# Analysis of Communities Detection Algorithms in Complex Networks

Moises Bruno L. Bissoto
Computer Science Department
University Federal of Tocantins
Palmas/Tocantins - Brazil

Ary Henrique M. Oliveira
Computer Science Department
University Federal of Tocantins
Palmas/Tocantins - Brazil

Glenda M. Botelho
Computer Science Department
University Federal of Tocantins
Palmas/Tocantins - Brazil

## ABSTRACT

Complex networks are an imminent multidisciplinary field defined by graphs that present a nontrivial topographic structure. An important information extracted from a complex network is its communities structure. In the literature, there are several communities detection algorithms, however, new research have emerged with the aim of detecting communities efficiently and with lower computational cost. Therefore, this work analyzes different algorithms for communities detection in complex networks with different characteristics, considering the Modularity measure, the execution time and the obtained communities number. The partitions obtained by the different algorithms presented high modularity values and it was observed that the influence of the number of vertices and edges in the execution time of some detection algorithms.

### General Terms

Complex Networks, Algorithms

### Keywords

Complex networks, Community detection algorithms, Modularity measure, Evaluation

## 1. INTRODUCTION

The research in the complex networks area began in the mid-1930s in the sociology field, when networks were used to model and analyze the society behavior and the relationships between individuals. Nowadays, they are used to solve problems in different areas and can be used in several aspects of the real world, such as social networks and the internet, biological neural networks, metabolic networks, food chains, among others.

Initially, the complex networks research were based on measures such as centrality (vertex more central) and connectivity (vertex with the highest connectivity). Technological advancement and increased computational power allowed robust analyzes in large-scale (millions or billions of vertices). These analyzes revealed characteristics that substantially differ real-world networks of random networks. For example, it is possible to observe the presence of distinct and robust organizational properties in real networks.

Each network has a particular structure and characteristics without a defined standard, but all have the presence of communities, which are groups of vertices that have high density of edges between them, with a low density of edges betwenn the group and the others, as can be seen in the Figure 1. Communities detection together with the knowledge extraction from their structure has been explored extensively in machine learning and data mining research [19] [10]. Community detection is one of the great challenges of the machine learning field due to the good part of the traditional algorithms present computationally unstable to treat large amounts of vertices and edges. These features make this theme relevant and promising.

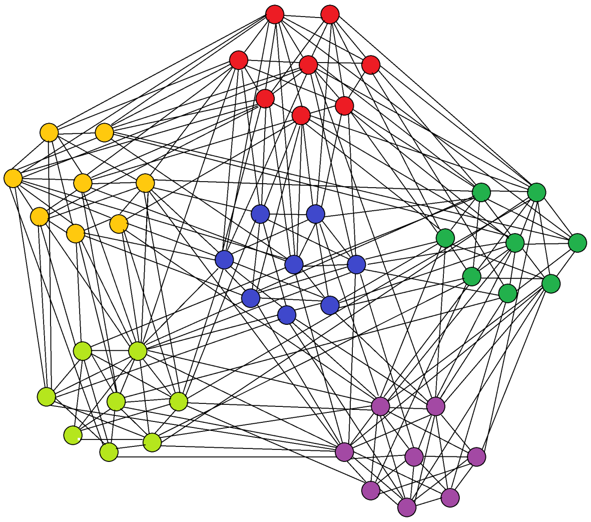


Fig. 1. Example of communities present in a network.

Communities detection correlates with clustering techniques of machine learning. This enables the easy adaptation of algorithms from one area to another. For example, partitioning methods are used to derive subgraphs representing the network. However, partitioning alone is not satisfactory mainly because the number and size of communities are not initially knonw. In this case, hierarchical clustering can be used to discover natural divisions of the network based on similarity concepts between vertices.

