The Effect of Network Realism on Community Detection Algorithms

Günce Orman, Vincent Labatut

The Effect of Network Realism on Community Detection Algorithms

Günce K. Orman and Vincent Labatut
Computer Science Department
Galatasaray University
Istanbul, Turkey
korman@gsu.edu.tr
vlabatut@gsu.edu.tr

Abstract— **Community detection consists in searching cohesive subgroups in complex networks. It has recently become one of the domain pivotal questions for scientists in many different fields where networks are used as modeling tools. Algorithms performing community detection are usually tested on real, but also on artificial networks, the former being costly and difficult to obtain. In this context, being able to generate networks with realistic properties is crucial for the reliability of the tests. Recently, Lancichinetti *et al.* [1] designed a method to produce realistic networks, with a community structure and power law distributed degrees and community sizes. However, other realistic properties such as degree correlation and transitivity are missing. In this work, we propose a modification of their approach, based on the preferential attachment model, in order to remedy this limitation. We analyze the properties of the generated networks and compare them to the original approach. We then apply different community detection algorithms and observe significant changes in their performances when compared to results on networks generated with the original approach.**

Keywords-complex networks; community detection; random networks, networks generation; networks properties

I. INTRODUCTION

Complex networks constitute a powerful modeling tool, able to represent most real-world systems. The objects composing the system are represented under the form of nodes while their interactions correspond to links. Among the various approaches used to study complex networks properties, community detection has become one of the most popular ones. A community is a cohesive subset of nodes with denser inner links, relatively to the rest of the network [2]. Tens of algorithms exist, based on a whole range of principles: hierarchical clustering, optimization methods, graph partitioning, spectral properties of the network, etc. [3]. Those algorithms are generally tested on a few real and/or artificial networks [4-7]. Using real networks has some limitations: building them is costly and difficult, reference communities are generally subjectively defined, and one network only represent a specific set of properties (size, transitivity, etc.), which makes it difficult to generalize the test results. On the contrary, it is easy to generate large collections of artificial networks exhibiting a wide range of properties and a predefined community structure.

The difficulty with this approach is to design a model able to generate networks with realistic properties, in order for algorithm testing to be relevant. Up to now, only a few methods have been designed for this purpose. The first one, is the model by Girvan and Newman (GN) [4], which produces networks taking roughly the form of sets of small interconnected Erdős-Rényi networks [8]. Although widely used to test and compare community detection algorithms [4, 5], the GN method is limited in terms of realism [1], which is why several variants were defined, producing bigger networks and communities with heterogeneous sizes [3, 7, 9]. More recently, a different approach appeared, based on some rewiring process [1, 10]. It increased the realism level even more by producing networks with power law distributed degree. Among these works, the LFR model proposed by Lancichinetti *et al*. [1] exhibits the most realistic properties, although it does not possess all the properties currently attributed to real-world networks [11].

Interestingly, improvement on the realistic aspect of the generated networks has a noticeable effect on most community detection algorithms [7, 9]. The fact Lancichinetti *et al*.'s method still has room for improvement naturally raises two questions, which we will try to answer in this work: 1) how is it possible to produce more realistic networks, and 2) will this have an effect on community detection algorithms. In the following section, we describe briefly the LFR model, its characteristics and the modification we proposed. We also describe a few community detection algorithms, to be used to test the effect of network realism on community detection. In section III, we present the properties of the networks generated with the modified method, and use them to compare the performances of the community detection algorithms. Finally, we comment these results and propose some further improvements in section IV.

II. METHODS

A. LFR Generative Model

The LFR model was proposed by Lancichinetti *et al*. [1] to randomly generate undirected and unweighted networks with mutually exclusive communities. Nodes degrees and community sizes are both distributed according to a power law. The model was subsequently extended to generate weighted

This project is supported by the Galatasaray University Research Fund.

and/or oriented networks, with possibly overlapping communities [12]. However, in this paper, we focus on non-oriented unweighted networks with non-overlapping communities, because almost all existing community detection algorithms are dedicated to this type of networks. This model allows to control directly the following parameters: number of nodes n, desired average $\langle k \rangle$ and maximum k_{max} degrees, exponent γ for the degree distribution, exponent β for the community size distribution, and mixing coefficient μ. The latter represents the desired average proportion of links between a node and nodes located outside its community, called inter-community links. Consequently, the proportion of intra-community links is $1-\mu$. The communities are well-defined when $\mu<(n-n_c^{max})/n$, where n and n_c^{max} are the number of nodes in the network and in the biggest community, respectively [12].

