

# Relaxation Time as Unique Characteristic for Networks Clustering

Andrii Snarskii <sup>1,2</sup>, Dmytro Lande <sup>1,2</sup> and Oleh Dmytrenko <sup>1,2</sup>

<sup>1</sup> National Technical University «Igor Sikorsky Kyiv Polytechnic Institute», 37, Prosp. Peremohy, Kyiv, 03056, Ukraine

<sup>2</sup> Institute for Information Recording of National Academy of Sciences of Ukraine, 2, Mykoly Shpaka Street, Kyiv, 03113, Ukraine

## Abstract

This paper researches new unique characteristics of networks – a network relaxation time and an individual node relaxation time, which characterize the stability of a complex network and, accordingly, each node separately to external perturbations. While researching of the complex networks, it is assumed that relaxation time is the number of iterative steps of the corresponding algorithm required to achieve the initial equilibrium numerical values of a certain characteristic after some external perturbation. In other words, the network relaxation time for each node characterize the resistance of a complex network and the individual node relaxation time characterize the resistance of each node to external perturbations, accordingly. In this work, to compute the relaxation time, the decelerated iterative HITS algorithm is used. It is shown, that these characteristics are unique numerical characteristic of network nodes, and they can be used to find the centroids of clusters and combine nodes into groups according to these characteristics – for complex networks clustering. The approbation of the presented characteristics of the relaxation time and the individual relaxation time was carried out on the example of clustering of random networks with clearly expressed clusters. In particular, a randomly generated matrix with dimension 30×30 and 3 clusters and a matrix with dimension 100×100 and 4 clusters were researched.

## Keywords

Complex Network, Network Relaxation Time, Individual Node Relaxation Time, HITS, PageRank, Clustering, Centroids of Clusters

## 1. Introduction

Complex networks are widespread in nature and technology. For example, networks such as the World Wide Web, peer-to-peer networks [1] and others are complex [2, 3]. They have non-trivial topological properties and, therefore, are of great interest to research.

Despite the fact that various networks, such as electrical, transport, and information networks, fall within the scope of the consideration of the theory of complex networks, the greatest contribution to the development of this theory is made by research of social networks [4], where binary relationships between people in a group can be represented in the form of a network, so these networks also play an important role. In these networks, each object is a node, and connections between nodes are an edge or link [5, 6].

An analysis of complex networks that includes the study of statistical and dynamical structural properties changes that characterise their behaviour, as well as the prediction of the evolution of such networks, are important directions [7].

Division of a set of network nodes (objects) into subsets (clusters), within which these objects are strongly connected, and the connection between individual clusters is relatively small, i.e. partitioning into clusters and, for example, detecting communities in social networks [8, 9] is an urgent task.

---

XXI International Scientific and Practical Conference "Information Technologies and Security" (ITS-2021), December 9, 2021, Kyiv, Ukraine  
EMAIL: asnarskii@gmail.com (A. Snarskii); dwlande@gmail.com (D. Lande); dmytrenko.o@gmail.com (O. Dmytrenko)  
ORCID: 0000-0002-4468-4542 (A. Snarskii); 0000-0003-3945-1178 (D. Lande); 0000-0001-8501-5313 (O. Dmytrenko)



© 2021 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

There are many methods of clustering graphs of complex networks [10, 11], which differ in the idea of combining similar nodes. Since different models use different algorithms, different clustering models are distinguished, in particular, such as connectivity models, centroid, statistical, group, neural models and others. For example, there are models based on connectivity. In these models, objects that are closer in space are more similar (related). And also, there are models based on finding the centroid. In these models, clusters are represented in the form of a central vector.

In this work, we propose to use a new characteristic for clustering complex networks – the relaxation time [12, 13].

## 2. Cluster network analysis

Cluster network analysis, as a rule, solves the problem of two-criteria optimization, namely:

1. Within each cluster  $K$ , elements (nodes) should be connected as much as possible, that is

$$\sum_{\substack{i \in K \\ j \in K}} a_{ij} \rightarrow \max. \quad (1)$$

Hereinafter,  $a_{ij}$  is some estimate of the relationship between the elements with indices  $i$  and  $j$ , which are included in the cluster  $K$ .  $a_{ij}$  can take non-negative values. This estimate can be calculated in various ways, for example, as the shortest path between nodes  $i$  and  $j$  [14].

