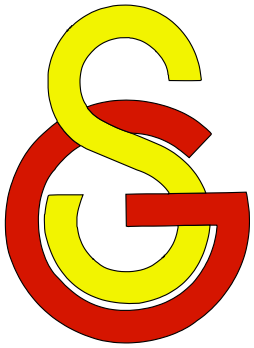
A Modification to Improve the Realism of Networks Generated with the LFR Model

Günce Keziban, Vincent Labatut

Galatasaray University

Çırağan cad. n°36, 34357 Ortaköy/İstanbul, Turkey

Engineering and Technology Faculty
Computer Science Department



Technical Report
TR201002121

A Modification to Improve the Realism of Networks Generated with the LFR Model

Günce Keziban Orman and Vincent Labatut

First version: February 12 2010
Last version: May 24 2010

A Modification to Improve the Realism of Networks Generated with the LFR Model

Günce Keziban 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 of nodes 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 algorithms testing. Recently, Lancichinetti *et al*. [1] designed a method called LFR, able 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 the LFR model, 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.

Proposition d'une modification du modèle LFR pour un meilleur réalisme des réseaux générés

Résumé. La détection de communauté consiste à rechercher des sous-ensembles de nœuds densément connectés dans des réseaux complexes. Il s'agit d'une problématique centrale pour des chercheurs issus des nombreux domaines différents dans lesquels les réseaux complexes sont utilisés comme outil de modélisation. Les algorithmes de détection de communauté sont généralement testés à la fois sur des réseaux réels et artificiels. Les premiers sont plus difficiles et coûteux à obtenir, tandis que le niveau de réalisme des derniers a un effet direct sur la fiabilité des tests. Récemment, Lancichinetti *et al*. [1] ont défini un modèle appelé LFR, qui permet de produire des réseaux possédant une structure de communauté, ainsi que des degrés et des tailles de communautés distribués selon une loi de puissance. Cependant, d'autres propriétés observées dans un grand nombre de réseaux réels sont manquantes, telles qu'une corrélation de degré non-nulle et une transitivité élevée. Dans cet article, nous proposons une modification du modèle LFR basée sur le modèle d'attachement préférentiel, afin de résoudre cette limitation. Nous analysons les propriétés des réseaux générés et les comparons à ceux obtenus avec la méthode originale. Nous appliquons ensuite différents algorithmes de détection de communauté à ces réseaux et observons des modifications significatives dans leurs performances, comparées à celles obtenus sur les réseaux issus de LFR.

Mots-clés: Réseaux complexes, Détection de communauté, Réseaux aléatoires, Génération de réseaux, Propriétés des réseaux.

1 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. Thanks to this versatility, they became very popular during the last decades, attracting the attention of scientists from many different domains, and are now used to model and analyze complex systems in areas like physics, biology, social sciences or computer science [2, 3]. Among the various approaches used to study complex networks properties, community detection has become one of the most popular ones. A community in a complex network is a cohesive subset of nodes with denser inner links, relatively to the rest of the network [4]. Because of the spread of complex networks, the community detection problem has been studied in many different areas too, resulting in tens of algorithms based on a whole range of principles: hierarchical clustering, optimization methods, graph partitioning, spectral properties of the network, etc.[5]

Until recently, those algorithms were tested on a few real and/or artificial networks. Limiting these tests to real networks can be considered as an issue for several reasons. First, building such networks is a costly and difficult task, and determining reference communities can only be done by experts. This leads to small networks, where actual communities are not always defined objectively, or even known. Second, a complex network is characterized by various properties like its average degree, degree distribution, shortest average path, etc. By definition, it is not possible to control these features in a real network. This means the algorithm is tested only on a very specific and limited set of features. Hopefully, artificial networks allow overcoming these limitations. First, by using some generative model, it is possible to automatically build many of them. Moreover, since they are generated, objective reference communities are defined. Finally, depending on the selected generative model, various properties can be directly controlled. The only point of concern with artificial networks is their level of realism. Indeed, for algorithm testing to be relevant, the generated networks must exhibit realistic properties. For this purpose, generative models are generally defined in order to mimic known real networks properties. Of course, current knowledge regarding these properties may not be exhaustive, and we can consequently never be completely assured the generated networks are perfectly realistic. For this reason, tests on artificial networks should be seen as complementary to tests on real networks.

