Overcoming unknown kinetic data for quantitative modelling of biological systems using fuzzy logic and Petri nets

Jure Bordon, Miha Moškon, Miha Mraz

University of Ljubljana,
Faculty of Computer and Information science,
Slovenia,
jure.bordon@fri.uni-lj.si

Abstract. Biological system modelling is used to guide experimental work, therefore reducing the time and cost of *in vivo* implementation of newly designed systems. We introduce an improved modelling method, based on fuzzy logic and Petri nets. By using fuzzy logic to linguistically describe a biological process, we avoid the necessity to use kinetic rates, which are often unknown. We introduce a new set of transition functions to enable the use of our method with existing Continuous Petri nets. With this we achieve the extension of usability and applicability of current Continuous Petri nets definition even for biological systems for which exact kinetic data are unknown. We demonstrate the contribution of our approach by using it to model the translation in a simple transcription-translation system. We compare the results obtained to the results of exiting ODE approaches.

Keywords: modelling biological systems, missing kinetic data, ODE, fuzzy logic, Petri nets

1 Introduction

Advances in synthetic biology are consistently opening new possibilities for the design and construction of complex biological systems. Because *in vivo* design is costly and time-consuming, various modelling methods can be used to check whether the desired behaviour of the system is achievable *in silico* first [1–3]. Furthermore, modelling enables us to test in what way small or substantial changes to the design of our system affects its behaviour and potentially change the design before implementing it *in vivo*. Which modelling method to use depends on the size of the system, the desired accuracy of simulation results and whether accurate kinetic rates, which determine system's dynamics, are known [4]. We usually describe a biological system as a set of chemical species, which are connected by interactions (chemical reactions). Once we define the desired behaviour of our system by choosing the right chemical species and interactions among them, the first step is to build a model. While existing deterministic and stochastic quantitative approaches [5–9] can produce a detailed prediction of system behaviour

M. Heiner (Ed.): BioPPN 2014, a satellite event of PETRI NETS 2014,
CEUR Workshop Proceedings Vol. 1159, 2014.

and therefore reduce the time and cost of such design, they heavily rely on kinetic rates. In synthetic biology biological systems are usually newly designed and the exact details of interactions (kinetic rates) are often unknown [10]. Consequently, existing quantitative approaches can only be used to build a model of a limited set of biological systems [11]. We can use parameter estimation techniques to extract kinetic rates from experimental data. However, due to the complexity of interactions, we often need to establish strict limitations on parameter values in order to get biologically relevant and realistic parameters [12]. The diagram on Figure 1 presents the role of modelling in designing a new biological system. With existing methods the first step of the design process presented on the diagram is only possible when we are building a model with well characterized parts (left side), while our approach can be used for modelling biological systems in the same way even if accurate kinetic data is unknown (right side). Existing methods are often used within the framework of Petri nets, a formalism that has been extended to suit the needs for continuous deterministic and stochastic approaches [13].

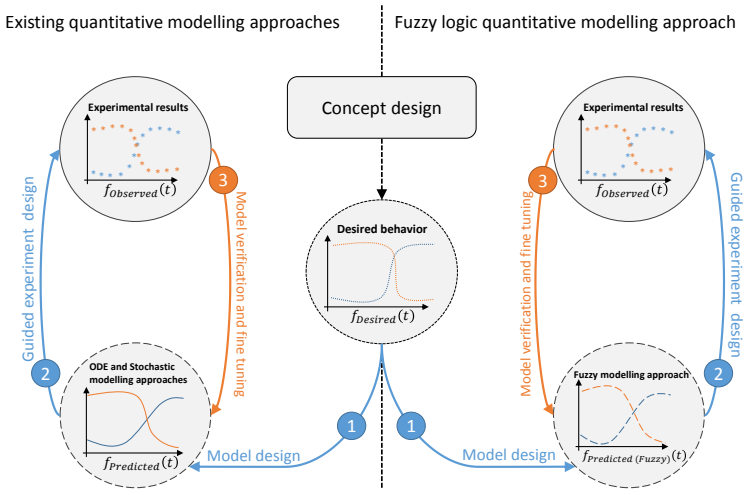

Fig. 1. Sequence of steps which can significantly reduce the time and cost of *in vivo* implementation. Because accurate kinetic data is needed for the first step, existing approaches are often not usable (left side). Proposed modelling method uses the same paradigm for model building (right side), but can be used even when accurate kinetic rates are unknown.

