Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

A general analytical model of queuing system for Internet of Things applications

Volodymyr G. Skobelev, Volodymyr V. Skobelev

Abstract

In the given paper a general analytical model for a queuing system with limited capacity buffer intended to control packets' traffic in Internet of Things applications is proposed. This model is based on the following assumptions. There is a fixed number of packets' classes. For each pair of these classes either preemptive or non-preemptive priority is set. Packets of each class arrive according to the Poisson process with the given arrival rate, and are transmitted without errors with the given transmission rate. The criterion on the structure of the set of priorities between the classes of packets avoiding unnecessary push-out of packets being in the transmission is proved. Continuous-Time Markov Chain associated with the proposed model has been defined and analyzed. Basic characteristics including blocking probability, push-out probability, delay, and utilization have been estimated for each class of packets. Basic measures for the proposed model, such as Grade of Service, cost function of operation, and performance are established.

Keywords: IoT, queuing system, limited capacity buffer, Continuous-Time Markov Chain, performance analysis.

1 Introduction

The rapid growth of the Internet of things (IoT) technologies caused the necessity of developing the means for effective processing of massive heterogeneous traffics at limited capabilities [1, 2, 3]. The word combination *limited capabilities* means that for storage of the accepted

packets before their transmission sufficiently small buffers are used. To process the packets with the quality of service (QoS) requirements, different models based on priorities assigned to packets entering the networks and algorithms for selection and transmission of the next packet have been proposed [4, 5, 6].

A probabilistic approach for analysis of traffic models in IoT has been developed in [7, 8]. A number of Markov chain based queuing models and measures that characterize QoS requirements of IoT have been proposed and analyzed in [9, 10, 11].

It should be noted the paper [11], where for the proposed Markovian model the basic performance measures for different traffic classes have been studied extensively. These measures include blocking probability, push out probability, delay, channel utilization, and overall system performance.

In the present paper some generalization of the analytical model proposed in [11] is defined and analyzed.

2 Proposed model

For investigation the performance of a single server queueing system with a finite capacity buffer the following model $\mathfrak{M}$ (Fig. 1) is proposed.

The model operates with k ($k \in \mathbb{N}, k > 2$) priority classes of packets $C_1, \ldots, C_k$ with different traffic types. The packets $p \in C_i$ ($i = 1, \ldots, k$) arrive according to the Poisson process with the arrival rate λ_i, and these Poisson processes are independent. The transmission time for the packets $p \in C_i$ ($i = 1, \ldots, k$) is exponentially distributed with the transmission rate μ_i. The transmission of the packets is error-free.

The Buffer consists of k queues, namely Queue1, ... , Queue$k$, where Queue$i$ ($i = 1, \ldots, k$) consists of the packets $p \in C_i$ being in The Buffer. The Buffer is not time slotted. The number of packets in the Queue$i$ ($i = 1, \ldots, k$) is denoted $|\text{Queue}i|$. The total number of packets in the queues, including the packet in the transmission, does not exceed the given integer n ($n > k$). Thus, $0 \leq \sum_{i=1}^{k} |\text{Queue}i| \leq n-1$.

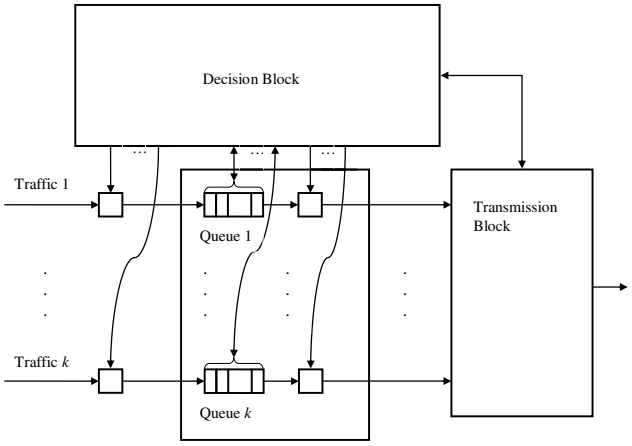

Figure 1. The model $\mathfrak{M}$

The Decision Block implements the queues management, i.e. push-out buffer mechanism, selection of the next packet for the transmission, and, if necessary, returning the packet, being in the transmission, to the beginning of the appropriate queue. In the latter case, the packet will be re-transmitted completely, i.e. from the very beginning.

The classes $C_1, \ldots, C_k$ are enumerated in the ascending order of priority. Preemptive and non-preemptive priorities are distinguished in the following way. The fixed set $U_i \subseteq \{C_1, \ldots, C_{i-1}\}$ $(i = 1, \ldots, k)$ consists of the classes over which class C_i has preemptive priority. The set $V_i = \{C_1, \ldots, C_{i-1}\} \setminus U_i$ $(i = 1, \ldots, k)$ consists of the classes over which C_i has non-preemptive priority. Since $U_i \cup V_i = \{C_1, \ldots, C_{i-1}\}$ and $U_i \cap V_i = \emptyset$, the set V_i is uniquely defined by the set U_i. In particular, $U_1 = V_1 = \emptyset$.

Remark 1. Preemptive and non-preemptive priorities can be characterized as follows. The transmission of a packet $p \in C_k$ continues until its completion. A packet $p_1 \in C_i$ $(i = 1, \ldots, k-1)$ can be in the transmission if and only if any element of the set $\{\text{Queue}j | j > i \& C_i \in U_j\}$ is the empty queue. Let a packet $p_1 \in C_i$ $(i = 1, \ldots, k-1)$ be in the transmission, and a packet $p_2 \in C_l$ enters

the model $\mathfrak{M}$. The following two situations are possible.

Let $C_i \notin U_l$. If $\sum\limits_{j=1}^{k} |\text{Queue}j| < n-1$, then the packet p_2 enters the Queue$l$. If $\sum\limits_{j=1}^{k} |\text{Queue}j| = n-1$ and $\sum\limits_{j=1}^{l-1} |\text{Queue}j| > 0$, then the packet p_2 enters the Queue$l$, and from the model $\mathfrak{M}$ is pushed-out the last packet in the Queuej_0, where $j_0 = \min\{j | j < l \& |\text{Queue}j| \neq 0\}$. Otherwise, the packet p_2 is pushed-out from the model $\mathfrak{M}$.

Let $C_i \in U_l$. The packet p_1 is pushed-out from the transmission and the transmission of the packet p_2 starts. For the packet p_1, the following two situations are possible. If $\sum\limits_{j=1}^{k} |\text{Queue}j| < n-1$, then the packet p_1 is placed at the beginning of the Queue$i$. If $\sum\limits_{j=1}^{k} |\text{Queue}j| = n-1$ and $\sum\limits_{j=1}^{i} |\text{Queue}j| > 0$, then the packet p_1 is placed at the beginning of the Queue$i$, and from the model $\mathfrak{M}$ the last packet in the Queuej_0, where $j_0 = \min\{j | 1 \leq j \leq i \& |\text{Queue}j| \neq 0\}$, is pushed-out. Otherwise, the packet p_2 is pushed-out from the model $\mathfrak{M}$.

If the structure of the sets U_i $(i = 2, \ldots, k)$ is not constrained, then some unnecessary push-out of packets being in the transmission can arise. The next criterion excludes these situations.

Theorem 1. *In the model $\mathfrak{M}$ there are no unnecessary push-outs of packets being in the transmission if and only if the following formula is true:*

$$(\forall i_1 = 2, \ldots, k-1)(\forall i_2 = i_1 + 1, \ldots, k)(U_{i_1} \subseteq U_{i_2} \vee C_{i_1} \in V_{i_2}). \quad (1)$$

Proof. 1. Suppose that formula (1) is true, and some packet $p_1 \in C_i$ $(i = 1, \ldots, k)$ is in the transmission. Then any element of the set $\{\text{Queue}j | j > i \& C_i \in U_j\}$ is the empty queue.

If $i = k$, then the transmission of the packet $p_1 \in C_i$ is continued until its completion.

Let $i < k$ and a packet $p_2 \in C_l$ enters the Queue l.

If $l \leq i$, or $l > i$ and $C_i \in V_l$, then the transmission of the packet $p_1 \in C_i$ is continued.

If $l > i$ and $C_i \notin V_l$ (i.e. $C_i \in U_l$), then the packet p_2 is the single element of the Queue l. The packet $p_1 \in C_i$ is pushed-out from the transmission, and the transmission of the packet $p_2 \in C_l$ starts.

Formula (1) can be rewritten in the following equivalent form:

$$(\forall i_1 = 2, \ldots, k-1)(\forall i_2 = i_1 + 1, \ldots, k)(C_{i_1} \in U_{i_2} \Rightarrow U_{i_1} \subseteq U_{i_2}).$$

Thus, for all $j = l+1, \ldots, k$, if $C_l \in U_j$, then $U_l \subseteq U_j$. Since $C_i \in U_l$, then $C_i \in U_j$ for all $j = l+1, \ldots, k$ such that $U_l \subseteq U_j$. But $|\text{Queue}\, j| = 0$ for all $j = l+1, \ldots, k$ such that $U_l \subseteq U_j$. Therefore, the transmission of the packet $p_2 \in C_l$ is continued either until its complition or till some packet $p_3 \in C_j$ such that $C_l \in U_j$ $(j = l+1, \ldots, k)$ enters the Queue j.

Thus, if formula (1) is true, then there are no unnecessary push-outs of packets being in the transmission.

2. Suppose that formula (1) is false. Then formula

$$(\exists i_1 = 2, \ldots, k-1)(\exists i_2 = i_1 + 1, \ldots, k)(U_{i_1} \not\subseteq U_{i_2} \& C_{i_1} \in U_{i_2})$$

is true.

Let $i_1 \in \{2, \ldots, k-1\}$ and $i_2 \in \{i_1+1, \ldots, k\}$ be some integers such that $U_{i_1} \not\subseteq U_{i_2}$. Then there exists some class $C_j \in U_{i_1}$ $(1 \leq j \leq i_1 - 1)$ such that $C_j \notin U_{i_2}$.

Let a packet $p_1 \in C_j$ be in the transmission. Then any element of the set $\{\text{Queue}\, l | l > j \& C_j \in U_l\}$ is the empty queue. Suppose also that a packet $p_2 \in C_{i_2}$ be in the Queue i_2. This situation is admissible since $C_j \notin U_{i_2}$, i.e. $C_j \in V_{i_2}$.

Let a packet $p_3 \in C_{i_1}$ enters the Queue i_1. Since $C_j \in U_{i_1}$, then the packet $p_1 \in C_j$ is pushed-out from the transmission and the transmission of the packet $p_3 \in C_{i_1}$ starts. But since $C_{i_1} \notin V_{i_2}$ (i.e. $C_{i_1} \in U_{i_2}$), then the packet $p_3 \in C_{i_1}$ is pushed-out from the transmission and the transmission of the packet $p_2 \in C_{i_2}$ starts.

Thus, if formula (1) is false, then there can be unnecessary push-out of packets being in the transmission. □

In what follows it is supposed that for the analyzed model $\mathfrak{M}$ formula (1) is true.

3 A brief analysis of the model $\mathfrak{M}$

Due to the assumptions made in Section 2, the model $\mathfrak{M}$ can be treated (in Kendall notation) as some $M/M/1/n$ queueing system with preemptive and non-preemptive priorities between the classes of packets.

Some rough estimations for the model $\mathfrak{M}$ can be established under the assumption that we are dealing with an ordinary $M/M/1/n$ queueing system, such that the total input stream of packets is the Poisson process with the arrival rate

$$\lambda = \sum_{i=1}^{k} \lambda_i,$$

and the transmission time for packets is exponentially distributed with the transmission rate μ.

Remark 2. Under this assumption, we are sweeping priorities between the classes of packets under the rug, partly taking into account their influence in the value of μ.

It is not simple to compute the transmission rate μ. Possibly, to do this some expert methods or results of computer simulation of the model $\mathfrak{M}$ functioning will be required. Nevertheless, the following inequalities are true

$$\min\{\mu_i | i = 1, \ldots, k\} \leq \mu \leq \max\{\mu_i | i = 1, \ldots, k\}. \tag{2}$$

It is well known that the utilization factor ϱ_i $(i = 1, \ldots, k)$ for the input stream of packets $p \in C_i$ is defined as $\varrho_i = \lambda_i \mu_i^{-1}$. Similarly, the utilization factor ϱ for the total input stream of packets into the model $\mathfrak{M}$ can be defined as

$$\varrho = \lambda \mu^{-1}. \tag{3}$$

Due to (2) and (3), the following inequalities are true

$$\frac{\lambda}{\max\{\mu_i | i = 1, \ldots, k\}} \leq \varrho \leq \frac{\lambda}{\min\{\mu_i | i = 1, \ldots, k\}}. \tag{4}$$

According to the results presented in [12], we can get the following estimations for the model $\mathfrak{M}$.

The stationary probability P_m $(m = 0, 1, \ldots, n)$ for m packets being in the model $\mathfrak{M}$ is estimated as

$$P_m = \varrho^m P_0, \tag{5}$$

where

$$P_0 = \begin{cases} (1-\varrho)(1-\varrho^{n+1})^{-1}, & if\ \varrho \neq 1 \\ (n+1)^{-1}, & if\ \varrho = 1 \end{cases}. \tag{6}$$

Due to (5) and (6), the saturation probability for the model $\mathfrak{M}$ is estimated as

$$P_n = \begin{cases} \varrho^n(1-\varrho)(1-\varrho^{n+1})^{-1}, & if\ \varrho \neq 1 \\ (n+1)^{-1}, & if\ \varrho = 1 \end{cases}. \tag{7}$$

The blocking probability P_{blck} that packets are blocked and rejected by the model $\mathfrak{M}$ since its capacity is full (i.e. there are n packets in the model $\mathfrak{M}$) is estimated as

$$P_{blck} = (P_0 + \varrho - 1)\varrho^{-1}. \tag{8}$$

Substituting (6) in (8), we get

$$P_{blck} = \begin{cases} \varrho^n(1-\varrho)(1-\varrho^{n+1})^{-1}, & if\ \varrho \neq 1 \\ (n+1)^{-1}, & if\ \varrho = 1 \end{cases}. \tag{9}$$

Therefore, the blocking probability P_{blck} equals to the saturation probability P_n.

The average number N of packets being in the model $\mathfrak{M}$ is estimated as

$$N = \sum_{m=0}^{n} mP_m. \tag{10}$$

Substituting (5) and (6) in (10), we get

$$N = \begin{cases} \varrho(1-\varrho)^{-1}(1-(n+1)P_n), & if\ \varrho \neq 1 \\ 0.5n, & if\ \varrho = 1 \end{cases}. \tag{11}$$

The average number of packets N_{trns} being in the transmission in the model $\mathfrak{M}$ is estimated as

$$N_{trns} = \varrho(1 - P_n). \tag{12}$$

Substituting (7) in (12), we get

$$N_{trns} = \begin{cases} \varrho(1-\varrho^n)(1-\varrho^{n+1})^{-1}, & if\ \varrho \neq 1 \\ n(n+1)^{-1}, & if\ \varrho = 1 \end{cases}. \tag{13}$$

The average number of packets N_{ques} that are waiting in the queues in the model $\mathfrak{M}$ is estimated as

$$N_{ques} = N - N_{trns}. \tag{14}$$

Substituting (11) and (13) in (14), we get

$$N_{ques} = \begin{cases} \varrho(1-\varrho)^{-1}(\varrho - (n+\varrho)P_n), & if\ \varrho \neq 1 \\ 0.5n(n-1)(n+1)^{-1}, & if\ \varrho = 1 \end{cases}. \tag{15}$$

Due to the Little's law, the average time T spent by a packet in the model $\mathfrak{M}$ is estimated as

$$T = N\lambda^{-1}(1 - P_n)^{-1}. \tag{16}$$

Substituting (7) and (11) in (16), we get

$$T = \begin{cases} \varrho(1-\varrho)^{-1}\lambda^{-1}(1 - n\varrho^n(1-\varrho)(1-\varrho^n)^{-1}), & if\ \varrho \neq 1 \\ 0.5(n+1)\lambda^{-1}, & if\ \varrho = 1 \end{cases}. \tag{17}$$

Similarly, due to the Little's law, the average time W spent by a packet being waiting in the queue in the model $\mathfrak{M}$ is estimated as

$$W = N_{ques}\lambda^{-1}(1 - P_n)^{-1}. \tag{18}$$

Substituting (7) and (15) in (18), we get

$$W = \begin{cases} \varrho(1-\varrho)^{-1}\lambda^{-1}(\varrho - n\varrho^n(1-\varrho)(1-\varrho^n)^{-1}), & if\ \varrho \neq 1 \\ 0.5(n-1)\lambda^{-1}, & if\ \varrho = 1 \end{cases}. \tag{19}$$

Using formulas (3), (5)-(7), (9), (11), (13), (15), (17) and (19), it is possible to estimate the basic values of the parameters of the model $\mathfrak{M}$ for different values of λ_i $(i = 1, \ldots, k)$, μ_i $(i = 1, \ldots, k)$, and n.

4 Associated Continuous-Time Markov Chain

Due to the assumptions made in Section 2, the following Continuous-Time Markov Chain $\mathcal{C}$ can be associated with the model $\mathfrak{M}$.

A state of the Chain $\mathcal{C}$ is any vector $(n_1, \ldots, n_k, a) \in \mathbb{Z}_+$, such that $0 \le \sum\limits_{i=1}^{k} n_i \le n$ and $0 \le a \le k$, where:

1. The integer n_i $(i = 1, \ldots, k)$ is the number of packets $p \in C_i$ in the Queue$i$ including the packet of the class C_i in the transmission, if it is there.

2. The integer a equals to 0, if the system is empty, and a is the number of the class of the packet in the transmission, otherwise, i.e.

$$a = 0 \Leftrightarrow (\forall i = 1, \ldots, k)(n_i = 0),$$

and

$$(\forall i = 1, \ldots, k)(a = i \Rightarrow n_i > 0\&$$

$$\&(\forall j = i+1, \ldots, k)(C_i \in U_j \Rightarrow n_j = 0)).$$

Due to Remark 1, the state transitions of the Continuous Time Markov Chain $\mathcal{C}$ can be defined as follows.

1. Let $\mathbf{s} = (\underbrace{0, \ldots, 0}_{k\ times}, 0)$.

For any $l = 1, \ldots, k$ there are the transitions:

$$\mathbf{s} \xrightarrow{\lambda_l} (\underbrace{0, \ldots, 0}_{l-1}, 1, \underbrace{0, \ldots, 0}_{k-l\ times}, l)$$

and

$$(\underbrace{0, \ldots, 0}_{l-1}, 1, \underbrace{0, \ldots, 0}_{k-l\ times}, l) \xrightarrow{\mu_l} \mathbf{s}.$$

2. Let $\mathbf{s} = (n_1, \ldots, n_k, a)$ $\left(0 < \sum\limits_{i=1}^{k} n_i < n,\ 1 \le a \le k\right)$.

For any $l \in \{1, \ldots, k\}$ such that $C_a \notin U_l$ there is the transition

$$\mathbf{s} \xrightarrow{\lambda_l} (n_1, \ldots, n_{l-1}, n_l + 1, n_{l+1}, \ldots, n_k, a).$$

For any $l \in \{1, \ldots, k\}$ such that $C_a \in U_l$ there is the transition

$$\mathbf{s} \xrightarrow{\lambda_l} (n_1, \ldots, n_{l-1}, n_l + 1, n_{l+1}, \ldots, n_k, l).$$

3. Let $\mathbf{s} = (n_1, \ldots, n_k, a)$ $\left(\sum\limits_{i=1}^{k} n_i = n,\ 1 \le a \le k\right)$.

For any $l \in \{2, \ldots, k\}$ such that $C_a \notin U_l$ and either $a \notin \{1, \ldots, l-1\}$ and $\sum\limits_{i=1}^{l-1} n_i > 0$, or $a \in \{1, \ldots, l-1\}$ and $\sum\limits_{i=1}^{l-1} n_i > 1$ there is the transition

$$\mathbf{s} \xrightarrow{\lambda_l} (n_1, \ldots, n_{j_0-1}, n_{j_0} - 1, n_{j_0+1}, \ldots, n_{l-1}, n_l + 1, n_{l+1}, \ldots, n_k, a),$$

where the integer j_0 is defined as follows:

1) $j_0 = \min\{j \in \{1, \ldots, l-1\} | n_j > 0\}$, if either $a \notin \{1, \ldots, l-1\}$ and $\sum\limits_{i=1}^{l-1} n_i > 0$, or $a \in \{1, \ldots, l-1\}$, $\sum\limits_{i=1}^{l-1} n_i > 1$ and $n_a \ge 2$;

2) $j_0 = \min\{j \in \{1, \ldots, l-1\} \setminus \{a\} | n_j > 0\}$, if $a \in \{1, \ldots, l-1\}$, $\sum\limits_{i=1}^{l-1} n_i > 1$ and $n_a = 1$.

For any $l \in \{2, \ldots, k\}$ such that $C_a \in U_l$ there is the transition

$$\mathbf{s} \xrightarrow{\lambda_l} (n_1, \ldots, n_{j_0-1}, n_{j_0} - 1, n_{j_0+1}, \ldots, n_{l-1}, n_l + 1, n_{l+1}, \ldots, n_k, l),$$

where $j_0 = \min\{j \in \{1, \ldots, l-1\} | n_j > 0\}$.

4. Let $\mathbf{s} = (n_1, \ldots, n_k, a)$ $\left(2 \le \sum\limits_{i=1}^{k} n_i \le n,\ 1 \le a \le k\right)$.

There is the transition

$$\mathbf{s} \xrightarrow{\mu_a} (n_1, \ldots, n_{a-1}, n_a - 1, n_{a+1}, \ldots, n_k, j_1),$$

where $j_1 = \max\{j | 1 \le j \le k \& n_j > 0\}$.

The state transitions defined above directly imply that the structure of the Continuous-Time Markov Chain $\mathcal{C}$ is substantially dependent on the structure of the sets U_i $(i = 2, \ldots, k)$. Nevertheless, for each

Continuous-Time Markov Chain $\mathcal{C}$ the infinitesimal generator matrix $Q_{\mathcal{C}}$ and the embedded chain transition matrix $P_{\mathcal{C}}$ can be constructed.

Let $\mathcal{C}$ be the given Continuous-Time Markov Chain with the set of the states $\mathbf{S}$, $\overrightarrow{\pi}$ be the probability stationary distribution (i.e. the limiting distribution) of the chain $\mathcal{C}$, and $\overrightarrow{\psi}$ be the probability stationary distribution of the embedded chain. We denote the component of the vector $\overrightarrow{\pi}$ that corresponds to the state $\mathbf{s}$ by $\pi_{\mathbf{s}}$ ($\mathbf{s} \in \mathbf{S}$), and the component of the vector $\overrightarrow{\psi}$ that corresponds to the state $\mathbf{s}$ – by $\psi_{\mathbf{s}}$ ($\mathbf{s} \in \mathbf{S}$).

Remark 3. The difference between the vectors $\overrightarrow{\pi}$ and $\overrightarrow{\psi}$ is as follows. For any state $\mathbf{s} \in \mathbf{S}$ the component $\pi_{\mathbf{s}}$ of the vector $\overrightarrow{\pi}$ is the long-term proportion of time that the chain $\mathcal{C}$ spends in the state $\mathbf{s}$ (i.e. the stationary probability for the chain $\mathcal{C}$ to be in the state $\mathbf{s}$). At the same time, for any state $\mathbf{s} \in \mathbf{S}$ the component $\psi_{\mathbf{s}}$ is the long-term proportion of transitions that the chain $\mathcal{C}$ makes into the state $\mathbf{s}$ (i.e. the stationary probability for the chain $\mathcal{C}$ to transit to the state $\mathbf{s}$).

The vector $\overrightarrow{\pi}$ can be computed as the solution of the equation

$$\overrightarrow{\pi} Q_{\mathcal{C}} = \mathbf{0},$$

that satisfies the conditions $\pi_{\mathbf{s}} \geq 0$ ($\mathbf{s} \in \mathbf{S}$) and $\sum\limits_{\mathbf{s} \in \mathbf{S}} \pi_{\mathbf{s}} = 1$, and the vector $\overrightarrow{\psi}$ can be computed as the solution of the equation

$$\overrightarrow{\psi} P_{\mathcal{C}} = \overrightarrow{\psi},$$

that satisfies the conditions $\psi_{\mathbf{s}} \geq 0$ ($\mathbf{s} \in \mathbf{S}$) and $\sum\limits_{\mathbf{s} \in \mathbf{S}} \psi_{\mathbf{s}} = 1$.

5 Analysis of the model $\mathfrak{M}$ on the basis of the Continuous-Time Markov Chain $\mathcal{C}$

For any state $\mathbf{s} \in \mathbf{S}$ of the Continuous-Time Markov Chain $\mathcal{C}$ we use $n_{\mathbf{s},i}$ ($i = 1, \ldots, k$) to denote the number of packets $p \in C_i$ in the Queue$i$ including the packet of the class C_i in the transmission, if it is there.

The average number N_i $(i = 1, \ldots, k)$ of packets of the class C_i in the model $\mathfrak{M}$ is estimated as

$$N_i = \sum_{\mathbf{s} \in \mathbf{S}} \pi_{\mathbf{s}} n_{\mathbf{s},i}. \tag{20}$$

Therefore, the average number N of packets being in the model $\mathfrak{M}$ is estimated as

$$N = \sum_{i=1}^{k} N_i. \tag{21}$$

Substituting (20) in (21), we get

$$N = \sum_{i=1}^{k} \sum_{\mathbf{s} \in \mathbf{S}} \pi_{\mathbf{s}} n_{\mathbf{s},i}. \tag{22}$$

We define the subsets $\mathbf{S}_m$ $(m = 0, 1, \ldots, n)$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ via identity

$$\mathbf{S}_m = \left\{ \mathbf{s} \in \mathbf{S} \mid \sum_{i=1}^{k} n_{\mathbf{s},i} = m \right\}.$$

The stationary probability P_m $(m = 0, 1, \ldots, n)$ for m packets being in the model $\mathfrak{M}$, including the packet in the transmission, if it is there, is estimated as

$$P_m = \sum_{\mathbf{s} \in \mathbf{S}_m} \pi_{\mathbf{s}}. \tag{23}$$

In particular, the saturation probability for the model $\mathfrak{M}$ is estimated as

$$P_n = \sum_{\mathbf{s} \in \mathbf{S}_n} \pi_{\mathbf{s}}. \tag{24}$$

The average number of packets N_{trns} being in the transmission in the model $\mathfrak{M}$ is estimated as

$$N_{trns} = 1 - P_0. \tag{25}$$

Due to (14), (22) and (25), the average number of packets N_{ques} that are waiting in the queues in the model $\mathfrak{M}$ is estimated as

$$N_{ques} = P_0 - 1 + \sum_{i=1}^{k} \sum_{\mathbf{s} \in \mathbf{S}} \pi_{\mathbf{s}} n_{\mathbf{s},i}. \tag{26}$$

The stationary probability β_m for the model $\mathfrak{M}$ to transit into the subset $\mathbf{S}_m$ $(m = 0, 1, \ldots, n)$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ is estimated as

$$\beta_m = \sum_{\mathbf{s} \in \mathbf{S}_m} \psi_{\mathbf{s}}. \tag{27}$$

In particular, the stationary probability β_n for the model $\mathfrak{M}$ to transit into the subset $\mathbf{S}_n$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ is estimated as

$$\beta_n = \sum_{\mathbf{s} \in \mathbf{S}_n} \psi_{\mathbf{s}}. \tag{28}$$

The subset $\mathbf{S}_n$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ can be characterized as follows: $\mathbf{s} \in \mathbf{S}_n$ if and only if when the next packet arrives, then either this packet, or some packet being in the model $\mathfrak{M}$ will be lost. Therefore, the subset $\mathbf{S}_n$ consists of all critical states for the model $\mathfrak{M}$. Hence, for the model $\mathfrak{M}$, the value of β_n is the probability of transition to a critical state, while the value of P_n is the probability for being in a critical state.