A few methods have been designed to generate networks with a community structure. The most popular one is certainly the model by Girvan and Newman (GN) [6]. First, an empty (i.e. without any link) network is created, then nodes are randomly assigned equal-sized virtual communities, and finally links are randomly drawn with probabilities p_{in} and p_{out} for intra- and inter-community links, respectively. This results in a set of interconnected Erdős-Rényi networks [7]. Several extensions were derived to generate weighted or oriented links, and hierarchical or

overlapping communities (see [5] for more details). 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. Although widely used to test and compare these algorithms [6, 8-10], the GN method is limited in terms of realism [1]: the generated networks are rather small compared to most real world networks studied in the literature [11]; all nodes have roughly the same degree; and all communities have the same size. Yet, it is well known real networks exhibit power law or exponential degree distribution [2, 11], and community sizes generally follow a power law [11, 12]. To tackle this problem, several GN variants were defined, producing bigger networks, and communities with heterogeneous sizes[5, 13, 14]. More recently, a different approach appeared, based on rewiring [1, 15]. First an initial network with desired properties (but no community structure) is randomly generated, then virtual communities are drawn, and finally some links are rewired so that these communities appear in the network, but without changing the existing degree sequence. The method described by Bagrow [15] uses the Barabási–Albert [16] model to generate the initial network, resulting in a power law degree distribution, but produces small networks with equal-sized communities. The method by Lancichinetti *et al*. (LFR) [1] is based on the configuration model [17], which generates networks with power law degree distribution, too. However, unlike Bagrow's method, LFR generates power law distributed community sizes, and the network size is not constrained. In other terms, LFR exhibits the most realistic properties among the presented generative approaches. Nevertheless, it also has some noticeable limitations regarding the low transitivity and close to zero degree correlation measured in the generated networks [18]. According to Newman [2], real world networks usually have a clearly non-zero degree correlation, and their transitivity, or clustering coefficient, is relatively high.

Interestingly, improvement on the realistic aspect of the generated networks has a noticeable effect on most community detection algorithms. A performance drop was observed when authors switched from equal-sized communities to heterogeneous distributions[13, 14]. This allowed to show one algorithm may not perform on the same level depending on the size of the considered communities. The introduction of a power law degree distribution also made the benchmarks more discriminatory, allowing to highlight differences between algorithms whose performances were considered similar before [1]. The fact Lancichinetti *et al*.'s method, which is the most realistic one to date, 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 section 2, we describe briefly the LFR method, its characteristics and the improvements 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 3, we present the properties of the networks generated with the modified method, and use them to compare the community detection algorithms performances. Finally, in section 4 we comment these results and propose some further improvements.

2 Methods

2.1 LFR Generative Method

The LFR method 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 method was subsequently extended to generate weighted and/or oriented networks, with possibly overlapping communities [19, 20], but here we focus on the first version.

This method 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 $1-\mu$. It is generally not possible to meet this constraint exactly, and the mixing coefficient is therefore only approximated in practice. It is an important parameter, because it determines how clearly the communities are defined in terms of structure. For small μ values, the communities are distinctly separated, whereas for high values, the network has almost no community structure, making community identification a difficult task. Intuitively, one may think the community structure remains clear for $\mu < 0.5$, because this value corresponds to a network where, in average, nodes have the same number of inter- and intra-community links. In practice though, some algorithms are able to successfully detect communities even when μ approaches 0.8 [18, 20]. Let us note p_{in} (resp. p_{out}) the ratio of existing to possible intra- (resp. inter-) community links, or in other terms: the probability to have a link between two nodes belonging to the same community (resp. different communities). Then the communities are well-defined when $p_{in} > p_{out}$, which translates to $\mu < (n - n_c^{max})/n$, where n and n_c^{max} are the number of nodes in the network and in the biggest community, respectively [20].

The LFR method is three-stepped. First, it uses the configuration model [17] 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, so that μ is approximated, but without changing the degree distribution.