The generative process first uses the configuration model [13] to generate a network with average degree $\langle k \rangle$, maximum degree k_{max} and power law degree distribution with exponent γ. Second, virtual communities are defined so that their sizes follow a power law distribution with exponent β. Third, an iterative process takes place to rewire certain links, in order to approximate μ, while preserving the degree distribution. By construction, the LFR method guaranties to obtain values considered as realistic [14, 15] for several properties: size of the network, power law distributed degrees and community sizes. Other properties are not directly controlled, so we studied them empirically [11]. It turns out LFR generates small-world networks, with relatively high transitivity and degree correlation. This is realistic [14], but holds only under certain circumstances. In particular, transitivity and degree correlation are dramatically affected by changes in μ, and become clearly unrealistic the closer it gets to 1. These properties are directly related to the network structure, which is the only information used by community detection algorithms, therefore they are particularly important. Moreover, the sensitivity to μ is also a concern: by increasing its value, not only do the communities become less separated, which is the desired behavior, but the network additionally becomes less realistic.

B. Proposed Modification

One of the possible causes for the observed unrealistic properties is the use of the configuration model (CM) [13] to generate the initial network during the LFR first step. On the one hand, the CM is very flexible in the sense it is able to produce networks with any size and degree distribution, but on the other hand it is known these networks have zero correlation [16] and low transitivity (when degrees are power law distributed) [14]. We propose to use a different generative model, with more realistic properties. We considered the Barabási–Albert preferential attachment model (BA) [17] and one of its variants called evolutionary preferential attachment (EV) [18]. The rest of the LFR model is not modified: community sizes are still drawn from a power law distribution, and the rewiring process must be applied to make the community structure appear.

The BA model [17] was designed as an attempt to explain the power law degree distribution observed in real networks by their building process. Starting from an initial connected network of m_0 as nodes, remaining nodes are added one by one and randomly linked to m existing nodes ($m \leq m_0$). These m nodes are selected with a probability which is a function of their current degree k: the higher the degree, the higher the probability. BA produces small-world networks with a power law degree distribution whose exponent tends towards 3 [17]. Transitivity is greater than in Erdős-Rényi networks, but nevertheless decreases with network size following a power law $\sim n^{-0.75}$, while the average degree depends directly on parameter m: $\langle k \rangle = 2m$ [14].

The EV model [18] is a variant of the BA model, supposed to produce networks with high transitivity and degree correlation. Unlike BA, the level of attraction of a node regarding new links is not determined by its degree, but by its performance on a prisoner's dilemma game. Every few iterations, each node "plays" either cooperation or defection against all its neighbors. It gets a total score depending on the individual results: 0 for bilateral defection or unilateral cooperation, 1 for bilateral cooperation, and b for unilateral defection, with $b>1$. An additional parameter, the selection pressure ε, is used to modulate the influence of the preferential attachment mechanism: all nodes are equiprobable when $\varepsilon=0$, whereas the nodes scores are fully considered for $\varepsilon=1$.

C. Community Detection Algorithms

To study the effects of network realism on the community detection process, we applied 4 popular algorithms: Newman *et al.*'s Fast Greedy algorithm (FG) [6] relies on a modularity-based agglomerative hierarchical approach. Its name is due to the use of a standard greedy method, making it relatively faster than earlier algorithms, and allowing it to process large networks. Pons and Latapy's Walktrap algorithm (WT) [7] follows another agglomerative hierarchical method, in which the distance between two nodes is defined in terms of random walk processes. Raghavan *et al'*s Label Propagation algorithm (LA) [19] analyzes information diffusion to identify communities. Each node is initially labeled with a unique value. Then, an iterative process takes place, where each node takes the label which is the most spread in its neighborhood. When the process ends, communities correspond to sets of nodes with identical labels. Blondel *et al.*'s Louvain algorithm (LV) [20] is the most recent of the considered algorithms. It relies on a two-stepped hierarchical modularity optimization method.

III. Results and Discussion

A. Generated Networks Properties

The networks were generated by applying first one of the three previously presented methods (CM, BA, EV) to produce initial networks, and then using the LFR approach to generate the communities sizes and perform rewiring. In other terms, the generating processes differ only in their first step. For simplicity matters, we will thereafter refer to the networks by using only the name of the model employed during the first step. Consequently, CM will correspond to the original LFR method, whereas BA and EV are modified versions based on the corresponding models.