2. The connectivity between any separate different clusters, for example,  $K_p$  and  $K_q$  should be minimal, that is

$$\sum_{\substack{i \in K_p \\ j \in K_q}} a_{ij} \rightarrow \min. \quad (2)$$

Often, in the general case, the sums for all indices are estimated ( $N$  is a number of clusters):

$$\sum_{p=1}^N \sum_{\substack{i \in K_p \\ j \in K_p}} a_{ij} \rightarrow \max, \quad \sum_{p=1}^N \sum_{\substack{q=1 \\ q \neq p}}^N \sum_{\substack{i \in K_p \\ j \in K_q}} a_{ij} \rightarrow \min. \quad (3)$$

By association, in our case, the first factor in the formula for determining the weight  $w_i$  corresponds to the first requirement, if it is of great importance, then it is connected by strong ties with a certain number of nodes (including from its own group). If the parameter  $x$  is of great importance, then the bonds are concentrated within their own group (cluster) – this corresponds to requirement 2.

We can assume that the individual nodes with the highest discriminant weight (the number chosen in advance) will constitute the centroids for clustering, or the basis for formation of clusters that will form the basis of the clusters.

There are numerous clustering algorithms such as K-means [15], LSA/LSI [4], most of which are used for networks. Recently, the algorithm based on the modularity property has been widely used due to the fact that it is built into the Gephi system [16].

By definition, modularity is equal to the proportion of the total number of edges that fall into a given cluster minus the predicted numbers of edges that would fall into the same groups if they were randomly distributed. The value of modularity lies in the interval  $[0, 1]$ . For a given partitioning of network nodes into clusters, modularity shows how many links in clusters in comparison to the links that are randomly distributed between all nodes without paying attention to clusters. There are various methods for calculating modularity. In the most commonly accepted version of the concept, edges are randomized in such a way that the degree of each node is preserved. The difference between the real number of edges between nodes  $i$  and  $j$  and the predicted number of edges between them is

$$a_{ij} - \frac{k_i k_j}{2m}, \quad (4)$$

where  $m$  is a number of edges in network,  $k_i$  and  $k_j$  are degrees of the nodes  $i$  and  $j$ , respectively.

The modularity is defined as the sum of all pairs

$$Q = \frac{1}{2m} \sum_{i,j} \left[ a_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j), \quad (5)$$

where  $\delta$  is Kronecker delta (shows whether nodes  $i$  and  $j$  are in the same cluster).

The matrix formulation of modularity is as follows. By definition, if the node belongs to the cluster then  $S_{ir}$  is equal to 1 and else it is equal to 0. Then

$$\delta(c_i, c_j) = \sum_r S_{ir} S_{jr}, \quad (6)$$

$$Q = \frac{1}{2m} \sum_{i,j} \sum_r \left[ a_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) = \frac{1}{2m} \text{Tr}(S^T B S), \quad (7)$$

where  $S$  is a matrix, having elements  $S_{ir}$ , and  $B$  is the so-called modularity matrix, which has elements (4).

The modularity of a network without clusters is equal 0 cause the sum of all columns and rows of the matrix that corresponds modularity is equal to 0.

Figure 1 shows a fragment of the interface of the Gephi system, which implements the calculation of the modularity function.