2.2 LFR Properties

By construction, the LFR method guaranties to obtain several realistic properties: size of the network, power law distributed degrees and community sizes. Moreover, some parameters give the user a direct control on these properties: network size (n),

degree distribution (γ, k_{max}, $\langle k \rangle$), community structure (β, μ). For these reason, we call them controlled properties.

However, real world networks are known to exhibit additional properties. They have the small world property [21], which states that for a fixed average degree, the average distance (i.e. the length of the shortest path) between pairs of nodes increases logarithmically (or slower) with the number of nodes n [2]. This property is important, because it is related to the network efficiency to propagate information. Another property of interest is related to the transitivity coefficient (also called clustering coefficient [21]), which assesses the density of triangles (three completely connected nodes) in the network. The higher the transitivity, the more probable it is to observe a link between two nodes both connected to a third one. A real network is supposed to have a higher transitivity than a random Erdős–Rényi network [7] possessing the same number of nodes and links, by a factor of order n [2]. Finally, degree correlation constitutes another interesting property, by describing how a node degree is related to its neighbors. Real networks usually show a non-zero degree correlation. If it is positive (resp. negative), the network is said to be assortatively (resp. disassortatively) mixed [2]. According to Newman, social networks tend to be assortatively mixed, while other kinds of networks are generally disassortatively mixed. Degree correlation is related to the concepts of hubs and authorities, which are specific nodes with central positions.

In the context of this work, the purpose of network generation is to compare algorithms which completely rely on the networks structure to identify communities. These three additional properties are hence particularly important, in the sense they are all related to the network structure. The LFR method does not allow controlling them directly, but these uncontrolled properties were analyzed on a wide range of parameters values [18]. At this occasion, it was shown LFR generates small world networks, with relatively high transitivity and degree correlation, but only under certain circumstances. Indeed, uncontrolled properties are affected by changes in certain parameters, and this is especially true for the mixing coefficient. All three properties exhibit realistic values when μ is almost zero, but when it gets closer to 1, one can observe strong decreases, resulting in unrealistic transitivity and degree correlation values. In all studies evaluating community detection algorithms, performances are assessed in function of some index representing the communities level of separation [1, 5, 6, 8-10, 13-15, 18]. In the case of the LFR method, this index is μ, so its influence on some structural property can be a serious limitation. Indeed, when comparing some algorithm performances on networks generated with different μ, one generally considers the only difference between the networks is how much communities are separated. But μ also affects other properties, possibly influencing the observed performance.

2.3 Proposed Modifications

One of the possible causes for the observed unrealistic properties is the use of the configuration model (CM) [17] 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 [22] and low transitivity (when degrees are power law distributed) [2]. We propose to use a different generative model, with more realistic properties. We considered the Barabási–Albert *preferential attachment* model [23] and one of its variants called *evolutionary preferential attachment* [24]. The rest of the method 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 Barabási–Albert preferential attachment model (BA) [16] was designed as an attempt to explain the power law degree distribution observed in real networks by the building process of these networks. Starting from an initial network containing m_0 connected nodes, a realistic iterative process is applied to simulate growth. At each iteration, one node is added to the network, and is randomly connected 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: $P(k_i) = k_i / \sum_j k_j$. In other terms: the higher a node degree, the higher its chances of being selected. This so-called preferential attachment mechanism results in a power law degree distribution, since degree increases faster for nodes with higher degree, as new nodes are added to the network. The exponent cannot be controlled though, and tends towards 3 [16]. The average distance is always less than in same-sized Erdős-Rényi networks, so it has the small world property [23]. The average degree depends directly on the m parameter: $\langle k \rangle = 2m$ [2]. Transitivity is greater than in Erdős-Rényi networks, but nevertheless decreases with network size following a power law $\sim n^{-0.75}$ [23].

The evolutionary preferential attachment (EV) [24] model is a variant of the BA model. It also uses the preferential attachment and growth mechanisms, except the attachment probabilities are not based on some topological properties, like the current degree in the case of BA, but on some nodal dynamic property, updated using the 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 unilateral cooperation or bilateral defection, 1 for bilateral cooperation, and b for unilateral defection, with $b > 1$. The first move is randomly chosen, whereas the next one depends on the respective results of the considered node and a randomly picked neighbor. If the neighbor's score is better, the node might switch to its strategy, with a probability depending on the difference between their scores. Nodes with higher scores are more attractive to a node added to the network, because by being connected to them, it may use a strategy which proved to be successful. According to its authors, this process is more realistic and leads to networks with high transitivity and degree correlation. Besides the parameters already needed by BA (n, m_0 and m), EV uses b (points scored for unilateral cooperation) and ε (selection pressure). The latter allows to modulate the influence of the preferential attachment mechanism: all nodes are equiprobable when $\varepsilon = 0$, whereas the nodes scores are fully considered for $\varepsilon = 1$.