Two approaches can be derived from hierarchical clustering: agglomerative and divisive [10]. In the agglomerative approach, each vertex is considered a unitary community and then edges are iteratively added to the graph to join the subgraphs until all the vertices form a single graph. The divisive approach starts with only one graph containing all the vertices. In the sequence, it proceeds by dividing until each vertex becomes an isolated graph or until the algorithm reaches a stopping criterion as the desired subgraphs number.

Other proprosed algorithms use different approaches such as the Betweenness measure, proposed by Girvan and Newman [9], which uses the minimum path calculation between vertices to support communities detection. The Modularity measure was also proposed to measure the quality of possible divisions in the network without the need for prior knowledge of its structure. From this work, several research have been carried out in order to optimize the modularity measure such as the extreme optimization [6] and the optimization using Monte Carlo method proposed by [11].

There are several communities detection algorithms in the Igraph tool [5], that is a free software with advanced features in the complex networks area with a vast library available for three languages and three operating systems. Igraph tool provides access to information for various purposes, especially the academic and commercial. This tool allows to perform several analyzes based on the application domain characteristics. The algorithms contained in the Igraph tool allows to perform studies to understand the relevance of partitioning in communities and to demonstrate the importance of an algorithm appropriate to each situation or context.

This paper analyzes different algorithms, contained in the Igraph tool, for communities detection in complex networks with different characteristics, considering the Modularity measure, the execution time and the communities number obtained by algorithms. The community detection algorithms considered were Betweenness [9], Fastgreedy [4], Eigenvector [14], Spinglass [18] and Walktrap [17]. Section 2 presents the fundamental concepts about complex networks, communities detection algorithms and Modularity measure. The section 3 presents the methodology used for the comparison of communities detection algorithms. Section 4 presents the discussion about the results and the section 5 presents the conclusions about the realized research and future works.

## 2. THEORICAL FOUNDATIONS

### 2.1 Complex Networks

A network can be defined as a set of points, called vertices, interconnected by connections called edges [12]. The study of networks has become important because it is possible to apply its concept in several distinct systems and areas, such as biological, social, chemical and man-made systems. All the mentioned systems, although of very distinct areas, present the concept of networks in common and their composition forms what is known by complex networks.

A complex network is defined as a graph that presents a non-trivial topological structure (non-regular pattern). However, there is difficulty in finding in the literature an universally accepted and clear conceptualization of a regular standard applicable to this context [2]. Some works understand Complex Networks as those that model large systems, favoring size [12] and/or relying on common sense to delimit their scope [1]. But, there are several other approaches suitable for this research area.

### 2.2 Communities Detection

Most of the real networks have non-homogeneous connection structures characterized by the presence of groups in which the vertices are more strongly connected to each other than to the rest of the network. This feature is called communities, as can be seen in Figure 2. Detecting these communities in large networks is a useful task because vertices belonging to the same community are more likely to share properties. In addition, the quantity and characteristics of the communities provide subsidies to identify the network type, as well as to understand its organization and dynamic evolution [8].

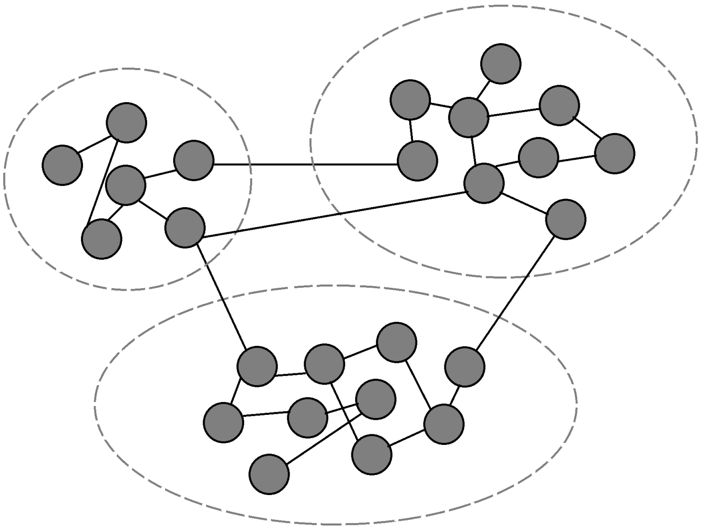


Fig. 2. Network with three communities well defined. Communities are groups whose vertices are most strongly interconnected.