We define the subsets $\mathbf{S}_n(a, i)$ $(a, i \in \{1, \ldots, k\}, C_a \notin U_i)$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ via identity

$$\mathbf{S}_n(a, i) = \left\{ \mathbf{s} = (n_{\mathbf{s},1}, \ldots, n_{\mathbf{s},k}, a) \in \mathbf{S}_n \mid \sum_{\substack{j=1 \\ j \neq a}}^{i-1} n_{\mathbf{s},j} = 0 \right\}.$$

The blocking probability γ_i $(i = 1, \ldots, k)$ that the next arrived packet $p \in C_i$ is blocked and rejected by the model $\mathfrak{M}$ (since its capacity is full) is estimated as

$$\gamma_i = \sum_{\substack{a=1 \\ C_a \notin U_i}}^{k} \sum_{\mathbf{s} \in \mathbf{S}_n(a,i)} \pi_{\mathbf{s}}. \tag{29}$$

Applying the Little's law, we get that the average delay for packets $p \in C_i$ $(i = 1, \ldots, k)$ being in the model $\mathfrak{M}$ is estimated as

$$\delta_i = \lambda_i^{-1}(1-\gamma_i)^{-1}N_i. \tag{30}$$

Substituting (20) in (30), we get

$$\delta_i = \lambda_i^{-1}(1-\gamma_i)^{-1}\sum_{\mathbf{s}\in\mathbf{S}}\pi_{\mathbf{s}}n_{\mathbf{s},i}. \tag{31}$$

We define the subsets $\mathbf{S}_n^{(1)}(a,i)$ $(i \in \{1,\ldots,k-1\},\ a \in \{i+1,\ldots,k\})$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ via identity

$$\mathbf{S}_n^{(1)}(a,i) = \left\{ \mathbf{s} = (n_{\mathbf{s},1},\ldots,n_{\mathbf{s},k},a) \in \mathbf{S}_n \,|\, \sum_{j=1}^{i-1} n_{\mathbf{s},j} = 0 \& n_{\mathbf{s},i} > 0 \right\},$$

and also we define the sets $J(i)$ $(i \in \{1,\ldots,k-1\})$ via identity

$$J(i) = \{j \in \{i+1,\ldots,k\} | C_i \in U_j\}.$$

The push-out probability α_i $(i \in \{1,\ldots,k-1\})$ that a packet $p \in C_i$ that is waiting in the buffer or being in the transmission is pushed out from the model $\mathfrak{M}$ and is lost upon the arrival of some packet when the buffer is full is estimated as

$$\alpha_i = \alpha_i^{(1)} + \alpha_i^{(2)}, \tag{32}$$

where

$$\alpha_i^{(1)} = \sum_{\mathbf{s}\in\mathbf{S}_n(i,i)} \pi_{\mathbf{s}} \left(\sum_{j\in J(i)} \lambda_j \right) \left(\sum_{j\in J(i)} \lambda_j + \mu_i \right)^{-1}$$

and

$$\alpha_i^{(2)} = \sum_{a=i+1}^{k} \sum_{\mathbf{s}\in\mathbf{S}_n^{(1)}(a,i)} \pi_{\mathbf{s}} \left(\sum_{j=i+1}^{k} \lambda_j \right) \left(\sum_{j=i+1}^{k} \lambda_j + \mu_a \right)^{-1}.$$

The total push-out probability α for the model $\mathfrak{M}$ is estimated as

$$\alpha = \sum_{i=1}^{k-1} \alpha_i. \tag{33}$$

Due to [11], a Grade of Service $\mathfrak{m}_{GoS}$ for the model $\mathfrak{M}$ can be estimated as

$$\mathfrak{m}_{GoS} = \sum_{i=1}^{k} \gamma_i + w\alpha, \tag{34}$$

where w is a penalty weight for the push-out probability over the blocking probability for packets. Substituting (29), (33) and (32) in (34), we get

$$\mathfrak{m}_{GoS} = \sum_{i=1}^{k} \sum_{\substack{a=1 \\ C_a \notin U_i}}^{k} \sum_{\mathbf{s} \in \mathbf{S}_n(a,i)} \pi_{\mathbf{s}} + w \sum_{i=1}^{k-1} (\alpha_i^{(1)} + \alpha_i^{(2)}). \tag{35}$$

Due to [11, 13], the performance $\mathfrak{m}_{prfrm}$ of the model $\mathfrak{M}$ is estimated as

$$\mathfrak{m}_{prfrm} = \mathfrak{m}_{GoS}^{-1}. \tag{36}$$

Substituting (35) in (36), we get

$$\mathfrak{m}_{prfrm} = \left(\sum_{i=1}^{k} \sum_{\substack{a=1 \\ C_a \notin U_i}}^{k} \sum_{\mathbf{s} \in \mathbf{S}_n(a,i)} \pi_{\mathbf{s}} + w \sum_{i=1}^{k-1} (\alpha_i^{(1)} + \alpha_i^{(2)}) \right)^{-1}. \tag{37}$$

We define the subsets $\mathbf{S}(i)$ $(i \in \{1, \ldots, k\})$ of the states of the Continuous-Time Markov Chain $\mathcal{C}$ via identity

$$\mathbf{S}(i) = \{\mathbf{s} = (n_{\mathbf{s},1}, \ldots, n_{\mathbf{s},k}, a) \in \mathbf{S} \,|\, a = i\}.$$

The utilization $\mathfrak{m}_{utlz}(i)$ $(i = 1, \ldots, k)$ of packets $p \in C_i$ in the model $\mathfrak{M}$ is estimated as

$$\mathfrak{m}_{utlz}(i) = \sum_{\mathbf{s} \in \mathbf{S}(i)} \pi_{\mathbf{s}}. \tag{38}$$

Due to [11, 13], a cost $\mathfrak{m}_{cst}$ of operation for the model $\mathfrak{M}$ can be estimated as

$$\mathfrak{m}_{cst} = \left(\sum_{i=1}^{k} \mathfrak{m}_{utlz}(i) \right)^{-1}. \tag{39}$$

Substituting (38) in (39), we get

$$\mathfrak{m}_{cst} = \left(\sum_{i=1}^{k} \sum_{\mathbf{s} \in \mathbf{S}(i)} \pi_{\mathbf{s}}(i) \right)^{-1}. \tag{40}$$

Due to [11, 13], the total performance $\mathfrak{m}_{TP}$ for the model $\mathfrak{M}$ can be estimated as

$$\mathfrak{m}_{TP} = \mathfrak{m}_{prfrm} \mathfrak{m}_{cst}^{-1}. \tag{41}$$

Substituting (37) and (40) in (41), we get

$$\mathfrak{m}_{TP} = \frac{\sum\limits_{i=1}^{k} \sum\limits_{\mathbf{s} \in \mathbf{S}(i)} \pi_{\mathbf{s}}(i)}{\sum\limits_{i=1}^{k} \sum\limits_{\substack{a=1 \\ C_a \notin U_i}}^{k} \sum\limits_{\mathbf{s} \in \mathbf{S}_n(a,i)} \pi_{\mathbf{s}} + w \sum\limits_{i=1}^{k-1} (\alpha_i^{(1)} + \alpha_i^{(2)})}. \tag{42}$$

Maximization of the value $\mathfrak{m}_{TP}$ is the main way for improving the quality of service (QoS) of packets by the model $\mathfrak{M}$.

6 Conclusions

In the given paper a general analytical model of the queueing system for IoT-applications has been proposed and analyzed.

The estimations established in the paper make it possible to improve the QoS of the real system by selecting the permissible values of parameters λ_i $(i = 1, \ldots, k)$, μ_i $(i = 1, \ldots, k)$, and the size k of the Buffer. This improvement, as a rule, can be achieved via computer simulation. For a special case, when there are 4 classes of packets, the results of computer simulation are presented in [11].

References

[1] H.D. Ma, "Internet of Things: objectives and scientific challenges," *Journal of Computer Science and Technology*, vol. 26, no. 6, pp. 919—924, 2011.

[2] E. Borgia, "The Internet of Things vision: key features, applications and open issues," *Computer Communications*, vol. 54, pp. 1—31, 2014.

[3] D. Gil, A. Ferrández, H. Mora-Mora, and J. Peral, "Internet of Things: a review of surveys based on context aware intelligent services," *Sensors*, vol. 16, no. 7, 1069: 1–23, 2016, Available: www.mdpi.com/journal/sensors. DOI: 10.3390/s16071069.

[4] S. Kausha and R.K. Sharma, "Modeling and analysis of adaptive buffer sharing scheme for consecutive packet loss reduction in broadband networks," *International Journal of Electrical and Electronics Engineering*, vol. 4, no. 3, pp. 428—435, 2010.

[5] I. Awan, M. Younas, and W. Naveed, "Modeling QoS in IoT applications," in *Proc. of the* 17^{th} *International Conference on Network-Based Information Systems*, (September 10–12, 2014, Salerno, Italy), 2014, pp. 99-–105.

[6] R.C. Bhaddurgatte and V. Kumar, "A review: QoS architecture and implementations in IoT environment," *Research & Reviews: Journal of Engineering and Technology*, S1, pp. 6—12, 2015.

[7] S.P. Eswaran, V. Ariharan, and J. Bapat, "Event driven opportunistic communication enabler for smart city," in *Proc. of the* 8^{th} *IEEE International Conference on Next Generation Mobile Apps, Services and Technologies*, (September 10-–12, 2014, Oxford, UK), 2014, pp. 313-–319.

[8] R. Sharma, N. Kumar, N.B. Gowda, and T. Srinivas, "Probabilistic prediction based scheduling for delay sensitive traffic in Internet of Things," *Procedia Computer Science*, vol. 52, pp. 90—97, 2015.

[9] S.P. Eswaran and J. Bapat, "Service centric Markov based spectrum sharing for Internet of Things (IoT)," in *Proc. of the 2015 IEEE Region 10 Symposium*, (May 13-15, 2015, Ahmedabad, India), 2015, pp. 9–12.

[10] T.T. Zin, P. Tin, and H. Hama, "Characterizing reliability measure for Internet of Things by Markov queue," *Data Science and Pattern Recognition*, vol. 2, no. 2, pp. 1–10, 2018. Available: http://www.ikelab.net/dspr-pdf/vol2-2/dspr-paper1.pdf.

[11] O. Salameh, M. Awad, and F. AbuAlrub, "A Markovian model for Internet of Things applications," *International Journal of Computer Networks & Communications*, vol. 11, no. 2, pp. 113–124, 2019.

[12] N. Chee-Hock and S. Boon-Hee, *Queueing Modelling Fundamentals. With Applications in Communication Networks*, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England: John Wiley & Sons Ltd, 2008.

[13] I. Candan, and M. Salameh, "Analytical modeling of a time-threshold based bandwidth allocation scheme for cellular networks," *Computer Communication*, vol. 30, no. 5, pp. 1036–1043, 2007.

Volodymyr G. Skobelev, Volodymyr V. Skobelev Received February 20, 2020

Volodymyr G. Skobelev
V.M. Glushkov Institute of Cybernetics of NAS of Ukraine
40 Glushkova ave., Kyiv, Ukraine, 03187
Phone: +38 063 431 86 05
E–mail: skobelevvg@gmail.com

Volodymyr V. Skobelev
V.M. Glushkov Institute of Cybernetics of NAS of Ukraine
40 Glushkova ave., Kyiv, Ukraine, 03187
Phone: +38 066 276 85 72
E–mail: volodimirvskobelev@gmail.com

Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

Apportionment "Population paradox" and the Paradox of population influence

Ion Bolun

Abstract

A new approach is grounded with respect to the population paradox (PPr). Further on, the paradox of population influence (PPi) is proposed. It is proven that Hamilton method is immune to the PPi, and that d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods – are not. By computer simulation, the percentage of non-immunity of Hamilton method to PPr, and the one of d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë divisor methods to PPi, is estimated. For a large range of initial data, this percentage, in the case of the four investigated divisor methods, does not exceed, on average, 0.6-0.8%, that is one case per a total of 120-170 cases.

Keywords: apportionment method, population paradox, paradox of population influence, computer simulation, comparative analyses.

MSC 2010: 62P25, 91B10, 91B12, 68U20

1 Introduction

The population paradox (PPr) was identified and defined in the early 1900s in the case of Hamilton apportionment method [1-3]. Initially, the apportionment of seats in the US House of Representatives at population growth was found to be inappropriate. Later on, the paradox became one of the key restrictions in the apportionment of elective bodies' mandates among parties based on voting results, and sometimes – in the distribution of discrete identical goods among beneficiaries.

In this paper, the reasonable area of coverage by PPr of multiple situations is investigated and a new approach, covering all possible situations, is proposed. In addition, the frequency of non-immunity to population paradox (in its traditional formulation and in the proposed one) of some well-known apportionment methods is estimated by computer simulation.

2 Aspects of Population paradox

There are many definitions of the Population paradox, but their essence is usually the same, for example:

1) "the population paradox occurs when state A loses a seat to state B even though the population of A grew at a higher rate than the population of B" [3];

2) "The population paradox occurs when, based on updated population figures, a reapportionment of a fixed number of seats causes a state to lose a seat to another state, although the percent increase in the population of the state that loses the seat is bigger than the percent increase in the population of the state that gains the seat" [4].

Thus, if Hamilton method [2, 3] had been used in 1901 to reallocate 386 seats in the US Congress House of Representatives, Virginia, at a higher rate of population increase than that of Maine, would have lost a seat, and Maine would have won a seat (Table 1) [5].

On the basis of formulations from [3, 4], the Population paradox is formalized in Definition 1, where for the first apportionment the following notations are used:

M – total number of seats;

n – number of states. Consider $n \leq M$;

V – total population for n states;

V_i – population of state i, $i = \overline{1,n}$;

x_i – number of seats to be allocated to state i, $i = \overline{1,n}$.

Table 1. The Population paradox for states Maine and Virginia [5]

State	Population		Population growth		Seats	
	1900	1901	rate,%	abs., pers.	1900	1901
Maine	694466	699114	1.0067	4648	3	4
Virginia	1854184	1873951	1.0107	19767	10	9

For the second apportionment, here and further on, the nominated and other notations will be completed with the apostrophe symbol ($'$), for example V' for V.

Definition 1. *The Population paradox (PPr) occurs, if at* $M' = M$, $n' = n$ *and*

$$V'_k/V_k > V'_j/V_j \tag{1}$$

the following relations take place

$$x'_k = x_k - 1, \tag{2}$$

$$x'_j = x_j + 1. \tag{3}$$

In this definition, there are doubts about the correctness of condition (1). Let's consider Example 1.

Example 1. *on Population paradox according to Definition 1 when the Hamilton method is applied. Let* $M = 100$, $n = 4$ *and, for the first apportionment,* $V_1 = 50000$, $V_2 = 1400$, $V_3 = 1300$ *and* $V_4 = 800$, *and for the second one:* $V'_1 = 50800$, $V'_2 = 1421$, $V'_3 = 1326$ *and* $V'_4 = 816$. *Then for the first apportionment we have* $Q = (50000 + 1400 + 1300 + 800)/100 = 535$, *and for the second one* $Q' = (50800 + 1421 + 1326 + 816)/100 = 543.63$. *Here* $Q = V/M$ *is the standard divisor, known also as Hare quota [2, 7]. The results of other calculations are shown in Table 2.*

In Table 2 and further on, the notation $R_i = V_i/Q$ is used, where R_i is the standard quota [3].

Table 2. Results of calculations to Example 1

State i	$Q = 535$			$Q' = 543.16$				
	V_i	R_i	x_i	V_i'	R_i'	x_i'	V_i'/V_i	$V_i' - V_i$
1	50000	93.46	93	50800	93.53	94	1.016	800
2	1400	2.62	3	1400	2.58	3	1	0
3	1300	2.43	2	1300	2.39	2	1	0
4	800	1.50	2	816	1.50	1	1.020	16
Total	53500		100	54316		100		

According to Table 2, the Hamilton solution for the second apportionment is to withdraw a seat from state 4 and reallocate it to state 1, in spite of the fact that the population growth rate of state 4 is 2%, while that of state 1 is 1.6%, i.e. smaller. Thus, according to Definition 1, the population paradox (PPr) would occur.

At the same time, one can mention that, in absolute terms, the population of state 4 has increased by only 16 inhabitants, while that of state 1 — by 800 inhabitants, that is, $800/16 = 50$ times more. Moreover, we have $800 > Q' = 543.16$, that is, at a proportional allocation of seats, more than one seat would correspond to the new population of state 1, while much less than one seat ($16 << Q' = 543.16$) – to the new population of state 4. Under such conditions, one cannot affirm that a paradox occurs. Therefore, not in all cases Definition 1 specifies situations of population paradox. Respectively, it would be appropriate to redefine the paradox situations implied by changes in the number of inhabitants.

3 Redefining the Population paradox

At a first glance, there are several aspects and, respectively, alternatives of comparison of two consecutive apportionments with a view to identifying situations in which population paradox occurs. In addition to the use of "population rate deviation" as a comparison criterion from one apportionment to the next one (Definition 1), the "population ab-

solute deviation" and the "absolute deviation of population influence power" are also discussed under this section as criteria.

3.1 Population absolute deviation as criterion

For the comparing of two consecutive apportionments, the decrease ($x'_k - x_k < 0$, see (2)) or the increase ($x'_j - x_j > 0$, see (3)), for each particular state, of the number of seats x_i from one apportionment to the next one, are used. At the same time, because the distribution of seats among states must be made proportionately to the number of inhabitants (V_i, $i = \overline{1,n}$, and respectively, V'_i, $i = \overline{1,n}$), it should be that the absolute increase (decrease) of the number of seats $x'_i - x_i$ be correlated with the absolute increase (decrease) of population $V'_i - V_i$, and not with its rate V'_i/V_i. That is, when defining the population paradox, instead of condition (1) it would be more appropriate to use the following one:

$$V'_k - V_k > V'_i - V_i. \tag{4}$$

Possibly, it is this approach that is taken into account in Example 2 and the population paradox definition from [6]: "The population paradox occurs when one state loses a seat and another state gains a seat, even though the first state's population increased more than the second state's population".

Example 2. [6] *Let $M = 25$, $n = 3$ (states A, B and C) and, for the first apportionment, $V_A = 13$, $V_B = 12$ and $V_C = 112$, and for the second one: $V'_A = 14$, $V'_B = 12$ and $V'_C = 114$. The results of some calculations, when applying the Hamilton method, are shown in Table 3.*

Data of Table 3 show that state A, the population of which increased by 1 mil, gains a seat and state C, with a 2 mil population increase, loses a seat. Based on these results, in [6] it is concluded that the Population paradox occurs. But this example is not one of Population paradox in sense of Definition 1, given that the population growth rate is of 1.077 in case of state A and of 1.018 – in the case of state C.

Table 3. Calculations for the example from [6], Hamilton method

State	Population, mil		Population growth		Standard quota		Seats	
	App. I	App. 2	Rate	Abs.	App. I	App. 2	App. I	App. 2
A	13	14	1.077	1	2.37	2.50	2	3
B	12	12	1	0	2.19	2.14	2	2
C	112	114	1.018	2	20.44	20.36	21	20
Total	137	140			25	25	25	25

Under condition (4) instead of condition (1), the term of "absolute deviation population paradox" or, shorter, "Absolute population paradox" (PPa) will be used in this paper. Respectively, instead of the traditional term of "Population paradox" (in the meaning of Definition 1), the term "rate deviation population paradox" or, shorter, "Rate population paradox" (PPr) will be used.

Thus, for PPa we have the following definition:

Definition 2. *The Absolute population paradox occurs, if at $M' = M$, $n' = n$ and at constraint (4) the relations (2) and (3) take place.*

The correlation between conditions (1) and (4) of population paradox according to Definition 1 (PPr) and, respectively, Definition 2 (PPa), is of interest.

Statement 1. *The conditions of Definition 1 supplemented with those of*

$$V_k \geq V_j, \tag{5}$$

fall under the conditions of Definition 2.

Indeed, it is sufficient to prove that, if relations (1) and (5) take place, then relation (4) also occurs. Let the conditions (1) and (5) occur. From (1) we have $V'_k/V_k - 1 > V'_j/V_j - 1$, i.e. $(V'_k - V_k)/V_k > (V'_j - V_j)/V_j$ or $(V'_k - V_k) > (V'_j - V_j)V_k/V_j$. The last inequality, if relation (5) takes place, implies that condition (4) occurs. ■

Considering the data of Table 1, it can be easily seen that the paradox regarding the apportionment of seats between states of Virginia and Maine in 1990 and 1991 falls under both conditions of Statement 1 and those of Definition 2.

It can be expected, at the same time, that there are also cases of compliance with the conditions of Definition 2 and non-compliance with the conditions of Statement 1.

Statement 2. *The area of situations, covered by conditions of Definition 2, is larger than the one covered by conditions of Statement 1.*

Indeed, relations (2) and (3) are common to Definition 2 and Statement 1, and, according to Statement 1, the conditions (1) and (5) imply the inequality (4). ▼

It remains to prove that there are also other situations of PPa covered by conditions (2)-(4) but not covered by inequalities (1)-(3), (5); i.e. not in all cases when inequalities (2)-(4) take place, relations (1)-(3), (5) occur as well. One of such cases is determined by conditions (2)-(4) and inequalities

$$V_k > V_j, \tag{6}$$

$$V'_k/V_k < V'_j/V_j, \tag{7}$$

the last inequality being an opposite of relation (1). It is sufficient to show that this case is a real one.

The compatibility of relations (4)-(6) is evident; also, because of (6), relation (4) can be complied with even under condition (7). As to inequalities (2) and (3), it is easier to comply with them in conditions (6) and (7) than in conditions (1) and (5). ■

Consequence 1. *The conditions of Definition 2 cannot be replaced by those of Statement 1.*

Indeed, according to Statement 2 the conditions of Definition 2 are larger than those of Statement 1. ■

The veracity of Statement 2 is confirmed also by Example 3.

Table 4. Results of calculations to Example 3

State i	$Q = 76.5$			$Q' = 78.52$				
	V_i	R_i	x_i	V_i'	R_i'	x_i'	V_i'/V_i	$V_i' - V_i$
1	3400	44.44	45	3502	44.60	44	1.03	102
2	2000	26.14	26	2100	26.74	27	1.05	100
3	1150	15.03	15	1150	14.65	15	1	0
4	1100	14.38	14	1100	14.01	14	1	0
Total	7650		100	7852		100		

Example 3. *on the Absolute population paradox when applying the Hamilton method. Let* $M = 100$*,* $n = 4$ *and the first apportionment be characterized by data* $V_1 = 3400$*,* $V_2 = 2000$*,* $V_3 = 1150$*,* $V_4 = 1100$*, and the second one by:* $V_1' = 3502$*,* $V_2' = 2100$*,* $V_3' = 1150$*,* $V_4' = 1100$*. Then for the first apportionment we have* $Q = (3400 + 2000 + 1150 + 1100)/100 = 76.5$*, and for the second one* $Q' = (3502 + 2100 + 1150 + 1100)/100 = 78.52$*. The obtained results are shown in Table 4.*

In Example 3, for states 3 and 4, the population does not change from the first apportionment to the second one (1150 and 1100, respectively); nor changes the number of mandates. But for states 1 and 2 we have: $V_1' - V_1 = 3502 - 3400 = 102$; $V_2' - V_2 = 2100 - 2000 = 100$; $V_1'/V_1 = 3502/3400 = 1.03$ and $V_2'/V_2 = 2100/2000 = 1.05$. So, $V_1 > V_2$ and $V_1'/V_1 < V_2'/V_2$, i.e. this case is different from the one of relations (1), (5). At the same time, we have $V_1' - V_1 > V_2' - V_2$, $x_1' < x_1$ and $x_2' > x_2$ and according to Definition 2 a PPa occurs. It should also be mentioned that, because of $V_1'/V_1 < V_2'/V_2$, $x_1' < x_1$ and $x_2' > x_2$, the PPr doesn't occur (see Definition 1).

3.2 Population influence power absolute deviation as criterion

The major shortcoming of the approach based on relation (4) is that the total number of inhabitants in the two consecutive apportionments, V and V', is usually different, while the number of seats is the same

($M = M'$). Under such conditions, the power of influence of one inhabitant [8] in the two apportionments, r and r', is different: $r = M/V$ and $r' = M/V'$. Therefore, the use of the number of inhabitants' absolute deviation ($V_i' - V_i$) as a criterion is not correct. For each state, the comparison should not be based on the number V_i of inhabitants, but on the legal power of influence of the decisions of the House of Representatives delegated by the V_i inhabitants [8] $R_i = rV_i = MV_i/V = V_i/Q$, $i = \overline{1,n}$, known also as standard quota. For these reasons, at $c = V'/V$, further on there are formulated and characterized cases (a), (b) and (c) -– claimants in defining the population paradox; at a first glance, the population paradox would occur if:

a) at $R_i' \geq R_i$, relation $x_i' < x_i$ would also occur;

b) at $R_k' - R_k > R_j' - R_j$, relations (2) and (3) would also occur;

c) at $R_k' - x_k > R_j' - x_j$, relations (2) and (3) would also occur.

Case (a) outlines the conditions, for a particular state (i) within the two apportionments, needed for the population paradox to take place. In the other two cases, (b) and (c), the identification of population paradox is based on comparing the characteristics of two states; at the same time, the respective relations also contain, as further on will be ascertained, a parameter ($c = V'/V$ or $Q' = V'/M$) that refers to the entire apportionment.

The first, out of the two conditions of case (a), reflects the following situation: if the legal power of influence of state i (R_i), delegated by its population (V_i), does not decrease, then the power of influence of this state on the House of Representatives decisions, determined by the number of seats allocated to it (x_i), should not decrease as well. This condition can also be represented in another form. We have $R_i' \geq R_i$, that is $MV_i'/V' \geq MV_i/V$, implying $V_i'/V' \geq V_i/V$ or $V_i' \geq cV_i$.

The first condition of case (b) refers to the following situation: if the increase of the legal power of influence of state k, delegated by its population in two consecutive apportionments, is greater than the one of state j, then no seats should be taken from state k to be reallocated

to state j. This condition can also be represented in another form. We have $R'_k - R_k > R'_j - R_j$, that is $MV'_k/V' - MV_k/V > MV'_j/V' - MV_j/V$, implying $VV'_k - V'V_k > VV'_j - V'V_j$ or $VV'_k - VV'_j > V'V_k - V'V_j$, therefore $V'_k - V'_j > c(V_k - V_j)$.

Finally, the essence of the first condition of case (c): if the increase of the legal power of influence of state k in the House of Representatives, determined by the x_k seats assigned to it according to the first apportionment, is greater than the one of state j, then no seats should be taken from state k to be reallocated to state j. This condition can also be represented in another form. We have $R'_k - x_k > R'_j - x_j$, that is $MV'_k/V' - x_k > MV'_j/V' - x_j$, implying $(V'_k - V'_j)/Q' > x_k - x_j$ or $V'_k - V'_j > Q'(x_k - x_j)$.

Let's consider these three cases. When comparing the two apportionments, there can be two approaches:

1) only states whose number of seats has been modified are taken into account;

2) all states, including those whose number of seats has not changed, are taken into account.

In this paper, approach 1 (the traditional, well known one, also used for PPr) is applied.