2.4 Community Detection Algorithms

To study the effects of the networks realism on the community detection process, we applied four popular algorithms: Newman *et al.*'s *Fast Greedy* algorithm (FG) [25] 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) [26] 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) [27] 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) [28] is the most recent of the considered algorithms. It relies on a two-stepped hierarchical modularity optimization method. The first step consists in detecting small communities by performing a greedy optimization on modularity. In the second step, the algorithm creates a network whose nodes represent the communities identified during at step one. These two steps are repeated to build the complete hierarchy of communities.

3 Results and Discussion

3.1 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 them by using only the name of the model employed during their first step. Consequently, CM will correspond to the original LFR method, whereas BA and EV are modified versions based on the corresponding model.

We selected our parameters values based on previous experiments in artificial networks generation [1, 18] and descriptions of real world networks measurement from the literature [2, 11, 23, 29]. Some parameters are common to all three processes: we fixed the size $n = 5000$ and the power law exponent for the community sizes distribution $\beta = 2$; 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 $\gamma = 3$, $\langle k \rangle = 15; 30$ 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 than the values specified for CM. EV additionally allows controlling transitivity, and we found out score $b = 1.5$ and selection pressure $\varepsilon = 0.99$ gave the best results.

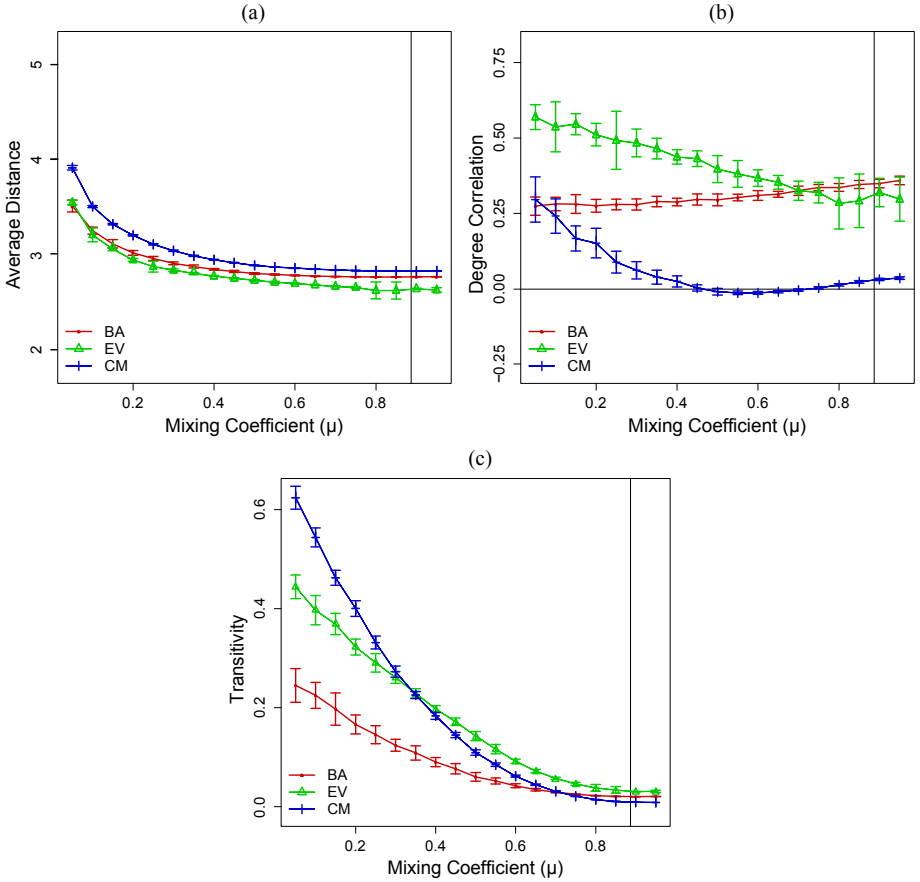


Figure 1. Influence of the mixing coefficient μ on the measured properties: (a) average distance, (b) degree correlation and (c) transitivity. Networks were generated with parameters $n = 5000$, $\gamma \approx 3$, $\beta = 2$, $\langle k \rangle \approx 30$ and using the LFR method on three different 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 $\mu = 0.86$ represent the average limit above which communities stop being clearly defined.

