τ_{rk} of each reaction k is based on the following formula:

$$\tau_{rk} = \left\lfloor \left(\frac{1}{a_k(m_i)} \right) \cdot ln \left(\frac{1}{r_k} \right) \right\rfloor + 1 \quad (k = 1, \ldots, m)$$

where $r_1, \ldots, r_m$ are m random numbers from the uniform distribution $U(0,1)$, $a_k(m_i) = c_i m_i$ is the mass-action propensity function where c_i is the reaction rate and m_i the number of tokens of the reactant. We consider a lower-bound value of the delay (i.e., 1), which is associated to the minimum delay in the discrete event-based simulation. The system starts by assigning a delay τ_{rk} to each reaction. At each simulation step, the system executes the reactions with the minimum delay and updates the simulation time and the propensity functions of the reactions involved in the execution. To perform the event-driven simulation of the SPN, we propose a methodology to implement SPNs through the Extended Finite State Machines (EFSMs) [13], which allow the network to be synthesized into executable SystemC code. SystemC [14] is the reference language in EDA to model and simulate complex systems at different abstraction levels. SystemC-based verification is the de-facto alternative to model checking when such a formal verification technique cannot deal with the state space complexity of the model.

Fig. 1(a) shows two basic reactions and their representation through Petri net models.

Fig. 1(b) shows, as a representative example, how the reactions related to the metabolite M_j of the top left example are represented by EFSMs. The activity of a metabolite is defined by its production and consumption behaviour. The production and consumption processes are represented by two concurrent EFSMs, each one based on a two-state machine. In the M_i-M_j reaction, when the consumption process of the preplace M_i is in the `ready` state or its concentration (m_i) has been updated by other reactions, the concentration m_i is

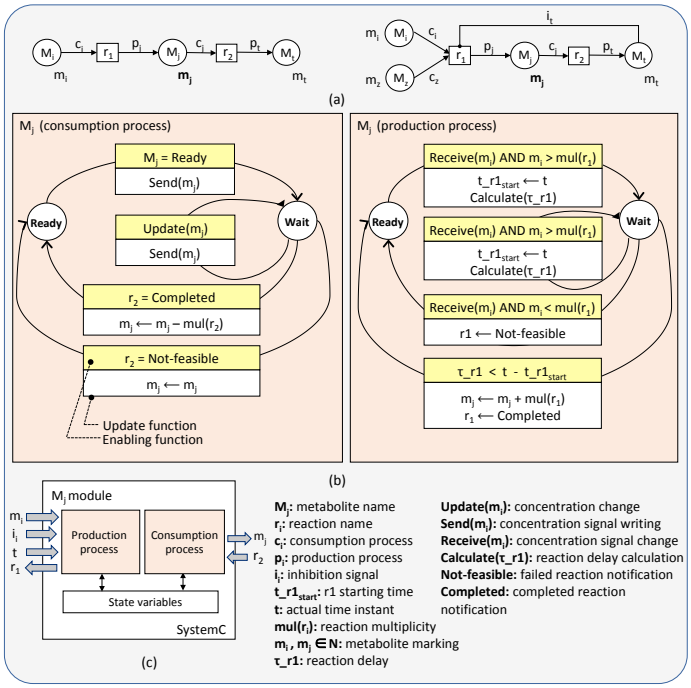

Fig. 1. Examples of basic reactions and their mapping in EFSM and SystemC models: (a) the Petri net models of the reactions, (b) the EFSM representation of the reactions related to M_j, (c) the SystemC template implementing M_j.

sent to M_j and its state changes to `wait`. The production process of M_j wakes up by receiving m_i and, if the reaction is feasible, the model calculates the reaction delay τ_{r1} and saves the current simulation instant (t_r1_{start}). Variable `t` represents the simulated time, and is updated by the SystemC simulation kernel at each simulation step. The transition leads the machine into the `wait` state, which implements the stochastic reaction delay based on mass-action law. The model blocks the process to the `wait` state until the reaction delay expires. However, during the waiting time, M_j may have to update τ_{r1} and t_r1_{start} due to a change in the concentration of the pre-place M_i. When the reaction is performed, the machine moves to the `ready` state, by increasing the M_j concentration by $mul(r_1)$ and sending a notify to the M_i consumption process. Thus, the M_i consumption process decreases m_i and returns to the `ready` state.

Fig. 1(c) shows the SystemC implementation of the starting SPN, which can be automatically synthesized by the corresponding EFSM model.