Similarly to quantitative Petri nets, fuzzy logic Petri nets have been established as a very promising modelling approach for qualitative analysis of biological

systems. Fuzzy logic uses linguistic terms and rules for system behaviour description, allowing intuitive design and model construction. It has been applied to several research areas such as: extracting activator/repressor relationship from micro-array data [14, 15], searching for basic motifs in unknown gene regulatory networks (positive/negative feedback loops, degradation, ...) [16] and qualitative description of gene regulation [17]. Additionally, in [18] authors show that fuzzy logic can serve as an alternative but more intuitive approach for modelling biological systems. In their work they apply fuzzy logic and Petri nets to quantitative modelling of biological systems and successfully demonstrate that Hill, Michaelis-Menten and mass-action functions can be approximated by fuzzy logic systems if kinetic data is available. In this paper we propose an improved modelling method that builds on established fuzzy logic and Petri nets approaches but further extend its uses to allow us to obtain quantitative results even when kinetic data is unknown. We inherit existing continuous Petri net definition and extend it to include necessary transition functions for our fuzzy approach. In addition, we can use the proposed method only for parts of the system where kinetic data is unknown, while preserving the accuracy of ODEs in other parts. Because the proposed method is based on linguistic description of the processes, we can use rough estimations and general knowledge about the process to obtain quantitative results. Rough estimations can be obtained by observing existing systems with similar chemical species [19, 20]. Even though we use these estimations and consequently obtain less accurate simulation results, they are still comparable to results of existing methods, are biologically relevant and can be used to guide experimental work.
This paper is organized as follows: in Section 2 we present the basics of fuzzy logic modelling and how fuzzy logic is used in the Petri net framework. In Section 3 we demonstrate the proposed method by constructing a model of translation in a simple transcription-translation system, in Section 4 simulation results obtained with ODE and proposed method are compared and in Section 5 we summarize what the main contribution of the method is and give some directions for future research.

2 Petri Nets as a Framework for Fuzzy Logic

2.1 Fuzzy Logic as a Modelling Approach

Fuzzy logic uses linguistic terms and rules to describe current system state and how the state of the system changes over time [21, 22]. Numerical (crisp) values, which are used for presenting chemical species' concentrations, are converted to fuzzy values. Fuzzy values are given by linguistic terms, presented as membership degree to *fuzzy sets*, such as *Low*, *Medium* and *High*. Conversion from crisp to fuzzy value is performed with *fuzzification* rules, which include the definitions and number of fuzzy sets and the shapes and positions of their membership functions. While a membership function can have arbitrary shape and position, the most commonly used functions are simple triangular [23]. In order to simulate system change and obtain fuzzy value of output variables, IF-THEN rules are

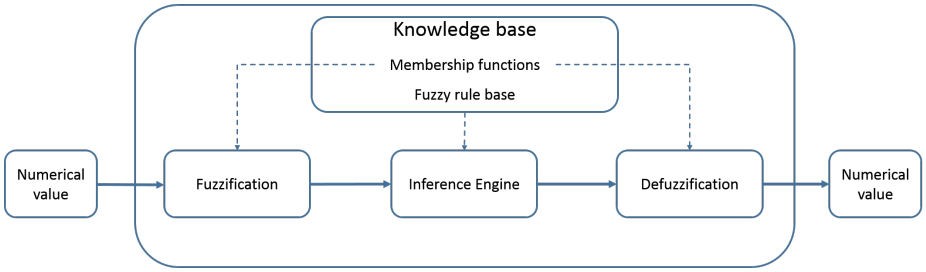

Fig. 2. Fuzzy logic modelling. Input variables are first converted to fuzzy variables by fuzzification. Once we have input fuzzy values, IF-THEN rules are applied to obtain output fuzzy values. Output variable is then defuzzified to obtain the crisp value. This sequence of steps can easily be translated to a Petri net.