The advantage of case (a), of the (a) – (c) described above, is that the paradoxical situation is found in the entire apportionment, independently of any particular state. But this case not always specifies a paradox. For example, it may occur that a seat from state k is taken over by a state j, the power of influence of which has increased more than the power of influence of state k. Thus, case (a) would be a paradox only if a seat from state k was taken over by a state j, the power of influence of which has increased less than the one of state k, that is, only if case (b) occurs. So, case (a) would be a paradox only if case (b) would take place; given this, it is excluded.

For this reason, only cases (b) and (c) remain. Out of these, only case (b) fully corresponds to the requirements of population paradox for reasons described further on. As a basis of comparison regarding the

first apportionment, the legal power of influence of each state delegated by its population, and not the one, determined by the number of seats allocated to state in the first apportionment, should be used. That is, condition $R'_k - R_k > R'_j - R_j$ should be used and not the $R'_k - x_k > R'_j - x_j$ one. The use of x_i instead of R_i, if $V_i \neq a_i Q$ (which usually takes place), favors (at $x_i = a_i + 1$) or disfavors (at $x_i = a_i$) state i. Also, when using PPr (see Definition 1), the comparison is made with V_i and not with x_i.

In case (b), in order to distinguish it from the already broadly used term "Population paradox" (see Definition 1) and also from the proposed new term for the last "Paradox of population rate" (PPr), in this paper the term "paradox of population influence absolute deviation" or, shorter, "Paradox of population influence" (PPi) will be used.

Thus, for PPi we have the definition below.

Definition 3. *The Paradox of population influence occurs, if at* $M' = M$, $n' = n$ *and*

$$V'_k - V'_j > V'(V_k - V_j)/V \tag{8}$$

the relations (2) and (3) take place.

3.3 Essential comparison of PPr, PPa and PPi approaches

The comparison by essence of PPr, PPa and PPi approaches regarding the population paradox can be made based on Definitions 1–3.

Regarding Definition 1 and Definition 2, the latter, as an approach, is closer to reflecting the conditions of population paradox manifestation, because it takes into account the absolute deviation of the number of votes $(V'_i - V_i)$, which is more appropriate for comparing the increase/decrease of the number of mandates from one poll to the next $(x'_i - x_i)$ than the ratio V'_i/V_i.

On the other hand, as it is shown in Section 3.2, the deviation $V'_i - V_i$ does not take into account the fact that the power of influence of an inhabitant in the two apportionments, r and r', differs. On the contrary, the relation (1), which is equivalent to the $R'_k/R_k >$

R'_j/R_j one, takes this fact into account. Indeed: we have $R_i = rV_i = MV_i/V$, implying $V_i = VR_i/M$, and similarly, $V'_i = R'_iV'/M$. So, $V'_k/V_k = R'_kV'/R_kV = cR'_k/R_k$ and, respectively, $V'_j/V_j = cR'_j/R_j$, Q.E.D. Therefore, from this point of view, Definition 1 better reflects the conditions of population paradox manifestation.

Thus, as regards Definitions 1 and 2, in one aspect (the power of influence of an inhabitant) – the PPr approach (Definition 1) is better, but in another aspect (the absolute deviation) – the PPa approach (Definition 2) is better. Definition 3 (Paradox of population influence – PPi) covers both these aspects; by comparison, this one uses pairs of states, and not separate states, and also for each distinct state i:

1) not the number of inhabitants V_i, but the power of influence of the elective body decisions R_i, delegated by the population V_i;

2) not the ratio V'_i/V_i, but, given the reasons mentioned in Section 3.2, the absolute deviation $D'_i - D_i$, suitable for comparing the increase/decrease of the number of seats x_i from one apportionment to the next one.

So, Definition 3 reflects more appropriately the population paradox manifestation and therefore the PPi approach is the only one that should be used for this purpose.

Statement 3. *At $V = V'$, the conditions of Absolute population paradox (Definition 2, PPa) and those of Paradox of population influence (Definition 3, PPi) coincide.*

Indeed, from (8), taking into account that $V = V'$, we have $V'_k - V'_j > V'(V_k - V_j)/V = V_k - V_j$ or $V'_k - V_k > V'_i - V_i$, that coincides with the (4) one. ■

4 Immunity of Hamilton method to PPr, PPa and PPi

For the population paradox within the meaning of Definition 3 (Paradox of population influence — PPi), Statement 4 takes place.

Statement 4. *Hamilton method is immune to the Paradox of population influence.*

Indeed, let's consider that condition (8) occurs. Taking into account that $Q' = cQ$ and $V_i = a_iQ + \Delta V_i$, where $a_i = \lfloor V_i/Q \rfloor$, relation (8) takes the form

$$a'_kQ' + \Delta V'_k - (a'_jQ' + \Delta V'_j) > c[a_kQ + \Delta V_k - (a_jQ + \Delta V_j)]$$

or

$$Q'[a'_k - a'_j - (a_k - a_j)] > c(\Delta V_k - \Delta V_j) + \Delta V'_j - \Delta V'_k. \qquad (9)$$

Obviously, the easiest case for state j to take over a seat from state k is: $x_k = a_k+1$, $x_j = a_j$, $x'_k = a'_k = a_k$ and $x'_j = a'_j+1 = a_j+1$, which can only be if $\Delta V_k > \Delta V_j$ and $\Delta V'_j > \Delta V'_k$. Thus, considering that $c > 0$, we have $c(\Delta V_k-\Delta V_j)+\Delta V'_j-\Delta V'_k > 0$ and $Q'[a'_k-a'_j-(a_k-a_j)] = 0$, that is condition (9) doesn't occur. Thus, conditions (2), (3) and (9) cannot occur simultaneously and, respectively, neither do conditions (2), (3) and (8). ■

In the context of Statement 1, let's examine two known examples of non-immunity of Hamilton method to the PPr, taken from [9, 10].

Example 4. *Let $M = 11$, $n = 3$, $V = 1000$ and $V' = 1100$. Other initial data taken from [9] and the results of calculations using Hamilton method are shown in Table 5.*

Table 5. Results of calculations to Example 4

State	V_i	V'_i	x_i	x'_i	V'_i/V_i	V'_i-V_i	R'_i-R_i	PPr/PPa/PPi
A	54	56	0	1	1.037	2	-0.034	Yes/Yes/No
B	243	255	3	2	1.049	12	-0.123	
C	703	789	8	8	1.122	86	0.157	

Data of Table 5 show that for states A and B the PPr and PPa paradoxes occur, but the PPi one doesn't. Even though the population of states A and B increased, and that of B increased more than that of A, their influence power (R_A and R_B) decreased and R_B decreased

more (-0.123) than R_A (-0.034). So, the loss of a seat by state B to state A is not a paradox.

Example 5. *Let $M = 10$, $n = 3$, $V = 10000$ and $V' = 9500$. Other initial data taken from [10] and the results of calculations using Hamilton method are systemized in Table 6.*

Table 6. Results of calculations to Example 5

State	V_i	V_i'	x_i	x_i'	V_i'/V_i	$V_i'-V_i$	$R_i'-R_i$	PPr/PPa/PPi
1	1450	1470	2	1	1.014	20	0.097	
2	3400	3380	3	4	0.994	-20	0.158	Nes/Yes/Yo
3	5150	4650	5	5	0.903	-500	-0.255	

Data of Table 6 show that for states A and B the PPr and PPa paradoxes occurs, but the PPi one doesn't. Even though the population of state A increased and that of state B decreased, their influence power (R_A and R_B) increased, and R_B increased more (0.158) than R_A (0.097). So, the loss of a seat by state A to state B is not a paradox.

Although it is not the PPr, but the PPi that adequately portrays the situations of population paradox (and the Hamilton method is immune to PPi), the frequency of non-immunity of the Hamilton method to PPr and PPa is of a certain interest.

For this purpose, the SIMAP application for computer simulation has been specially developed and used.

The total number of votes for the second poll V' is determined as $V' = cV$. The values V_i', $i = 1, 2, \ldots, n$ for the second poll are random sizes determined as $V_i' = p_i V_i$, $i = 1, 2, \ldots, n$, with corrections required to make $V' = V_1' + V_2' + \ldots + V_n'$, where p_i is a stochastic size of uniform distribution in the range $[(c-1)(1-d); (c-1)(1+d)]$, $c < 2$, and d is a constant. For each variant of initial data (M, n, p and d) here and further on there was used a sample of 1 million random alternatives.

The character of percentages P_{Pr} and P_{Pa} dependence on n and d at $M = 101$ and $p = 0.02$ for the Hamilton method ($P_{Pr}(H)$ and $P_{Pa}(H)$) can be seen in Figure 1 and Figure 2. For $6 \leq M \leq 501$,

$3 \le n \le 50$, $0.02 \le p \le 0.1$ and $0.1 \le d \le 1$, $n < M$, the relations $0.018\% \le P_{Pr}(H) \le 4.66\%$ and $0.077\% \le P_{Pa}(H) \le 78.58\%$ occur.

In all cases, for the same values of initial data (M, n, p and d), the relation $P_{Pr}(H) < P_{Pa}(H)$ occurs; also the difference $P_{Pa}(H) - P_{Pr}(H)$, at $6 \le M \le 501$, $2 \le n \le 50$, $0.02 \le p \le 0.1$ and $0.1 \le d \le 1$, $n < M$, is increasing with the increase of M, p and d and, in most cases, with the increase of n, but the ratio $P_{Pa}(H)/P_{Pr}(H)$ is decreasing on n and d (Figure 3).

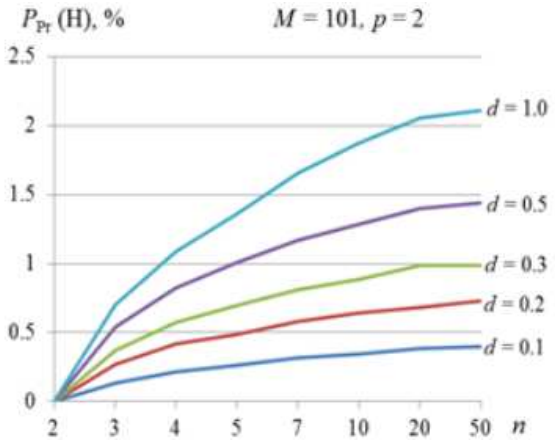

Figure 1. Dependence of P_{Pr} on n and d for Hamilton method

5 The non-immunity of d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods to the Paradox of population influence

The well-known d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods [2, 3, 11] are not immune to the PPi paradox. Examples 6–9 for each of the four methods are done below.

Example 6. *Let* $M = 101$, $n = 5$, $V = 10640$ *and* $V' = 10562$. *Other*

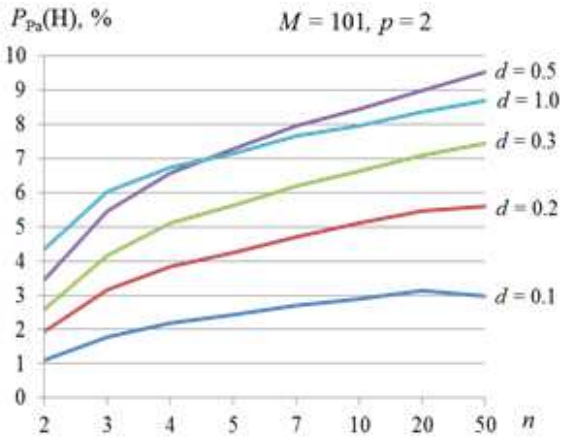

Figure 2. Dependence of P_{Pa} on n and d for Hamilton method

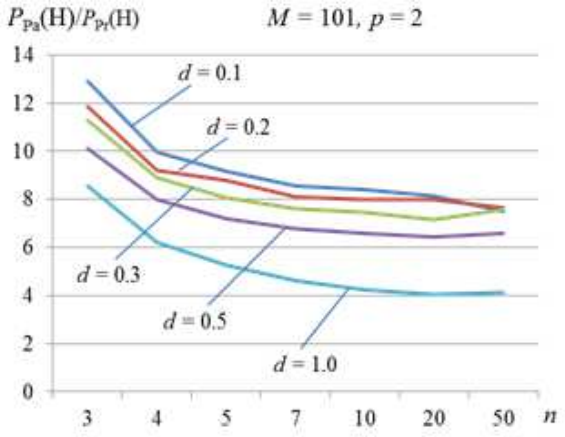

Figure 3. Dependence of P_{Pa}/P_{Pr} on n and d for Hamilton method

initial data and the results of calculations using the d'Hondt method are systemized in Table 7.

Table 7. Results of calculations to Example 6

State	V_i	V_i'	x_i	x_i'	V_i'/V_i	$V_i'-V_i$	$R_i'-R_i$	PPr/PPa/PPi
A	9900	9800	94	95	0.990	-100	-0.262	
B	210	220	2	2	1.048	10	-0.110	
C	210	220	2	2	1.048	10	-0.110	No/Yes/Yes
D	210	220	2	2	1.048	10	-0.110	
E	110	102	1	0	0.927	-8	-0.069	

Data of Table 7 show that for states A and E the PPi paradox occurs. Even though the power of influence of states A and E (R_A and R_E) decreased, and notwithstanding the fact that the power of influence of state A decreased more than that of state E ($R_E - R_E' = 0.069 < 0.262 = R_A - R_A'$), state E lost a seat in favor of state A. To be mentioned that, for this particular example, a similar situation occurs in relation with criterion $V_i' - V_i$ (see Table 7).

Example 7. *Let $M = 101$, $n = 5$, $V = 8800$ and $V' = 8873$. Other initial data and the results of calculations using Sainte-Laguë method are systemized in Table 8.*

Table 8. Results of calculations to Example 7

State	V_i	V_i'	x_i	x_i'	V_i'/V_i	$V_i'-V_i$	$R_i'-R_i$	PPr/PPa/PPi
A	8300	8230	94	95	0.992	-70	-1.581	
B	150	200	2	2	1.333	50	0.555	
C	150	200	2	2	1.333	50	0.555	No/Yes/Yes
D	150	200	2	2	1.333	50	0.555	
E	50	43	1	0	0.860	-7	-0.084	

Data of Table 8 show that for states A and E the PPi paradox occurs. Even though the influence power (R_A and R_E) of states A and E decreased, and those of state A decreased more than those of state E ($R_E - R'_E = 0.084 < 1.581 = R_A - R'_A$), state E lost a seat to state A. To mention that for this example a similar situation is with criterion $V'_i - V_i$ (see Table 8).

Example 8. *Let $M = 101$, $n = 5$, $V = 8675$ and $V' = 8720$. Other initial data and the results of calculations using Huntington-Hill method are systemized in Table 9.*

Table 9. Results of calculations to Example 8

State	V_i	V'_i	x_i	x'_i	V'_i/V_i	$V'_i - V_i$	$R'_i - R_i$	PPr/PPa/PPi
A	8130	8120	93	94	0.999	-10	-0.604	
B	140	160	2	2	1.143	20	0.223	
C	140	160	2	2	1.143	20	0.223	No/Yes/Yes
D	140	160	2	2	1.143	20	0.223	
E	125	120	2	1	0.960	-5	-0.065	

Data of Table 9 show that for states A and E the PPi paradox occurs. Even though the influence power (R_A and R_E) of states A and E decreased, and those of state A decreased more than those of state E ($R_E - R'_E = 0.065 < 0.604 = R_A - R'_A$), state E lost a seat to state A. To mention that for this example a similar situation is with criterion $V'_i - V_i$ (see Table 9).

Example 9. *Let $M = 101$, $n = 5$, $V = 8685$ and $V' = 8755$. Other initial data and the results of calculations using Adapted Sainte-Laguë method are systemized in Table 10.*

Data of Table 10 show that for states A and E the PPi paradox occurs. Even the influence power (R_A and R_E) of states A and E decreased, and those of state A decreased more than those of state E ($R_E - R'_E = 0.128 < 2.256 = R_A - R'_A$), state E lost a seat to state A.

Table 10. Results of calculations to Example 9

State	V_i	V_i'	x_i	x_i'	V_i'/V_i	$V_i'-V_i$	$R_i'-R_i$	PPr/PPa/PPi
A	8130	8000	93	94	0.984	-130	-2.256	
B	140	210	2	2	1.5	70	0.795	
C	140	210	2	2	1.5	70	0.795	No/Yes/Yes
D	140	210	2	2	1.5	70	0.795	
E	135	125	2	1	0.926	-10	-0.128	

To mention that for this example a similar situation is with criterion $V_i' - V_i$ (see Table 10).

It is of interest how often the d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods are not immune to the Paradox of population influence. With this aim, the SIMAP computer application was used, for the same initial data as for the Hamilton method in Section 4. The character of percentages P_{Pi} dependence on n and d for $M = 101$ and $p = 0.02$ for the d'Hondt ($P_{Pi}(dH)$), Sainte-Laguë ($P_{Pi}(SL)$), Huntington-Hill ($P_{Pi}(dH)$) and Adapted Sainte-Laguë ($P_{Pi}(ASL)$) methods can be seen in Figures 4–7.

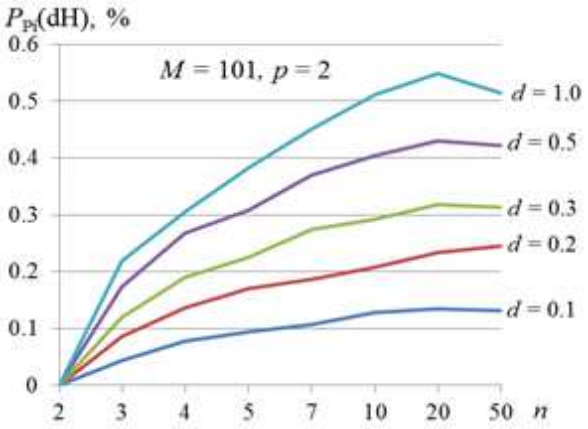

Figure 4. Dependence of P_{Pi} on n and d for d'Hondt method

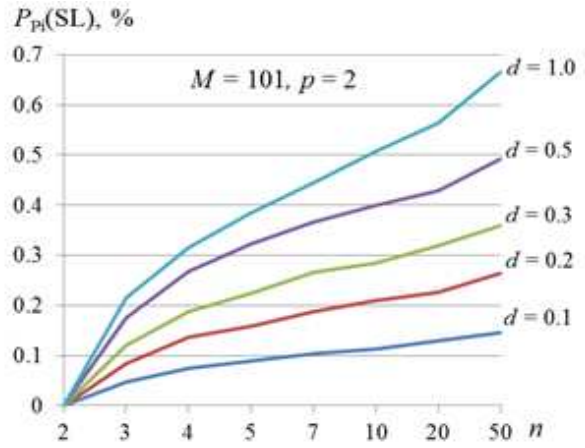

Figure 5. Dependence of P_{Pi} on n and d for Sainte-Laguë method

From Figures 4–7 one can see that, in terms of the percentage of non-immunity to the Paradox of population influence, there are no essential differences between the d'Hondt and Sainte-Laguë methods, nor between the Huntington-Hill and the Adapted Sainte-Laguë methods. Additional comparative data show that, from the point of view of immunity to PPi, in some cases the d'Hondt method is better than the Sainte-Laguë one and vice-versa; but the difference is not significant. The same situation is in the case of comparative analyses of Huntington-Hill and Adapted Sainte-Laguë methods.

Calculations performed show also that for $6 \leq M \leq 501$, $3 \leq n \leq 50$, $0.02 \leq p \leq 0.1$ and $0.1 \leq d \leq 1$, $n < M$, the following relations occur: $0.003\% \leq P_{Pi}(dH) \leq 0.652\%$ ($M = 501$, $n = 50$, $p = 0.1$, $d = 0.1$), $0.003\% \leq P_{Pi}(SL) \leq 0.806\%$ ($M = 101$, $n = 50$, $p = 0.1$, $d = 1$), $0.001\% \leq P_{Pi}(HH) \leq 0.617\%$ ($M = 501$, $n = 50$, $p = 0.1$, $d = 0.1$) and $0.001\% \leq P_{Pi}(ASL) \leq 0.621\%$ ($M = 501$, $n = 50$, $p = 0.1$, $d = 0.1$). So, for the specified range of initial data, the non-immunity to the Paradox of population influence, when using one of these four apportionment methods, does not exceed, on average, $0.6 - 0,8\%$, that is one case per 120-170 cases in total. For example, no one such case has been identified for the 11 apportionments of seats

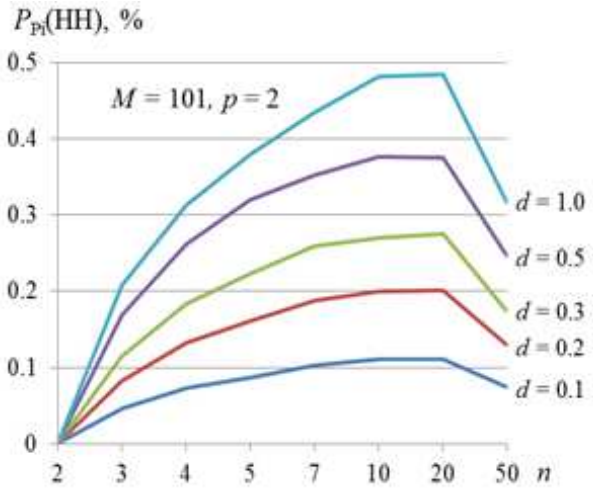

Figure 6. Dependence of P_{Pi} on n and d for H.-Hill method

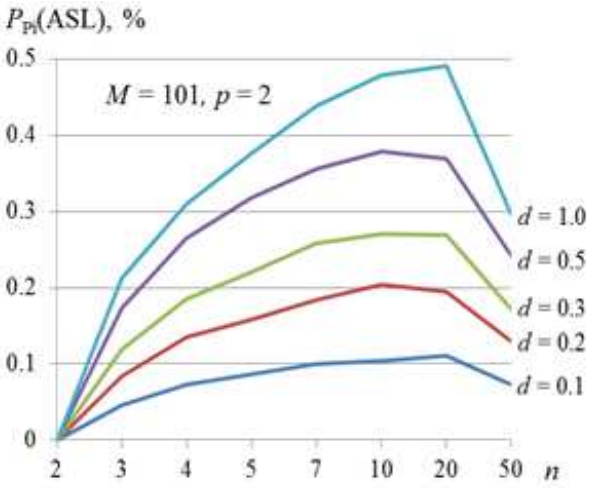

Figure 7. Dependence of P_{Pi} on n and d for ASL method

in the US Congress House of Representatives in 1900-2010 years, made applying the Webster (1900, 1910 and 1930 Census) and Huntington-Hill (1940-2010 Census) methods.

6 Conclusions

The well-known population paradox (PPr), based on the population deviation rate from one apportionment to the next one, is not always a true paradox. A new formulation of conditions of population paradoxical situations is proposed, which is based on the absolute deviation of the population power of influence. In order to distinguish it from the traditional one, for this particular case the term "Paradox of population influence" (PPi) is used. Of course, it would be better to use the term "population paradox", but in the new formulation. Thus, we count on the fact that the use of term PPi is temporary.

It is well known that the Hamilton method is not immune to the Population paradox (PPr), whereas the d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë divisor methods are [1–3, 11]. As for the Paradox of population influence (PPi), the situation is opposite: the Hamilton method is, and the d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods are not immune to it.

By computer simulation using the SIMAP application, the percentage of non-immunity of Hamilton method to PPr ($P_{Pr}(H)$) and PPa ($P_{Pa}(H)$), and of d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë methods to PPi ($P_{Pi}(dH)$, $P_{Pi}(SL)$), $P_{Pi}(HH)$ and $P_{Pi}(ASL)$, respectively) is estimated. It has been found that, for $6 \leq M \leq 501$, $3 \leq n \leq 50$, $0.02 \leq p \leq 0.1$ and $0.1 \leq d \leq 1$, $n < M$, the following relations occur:

- $0.018\% \leq P_{Pr}(H) \leq 4.66\%$;
- $0.077\% \leq P_{Pa}(H) \leq 78.58\%$;
- $0.003\% \leq P_{Pi}(dH) \leq 0.652\%$;

- $0.003\% \leq P_{Pi}(SL) \leq 0.806\%$;
- $0.001\% \leq P_{Pi}(HH) \leq 0.617\%$;
- $0.001\% \leq P_{Pi}(ASL) \leq 0.621\%$.

To mention that in all cases, for same values of initial data (M, n, p and d), the relation $P_{Pr}(H) < P_{Pa}(H)$ occurs. Also, for the specified range of initial data, the percentage of non-immunity to the Paradox of population influence, when using one of the four examined divisor methods (d'Hondt, Sainte-Laguë, Huntington-Hill and Adapted Sainte-Laguë), does not exceed, on average, 0.6-0,8%, that is one case per 120-170 cases in total.

References

[1] M.L. Balinski and H.P. Young, *Fair Representation: Meeting the Ideal of One Man, One Vote*, 2nd ed. Washington, DC, USA: Brookings Institution Press, 2001, 195 p. ISBN-10: 081570111X. ISBN-13: 978-0815701118.

[2] Ch.M. Biles, "Congressional Apportionment: A Liberal Arts Perspective," Congressional Apportionment. 24, 2016, Humboldt State University. [Online]. Available: http://digitalcommons.humboldt.edu/ apportionment/24. Accessed on: 20.09.2019.

[3] P. Tannenbaum, *Excursion in Modern Mathematics*, 9th ed. USA: Pearson Education, Inc., 2017, 598 p.

[4] Ch.D. Miller, V.E. Heeren, J. Hornsby, and Ch. Heeren, *Mathematical Ideas*, 14th ed. USA: Pearson Education, Inc., 2019, 1032 p.

[5] H. Nurmi, *Voting Paradoxes and How to Deal With Them*, Berlin Heidelberg: Springer-Verlag, 1999, 145 p.

[6] R.T. Koether, Apportionment Paradoxes (Hampden-Sydney College), Mon, Mar 2, 2015. [Online]. Available: http://people.hsc.edu/faculty-staff/robbk/Math111/Lectures/Spring%202015/Lecture%2018%20-%20Apportionment%20Paradoxes.pdf. Accessed on: 20.09.2019.

[7] U. Kohler and J. Zeh, "Apportionment methods," *The Stata Journal*, vol. 12, no. 3, pp. 375—392, 2012.

[8] I. Bolun, "Seats allocation in party-list elections," *Economica*, vol. 76, no. 2, pp. 138–151, 2011.

[9] M.Inigo, J. Jameson, K. Kozak, M. Lanzetta, and K. Sonier, *College Mathematics for Everyday Life*, 2nd ed. USA: Coconino Community College in Flagstaff, 2016, 393 p.

[10] E.A. Robinson and D.H. Ullman, *A Mathematical Look at Politics*, USA: CRC Press, 2010, 480 p.

[11] I. Bolun and A. Costas, "Computer Simulation of Multi-optional Decisions," in *Workshop on Foundations of Informatics, Proc. FOI-2015*, (August 24-29, 2015. Chisinau), 2015, pp. 342–360.

Ion Bolun,

Technical UniversiUy of Moldova
Studentilor, 9/7, Chisinau, Moldova
Phone: +373 22509908
E-mail: ion.bolun@isa.utm.md

Received December 3, 2019

Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

Process Modeling and Extraction of Patterns of Computer Crimes Using Data Mining

Abbas Karimi, Saber Abbasabadei, Javad Akbari Torkestani, Faraneh Zarafshan

Abstract

The main purpose of this research is to model the process and extract patterns of computer crime using data mining and employing MATLAB software for data modeling. The results have been simulated and presented graphically. The simulation results show that this system can be considered as one of the most effective and lowest cost ways to identify the cyber-criminal behavior, therefore, computer crime experts can run effectively this model on their systems.

Keywords: Computer Crimes, Data Mining, Neural Network.