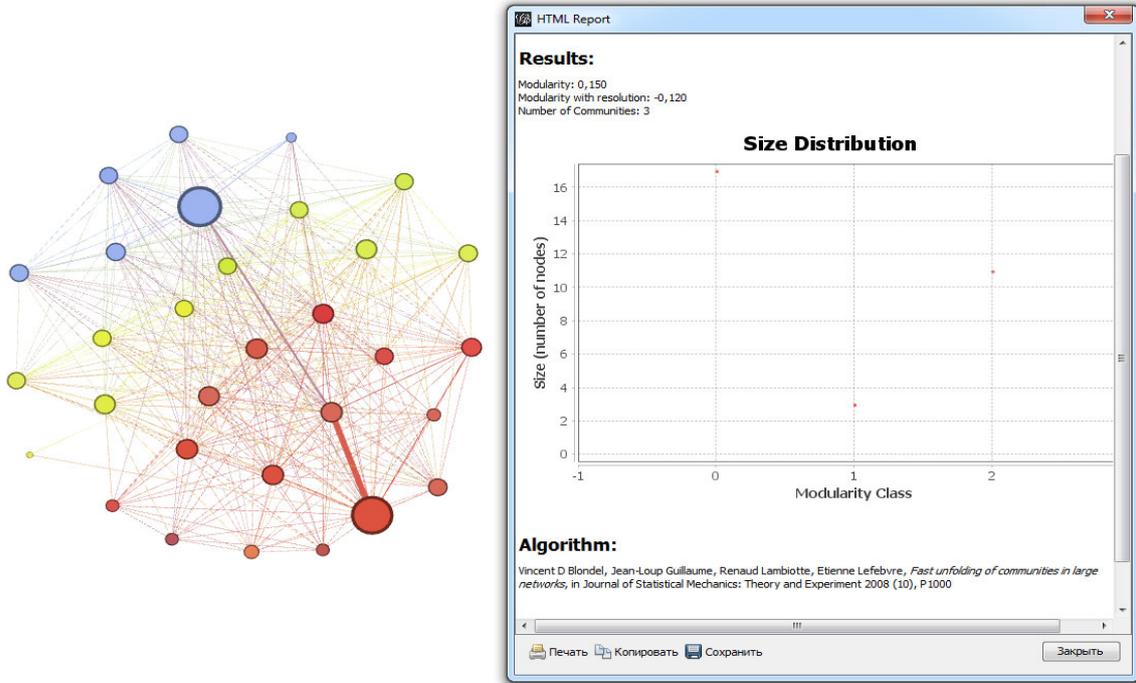


Figure 1: Fragment of the interface for calculating the modularity in the Gephi system

### 3. Physical meaning of relaxation time

In physics, relaxation is understood as the process of establishing thermodynamic (including stochastic) equilibrium in a physical system. It is assumed that the system consists of a large number of particles.

The system is characterized by a large number of parameters, and different sets of parameters move towards equilibrium at different speeds. The parameters characterizing the microstate of the system are the first to relax. And ultimately, we can talk about local equilibrium when the thermodynamic parameters relax, such as example, temperature, the concentration of particles or charges, etc.

For such phenomenological parameters responsible for transfer processes in physical systems, the Le Chatelier-Brown principle was formed [17]. If the physical quantity  $f$  characterizing the system

(temperature, charge density, etc.) has the value  $f_0$  i  $|f - f_0|/f_0 \ll 1$ , then in the first approximation  $f$  that go to equilibrium is given by the kinetic equation

$$\frac{df}{dt} = -\lambda(f - f_0), \quad (8)$$

The solution of this equation is

$$f(t) = f_0 + (f(0) - f_0)e^{-\frac{t}{\tau}} \quad (9)$$

where  $f(0) - f_0$  is an initial perturbation, and

$$\tau = \frac{1}{\lambda} \quad (10)$$

is a relaxation time.

So, for example, for an environment with conductivity  $\sigma$  and the relative permittivity  $\varepsilon$ , Maxwell's equation and Ohm's law  $\vec{j} = \sigma \vec{E}$  are can be written as

$$\text{div} \varepsilon \vec{E} = 4\pi\rho, \quad \frac{\partial \rho}{\partial t} + \text{div} \vec{j} = 0 \quad (11)$$

where  $\vec{E}$  is an electric field strength,  $\vec{j}$  is a charge density.

In the simplest case of a homogeneous environment

$$\frac{\partial \rho}{\partial t} + 4\pi \frac{\sigma}{\varepsilon} \rho = 0 \quad (12)$$

the next equation immediately follows

$$\rho(t) = \rho_0 e^{-\frac{t}{\tau}} \quad (13)$$

where the relaxation time is

$$\tau = \frac{\varepsilon}{4\pi\sigma} \quad (14)$$

i.e., the higher conductivity  $\sigma$  and the lower relative permittivity  $\varepsilon$ , the faster the heterogeneity of the charge distribution is resolved, i.e., the equilibrium is reached faster.

The concept of relaxation is used in the methods for approximate solution of systems of linear equations, in iterative methods, for example, in methods of coordinate relaxation, block and group relaxation, etc.

These examples can be described in the following terms. Let us consider a system of equations  $\vec{L}f = 0$ , the solution of which is the function  $f_0$ . To find solutions, some initial approximation  $f$  of the function  $f_0$  is chosen and such an iterative procedure is found. In explicit form it is represented as

$$f_{n+1} = \hat{T}f_n \quad (15)$$

or in implicit form as

$$F(f_{n+1}, f_n) = 0 \quad (16)$$

that is, there is such an operator  $\hat{T}$  that

$$\lim_{n \rightarrow \infty} f_n = f_0 \quad (17)$$

Here we assume an approach and terminology that allows us to consider the described methods in the only way.

Let us consider a system described by a set of parameters (numbers, functions)  $f$ . The equilibrium state of the system is given by equations (11-12)

$$\hat{L}f = 0 \quad (18)$$

is a solution for which  $f_0$  defines the equilibrium state of the system. Let's perturbing the system out of equilibrium by defining some set  $f_1 \neq f_0$ . By choosing some iterative scheme (15), we get the sequence

$$f_1, f_2, \dots, f_n \quad (19)$$

where  $n$  is considered as the discrete time.