Communities detection in complex networks is a growing and relatively recent research area involving clustering techniques present in machine learning, more specifically, in the unsupervised learning model. However, many of the classical algorithms for detecting groupings in graphs are inefficient when applied to complex networks, with very large number of vertices and edges and that model a real system, which have a dynamic behavior.

So, many algorithms proposed for this purpose have appeared in the literature, most of them based on the hierarchical model of clustering [10], because the number and size of the communities present in the network is not known a priori and this model allows the generation of a tree, known as Dendogram, that shows the order of formation of the clusters for different amounts of groups. There are also other algorithms that are not based on clustering, such as spectral methods [14] (which use eigenvectors and eigenvalues matrix derived from the network), local methods [3] (which evaluate the current community and its neighboring communities), Betwenness measure [9] (which is based on the calculation of the minimum path) and Modularity measure [2] (which estimates the quality of possible divisions of the network into communities).

### 2.3 Modularity Measure

Newman [6] defined a modularity function Q, which measures the quality of a possible division of the network into communities, that is, of a given division of the graph to be significant or not. This function is given by equation 1.

$$Q = \sum_{i} (e_{i,i} - a_i^2) \tag{1}$$

where $e_{i,i}$ is the fraction of the network edges that are inserted into the community $i$, and $a_i^2$ is this same fraction, but considering that edges are inserted randomly. $Q$ with values close to 0 indicates a low probability of the network being divided into real communities. In this sense, it is observed that values positive and distant of 0 (values equal or greater than $0.3$ are already considered significant), increases the chance that these groupings do not exist only at random (their presence is intrinsic to the structure and semantics of the network). In the same work, Newman proposes the use of this measure together with a greedy agglomerative hierarchical algorithm, which started from a state in which each vertex represents a community. Then, communities are connected two by two, repeatedly, until all the vertices are part of the same community. $Q$ is calculated for the initial state and in each fusion between two communities $i$ and $j$, the value of the variation in $Q$ can be measured as equation 2:

$$\Delta Q = 2(e_{i,i} - a_i a_j) \tag{2}$$

where $e_{i,i}$ is the fraction of edges that connect the community $i$ to community $j$, $a_i$ is the total fraction of edges that connect the community $i$ to other network communities and can be calculated by $\sum_k e_{i,k}$, as well as $a_j$ is the total fraction of edges that connect the community $j$ to other network communities and can be calculated as the same way that $a_i$. Thus, the division of the network that obtains the maximum result of $Q$ will be considered the best possible division of the network in communities.

## 3. METODOLOGY

In order to compare the performance of some communities detection algorithms in complex networks, this paper uses Newman's modularity [13] to measure how much the network division in communities was significant. In addition, two other measures were used: the execution time of the algorithm and the obtained communities number (in the case of the best value of the modularity measure). To allow the experiments to run, various features have been used and are specified in the following subsections.

### 3.1 Complex networks

Four complex networks were selected for the experiments. These complex networks represent graph, which have different amounts of vertices and edges. In addition, they can be directed, non-directed, connected or disconnected. With this, it was possible to test the algorithms in different situations, providing more reliable results. The used complex networks are available in the online repository Nexus, an extension of the Igraph website [5], that is made available for free. Some characteristics of the selected complex networks are presented below.

—*Zachary's Karate Club [20]:* social network of relationships among 34 members of a karate club at a University of the United States. The network is represented by a graph with $34$ vertices and $78$ non-directed edges.

—*American College Football [9]:* American football network between high school colleges. The network is represented by a graph with $115$ vertices (each vertex represents a team) and $615$ non-directed edges, where each edge corresponds to a game between two teams.