We produced 25 networks for each combination of parameters, and averaged the measured properties. Figure 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 2) for $\mu < 0.86$. 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 $\mu > 0.3$. 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 algorithms 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 $\mu \approx 0$, but it also has almost zero transitivity for $\mu \approx 1$, exhibiting a serious sensitiveness to μ. Other methods also show a decreasing transitivity when μ increases, but the range is much smaller, mainly because their values for $\mu \approx 0$ are significantly smaller: around 0.25 and 0.45 for BA and EV, respectively. Like CM, they reach almost zero value when $\mu \approx 1$. 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 $\mu > 0.3$. Note that in the literature, real world networks with a transitivity greater than 0.3 are considered highly transitive [11], 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, independently of the way the network is created. Testing this hypothesis would require being able to quantify the separation level of communities in real world networks (using Newman's modularity [4], for instance), in order to compare it to the measured transitivity. The authors are not aware of any work of this kind, which is also out of this paper scope.

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 $\mu > 0.4$. EV shows the highest degree correlation, with values greater than 0.5 for $\mu \approx 0$. It also decreases when μ increases, resulting in values close to 0.25 for $\mu \approx 1$. Finally, unlike other models, BA degree correlation slightly increases with μ, ranging approximately from 0.25 ($\mu \approx 0$) to 0.35 ($\mu \approx 1$). Although its values are lower than for EV, it is also

more stable both in terms of sensitivity to μ and low standard-deviation (especially for $\mu > 0.7$).

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 $\mu \approx 0$. 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 subsection will be dedicated to study how these differences in stability and realism translate in terms of community detection performances.

3.2 Community Detection Performances

We applied the four community detection algorithms presented in section 2 on all the networks we generated. These algorithms do not need the user to specify any parameter, except for WT, for which we used the default value to define the length of the random walks. We compared the performances using the normalized mutual information (NMI). This measure was defined in the context of conventional clustering to compare two different partitions of one data set [3, 30] and was later used to assess community detection performance [1, 19, 20, 31].

Figure 2 shows the results for all four algorithms, in function of the mixing coefficient μ. 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, 18-20], the accuracy of all algorithms decreases along μ increases, i.e. communities become more mixed and difficult to distinguish. When $\mu \approx 0$, all algorithms manage to successfully identify communities, whereas when the mixing limit of $\mu > 0.86$ is reached, 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 $\mu \approx 1$. 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 strongly on the considered algorithm. FG does not seem to be sensitive at all, since its performances for all three models are not significantly different. This

suggests the information it uses to identify communities is not related at all to transitivity nor degree correlation. FG is essentially based on 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. More efficient modularity optimization approaches such as Spinglass [32] could be applied on the same networks to verify this hypothesis (the authors could not achieve this task by lack of time, Spinglass being extremely long compared to other algorithms, especially those presented here).

