

# Comparison of different mathematical models of an *E. coli* fed-batch cultivation process using a generalized net model

Dimitar Dimitrov<sup>1,2</sup> and Olympia Roeva<sup>2</sup>

<sup>1</sup> Faculty of Mathematics and Informatics, Sofia University  
5 James Bourchier Str., Sofia, Bulgaria  
e-mail: dgdimitrov@fmi.uni-sofia.bg

<sup>2</sup> Department of Bioinformatics and Mathematical Modelling  
Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences  
105 Acad G. Bonchev Str., 1113 Sofia, Bulgaria  
e-mail: olympia@biomed.bas.bg

**Abstract:** In this paper, a generalized net model for comparison of different mathematical models of an *E. coli* fed-batch cultivation process is developed. The herewith proposed GN model is simulated using GN Lite – a software package for modeling and simulation with GNs. The GN model compares the set of mathematical models of a considered cultivation process and chooses, based on a predefined criterion, the best process model. During the simulation, the GN model calls the Matlab software environment to solve the process model, presented as a system of non-linear differential equations. In the end of the simulation, the software package plots the main process variables' dynamics with time.

**Keywords:** Generalized net, GN Lite, GNTicker, GN IDE, *E. coli*, Fed-batch cultivation, Mathematical models.

**AMS Classification:** 68Q85.

## 1 Introduction

Cultivation of recombinant micro-organisms e.g. *E. coli*, in many cases is the only economical way to produce pharmaceutic biochemicals such as interleukins, insulin, interferons, enzymes and growth factors. To maximize the volumetric productivities of bacterial cultures it is important to grow *E. coli* to high cell concentration. This sets the challenge to model and optimize fed-batch cultivation processes.

Model parameter estimation problem is clearly about making many decisions often under uncertainty and with multiple conflicting criteria. In order to solve such a problem a variety of different optimization methods are available. As each optimization method has different properties suited for different type of problems, there is no simple answer to which method to use for a particular problem. It is all a matter of opinion; very much depending on the nature of the problem and the availability of different optimization software that fits the problem statement.

In this paper the apparatus of Generalized nets (GNs) is used to describe a GN model for comparison of different mathematical models of an *E. coli* fed-batch cultivation process.

Up to now GNs are used as a tool for modelling of parallel processes in several areas [1, 2, 3, 4, 5] - economics, transport, medicine, computer technologies etc. The theory of the GNs [2, 3, 5] proved to be quite successful when applied to the description of the functioning of expert systems, machine learning and different technological processes. Recently attempts have been made to describe various models, optimization procedures, control loops considering bioprocesses, based on GNs [12, 13, 14, 17].

Proposed here GN model is simulated using GN Lite - a software package for modeling and simulation with GNs [19]. GN Lite implements the current state of the most aspects of GN theory, compared to other software tools for GNs [19]. Its graphical environment has user-friendly interface which assists the user through the whole modeling process [6, 7]. Some of the features of GN Lite are the following:

- implements the algorithm for GN functioning;
- defines an XML format for describing GN models;
- provides a programming language for predicates and characteristic functions in GNs;
- supports integration of MatLab code into GN models;
- WYSIWYG editor of GN models;
- visual simulation of GN models.

## 2 Description of the simulation software

GN Lite is a software product for working with GNs. It is built up in component oriented style. The core of GN Lite is GNTicker [18] - a software interpreter of GN models. It implements most aspects of GN theory, compared to other software tools for GNs [8]. Optimized versions of the algorithms for transition and GN functioning are used in GNTicker [18].

GNTicker is the first GN-related software that uses XML format (XGN) for GN model definitions. This approach has many advantages compared to previous ones. XML ensures platform independence so developers of new GN applications are not limited to use particular operating system or programming language. XML is also popular - plenty of free software tools are available for XML manipulation.

GNTicker provides procedural, weakly-typed language called GNTicker Characteristic Function Language (GNTCFL). It can be used for transition predicates, characteristic functions, token merging rules, as well as additional utility functions. A number of primitive functions are implemented.