1 Introduction

Nowadays, the cybercrime has become a global problem due to the progress of information and communication technology. Two factors have been identified as the main variables to predict the rate of cybercrime: the rate of computer use and membership in social networks. Computer and electronic crimes, depending on its nature are divided into three categories including: hacking, phishing and identity fraud. Hacking means an attempt to exploit a computer system or a private network inside a computer; phishing is the fraudulent attempt to obtain sensitive information such as usernames, passwords and credit card details by disguising oneself as a trustworthy entity in an electronic communication; identity fraud is the use by one person of another person's

personal information, without authorization and making huge profits from it [2]. Cybercrime analysis can be performed on the computer networks in different conditions. Analysis of cybercrime reports can be carried out in different conditions to identify quickly the offenders by police and preventing those by the detention of offenders and their mobile phones [3]. Abnormal traffic conditions in computer networks may mean that a computer has been hacked, and the sensitive information has been sent to an unauthorized destination, or existence of abnormalities in transactions data of credit cards may cause to hack the identifications and credit card. These problems can be solved by various methods of analyzing such crimes [4]. In the era of information and communication technology, many organizations, companies, etc. use databases for their commercial, educational and statistical affairs to make decisions and providing different reports for their managers, planners and researchers. These organizations use data mining (the quick and accurate discovery of information) for their scientific, technical and economic development. New techniques for data mining such as clustering, classification etc. can be used in various educational fields or other cases. Also, decision makers of the organizations employ the high-level data mining to introduce and analyze data. To develop the classification of computer crimes, one can use the models which are controlled by the machine learning algorithms such as the neural network, and these algorithms are applied in various processes of data mining, such as text mining [5]. Compression between the text mining process and other text processing techniques shows that the data mining algorithms are used for activities such as data collecting, preparing and extracting the knowledge and words required, improving the business value level, facilitating decision making process and also cost reduction. The data mining method can be used to identify the relationships among complicated data related to internet theft or violations of cyberspace laws using the existing databases and data mining algorithms. In the crime areas, the rate of crimes can be anticipated and prevented by more precise monitoring using the data mining algorithms. Generally, there is a functional category for implementing works in the field of crimes identification and their forecasting and

preventing in this framework that are distinguished in terms of the application of data mining techniques. The data mining algorithms of crime analysis are used for creating a model from databases related to the crimes in this way. To create a model, at first, an algorithm must analyze a set of data to find a specific pattern and processes of implementing them; then one can use the results of this analysis by defining the characteristics of the extraction models. Finally, we need to know how to extract the patterns of computer crimes' occurrence using the data mining techniques.

2 Literature

Sohrabi et al. stated that semi-supervised learning method is a proper performance that is used based on data mining algorithms such as support-vector machine for predicting the type of crime [6]. Javideh et al. expressed that it is assumed that the internet theft is lesser between the provinces; the information thefts were separately investigated in different provinces and then the outputs improved [7]. During a survey on computer crimes, Ghayom et al., found out that data mining has positive and negative aspects to explore the techniques of detecting computer crimes [8]. Caneppele et al. argue that internet development has changed the lifestyle and everyday activities of people and the violations have increased in the crime prevention strategies. The new process leads to consolidation of private security; and companies involve indirectly to collecting general data that can be used to study the computer crimes [9].

3 Materials and Methods

Systems that classify based on text mining have two input sets. The first one is a training set in which the data are in the specialized categories as a default and enter into the system with their classification structure; the system is trained based on them or some characteristics are provided by selecting and extracting them from the system.

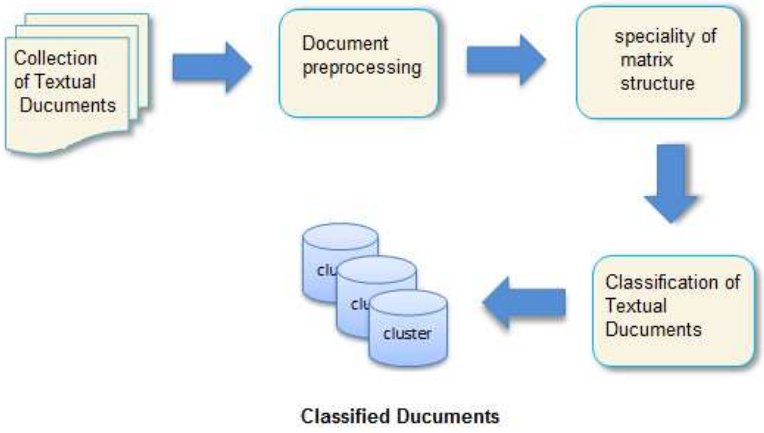

Figure 1. The process of text mining

The other set is the inputs that are entered into the system after the training stage to determine the category. Classification leads to the identification of characteristics that specify which group each case belongs to. This model can be used to understand existing words and to predict how any new model works. It is possible to classify specific words and keywords that are found through text mining to identify the extracted computer crimes by utilizing neural networks of Multilayer Perceptron (MLP) to obtain accurate conclusions from the patterns of these crimes, and deal with them. Support Vector Machine (SVM) can also provide accurate results. Finally, the extracted data or words from these methods are implemented with MATLAB software and the results are simulated and presented as a graph.

For finding the link between words and indexed documents and keywords (w) in the set of words, each word is associated with a subset, and each pair of words (W, W) is known as a rule of relationship. With this rule, for extracting the interested keywords from a set of words, first of all, the text will be read; the extra words deleted after processing and then the total number of document words will be counted one by one.

The number of the remained words is also calculated after the removal of extra words to obtain the number of repetitions of each word. Then, the frequency value of the expression is calculated as follows:

$$TF = \frac{T_w}{\sum w}. \tag{1}$$

In this equation, T_w is the number of times a word is used in the set of words and $\sum w$ is the total number of words in a document. According to the similarity rank of each word, they are put together in the same way as the following:

$$WS(w_i, w_j) = \frac{S(w_i, w_j) - \min S}{\max S - \min S}. \tag{2}$$

In this equation, $WS(w_i, w_j)$ indicates weighted similarity of w_j and w_i, $S(w_i, w_j)$ is the similarity number between the two words, $minS$ shows the smallest similarity number between all of the similar word pairs, $maxS$ is the largest similarity number between all the pair words. Therefore, this equation is computed for all of similar word pairs. At the next level, the reverse similarity rank will be computed to validation of the importance of each word. The Extra words are deleted and all computations are done again. The effect of a deleted word is defined by calculating the difference between the similarity of n and $n-1$ words; its similarity rank is as follows:

$$OS = OS(n-1) - OS(n). \tag{3}$$

In this equation, OS indicates the overall similarity of words, $n-1$ shows the previous word pair, and n is the current word pair.

4 Classification through neural network

Neural networks have a less classification error rate than decision trees, but they also require more time to learn. The main problems in this network are the classification rules that are learned by a few ways in a set of educational data because the learning or training of the

neural networks require a long time for obtaining high classification accuracy. The active amount of nodes in the hidden layer is calculated by transferring the sum of the weight of the input values in nonlinear active function. w_i^m allows the weights to connect the input node to the hidden node m. The input pattern is $x^i, i \in 1, 2, ..., k$ in which k is the number of word pairs in a set of words.

$$g_\theta(A) = diag(\theta_1\lambda_1, \ldots \theta_m\lambda_n), \theta \in R^n, \tag{4}$$

where $f(.)$ is an active function

$$f(x) : \delta(x) = \frac{e^x - e^{-x}}{e^x - e^{-x}}. \tag{5}$$

Active function of hidden nodes creates an active value range of hidden nodes. When the active value of all hidden nodes is computed, the ν_p^m output calculation of the network for the input pair x_i is defined as follows:

$$S_p^i = \sigma(\sum_{m=1}^{h} a^m \nu_p^m). \tag{6}$$

That ν_p^m is the connection weight between hidden node m and output node p, and h is the number of hidden nodes in the network.

In the neural network, the first input is multiplied by the relevant weight factor to communication line of that input. Then the same procedure is repeated for the second input and other ones. Finally, all the resulted values are added together in the target function:

$$\sum_{i=1}^{n} w_i x_i = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + ... + w_nx_n. \tag{7}$$

The sum of the above values must be compared with the threshold value of the interested neuron. In comparing the threshold, if the obtained sum is more than the threshold value, then the output of neurons would be 1, and if it is less than the threshold value, it would be zero.

Table 1. The values obtained for 5 neurons

Test Error	Learning Error	Type of hidden layer membership functions	Type of Input membership functions	Layers row number	
37.605	40.504	-	tan sig	2	1
8.790	10.604	Tan sig	Tan sig	3	2*
38.489	3.210	Tan sig	Tan sig	4	3
4.100	20.918	Tan sig	Tan sig	5	4

4.1 Simulation result for tan-sig membership function

The result of tan-sig membership function is shown in Table 2.

As it is shown in Table 1, the interested result (the lowest value for test and learning error) that was expected for the number of 5-neurons with constant membership function is obtained for a number of 3-layers. Figures of network view, performance and regression of network learning are shown in Figure 2, and charts are represented in Figures 3 and 4 respectively, and the corresponding row in Table 1 is shown by (*).

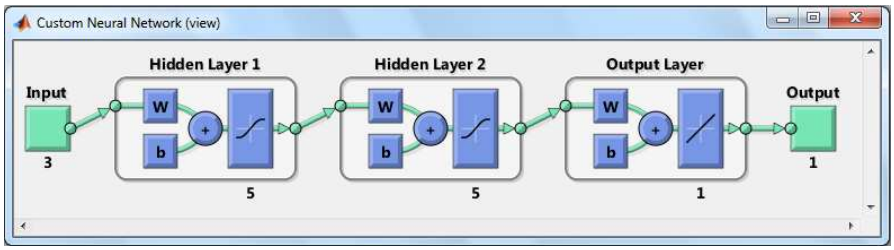

Figure 2. Network view corresponding to the second row of Table 1

As it can be seen in the chart in Figure 3, the blue line represents the network learning, the green line is related to validation and the red line shows the network test. The software uses validation performance for the early stop to avoid over- fitting of the neural network. The next step in network validation is to create a regression chart that shows

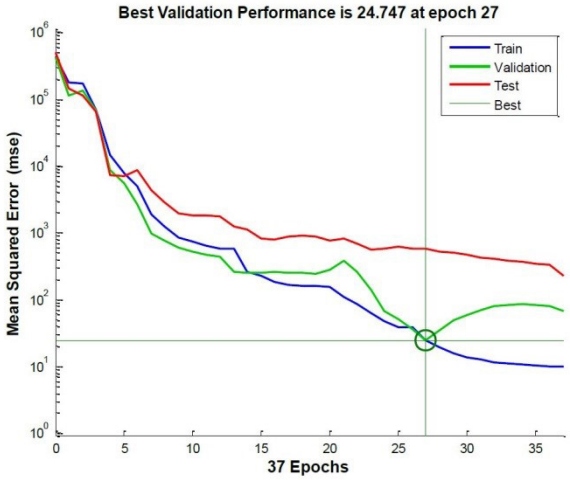

Figure 3. Network performance for the best output of Table 1 for the learning steps

the relationship between network outputs and goals. If the learning becomes complete, the network outputs and goals would be exactly equal, but the relation between these two would be rarely completed in learning phase. Three axes with charts in Figure 4 provide learning data, validation and test. The cutoff line in each axis shows the best result (output=goals). The continuous line represents the best linear fit of regression between the outputs and goals. The value of R shows the relation between the outputs and goals. If R = 1, it indicates the exact linear relationship between the outputs and goals. If R is close to zero, then there will be no linear relationship between the outputs and goals. In this figure learning data and validation are compatible. The results of the test also indicate that the values of R are greater than 0.9.

Table 2 shows the best result for the number of 7 neurons with constant membership function for the obtained number of 3 layers, and the performance chart for the network learning is shown in Figure 4.

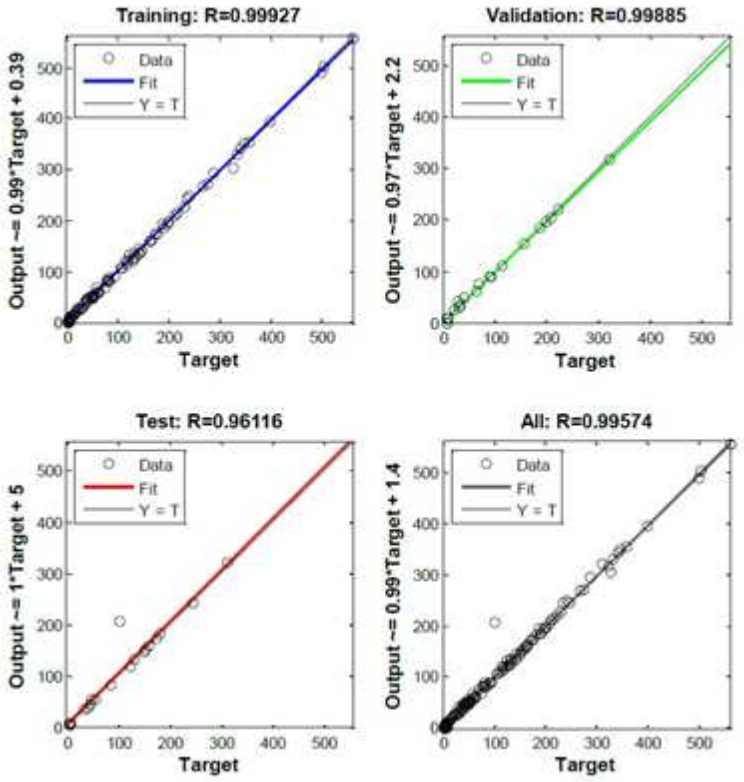

Figure 4. The network performance for the best output of Table 1 for the learning steps

Table 2. The values obtained for 7 neurons

Test Error	Learning Error	Type of hidden layer membership functions	Type of Input membership functions	Layers number	row
10.203	15.942	-	Tan sig	2	1
0.982	1.780	Tan sig	Tan sig	3	2*
137.215	61.536	Tan sig	Tan sig	4	3
50.160	119.480	Tan sig	Tan sig	5	4

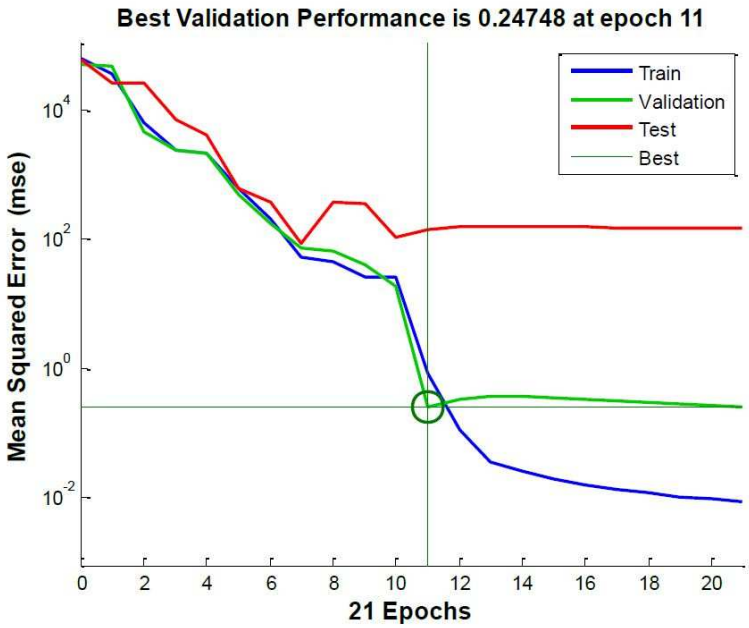

Figure 5. Network performance for the best output of Table 2 for the learning steps

Figure 3 represents the best validation performance that is equal to 0.248 in the 11th period, which is an appropriate value, and it is also clear that the training process has stopped in the 21st period. In Figure 5, R has a high value, and among the four axes, the R value of the test mode is lower than the others.

5 Results and comparing them with those of other researches

In the previous studies, various methods such as conducting a survey on learning with monitoring and learning techniques about criminal

identification and the use of machine learning methods, including stepwise regression, penalty method and random forests were used to create predictive models of violent crimes, and data mining was used to investigate and detect crimes. In the present study, in order to investigate and detect crimes, the intelligent methods of the neural network and modeling in MATLAB software have been used. The results of this study show that the gap between the theory and the implementation must be reduced, especially in the police area, in order to use artificial neural networks, where cybercrime is directly related to the increase of the crime in society. Cybercrimes lead to increase of the financial, psychological, cultural, social, political, and security damages.

This research can be very valuable and practical, since it has described the applied evidences using artificial intelligence in the field of cybercrime. The method used in this study, in addition to identifying crimes, saves lives, property, privacy, and security of human beings by identifying and detecting attacks and cybercrimes as well as reducing the possibility of committing such crimes by the offenders and consequently, reducing the damage of these violations. The used method in this research is most effective because of a low cost and has an optimal output.

6 Findings and Suggestions

1. It will be possible to implement this research as a software package for cybercrime detection systems in the future.

2. This research can be a background to design and produce online cybercrime recognition software in the future and can be used in the same way for other software that is used in cyber police centers employing artificial neural networks and special development programming techniques for detection of cybercrime in police stations.

3. The information and documentation of this research on cybercrime and analyzing it from various aspects is the basis of future

works of researchers in the law and psychology sciences on cyber criminology.

4. Using the results of this study, one can conclude a combination of which types of the cybercrime cause more cost and damage, and as a result, the legislator can punish more those who commit a combination of several specific violations.

5. Server data and systems of some organizations and corporations and cybercrime cases have been studied and documented. The results showed that:

   - Most cyber-attacks in organizations involve infiltrating office websites and intranet attacks
   - The largest cluster in terms of crime similarities is a cluster, which is used by criminals for their criminal acts using phishing techniques.

6. This system can be considered as one of the most effective and lowest cost ways to identify the cyber-criminal behavior, therefore, computer crime experts can run effectively this model on their systems.

7. The statistical and social studies of this research can be useful for scholars and researchers in this field. In another word, they can use these data in writing their future theses and articles on the psychology of the cybercrime.

References

[1] M. Delshad, B. Tudeh Za'im, and E. Rastegar, "Computer crime analysis using artificial intelligence methods and data mining for pre-trial crime," *Computer and Information Technology*, vol. 24, pp. 12, 2018. (in Farsi)

[2] M. Jantani, "Investigation and identification of the proposed algorithm in the case of electronic crimes on the website of the national court of justice computation," pp. 5–77, 2018. (in Farsi)

[3] D. K. Tayal, A. Jain, S. Arora, S. Agarwal, T. Gupta, and N. Tyagi, "Crime detection and criminal identification in india using data mining techniques," *AI & SOCIETY*, vol. 30, no. 1, pp. 117–127, apr 2014. (in Farsi)

[4] Shiju Sathyadevan, Devan M. S, and Surya Gangadharan S., "Crime analysis and prediction using data mining," in *2014 First International Conference on Networks & Soft Computing (ICNSC2014)*, (Guntur, India), IEEE, aug 2014, pp. 406–412. DOI: 10.1109/cnsc.2014.6906719. Available: https://doi.org/10.1109%2Fcnsc.2014.6906719. (in Farsi)

[5] S. M. A. M. Gadal and R. A. Mokhtar, "Anomaly detection approach using hybrid algorithm of data mining technique," in *2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)*, (Khartoum), IEEE, jan 2017, pp. 1–6. DOI: 10.1109/iccccee.2017.7867661. Available: https://doi.org/10.1109%2Ficccccee.2017.7867661. (in Farsi)

[6] B. Sohrabi, I. R. Vanani, and E. Abedin, "Human resources management and information systems trend analysis using text clustering," *International Journal of Human Capital and Information Technology Professionals*, vol. 9, no. 3, pp. 1–24, jul 2018. DOI: 10.4018/ijhcitp.2018070101. Available: https://doi.org/10.4018%2Fijhcitp.2018070101. (in Farsi)

[7] B. M. Mustafa Javideh, Einollah Khanjari, "Suggesting models for intelligent identification of burglars using local and behavioral information," in *International Congress on Innovation in Engineering and Technology Development*, 2017, pp. 6–7. (in Farsi)

[8] M. Ghayom, B. Pes, and S. Serusi,"Data mining for detecting bitcoin ponzi schemes," in *2018 Crypto Valley Conference on*

Blockchain Technology (CVCBT). IEEE, 2018, pp. 75–84. (in Farsi)

[9] S. Caneppele and M. F. Aebi, "Crime drop or police recording flop? on the relationship between the decrease of offline crime and the increase of online and hybrid crimes," *Policing: A Journal of Policy and Practice*, vol. 13, no. 1, pp. 66–79, sep 2017. DOI: 10.1093/police/pax055. Available: https://doi.org/10.1093%2Fpolice%2Fpax055. (in Farsi)

Abbas Karimi, Saber Abbasabadi,
Javad Akbari Torkestani, Frane Zarafshan

Received June 16, 2019
Revised February 15, 2020

Abbas Karimi
Assistant Professor, Department of Computer Engineering,
Arak Branch, Islamic Azad University, University, Arak, Iran.
E–mail: saber.abbasabadi1398@gmail.com

Saber Abbasabadei
PhD student, Department of Computer Engineering,
Arak Branch, Islamic Azad University, University, Arak, Iran.
E–mail: saber.abbasabadey1@gmail.com

Javad Akbari Torkestani
Associate Professor, Department of Computer Engineering,
Arak Branch, Islamic Azad University, University, Arak, Iran.
E–mail: j-akbari@iau-arak.ac.ir

Faradeh Zarafshan
Assistant Professor, Department of Computer Engineering,
Ashtian Branch, Islamic Azad University, Ashtian, Iran.
E–mail: fzarafshan@aiau.ac.ir

Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

Computation of general Randić polynomial and general Randić energy of some graphs

Harishchandra S. Ramane, Gouramma A. Gudodagi

Abstract

The general Randić matrix of a graph G, denoted by $GR(G)$ is an $n \times n$ matrix whose (i, j)-th entry is $(d_i d_j)^{\alpha}$, $\alpha \in \mathbb{R}$ if the vertices v_i and v_j are adjacent and 0 otherwise, where d_i is the degree of a vertex v_i and n is the order of G. The general Randić energy $E_{GR}(G)$ of G is the sum of the absolute values of the eigenvalues of $GR(G)$. In this paper, we compute the general Randić polynomial and the general Randić energy of path, cycle, complete graph, complete bipartite graph, friendship graph and Dutch windmill graph.

Keywords: General Randić eigenvalues, general Randić energy, Randić index, degree of a vertex.

MSC 2010: 05C50, 05C07.

1 Introduction

Topological indices are the numerical quantities of a graph which are invariant under graph isomorphism. The interest in topological indices is mainly related to their use in quantitative structure-property relationship (QSPR) and quantitative structure-activity relationship (QSAR) [16].

Throughout the paper we consider only simple finite graphs, without directed, multiple or weighted edges and without loops. Let G be a simple graph with n vertices and m edges. Let the vertex set of G be $V(G) = \{v_1, v_2, \ldots, v_n\}$. If two vertices v_i and v_j of G are adjacent, then we write $v_i \sim v_j$. For $v_i \in V(G)$, the degree of the vertex v_i,

denoted by d_i, is the number of vertices adjacent to v_i.

The general Randić (GR) matrix of a graph G is a square matrix $GR(G) = (d_{ij})_{n\times n}$ in which

$$d_{ij} = \begin{cases} (d_i d_j)^{\alpha} & \text{if} \quad v_i \sim v_j \\ 0 & \text{otherwise,} \end{cases}$$

where $\alpha \in \mathbb{R}$.

If $\alpha = -1/2$, then the above definition reduces to the Randić matrix, which was invented by Milan Randić [23] in 1975 as a molecular structure descriptor. In 1998, Bollobás and Erdös [2] generalized this index as $R_\alpha = R_\alpha(G) = \sum_{v_i \sim v_j} (d_i d_j)^{\alpha}$, called general Randić index. The Randić index concept suggests that it is a purposeful to associate to the graph G a symmetric square matrix $R(G)$. The Randić matrix [3], [4], [9], [13] is denoted by $R(G) = (r_{ij})_{n\times n}$, where

$$r_{ij} = \begin{cases} \frac{1}{\sqrt{d_i d_j}} & \text{if} \quad v_i \sim v_j \\ 0 & \text{otherwise.} \end{cases}$$

Denote the eigenvalues of the GR matrix of G by $\lambda_1, \lambda_2, \ldots, \lambda_n$ and order them in nonincreasing order. Similar to the characteristic polynomial of a matrix, we consider the general Randić (GR) polynomial of G as $det(\lambda I - GR(G)) = \phi_{GR}(G, \lambda)$, where I is the identity matrix of order n. The general Randić energy is defined as $E_{GR}(G) = \sum_{i=1}^{n} |\lambda_i|$.

The $E_{GR}(G)$ is defined in analogous to the ordinary graph energy defined as the sum of the absolute values of the eigenvalues of the adjacency matrix [15]. The ordinary graph energy is closely related to the total π-electron energy of a non-saturated hydrocarbons as calculated with the Huckel molecular orbital (HMO) method in chemistry [11]. Detail information about the graph energy can be found in [12], [14], [20]. There are many other kinds of graph energies, such as incidence energy [5], [6], distance energy [18], Laplacian energy [17], matching energy [7], [19], [21], Randić energy [23] and skew energy [22].

In this paper we obtain the GR-polynomial and GR-energy of some specific graphs. These results generalise the results obtained in paper [1].

Remark 1. *Given graph G, its general Randić energy $E_{GR}(G)$, is directly obtained from its general Randić polynomial $\phi_{GR}(G)$ by :*

(a) finding the solutions, λ_i's, (which are eigenvalues) for the equation

$$\phi_{GR}(G) = 0,$$

(b) and computing $E_{GR}(G) = \sum_{i=1}^{n} |\lambda_i|$.

2 GR-polynomial and GR-energy:

Let P_n, C_n, K_n, $K_{p,q}$, and $S_n = K_{1,n-1}$ denote the path, the cycle, the complete graph, complete bipartite graph and star graph respectively on n vertices.

Theorem 2.1 For $n \geq 5$ and $\alpha \in \mathbb{R}$, the GR polynomial of the path P_n is

$\phi_{GR}(P_n, \lambda) = \lambda^2\Lambda_{n-2} - 2(4)^\alpha\lambda\Lambda_{n-3} + (16)^\alpha\Lambda_{n-4}$, where for every $k \geq 3$, $\Lambda_k = \lambda\Lambda_{k-1} - (16)^\alpha\Lambda_{k-2}$ with $\Lambda_1 = \lambda$ and $\Lambda_2 = \lambda^2 - (16)^\alpha$.

Proof. For every $k \geq 3$, consider

$$B_k = \begin{bmatrix} \lambda & -4^\alpha & 0 & 0 & \dots & 0 & 0 & 0 \\ -4^\alpha & \lambda & -4^\alpha & 0 & \dots & 0 & 0 & 0 \\ 0 & -4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\ 0 & 0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\ 0 & 0 & 0 & 0 & \dots & -4^\alpha & \lambda & -4^\alpha \\ 0 & 0 & 0 & 0 & \dots & 0 & -4^\alpha & \lambda \end{bmatrix}_{k\times k},$$

and let $\Lambda_k = det(B_k)$. It is easy to see that $\Lambda_k = \lambda\Lambda_{k-1} - (16)^\alpha \Lambda_{k-2}$.