B. Parameter estimation through Assertion-based verification (ABV)

In EDA, functional verification based on assertions represents one of the main applied and investigated techniques that combines simulation-based (i.e., dynamic) and formal (i.e., static) verification [15], [16]. Assertions are formal descriptions that allow system designers to detect functional errors in the model and in the model evolution over time. They are also combined to techniques of automatic input pattern generation [17] that extrapolate *system configurations* to prove the satisfiability or unsatisfiability of the system properties. The proposed methodology applies simulation-based ABV, by

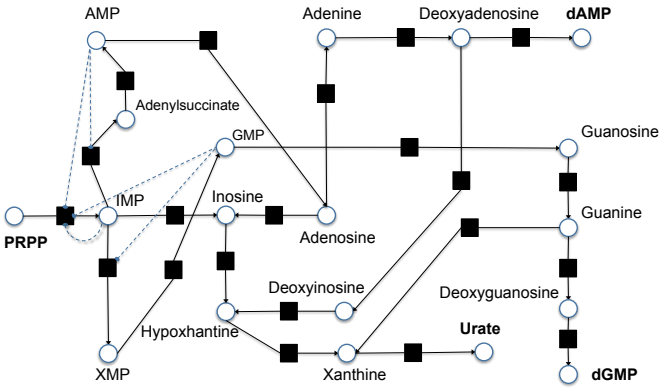

Fig. 2. Petri net model of the purine pathway.

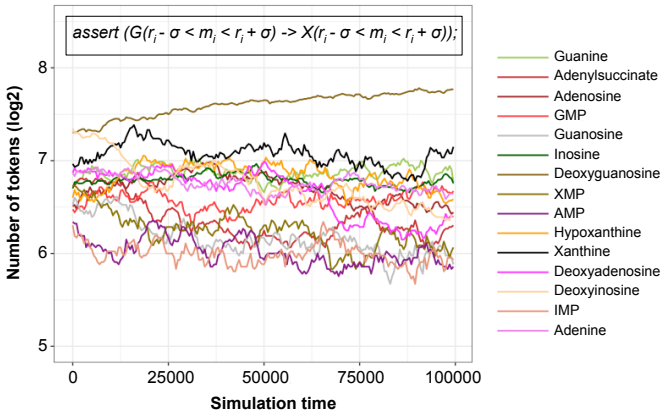

Fig. 3. Example of parameterization of the purine pathway in PLP-specific condition that lead the metabolites to stability within a simulation time of 10^5 clock cycles. At the top, the PSL assertion used to check stability.

which assertions are defined in PSL language [18], they are automatically synthesized into *checkers* [19], and plugged to the SystemC model representing the network. In our context, checkers aim at monitoring the concentration of the metabolites and to give a *score* (i.e., fitness) to an input generation module, which implements a genetic algorithm to generate good configurations of the kinetic parameters. The module generates a configuration of parameters and runs a dynamic simulation of the network for such a set of input values for a given simulation time. Then, the module generates a new different configuration for a new simulation. A proper fitness function evaluates the goodness of each potential solution estimated through simulation. The run ends when the module finds the parameter configuration that allows the system properties to be satisfied. The definition of the fitness function depends on the property to check and it is formulated as the distance between *state vectors* representing the *simulation trend* $s_i(t)$ in comparison to a defined *reference trend* $r_i(t)$ (i.e., the target behaviour of the system). The ABV checks the simulation trend $s_i(t)$ and, eventually, it stops the simulation to provide a score compared to $r_i(t)$. Given the state vectors $S = [s_1, s_2, \ldots, s_n]$ and $R = [r_1, r_2, \ldots, r_n]$ representing, respectively, the simulation and the reference trend, the score is defined as follow:

$$score_c(t) = d(S, R)$$

where d is a distance function (e.g., Euclidean distance). The score of each found parameterization can be used to prioritize configurations selecting those that are closest to the reference trend.

III. Results and Discussion

We applied our methodology to understand how the dynamics of the purine pathway changes between normal and autoreactive conditions. The relative concentrations of the metabolites were determined from metabolomics analysis (http://www.metabolon.com/) in lysates from naive lymphocytes and actively proliferating PLP-specific T cells [20]. The purine pathway was summarized in our Petri net model starting from PRPP transformation to IMP and ending to the production of dAMP, dGMP and urate as waste products of purine synthesis and catabolism (Fig.2). We multiplied the relative concentration of each metabolite by a factor of 10^3 to obtain an integer number of tokens. Due to the lack of experimental data for PRPP, we assigned to it the same concentration of IMP. Here with the term *concentration* we refer to the number of tokens of a metabolite. We simulated our model generating reaction delays in the range $[1-1000]$ of all network reactions, which keep stable the concentrations of each metabolite (i.e., steady state) except dAMP, dGMP and urate (Fig.3). We assumed that the pathway is considered at steady state if the concentration of each element does not differ by more than σ, set to a $\pm 50\%$ from the initial concentration, and it is maintained stable throughout a simulation time of 10^5 simulation cycles. We specified this property formally through ABV. (Fig. 3) In our model the inhibition mechanisms are represented through the inhibition arcs, which is an extension of the classical Petri nets to represent the inhibition of a molecule when its concentration exceeds a certain threshold. We assumed that a metabolite can inhibit a reaction when it grows up by 30% from its initial concentration. Thresholds for steady state and inhibition were chosen empirically by tuning the system to obtain the stability of all network metabolites. The genetic algorithm used by the automatic input generator has been configured with a population of 250 individuals, a mutation probability of 0.05 and a crossover probability of 0.1. We defined the reference trend of the system as follows:

$$r_i(t) = m_i, \forall t > 0, i = 1, 2, ..., n$$

where m_i is the starting concentration of a metabolite and t is the simulation time. The state vector of the simulation trend was defined as $S = [c_1, c_2, \ldots, c_n]$, where the terms c_i are the set of angular coefficients of the linear functions $s_i(t)$, linking the starting and the final concentrations of the metabolites. The state vector of the reference trend was defined as $R = [0, 0, \ldots, 0]$. The fitness function was defined as the inverse of the Euclidean distance between the simulation trend and the reference trend. The selection method used is *rank-based*, taking always individuals with the best fitness for reproduction.

We obtained 10 parameterizations of the purine pathway for each condition. Times for the found parameterizations were on average 9.46 minutes for the network in naive condition and 7.49 minutes for the PLP-specific. Our simulations led to interesting differences in the regulation of the purine pathway,

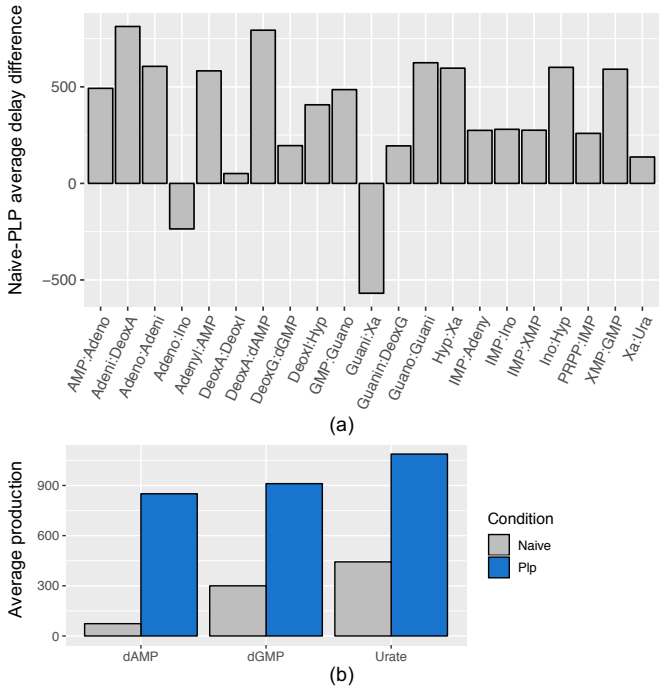

Fig. 4. Results of the analysis of 10 parameterizations of the purine pathway in condition of stability for naive and PLP-specific cells. (a) Average difference of the delays obtained parameterizing the pathway in naive and PLP-specific condition. (b) Difference in the final concentration of the metabolites dAMP, dGMP and urate.

suggesting that most of chemical reactions are highly favored in PLP-specific cells versus naive lymphocytes (Fig. 4(a)). In fact, all metabolic reactions, with the exception of the reactions from Guanine to Xanthine (Guani:Xa) and from Adenosine to Inosine (Adeno:Ino), are speeded up in PLP-specific condition, having a lower average delay time (Fig. 4(a)). Overall, the observed speed-up in the PLP-specific condition resulted in a greater production of the fundamental elements of the pathway dAMP, dGMP and urate (Fig. 4(b)). Notably, the increased urate, dGMP and dAMP production in PLP-specific network reflects our metabolomics data and a well-known metabolic feature of proliferating lymphocytes [21], validating the potentiality of our methodology in simulating metabolic processes. Improvements of the methodology concern the refinement of the fitness function to speed-up the convergence of the genetic algorithm, the implementation of a module for the robustness/sensitivity analysis of the system and the test of our methodology on more complex Petri net models.

IV. Conclusion

This paper presented a methodology to apply languages, techniques, and tools well established in the field of EDA to simulate and automatically parametrize the SPN model of metabolic networks. We applied the methodology to study the purine metabolism pathway starting from metabolomics data obtained from naive lymphocytes and autoreactive T cells implicated in the induction of experimental autoimmune disorders. Thanks to the model parametrization automatically performed by the proposed framework, we were able to simulate the system under different conditions. The simulation results suggest that the entire purine pathway is speeded-up in PLP-specific cells versus naive lymphocytes, according to our experimental data and literature.