applied to input fuzzy variables. Example of such rule is `IF` x `is High THEN` y `is Low`, where x is the input variable and y is the output variable. Since biological processes often have more than one input, we will need to use rules that combine the effect of input variables with operators AND and OR. An example of such rule is `IF` x_1 `is High AND` x_2 `is Low THEN` y `is Low`, where x_1 and x_2 are input variables and y is the output variable. Final step of fuzzy logic reasoning is obtaining crisp value of output variable, which is termed *defuzzification* and is performed by applying center-of-gravity (COG) method. Figure 2 shows these three steps as a sequence of actions: fuzzification, applying IF-THEN rules and defuzzification. Fuzzy logic can be used to intuitively model biological processes. IF-THEN rules are used to describe underlying dynamics where input variables are presented by current concentrations of chemical species and output variables define changes of concentrations. If we augment this description with rough estimation of reaction speed and therefore introduce the component of time, we can obtain quantitative results. In addition, the sequence of three steps can be efficiently used within the Petri net framework [24, 25].

2.2 Fuzzy Logic and Petri Nets

By using Petri net formalism it is possible to intuitively build the Petri net graph of the model. Once the Petri net is constructed using different modelling methods is easy. We only need to change the underlying transition function and firing rules. Continuous Petri nets use real numbers in places (marking values), meaning that transitions also no longer consume and produce whole tokens, but instead change the marking of an input or output place by a real value. New marking values in places are calculated by adding the contribution from input transitions and subtracting the value that is consumed due to output transitions. This allows a continuous flow throughout the Petri net, which can be used to present a system of ODEs [13]. Similar approach is used with the proposed fuzzy logic modelling method. Input and output of fuzzy part is identical to that of existing continuous Petri net [26]. However, when using fuzzy logic, we first

need to fuzzify the input variable (additional transition function) and calculate the membership to each defined fuzzy set. By applying the defined IF-THEN rules (one transition for each rule), we obtain the fuzzy value of output variable and then defuzzify (center-of-gravity transition function) it to obtain the crisp value. We use this crisp value to change the marking of a place the same way we do in continuous Petri nets, by adding or subtracting a real value. We will use existing continuous Petri net definition from [26]. We will add a new set of functions that are needed for fuzzy logic. This set of functions will include fuzzification functions, functions for applying IF-THEN rules and defuzzification function. Existing definition $PN_{Continuous} = \langle P, T, f, v, m_0 \rangle$ is extended by a set of functions $v_{fuzzy} = (f_{fuzzification}, f_{defuzzification}, f_{IF-THEN})$. Functions in $f_{fuzzification}$ define how we obtain fuzzy value from an input crisp value. An example of such function is a triangular membership function for a fuzzy set A:

$$\mu_A(x) = \begin{cases} \frac{x-a}{b-a} & a \leq x \leq b, \\ \frac{c-x}{c-b} & b \leq x \leq c, \\ 0 & otherwise, \end{cases} \tag{1}$$

where x is the crisp value of the input variable and parameters a, b, c the x-coordinates of triangle vertices that define the shape of membership function. Function $f_{defuzzification}$ gives us the opposite rule and defines how we obtain crisp value from fuzzy value by applying the center-of-gravity method (COG).