—*Neural Network [7]:* directed and weighted network representing the neural network of C. Elegans. In total there is a graph with $297$ vertices and $2.359$ edges.

—*Coauthorship Network Science (Netscience) [14]:* Network of scientists co-authorship. In total, there is a disconnected graph with $1.589$ vertices and $2.742$ non-directed and weighted edges.

### 3.2 Communities Detection Algorithms

Communities detection algorithms belonging to different approaches (divisive, agglomerative, spectral and modularity optimization) were selected, enabling a direct comparison of its performance in different complex networks. All selected algorithms are contained in the Igraph library [5].

—*Betweenness*[9]: divisive algorithm based on the count of minimum paths between vertices. In this way, it is defined that edges with low value of Betweenness belong to the same community and edges with high value of Betweenness separate different communities.

—*FastGreedy*[4]: algorithm that optimizes Newman's original modularity measure [6] using a greedy search.

—*Eigenvector*[14]: algorithm that reformulates the modularity concept in terms of eigenvectors and eigenvalues of a new matrix, the modularity matrix.

—*Spinglass*[18]: algorithm that optimizes Newman's modularity using Simulated Annealing.

—*Walktrap*[17]: algorithm that considers short random paths tend to be in the same community.

### 3.3 Modularity measure

The modularity measure implemented in the Igrapy tool is a version optimized of the original measure and was proposed by Newman and Girvan in [16]. The equation 3 shows the used modularity meansure $Q$:

$$Q = \frac{1}{2m} \sum_{ij} [A_{ij} - \frac{k_i k_j}{2m} \delta(c_i c_j)] \tag{3}$$

Where $m$ is the edges number of the network, $A_{ij}$ is the element $ij$ of the incidence matrix $A$, $k_i$ is the degree of the vertex $i$, $k_j$ is the degree of the vertex $j$, $c_i$ is the value of the component of the vertex $i$, $c_j$ is the same value for vertex $j$. The function $\delta$ returns 1 case $i$ and $j$ belong to the same component and 0 otherwise. The summation runs through all pairs of vertices $(i, j)$ of the network. This function in the Igraph library receives the original network, the tree (dendogram) that contains the formation hierarchical of the communities and a parameter that indicates which position in the tree to be analyzed. Given this, a loop was used to sweep every tree and select the division to return the highest modularity value. However, the variation in $Q$ was not calculated, beucause $Q$ was directly calculated for each level of the tree (dendogram).

### 3.4 Computational Environment

The experiments were performed on a machine with a processor Core 2 Duo of 2.00 GHz*2 and 2GB of RAM, containing 64-bit Linux operating system (Ubuntu, version 12.04 LTS). In addition, the Igrapy library [5] was used, which is a collection of network analysis tools, with emphasis on efficiency, portability and use easy. The Igraph library is open source, free and can be programmed in GNU R, Python and C/C ++. The Igraph library can be found at http://igraph.org/.

## 4. RESULTS AND ANALYSIS

The selected communities detection algorithms were evaluated considering three measures: modularity, execution time (measured in seconds - *s*) and communities number obtained for the highest modularity value. Table 1 presents the results for the Zachary's Karate Club network [20] and Figure 3 graphically compares the modularity measures obtained by the different algorithms. It is noticed that all the algorithms obtained satisfactory modularity values, low execution time and similar communities number.

Table 1. Results of the Karate Club network.

| Algorithms | Modularity Value | Time(s) | Communities |
|---|---|---|---|
| Betweenness | 0.4012985 | 0.060 | 5 |
| FastGreedy | 0.3806706 | 0.023 | 3 |
| Eigenvector | 0.3934089 | 0.127 | 5 |
| Spinglass | 0.4449036 | 0.689 | 4 |
| Walktrap | 0.3532216 | 0.001 | 5 |