GNTCFL code can call functions written for MatLab [9]. This allows complex computations, such as differential equations solving, to be performed during GN simulation without the need for manual GNTCFL implementation of corresponding algorithms.

GN Lite has client-server architecture. Client applications such as graphical environments communicate with the simulation core via TCP/IP. The GNTicker Trace Protocol (GNTP) is used.

The second main component of GNLite is GN IDE, a visual modeling tool for GNs. Its graphical user interface provides facilities for visual constructing of GN models, running simulations, tracing GN execution, exporting models to various file formats, etc. GN IDE uses XGN files for storing GN models. GN IDE is written in Java. It can be run on any platform with Java Runtime Edition (JRE) installed, as well as in browser using Java Web Start. A screenshot of GN IDE is shown in Figure 1.

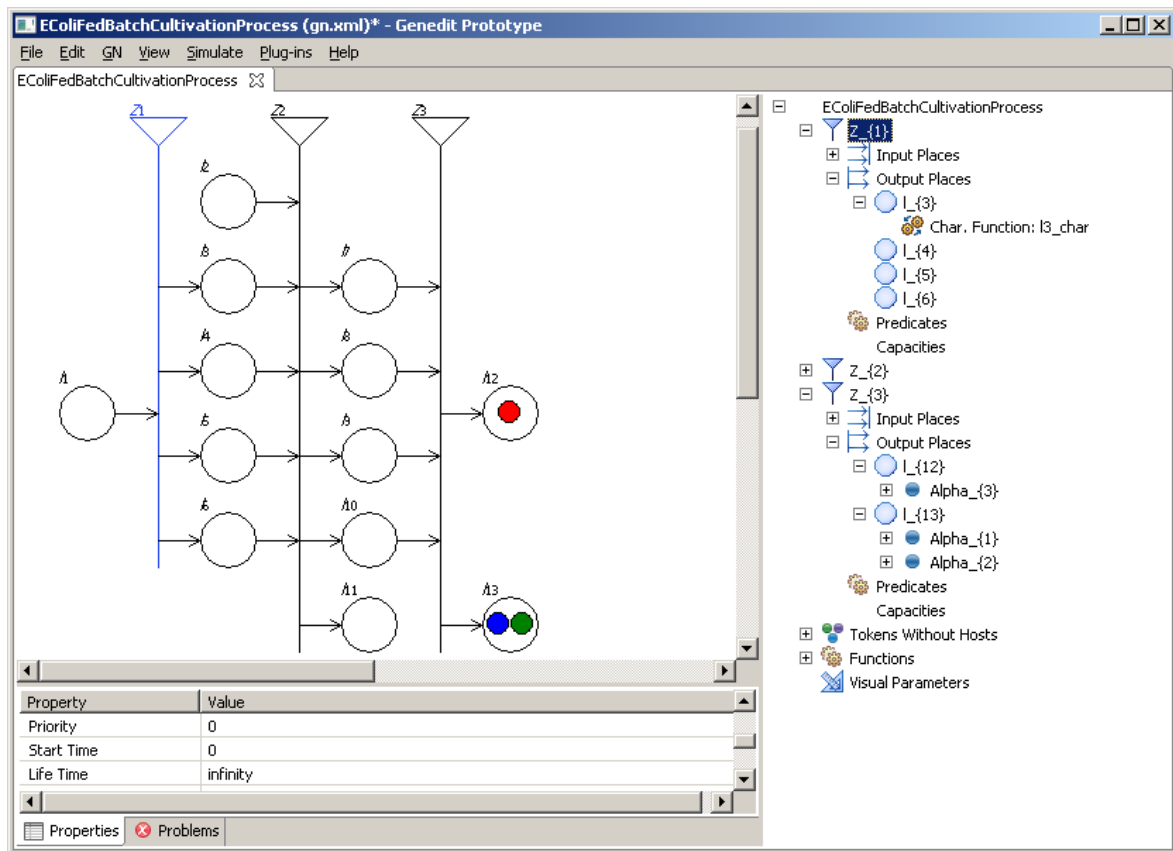


Figure 1: Screenshot of GN IDE

Thanks to the extensible, component-oriented design, new GN-related applications can easily be developed, using the already available components. Some examples are GNProfy and GN-vis [19]. New functionality can easily be implemented over the top of the existing one without the need to be written from scratch. Even main components such as the interpreter and the graphical environment can be replaced with third-party products.