In the usual case, for a descending sequence  $|f_2 - f_1|, |f_3 - f_2|, \dots$  let's define a value  $\Delta$ , such that if

$$|f_{n+1} - f_{n_c}| \leq \Delta f_1, f_2, \dots, f_n \quad (20)$$

that value  $n_c$ , which determines (20) will be considered (called) as the relaxation time  $\tau = n_c$ .

So, in the case of researching complex networks, the relaxation time is the number of iterative steps  $\tau$  of the corresponding algorithm, which are necessary to reach the initial equilibrium numerical values of a certain characteristic after its perturbation with some predetermined accuracy  $\Delta$  – the condition of convergence (usually  $\Delta=10^{-4}$ ).

When the recovery of the whole network is researched, the number of iteration steps required for the value of each node to recover (the whole network is recovered) is called the relaxation time of the network  $\tau^{(m)}$  after the perturbation of a certain  $m$ -th node or  $\max_k(\tau_k^{(m)})$  ( $k=1, \dots, N$ , where  $N$  is a number of nodes in network). And the number of iterations which are required to recover the particular node whose numerical value was perturbed, will be called the individual relaxation time indicator of the node  $\tau_m^{(m)}$  (or simply  $\tau_m$ ). That is, the first characteristic characterizes the node in terms of the stability of the whole network after an external perturbation, the last one characterizes the individual stability of the node after its perturbation.

## 4. Algorithm

For example, let's consider some complex network (a directed graph), for which *PageRank* [18] (*PR* of each node) can be found. PageRank is one of the algorithms for evaluating the importance and ranking of web pages by hyperlinks, it was created by Sergey Brin and Larry Page in 1996 at Stanford University. The principle of calculating the PageRank of nodes is based on the model of "random wandering" of the user according to the following algorithm: he opens a random node (web page) from which he goes by a randomly selected link. It then moves to another web page and activates a random link again, and so on, constantly going from page to page, never coming back. Sometimes, when, with some probability  $1 - \delta$ , he gets bored with this wandering, or the page does not have links to other pages, he goes to a random web page again – not by a link, but by manually typing in some URL. It is assumed that the probability that a user wandering the network will go to a certain web page is its rank. Obviously, a node's PageRank is higher the more other nodes link to it, and the more popular those nodes are.

Let's assume, there are  $n$  nodes  $\{d_1, d_2, \dots, d_n\}$  that refer to some node (web page  $A$ ), and  $C(A)$  is the total number of links from a node to other nodes. Some fixed value  $\delta$  is defined as the probability that the user, viewing any web page from the set  $D$ , will go to the node  $A$  by a link, and not by typing its URL explicitly.

Within the framework of the model, the probability of this user continuing to surf the web from  $N$  web pages without using links, by manually entering an address (URL) from a random page will be  $1 - \delta$  (alternative to following links). The *PageRank* index for a node is considered as the probability that the user will be at this node at some random moment in time:

$$PR(A) = \frac{1 - \delta}{N} + \delta \sum_{i=1}^n \frac{PR(d_i)}{C(d_i)}. \quad (21)$$

According to this formula, the node rank is calculated by a simple iterative algorithm.

In a complex network of  $N$  nodes, the *PR* value of each node  $PR_i^0, i = 1, 2, \dots, N$  determines the equilibrium state of this network. After selecting the value of  $PR_i$  for each node  $i$  that deviates from the value  $PR_i - PR_i^0 \neq 0$ , the iterative procedure for finding *PR* is used. The relaxation time of node  $j$  is the number of steps  $m$  for which

$$|PR_j^0 - PR_i| \leq \Delta \quad (22)$$

Note that there are different variants of setting initial values (deviations) for nodes. They can be the same, they can alternately set the deviations of individual nodes, leaving others at equilibrium values, etc.

In this work, as a characteristic whose numerical value is perturbed, a characteristic corresponding to the iterative HITS (Hyperlink Induced Topic Search) rank algorithm was used. This algorithm was proposed and developed in 1998 by J. Kleinberg [19] to select the best "authors" (primary sources that refer to other documents) and "hub" (documents that refer to these primary sources) from an array of documents. For each document  $j$ , its authority  $a(j)$  (Authority) and hub  $h(j)$  are calculated according to the formulas:

$$a(j) = \sum_{i \rightarrow j} h(i), \quad h(j) = \sum_{j \rightarrow i} a(i), \quad (23)$$

Let introduce the relaxation time of the  $k$ -th node in the complex system –  $\tau_k$ . At the initial stage, a set of values of nodes  $s_k^0$  ( $\vec{s}^0$  is a vector form) define as the equilibrium state. This state is determined in accordance with the certain rule, for example, – the HITS or PageRank iterative algorithm, or any other.