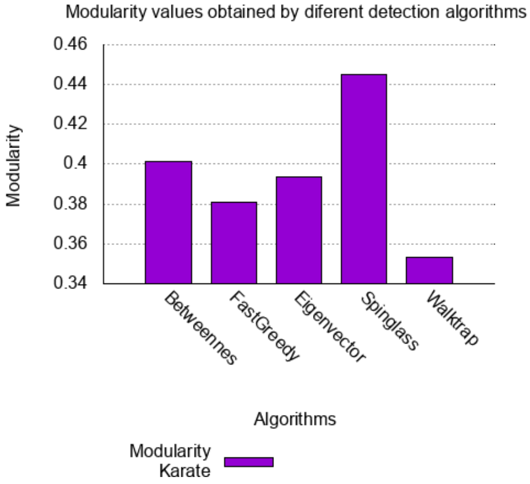


Fig. 3. Modularity values obtained in the Karate Club network by the diferent detection algorithms.

Table 2 presents the results obtained in the American College Football network [9] and the Figure 4 graphically demonstrates the modularity measures found by algorithms. The network is more robust, with 115 vertices and 615 edges. It should be noted that the FastGreedy [4] algorithm was unable to work with this network. The other algorithms obtained high values of modularity, similar communities and low execution time (mainly the Walktrap algorithm [17]).

Table 2. Results of the Football network.

| Algorithms | Modularity | Time(s) | Communities |
|---|---|---|---|
| Betweenness | 0.6009076 | 0.611 | 10 |
| FastGreedy | - | - | - |
| Eigenvector | 0.4884414 | 0.034 | 8 |
| Spinglass | 0.5972113 | 1.243 | 12 |
| Walktrap | 0.6042796 | 0.004 | 10 |

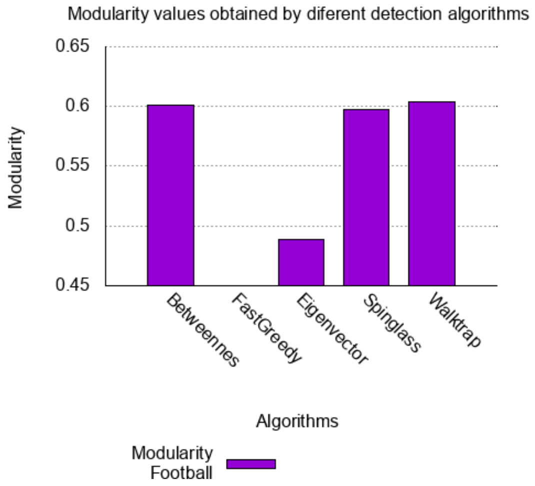


Fig. 4. Modularity values obtained in the Football network by the diferent detection algorithms.

Results contained in Table 3 refer to the Neural Network [7] and the Figure 5 presents the modularity values obtained for this network. This network presents the more robust graph than the previous ones, composed by 297 vertices and 2359 directed edges. It should be noted that the FastGreedy [4] and Eigenvector [14] algorithms do not work with directed graphs. The other results were quite different. The algorithm Betweenness [9] presented extremely low modularity and a high number of communities. The algorithm Walktrap [17] presented a satisfactory modularity value and the shortest execution time. Finally, the Spinglass algorithm [18] presented satisfactory modularity value, but the longest execution time.

Table 3. Results of the Neural Network.

| Algorithms | Modularity | Time(s) | Communities |
|---|---|---|---|
| Betweenness | 0.0886453 | 7.159 | 198 |
| FastGreedy | - | - | - |
| Eigenvector | - | - | - |
| Spinglass | 0.4905347 | 13.659 | 6 |
| Walktrap | 0.4693830 | 0.021 | 24 |

Finally, Table 4 presents the results for the Netscience network [14] and Figure 6 graphically shows the obtained modularity values. The Netscience network [15] is the most robust of all, represented by a disconnected graph with 1589 vertices and 2742 edges. It should be noted that the Spinglass algorithm [18] does not work with disconnected graphs. Already the other algorithms presented high modularity values (probably due to the graph being disconnected) and resulted in different communities numbers, but with close values. The FastGreedy algorithm [4] had the best execution time, even with a high number of vertices. Already the Betweennes algorithm obtained the worst execution time.