We selected our parameters values based on previous experiments in artificial networks generation [1, 11] and

descriptions of real-world networks measurement from the literature [14, 15]. Some parameters are common to all three processes: we fixed the size n=5000 and the power law exponent for the community sizes distribution β=2; and made the mixing coefficient μ range from 0.05 to 0.95 with a 0.05 step. Other parameters are model-dependent. In particular, with the original LFR method based on CM, it is possible to specify the desired power law exponent γ for the degree distribution, and average $\langle k \rangle$ and maximal degrees k_{max}. We used the values γ=3, $\langle k \rangle$=15, 30 and k_{max}=45, 90. The alternative models do not allow as much control as the CM, and we had to adjust their parameters so that the resulting networks had approximately the same degree-related properties. Preferential attachment does not give any control on γ, which tends towards 3 by construction. To control the average degree, we used m=7, 15 for both BA and EV. The maximal degree is not controlled, but the values measured in the resulting networks are of the same order as the values specified for CM. EV additionally allows controlling transitivity, and we found out score b=1.5 and selection pressure ε=0.99 gave the best results.

We produced 25 networks for each combination of parameters, and averaged the measured properties. Fig. 1 shows the results for average distance, degree correlation and transitivity. Results were very similar for $\langle k \rangle$=15 and 30, so we only present the latter here, but comments apply to both. The largest communities in the generated networks have around 700 nodes, so communities are supposed to be structurally well-defined (cf. section II) for μ<0.86. This mixing limit is represented on the plots under the form of a vertical line.

The average distance is rather similar for all three models, both in terms of absolute value and sensitivity to μ. It ranges approximately from 2.5 to 4, and is relatively stable, especially for μ>0.3. On the one hand, the stability of this property is a good point, since it means networks with much separated communities (small μ) and networks with very mixed communities (high μ) have comparable average distances. Consequently, the effect of this property can be considered as negligible when comparing algorithm performances on networks generated with various μ values. But on the other hand, since all three models lead to very close average distances, this property cannot be used to compare them in terms of realism of the generated networks.

CM has the highest transitivity, with values around 0.6 (the theoretical minimum and maximum being 0 and 1, respectively) for μ≈0, but it also has almost zero transitivity for μ≈1, exhibiting a serious sensitiveness to μ. Other methods also show a decreasing transitivity when μ increases, but the range is much smaller, partly because their values for μ≈0 are significantly smaller: around 0.25 and 0.45 for BA and EV, respectively. Like CM, they reach close to zero values when μ≈1. So contrarily to what we expected, networks generated with EV do not have a higher transitivity than CM, at least for small μ. However, thanks to its lesser sensitivity to μ, EV has a better transitivity for μ>0.3. Note that in the literature, real-world networks with 0.2-0.3 transitivity are considered highly transitive [15], so we can state all three models exhibit realistic transitivity for small μ. The issue is more about their sensitivity to μ, leading to non realistic values for high μ. This non-linear decrease in transitivity observed for all three models could be linked to the rewiring process performed by the LFR method. In this case, the final transitivity would never be stable, whichever model is used to generate the initial networks. But testing this hypothesis would require an exhaustive analysis of the side-effects of rewiring on networks, which is out of the scope of this work. Another explanation would be that network transitivity is directly related to the nature of community structure itself, independent of the way the network is created. Testing this hypothesis would require quantifying the separation level of communities in real-world networks (using Newman's modularity [2], for instance), in order to compare it to the transitivity we measured. But we are not aware of any work of this kind, which is also off-limits for this article.

Considering the degree correlation, there is a clear difference between CM and the other two models. CM degree correlation has acceptable values for small μ (0.25), but it decreases rapidly and oscillates around zero for μ>0.4. EV shows the highest degree correlation, with values greater than 0.5 for μ≈0. It also decreases when μ increases, resulting in values close to 0.25 for μ≈1. Finally, unlike other models, BA degree correlation slightly increases with μ, ranging approximately from 0.25 to 0.35. Although its values are lower than for EV, it is also more stable both in terms of sensitivity to