Therefore

$$\begin{aligned}
\phi_{GR}(P_n, \lambda) &= det(\lambda I - GR(P_n)) \\
&= \begin{vmatrix}
\lambda & -2^\alpha & 0 & 0 & \dots & 0 & 0 & 0 \\
-2^\alpha & \lambda & -4^\alpha & 0 & \dots & 0 & 0 & 0 \\
0 & -4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\
0 & 0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\
0 & 0 & 0 & 0 & \dots & -4^\alpha & \lambda & -2^\alpha \\
0 & 0 & 0 & 0 & \dots & 0 & -2^\alpha & \lambda
\end{vmatrix}_{n\times n}.
\end{aligned}$$

$$\begin{aligned}
\phi_{GR}(P_n, \lambda) &= \lambda \begin{vmatrix}
\lambda & -4^\alpha & 0 & \dots & 0 & 0 & 0 \\
-4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\
0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\
0 & 0 & 0 & \dots & -4^\alpha & \lambda & -2^\alpha \\
0 & 0 & 0 & \dots & 0 & -2^\alpha & \lambda
\end{vmatrix} \\
&\quad + 2^\alpha \begin{vmatrix}
-2^\alpha & 0 & 0 & \dots & 0 & 0 & 0 \\
-4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\
0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\
0 & 0 & 0 & \dots & -4^\alpha & \lambda & -2^\alpha \\
0 & 0 & 0 & \dots & 0 & -2^\alpha & \lambda
\end{vmatrix}.
\end{aligned}$$

Further,

$$\begin{aligned}
\phi_{GR}(P_n,\lambda) &= \lambda \left(\lambda\Lambda_{(n-2)} + 2^{\alpha} \begin{vmatrix} \lambda & -4^{\alpha} & 0 & \dots & 0 & 0 \\ -4^{\alpha} & \lambda & -4^{\alpha} & \dots & 0 & 0 \\ 0 & -4^{\alpha} & \lambda & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \lambda & 0 \\ 0 & 0 & 0 & \dots & 0 & -2^{\alpha} \end{vmatrix} \right) \\
&\quad -(4)^{\alpha} \left(\lambda \begin{vmatrix} \lambda & -4^{\alpha} & \dots & 0 & 0 \\ (-4)^{\alpha} & \lambda & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \lambda & -4^{\alpha} \\ 0 & 0 & \dots & -4^{\alpha} & \lambda \end{vmatrix} \right. \\
&\quad \left. + \; 2^{\alpha} \begin{vmatrix} \lambda & -4^{\alpha} & \dots & 0 & 0 \\ -4^{\alpha} & \lambda & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \lambda & 0 \\ 0 & 0 & \dots & -4^{\alpha} & -2^{\alpha} \end{vmatrix} \right).
\end{aligned}$$

Hence,

$$\begin{aligned}
\phi_{GR}(P_n,\lambda) &= \lambda^2\Lambda_{n-2} - 4^{\alpha}\lambda\Lambda_{n-3} - 4^{\alpha}\lambda\Lambda_{n-3} + 16^{\alpha}\Lambda_{n-4}. \\
&= \lambda^2\Lambda_{n-2} - 2(4)^{\alpha}\lambda\Lambda_{n-3} + 16^{\alpha}\Lambda_{n-4}.
\end{aligned}$$

□

Theorem 2.2 For $n \geq 3$ and $\alpha \in \mathbb{R}$, the GR polynomial of the cycle C_n is

$$\phi_{GR}(C_n,\lambda) = \lambda\Lambda_{n-1} - 2(16)^{\alpha}\Lambda_{n-2} - (4)^{\alpha n}2,$$

where for every $k \geq 3$, $\Lambda_k = \lambda\Lambda_{k-1} - (16)^{\alpha}\Lambda_{k-2}$ with $\Lambda_1 = \lambda$ and $\Lambda_2 = \lambda^2 - (16)^{\alpha}$.

Proof. Similar to the proof of Theorem 2.1, for every $k \geq 3$, we consider

$$B_k = \begin{bmatrix} \lambda & -4^\alpha & 0 & 0 & \dots & 0 & 0 & 0 \\ -4^\alpha & \lambda & -4^\alpha & 0 & \dots & 0 & 0 & 0 \\ 0 & -4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\ 0 & 0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\ 0 & 0 & 0 & 0 & \dots & -4^\alpha & \lambda & -4^\alpha \\ 0 & 0 & 0 & 0 & \dots & 0 & -4^\alpha & \lambda \end{bmatrix}_{k\times k},$$

and let $\Lambda_k = det(B_k)$. It is easy to see that $\Lambda_k = \lambda\Lambda_{k-1} - (16)^\alpha \Lambda_{k-2}$.

Therefore

$$\phi_{GR}(C_n, \lambda) = det(\lambda I - GR(C_n))$$
$$= \begin{vmatrix} \lambda & -4^\alpha & 0 & 0 & \dots & 0 & 0 & -4^\alpha \\ -4^\alpha & \lambda & -4^\alpha & 0 & \dots & 0 & 0 & 0 \\ 0 & -4^\alpha & \lambda & -4^\alpha & \dots & 0 & 0 & 0 \\ 0 & 0 & -4^\alpha & \lambda & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \lambda & -4^\alpha & 0 \\ 0 & 0 & 0 & 0 & \dots & -4^\alpha & \lambda & -4^\alpha \\ -4^\alpha & 0 & 0 & 0 & \dots & 0 & -4^\alpha & \lambda \end{vmatrix}_{n\times n}.$$

$$\phi_{GR}(C_n,\lambda) =$$

$$= \lambda\Lambda_{(n-1)} + 4^{\alpha} \begin{vmatrix} -4^{\alpha} & 0 & 0 & \dots & 0 & 0 & -4^{\alpha} \\ -4^{\alpha} & \lambda & -4^{\alpha} & \dots & 0 & 0 & 0 \\ 0 & -4^{\alpha} & \lambda & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \lambda & -4^{\alpha} & 0 \\ 0 & 0 & 0 & \dots & -4^{\alpha} & \lambda & -4^{\alpha} \\ 0 & 0 & 0 & \dots & 0 & -4^{\alpha} & \lambda \end{vmatrix}_{(n-1)\times(n-1)}$$

$$+(-1)^{(n+1)}[-(4)^{\alpha}] \begin{vmatrix} -4^{\alpha} & 0 & 0 & \dots & 0 & 0 & -4^{\alpha} \\ \lambda & -4^{\alpha} & 0 & \dots & 0 & 0 & 0 \\ -4^{\alpha} & \lambda & -4^{\alpha} & \dots & 0 & 0 & 0 \\ 0 & -4^{\alpha} & \lambda & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \lambda & -4^{\alpha} & 0 \\ 0 & 0 & 0 & \dots & -4^{\alpha} & \lambda & -4^{\alpha} \end{vmatrix}_{(n-1)\times(n-1).}$$

$$\phi_{GR}(C_n,\lambda) = \lambda\Lambda_{(n-1)} - 16^{\alpha}\Lambda_{(n-2)} +$$

$$+(-1)^{n}(-16^{\alpha}) \begin{vmatrix} -4^{\alpha} & \lambda & -4^{\alpha} & \dots & 0 & 0 \\ 0 & -4^{\alpha} & \lambda & \dots & 0 & 0 \\ 0 & 0 & -4^{\alpha} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \\ 0 & 0 & 0 & \dots & -4^{\alpha} & \lambda \\ 0 & 0 & 0 & \dots & 0 & -4^{\alpha} \end{vmatrix}_{(n-2)\times(n-2)}$$

$$+(-1)^{(n+1)}[-(4)^{\alpha}] \left((-4)^{\alpha} \begin{vmatrix} -4^{\alpha} & 0 & \dots & 0 \\ \lambda & -4^{\alpha} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & -4^{\alpha} \end{vmatrix}_{(n-2)\times(n-2)} \right.$$

$$\left. +(-1)^{n}[-4^{\alpha}]\Lambda_{n-2} \right).$$

Therefore,

$$\begin{aligned}\phi_{GR}(C_n,\lambda) &= \lambda\Lambda_{n-1} - 16^{\alpha}\Lambda_{n-2} + (-1)^n(-16^{\alpha})[-(4)^{\alpha}]^{n-2} \\ &+ (-1)^{n+1}[-(4)^{\alpha}]^n + (-1)^{2n+1}(16)^{\alpha}\Lambda_{n-2} \\ &= \lambda\Lambda_{n-1} - 2(16^{\alpha})\Lambda_{n-2} - (2)4^{(\alpha n)}.\end{aligned}$$

□

Lemma 2.3 [8] If M is a nonsingular square matrix, then

$$det\begin{pmatrix} M & N \\ P & Q \end{pmatrix} = det(M)\, det(Q - PM^{-1}N).$$

Theorem 2.4 For $n \geq 2$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of the complete graph K_n is

$$\phi_{GR}(K_n,\lambda) = (\lambda - (n-1)^{2\alpha+1})(\lambda + (n-1)^{2\alpha})^{(n-1)},$$

(ii) the GRE of K_n is

$$E_{GR}(K_n) = 2(n-1)^{2\alpha+1}.$$

Proof. It is easy to see that the GR matrix of K_n is $(n-1)^{2\alpha}(J_n - I)$, where J_n is a matrix whose all entries are equal to one and I is an identity matrix. Therefore

$$\begin{aligned}\phi_{GR}(K_n,\lambda) &= \left|\lambda I - (n-1)^{2\alpha}J_n + (n-1)^{2\alpha}I\right| \\ &= \left|(\lambda + (n-1)^{2\alpha})I - (n-1)^{2\alpha}J_n\right|.\end{aligned}$$

Since the eigenvalues of J_n are n (once) and 0 ($n-1$ times), the eigenvalues of $(n-1)^{2\alpha}J_n$ are $n(n-1)^{2\alpha}$ (once) and 0 ($n-1$ times).

Hence

$$\phi_{GR}(K_n,\lambda) = (\lambda - (n-1)^{2\alpha+1})(\lambda + (n-1)^{2\alpha})^{(n-1)}.$$

(ii) It follows from Remark 1. □

Theorem 2.5 For any positive integers $p, q \geq 1$ and $\alpha \in \mathbb{R}$,

(i) The GR polynomial of complete bipartite graph $K_{p,q}$ is

$$\phi_{GR}(K_{p,q}, \lambda) = \lambda^{p+q-2}(\lambda^2 - (pq)^{2\alpha+1}),$$

(ii) $E_{GR}(K_{p,q}) = 2\sqrt{(pq)^{2\alpha+1}}$.

Proof. It is easy to see that the GR matrix of $K_{p,q}$ is

$$GR(K_{p,q}) = (pq)^{\alpha} \begin{pmatrix} O_{p\times p} & J_{p\times q} \\ J_{q\times p} & O_{q\times q} \end{pmatrix}.$$

Therefore,

$$\phi_{GR}(K_{p,q}, \lambda) = \begin{vmatrix} \lambda I_p & -(pq)^{\alpha} J_{p\times q} \\ -(pq)^{\alpha} J_{q\times p} & \lambda I_q \end{vmatrix}.$$

Using Lemma 2.3 we have

$$\begin{aligned} \phi_{GR}(K_{p,q}, \lambda) &= |\lambda I_p| \left| \lambda I_q - (-pq)^{\alpha} J_{q\times p} \frac{I_p}{\lambda} (-pq)^{\alpha} J_{p\times q} \right| \\ &= \lambda^{p-q} \left| \lambda^2 I_q - p\,(pq)^{2\alpha} J_q \right| \quad \text{since } J_{q\times p} J_{p\times q} = p J_q. \end{aligned}$$

Since the eigenvalues of J_n are n (once) and 0 ($n-1$ times), the eigenvalues of $p(pq)^{2\alpha} J_q$ are $(pq)^{2\alpha+1}$ (once) and 0 ($q-1$ times). Therefore

$\phi_{GR}(K_{p,q}, \lambda) = \lambda^{p+q-2}(\lambda^2 - (pq)^{2\alpha+1})$.

(ii) It follows from Remark 1.

□

Corollary 2.6 For $n \geq 2$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of the star $S_n = K_{1,n-1}$ is

$$\phi_{GR}(S_n, \lambda) = \lambda^{(n-2)}(\lambda^2 - (n-1)^{2\alpha+1}),$$

(ii) the GRE of S_n is

$$E_{GR}(S_n) = 2\sqrt{(n-1)^{2\alpha+1}}.$$

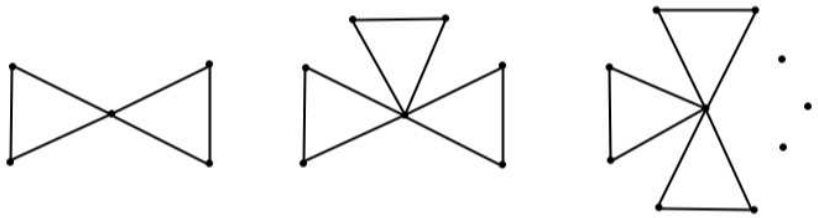

Figure 1. Friendship graphs F_2, F_3 and F_n respectively

Let n be any positive integer and F_n be a friendship graph with $2n+1$ vertices and $3n$ edges. In other words, the friendship graph F_n is a graph that can be constructed by coalescence n copies of the cycle C_3 of length 3 with common vertex. The Friendship theorem of Erdös et al. [10], states that graphs with the property that every two vertices have exactly one neighbour in common are exactly the friendship graphs. The Fig. 1 shows some examples of friendship graphs. Here we compute the GRE of friendship graphs.

Theorem 2.7 For $n \geq 2$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of friendship graph F_n is

$$\begin{aligned}\phi_{GR}(F_n, \lambda) &= (\lambda^2 - 4^{2\alpha})^{n-1}(\lambda + 4^{\alpha})\left(\lambda - \left[2^{2\alpha-1} + 2^{2\alpha-1}\sqrt{1+8n^{2\alpha+1}}\right]\right) \\ &\quad \left(\lambda - \left[2^{2\alpha-1} - 2^{2\alpha-1}\sqrt{1+8n^{2\alpha+1}}\right]\right),\end{aligned}$$

(ii) the GR energy of friendship graph F_n is

$$E_{GR}(F_n) = \begin{cases} 4^{\alpha}(2n-1)+2^{2\alpha} & \text{if } n^{2\alpha+1} \leq 0 \\ 4^{\alpha}(2n-1)+2^{2\alpha}\sqrt{1+8n^{2\alpha+1}} & \text{if } n^{2\alpha+1} > 0. \end{cases}$$

Proof. The GR matrix of F_n is

$$GR(F_n) = \begin{bmatrix} 0 & (4n)^{\alpha} & (4n)^{\alpha} & \dots & (4n)^{\alpha} & (4n)^{\alpha} \\ (4n)^{\alpha} & 0 & 4^{\alpha} & \dots & 0 & 0 \\ (4n)^{\alpha} & 4^{\alpha} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ (4n)^{\alpha} & 0 & 0 & \dots & 0 & 4^{\alpha} \\ (4n)^{\alpha} & 0 & 0 & \dots & 4^{\alpha} & 0 \end{bmatrix}_{(2n+1)\times(2n+1)}.$$

Now, for computing $|\lambda I - GR(F_n)|$, we consider its first row. The cofactor of the first array in this row is

$$\begin{vmatrix} \lambda & -4^{\alpha} & \dots & 0 & 0 \\ -4^{\alpha} & \lambda & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \lambda & -4^{\alpha} \\ 0 & 0 & \dots & -4^{\alpha} & \lambda \end{vmatrix}_{(2n)\times(2n)}$$

and the cofactor of another arrays in the first row are similar to

$$\begin{vmatrix} -(4n)^{\alpha} & -4^{\alpha} & \dots & 0 & 0 \\ -(4n)^{\alpha} & \lambda & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ -(4n)^{\alpha} & 0 & \dots & \lambda & -4^{\alpha} \\ -(4n)^{\alpha} & 0 & \dots & -4^{\alpha} & \lambda \end{vmatrix}_{(2n)\times(2n)}.$$

Now solving the above two determinants, we get

$$\begin{aligned}\phi_{GR}(F_n,\lambda) &= \lambda(\lambda^2-4^{2\alpha})^n+(4n)^\alpha 2n\left[(-(4n)^\alpha\lambda-(4n)^\alpha 4^\alpha)(\lambda^2-4^{2\alpha})^{(n-1)}\right]\\ &= (\lambda^2-4^{2\alpha})^{n-1}(\lambda+4^\alpha)\left(\lambda-\left[2^{2\alpha-1}+2^{2\alpha-1}\sqrt{1+8n^{2\alpha+1}}\right]\right)\\ &\quad \left(\lambda-\left[2^{2\alpha-1}-2^{2\alpha-1}\sqrt{1+8n^{2\alpha+1}}\right]\right).\end{aligned}$$

(ii) It follows from Remark 1.

□

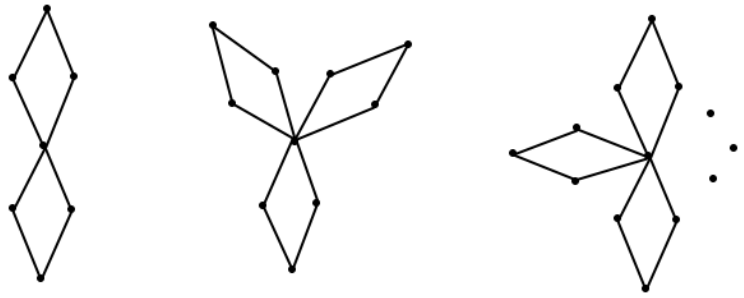

Figure 2. Dutch Windmill graph D_4^2 , D_4^3 and D_4^n respectively

Let n be any positive integer and D_4^n be Dutch Windmill graph with $3n+1$ vertices and $4n$ edges. In other words, the graph D_4^n is a graph that can be constructed by coalescing n copies of the cycle C_4 of length 4 with a common vertex. Figure 2 shows some examples of Dutch Windmill graphs. Here we compute the GRE of Dutch Windmill graphs.

Theorem 2.8 For $n \geq 2$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of Dutch Windmill graph D_4^n is

$$\phi_{GR}(D_4^n,\lambda) = \lambda^{n+1}(\lambda^2-(2)4^{2\alpha})^{n-1}[\lambda^2-(2)4^{2\alpha}-2n(4n)^{(2\alpha)}],$$

(ii) $E_{GR}(D_4^n) = 2\sqrt{2}\ 4^\alpha(n-1) + 2^{2\alpha+1}\sqrt{2(1+n^{2\alpha+1})}$.

Proof. The GR matrix of D_4^n is

$$\begin{bmatrix} 0 & (4n)^\alpha & (4n)^\alpha & 0 & \dots & (4n)^\alpha & (4n)^\alpha & 0 \\ (4n)^\alpha & 0 & 0 & 4^\alpha & \dots & 0 & 0 & 0 \\ (4n)^\alpha & 0 & 0 & 4^\alpha & \dots & 0 & 0 & 0 \\ 0 & 4^\alpha & 4^\alpha & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ (4n)^\alpha & 0 & 0 & 0 & \dots & 0 & 0 & 4^\alpha \\ (4n)^\alpha & 0 & 0 & 0 & \dots & 0 & 0 & 4^\alpha \\ 0 & 0 & 0 & 0 & \dots & 4^\alpha & 4^\alpha & 0 \end{bmatrix}_{(3n+1)\times(3n+1)}.$$

Let $A = \begin{pmatrix} \lambda & 0 & -4^\alpha \\ 0 & \lambda & -4^\alpha \\ -4^\alpha & -4^\alpha & \lambda \end{pmatrix}$, $B = \begin{pmatrix} -(4n)^\alpha & 0 & 0 \\ -(4n)^\alpha & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ and $C = \begin{pmatrix} -(4n)^\alpha & 0 & -4^\alpha \\ -(4n)^\alpha & \lambda & -4^\alpha \\ 0 & -4^\alpha & \lambda \end{pmatrix}$.

Then

$$\begin{aligned} \phi_{GR}(D_4^n, \lambda) &= det(\lambda I - GR(D_4^n)) \\ &= \lambda(det(A))^n + 2n(4n)^\alpha det \begin{pmatrix} C & O & O & \dots & O \\ B & A & O & \dots & O \\ B & O & A & \dots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ B & O & O & \dots & A \end{pmatrix}_{(3n)\times(3n)}. \end{aligned}$$

Now, by the straightforward computation we have the result.

(ii) It follows from Remark 1.

□

Let n be any positive integer and D_5^n be Dutch Windmill graph with $4n+1$ vertices and $5n$ edges. In other words, the graph D_5^n is a graph that can be constructed by coalescing n copies of the cycle C_5

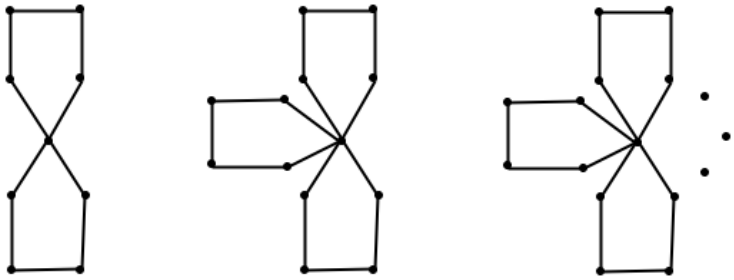

Figure 3. Dutch Windmill graph D_5^2, D_5^3 and D_5^n respectively

of length 5 with a common vertex. Figure 3 shows some examples of Dutch Windmill graphs.

Theorem 2.9 For $n \geq 2$ and $\alpha \in \mathbb{R}$, the GR polynomial of Dutch Windmill graph D_5^n is

$$\begin{aligned}\phi_{GR}(D_5^n, \lambda) &= (\lambda^4 - 3\lambda^2 4^{2\alpha} + 4^{4\alpha})^{(n-1)} (\lambda^5 - 3\lambda^3 4^{2\alpha} + \lambda 4^{4\alpha} - \\ &\quad -2n\lambda^3 (4n)^{(2\alpha)} + 4n\lambda (4n)^{(2\alpha)} 4^{2\alpha} - 2n(4n)^{(2\alpha)} 4^{3\alpha}).\end{aligned}$$

Proof. The GR matrix of D_5^n is

$$\begin{bmatrix} 0 & (4n)^\alpha & (4n)^\alpha & 0 & 0 & \dots & (4n)^\alpha & (4n)^\alpha & 0 & 0 \\ (4n)^\alpha & 0 & 0 & 0 & 4^\alpha & \dots & 0 & 0 & 0 & 0 \\ (4n)^\alpha & 0 & 0 & 4^\alpha & 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & 0 & 4^\alpha & 0 & 4^\alpha & \dots & 0 & 0 & 0 & 0 \\ 0 & 4^\alpha & 0 & 4^\alpha & 0 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ (4n)^\alpha & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 4^\alpha \\ (4n)^\alpha & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 4^\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 4^\alpha & 0 & 4^\alpha \\ 0 & 0 & 0 & 0 & 0 & \dots & 4^\alpha & 0 & 4^\alpha & 0 \end{bmatrix}_{(3n+1)\times(3n+1).}$$

Let

$$A = \begin{pmatrix} \lambda & 0 & 0 & -4^\alpha \\ 0 & \lambda & -4^\alpha & 0 \\ 0 & -4^\alpha & \lambda & -4^\alpha \\ -4^\alpha & 0 & -4^\alpha & \lambda \end{pmatrix}, \; B = \begin{pmatrix} -(4n)^\alpha & 0 & 0 & 0 \\ -(4n)^\alpha & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\text{and } C = \begin{pmatrix} -(4n)^\alpha & 0 & 0 & -4^\alpha \\ -(4n)^\alpha & \lambda & -4^\alpha & 0 \\ 0 & -4^\alpha & \lambda & -4^\alpha \\ 0 & 0 & -4^\alpha & \lambda \end{pmatrix}.$$

Then

$$\begin{aligned} \phi_{GR}(D_5^n, \lambda) &= det(\lambda I - GR(D_5^n)) \\ &= \lambda(det(A))^n + 2n(4n)^\alpha det \begin{pmatrix} C & O & O & O & \dots & O \\ B & A & O & O & \dots & O \\ B & O & A & O & \dots & O \\ B & O & O & A & \dots & O \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ B & O & O & O & \dots & A \end{pmatrix}_{(4n)\times(4n)}. \end{aligned}$$

Now, by the straightforward computation we have the result.

□

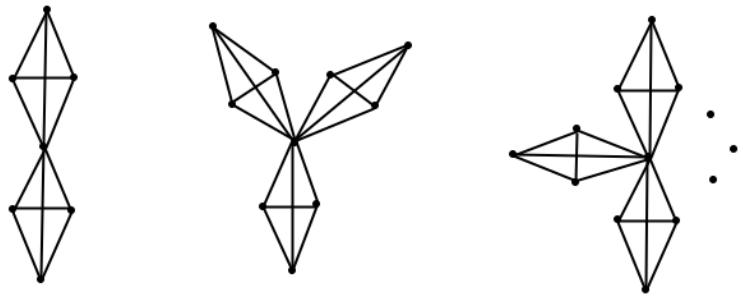

Figure 4. K_4-Windmill graph K_4^2, K_4^3 and K_4^n respectively

Let n be any positive integer and K_4^n be K_4-Windmill graph with $4n+1$ vertices and $6n$ edges. In other words, the graph K_4^n is a graph that can be constructed by coalescing n copies of the complete graph K_4 with a common vertex. Figure 4 shows some examples of K_4-Windmill graphs.

Theorem 2.10 For $n \geq 2$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of K_4^n-Windmill graph is

$$\begin{aligned}\phi_{GR}(K_4^n, \lambda) &= [(\lambda + 9^\alpha)^2(\lambda - 2(9)^\alpha)]^{(n-1)}(\lambda + 9^\alpha) \\ &\quad \left(\lambda - \left[9^\alpha + 9^\alpha\sqrt{1+3n^{2\alpha+1}}\right]\right) \\ &\quad \left(\lambda - \left[9^\alpha - 9^\alpha\sqrt{1+3n^{2\alpha+1}}\right]\right).\end{aligned}$$

(ii) the GR energy of K_4-Windmill graph is

$$E_{GR}(K_4^n) = \begin{cases} 4n(9)^\alpha & \text{if } n^{2\alpha+1} \leq 0 \\ 2(9)^\alpha[(2n-1) + \sqrt{1+3n^{2\alpha+1}} & \text{if } n^{2\alpha+1} > 0. \end{cases}$$

Proof. The GR matrix of K_4^n is

$$\begin{bmatrix} 0 & (9n)^\alpha & (9n)^\alpha & (9n)^\alpha & \dots & (9n)^\alpha & (9n)^\alpha & (9n)^\alpha \\ (9n)^\alpha & 0 & 9^\alpha & 9^\alpha & \dots & 0 & 0 & 0 \\ (9n)^\alpha & 9^\alpha & 0 & 9^\alpha & \dots & 0 & 0 & 0 \\ (9n)^\alpha & 9^\alpha & 9^\alpha & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ (9n)^\alpha & 0 & 0 & 0 & \dots & 0 & 9^\alpha & 9^\alpha \\ (9n)^\alpha & 0 & 0 & 0 & \dots & 9^\alpha & 0 & 9^\alpha \\ (9n)^\alpha & 0 & 0 & 0 & \dots & 9^\alpha & 9^\alpha & 0 \end{bmatrix}_{(3n+1)\times(3n+1)}.$$

Let

$$A = \begin{pmatrix} \lambda & -9^\alpha & -9^\alpha \\ -9^\alpha & \lambda & -9^\alpha \\ -9^\alpha & -9^\alpha & \lambda \end{pmatrix}, \quad B = \begin{pmatrix} -(9n)^\alpha & 0 & 0 \\ -(9n)^\alpha & 0 & 0 \\ -(9n)^\alpha & 0 & 0 \end{pmatrix}$$

and $C = \begin{pmatrix} -(9n)^\alpha & -9^\alpha & -9^\alpha \\ -(9n)^\alpha & \lambda & -9^\alpha \\ -(9n)^\alpha & -9^\alpha & \lambda \end{pmatrix}$.

