Mini Review

BIOMEDICAL Journal of Scientific & Technical Research

ISSN: 2574 -1241

DOI: 10.26717/BJSTR.2025.60.009449

Simulation of Biometric Identification System by Colored Petri Nets

Petrosyan Goharik[1]*, Avetisyan Andrey[2] and Burmin Valery[3]

[1]*Senior Researcher of laboratory 302, Institute of Physics of the Earth named after. O.Yu. Schmidt, RAS, Moscow, Russian Federation, Russia*

[2]*Leading Researcher, Institute of Geophysics and Engineering Seismology named after Academician A. Nazarov NAS RA, Advisor to the Director, Lecturer, Gyumri branch of the Armenian State University of Economics Gyumri, Republic of Armenia*

[3]*Chief Researcher of laboratory 302, Institute of Physics of the Earth named after. O.Yu. Schmidt, RAS Moscow, Russian Federation, Russia*

***Corresponding author:** Petrosyan Goharik, Senior Researcher of laboratory 302, Institute of Physics of the Earth named after. O.Yu. Schmidt, RAS, Moscow, Russian Federation, Russia

ARTICLE INFO

Received: January 22, 2025

Published: January 30, 2025

Citation: Petrosyan Goharik, Avetisyan Andrey and Burmin Valery. Simulation of Biometric Identification System by Colored Petri Nets. Biomed J Sci & Tech Res 60(3)-2025. BJSTR. MS.ID.009449.

ABSTRACT

The article discusses some issues of modeling and digitalization of real-time systems using Petri nets. A system for Identification and Secret-key Generation using a Colored Petri Net, which is a modern extension of the Classical Petri Net, is modeled. The main goal of modeling a system for Identification and Secret-key Generation System using a Colored Petri Net is to identify the presence of errors and accidents, operating behavior, and the effectiveness of the model. The graph of the Colored Petri Net (CPN) of the system for Identification and Secret-key Generation System describes the workflow and logical actions much more simply, since Petri nets are convenient mechanisms for modeling, checking, and validating complex systems.

Keywords: Petri Net; Colored Petri Net; Token, Identification; Secret-Key Generation System

Introduction

Petri nets as graphical and mathematical tools provide a uniform environment for modeling, formal analysis and design of discrete system events [1-4]. One of the main advantages of using Petri net models is that the same model is used both for analyzing behavioral properties and assessing efficiency, and for systematically constructing discrete event simulators. Petri nets and their extensions can be used to model and optimize the flow of materials, resources and activities in various fields such as manufacturing, transport, logistics, etc. [5] Petri nets were named after Carl A. Petri, who created a network mathematical tool for studying connections with automata in 1962 [1-3]. A Petri net (also known as a place/transition net or P/T net) is one of several mathematical modeling languages for describing discrete distributed systems. The further development of Petri nets was facilitated by the fact that they have properties that can be used to model both process synchronization and asynchronous events, parallel operations and conflicts, or resource sharing [1,6]. These properties are used, for example, in industrial automated systems, communication systems, computer systems, nuclear power plants, and aviation systems [6]. Petri nets are used to design the architecture of Nokia mobile phones, in banking operations, interconnections at Hewlett Packard, in a word, they are used in real-time systems. Using Petri nets, it is possible to design universal or logical gates that are firmly integrated into the logic of the digital life of society [6].

Classical Petri nets have limited properties from the point of view of modeling complex systems. In 1996, a group at Aarhus University led by Professor Jensen Kurt created a theory and software (CPN-Tools) for working with CPN [5,7-11]. Colored Petri nets (CPNs) are a graphically oriented language for the design, specification, simulation and verification of systems. It is particularly well suited for systems

 DOI: 10.26717/BJSTR.2025.60.009449

consisting of a number of processes that communicate and synchronize. Typical examples of application areas are communication protocols, distributed systems, automated manufacturing systems, workflow analysis and VLCI (very large-scale integration) chips [10,11]. The CPN language allows you to present a model as a set of modules, which allows you to represent complex networks (and systems) in a hierarchical order. The goal of an in-depth study of various extensions of Petri nets (Colored Petri nets, Temporary and Stochastic Petri nets, etc.) [5,8,9,12] for modeling real-time systems (from an optimization point of view) leads to the design of such technical means where it is necessary to minimize costs resources, time and maximize speed. Petri nets are similar in their properties and modeling capabilities to Neural networks and Markov chains.