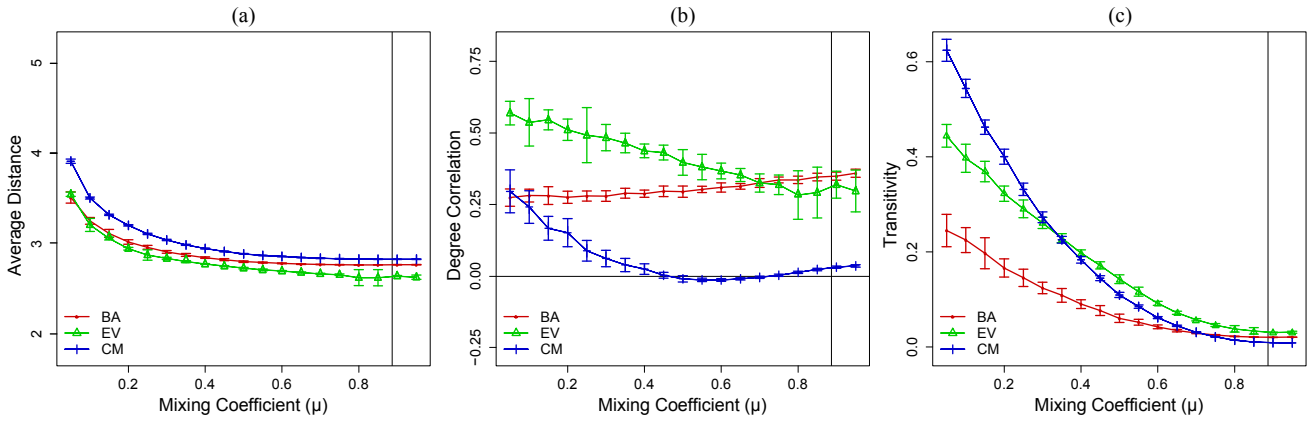

Figure 1. Influence of the mixing coefficient μ on the measured properties: (a) average distance, (b) degree correlation and (c) transitivity. Networks were generated with parameters n=5000, γ≈3, β=2 and $\langle k \rangle$≈30 and using the LFR method on three different generative models: configuration model (CM), Barabási–Albert model (BA) and evolutionary preferential attachment model (EV). Each point corresponds to an average over 25 generated networks. The vertical lines at μ=0.86 represent the average limit above which communities stop being clearly defined.

μ and low standard-deviation (especially for μ>0.7).

In conclusion to this section, we can state EV and BA are slightly above CM in terms of realism. All three of them have extremely similar results on the average distance. They have significantly different transitivity, but all three are realistic, at least for small μ values. Concerning the degree correlation, both BA and EV exhibit realistic values for any μ, whereas CM is realistic only for μ≈0. The main difference between the reviewed generative models is related to their sensitivity to μ. CM is clearly the most sensitive, showing the largest range of values for both transitivity and degree correlation, whereas BA is the most stable. However, EV generally has highest values than BA, so it is difficult to decide which one is the most adapted. The next section will be dedicated to study how these differences in stability and realism translate in terms of community detection performances.

B. Community Detection Performances

We applied the four community detection algorithms presented in section II on all the networks we generated. We compared the performances using the normalized mutual information (NMI), which was used for this purpose in previous studies [1, 11, 12, 21].

Fig. 2 shows the results for LP, LV and WT, in function of μ. FG was omitted because its performances do not vary significantly depending on the generative model (see [22] for more details). Although we applied the algorithms on networks with average degree $\langle k \rangle$=15 and 30, there was no relevant difference between the results: the performances were uniformly slightly better for 30 than for 15. Consequently, our plots show only the former. Generally, as expected from previous studies [1, 11, 12], the accuracy of all algorithms decreases along μ increases, i.e. communities become more mixed and difficult to distinguish. When μ≈0, all algorithms manage to successfully identify communities, whereas when μ>0.86, they all perform badly. The way the performance evolves in function of μ depends on the algorithm, though. It is almost linear for FG, which has poor performances even for values of μ far from the mixing limit. For the other algorithms, the performance stays close to the maximum until some individual limit is reached, at which point a sudden drop occurs. This individual limit is very close to 0.7 for LV and WT, whereas it is around 0.5 for LP. The main differences between LV and WT are the former's performance slightly decreases before suddenly dropping off, whereas the latter's stays maximal; and LV performance are bellow WT's when μ≈1. So a clear hierarchy appears between algorithms, in terms of general accuracy: FG<LP<LV<WT.