$$y = \frac{\sum_{i=1}^{n} \overline{y_i} \cdot \mu[i]}{\sum_{i=1}^{n} \mu[i]}, \tag{2}$$

where y is the crisp value, $\overline{y_i}$ x-coordinate at which membership function of fuzzy set i has the highest possible degree of membership (parameter b from Eqn. 1) and $\mu[i]$ current degree of membership for fuzzy set i. Output fuzzy value is obtained by applying IF-THEN rules to the input variables. With basic (one input and one output) IF-THEN rules $f_{IF-THEN}$ is simple. If we have an input variable x, an output variable y and a rule `IF x is Low THEN y is High`, x membership degree to its fuzzy set `Low` is assigned to y membership degree to its fuzzy set `High`. This process is then repeated for all rules to obtain fuzzy value of y. However, biological processes usually have more than one input chemical species, therefore we need to use rules with more than one input variable. When applying IF-THEN rules with more than one input variables we usually define the rules using operator `AND`, which acts as a function $Min(\mu_1[i], \mu_2[i], ..., \mu_n[i])$, where $\mu_j[i]$ is a membership degree of variable j to its fuzzy set i. If we have two input variables x_1, x_2, an output variable y and a rule `IF` x_1 `is Low AND` x_2 `is High THEN` y `is High`, y degree of membership to its fuzzy set `High` would be assigned as a lower of the two values: x_1 degree of membership to its fuzzy set `Low` and x_2 membership degree to its fuzzy set `High`, which we can also note as $\mu_y[High] = Min(\mu_{x_1}[Low], \mu_{x_2}[High])$. Once we define the set of these three types of functions (fuzzification, defuzzification, IF-THEN rules), we have all the tools needed to construct a fuzzy Petri net model of a biological process.

3 Simple Transcription-Translation System: Modelling Translation With Fuzzy Logic and Petri Nets (case study)

We present model construction using proposed method on a simple transcription-translation system introduced in [18] to verify a qualitative modelling technique by qualitatively comparing its results with the results of an ODE approach. This system consists of 5 chemical species: DNA, mRNA, Transcription resource (TsR), Translation resource (TlR) and protein (GFP). The dynamics of the system are governed by transcription (TsR consumption, mRNA production), translation (GFP production) and the decay of mRNA and TlR as shown on Figure 3.

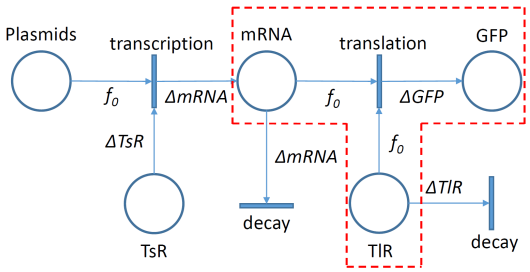

Fig. 3. Petri net of a simple transcription-translation system. We use fuzzy approach on the marked part of the Petri net (translation). Our simulations will observe how concentration of GFP changes over time, if we insert DNA at different time points. The limiting factors for system stability are limited amount of transcription and translation resources: TsR consumption and TlR degradation.

We will adopt the ODE model of this system from [18]. It is defined as the following set of differential equations:

$$\frac{d[mRNA]}{dt} = \frac{k_{ts} \cdot [TsR] \cdot [DNA]}{m_{ts} + [DNA]} - \delta_{mRNA} \cdot [mRNA], \tag{3}$$

$$\frac{d[TsR]}{dt} = -\frac{k_{TsR} \cdot [TsR] \cdot [DNA]}{m_{ts} + [DNA]}, \tag{4}$$

$$\frac{d[GFP]}{dt} = \frac{k_{tl} \cdot [TlR] \cdot [mRNA]}{m_{tl} + [mRNA]} - k_{mat} \cdot [GFP], \tag{5}$$

$$\frac{d[TlR]}{dt} = -\frac{\delta_{TlR} \cdot [TlR]}{m_{TlR} + [TlR]}. \tag{6}$$

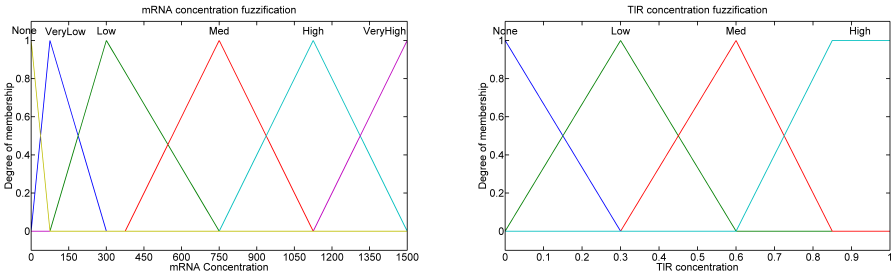

Fig. 4. Membership functions for input variable fuzzy sets: mRNA (left) is described by 6 fuzzy sets - None, VeryLow, Low, Medium, High and VeryHigh - and TlR (right) by 4 fuzzy sets - None, Low, Medium and High.