Calculation of  $\vec{s}^0$  in accordance with the selected rule (the iterative algorithm) can always be rewritten in iterative form:

$$\vec{s}(n+1) = \vec{s}(n) + \hat{L}\vec{s}(n), \quad n = 0, 1, \dots \quad (24)$$

where the numbers of nodes correspond to the vector elements numbers of  $\vec{s}$ ;  $\hat{L}$  is the operator of one the iterative HITS or PageRank algorithms that are considered in this work;  $\vec{s}(0)$  is the initial values of nodes.

$$\vec{s}(0) = \lim_{n \rightarrow \infty} \vec{s}(n), \quad n = 0, 1, \dots \quad (25)$$

and, respectively

$$\hat{L}\vec{s}(0) = 0. \quad (26)$$

The initial values of nodes  $\vec{s}(0)$  is considered as the solution –  $\vec{s}^0$  (these values are equilibrium). Next, we propose to make some deviation of the score (called perturbation) of the  $m$ -th node, as it is shown in the next example:

$$\vec{s}_i(0) = \vec{s}_i^0 + \alpha \delta_{im} \vec{s}_i^0, \quad (27)$$

where  $\alpha$  defines the deviation (perturbation) score of the  $m$ -th vector element;  $\delta_{ik}$  is the Kronecker symbol.

The described above means vector forms of the deviation from the equilibrium state one of the elements of the vector  $\vec{s}^0$ . The perturbation of vector  $\vec{s}^0$  due to the perturbation of one of its elements leads a non-equilibrium state of the system.

According to the equation (24), for  $n=0$  it can be rewritten:

$$s_i = s_i(0) + \sum_k L_{ik} s_i(0), \quad (28)$$

taking into account (27) it looks as follows:

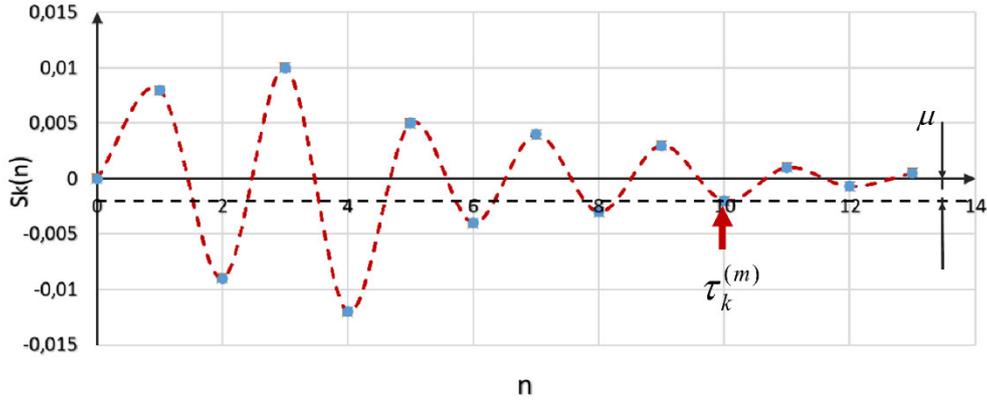
$$s_i(1) = s_i(0) + \sum_k L_{ik} s_i^0 + \alpha \sum_k L_{ik} \delta_{km} s_k^0 = s_i^0 + \alpha q_i^{(m)}, \quad (29)$$

where the vector  $q_i^{(m)} = \sum_k L_{ik} \delta_{km} s_k^0$  can be defined as

$$q_i^{(m)} = \begin{pmatrix} L_{11} & L_{12} & \dots & L_{1m} & \dots & L_{1N} \\ L_{21} & \dots & \dots & L_{2m} & \dots & L_{2N} \\ \vdots & & & \vdots & & \\ L_{m1} & & & \vdots & & \\ \vdots & & & \vdots & & \\ L_{N1} & & & L_{Nm} & & L_{NN} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \vdots \\ s_m^0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} L_{1m} \\ L_{2m} \\ \vdots \\ \vdots \\ L_{1m} \end{pmatrix} s_m^0. \quad (30)$$

Taking into account the initial condition  $s_i(1)$ ,  $s_i(n)$  come back to the initial equilibrium state  $\vec{s}^0$  after increasing a number of iterations  $n$  in accordance to (25).

The initial scores  $s_k(0) = s_k^0$ ,  $k \neq m$ . After  $n \geq 10$  (more than 10 step) the values  $s_k(n \geq 10)$  decrease and convergence to the value  $\mu$ .