Then

$$\begin{aligned} \phi_{GR}(K_4^n, \lambda) &= det(\lambda I - GR(K_4^n)) \\ &= \lambda(det(A))^n + 3n(9n)^\alpha det \begin{pmatrix} C & O & O & \dots & O \\ B & A & O & \dots & O \\ B & O & A & \dots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ B & O & O & \dots & A \end{pmatrix}_{(3n)\times(3n)}. \end{aligned}$$

Now, by the straightforward computation we have the result.

(ii) It follows from Remark 1.

□

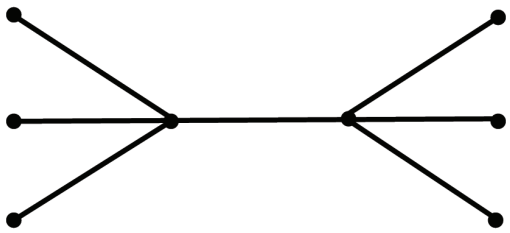

Figure 5. Double star $S(3,3)$

For $p, q \geq 1$ the double star $S(p,q)$ is the graph on the points $\{v_0, v_1, \dots, v_p, w_0, w_1, \dots, w_q\}$ with lines $\{(v_0, w_0), (v_0, v_i), (w_0, w_j) : 1 \leq i \leq p, 1 \leq j \leq q\}$ (see Fig. 5).

Theorem 2.11 For $p, q \geq 1$ and $\alpha \in \mathbb{R}$,

(i) the GR polynomial of double star graph $S(p,q)$ is

$$\begin{aligned}\phi_{GR}(S(p,q),\lambda) &= \lambda^{p+q-4}\left[\lambda^4-\lambda^2\left((p-1)p^{2\alpha}+(q-1)q^{2\alpha}\right.\right.\\ &\quad \left.\left.+(pq)^{2\alpha}\right)+(p-1)(q-1)(pq)^{2\alpha}\right].\end{aligned}$$

(ii)

$$\begin{aligned}E_{GR}(S(p,q)) &= \sqrt{2}\sqrt{X+\sqrt{X^2-4(p-1)(q-1)(pq)^{2\alpha}}}\\ &\quad +\sqrt{2}\sqrt{X-\sqrt{X^2-4(p-1)(q-1)(pq)^{2\alpha}}},\end{aligned}$$

where $X=(p-1)\,p^{2\alpha}+(q-1)\,q^{2\alpha}+(pq)^{2\alpha}$.

Proof. The GR polynomial of $S(p,q)$ is

$$\phi_{GR}(S(p,q),\lambda)=det(\lambda I-GR(S(p,q)))=$$
$$\begin{vmatrix} \lambda & -(pq)^{\alpha} & -q^{\alpha} & \dots & -q^{\alpha} & 0 & \dots & 0\\ -(pq)^{\alpha} & \lambda & 0 & \dots & 0 & -p^{\alpha} & \dots & -p^{\alpha}\\ -q^{\alpha} & 0 & \lambda & \dots & 0 & 0 & \dots & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots\\ -q^{\alpha} & 0 & 0 & \dots & \lambda & 0 & \dots & 0\\ 0 & -p^{\alpha} & 0 & \dots & 0 & \lambda & \dots & 0\\ 0 & -p^{\alpha} & 0 & \dots & 0 & 0 & \dots & 0\\ 0 & -p^{\alpha} & 0 & \dots & 0 & 0 & \dots & \lambda \end{vmatrix}_{(p+q)\times(p+q).}$$

Using Lemma 2.3,

$$\phi_{GR}(S(p,q),\lambda) = \lambda^{p+q-4}\begin{vmatrix} \lambda^2-(q-1)q^{2\alpha} & -\lambda(pq)^{\alpha}\\ -\lambda(pq)^{\alpha} & \lambda^2-(p-1)p^{2\alpha}\end{vmatrix}.$$

Now, by the straightforward computation we have the result.

(ii) It follows from Remark 1. □

3 Conclusion

In this paper we obtained the expression for the GR polynomial and GR energy of some specific graphs. These results generalise the results obtained in paper [1]. The results in [1] follow from our work by letting $\alpha = -(1/2)$.

Acknowledgement: The author H. S. Ramane is thankful to UGC, New Delhi for support through research grant under UGC-SAP DRS-III for 2016-2021: F. 510/3/DRS-III/2016(SAP-I) Dated: 29th Feb. 2016. Another author G. A. Gudodagi is thankful to the Karnatak University, Dharwad for UGC-UPE scholarship No. KU/SCH/UGC-UPE/2014-15/901 and also KLE's G. I. Bagewadi College, Nipani for support.

References

[1] S. Alikhani and N. Ghanbari, "Randić energy of specific graphs," *Appl. Math. Comput.*, vol. 269, pp. 722–730, 2015.

[2] B. Bollobás and P. Erdös, "Graphs of extremal weights," *Ars Combin.*, vol. 50, pp. 225–233, 1998.

[3] S. B. Bozkurt, A. D. Güngör, I. Gutman, and A. S. Cevik, "Randić matrix and Randić energy," *MATCH Commum. Math. Comput. Chem.*, vol. 64, pp. 239–250, 2010.

[4] S. B. Bozkurt, A. D. Güngör, and I. Gutman, "Randić spectral radius and Randić energy", *MATCH Commum. Math. Comput. Chem.*, vol. 64, pp. 321–334, 2010.

[5] S. B. Bozkurt and D. Bozkurt, "On incidence energy," *MATCH Commun. Math. Comput. Chem.*, vol. 72, pp. 215–225, 2014.

[6] S. B. Bozkurt and I. Gutman, "Estimating the incidence energy," *MATCH Commun. Math. Comput. Chem.*, vol. 70, pp. 143–156, 2013.

[7] L. Chen and Y. Shi, "Maximal matching energy of tricyclic graphs," *MATCH Commun. Math. Comput. Chem.*, vol. 73, pp. 105–119, 2015.

[8] D. Cvetković, M. Doob, and H. Sachs, *Spectra of graphs-Theory and Applications*, New York: Academic Press, 1980.

[9] M. Dehmer, M. Moosbrugger, and Y. Shi, "Encoding structural information uniquely with polynomial-based descriptors by employing the Randić matrix," *Appl. Math. Comput.*, vol. 268, pp. 164–168, 2015.

[10] P. Erdös, A. Rényi, and V. T. Sós, "On a problem of graph theory," *Studia Sci. Math. Hungar.*, vol. 1, pp. 215–235, 1966.

[11] I. Gutman and O. E. Polansky, *Mathematical Concepts in Organic Chemistry*, Berlin: Springer–Verlag, 1986.

[12] I. Gutman, "Topology and stability of conjugated hydrocarbons. The dependence of total π-electron energy on molecular topology," *J. Serb. Chem. Soc.*, vol. 70, pp. 441–456, 2005.

[13] I. Gutman, B. Furtula, and S. B. Bozkurt, "On Randić energy," *Linear Algebra Appl.*, vol. 442, pp. 50–57, 2014.

[14] I. Gutman, X. Li and J. Zhang, "Graph energy," in: *Analysis of Complex Networks. From Biology to Linguistics*, M. Dehmer and F. Emmert-Streib, Eds., Weinheim: Wiley-VCH, 2009, pp. 145–174.

[15] I. Gutman, "The energy of a graph," *Ber. Math. Statist. Sekt. Forschungsz. Graz*, vol. 103, pp. 1–22, 1978.

[16] I. Gutman, "Degree-based topological indices," *Croat. Chem. Acta.*, vol. 86, pp. 351–361, 2013.

[17] I. Gutman and B. Zhou, "Laplacian energy of a graph", *Linear Algebra Appl.*, vol. 414, pp. 29–37, 2006.

[18] G. Indulal, I. Gutman, and A. Vijayakumar, "On distance energy of graphs", *MATCH Commun. Math. Comput. Chem.*, vol. 60, pp. 461–472, 2008.

[19] S. Ji, X. Li, and Y. Shi, "Extremal matching energy of bicyclic graphs," *MATCH Commun. Math. Comput. Chem.*, vol. 70, pp. 697–706, 2013.

[20] X. Li, Y. Shi, and I. Gutman, *Graph Energy*, NewYork: Springer, 2012.

[21] H. H. Li, Y. X. Zhou, and L. Su, "Graphs with extremal matching energies and prescribed parameters," *MATCH Commun. Math. Comput. Chem.*, vol. 72, pp. 239–248, 2014.

[22] H. S. Ramane, K. C. Nandeesh, I. Gutman, and X. Li, "Skew equienergetic digraphs," *Trans. Comb.*, vol. 5, pp. 15–23, 2016.

[23] M. Randić, "On characterization of molecular branching," *J. Amer. Chem. Soc.*, vol. 97, pp. 6609–6615, 1975.

Harishchandra S. Ramane, Gouramma A. Gudodagi, Received September 14, 2019
Revised December 21, 2019

Harishchandra S. Ramane
Department of Mathematics,
Karnatak University,
Dharwad - 580003, India
Phone:+919945031752
E–mail: hsramane@yahoo.com

Gouramma A. Gudodagi
Department of Mathematics,
KLE's G. I. Bagewadi Arts, Science and Commerce College,
Nipani – 591237, India
Phone:+918095319099
E–mail: gouri.gudodagi@gmail.com

Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

Digital signature scheme with doubled verification equation

D.N. Moldovyan A.A. Moldovyan N.A. Moldovyan

Abstract

A novel design of the signature schemes based on the hidden discrete logarithm problem is proposed, which is characterized in using special criterion oriented to providing security to potential quantum attacks. The criterion consists in the requirement to ensure practical intractability of the task of constructing a periodic function with a period depending on the value of a discrete logarithm in a hidden cyclic group. A signature scheme satisfying the mentioned criterion is introduced. A 4-dimensional finite non-commutative associative algebra is considered as algebraic support. To implement the signature scheme, a commutative hidden group defined by generator system $< N, Q >$, where vectors N and Q have the same prime order, is exploited. For further development of the introduced method, an 8-dimensional algebra is proposed.

Keywords: finite non-commutative algebra, hidden logarithm problem, public-key cryptoscheme, digital signature, post-quantum cryptoscheme.

MSC 2010: 94A60, 16Z05, 14G50, 11T71, 16S50.

1 Introduction

Development of practical post-quantum signature schemes represents a current challenge in the area of the applied and theoretic cryptography [1], [2]. Currently nine signature schemes proposed in framework of the NIST competition [3] are considered as candidates for post-quantum signature standard. A significant disadvantage of those schemes is a large size of public key and signature, except for GeMSS

and Rainbow signature schemes. In the latter, the signature size is relatively small, but the public key size is extremely large. In terms of the trade off between performance and size of the public key and the signature, the preferred post-quantum signature schemes are Falcon-512 (657-byte signature; 897-byte public key) and Dilithium-1024x768 (2044-byte signature; 1184-byte public key).

A promissing approach to the design of the public-key post-quantum cryptoschemes with sorter size of signature and public key represents using so called hidden discrete logarithm problem (HDLP) as the post-quantum cryptographic primitive [4], [5]. Several HDLP-based signature schemes are described in the papers [6], [7]. Usually the HDLP used in the signature schemes is set in the m-dimensional ($m = 4, 6$) finite non-commutative associative algebras (FNAAs) as follows.

One selects a random integer $x < q$ and a random cyclic group contained in the used FNAA and generated by some m-dimensional vector N having order equal to the prime q of sufficiently large size. Then he computes vector N^x and performs homomorphism-map operations ψ_1 and ψ_2 obtaining public key in the form of the following two vectors $Y = \psi_1(N^x)$ and $Z = \psi_2(N)$ or three vectors (Y, Z, T), where vector T plays the role of a fitting parameter in verification equation. The cyclic group generated by the vector N is called the base group. The vectors Y, Z, and T are contained in other three different cyclic groups (contained in the used FNAA as different subsets of vectors).

Due to mutual commutativity of each of the masking operations ψ_1 and ψ_2 with the exponentiation operation, different signature schemes based on the computational difficulty of the discrete logarithm problem (see, for example, [8], [9]) can be used as prototypes of the HDLP-based cryptoschemes.

The earlier proposed rationale of the security of the known HDLP-based signature schemes to the quantum attacks (attacks with using a hypothetic quantum computer) is quite straightforward: in the case of the HDLP-based signature schemes, potential attacker knows no elements of the base cyclic group in which the exponentiation operation is performed, therefore, to compute the value x, one cannot directly use

the Shor quantum algorithm [10] for finding logarithm in a cyclic group.

For a more convincing justification of the security to quantum attacks, an additional criterion can be adopted, which is aimed at preventing the possibility of constructing periodic functions with a period depending on the value of the discrete logarithm in the base cyclic group, regardless of the fact that the periodic function takes on only values from the same finite group. The HDLP-based signature schemes proposed in [6],[7],[11] do not satisfy this criterion, since one can define the following periodic function $F(i,j) = Y^i \circ T \circ Z^j$ in two integer variables i and j, which contains a period with the length equal to $(-1, x)$. Indeed, we have $Y^i \circ T \circ Z^j = Y^{i-1} \circ T \circ Z^{j+x}$. This function takes on values in different groups contained in the FNAA used as algebraic carrier of the signature scheme, however, one can suppose that an advanced quantum algorithm for evaluating the period of the function $F(i,j)$ can be potentially developed.

In the present paper a new HDLP-based signature scheme is proposed which meets the criterion of ensuring practical intractability of the task of constructing a periodic function with a period depending on the value of a discrete logarithm in a hidden cyclic group. The proposed criterion introduces significant limitations in the development of the HDLP-based signature schemes, which were overcome by using a three-element signature and doubling the verification equation. Besides, a commutative group defined by generator system $< N, Q >$, where the vectors N and Q have the same prime order, is applied as the hidden group in which the basic exponentiation operation is performed. A 4-dimensional FNAAs set over the ground finite field $GF(p)$, where prime $p = 2q + 1$ and q is a 255-bit prime, are proposed as algebraic support for implementing the proposed signature scheme. This algebra contains p^2 global left-sided units and p^2 different isomorphic commutative groups of the order $(p-1)^2$. As a promising algebraic support for further development of the proposed method for constructing post-quantum signature schemes, an 8-dimensional algebra with a global two-sided unit is proposed.

2 Preliminaries

2.1 Defining FNAAs

Suppose the m-dimensional vector space is defined over the ground finite field $GF(p)$. Introducing the vector multiplication operation that is distributive at the left and at the right relatively the addition operation, one gets the m-dimensional finite algebra. If the defined multiplication operation is non-commutative and associative, then we have FNAA. To define the vector multiplication operation, one can use the notion of formal basis vectors denoted as $\mathbf{e}_0 = (1, 0, 0 \ldots, 0)$, $\mathbf{e}_1 = (0, 1, 0 \ldots, 0)$, ... $\mathbf{e}_{m-1} = (0, 0 \ldots, 0, 1)$ and representation of some two vectors $A = (a_0, a_1, \ldots a_{m-1})$ and $B = (b_0, b_1, \ldots b_{m-1})$ in the form of the following summs of the single component vectors $a_i\mathbf{e}_i$ and $b_i\mathbf{e}_i$: $A = \sum_{i=0}^{m-1} a_i\mathbf{e}_i$ and $B = \sum_{j=0}^{m-1} b_j\mathbf{e}_j$.

The vector multiplication operation (denoted as $\circ$) is defined by the following formula $A \circ B = \sum_{j=0}^{m-1} \sum_{i=0}^{m-1} a_i b_j \, (\mathbf{e}_i \circ \mathbf{e}_j)$, where the product $\mathbf{e}_i \circ \mathbf{e}_j$ for all possible pairs of the integers i and j is to be replaced by some single-component vector $\lambda\mathbf{e}_k$. The rule of the mentioned substitution is usually given by so called basis vector multiplication table (BVMT), like Table 1 (see Subsection 2) and Table 2 (Section 3).

It is assumed that the intersection of the ith row and the jth column defines the cell which contains the value $\lambda\mathbf{e}_k = \mathbf{e}_i \circ \mathbf{e}_j$, where the value $\lambda \neq 1$ is called structural coefficient. To build a FNAA, one should compose and use some BVMT defining non-commutative associative multiplication operation. Clearly, to implement the associativity property, it is sufficient to use the BVMP for which the condition $(\mathbf{e}_i \circ \mathbf{e}_j) \circ \mathbf{e}_k = \mathbf{e}_i \circ (\mathbf{e}_j \circ \mathbf{e}_k)$ holds true for all possible triples (i, j, k).

2.2 Finite algebra with multiplicative group possessing two-dimensional cyclicity

In the paper [12] it is shown that the multiplicative group Γ of the finite 2-dimensional commutative algebra with the multiplication operation defined by Table 1, where the structural coefficient λ is a quadratic residue in $GF(p)$, has order $\Omega = (p-1)^2$ and includes the generator

system $< G'_1, G'_2 >$, where each of the vectors G'_1 and G'_2 has order $\omega = p - 1$. One can easily show the group Γ' contains p finite cyclic groups Γ'_c of the order $p-1$. The finite groups generated by a generator system in which each element has the same order value are called groups with multi-dimensional cyclicity [13].

When constructing public-key cryptosystems based on the computational complexity of the discrete logarithm problem, one uses cyclic groups whose order is equal to a prime number of sufficiently large size. This defines interest to the case of defining the finite algebras over the field $GF(p)$ whose characteristic p is such that the integer $p-1$ contains a large prime divisor, for example $p = 2q+1$, where q is a prime. In the last case the group Γ' contains the commutative subgroup Γ generated by the generator system $< G_1, G_2 >$, in which each of the vectors G_1 and G_2 has order q. Evidently, some fixed integers i and j ($0 < i < q$; $0 < j < q$) define the vector $G_{ij} = G_1^i \circ G_2^j$ having order equal to q, which is a generator of some cyclic group Γ_c of the prime order q. One can easily see that the following proposition holds true.

Table 1. The BVMT setting the 2-dimensional commutative algebra.

$\circ$	$\mathbf{e}_0$	$\mathbf{e}_1$
$\mathbf{e}_0$	$\mathbf{e}_0$	$\mathbf{e}_1$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\lambda\mathbf{e}_0$

Proposition 1. *For $k = 0, 1, \ldots, q-1$ each of the formulas $G_k = G_{ij} \circ G_1^k$ and $G_k = G_{ij} \circ G_2^k$, where $i, j = 1, 2, \ldots, q-1$, defines q generators of q different cyclic groups having order q.*

Proposition 1 is used in the designed signature schemes (see Sections 4 and 5) to prevent construction of the periodic functions on the basis of using the elements of the public key, the period of which is defined by discrete logarithm in the hidden cyclic group. One can note that the subgroup Γ contains $q^2 - 1$ elements $G \neq (1, 0)$ that are distributed among $q + 1$ different cyclic groups of order q which include only one common element, namely, the unit element (1,0).

3 Algebraic support of the proposed signature scheme

3.1 The used 4-dimensional algebra

For development of the HDLP-based signature scheme satisfying the criterion of practical intractability of the task of constructing a periodic function with a period depending on the value of a discrete logarithm in a hidden cyclic group we have used the 4-dimensional FNAA containing p^2 different global left-sided units L, which is defined over the field $GF(p)$ using the BVMT presented as Table 2.

To obtain the formula describing the set of the L-units, the following vector equation is to be considered: $X \circ A = A$, where $A = (a_0, a_1, a_2, a_3)$ is a fixed 4-dimensional vector and $X = (x_0, x_1, x_2, x_3)$ is the unknown. Using Table 2, one can represent the vector equation in the form of the following system of four linear equations:

$$\begin{cases} (x_0 + x_1)\, a_0 + \lambda\, (x_2 + x_3)\, a_2 = a_0; \\ (x_0 + x_1)\, a_2 + (x_2 + x_3)\, a_0 = a_2; \\ (x_0 + x_1)\, a_1 + \lambda\, (x_2 + x_3)\, a_3 = a_1; \\ (x_0 + x_1)\, a_3 + (x_2 + x_3)\, a_1 = a_3. \end{cases} \tag{1}$$

Using the variable substitution $u_1 = x_0 + x_1$ and $u_2 = x_2 + x_3$, one can represent the system (1) in the form of the following two independent systems of two linear equations:

$$\begin{cases} u_1 a_0 + \lambda u_2 a_2 = a_0; \\ u_1 a_2 + u_2 a_0 = a_2; \end{cases} \tag{2}$$

$$\begin{cases} u_1 a_1 + \lambda u_2 a_3 = a_1; \\ u_1 a_3 + u_2 a_1 = a_3. \end{cases} \tag{3}$$

For arbitrary vector A satisfying the conditions $a_0^2 \neq \lambda a_1^2$ and $a_1^2 \neq \lambda a_3^2$, each of the systems (2) and (3) has the same unique solution $u_1 = 1$ and $u_2 = 0$. One can easily see that the indicated solution satisfies the systems (2) and (3) for all elements of the considered FNAA (in

Table 2. The BVMT for defining the 4-dimensional FNAA ($\lambda \neq 0$).

$\circ$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_0$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_1$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_2$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\lambda\mathbf{e}_0$	$\lambda\mathbf{e}_1$
$\mathbf{e}_3$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\lambda\mathbf{e}_0$	$\lambda\mathbf{e}_1$

the cases $a_0^2 \neq \lambda a_1^2$ and $a_1^2 \neq \lambda a_3^2$ there exist some additional solutions defining local left-sided units). Thus, the solution $(u_1, u_2) = (1, 0)$ defines the set of the global left-sided units X, the coordinates of which satisfy the conditions $x_0 + x_1 = u_1 = 1$ and $x_2 + x_3 = u_2 = 0$. These left-sided units are called global, since every of them acts as the left-sided unit on every vector in the FNAA. The set of p^2 global left-sided units L is described as follows:

$$L = (l_0, l_1, l_2, l_3) = (h, 1-h, k, -k)\,, \tag{4}$$

where $h, k = 0, 1, 2, \ldots p-1$.

The considered FNAA contains local right-sided units R acting in some subsets of the algebra elements. The local right-sided unit R_A relating to some vector A can be computed as solution of the vector equation $A \circ X = A$ that can be easily reduced to the following two systems of two linear equations:

$$\begin{cases} (a_0 + a_1)\, x_0 + \lambda\,(a_2 + a_3)\, x_2 = a_0; \\ (a_2 + a_3)\, x_0 + (a_0 + a_1)\, x_2 = a_2; \end{cases} \tag{5}$$

$$\begin{cases} (a_0 + a_1)\, x_1 + \lambda\,(a_2 + a_3)\, x_3 = a_1; \\ (a_2 + a_3)\, x_1 + (a_0 + a_1)\, x_3 = a_3. \end{cases} \tag{6}$$

Each of the systems (5) and (6) has the same main determinant Δ_A:

$$\Delta_A = (a_0 + a_1)^2 - \lambda\,(a_2 + a_3)^2\,. \tag{7}$$

Thus, in the case $\Delta_A \neq 0$, the equation $A \circ X = A$ has a unique solution $X = R_A$ and the single right-sided unit $R_A = (r_0, a_1, r_2, a_3)$ relates to the vector A. One can obtain the following formula for computing the value R_A:

$$R_A = \left(\frac{a_0 (a_0 + a_1) - \lambda a_2 (a_2 + a_3)}{\Delta}, \frac{a_1 (a_0 + a_1) - \lambda a_3 (a_2 + a_3)}{\Delta}, \right.$$
$$\left. \frac{a_1 a_2 - a_0 a_3}{\Delta}, \frac{a_0 a_3 - a_1 a_2}{\Delta} \right). \tag{8}$$

The value R_A acts as the local right-sided unit in the frame of the sequence of the vectors $A, A^2, \ldots, A^i, \ldots$. Besides, the latter sequence is periodic and composes a finite cyclic group with the unit R_A, i. e., the element R_A is the single local two-sided unit E_A relating to the vector A (and to cyclic group generated by the vector A).

Proposition 2. *The local right-sided unit R_A is simultaneously the local two-sided unit E_A relating to the vector A.*

Proof. It is sufficient to show that the vector R_A is contained in the set (4) of the global left-sided units. Suppose in (4) we have $h = r_0$ and $k = r_2$. Then one can compute

$$1 - h = 1 - r_0 = 1 - \frac{a_0 (a_0 + a_1) - \lambda a_2 (a_2 + a_3)}{\Delta} =$$
$$= \frac{a_1 (a_0 + a_1) - \lambda a_3 (a_2 + a_3)}{\Delta} = r_1;$$

$$-k = -r_2 = -\frac{a_1 a_2 - a_0 a_3}{\Delta} = \frac{a_0 a_3 - a_1 a_2}{\Delta} = r_3.$$

Thus, the vector R_A is equal to the global left-sided unit corresponding to the integers $h = r_0$ and $k = r_2$. □

Proposition 3. *Suppose the vector A is such that $\Delta_A \neq 0$. Then there exists some integer ω such that $A^\omega = E_A$ and the local two sided-unit E_A is the unit of the cyclic group generated by the vector A.*

Proof. Let us consider the sequence of the vectors $A, A^2, \dots A^h, \dots A^k, \dots$. For all integer values i one has $A^i \neq O$, where $O = (0,0,0,0)$, since $\Delta_A \neq 0$. Due to finiteness of the considered algebras and condition $\Delta_A \neq 0$, the indicated sequence is periodic, i. e., for some integer h and some minimum integer $k > h$ we have the following:

$$A^k = A^h \Rightarrow A^h \circ A^{k-h} = A^h \Rightarrow A^{h-1} \circ \Big(A \circ A^{k-h} - A\Big) = O \Rightarrow$$
$$A \circ A^{k-h} - A = O \Rightarrow A \circ A^{k-h} = A \Rightarrow A^{k-h} = R_A,$$
$$A \circ A^{k-h} = A \Rightarrow A^{k-h} \circ A = A \Rightarrow A^{k-h} = L_A = R_A = E_A.$$

Thus, the vector E_A is the unit of the cyclic group containing elements $\big\{A, A^2, \dots A^\omega\big\}$, where $\omega = k - h$, and Proposition 3 holds true. □

The Proposition 3 shows $A^{\omega-i} \circ A^i = A^i \circ A^{\omega-i} = E_A$, i. e., the vector $A^{\omega-i}$ is the inverse value of the vector A^i relatively the local two-sided unit E_A. Therefore, the value ω can be called the local order of the vector A and the last can be called a locally invertible vector.