For now, GN Lite is tested on Windows (both 32 and 64 bit versions) and Ubuntu Linux operating systems.

### 3 *E. coli* Fed-batch cultivation process model

The mathematical model, based on the mass balance of the components (biomass and glucose), is presented by the following differential equations [15]:

$$\frac{dX}{dt} = \mu_{max} \frac{S}{k_S + S} X - \frac{F_{in}}{V} X \quad (1)$$

$$\frac{dS}{dt} = -\frac{1}{Y_{S/X}} \mu_{max} \frac{S}{k_S + S} X + \frac{F_{in}}{V} (S_{in} - S) \quad (2)$$

$$\frac{dV}{dt} = F_{in} \quad (3)$$

where:  $X$  is biomass concentration, [g/l];  $S$  – substrate concentration, [g/l];  $F_{in}$  – feeding rate, [l/h];  $V$  – bioreactor volume, [l];  $S_{in}$  – substrate concentration in the feeding solution, [g/l];  $\mu_{max}$  – maximum value of the specific growth rate, [ $h^{-1}$ ];  $k_S$  – saturation constant, [g/l];  $Y_{S/X}$  – yield coefficient, [-].

Using real experimental data of the *E. coli* MC4110 fed-batch cultivation process the model parameter estimation problem have been solved applying different optimization techniques (Genetic algorithm, Tabu search, Firefly algorithm) [11, 15, 16]. The range of the model parameters is considered as follows:  $\mu_{max} \in [0, 0.9]$ ,  $k_S \in [0, 2]$ ,  $Y_{S/X} \in [0, 1]$ .

Initial process parameters are presented in Table 1. The resulting mathematical models have the following parameters values:

- Genetic algorithm:  $\mu_{max} = 0.52 \text{ h}^{-1}$ ,  $k_S = 0.023 \text{ g/l}$ ,  $Y_{S/X} = 0.5$  [15];
- Tabu search:  $\mu_{max} = 0.49 \text{ h}^{-1}$ ,  $k_S = 0.03 \text{ g/l}$ ,  $Y_{S/X} = 0.5$  [11];
- Firefly algorithm:  $\mu_{max} = 0.49 \text{ h}^{-1}$ ,  $k_S = 0.02 \text{ g/l}$ ,  $Y_{S/X} = 0.49$  [16];
- Test model:  $\mu_{max} = 2.4 \text{ h}^{-1}$ ,  $k_S = 6 \text{ g/l}$ ,  $Y_{S/X} = 5$ .

The fourth set of model parameters are defined in a way to test if considered parameters are in a defined range.

Table 1: Initial process parameters

| Process parameters | $t_0$  | $X(t_0)$ | $S(t_0)$ | $V(t_0)$ | $S_{in}$ |
|--------------------|--------|----------|----------|----------|----------|
| Value              | 6.68 h | 1.25 g/l | 0.8 g/l  | 1.35 l   | 100 g/l  |

The model choice is done based on the evaluation of a distance measure  $J$  between experimental and model predicted values of state variables, represented by the vector  $\mathbf{y}$ :

$$J = \sum_{i=1}^n \sum_{j=1}^m \left\{ [\mathbf{y}_d(i) - \mathbf{y}_{mod}(i)]_j \right\}^2 \rightarrow \min \quad (4)$$

where  $n$  is the number of data for each state variable;  $m$  – the number of state variables;  $\mathbf{y}_d$  – the known experimental data;  $\mathbf{y}_{mod}$  – model predictions with a given set of the parameters.