Description of Petri Nets

A Petri net is a directed bipartite graph in which nodes represent transitions (i.e., discrete events that can occur, represented by rectangles or bars), places (i.e., conditions, represented by circles), and directed arcs (describing what places located before and/or after the conditions under which transitions occur are indicated by arrows). Places (positions) of Petri nets can contain tokens, the presence and quantity of which changes during the operation of the network. Using positions containing the corresponding number of chips, the state of the network is described, and using transitions, the actions that took place in the network [1-4]. A Petri net model can be described by a set of linear algebraic equations or other mathematical models that reflect the behavior of the system. This opens up the possibility for formal analysis of the model [6]. This in turn allows for formal verification of properties associated with the behavior of the underlying system, such as precedence relationships between events, parallel operations, appropriate synchronization, and the elimination of deadlocks, repetitive actions, and mutual exclusion of shared resources. A simulation-based model can only provide a limited set of states of the system object being modeled and, thus, can only show the presence (but not the absence) of errors in the model and the underlying requirements specifications [6].

Petri nets have been used to model fault tolerance and real-time security. Colored Petri nets are a modern extension of ordinary or classical Petri nets [8,11,12], where the idea of a data type and a number of other ideas inherent in programming languages are added. As in classical Petri nets, each position is assigned a name. However, the names do not have any formal meaning; rather, they give the network greater clarity. Each position is accompanied by a data type that defines the type of data in the position. Each token has a specific meaning related to the type of data attached to the item containing the token. Each chip has an integer attached to it, which is the coefficient of the given marking in the position, i.e., the number of chips with this value in the given position is determined. For a given point in time, the network is characterized by its state, which represents the types and number of chips in each position. Before the CPN starts operating, its positions are assigned a certain number of tokens belonging to the corresponding data types that describe the initial state of the system, i.e., the network is initialized [10,11]. The types of chips themselves are called colors. Colored Petri nets are similar to high-level programming languages, while classical Petri nets are similar to low-level programming languages. In Colored Petri Nets, unlike standard Petri Nets, a position can contain tokens of arbitrary complexity - records, lists, etc., which allows you to create more reliable models and simulate complex systems [7]. Here, too, the events of the simulated system are described by transitions.

The number of tokens to be removed or added is determined by arc expressions, which can have a simple form, for example, be a constant belonging to some type, in which case the arc can only transmit a token that has a given value. The expression can also consist of variables and contain a conditional function that determines the number of tokens transmitted along this arc. If an expression does not contain variables, it is called a closed expression. In arc expressions, variables must be assigned a specific value in order for the expressions to be calculated or evaluated. However, not all such assignments result in a possible transition. In order for the corresponding input positions to contain the corresponding number of tokens with such a value, only in this case is it permissible to perform a transition. The above-mentioned properties make the structure of the Colored Petri net dynamic in the following sense: if the number of transmitted tokens in classical Petri nets is determined by the number of arcs, and these arcs are fixed in advance and they determine the static structure of the network, then in Colored Petri nets the number of tokens during the firing of the transition , are determined by an arc expression, which, in the case of different values of the variables included in it, can transmit a different number of tokens. This allows you to reduce the number of arcs and positions in the network compared to classical Petri nets.

In addition, in complex real-time systems it is often necessary to apply the priority principle for various orderings of transition firing, depending on the conditional function assigned to the arc. In this regard, classical Petri Nets have limited capabilities for modeling and implementing such systems, compared to Colored Petri nets [1-4,7].