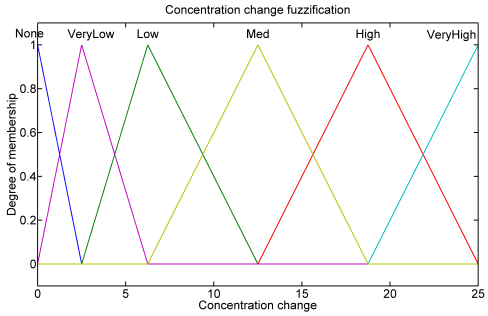

Fig. 5. Membership functions for output variable (concentration change) fuzzy sets - None, VeryLow, Low, Medium, High, VeryHigh.

The ODE model from [18] assumes that concentration of TlR and mRNA do not change as the result of translation. mRNA concentration increases as a result of transcription and only decreases as a result of degradation. Additionally, TlR concentration also only decreases as a result of degradation. To verify the proposed method, we will assume that k_{tl} and/or m_{tl} from Eqn. (5) are unknown when constructing the fuzzy logic model. We evaluate the fuzzy logic approach by constructing a fuzzy Petri net model of translation, replace the translation part of Eqn. (5) with our fuzzy description as shown on Figure 3 and compare the simulation results to the initial ODE model. First step in constructing a fuzzy logic model is to define membership functions for fuzzification and defuzzification of our input variables (concentration of mRNA and TlR) and output variable (concentration change of GFP). Membership functions we use for both input and output variable fuzzy sets are shown on Figures 4 and 5.

According to [27] we assume that mRNA concentration is the strongest factor of maximum translation speed (maximum change in concentration). TlR therefore reaches highest possible contribution before reaching its maximum concentration, while on the other hand even small amounts of mRNA should result in GFP concentration change.

When defining membership functions for output variables, we need to take into account the rough estimation of translation speed. Our rough estimation is based on data from different biological systems, using different chemical species. Considering translation rate, maximum concentration of mRNA and type of chemical species from [18, 19, 27, 28], our rough estimation is that the maximum change in concentration of a protein as a result of translation is $25nM/min$. How input variables affect output variable is defined by the IF-THEN rules shown in Table 1.

Table 1. The set of rules that defines how input variables affect output variable. If either of the input variables is None change in concentration will also be None. In all other cases, increasing both input variable concentration will increase the change in concentration of GFP, reaching highest change when both inputs are at their highest values.

TlR \ *mRNA*	None	VeryLow	Low	Med	High	VeryHigh
None	None	None	None	None	None	None
Low	None	VeryLow	Low	Low	Low	Med
Med	None	VeryLow	Low	Med	Med	High
High	None	VeryLow	Low	Med	High	VeryHigh

IF-THEN rules are defined so they reflect the descriptive knowledge we have about translation. The more there is of either mRNA or TlR, the higher the change in concentration of GFP should be; if one of the inputs is low, change in concentration changes accordingly; if any of the inputs is missing, there should not be any concentration change, etc. Once we obtain the fuzzy value of our concentration change by applying IF-THEN rules, we need to defuzzify it in order to get a crisp value, which we can then use in calculating the new concentration of the GFP. Fuzzy output is translated into a crisp value according to the Eqn. (2). This crisp value is then used just as it would have been if it was a result of a step in numerical solving of system of ODEs. We constructed the Petri net for our fuzzy description of translation as a series of three steps - fuzzification, applying IF-THEN rules and defuzzification (Figure 6). We can use this PN to replace the translation transition from Figure 3, if parameter values for Eqn. (5) are unknown.
Using this constructed Petri net, we will observe how concentration of GFP changes over time and when it reaches its maximum value if we add the DNA at different time points and compare the simulation results to the ODE model.