**Figure 2:** Shema of values convergence the of the  $k$ -th node

The value  $\tau_k^{(m)}$  is the  $k$ -th node relaxation time after the perturbation of the  $m$ -th node, when the following condition is satisfied

$$|s_k(n \geq \tau_k^{(m)})| \leq \mu. \quad (31)$$

In the other words, the maximum value of  $\tau_k^{(m)}$  for  $\forall_k \max(\tau_k^{(m)})$  after the perturbation of the  $m$ -th node is the network relaxation time [12] for the  $m$ -th node.

And respectively, the relaxation time of perturbed node called the individual relaxation time of this node.

## 5. Deceleration of algorithm

When calculating the relaxation time for many small networks, a problem arises, which is that the relaxation time (a discrete number of algorithm steps) can be very small, that is, the effect of "small distributive force" occurs. To solve this problem, it is proposed to apply an artificial deceleration of the iterative algorithm. That is, the so-called decelerated iterative algorithms for HITS and PageRank can be used to calculate the relaxation time. Cause the number of connections between the nodes is large, the process of iterative recalculation of the values of the nodes after perturbation of some node and achieving their initial values is fast. It means that we need to make only a few iterative steps to achieve the initial equilibrium state of nodes after perturbation. In other words, a network relaxation is fast. In order to decelerate the relaxation time, we propose to decelerate the HITS or PageRank algorithm, respectively. After the deceleration, the process of convergence to the equilibrium initial values of nodes after their perturbation will be slowed. So, the corresponding decelerated HITS or PageRank algorithm is applied

$$PR_0(j) \rightarrow PR_1(j) \rightarrow PR_2(j) \rightarrow \dots \rightarrow PR_{\text{equilibrium}}(i), \quad (32)$$

$$\hat{A}PR_1(i) = PR_2(i), \quad \hat{A}PR_n(i) = PR_{n+1}(i), \quad (33)$$

$$PR_{n+1}(i) \leftarrow \beta PR_{n+1}(i), \quad (34)$$

where  $0 < \beta < 1$  is a deceleration factor,  $\hat{A}$  is the HITS or PageRank algorithm operator.

## 6. Examples of network analysis

Research of complex networks shows that the values of the network characteristics of nodes after external perturbation and the next recalculation of these characteristics, recover their initial equilibrium values during some individual time for each node. Comparing the relaxation time with the network characteristic, the value of which was perturbed, a vague dependence exists, which shows that the relaxation time is a unique and incomparable numerical characteristic of network nodes [12].

Within this work, the individual relaxation time  $\tau_m^{(m)}$  is researched, i.e., the relaxation time of the node, which was deviated from the equilibrium state. In this case, we will further use the notation  $\tau_m$ , omitting the upper symbol in  $\tau_m^{(m)}$ .

Also in the work, the general relaxation time of the network for the  $m$ -th node is researched –  $\max_k(\tau_k^{(m)})$  the highest value of  $\tau_k^{(m)}$  among  $\forall_k$  when the  $m$ -th node was disturbed.

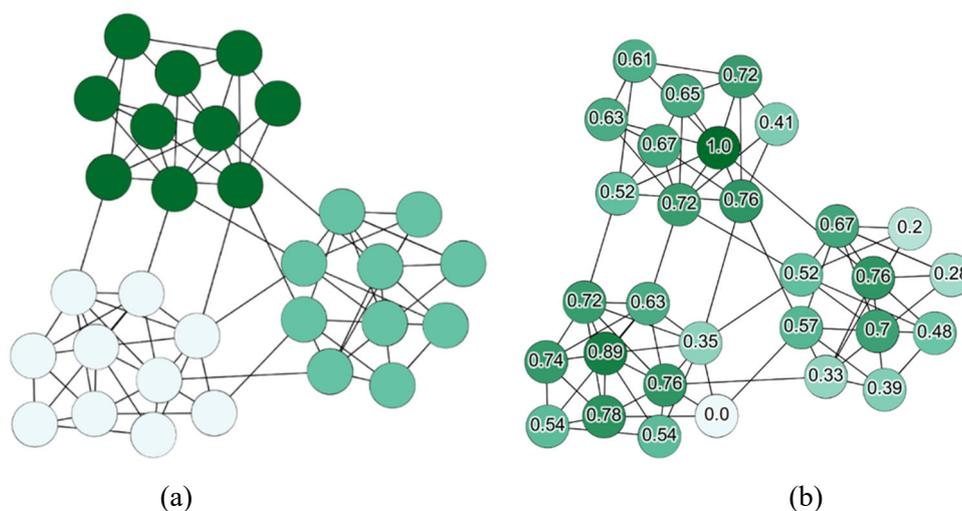
The approbation of the presented characteristics of the relaxation time and the individual relaxation time was carried out on the example of clustering of random networks with clearly expressed clusters. In particular, a randomly generated matrix with dimension  $30 \times 30$  and 3 clusters and a matrix with dimension  $100 \times 100$  and 4 clusters were researched.