## 4 The generalized net model

In Figure 2, the graphical structure of the constructed GN is shown. Its formal definition is  $E = \langle \langle \{Z_1, Z_2, Z_3\}, *, *, *, *, *, * \rangle, \langle \{\alpha, \beta\}, *, * \rangle, \langle \{x_0^\alpha, x_0^\beta\}, *, * \rangle, \Phi, * \rangle$ , where the three transitions are described consequently, as follows.

$$Z_1 = \langle \{l_1\}, \{l_3, l_4, l_5, l_6\}, *, *, r_{Z_1}, *, \square_{Z_1} \rangle$$

where  $\square_{Z_1} = l_1$  and  $r_{Z_1}$  is the following index matrix:

$$r_{Z_1} = \begin{array}{c|cccc} & l_3 & l_4 & l_5 & l_6 \\ \hline l_1 & true & true & true & true \end{array}$$

$$Z_2 = \langle \{l_2, l_3, l_4, l_5, l_6\}, \{l_7, l_8, l_9, l_{10}, l_{11}\}, *, *, r_{Z_2}, *, \square_{Z_2} \rangle$$

where  $r_{Z_2}$  is the following index matrix of predicates:

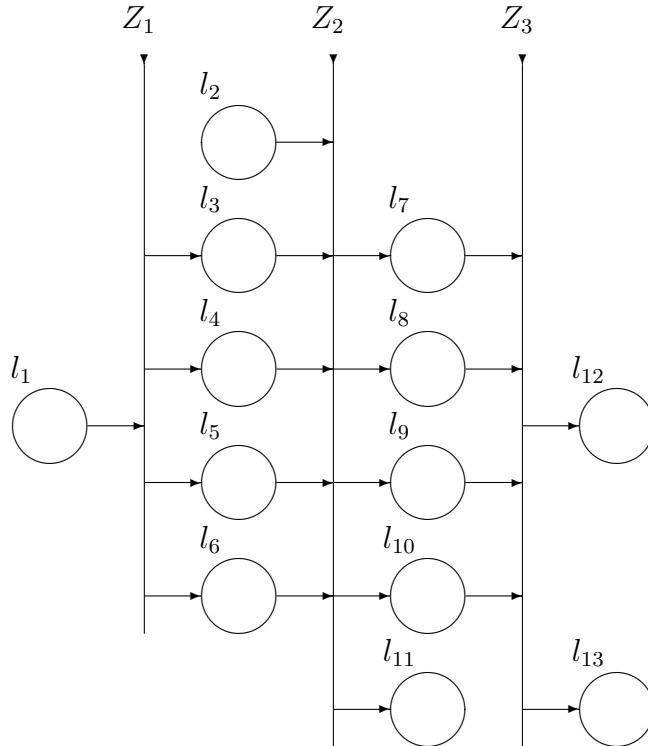
$$r_{Z_2} = \begin{array}{c|ccccc} & l_7 & l_8 & l_9 & l_{10} & l_{11} \\ \hline l_2 & check(l_3) & check(l_4) & check(l_5) & check(l_6) & false \\ l_3 & check(l_3) & false & false & false & \neg check(l_3) \\ l_4 & false & check(l_4) & false & false & \neg check(l_4) \\ l_5 & false & false & check(l_5) & false & \neg check(l_5) \\ l_6 & false & false & false & check(l_6) & \neg check(l_6) \end{array}$$


Figure 2: Graphical structure of the GN model

where  $check(\langle place \rangle)$  is a predicate that checks if the parameters in the token at the given place are correct, as described in the previous section, while  $\square_{Z_2} = \wedge(l_2, l_3, l_4, l_5, l_6)$ .

$$Z_3 = \langle \{l_7, l_8, l_9, l_{10}\}, \{l_{12}, l_{13}\}, *, *, r_{Z_3}, *, \square_{Z_3} \rangle$$

where  $r_{Z_3}$  is the following index matrix:

$$r_{Z_3} = \begin{array}{c|cc} & l_{12} & l_{13} \\ \hline l_7 & isMinError(l_7) & \neg isMinError(l_7) \\ l_8 & isMinError(l_8) & \neg isMinError(l_8) \\ l_9 & isMinError(l_9) & \neg isMinError(l_9) \\ l_{10} & isMinError(l_{10}) & \neg isMinError(l_{10}) \end{array}$$

where  $isMinError(\langle place \rangle)$  is a predicate that checks whether the error characteristic of the token at the given place is minimal, and  $\square_{Z_3} = \vee(l_7, l_8, l_9, l_{10})$ .