The effect of the generative model on community detection performance depends on the considered algorithm. FG does not seem to be sensitive at all, which suggests the information it uses to identify communities is not related at all to transitivity nor degree correlation. FG essentially applies a modularity optimization approach, so on the one hand, this raises a question regarding the sensitivity of modularity to these properties. On the other hand, FG is not the best algorithm for modularity optimization, plus LV, which is also modularity-based, shows signs of sensitivity to the model. LP, which is not modularity-based, is much more sensitive to the generative model. EV and BA provoke close low drop-off limits, around 0.4 and 0.5, whereas it is approximately 0.7 for CM. However, note these values may not precisely represent the actual performance, due to the high variance observed in LP results. LP performance is far better for CM than for the other models, which could suggest it finds more realistic networks harder to process. More precisely, the way models are ordered in terms of performance is the exact opposite of their order in terms of degree correlation. We suppose LP does not handle well networks with positive degree correlation, maybe because such a property modifies the way labels spread in the network.

As stated before, LV is modularity-based but, unlike FG, it performs differently depending on the model. WT does not rely on modularity to identify communities, but generally uses it as a criterion to select the best cut in the output dendrogram. Both algorithms do not show any model-sensitiveness until they reach their drop-off limit. Then performances are clearly better for CM and EV than for BA. In the case of WT, CM leads to even higher performances than EV on the range 0.55-0.75. This order fits with the models transitivity, so we could assume LV performs better when this property is high enough. However, EV transitivity is higher than CM's for μ>0.3 and this does not appear at all in the performance plot. On the contrary, the performance for CM stays above the other models until 0.8, whereas its transitivity is roughly the same.

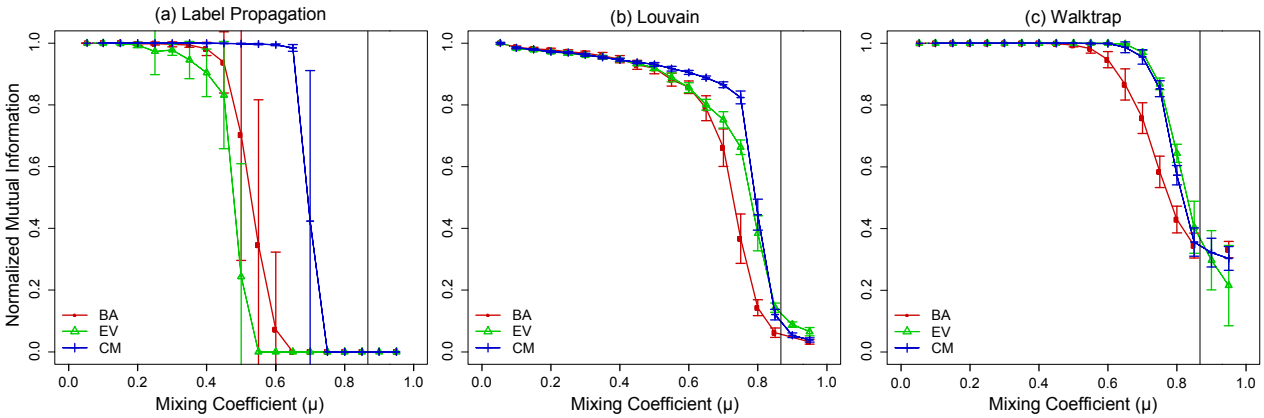

Figure 2. Community detection performances in function of the mixing coefficient μ, for the LP (a), LV (b) and WT (c) algorithms. The networks are the same than in Fig.1 (n=5000, γ≈3, β=2 and $\langle k \rangle$≈30). Each point corresponds to an average over 25 processed networks. The vertical lines at μ=0.86 represent the average limit above which communities stop being clearly separated. Performances are expressed in terms of normalized mutual information.

The compared algorithms use different principles and mechanisms to identify communities, which can explain why their performances are influenced in various ways by the studied generative models. However, if we do not take FG into account, it generally appears the Barabási–Albert model is the most difficult to process, whereas the configuration model is associated to the highest results. The evolutionary preferential attachment model lies somewhere in between (LV), sometimes closer to the former (LP) and sometimes closer to the latter (WT). Drawing more solid conclusions will necessitate further experiments possibly involving additional community detection algorithms and network properties. However, for now, we would say results measured on the BA-based LFR method are the most reliable, because of the stability of the generated networks properties to changes in μ.

IV. Conclusion

In this paper, we proposed a modification of the LFR model designed by Lancichinetti *et al*. [1], aiming at improving the realism of the networks it generates. It consists in replacing the configuration model (CM) LFR relies on by the Barabási–Albert (BA) [17] and evolutionary preferential attachment (EV) [18] models, which are known to produce more realistic networks. Our modification allows producing networks with comparable average distance, more realistic and stable degree correlation and more stable transitivity, compared to the original LFR method. For these properties, EV exhibits slightly better absolute values but BA is more stable.