For each network described by the corresponding matrix, after perturbing the value of the network characteristic  $h(j)$  using decelerated HITS algorithm, the network relaxation time and the individual node relaxation time were calculated. For each network, the deceleration factor  $\beta$  of the HITS algorithm and the convergence condition  $\mu$  of the numerical values  $h(j)$  and  $a(j)$  to the preperturbed values were chosen individually.

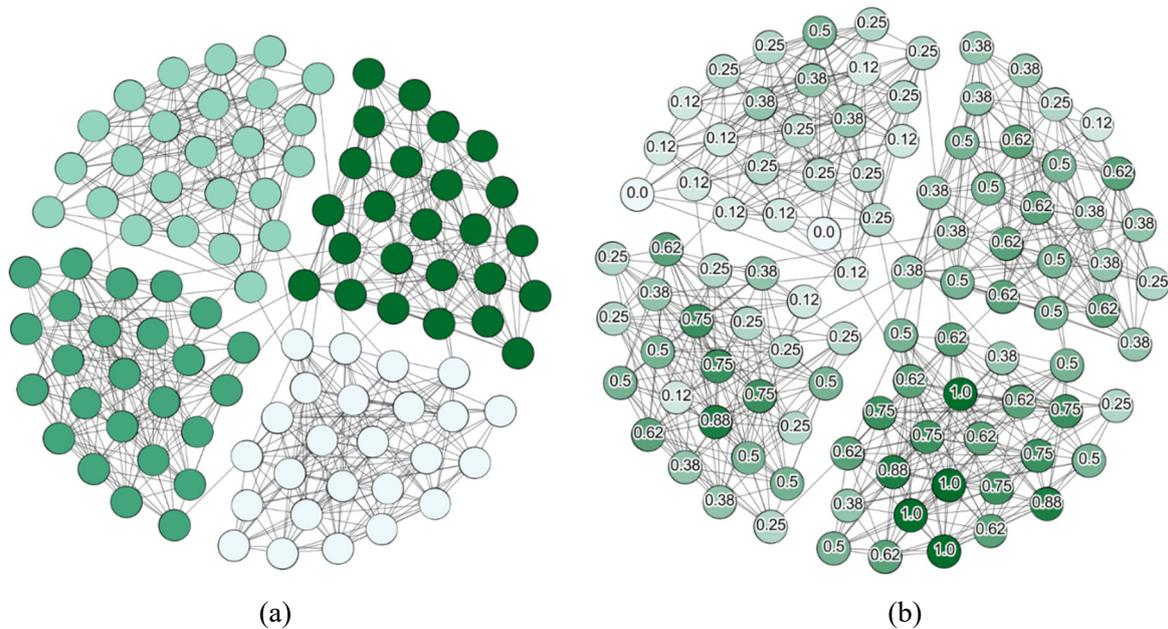
Also, in order to avoid large numerical values that sometimes arise after the using of decelerated algorithms, all the resulting values of the relaxation time indicators were normalized in the interval  $[0,1]$ .

Figures 3 and 4 show a visual representation of networks. In each of the figures, the nodes are colored according to the modularity class (figures 3(a) and 4(a)) and by the network relaxation time and the individual node relaxation time (for the presented examples, the normalized numerical values of these characteristics coincide and are presented in the form of node labels – Figures 3(b) and 4(b)).

The relaxation time of the network, which presented in Figure 1, was obtained as a result of deceleration the HITS algorithm with a deceleration factor  $\beta = 0.9$  and convergence condition  $\mu = 10^{-6}$ . For the network presented in figure 2 –  $\beta = 0.9$  та  $\mu = 10^{-4}$ .



**Figure 3:** An example of a network corresponding to a randomly generated  $30 \times 30$  matrix with 3 clusters where groups of nodes are united (colored) by a) modularity class and b) relaxation time (numeric values are represented as node labels)



**Figure 4:** An example of a network corresponding to a randomly generated 100x100 matrix with 4 clusters where groups of nodes are united (colored) by a) modularity class and b) relaxation time (numeric values are represented as node labels)

It can be noticed that the groups of nodes united by the modularity class differ from the groups obtained by the union of nodes by the relaxation time. This means that the network relaxation time and the individual node relaxation time are unique network node characteristics and can be used to find cluster centroids and cluster nodes based on these characteristics.

## 7. Conclusion

This paper researches the proposed network node characteristics – the network relaxation time and the individual node relaxation time. The approbation of the presented characteristics of the relaxation time and the individual relaxation time was carried out on the example of clustering of random networks with clearly expressed clusters. It is shown, that these characteristics are unique numerical characteristic of network nodes, and they can be used to find the centroids of clusters and combine nodes into groups according to these characteristics, in other words, it can be used for complex networks clustering.