To allow a better analysis of the results, measures were analyzed together. The graphics shown in the Figures 7 and 8, respectively, present the relationship between modularity versus vertices number and modularity versus edges number. With that, results showed that with few vertices and edges, the different algorithms present similar modularity values. However, with increasing number of vertices and edges, it is not possible to establish a pattern.

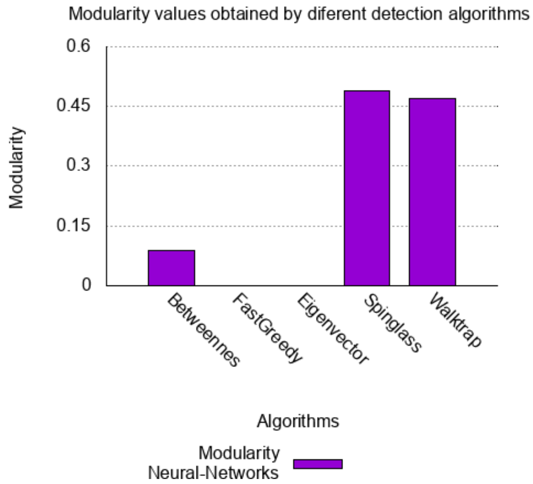


Fig. 5. Modularity values obtained in the Neural network by diferent detection algorithms.

Table 4. Results of the Netscience network.

| Algorithms | Modularity | Time(s) | Communities |
|---|---|---|---|
| Betweenness | 0.9251278 | 17.306 | 405 |
| FastGreedy | 0.9551002 | 0.0007 | 403 |
| Eigenvector | 0.9408197 | 0.390 | 411 |
| Spinglass | - | - | - |
| Walktrap | 0.9559724 | 0.045 | 416 |

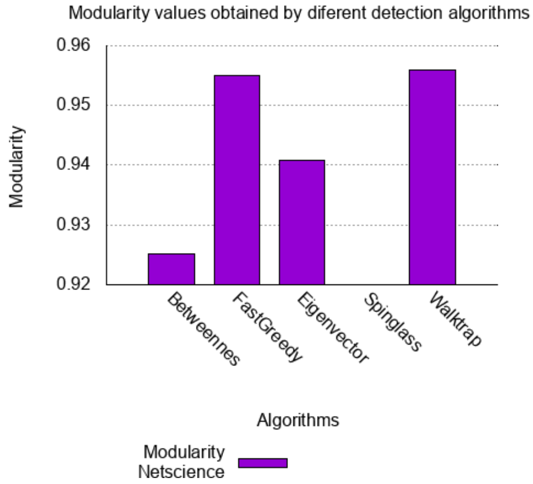
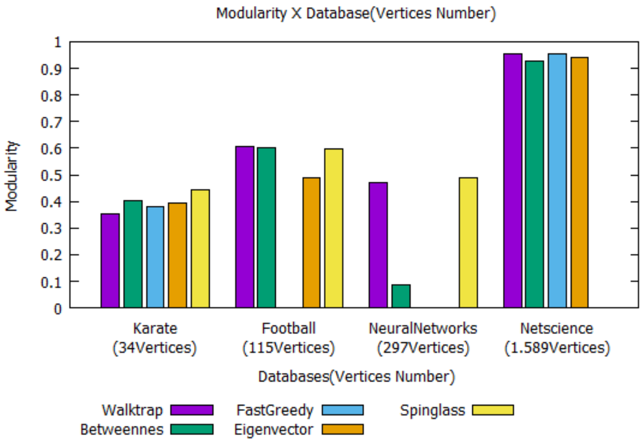