References

[1] M. Heiner and I. Koch, "Petri net based model validation in systems biology," in *ICATPN*, vol. 3099. Springer, 2004, pp. 216–237.

[2] A. Sackmann, D. Formanowicz, P. Formanowicz, I. Koch, and J. Blazewicz, "An analysis of the Petri net based model of the human body iron homeostasis process," *Computational Biology and Chemistry*, vol. 31, no. 1, pp. 1–10, 2007.

[3] S. K. Hahl and A. Kremling, "A comparison of deterministic and stochastic modeling approaches for biochemical reaction systems: On fixed points, means, and modes," *Frontiers in genetics*, vol. 7, p. 157, 2016.

[4] L. Albergante, J. Timmis, L. Beattie, and P. M. Kaye, "A petri net model of granulomatous inflammation: implications for il-10 mediated control of leishmania donovani infection," *PLoS computational biology*, vol. 9, no. 11, p. e1003334, 2013.

[5] L. Napione et al., "On the use of stochastic petri nets in the analysis of signal transduction pathways for angiogenesis process," in *International Conference on Computational Methods in Systems Biology*. Springer, 2009, pp. 281–295.

[6] M. Heiner, M. Herajy, F. Liu, C. Rohr, and M. Schwarick, "Snoopy–a unifying petri net tool," in *Int. Conf. on Application and Theory of Petri Nets and Concurrency*. Springer, 2012, pp. 398–407.

[7] P. Balazki, K. Lindauer, J. Einloft, J. Ackermann, and I. Koch, "Monalisa for stochastic simulations of petri net models of biochemical systems," *BMC bioinformatics*, vol. 16, no. 1, p. 215, 2015.

[8] J. Fisher and T. A. Henzinger, "Executable cell biology," *Nature Biotechnology*, vol. 25, pp. 1239 – 1249, 2007.

[9] N. Bombieri, R. Distefano, G. Scardoni, F. Fummi, C. Laudanna, and R. Giugno, "Dynamic modeling and simulation of leukocyte integrin activation through an electronic design automation framework," in *International Conference on Computational Methods in Systems Biology*. Springer, 2014, pp. 143–154.

[10] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *The journal of physical chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.

[11] ——, "A general method for numerically simulating the stochastic time evolution of coupled chemical reactions," *Journal of computational physics*, vol. 22, no. 4, pp. 403–434, 1976.

[12] C. Dittamo and D. Cangelosi, "Optimized parallel implementation of gillespie's first reaction method on graphics processing units," in *Computer Modeling and Simulation, 2009. ICCMS'09. International Conference on*. IEEE, 2009, pp. 156–161.

[13] K.-T. Cheng and A. Krishnakumar, "Automatic generation of functional vectors using the extended finite state machine model," *ACM TODAES*, vol. 1, no. 1, pp. 57–79, 1996.

[14] Accellera Systems Initiative, "IEEE 1666-2011: The standard SystemC language," http://www.systemc.org.

[15] C. N. CoelhoJr. and H. D. Foster, *Assertion-based Verification*. A: Springer, 2008, vol. 4.

[16] N. Bombieri et al., "On the evaluation of transactor-based verification for reusing TLM assertions and testbenches at RTL," in *Proc. of ACM/IEEE DATE*, vol. 1, 2006, pp. 1–6.

[17] M. Boulé and Z. Zilic, *Generating hardware assertion checkers: for hardware verification, emulation, post-fabrication debugging and on-line monitoring*. A: Springer, 2008.

[18] IEEE, "Property specification language - psl," 2017, https://standards.ieee.org/findstds/standard/1850-2010.html.

[19] Y. Abarbanel et al., "Focs–automatic generation of simulation checkers from formal specifications," in *Computer Aided Verification*. Springer, 2000, pp. 538–542.

[20] L. Piccio, B. Rossi, E. Scarpini, C. Laudanna, C. Giagulli, A. C. Issekutz, D. Vestweber, E. C. Butcher, and G. Constantin, "Molecular mechanisms involved in lymphocyte recruitment in inflamed brain microvessels: critical roles for p-selectin glycoprotein ligand-1 and heterotrimeric gi-linked receptors," *The Journal of Immunology*, vol. 168, no. 4, pp. 1940–1949, 2002.

[21] T. Eleftheriadis, G. Pissas, A. Karioti, G. Antoniadi, S. Golfinopoulos, V. Liakopoulos, A. Mamara, M. Speletas, G. Koukoulis, and I. Stefanidis, "Uric acid induces caspase-1 activation, il-1β secretion and p2x7 receptor dependent proliferation in primary human lymphocytes," *Hippokratia*, vol. 17, no. 2, p. 141, 2013.