4 Results

Both ODE and Fuzzy logic models were built in MATLAB Simulink. Petri nets serve as a powerful framework for both approaches, however computing underlying numerical solutions can be done by an external engine like MATLAB. We

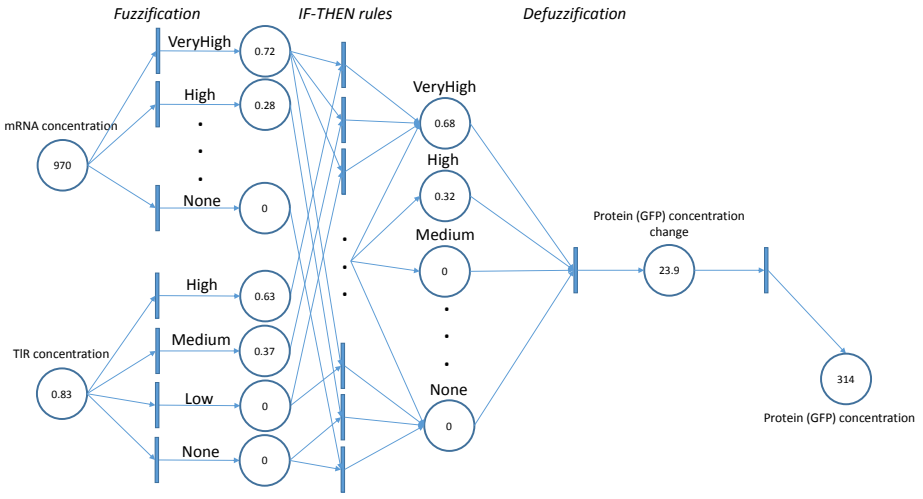

Fig. 6. Translation model using fuzzy logic and Petri nets. Inputs and output of this model are the same as with ODE: mRNA concentration, TlR concentration and change in concentration of the protein (GFP). This Petri net can be inserted into 3 to get the full model of the system. For reactions other than translation ODEs are used (Eqns. (3),(4),(6) and degradation part of Eqn. (5).

used MATLAB Simulink built-in ode4 (Runge-Kutta) solver and set the simulation time to 1000 minutes with a 0.1 minute fixed time step. Initial concentrations of both TsR and TlR were set to 1 nM, all others were set to 0 nM. During the simulation we inserted 3.4 nM of DNA at 6 different time points (six different simulations with same initial concentrations): 0 minutes, 37 minutes, 73 minutes, 112 minutes, 153 minutes and 187 minutes (these concentrations and time points were chosen according to [18] in order to make comparison of simulation results relevant). To avoid discontinuity of ODE solving, the input and output of the fuzzy component is evaluated for every step of the simulation. This slightly increases computation time of the simulation. Figure 7 shows simulation results of two different models.

Simulation results from both models show that the plateau of protein concentration is reached at the same time (at about 200 minutes) which is the result of translation resource degrading to 0, stopping translation entirely. Since we did not include protein degradation, its concentration stays unchanged for the remaining time of simulation. We see that even though we described translation with fuzzy approach we still get comparable quantitative results. The error introduced due to using rough estimation of translation speed instead of exact translation rate is noticeable. However, we did not use any exact parameter values for translation with the proposed method and still managed to obtain quantitatively and biologically relevant results, which are comparable to those obtained with (strict) ODE approach. In addition, because we only changed how