Token  $\alpha$  enters place  $l_1$  with four sets of parameter values as initial characteristic. Token  $\beta$  enters place  $l_2$  with values for  $So$ ,  $F$ ,  $t$ ,  $V_0$ ,  $X_d$ , and  $S_d$ . Characteristic function  $\Phi$  is defined as follows:

After the token  $\alpha$  is split into four tokens,  $\Phi_{l_i}, 3 \leq i \leq 6$  keeps only  $(i - 2)$ -th set of parameter values. Token  $\beta$  is split into four tokens too.  $\Phi_{l_i}, 7 \leq i \leq 10$  merges  $\alpha$  and  $\beta$  tokens in its corresponding position. Then it calls a MatLab function named *simul* with the following declaration:

$$function[J, X_{mod}, S_{mod}] = simul(t, \mu_{max}, K_s, K_1, So, V_0, F, X_d, S_d)$$

Token splitting and merging must be enabled for the net, e.g. operator  $D_{2,2}$  must be defined for  $E$ .

## 5 Simulation results

The simulation results presented here are performed on Windows 32-bit operating system using GN Lite described already in Section 2.

A token  $\alpha$  enters the GN model with its characteristics being the four mathematical models, presented in Section 3.

In the first transition, the token  $\alpha$  is split into four new differently colored tokens corresponding to each model. Main initial *E. coli* fed-batch cultivation process characteristics [14] are defined in token  $\beta$  in place  $l_2$ .

In the next transition,  $Z_2$ , the considered models are checked for correctness according to the parameter intervals defined. Models that have passed the check are moved to the corresponding output place ( $l_7, l_8, l_9, l_{10}$ ). The rest leave the net through place  $l_{11}$ . In this case, the first three models (see Section 3) have passed the check and the fourth model leaves the net through place  $l_{11}$ .

Token  $\beta$  is split for each output place corresponding to a valid model. After transition  $Z_2$ , the tokens in places  $l_7$  to  $l_{10}$  are merged.

For each token in  $l_i$ ,  $7 \leq i \leq 10$ , the characteristic function  $\Phi$  calls the MatLab function *simul*. In MatLab environment, the defined nonlinear system of differential equations is solved for each set of model parameters. The function returns three values:  $J$ ,  $X_{mod}$  and  $S_{mod}$ , and adds them to the characteristics of the corresponding tokens.

According to the  $J$  values, transition  $Z_3$  chooses the token with minimal error and passes it to place  $l_{12}$ . The rest tokens are moved to  $l_{13}$ . The token in  $l_{12}$  is the mathematical model of the considered process which exhibits the highest degree of accuracy. In this case, this is the first model (obtained using Genetic algorithms) described in Section 3:  $\mu_{max} = 0.52 \text{ h}^{-1}$ ,  $k_S = 0.023 \text{ g/l}$ ,  $Y_{S/X} = 0.5$ , [15].

GN IDE allows us to plot chart diagrams. In Figure 3 the dynamics of the main process variables (biomass and substrate) are shown. Real experimental data are compared with the model predicted ones. The presented graphical results show a very good correlation between the real and the modelled process.