Description and Modeling of The System for Identifying and Generating Secret Keys

In the modern world, biometrics or biometric recognition accompanies us at almost every step, is widely used both in public administration and in private companies (military forces, border control, healthcare, financial and banking sector, physical access control or individual login, payments of social benefits, etc.) [13,14]. Biometric identification systems have been studied by O'Sullivan, et al. [15] and Willems, et al [14]. [16,17]. They proposed storing biometric registration sequences in clear text and determined the corresponding identification capability. More recently, Trucel [17] analyzed the

DOI: 10.26717/BJSTR.2025.60.009449

trade-off between the capacity of a biometric identification system and the amount of memory (compression ratio) required for biometric templates. It should be noted that Trucel's method implements a kind of privacy protection scheme [17]. Recall that the privacy level, introduced by Ahlswede and Cisar [17,18], can be considered as the amount of total secret information that can be obtained in an authentication system in which the supporting data is available (public). Interestingly, this secrecy, equal to the mutual information between biometric registration and authentication sequences in biometric settings, is equal to the identification power found by O'Sullivan, et al. [15], Willems, et al. [14]. In this system, two terminals monitor the biometric sequences of registration and identification of a group of individuals. The first terminal generates a secret key for each registered person and stores the corresponding supporting data in a public database.

This auxiliary data, on the one hand, contributes to the reliable recovery of the secret key, and on the other hand, makes it possible to determine the identity of the person for the second terminal based on the presented biometric identification sequence. The database assumes that all supporting data is publicly available. Because the biometric secrets created by the first terminal are used, for example, for data encryption. Figure 1 shows the work of Colored Petri Net. In Figure 1 in Declaration those variables and types that participate in the Colored Petri Net are declared.