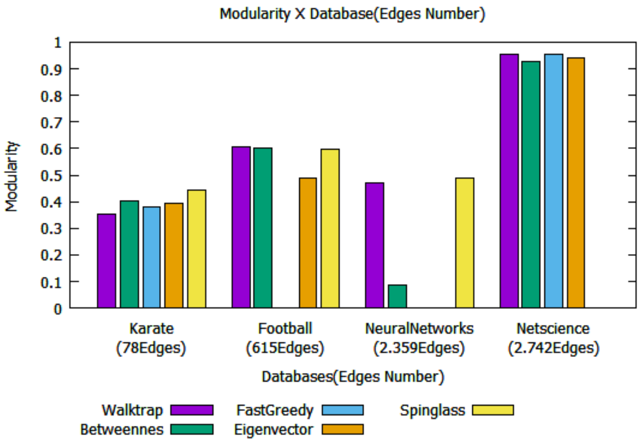
Fig. 6. Modularity values obtained in the Netscience network by diferent detection algorithms.

The graphics shown in the Figures 9 and 10 present, respectively, the relation between execution time versus vertices number and execution time versus edges number. It is possible to observe that the execution time increases when the number of vertices and edges also increases, except for the Walktrap, Eigenvector and FastGreedy algorithms. It should be noted that, in most cases, the Spinglass and Betweenness algorithms present the longest execution time.



5.png

Fig. 7. Relationship between modularity versus vertices number.



6.png

Fig. 8. Relationship between modularity versus edges number.

## 5. CONCLUSIONS

In this work, experiments were conducted in four known complex networks: Zachary's Karate Club, American College Football, Neural Network and Coauthorship Network Science (Netscience). These experiments had the objective of evaluating the comunnities detection algorithms Betweenness, FastGreedy, Eigenvector, Spinglass and Walktrap, belonging to the different approaches, by calculating the modularity value of the partition obtained by the algorithms, the execution time and the communities number found in the partition with the greatest modularity value. In most cases, algorithms resulted in partitions with satisfactory modularity values. However, there is not consensus between the different algorithms in relation to the communities number contained in the partition with the highest modularity value. As this information generally is not known a priori, it is expected that partitions with the highest modularity value will result in better division of the network into communities.

It was also possible to observe that the algorithms present an increase in the execution time as the number of vertices and edges of the network increases, but some of them increase in less significant quantities than others. For example, the algorithms that maintained

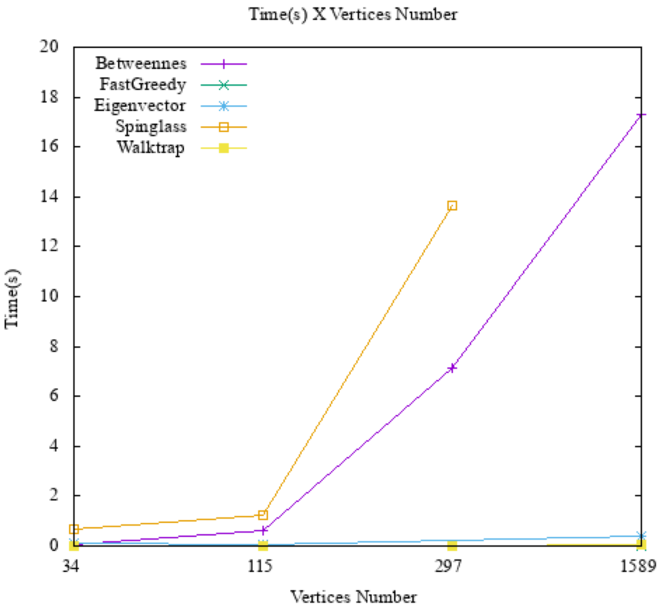


Fig. 9. Relationship between execution time versus vertices number.