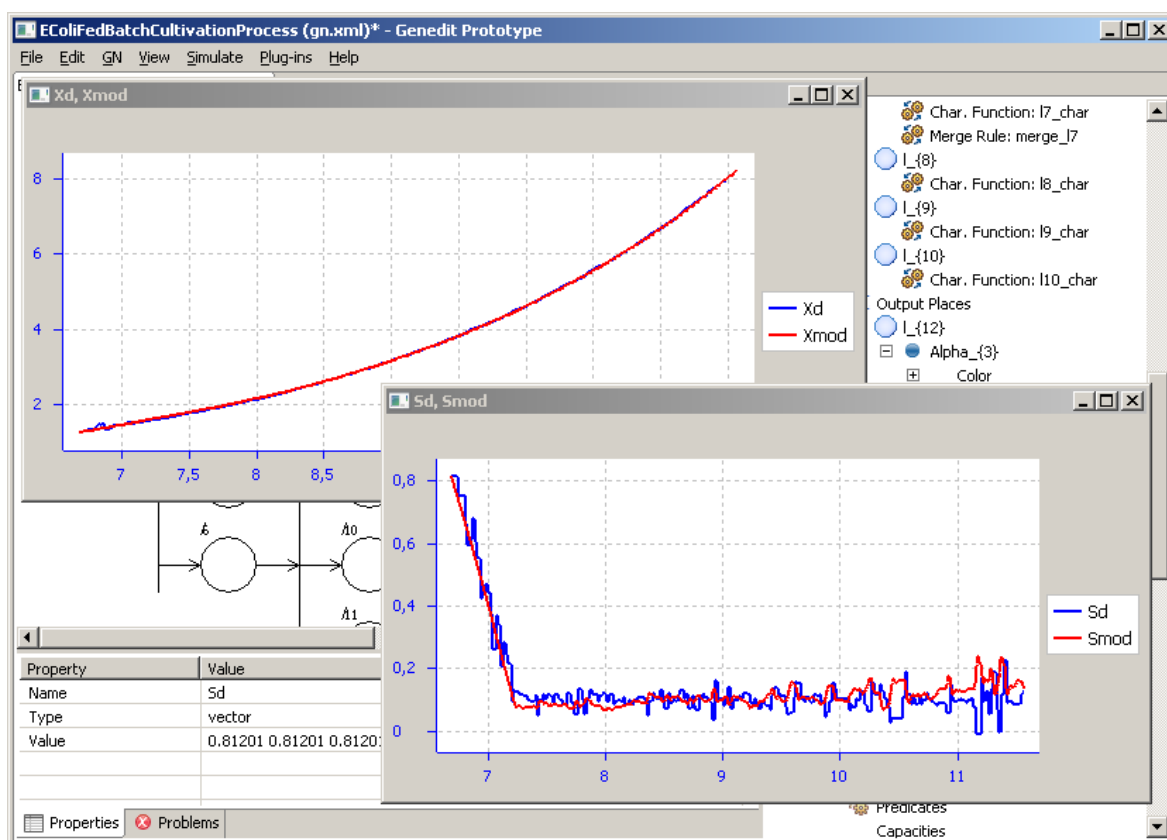


Figure 3: Simulation results

## 6 Conclusion

A generalized net model for comparison of different mathematical models of an *E. coli* fed-batch cultivation process, is here developed. The mathematical model is presented by a system of ordinary differential equations, describing the considered cultivation process variables – biomass and

substrate. For comparison of the models, a set of resulting mathematical models parameters, applying different optimization techniques, is used. The presented GN model compares the models obtained by a Genetic algorithm, Tabu search algorithm and Firefly algorithm, and chooses, based on a predefined criterion, the best process model. The herewith proposed GN model is simulated using GN Lite – a software package for modelling and simulation with generalized nets. GN Lite implements the current state of the GN theory, and its graphical environment has user-friendly interface which assists the user through the whole modelling process.

During simulation, the GN model calls the Matlab software environment to solve the processed model. In the end of the simulation, the software package plots the dynamics of the main process variables of the model of highest degree of accuracy.

The simulation of the presented GN model works out on Windows (both 32-and 64-bit versions) and Ubuntu Linux operating systems.

The developed GN model can be considered as a separate module or as a part of one GN model that describes a set of optimization algorithms, solving the model parameters optimization problem, and then compares the obtained mathematical models and chooses the best one.

## Acknowledgements

This work is supported by the National Science Fund Grant DID 02/29 “Modelling of Processes with Fixed Development Rules (ModProFix)”.