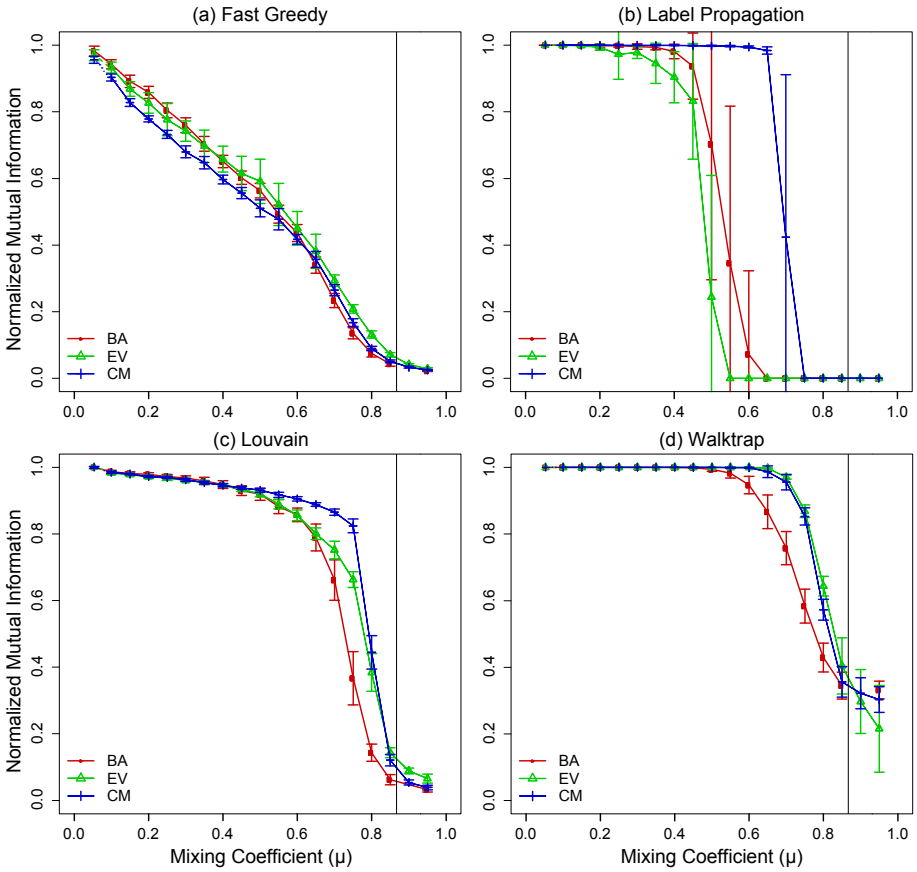


Figure 2. Community detection performances in function of the mixing coefficient μ, for (a) Fast Greedy, (b) Label Propagation, (c) Louvain and (d) Walktrap algorithms. The networks are the same than in Fig.1 ($n = 5000$, $\gamma \approx 3$, $\beta = 2$ and $\langle k \rangle \approx 30$). Each point corresponds to an average over 25 processed networks. The vertical lines at $\mu = 0.86$ represent the average limit above which communities stop being clearly defined. Performances are expressed in terms of normalized mutual information.

LP, which is not modularity-based, is much more sensitive to the generative model. EV and BA have 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. This could mean LP does not handle well networks with positive degree correlation, maybe because such a property modifies the way labels spread in the network. But this observation is difficult to confirm here. Indeed, for $\mu > 0.7$ BA has a higher correlation degree than EV, but the effect on performance cannot be discussed because NMI has already reached 0 for these values of μ.

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 it is generally used as a criterion to select the best partition (or cut) in the hierarchy it outputs (called 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 $0.55 \sim 0.75$. 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 $\mu > 0.3$ 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.

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). It remains difficult to explain exactly why in general we observed differences in performances, because the data seem to be incomplete for that matter. Drawing more solid conclusions would necessitate first applying other algorithms, if possible using a wide range of principles to detect communities. Additionally, one could consider studying other network properties, possibly responsible for the changes in performance. However, for now, we would say results measured on the BA-based LFR method are the more reliable, because of the stability of generated networks properties to changes in the mixing coefficient μ.

4 Conclusion

In this paper, we proposed an improvement for the LFR method designed by Lancichinetti *et al*. [1] to randomly produce realistic complex networks with community structure. LFR uses the configuration model (CM) [17] to randomly generate a network with power law distributed degree, and then rewires it partially