The proposed and researched in this work numerical network and node characteristics can be used during research and analysis of the network structure, making it possible to identify the most important structural elements. Also, results of the research can be used when building personal search interfaces for users of information and search systems. In turn, it will simplify the process of finding the necessary information.

The given algorithm, like the well-known LSA algorithm, is a cluster analysis algorithm that uses a matrix representation of data. The complexity of the algorithm is determined by the complexity of the PageRank, HITS or any other iterative algorithm, which is used to recalculate node weights.

The novelty of the algorithm is defined by the approach for calculating the weight of nodes (the values of relaxation time of network or relaxation time of nodes), the most significant of which can be used as centres for determining clusters (centroids). If it is necessary to improve the quality of the proposed approach, certain centroids can be passed as input to other well-known algorithms such as K-means.

## 8. References

- [1] A. L. Barabási, H. Jeong, Z. Néda, E. Ravasz, A. Schubert, T. Vicsek, "Evolution of the social network of scientific collaborations." *Physica A: Statistical mechanics and its applications* 311.3-4 (2002): 590-614. doi: 10.1016/S0378-4371(02)00736-7
- [2] M. E. J. Newman, "The structure and function of complex networks." *SIAM review* 45.2 (2003): 167-256. doi:10.1137/S003614450342480
- [3] S. H. Strogatz: "Exploring complex networks." *nature* 410.6825 (2001): 268-276. doi: 10.1038/35065725
- [4] A. A. Snarskii, D. V. Lande, *Modeling of complex networks: tutorial*, K.: Engineering, 2015. ISBN 978-966-2344-44-8
- [5] R. Cohen, S. Havlin, *Complex networks: structure, robustness and function*. Cambridge university press, 2010. doi: 10.1017/CBO9780511780356
- [6] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D. U. Hwang, *Complex networks: Structure and dynamics*, *Physics reports* 424.4-5 (2006): 175–308. doi: 10.1016/j.physrep.2005.10.009
- [7] R. Albert, A. L. Barabási, "Statistical mechanics of complex networks." *Reviews of modern physics* 74.1 (2002): 47. doi: 10.1103/RevModPhys.74.47
- [8] T. Murata (Ed.), *Handbook of social network technologies and applications*. Springer Science & Business Media, 2010. doi: 10.1007/978-1-4419-7142-5\_12
- [9] A. Lancichinetti, S. Fortunato, "Community detection algorithms: a comparative analysis." *Physical review E* 80.5 (2009): 056117. doi: 10.1103/PhysRevE.80.056117
- [10] V. Estivill-Castro, "Why so many clustering algorithms: a position paper." *ACM SIGKDD explorations newsletter* 4.1 (2002): 65-75.
- [11] S. White, P. Smyth, "A spectral clustering approach to finding communities in graphs," in: *Proceedings of the 2005 SIAM international conference on data mining*, 2005, pp. 274-285. doi: 10.1137/1.9781611972757.25
- [12] D. Lande, A. Snarskii, O. Dmytrenko, I. Subach, "Relaxation time in complex network," in: *Proceedings of the 15th International Conference on Availability, Reliability and Security, ARES '20*, 99, 2020, pp. 1–6. doi: 10.1145/3407023.3409231
- [13] D. V. Lande, O. O. Dmytrenko, A. A. Snarskii, "Research of network's relaxation coefficient as characteristics of network's nodes." *Data Recording, Storage & Processing* 21.1 (2019): 83-94. doi: 10.35681/1560-9189.2019.1.1.179714
- [14] D. Lande, I. Subach, O. Puchkov, A. Soboliev, "A Clustering Method for Information Summarization and Modelling a Subject Domain." *Information & Security* 50.1 (2021): 79-86. doi: 10.11610/isij.5013
- [15] A. Likas, N. Vlassis, J. J. Verbeek, "The global k-means clustering algorithm." *Pattern recognition* 36.2 (2003): 451-461. doi: 10.1016/S0031-3203(02)00060-2
- [16] Gephi: makes graph handy, 2022. URL: <http://gephi.org>
- [17] K. Knox, "Le Châtelier's Principle." *Journal of Chemical Education* 62.10 (1985): 863.
- [18] A.N. Langville, C.D. Meyer, "Google's PageRank and beyond." *Google's PageRank and Beyond*. Princeton university press, 2011. doi: 10.1515/9781400830329
- [19] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment." *Journal of the ACM (JACM)* 46.5 (1999): 604-632.