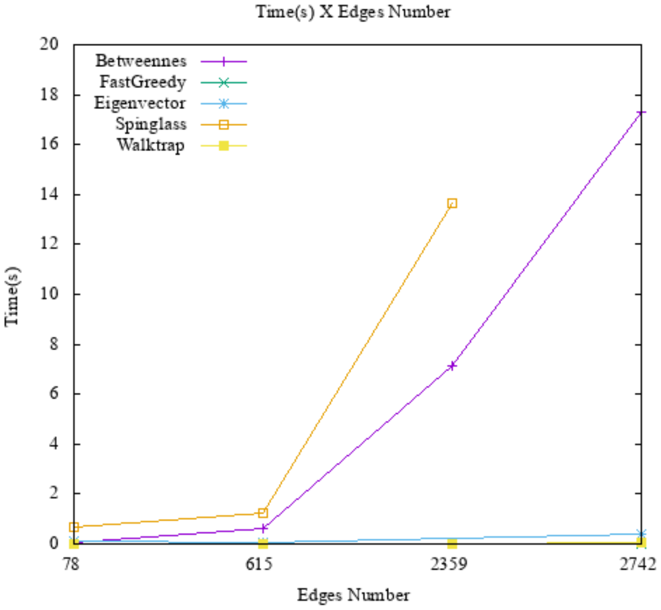


Fig. 10. Relationship between execution time versus edges number.

a low execution time were Walktrap, FastGreedy and Eigenvector, which are not so influenced by the number of vertices and edges, being executed quickly. It is important to notice that these algorithms also presented satisfactory modularity values.
The next step for further improvements consist of evaluating the communities detection algorithms considering other databases (including databases with a greater number of vertices and edges, for example, the Internet database). In addition, it is necessary to compare the results with others found in the literature. In this way, it will be possible to confirm more accurately the behavior obtained by the algorithms in the realized experiments.

## 6. REFERENCES

[1] Barabsi AL. *Linked: How everything is connected to everything else and what it means for business, science, and everyday life*. Basic Books, 2014.

[2] A. F. Angelis. Redes complexas (complex networks). Technical report, So Paulo University, 2005.

[3] James Bagrow and E.M. Bollt. A local method for detecting communities. 72:046108, 11 2005.

[4] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Physical Review E, APS*, 70(6):1–6, aug 2004.

[5] G. Csardi and T. Nepusz. The igraph software package for complex network research. *Inter Journal, Complex Systems*, 1695(5), 2006.

[6] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. *Physical Review E, APS*, 72(2):145–157, 2005.

[7] J. G. White et al. The structure of the nervous system of the nematode caenorhabditis elegans. In *Philosophical Transactions of the Royal Society of London*, volume 314, pages 1–340. Biological Sciences, 1986.

[8] L. F. Costa et al. *Characterization of complex networks: A survey of measurements. Advances in Physics*, volume 56. Taylor & Franciss, 2014.

[9] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences, National Acad Sciences*, 99(12):7821–7826, 2002.

[10] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[11] C. P. Massen and J. P. K. Doye. Identifying communities within energy landscapes. *Physical Review E, APS*, 71(4):145–157, 2005.

[12] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45(2):167–256, 2003.

[13] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physical Review E, APS*, 69:1–5, 2006.

[14] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physics Review E, APS*, 74(3):1–22, jan 2006.

[15] M. E. J. Newman. Modularity and community structure in networks. *National Academy of Sciences of the USA*, 103(23):85778582, 2006.

[16] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E, APS*, 69(2):026113, 2004.

[17] P. Pons and M. Latapy. Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications*, 10(2):191–218, 2006.

[18] J. Reichadt and S. Bornholdt. Statistical mechanics of community detection. *Physical Review E*, 74:1–16, 2006.

[19] D. Stauffer, A. Aharony, L. da Fontoura Costa, and J. Adler. Effcient hopfeld pattern recognition on a scale-free neural network. *The European Physical Journal B-Condensed Matter and Complex Systems*, 32(3):395–399, 2003.

[20] W. W. Zachary. An information flow model for conflict and fission in small groups. *Anthropological Research*, 33:452473, 1977.