to make a community structure appear. An important parameter called mixing coefficient allows controlling the level of separation between communities. This type of method is used to benchmark community detection algorithms on huge collections of artificial networks, before testing them on real world ones [1, 18, 19]. For this reason, their realistic aspect is of utmost importance and is known to affect the algorithms performances [13]. To our knowledge, LFR is currently the method generating the most realistic networks, but it has some limitations regarding certain properties [18]: degree correlation and transitivity (a.k.a. clustering) are not realistic and are not stable to changes in the mixing coefficient. Our improvement consists in replacing the configuration model by the Barabási–Albert (BA) [16] and evolutionary preferential attachment (EV) [24] models, which are known to produce more realistic networks regarding these properties. We generated several collections of networks with the LFR method, based successively on CM, BA and EV. We used realistic values for all the directly controlled properties (number of nodes, average degree and power law exponent) and compared the values of the uncontrolled ones (average distance, transitivity and degree correlation). We found out our improvement allows producing networks with comparable average distance, more realistic and stable degree correlation and more stable transitivity, compared to the original LFR method (using CM). For these properties, EV exhibits 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) [25], Label Propagation (LP) [27], Louvain (LV) [28] and Walktrap (WT) [14]. We assessed their performances thanks to the normalized mutual information measure, already used for the same purpose before [1, 19, 20, 31]. For all algorithms and on all networks we observed the usual decrease in performance caused when increasing the mixing coefficient [1, 19, 20]. Moreover, three algorithms out of four showed significant changes in their performances depending on the considered generative model (CM, BA or EV). FG is not sensitive at all, but has the poorest results and seems a bit out of date compared to more recent algorithms. Globally, for the three sensitive algorithms, the highest performances are obtained when applied to CM, whereas the lowest correspond to BA. The results on EV networks depend on the considered algorithm: close to BA for LP, close to CM for WT, and somewhere in between for LV. 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 the mixing coefficient 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 we proposed were efficient. But they also resulted in a loss of control, since the replacement models (BA and EV) do not allow to specify directly as many properties as CM. For instance, in CM the exponent of the degree power law distribution is a parameter, whereas BA can

generate only networks with an exponent 3 (which, hopefully, is an extremely realistic value). We suppose it is one of the reasons why Lancichinetti *et al*. chose to base their approach on this model in the first place. Moreover, the improvement was not as strong as expected, especially concerning the transitivity, which is still very sensitive to changes in the mixing coefficient. Different ways can be explored to try to solve these limitations. First, it would be interesting to study the side effects of the rewiring process used in the LFR approach, by simply comparing the generated networks properties before and after the wiring step. This work is necessary to know if some properties observed in the final networks depend on the initial (pre-rewiring) network or on the rewiring process itself. Second, many other models exist to generate networks with power law distributed degree [33-40]. A systematic review could allow detecting more flexible models, offering more control on the generated networks properties, and more realistic properties. It is a long and difficult task though, because source codes are rarely easily available.

Concerning the effect of realism on community detection algorithms, our work can be extended in two ways. First, other algorithms could be applied to networks generated with the modified LFR method. Indeed, the four algorithms we compared exhibit rather different reactions to the models we used, making it difficult to infer general remarks concerning their effects on community detection performance. For example, we observed FG, which is a modularity-based algorithm, was not sensitive to the selected generative model, whereas LV, which optimizes modularity too, is. By considering algorithms such as *Spinglass* [32], which is known to be a good estimator of maximal modularity, we could study the influence of the generative model on modularity. 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. We chose the most widely used ones in the context of real networks analysis, but many others exist [41, 42], even if they have not been so popular up to now.

References

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

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

[3] A. L. N. Fred and A. K. Jain, "Robust Data Clustering," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition. vol. 2: IEEE Computer Society, 2003, pp. 128-136.

[4] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," Phys Rev E, vol. 69, p. 026113, 2004.