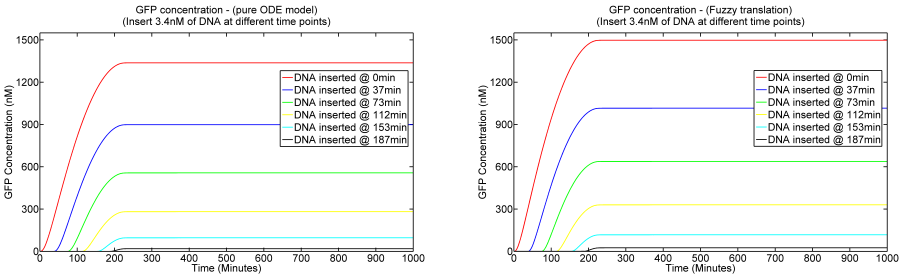

Fig. 7. We observed how GFP concentration changes if we add 3.4 nM of DNA at different points when using strict ODE model (left) and when using the proposed fuzzy approach for modelling translation (right).

we model translation, trajectories for other processes stay unchanged. Simulation results indicate that fuzzy logic is a viable modelling approach even when kinetic data is unknown. By exploiting information we have about the system for similar models and biological systems, we can successfully build a quantitative model even when accurate parameters are unknown. By using our approach with Petri nets, we can easily change the underlying description of a process for which kinetic data is unknown while preserving accuracy of ODEs for the parts of system where it is possible.

5 Summary

We presented the Fuzzy logic approach for modelling biological processes, which avoids using exact kinetic data. Proposed method uses a rough estimation of process dynamic to obtain quantitative simulation results. This estimation is extracted from existing base of knowledge about modelling biological processes by inspecting similar systems and chemical species. With introducing this method to Petri nets we managed to further extend their usability and applicability to continuous approaches, even when kinetic data is unknown. We showed its uses on a simple transcription-translation system by substituting the ODE translation description with the proposed fuzzy approach, achieving quantitatively and biologically relevant results, without using exact kinetic data. Adding additional functions for fuzzification, application of IF-THEN rules and defuzzification increases the complexity of the Petri net model. However, these functions are very simple and can be evaluated the same way that ODEs are. In addition, these three stages of fuzzy logic are repeated for every process for which we use the proposed approach and while we need to manually define fuzzy sets, membership functions and IF-THEN rules, once those are defined we could generate the Fuzzy Petri net automatically. The number of transitions and edges for fuzzification and defuzzification stages are defined by the number of fuzzy sets, while the functions for these transitions are defined by the shape of membership functions. Number of edge and transition functions in IF-THEN rule stage are defined by

IF-THEN rules (e.g. IF x_1 is High AND x_2 is Low THEN y is Low would generate a transition with two input edges - from places $x1_{High}$ and $x2_{Low}$ - and one output edge - to place y_{Low}; the function in the transition would be Min(Input 1, Input 2)). Moreover, we could use hierarchical Petri net structure, where top level would resemble the Petri net shown on Figure 3, while the fuzzification, IF-THEN rules and defuzzification stages (Figure 6) would be presented as a lower level Fuzzy Petri net that describes all three stages as one transition (in our case translation). Our future research also includes using this approach on a more complex system and observe how inaccuracy of our rough estimation changes the overall trajectory of concentrations. We would also like to consider using experimental data for fine tuning our estimations, which would bring the accuracy of simulation results even closer to those of existing methods.

Acknowledgement

Results presented here are in scope of PhD thesis that is being prepared by Jure Bordon, University of Ljubljana, Faculty of Computer and Information science.