## References

- [1] Aladjov, H., K. Atanassov. A Generalized Net for Genetic Algorithms Learning, *Proc. of the XXX Spring Conference of the Union of Bulgarian Mathematicians*, Borovets, Bulgaria, April 8-11 2001, 242–248.
- [2] Atanassov, K. *Generalized Nets*, World Scientific, Singapore, 1991.
- [3] Atanassov, K. *Generalized Nets and Systems Theory*, “Prof. M. Drinov” Academic Publishing House, Sofia, 1997.
- [4] Atanassov, K., H. Aladjov. *Generalized Nets in Artificial Intelligence, Vol. 2: Generalized Nets and Machine Learning*, “Prof. M. Drinov” Academic Publishing House, Sofia, 2000.
- [5] Atanassov, K. *On Generalized Nets Theory*, “Prof. M. Drinov” Academic Publishing House, Sofia, 2007.
- [6] Dimitrov, D. G. GN IDE - A Software Tool for Simulation with Generalized Nets, *Proc. of Tenth Int. Workshop on Generalized Nets*, Sofia, 5 December 2009, 70–75.
- [7] Dimitrov, D. G. A Graphical Environment for Modeling and Simulation with Generalized Nets. *Annual of “Informatics” Section, Union of Scientists in Bulgaria*, Vol. 3, 2010, 51–66 (In Bulgarian).



- [8] Dimitrov, D. G. Software Products Implementing Generalized Nets. *Annual of "Informatics" Section*, Union of Scientists in Bulgaria, Vol. 3, 2010, 37–50 (In Bulgarian).
- [9] Dimitrov, D. G. Integration of the Generalized Nets Simulator GNTicker with MATLAB. *Annual of "Informatics" Section*, Union of Scientists in Bulgaria, Vol. 4, 2011, 49–52 (In Bulgarian).
- [10] Georgieva, O., I. Hristozov, T. Pencheva, St. Tzonkov, B. Hitzmann. Mathematical Modelling and Variable Structure Control Systems for Fed-batch Fermentation of *Escherichia coli*, *Chemical and Biochemical Engineering Quarterly*, Vol. 17, 2003, No. 4, 293–299.
- [11] Kosev, K., T. Trenkova, Roeva O. Tabu Search for Parameter Identification of an Fermentation Process Model, *Journal of International Scientific Publication: Materials, Methods & Technologies*, Vol. 6, 2012, No. 2, 457–464.
- [12] Roeva, O. Generalized Net Model of Oxygen Control System using Intuitionistic Fuzzy Logic, *Proceedings of the 1st International Workshop on Intuitionistic Fuzzy Sets, Generalized Nets and Knowledge Engineering*, University of Westminster, London, UK, 6-7 September 2006, 49–55.
- [13] Roeva, O., T. Pencheva. Generalized Net for Control of Temperature in Fermentation Processes, *Issues in Intuitionistic Fuzzy Sets and Generalized Nets*, Wydawnictwo WSISiZ, Warszawa, Vol. 4, 2007, 49–58.
- [14] Roeva, O., T. Pencheva, St. Tzonkov. Generalized Net for Proportional-Integral-Derivative Controller, In Book Series: "*Challenging Problems of Sciences*" - *Computer Sciences*, 2008, 241–247.
- [15] Roeva, O. Parameter Estimation of a Monod-type Model based on Genetic Algorithms and Sensitivity Analysis, *Lecture Notes on Computer Science*, Springer-Verlag Berlin Heidelberg, Vol. 4818, 601–608.
- [16] Roeva, O. Optimization of *E. coli* Cultivation Model Parameters Using Firefly Algorithm, *Int. J. Bioautomation*, Vol. 16, 2012, No. 1, 23–32.
- [17] Shannon, A., O. Roeva, T. Pencheva, K. Atanasov. *Generalized Nets Modelling of Biotechnological Processes*, "Prof. M. Drinov" Academic Publishing House, Sofia, 2004.
- [18] Trifonov, T., K. Georgiev. GNTicker - A software tool for efficient interpretation of generalized net models, *Issues in Intuitionistic Fuzzy Sets and Generalized Nets*, Vol. 3, 2005, 71–78.
- [19] Trifonov, T., K. Georgiev, K. Atanasov. Software for modelling with Generalised Nets, *Issues in Intuitionistic Fuzzy Sets and Generalized Nets*, Vol. 6, 2008, 36–42.