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

[6] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," Proceedings of the National Academy of Sciences of the United States of America, vol. 99, pp. 7821-7826, Jun 11 2002.
[7] P. Erdos and A. Renyi, "On random graphs," Publicationes Mathematicae, vol. 6, pp. 290-297, 1959.
[8] L. Donetti and M. A. Munoz, "Detecting network communities: a new systematic and efficient algorithm," J Stat Mech, p. P10012, 2004.
[9] J. Duch and A. Arenas, "Community detection in complex networks using extremal optimization," Phys Rev E, vol. 72, p. 027104, 2005.
[10] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," PNAS USA, vol. 101, pp. 2658-2663, Mar 2 2004.
[11] 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, vol. 0711.3199, 2008.
[12] R. Guimerà, L. Danon, A. Díaz-Guilera, F. Giralt, and A. Arenas, "Self-similar community structure in a network of human interactions," Phys Rev E, vol. 68, p. 065103, 2003.
[13] L. Danon, A. Diaz-Guilera, and A. Arenas, "The effect of size heterogeneity on community identification in complex networks," J Stat Mech, p. 11010, Nov 2006.
[14] P. Pons and M. Latapy, "Computing communities in large networks using random walks," arXiv, vol. physics/0512106, 2005.
[15] J. P. Bagrow, "Evaluating local community methods in networks," Journal of Statistical Mechanics-Theory and Experiment, pp. -, May 2008.
[16] A. Barabasi and R. Albert, "Emergence of scaling in random networks," Science, vol. 286, p. 509, 1999.
[17] 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.
[18] 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.
[19] A. Lancichinetti and S. Fortunato, "Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities," Phys. Rev. E, vol. 80, p. 016118, 2009.
[20] A. Lancichinetti and S. Fortunato, "Community detection algorithms: a comparative analysis," Phys. Rev. E, vol. 80, p. 056117, 2009.
[21] D. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," Nature, vol. 393, pp. 409-410, 1998.
[22] M. Serrano and M. Boguñá, "Weighted Configuration Model," in AIP Conference. vol. 776, 2005, p. 101.
[23] A. Barabasi and R. Albert, "Statistical mechanics of complex networks," Reviews of Modern physics, vol. 74, pp. 47-96, 2002.
[24] 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.
[25] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," Physical Review E, vol. 70, pp. -, Dec 2004.
[26] P. Pons and M. Latapy, "Computing communities in large networks using random walks," Computer and Information Sciences - Iscis 2005, Proceedings, vol. 3733, pp. 284-293, 2005.

[27] 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.
[28] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," Journal of Statistical Mechanics-Theory and Experiment, pp. -, Oct 2008.
[29] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D. Hwang, "Complex networks: structure and dynamics," Physics Reports, vol. 424, pp. 175-308, 2006.
[30] L. I. Kuncheva and S. T. Hadjitodorov, "Using diversity in cluster ensembles," in IEEE International Conference on Systems, Man & Cybernetics. vol. 1-7, 2004, pp. 1214-1219.
[31] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," J Stat Mech, p. P09008, 2005.
[32] J. Reichardt and S. Bornholdt, "Statistical mechanics of community detection," Phys Rev E, vol. 74, p. 016110, 2006.
[33] Q. Chen and S. Chen, "A highly clustered scale-free network evolved by random walking," Physica A, vol. 383, pp. 773-781, 2007.
[34] L. Chuang, J. Jian-Yuan, C. Xiao-Jie, C. Rui, and W. Long, "Prisoner's Dilemma Game on Clustered Scale-Free Networks under Different Initial Distributions," Chin. Phys. Lett., vol. 26, p. 080202, 2009.
[35] C. Dangalchev, "Generation models for scale-free networks," Physica a-Statistical Mechanics and Its Applications, vol. 338, pp. 659-671, Jul 15 2004.
[36] P. Holme and B. J. Kim, "Growing scale-free networks with tunable clustering," Phys Rev E, vol. 65, p. 026107, 2002.
[37] K. Klemm and V. M. Eguiluz, "Growing scale-free networks with small-world behavior," Phys Rev E, vol. 65, 2002.
[38] J. Saramäki and K. Kaski, "Scale-free networks generated by random walkers," Physica A, vol. 341, pp. 80-86, 2004.
[39] D. Shi, X. Zhu, and L. Liu, "A Simple Model of Scale-free Networks Driven by Both Randomness and Adaptability," arXiv, vol. physics/0409061v2, 2004.
[40] W. M. Tam, F. C. M. Lau, and C. K. Tse, "Construction of Scale-Free Networks with Adjustable Clustering," in International Symposium on Nonlinear Theory and its Applications (NOLTA) Budapest, Hungary, 2008.
[41] L. da Fontura Costa, F. A. Rodrigues, G. Travieso, and P. R. Villas Boas, "Characterization of complex networks: A survey of measurements," Advances in Physics, vol. 56, pp. 167-242, 2007.
[42] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon, "Network motifs: simple building blocks of complex networks," Science, vol. 298, pp. 824-827, 2002.