In order to study the effect of our modification on the community detection process, we applied four different algorithms on the generated collections: Fast Greedy (FG) [6], Label Propagation (LP) [19], Louvain (LV) [20] and Walktrap (WT) [7]. For all algorithms and on all networks, the performances decrease when the mixing coefficient μ increases, as observed in previous studies [1, 11, 12]. LP, LV, and WT show significant changes in their performances depending on the considered generative model (CM, BA or EV), whereas FG is not sensitive at all. For the three sensitive algorithms, the highest performances are obtained when applied to CM, the lowest correspond to BA, and the results on EV networks depend on the considered algorithm. We could not determine if the observed changes in performance were due to some property in particular, though. BA seems to be the most interesting model in terms of discrimination of the community detection algorithms, because its stability to changes in μ allows to consistently compare performances for different levels of separation of the communities.

Our goal was to improve the realism of the networks generated by the LFR method, and from this point of view the modifications were efficient. But they also resulted in a loss of control on some other network properties, and the improvements were not as strong as expected. This could be solved by using other realistic generative models to replace the CM. It would also be interesting to assess how much the generated network properties depend on the initial generative model and on the LFR rewiring step itself. Concerning the effect of realism on community detection algorithms, our work can be extended in two ways. First, it could be generalized by applying other algorithms relying on different community detection approaches to the generated networks. Second, we could consider other properties to characterize networks. Maybe we did not find any strong relationships between the generated networks properties and the performance changes because we did not focus on the relevant properties.

References

[1] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," Phys Rev E, vol. 78, p. 046110, 2008.

[2] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," Phys Rev E, vol. 69, 2004.

[3] S. Fortunato, "Community detection in graphs," Physics Reports, vol. 486, pp. 75-174, Feb 2010.

[4] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," PNAS, vol. 99, pp. 7821-7826, 2002.

[5] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," Phys Rev E, vol. 72, p. 027104, 2005.

[6] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," Phys Rev E, vol. 70, 2004.

[7] P. Pons and M. Latapy, "Computing communities in large networks using random walks," arXiv, physics/0512106, 2005.

[8] P. Erdos and A. Renyi, "On random graphs," Publicationes Mathematicae, vol. 6, pp. 290-297, 1959.

[9] L. Danon, A. Diaz-Guilera, and A. Arenas, "The effect of size heterogeneity on community identification in complex networks," J Stat Mech, p. 11010, 2006.

[10] J. P. Bagrow, "Evaluating local community methods in networks," Journal of Statistical Mechanics-Theory and Experiment, 2008.

[11] G. K. Orman and V. Labatut, "A Comparison of Community Detection Algorithms on Artificial Networks," Lecture Notes in Artificial Intelligence, vol. 5808, pp. 242–256, Oct 2009.

[12] A. Lancichinetti and S. Fortunato, "Community detection algorithms: a comparative analysis," Phys Rev E, vol. 80, 2009.

[13] M. Molloy and B. Reed, "A critical point for random graphs with a given degree sequence," Random Structures and Algorithms, vol. 6, pp. 161-179, 1995.

[14] M. E. J. Newman, "The structure and function of complex networks," SIAM Review, vol. 45, pp. 167-256, 2003.

[15] L. da Fontura Costa, O. N. Oliveira Jr., G. Travieso, r. A. Rodrigues, P. R. Villas Boas, L. Antiqueira, *et al.*, "Analyzing and Modeling Real-World Phenomena with Complex Networks: A Survey of Applications," arXiv, 0711.3199, 2008.

[16] M. Serrano and M. Boguñá, "Weighted Configuration Model," in AIP Conference. vol. 776, 2005, p. 101.

[17] A. Barabasi and R. Albert, "Emergence of scaling in random networks," Science, vol. 286, p. 509, 1999.

[18] J. Poncela, J. Gomez-Gardeñes, L. M. Floría, A. Sanchez, and Y. Moreno, "Complex Cooperative Networks from Evolutionary Preferential Attachment," PLoS ONE, vol. 3, p. e2449, 2008.

[19] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," Phys Rev E, vol. 76, p. 036106, 2007.

[20] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," J Stat Mech, p. P10008, 2008.

[21] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," J Stat Mech, p. P09008, 2005.

[22] G. K. Orman and V. Labatut, "A Modification to Improve the Realism of Networks Generated with the LFR Model," Galatasaray University, Istanbul, TR, Technical Report 201002121, 2010.