Proposition 4. *If the vector equation $X \circ A = B$ has solution $X = S$, where $\Delta_S \neq 0$, then p^2 different values $X_i = S \circ L_i$, where L_i takes on all values from the set (4), are also solutions of the given equation.*

Proof. $(S \circ L_i) \circ A = S \circ (L_i \circ A) = S \circ A = B$. Suppose $S \circ L_i = S \circ L_j$, then $S \circ (L_i - L_j) = (0,0,0,0)$ and $L_i = L_j$. Therefore, the number of different solutions is equal to the number of different L-units, i. e., to p^2. The Proposition 4 is proven. □

Proposition 5. *Suppose the vector L is a global left-sided unit. Then the map of the FNAA defined by the formula $\varphi_L(X) = X \circ L$, where the vector X takes on all values in the algebra, is a homomorphism.*

Proof. For two arbitrary vectors X_1 and X_2 we have
$$\varphi_L(X_1 \circ X_2) = (X_1 \circ X_2) \circ L = (X_1 \circ L) \circ (X_2 \circ L) =$$
$$= \varphi_L(X_1) \circ \varphi_L(X_2);$$
$$\varphi_L(X_1 + X_2) = (X_1 + X_2) \circ L = X_1 \circ L + X_2 \circ L =$$
$$= \varphi_L(X_1) + \varphi_L(X_2).$$
□

Proposition 6. *All locally invertible vectors of the considered 4-dimensional FNAA form p^2 different groups with p^2 different units $E = (h, 1-h, k, -k)$, where $h, k = 0, 1, 2, \dots p-1$.*

Proof. Suppose the set $\{A_1, A_2, \dots A_i, \dots A_\Omega\}$ of locally invertible vectors includes all vectors relating to a fixed local two-sided unit E (including the vector E) and only such vectors. It is easy to see this set is the group Γ_E with the unit E. Every fixed global left-sided unit L' from the set (4) is the unit E' of some group $\Gamma_{E'}$ representing a set of locally invertible vectors $\{A'_1, A'_2, \dots A'_i, \dots A'_\Omega\}$. Indeed, due to the Proposition 5, we have $A'_i = A_i \circ L'$ for $i = 1, 2, \dots \Omega$, and $E' = E \circ L' = L'$. The considered FNAA contains p^2 different global left sided units $E = (h, 1-h, k, -k)$, where $h, k = 0, 1, 2, \dots p-1$, every one of which defines a unique group of the order Ω. □

Proposition 7. *If the structural coefficient λ is a quadratic non-residue, then the considered 4-dimensional FNAA contains $p^2\left(p^2-1\right)$ locally invertible vectors. If λ is a quadratic residue, then the algebra contains $p^2\left(p-1\right)^2$ locally invertible vectors.*

Proof. Condition of the local invertibility of the vector A is $\Delta_A \neq 0$. Let us compute the number of non-invertible vectors using the condition $\Delta_A = 0$. Using the formula (7), one can represent the last condition in the form of the following equation

$$a_0^2 + 2a_0a_1 + a_1^2 - \lambda\left(a_2 + a_3\right)^2 = 0.$$

If the coordinates of the vector A satisfy the last equation, then A is a non-invertible vector. Solving the equation relatively the unknown value a_0, one can get $a_0 = -a_1 \pm \sqrt{\lambda\left(a_2 + a_3\right)^2}$.

If λ is a quadratic non-residue, then solution exists only in the case $a_2 + a_3 = 0$, i. e. we have p different variants of the values of coodinates a_2 and a_3. In every of such variants the solution exists for arbitrary value a_1. Thus, the number of non-invertible vectors is equal to p^2. Correspondingly, the number μ of invertible vectors contained in the algebra is equal to $\mu = p^4 - p^2 = p^2\left(p^2-1\right)$.

If λ is a quadratic residue, then we have one value of the square root for the case $a_2 + a_3 = 0$ (p variants of the pairs of the values (a_2, a_3) : $a_2 + a_3 = 0$) and p^2 variants of the triples (a_1, a_2, a_3) for which the considered equation has a solution, i. e., p^2 non-invertible vectors. For the case $a_2 + a_3 \neq 0$ we have two values of the square root ($p^2 - p$ variants of the pairs of the values (a_2, a_3) : $a_2 + a_3 \neq 0$) and $2p\left(p^2 - p\right)$ variants of the triples (a_1, a_2, a_3) for which the considered equation has a solution, i. e., in the second case we have $2p\left(p^2 - p\right)$ non-invertible vectors.

Thus, taking into account both of the cases, one gets the number of non-invertible vectors equal to $2p\left(p^2 - p\right) + p^2 = 2p^3 - p^2$ and the value $\mu = p^4 - \left(2p^3 - p^2\right) = p^2\left(p - 1\right)^2$. The Proposition 7 is proven. □

Proposition 8. *The considered 4-dimensional FNAA contains p^2 isomorphic commutative groups and every locally invertible vector of the algebra is contained only in one of these groups. If the structural coefficient λ is a quadratic residue (non-residue), then every of these groups is cyclic (has 2-dimensional cyclicity) and its order is equal to $\Omega = p^2 - 1$ $\left(\Omega = (p-1)^2\right)$.*

Proof. Due to the Proposition 6, one should only derive a formula for the order Ω of every of p^2 isomorphic groups contained in the considered algebra and show that the algebra contains at least one cyclic group or one commutative group having 2-dimensional cyclicity.

Clearly we have $\Omega = \frac{\mu}{p^2} = p^2 - 1$, if the value λ is a quadratic residue, and $\Omega = \frac{\mu}{p^2} = (p-1)^2$, if the value λ is a quadratic non-residue.

One can easily see that the set of the vectors $(h, 0, k, 0)$, where $h, k = 0, 1, 2, \ldots p - 1$, represents subalgebra that is isomorphic with the commutative 2-dimensional algebra described in Subsection 2.2, therefore, the set of the invertible vectors in this subalgebra represents a cyclic group, if the value λ is a quadratic non-residue, or commutative group with 2-dimensional cyclicity, if the value λ is a quadratic residue [12]. The Proposition 8 is proven. □

Suppose the vector B is such that $\Delta_B \neq 0$ and L is a random global left-sided unit L. One can compute the single vector A that satisfies

the condition

$$B \circ A = L. \tag{9}$$

The main determinant of the system of linear equations, which corresponds to the vector equation (9), is equal to $\Delta_B \neq 0$, therefore, the equation (9) has a unique solution.

Proposition 9. *Suppose $B \circ A = L$. Then the formula $\psi_L = A \circ X \circ B$, where the vector X takes on all values in the considered 4-dimensional FNAA, sets the homomorphism map.*

Proof. For two arbitrary 4-dimensional vectors X_1 and X_2, one can get the following:
$\psi_L(X_1 \circ X_2) = A \circ (X_1 \circ X_2) \circ B = A \circ (X_1 \circ L \circ X_2) \circ B =$
$= (A \circ X_1 \circ B) \circ \left(A \circ X_2 \circ B^t\right) = \psi_L(X_1) \circ \psi_L(X_2)$;
$\psi_L(X_1 + X_2) = A \circ (X_1 + X_2) \circ B = (A \circ X_1 \circ B) + (A \circ X_2 \circ B) =$
$= \psi_L(X_1) + \psi_L(X_2)$. □

Proposition 10. *The homomorphism-map operation $\psi_L(X) = A \circ X \circ B$ and the exponentiation operation X^k are mutually commutative, i. e., the equality $A \circ X^k \circ B = (A \circ X \circ B)^k$ holds true.*

Proof. Due to Proposition 9, we have $\psi_L(X^k) = (\psi_L(X))^k$, i. e., $A \circ X^k \circ B = (A \circ X \circ B)^k$. □

3.2 Perspective 8-dimensional algebra

In algebras with a global two-sided unit, local masking operations can be applied that operate within the set of non-invertible elements of the algebra. Methods for setting local masking operations are quite diverse and are of interest for building digital signature schemes (see, for example, [14]). The possibility of setting new types of masking operations is due to the fact that a large number of local left-sided units and a large number of local right-sided units operate simultaneously on some fixed subsets of non-invertible elements. Implementation of the signature schemes with doubled verification equation on the base on FNAAs of such type represent significant interest. However, the

developer needs to use FNAAs containing commutative groups with two-dimensional cyclicity.

Algebra with the multiplication operation specified in Table 3 solves this problem when selecting a structural coefficient λ equal to the quadratic residue in the field $GF(p)$. However, this algebra is an object of independent research focused on obtaining formulas that define the criterion of non-invertibility of vectors and describe the sets of local right-sided and local left-sided units for a fixed non-invertible vector, as it had been done for 4-dimensional FNAAs considered in the works [6], [14].

Besides, one should develop a procedure for computing the generator system $< N, Q >$ defining the hidden commutative group with two-dimensional cyclicity. The following three possibilities represent interest for implementing the signature schemes with doubled verification equation: i) N is a non-invertible vector and Q is invertible; ii) N and Q are non-invertible vectors; iii) N and Q are invertible vectors. In each of the cases for many different fixed pairs of integers (u, w) the vectors $N^u \circ Q^w$ are generators of different cyclic groups of the same order q.

Table 3. The BVMT defining the 8-dimensional FNAA with global two-sided unit ($\lambda \neq 0$, $\mu \neq 0$, $\mu \neq 1$)

$\circ$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_6$	$\mathbf{e}_7$
$\mathbf{e}_0$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mu\mathbf{e}_6$	$\mu\mathbf{e}_7$	$\mu\mathbf{e}_0$	$\mu\mathbf{e}_1$	$\mathbf{e}_6$	$\mathbf{e}_7$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\lambda\mathbf{e}_0$	$\mu\mathbf{e}_7$	$\lambda\mu\mathbf{e}_6$	$\mu\mathbf{e}_1$	$\lambda\mu\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\mathbf{e}_6$
$\mathbf{e}_2$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_2$	$\mathbf{e}_3$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_3$	$\mathbf{e}_5$	$\lambda\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\mathbf{e}_2$	$\mathbf{e}_5$	$\lambda\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\mathbf{e}_2$
$\mathbf{e}_4$	$\mathbf{e}_4$	$\mathbf{e}_5$	$\mu\mathbf{e}_2$	$\mu\mathbf{e}_3$	$\mu\mathbf{e}_4$	$\mu\mathbf{e}_5$	$\mathbf{e}_2$	$\mathbf{e}_3$
$\mathbf{e}_5$	$\mathbf{e}_5$	$\lambda\mathbf{e}_4$	$\mu\mathbf{e}_3$	$\lambda\mu\mathbf{e}_2$	$\mu\mathbf{e}_5$	$\lambda\mu\mathbf{e}_4$	$\mathbf{e}_3$	$\lambda\mathbf{e}_2$
$\mathbf{e}_6$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_6$	$\mathbf{e}_7$	$\mathbf{e}_0$	$\mathbf{e}_1$	$\mathbf{e}_6$	$\mathbf{e}_7$
$\mathbf{e}_7$	$\mathbf{e}_1$	$\lambda\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\mathbf{e}_6$	$\mathbf{e}_1$	$\lambda\mathbf{e}_0$	$\mathbf{e}_7$	$\lambda\mathbf{e}_6$

For the said 8-dimensional algebra, one can select a random invertible vector N of the order q and number β having order q in $GF(p)$, for

which the pair of the vectors N and $Q = \beta N$ (scalar multiplication) with high probability compose the generator system of some commutative group with two-dimensional cyclicity. When using this pair of vectors and various variants of the automorphism map operation as masking operations, the said 8-dimensional FNAA can be used as algebraic support for implementing some versions of the signature scheme described in the next section.

For example, in the case of prime $p = 2q + 1 = 501659$, prime $q = 250829$, $\lambda = 4$, $\mu = 2$, and $\beta = 123456$ we have:
$N = (22334; 57857; 35656; 45457; 17645; 61268; 62597; 57864)$
$Q = \beta N =$
$= (148440; 172950; 391070; 381818; 177742; 389465; 419996; 33824)$
$N^q = Q^q = E = (501658;\ 0;\ 501658;\ 0;\ 1;\ 0;\ 2;\ 0)$,
where E is the global two-sided unit.

4 The proposed signature scheme

4.1 Generation of the hidden commutative group

The 4-dimensional FNAA described in Section 3 and defined over the field $GF(p)$ with characteristic $p = 2q + 1$, where q is a 255-bit prime, is used as algebraic support of the designed signature scheme. The hidden finite group $\Gamma_{<N,Q>}$ is generated as computation of its basis $< N, Q >$ that includes two vectors N and Q each of which has order equal to the prime q. The basis $< N, Q >$ is computed as follows:

1. Generate a random value d that is a primitive element modulo p. The primitive element d defines a locally invertible vector $G_1 = (d, 0, 0, 0)^z \neq (1, 0, 0, 0)$, where $z = \frac{p-1}{q}$, having order equal to the prime q.

2. Generate the vector $G_2 = (b, 0, r, 0)^z$, where $b < p - 1$ and $r < p - 1$ are such random numbers that the vector G_2 has order equal to q. (For example, generate several different pairs of random numbers b' and r' and compute $G'_2 = (b', 0, r', 0)^z$ and take the value $G'_2 \neq (1, 0, 0, 0)$ as the vector G_2.)

3. Generate a random global left-sided unit L_r and a random num-

ber $u < q$ and compute the vectors N and Q of the order q as follows:

$$N = G_1 \circ G_2^u \circ L_r; \qquad Q = G_2 \circ L_r.$$

The vectors G_1 and G_2 represent the basis $< G_1, G_2 >$ of the commutative group with the unit element equal to $E = (1, 0, 0, 0)$ (see the proof of the Proposition 8). Since the multiplication at the right by any global left-sided unit defines a homomorphism map of the algebra, the vectors N and Q define the basis $< N, Q >$ of some commutative group of the order equal to q^2, which has 2-dimensional cyclicity. It is easy to see the unit element of the hidden commutative group $\Gamma_{<N,Q>}$ is the vector $E = (1, 0, 0, 0) \circ L_r = L_r$.

4.2 Generation of parameters of masking operations

The exponentiation operations performed in two different cyclic groups contained in the hidden commutative group are used as base operations. Vector N is used as generator of the first of these cyclic groups. Generator J of the second cyclic group is computed as follows:

1. Generate two random integers $t < q$ and $u < q$.
2. Compute the vector $J = N^t \circ Q^u$.

The values N, J, N^x, and J^x, where $x < q$ is an element of the private key, are used to compute the elements $\psi_0 (N \circ Q)$, $\psi_1 (N^x)$, $\psi_0 (J \circ Q)$, and $\psi_2 (J^x)$, of the public key. Thus, the elements N and J are masked performing multiplication by the vector Q followed by performing the ψ_0-map operation, and the vectors N^x and J^x are masked performing the ψ_1-map and ψ_2-map operations, correspondingly. Parameters of the homomorphism-map operations $\psi_0(X) = C \circ X \circ D$, $\psi_1(X) = A_1 \circ X \circ B_1$, and $\psi_2(X) = A_2 \circ X \circ B_2$ are computed as follows:

1. Select a random global left-sided unit L_0 (for example, using the formula (4)), generate a random locally invertible vector D, and compute the value C as solution of the following vector equation $D \circ C = L_0$. (It has a unique solution, since $\Delta_D \neq 0$.)

2. Select a random global left-sided unit L_1, generate a random locally invertible vector B_1, and compute the value A_1 as solution of the vector equation $B_1 \circ A_1 = L_1$.

3. Generate two random integers $h < p - 1$ and $k < p - 1$, take the

global left-sided unit $L_2 = (h, 1-h, k, -k)$, generate a random locally invertible vector B_2 ($\Delta_{B_2} \neq 0$), and compute the value A_2 as solution of the vector equation $B_2 \circ A_2 = L_2$.

4.3 Public and private kyes

Public key represents the following two triples of the vectors (Y_1, Z_1, T_1) and (Y_2, Z_2, T_2) which are computed as follows:

1. $Y_1 = A_1 \circ N^x \circ B_1$; $\quad Z_1 = C \circ N \circ Q \circ D$.
2. $Y_2 = A_2 \circ J^x \circ B_2$: $\quad Z_2 = C \circ J \circ Q \circ D$.
3. $T_1 = A_1 \circ D \circ L$, where L is a random global left-sided unit.
4. $T_2 = A_2 \circ D \circ L'$, where L' is a random global left-sided unit.

One can consider private key as the set of all secret elements used to compute the public key and the signature. In another interpretation, the private key is a set of secret elements that are needed to calculate only the signature. We will use the second interpretation for which we have the private key representing the set of the values N, J, x, Q, A_1, A_2, and D.

4.4 Signature generation procedure

Suppose one is to compute a signature to the electronic document M, using some specified 256-bit hash-function f_H. The signature generation algorithm is as follows:

1. Generate a random integer $k < q$ and a random locally invertible 4-dimensional vector K. Then compute the vectors V_1 and V_2:

$$\begin{cases} V_1 = A_1 \circ N^k \circ K; \\ V_2 = A_2 \circ J^k \circ K. \end{cases}$$

2. Compute the hash-function value e (the first signature element) from the document M to which the vectors V_1 and V_2 are concatenated: $e = f_H(M, V_1, V_2)$.

3. Compute the second signature element $s = k + xe \bmod q$.

4. Compute the third signature element S as solution of the following vector equation: $Q^s \circ D \circ S = K$.

The last vector equation has a unique solution, since the product of the locally invertible vectors Q^s and D is a locally invertible vector, i. e., the main determinant Δ of the system of four linear equations corresponding to the last vector equation satisfies the condition $\Delta \neq 0$. The major contribution to the computational complexity of the fourth step in the last procedure is introduced by the exponentiation operation.

Thus, in the introduced signature scheme the digital signature is composed from three elements, two 256-bit integers e and s and one vector S. On the whole, the computational difficulty of the signature generation procedure can be estimated as three exponentiation operations in the FNAA used as algebraic support (roughly equal to three exponentiations mod$p$ for 1024-bit prime p, for example, in the Schnorr signature scheme [8]).

4.5 Signature verification procedure

Suppose one is to verify the signature (e, s, S) to the document M, using the public key $(Y_1, Z_1, T_1;\ \ Y_2, Z_2, T_2)$. The signature verification procedure is as follows:

1. Using the public key, compute the vectors V_1' and V_2':

$$\begin{cases} V_1' = Y_1^{-e} \circ T_1 \circ Z_1^s \circ S; \\ V_2' = Y_2^{-e} \circ T_2 \circ Z_2^s \circ S. \end{cases}$$

2. Compute the hash-function value e' from the document M to which the vectors V_1' and V_2' are concatenated: $e' = f_H\,(M, V_1', V_2')$.

3. Compute the value Δ_S (see formula (7)) corresponding to the locally invertible vector $S = (s_0, s_1, s_2, s_3)$.

4. If $e' = e$ and $\Delta_S \neq 0$, then the signature is genuine. Otherwise the signature is rejected as the false one.

4.6 Correctness proof

Correctness proof of the signature scheme consists in proving that the signature (e, s, S) computed correctly will pass the verification proce-

dure as genuine signature. Taking into account the mutual commutativity of the ψ-map operation with the exponentiation operation, for the vectors V_1' and V_2' computed at the first step of the signature verification procedure, we have the following:

$$\begin{aligned}
V_1' &= Y_1^{-e} \circ T_1 \circ Z_1^s \circ S = \\
&= (A_1 \circ N^x \circ B_1)^{-e} \circ T_1 \circ (C \circ (N \circ Q) \circ D)^s \circ S = \\
&= A_1 \circ N^{-ex} \circ B_1 \circ A_1 \circ D \circ L \circ C \circ (N^s \circ Q^s) \circ D \circ S = \\
&= A_1 \circ N^{-ex} \circ L_1 \circ D \circ C \circ N^s \circ D \circ C \circ Q^s \circ D \circ S = \\
&= A_1 \circ N^{-ex} \circ L_0 \circ N^{k+ex} \circ L_0 \circ Q^s \circ D \circ S = \\
&= A_1 \circ N^{-ex+k+ex} \circ K = A_1 \circ N^k \circ K = V_1; \\
V_2' &= Y_2^{-e} \circ T_2 \circ Z_2^s \circ S = \\
&= (A_2 \circ J^x \circ B_2)^{-e} \circ T_2 \circ (C \circ (J \circ Q) \circ D)^s \circ S = \\
&= A_2 \circ J^{-ex} \circ B_2 \circ A_2 \circ D \circ L' \circ C \circ (J^s \circ Q^s) \circ D \circ S = \\
&= A_2 \circ J^{-ex} \circ L_2 \circ D \circ C \circ J^s \circ D \circ C \circ Q^s \circ D \circ S = \\
&= A_2 \circ J^{-ex} \circ L_0 \circ J^{k+ex} \circ L_0 \circ Q^s \circ D \circ S = \\
&= A_2 \circ N^{-ex+k+ex} \circ K = A_2 \circ N^k \circ K = V_2.
\end{aligned}$$

Since $V_1' = V_1$ and $V_2' = V_2$, the equality $e' = e$ holds true. Besides, the correctly computed signature element S is a locally invertible vector, therefore, the inequality $\Delta_S \neq 0$ holds true.

5 Alternative design

When using the 8-dimensional FNAA with global two-sided unit as algebraic support, one can propose the following signature scheme with doubled verification equation, in which automorphism-map operations $\alpha_V(X) = V \circ X \circ V^{-1}$ (mutually commutative with the exponentiation operation) are used as masking ones.

Computation of the public key is performed as follows:

1. Generate at random the invertible vector N of the order q and integer $\beta \in GF(p)$ of the order q and compute the vectors $Q = \beta N$

and $J = N \circ Q^z$, where z is a random integer ($z < q$).

2. Generate at random the integer $x < q$ and the invertible vectors A_1 and A_2. Then compute the public-key elements $Y_1 = A_1 \circ N^x \circ A_1^{-1}$ and $Y_2 = A_2 \circ J^x \circ A_2^{-1}$.

3. Generate at random the integer $u < q$ and the invertible vector B_1. Then compute the vector $B_2 = B_1 \circ Q^u$ and the public-key elements $Z_1 = B_1 \circ N \circ Q \circ B_1^{-1}$, $T_1 = A_1 \circ B_1^{-1}$, $Z_2 = B_2 \circ J \circ Q \circ B_2^{-1}$, and $T_2 = A_2 \circ B_2^{-1}$.

The signature generation is performed as follows:

1. Generate at random the integer $k < q$ and the invertible vector K. Then compute $V_1 = A_1 \circ N^k \circ K$ and $V_2 = A_2 \circ J^k \circ Q^{-u} \circ K$.

2. Using a specified hash function f_H, compute the first signature element e: $e = f_H(M, V_1, V_2)$, where M is a document to be signed.

3. Compute the second signature element s: $s = k + ex \bmod q$.

4. Compute the third signature element $S = B_1 \circ Q^{-s} \circ K$.

The signature verification is performed as follows:

1. Using the signature (e, s, S) and the public key $(Y_1, Z_1, T_1;\ \ Y_2, Z_2, T_2)$, compute the vectors V_1' and V_2': $V_1' = Y_1^{-e} \circ T_1 \circ Z_1^s \circ S;\ \ V_2' = Y_2^{-e} \circ T_2 \circ Z_2^s \circ S$.

2. Compute the hash-function value $e' = f_H(M, V_1', V_2')$.

3. If $e' = e$ and S is an invertible vector, then the signature is genuine. Otherwise the signature is rejected.

Correctness proof of the signature scheme:

$$
\begin{aligned}
V_1' &= Y_1^{-e} \circ T_1 \circ Z_1^s \circ S = \\
&= \left(A_1 \circ N^x \circ A_1^{-1}\right)^{-e} \circ A_1 \circ B_1^{-1} \circ \left(B_1 \circ (N \circ Q) \circ B_1^{-1}\right)^s \circ S = \\
&= A_1 \circ N^{-ex} \circ N^s \circ Q^s \circ B_1^{-1} \circ S = \\
&= A_1 \circ N^{-ex+k+ex} \circ Q^s \circ B_1^{-1} \circ B_1 \circ Q^{-s} \circ K = \\
&= A_1 \circ N^k \circ K = V_1;
\end{aligned}
$$

$$
\begin{aligned}
V_2' &= Y_2^{-e} \circ T_2 \circ Z_2^s \circ S = \\
&= \left(A_2 \circ J^x \circ A_2^{-1}\right)^{-e} \circ A_2 \circ B_2^{-1} \circ \left(B_2 \circ (J \circ Q) \circ B_2^{-1}\right)^s \circ S = \\
&= A_2 \circ J^{-ex} \circ J^s \circ Q^s \circ B_2^{-1} \circ S = \\
&= A_2 \circ J^{-ex+k+ex} \circ Q^s \circ Q^{-u} \circ B_1^{-1} \circ B_1 \circ Q^{-s} \circ K = \\
&= A_2 \circ J^k \circ Q^{-u} \circ K = V_2.
\end{aligned}
$$

6 Discussion

In the known signature scheme [6] based on the computational difficulty of the HDLP the public key (Y, Z, T) is formed as a homomophism mapping of some elements N and N^x belonging to the same hidden cyclic group: $Z = \psi'(N)$ and $Y = \psi''(N^x)$, where ψ' and ψ'' are different homomorphism-map operations. Therefore, using these two values and the fitting element T of the public key, it is possible to construct a periodic function $f(i,j) = Y^i \circ T \circ Z^j$ containing a period that is determined by the value of the discrete logarithm x. Indeed, the condition $Y^{i-1} \circ T \circ Z^j = Y^i \circ Z \circ T^{j+x}$ holds true. In this case the assumed resistance to quantum attacks is justified by the fact that the values taken on by the function $f(i,j)$ lie in many different cyclic groups contained in the algebra.

To provide an advance justification of the HDLP-based signature scheme as candidates for post-quantum ones, a method for eliminating the periodicity with the period length depending on the value x is used. The method consists in using the commutative hidden group that is generated by the generator system $< N, Q >$ in which each of the vectors has order equal to the prime value q and computing the value Z as the vector $Z = \psi_0(N \circ Q)$. Multiplying by the vector Q destroys the periodicity associated with the value of the discrete logarithm. Indeed, the vector Q can not be represented as a power of the vector N, since these two vectors lie in different cyclic groups, so the construction of a periodic function with a period length other than the prime q, with the use of public key elements, seems computationally intractable.

However, the use of the product $N \circ Q$ as the preimage of the vector

Table 4. Comparison with the signature schemes Falcon-512, Dilithium-1024x768, and RSA-2048.

Signature scheme	signature size, bytes	publi-key size, bytes	sign. gener. rate, arb. un.	sign. verific. rate, arb. un.
Section 4	192	768	70	50
Section 5	320	1536	17	12
Falcon	657	897	50	25
Dilithium	2044	1184	15	10
RSA-2048	256	256	10	> 50

Z leads to the fact that the multiplier Q contributes also to the result of calculating the right part of the verification equation, which depends on the computed second signature element s. To compensate for this contribution, the calculation of the third element of the signature in the form of an invertible vector S is used. In order to prevent the possibility of signature forgery using the element S as a fitting parameter, the proposed signature scheme uses a doubled verification equation.

Table 4 presents a rough comparison of the proposed two signatures schemes (note different algebras used to implement these schemes) with the candidates for post-quantum signatures Falcon-512 and Dilithium-1024x768 [3].

In comparison with the known HDLP-based signature algorithms, a certain disadvantage of the proposed new signature scheme is the increased size of the signature and about two times higher computational complexity of the signature generation and verification algorithms. However, this disadvantage is quite acceptable in the light of ensuring the implementation of an enhanced criterion aimed at ensuring resistance to potential quantum attacks.

In the method [15] providing formal security proof of the Schnorr signature scheme [8] it is considered a forger that can compute the fitting signature element s equally well for different hash functions f_H and f_H^*. That model of the reductionist security proof is well applicable

to the HDLP-based signature schemes described in papers [6], [7], [14]. However, the said model is not applicable to the proposed signature schemes. In this connection, one can propose a topic for future development of the proposed design approach, i. e., development of the provably secure signature schemes satisfying the advanced criterion of postquantum resistance.

On the whole, it seems the post-quantum security estimate of the introduced two signature algorithms and known HDLP-based signatures is mainly connected with finding algebraic methods for reducing the HDLP to the ordinary HDLP in a finite field $GF(p^h)$ for some fixed value $h \geq 1$. This item represents a topic of individual study.

7 Conclusion

In this paper, a new design of the HDLP-based signature schemes is proposed, which is characterized in using the hidden commutative group having 2-dimensional cyclicity. Thanks to the latter, it is possible to specify the calculation of the corresponding elements Z_1 and Y_1 (Z_2 and Y_2, respectively) of the public key in such a way that, when constructing periodic functions using these two elements, we obtain a period length value equal to the prime order of the elements of the hidden commutative group. Method which provides masking the private value $x < q$ consists in calculation of elements Z_1 and Y_1 as homomorphic (or as automorphic) images of vectors belonging to different cyclic groups, available in the hidden group with the 2-dimensional cyclicity.