References

1. Kitney, R., Freemont, P.: Synthetic biology - the state of play. FEBS Lett **586**(15) (July 2012) 2029–2036
2. Chen, Y.Y., Galloway, K.E., Smolke, C.D.: Synthetic biology: advancing biological frontiers by building synthetic systems. Genome biology **13**(2) (2012) 240
3. Chen, L., Wang, R.: Designing gene regulatory networks with specified functions. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: REGULAR PAPERS **53** (2006) 2444–2450
4. Alon, U.: An Introduction to Systems Biology: Design Principles of Biological Circuits. Chapman and Hall/CRC mathematical & computational biology series. Chapman & Hall/CRC (2007)
5. de Jong, H.: Modeling and simulation of genetic regulatory systems: A literature review. Journal of Computational Biology **9** (2002) 67–103
6. Weiss, J.N.: The hill equation revisited: uses and misuses. FASEB Journal **11** (1997) 835–841
7. Gillespie, D.T.: Stochastic simulation of chemical kinetics. Annu. Rev. Phys. Chem. **58** (2007) 35–55
8. Andrianantoandro, E., Basu, S., Karig, D.K., Weiss, R.: Synthetic biology: new engineering rules for an emerging discipline. Molecular Systems Biology **May** (2006) 1–14
9. Cheng, A., Lu, T.K.: Synthetic biology: An emerging engineering discipline. Annu. Rev. Biomed. Eng. **14** (2012) 155–178
10. Sun, J., Garibaldi, J.M., Hodgman, C.: Parameter estimation using metaheuristics in systems biology: a comprehensive review. Computational Biology and Bioinformatics, IEEE/ACM Transactions on **9**(1) (2012) 185–202
11. Heath, A.P., Kavraki, L.E.: Computational challenges in systems biology. Computer Science Review **3**(1) (2009) 1–17

12. Lillacci, G., Khammash, M.: Parameter estimation and model selection in computational biology. PLoS computational biology **6**(3) (2010) e1000696
13. Heiner, M., Gilbert, D., Donaldson, R.: Petri nets for systems and synthetic biology. Formal methods for computational systems biology (2008) 215–264
14. Hamed, R.I., Ahson, S., Parveen, R.: A new approach for modelling gene regulatory networks using fuzzy Petri nets. Journal of Integrative Bioinformatics **7**(1) (2010) 113
15. Maraziotis, I.A., Dragomir, A., Thanos, D.: Gene regulatory networks modelling using a dynamic evolutionary hybrid. BMC Bioinformatics **11(140)** (2010) 1–17
16. Küffner, R., Petri, T., Windhager, L., Zimmer, R.: Petri nets with fuzzy logic (pnfl): reverse engineering and parametrization. PLoS One **5**(9) (2010) e12807
17. Gendrault, Y., Madec, M., Wlotzko, V., Lallement, C., Haiech, J.: Fuzzy logic, an intermediate description level for design and simulation in synthetic biology. In: Biomedical Circuits and Systems Conference (BioCAS), 2013 IEEE, IEEE (2013) 370–373
18. Windhager, L.: Modeling of dynamic systems with Petri nets and fuzzy logic. PhD thesis, Ludwig-Maximilians-Universität München (April 2013)
19. Tigges, M., Marquez-Lago, T.T., Stelling, J., Fussenegger, M.: A tunable synthetic mammalian oscillator. Nature **457** (2009) 309–312
20. Fung, E., Wong, W.W., Suen, J.K., Bulter, T., Lee, S.g., Liao, J.C.: A synthetic gene–metabolic oscillator. Nature **435**(7038) (2005) 118–122
21. Zadeh, L.: Fuzzy logic and approximate reasoning. Synthese **30**(3) (1975) 407–428
22. Zadeh, L.: Fuzzy logic= computing with words. Fuzzy Systems, IEEE Transactions on **4**(2) (1996) 103–111
23. Klir, G.J., Yuan, B.: Fuzzy sets and fuzzy logic. Prentice Hall New Jersey (1995)
24. Pedrycz, W., Gomide, F.: A generalized Fuzzy Petri net model. Fuzzy Systems, IEEE Transactions on **2**(4) (1994) 295–301
25. Hamed, R.: Intelligent method of Petri net formal computational modeling of biological networks. In: Computer Science and Electronic Engineering Conference (CEEC), 2013 5th. (Sept 2013) 162–167
26. 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. Volume 4024 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2006) 181–200
27. Atkinson, M.R., Savageau, M.A., Myers, J.T., Ninfa, A.J.: Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in¡ i¿ escherichia coli¡/i¿. Cell **113**(5) (2003) 597–607
28. Michael B. Elowitz, S.L.: A synthetic oscillatory network of transcriptional regulators. Nature **403** (2000) 335–338