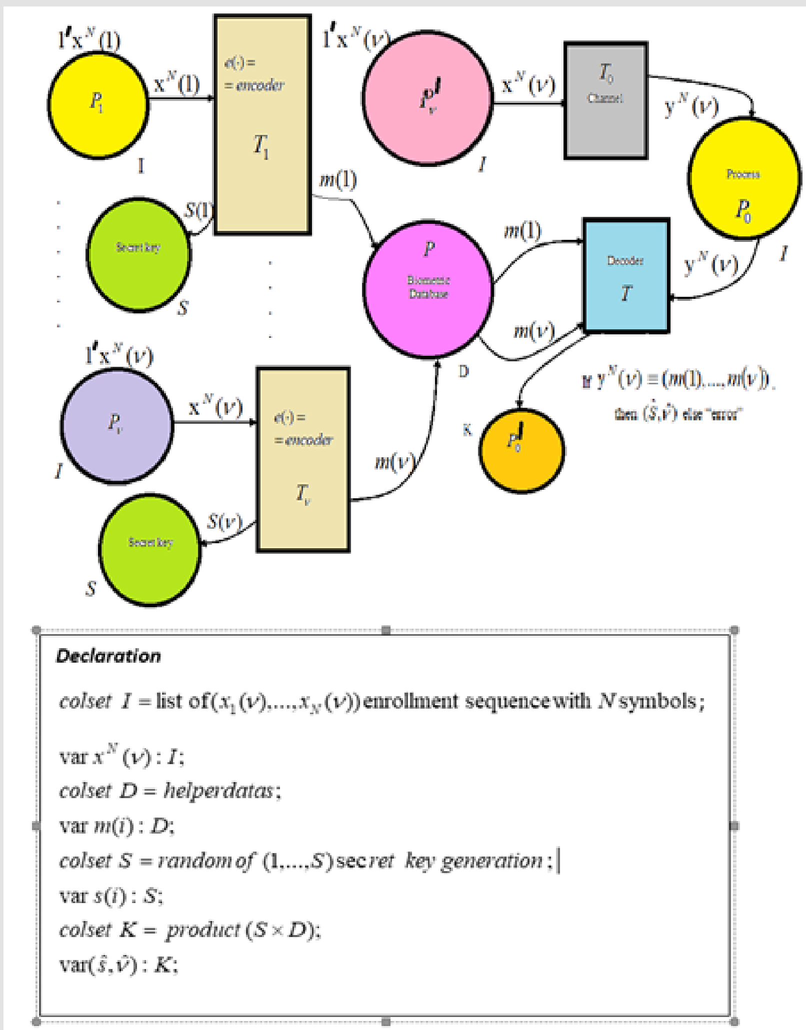

Figure 1: CPN model of a system for identifying and generating secret keys.

Conclusion

The article describes network modeling of a system for biometric identification and generation of secret keys using a Colored Petri Net. Figure 1 shows the network and describes in detail the process of registering and identifying data of individuals, as well as generating secret keys. Colored Petri nets are ideal for modeling real systems, studying its individual elements, checking, validating the system, detecting and preventing emergency situations, for implementing and replacing individual system components with optimal options. And this means that before creating such a system and implementing the process described above, Colored Petri nets are a very convenient simulation system from the point of view of process optimization and cost minimization.

References

1. Murata T (1989) Petri Nets: Properties, Analysis and Applications. Proceedings of the IEEE 77(4): 541-580.
2. Kotov V E (1984) Petri nets. M Nauka.
3. Peterson J (1984) Petri net theory and systems modeling. M Mir.
4. Peterson J L (1981) Petri Net Theory and the Modeling of Systems. PRENTICE-HALL Englewood Cliffs N J.
5. Nagasaki M, Doi A, Matsuno H, Miyano S (2004) A Versatile Petri Net Based Architecture for Modeling and Simulation of Complex Biological Processes. Genome Informatics 15(1): 180-197.
6. Shalyto AA (2000) Algorithmization and programming for logic control systems and “reactive” systems. Automation and telemechanics 1: 3-39.
7. Kristensen Lars M, Christensen S, Jensen K (1998) The practitioner’s guide to Coloured Petri Nets. Springer Verlag 2: 98-132.
8. Genrich H, Küffner R, Voss K (2000) Executable Petri net models for the analysis of metabolic pathways. In: Proceedings of the 21st International Conference on Application and Theory of Petri Nets Workshop Proceedings Practical Use of High-level Petri Nets Aarhus, p. 1-14.
9. Genrich H, Küffner R, Voss K (2001) Executable Petri net models for the analysis of metabolic pathways. International Journal on Software Tools for Technology Transfer 3(4): 394-404.
10. Jensen K Introduction to the practical use of Coloured Petri nets.
11. Jensen K (1996) Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use. Springer 1(3).
12. Matsuno H, Doi A, Nagasaki M, Miyano S (2000) Hybrid Petri net representation of gene regulatory network. Proc Pacific Symposium on Biocomputing, pp. 338-349.
13. S Pankanti, R M Bolle, A Jain (2002) Biometrics-The Future of Identification. IEEE Computer 33(2): 46-49.
14. F WIllems, T Kalker, J Goselig, J P Linnartz (2003) On the capacity of a biometric identification system. Intenational Symposium on Information Theory Yokohama Japan, p. 82.
15. JA O’Sullivan, N A Schmid (2004) Performance prediction methodology for biometric systems using a large deviations approach. IEEE Trans On it Signal Proc 52(10): 3036-3045.
16. E Trucel (2006) Capacity/storage tradeoff in high-dimensional identification system. IEEE International Symposium on Information Theory Washington USA, pp. 1929-1933.
17. https://ieeexplore.ieee.org/document/7593567.
18. https://www.hstoday.us/industry/idemia-launches-saas-based-automatic-biometric-identification-system/.

ISSN: 2574-1241

DOI: 10.26717/BJSTR.2025.60.009449

Petrosyan Goharik. Biomed J Sci & Tech Res