One of directions of the further development of the HDLP-based signature schemes is connected with finding new algebras providing possibility to set the hidden commutative groups with 2-dimensional and 3-dimensional cyclic structure.

Acknowledgement. The authors thank anonymous Referee for valuable remarks.

This work was supported by the budget theme No. 0060-2019-010.

References

[1] *Post-Quantum Cryptography. Proceedings of the 10th International Conference, PQCrypto 2019, Chongqing, China, May 8–10, 2019*, (Lecture Notes in Computer Science, vol. 11505), 2019, 420 p.

[2] *Post-Quantum Cryptography. 9th International Conference, PQCrypto 2018 Proceedings, Fort Lauderdale, FL, USA, April 9-11, 2018*, (Lecture Notes in Computer Science, vol. 10786), 2018, 530 p.

[3] Derek Zimmer. *NIST Round 2 and Post-Quantum Cryptography – The New Digital Signature Algorithms*. (2019) [Online]. Available: https://www.privateinternetaccess.com/blog/2019/02/nist-round-2-and-post-quantum-cryptography-the-new-digital-signature-algorithms/

[4] D.N. Moldovyan, "Non-Commutative Finite Groups as Primitive of Public-Key Cryptoschemes," *Quasigroups and Related Systems*, vol. 18, no. 2, pp. 165–176, 2010.

[5] A. S. Kuzmin, V. T. Markov, A. A. Mikhalev, A. V. Mikhalev, and A. A. Nechaev, "Cryptographic Algorithms on Groups and Algebras," *Journal of Mathematical Sciences*, vol. 223, no. 5, pp. 629–641, 2017.

[6] A. A. Moldovyan and N. A. Moldovyan, "Post-quantum signature algorithms based on the hidden discrete logarithm problem," *Computer Science Journal of Moldova*, vol. 26, no. 3(78), pp. 301–313, 2018.

[7] N. A. Moldovyan and A. A. Moldovyan, "Finite Non-commutative Associative Algebras as carriers of Hidden Discrete Logarithm Problem," *Bulletin of the South Ural State University. Ser. Mathematical Modelling, Programming & Computer Software (Bulletin SUSU MMCS)*, vol. 12, no. 1, pp. 66–81, 2019.

[8] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptology*, vol. 4, pp. 161–174, 1991.

[9] *Information technology – Security techniques – Digital Signatures with appendix – Part 3: Discrete logarithm based mechanisms*, International Standard ISO/IEC 14888-3:2006(E), 2006.

[10] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on quantum computer," *SIAM Journal of Computing*, vol. 26, pp. 1484–1509, 1997.

[11] N. A. Moldovyan, "Finite Non-commutative Associative Algebras for Setting the Hidden Discrete Logarithm Problem and Post-quantum Cryptoschemes on Its Base," *Buletinul Academiei de Stiinte a Republicii Moldova. Matematica*, no. 1(89), pp. 71–78, 2019.

[12] N.A. Moldovyan, P.A. Moldovyanu, "New primitives for digital signature algorithms," *Quasigroups and Related Systems*, vol. 17, no. 2, pp. 271–282, 2009.

[13] N.A. Moldovyan, "Fast signatures based on non-cyclic finite groups," *Quasigroups and Related Systems*, vol. 18, no. 1, pp. 83–94, 2010.

[14] D. N. Moldovyan, "New Form of the Hidden Logarithm Problem and Its Algebraic Support," *Buletinul Academiei de Stiinte a Republicii Moldova. Matematica*, no. 1(92), pp. XX–XX, 2020, to be published.

[15] D. Pointcheval, J. Stern, "Security Arguments for Digital Signatures and Blind Signatures," *Journal of Cryptology*, vol. 13, pp. 361–396, 2000.

D. N. Moldovyan, A. A. Moldovyan, N. A. Moldovyan Received December 11, 2019
Revised March 07, 2020

St. Petersburg Institute for Informatics and Automation of
Russian Academy of Sciences
14 Liniya, 39, St.Petersburg, 199178
Russia
E–mail: nmold@mail.ru

Computer Science Journal of Moldova, vol.**28**, no.1(82), 2020

Community Detection Based on Node Similarity without thresholds

Makhlouf Benazi Chaabane Lamiche

Abstract

To identify communities in social networks represented by a graph, we simply need to detect the edges that connect vertices of different communities and remove them, but the problem is what measure has to be used to identify these edges? and, how we use it? To tackle this problem, this paper proposes an efficient algorithm based on node similarity. This algorithm neither needs a predefined number of communities nor threshold to determine which edges to be deleted. The algorithm tries to add new edges for the most similar nodes to strengthen intra-community links and remove edges between the least similar nodes to weaken links between communities. In order to prove its efficiency, the algorithm was evaluated with synthetic and real-world networks.

Keywords: Social network, Community detection, node similarity, modularity, GN algorithm.

MSC 2010: 91C20.

1 Introduction

Today, with the emergence of new technologies of communication and the Internet especially web 2.0, social networks have grown exponentially in size and complexity. In order to understand the structure of these networks, analyze its characteristics and extract useful information and knowledge, several fields of research in social network analysis

have appeared. Some focus on the identification of the most influential individual (leadership), others on missing links, and the third – on network dynamics. In this paper we will focus on community detection.

The concept of community can be seen as a group of densely interconnected nodes compared to other nodes [1]. To detect the communities that constitute a network, researchers working in this field have proposed several approaches, see for instance [2], [3], recent surveys on the subject. These methods can roughly be grouped as methods emerging from graph theory known as partitioning algorithm such as spectral bisection algorithm [4], and methods from sociology known as hierarchical clustering algorithms [5], [6]. Researchers working in the field of sociology noticed that individuals belonging to the same community share some similarities, such as gender, age, common interests or professional activity etc. After measuring the similarity matrix, they incorporate the two nodes with the highest similarity together iteratively (agglomerative) such as Newman Fast Greedy FN [7] and the Concor algorithm of [8] or they iteratively remove the edge with the lowest similarity (divisive) such as Radicchi [9], Spectral [10] and Girvan-Newman (GN) algorithm [11]. We find also approximation algorithms which seek to maximize or minimize the value of a given quality function. Certainly, the most popular is the modularity [6]; such is in fast Newman algorithm (FN) [7], Genetic Algorithm [12], Simulated annealing [13], [14], PSO [15].

In this paper we propose a new algorithm for communities detection based on node similarity, where we use four different similarity measures to add new edges so that communities can get bit-by-bit closer to what we call a clique in graph theory, and to identify edges that will be removed so that communities can break away from each other.

The rest of this paper is organized as follows. We firstly present some basic concepts, so graph, similarity and Modularity are briefly introduced in Section 2. Then some research related to our algorithm is considered in Section 3. In Section 4 our approach is presented in detail. In order to show the effectiveness of our approach, in Section 5 we test our algorithm on different artificial and real-world networks with four different similarity measures, and make comparisons with Girvan-

Newman (GN) algorithm [11] and Fast Newman (FN) algorithm [7]. Finally, we conclude and present some perspectives in Section 6.

2 Preliminary Knowledge

2.1 Graph

We often have recourse to graphs to analyze a social network in order to identify local or global patterns, to locate influential entities, or to examine network dynamics. A graph G (undirected and unweighted) is an ordered pair $G(V, E)$, where V is the set of nodes or vertices that represent individuals, and $E \subset V \times V$ is the set of edges or links that represent interactions between individuals (friendship, collaboration, love etc.). The degree k_i of a vertex i is the number of edges connecting i to the rest of the graph. If A is the adjacency matrix of G, then $k_i = \sum_j A_{i,j}$.

2.2 Graph partitioning

The goal of graph partitioning is to divide G into k disjoint sub-graphs $G_i = (V_i, E_i)$, in which $\forall\ i \neq j\ :\ V_i \cap V_j = \Phi\ and\ \bigcup_1^k V_i = V$.

2.3 Modularity

There are an exponential number of diverse alternative partitions. Enumerating all these partitions is an NP-Complete problem [16]. Moreover, not all partitions of a graph are equally good. In order to choose the best partition Newman and Girvan [11] introduced a metric that computes the difference between the fraction of edges for a given partition of the original graph and a random graph having a similar degree distribution as the original. This metric is known as Modularity.

$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{i,j} - \frac{k_i k_j}{2m} \right] \delta(C_i, C_j), \tag{1}$$

where m is the number of edges. A denotes the adjacency matrix, where $A(i, j)$ is equal to 1 if there is an edge between nodes i and j, and 0

otherwise. k_i and k_j are the degrees of vertices i and j respectively, C_i and C_j are the communities of the vertices i and j respectively, and $\delta(C_i, C_j)$ is equal to 1 if vertices i and j belong to the same community, and 0 otherwise.

2.4 Structural similarity

A community represents a set of nodes that are more similar to each other, but dissimilar from the rest of the network [6]. But how similar are two nodes? To answer this question several methods have been proposed in the literature. Some of these methods measure the distance between two nodes, others – the local paths and the third count how many neighbors two nodes have in common. Here are some of those measures that have been tested with our algorithm.

In what follows $\Gamma(x)$ represents the set of neighbors of node x, $|\Gamma(x)|$ denotes its degree, n is the number of nodes, and $A(x, i)$ denotes an element of the adjacency matrix.
Jaccard Index: [17]

$$S_{xy} = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}, \tag{2}$$

Cosine Index: [18]

$$S_{xy} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\sqrt{|\Gamma(x)| \times |\Gamma(y)|}}, \tag{3}$$

Pearson Coefficient: [19]

$$S_{xy} = \frac{\sum ((A(x,i) - \bar{x}) * (A(y,i) - \bar{y}))}{\sqrt{\sum (A(x,i) - \bar{x})^2} * \sqrt{\sum (A(y,i) - \bar{y})^2}}, \tag{4}$$

where: $\bar{x} = \frac{|\Gamma(x)|}{n}$.
Hub Promoted Index: [20]

$$S_{xy} = \frac{|\Gamma(x) \cap \Gamma(y)|}{\min(|\Gamma(x)|, |\Gamma(x)|)}. \tag{5}$$

3 Related Work

Our approach is closely related to Girvan Newman algorithm [11], which can be expressed as follows:

1. Calculate edge betweenness for every edge in the graph.
2. Remove the edge with highest edge betweenness.
3. Calculate edge betweenness for remaining edges.
4. Repeat steps 2-4 until all edges are removed.

Despite its simplicity, Girvan Newman Algorithm has several fundamental limitations. First, it uses betweenness centrality to identify which edge to delete. Betweenness centrality is a global measure that identifies edges with the highest number of the shortest paths which pass through them. The calculation of edge betweenness has time complexity $O(MN)$; therefore, the algorithm's time complexity is $O(M^2N)$. Here M is the number of edges and N is the number of nodes, unlike our approach, where we use a local similarity measure based on neighborhood. Removing a single edge at each iteration is another problem which will be remedied in our approach by removing multiple edges at every iteration. In order to increase the accuracy of our algorithm new edges are added at each iteration to strengthen communities. Finally, there is no implicit stopping criterion, either we give the number of communities or the algorithm will continue until all edges are removed, in the latter case the algorithm returns partition with the highest modularity.

Our approach is also related to algorithms that use node similarity like in [21], where the authors proposed an algorithm for detecting community. In their paper they introduce the similarity threshold ϵ which takes different values for different datasets. For example, with Zachary's karate club network the authors put $\epsilon = 0.9$ and with college football network they put $\epsilon = 0.75$. But with this strategy how to determine the value of this threshold? We have observed that this strategy is frequently used in the literature, as in [22] and [23], unlike our approach, where we do not use any parameter or threshold.

We find also approaches like DBSCAN [24], DENGRAPH [25], SCAN [26], DEEN [27], and SMP [28]. However not all these algorithms are free-parameter, they always depend overly on manually choosing thresholds of measure that have been used, a minimum cluster size or cluster number, which can be difficult to determine.

4 The Algorithm

We have developed an effective algorithm based on nodes similarity. The main idea in our algorithm is: 1) compute the similarity matrix at every iteration; 2) try to remove edges that have the least similarity on both sides; and 3) add new edges for nodes that have the most similarity on both sides for every node in the graph. In other terms, to delete a link (i, j), the node least similar to node i must be j and the node least similar to node j must be i, and to add a new link (i, j), the node most similar to node i must be j and the node most similar to node j must be i.

We can describe our algorithm as follows:

1. Sort nodes in descending order using node degree.
2. Calculate similarity matrix S (eq. 2, 3, 4 or 5).
3. Remove all edges that fit both of the following conditions:
   - For every two nodes i, j that have common edge, the node least similar to node i must be j, and the node least similar to node j is i.
   - The deletion does not generate an orphan node.
4. For every two nodes i, j add new edge if the node most similar to node i is j, and the node most similar to node j is i.
5. Repeat steps 2-4 until the set of deleted links is the same added or a maximum number of iterations is reached.
6. Compute community vectors using Hopcroft & Tarjan algorithm [29].

7. Return Community vector with the best modularity calculated using formula (1).

5 Experimental Evaluation

In order to measure the accuracy of our algorithm vis-à-vis ground truth and compare it to other algorithms we use NMI (Normalized Mutual Information) [30] as measure of partition (i.e., clustering) similarity. It takes values in the interval [0, 1].

The NMI of two partitions A and B of a graph is given as follows:

$$I(A,B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} M_{ij}\ log\left(\frac{M_{ij}n}{M_{i.}M_{.j}}\right)}{\sum_{i=1}^{C_A} M_{i.}\ log\left(\frac{M_{i.}}{n}\right) + \sum_{j=1}^{C_B} M_{.j}\ log\left(\frac{M_{.j}}{n}\right)}, \qquad (6)$$

where C_A and C_B denote the numbers of communities in partitions A and B respectively. The notation M_{ij} denotes the element of matrix (M) $C_B \times C_B$, representing the number of nodes in the i^{th} community of A that appear in the j^{th} community of B. The sum over row i of matrix M is denoted by $M_{i.}$, and that over column j – by $M_{.j}$; and n is the number of nodes. I(A, B) = 1, if A=B. I(A, B) = 0, if A and B are completely different.

In this section, we applied our algorithm to synthetic and real-world networks.

5.1 Synthetic Benchmark Networks

Girvan and Newman [31] proposed a benchmark network with four communities, with every community containing 32 vertices (for a total number of vertices $n = 128$), and fixed the average total degree of each node to 16: $k = Z_{in} + Z_{out} = 16$, where Z_{in} is the number of edges connecting a node with the others in its own community, and Z_{out} is the number of edges connecting a node with the rest of the network. We vary Z_{out} from 0 to 8. The values of the NMI measures are shown in Figure 1. As it can be seen from Figure 1, the results given by Pearson Coefficient and Hub Promoted Index are better than the other indices

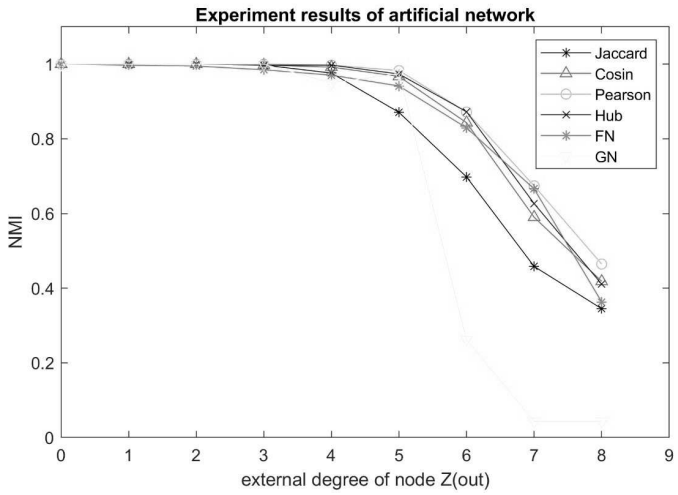

Figure 1. Experiment results of artificial network

and better than Fast Newman (FN) algorithm whatever the value of Z_{out}. Jaccard index obtains the worst performances. We observe also that GN fails to find the true partition when $Z_{out} > 5$.

5.2 Real-world Networks

Just like in other algorithms for detection community, we selected the following 4 well known real-world networks to further verify the performance of our algorithm: 1) the Zachary's Karate Club network; 2) Dolphins network; 3) the American College Football network; 4) Krebs' political books. The properties of these four real-world networks are listed in Table 1.

To verify the performance of our algorithm, we compared it with two algorithms considered as reference in the field of communities detection, the GN algorithm and FN algorithm. GN or Girvan and Newman [11] algorithm calculates the betweenness centrality of all edges (number of the shortest paths passing through an edge) and removes the edge with the biggest betweenness recursively. The second algo-

Networks	Ref.	C	N	M	Q
Zachary	[32]	2	34	78	0.37
Dolphins	[33]	2	62	159	0.38
Football	[31]	12	115	613	0.55
Polbooks	[10]	3	105	441	0.41

Table 1. Properties of real-world networks employed in the tests. C – number of communities, N – number of nodes, M – number of edges, Q – original modularity.

rithm is FN (Fast Newman) [7], a greedy algorithm that tries to maximize the modularity.

Since FN is a stochastic optimization algorithm, we perform the experiments 10 times on each network. The average value of Q and NMI are calculated. The results are shown in Table 2. The best results are shown in **bold**.

It can be seen from Table 2 and Figure 3 that our algorithm can correctly detect the community structure on Zachary's karate club (see Figure 2, where solid lines indicate edges that have been added and dashed lines indicate edges that have been deleted), except when we use Pearson as similarity measure although it can detect it from the first iteration. But because we have used the modularity maximization as criterion to choose the best partition, it will select other partition that has the best modularity. The same phenomenon is seen with Dolphins network, when we use hub as similarity measure, where the correct partition is found in the third iteration, but the maximum modularity is found in the 5_{th} iteration. Figure 4 shows the evolution of NMI and modularity for Dolphins network. On Football network, our approach performs well especially when we use Jaccard, Cosine or Hub as similarity measure, until 0.9306. Regarding the Polbooks network, the NMI reached its maximum 0.5836 when using Cosine index as similarity measure.

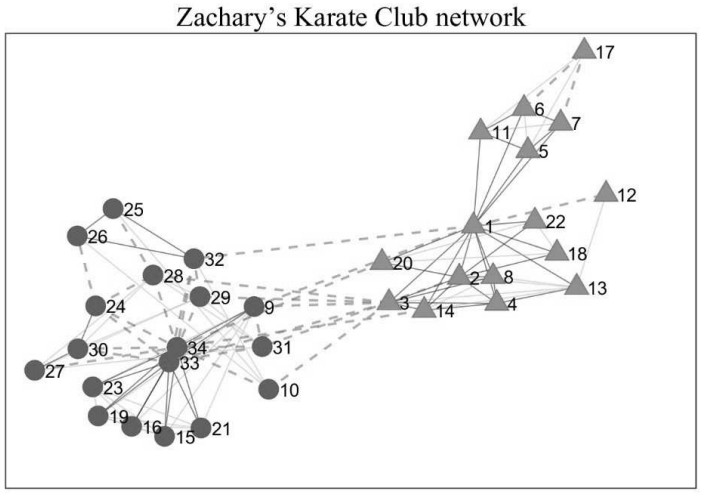

Figure 2. Result of Karate Club network obtained by our algorithm.

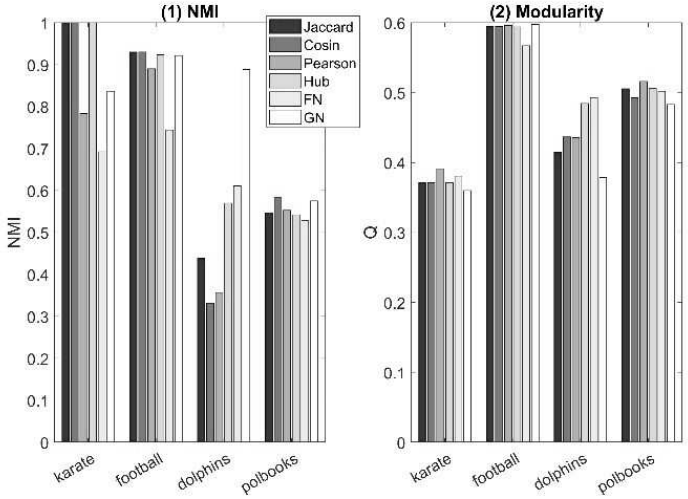

Figure 3. NMI (1) and modularity (2) of four real-world networks obtained by our algorithm and fast Newman.

algorithm	**Zachary**		**Dolphins**	
	Q	**NMI**	**Q**	**NMI**
Fast Newman	0.3807	0.6925	**0.4942**	**0.5919**
Girvan Newman	0.3600	0.8365	0.3787	0.8888
Jaccard	0.3715	**1**	0.4160	0.4200
Cosine	0.3715	**1**	0.4338	0.3260
Pearson	**0.3949**	0.7534	0.4417	0.3573
Hub	0.3715	**1**	0.4849	0.5719
algorithm	**Football**		**Polbooks**	
	Q	**NMI**	**Q**	**NMI**
Fast Newman	0.5698	0.7460	0.5019	0.5292
Girvan Newman	0.5976	0.9218	0.4831	0.5754
Jaccard	0.5946	**0.9306**	0.5057	0.5471
Cosine	0.5946	**0.9306**	0.4926	**0.5836**
Pearson	**0.6012**	0.9019	**0.5176**	0.5575
Hub	0.6007	0.9195	0.5040	0.5413

Table 2. NMI and modularity of four real-world networks obtained by our algorithm and fast Newman.

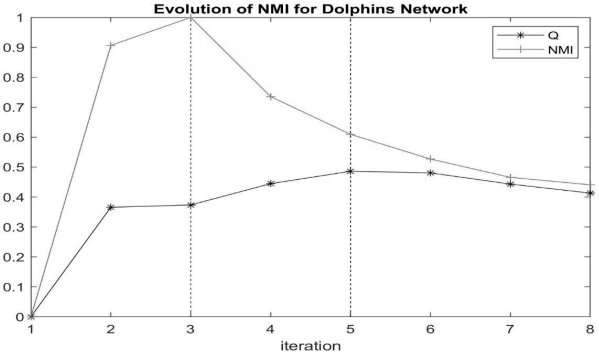

Figure 4. Evolution of NMI and modularity for Dolphins network.

6 Conclusions

Community detection is a very hard problem that has not yet been satisfactorily solved despite many methods have been proposed. In this paper, we have proposed a new algorithm to find high quality communities in social network based on node similarity that, we think, performs well. Experimental results show that the algorithm achieves better performance compared to FN and GN algorithms, especially on real world networks. Finally, it is worth to mention that our algorithm can be used with several similarity measures, which makes it more convenient and more flexible in real application, especially if we know that each application domain has its own measure of similarity. In future work, we will focus on optimizing the complexity of our algorithm to be able to apply it on larger networks.

References

[1] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[2] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Physics reports*, vol. 659, pp. 1–44, 2016.

[3] Q. Cai, L. Ma, M. Gong, and D. Tian, "A survey on network community detection based on evolutionary computation," *International Journal of Bio-Inspired Computation*, vol. 8, no. 2, pp. 84–98, 2016.

[4] E. R. Barnes, "An algorithm for partitioning the nodes of a graph," *SIAM Journal on Algebraic Discrete Methods*, vol. 3, no. 4, pp. 541–550, 1982.

[5] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.

[6] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[7] M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review E*, vol. 69, no. 6, p. 066133, 2004.

[8] R. L. Breiger, S. A. Boorman, and P. Arabie, "An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling," *Journal of mathematical psychology*, vol. 12, no. 3, pp. 328–383, 1975.

[9] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proceedings of the national academy of sciences*, vol. 101, no. 9, pp. 2658–2663, 2004.

[10] M. E. Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[11] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[12] C. Pizzuti, "Ga-net: A genetic algorithm for community detection in social networks," in *International conference on parallel problem solving from nature*. Springer, 2008, pp. 1081–1090.

[13] R. Guimera and L. A. N. Amaral, "Functional cartography of complex metabolic networks," *nature*, vol. 433, no. 7028, p. 895, 2005.

[14] Z. Masdarolomoor, R. Azmi, S. Aliakbary, and N. Riahi, "Finding community structure in complex networks using parallel approach," in *2011 IFIP 9th International Conference on Embedded and Ubiquitous Computing*. IEEE, 2011, pp. 474–479.

[15] C. Cao, Q. Ni, and Y. Zhai, "A novel community detection method based on discrete particle swarm optimization algorithms in complex networks," in *2015 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2015, pp. 171–178.

[16] M. R. Garey, D. S. Johnson, and L. Stockmeyer, "Some simplified np-complete problems," in *Proceedings of the sixth annual ACM symposium on Theory of computing*. ACM, 1974, pp. 47–63.

[17] P. Jaccard, "Étude comparative de la distribution florale dans une portion des alpes et des jura," *Bull Soc Vaudoise Sci Nat*, vol. 37, pp. 547–579, 1901.

[18] G. Salton and M. J. McGill, *Introduction to modern information retrieval*. mcgraw-hill, 1983.

[19] K. Pearson, "Vii. note on regression and inheritance in the case of two parents," *proceedings of the royal society of London*, vol. 58, no. 347-352, pp. 240–242, 1895.

[20] E. Ravasz, A. Somera, D. Mongru, Z. Oltvai, and A. Barabasi, "Community structure in social and biological networks," *Science*, vol. 297, p. 1553, 2002.

[21] Y. Pan, D.-H. Li, J.-G. Liu, and J.-Z. Liang, "Detecting community structure in complex networks via node similarity," *Physica A: Statistical Mechanics and its Applications*, vol. 389, no. 14, pp. 2849–2857, 2010.

[22] T. Dang and E. Viennet, "Community detection based on structural and attribute similarities," in *International conference on digital society (icds)*, 2012, pp. 7–12.

[23] X. Wang, G. Liu, J. Li, and J. P. Nees, "Locating structural centers: A density-based clustering method for community detection," *PloS one*, vol. 12, no. 1, p. e0169355, 2017.

[24] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[25] T. Falkowski, A. Barth, and M. Spiliopoulou, "Dengraph: A density-based community detection algorithm," in *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE Computer Society, 2007, pp. 112–115.

[26] X. Xu, N. Yuruk, Z. Feng, and T. A. Schweiger, "Scan: a structural clustering algorithm for networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2007, pp. 824–833.

[27] P. Jancura, D. Mavroeidis, and E. Marchiori, "Deen: a simple and fast algorithm for network community detection," in *International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics*. Springer, 2011, pp. 150–163.

[28] K. Zhou, A. Martin, and Q. Pan, "A similarity-based community detection method with multiple prototype representation," *Physica A: Statistical Mechanics and its Applications*, vol. 438, pp. 519–531, 2015.

[29] J. Hopcroft and R. Tarjan, "Algorithm 447: efficient algorithms for graph manipulation," *Communications of the ACM*, vol. 16, no. 6, pp. 372–378, 1973.

[30] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, p. P09008, 2005.

[31] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[32] W. W. Zachary, "An information flow model for conflict and fission in small groups," *Journal of anthropological research*, vol. 33, no. 4, pp. 452–473, 1977.

[33] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations," *Behavioral Ecology and Sociobiology*, vol. 54, no. 4, pp. 396–405, 2003.

Makhlouf Benazi, Chaabane Lamiche

Received September 15, 2019
Revised January 21, 2020

Makhlouf Benazi
Department of computer science,
Faculty of mathematics and computer science,
Mohamed Boudiaf University of Msila, Msila, 28000, Algeria
E–mail: Makhlouf.benazi@univ-msila.dz

Chaabane Lamiche
Department of computer science,
Faculty of mathematics and computer science,
Mohamed Boudiaf University of Msila, Msila, 28000, Algeria
E–mail: Chaabane.lamiche@univ-msila.dz