Overlapping Community Detection in Networks: The State-of-the-Art and Comparative Study

JIERUI XIE, Rensselaer Polytechnic Institute
STEPHEN KELLEY, Oak Ridge National Laboratory
BOLESLAW K. SZYMANSKI, Rensselaer Polytechnic Institute

43

This article reviews the state-of-the-art in *overlapping* community detection algorithms, quality measures, and benchmarks. A thorough comparison of different algorithms (a total of fourteen) is provided. In addition to community-level evaluation, we propose a framework for evaluating algorithms' ability to detect overlapping nodes, which helps to assess overdetection and underdetection. After considering community-level detection performance measured by normalized mutual information, the Omega index, and node-level detection performance measured by F-score, we reached the following conclusions. For low overlapping density networks, SLPA, OSLOM, Game, and COPRA offer better performance than the other tested algorithms. For networks with high overlapping density and high overlapping diversity, both SLPA and Game provide relatively stable performance. However, test results also suggest that the detection in such networks is still not yet fully resolved. A common feature observed by various algorithms in real-world networks is the relatively small fraction of overlapping nodes (typically less than 30%), each of which belongs to only 2 or 3 communities.

Categories and Subject Descriptors: A.1 [**General Literature**]: Introductory and Survey; I.5.3 [**Clustering**]: Clustering—*Algorithms*; H.3.3 [**Clustering**]: Information Search and Retrieval—*Clustering*; E.1 [**Data**]: Data Structures—*Graphs and networks*

General Terms: Algorithms, Performance

Additional Key Words and Phrases: Overlapping community detection, social networks

ACM Reference Format:
Xie, J., Kelley, S., and Szymanski, B. K. 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. ACM Comput. Surv. 45, 4, Article 43 (August 2013), 35 pages.
DOI: http://dx.doi.org/10.1145/2501654.2501657

The work of J. Xie and B. K. Szymanski was supported in part by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053 and by the Office of Naval Research Grant No. N00014-09-1-0607. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies either expressed of implied of the Army Research Laboratory, the Office of Naval Research, or the U.S. Government.
Authors' addresses: J. Xie (corresponding author), Department of Computer Science, Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180; email: jierui.xie@gmail.com; S. Kelley, Oak Ridge National Laboratory, Oak Ridge, TN 37831; B. K. Szymanski, Department of Computer Science, Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180.

DOI: http://dx.doi.org/10.1145/2501654.2501657

1. INTRODUCTION

Community or modular structure is considered to be a significant property of real-world social networks as it often accounts for the functionality of the system. Despite the ambiguity in the definition of *community*, numerous techniques have been developed for both efficient and effective community detection. Random walks, spectral clustering, modularity maximization, differential equations, and statistical mechanics have all been used previously. Much of the focus within community detection has been on identifying *disjoint* communities. This type of detection assumes that the network can be partitioned into dense regions in which nodes have more connections to each other than to the rest of the network. Recent reviews on disjoint community detection are presented in Danon et al. [2005], Lancichinetti and Fortunato [2009], Leskovec et al. [2010], and Fortunato [2010].

However, it is well-understood that people in a social network are naturally characterized by multiple community memberships. For example, a person usually has connections to several social groups like family, friends, and colleagues; a researcher may be active in several areas. Further, in online social networks, the number of communities an individual can belong to is essentially unlimited because a person can simultaneously associate with as many groups as he wishes. This also happens in other complex networks such as biological networks, where a node might have multiple functions. In Kelley et al. [2011] and Reid et al. [2011], the authors showed that the *overlap* is indeed a significant feature of many real-world social networks.

For this reason, there is growing interest in overlapping community detection algorithms that identify a set of clusters that are not necessarily disjoint. There could be nodes that belong to more than one cluster. In this article, we offer a review on the state-of-the-art in this area.

2. PRELIMINARIES

In this section, we present basic definitions that will be used throughout the article. Given a network or graph $G = \{E, V\}$, V is a set of n nodes and E is a set of m edges. For dense graphs $m = O(n^2)$, but for sparse networks $m = O(n)$. The network structure is determined by the $n \times n$ adjacency matrix A for unweighted networks and weight matrix W for weighted networks. Each element A_{ij} of A is equal to 1 if there is an edge connecting nodes i and j; and it is 0 otherwise. Each element w_{ij} of W takes a nonnegative real value representing strength of connection between nodes i and j.

In the case of overlapping community detection, the set of clusters found is called a *cover* $C = \{c_1, c_2, \ldots, c_k\}$ [Lancichinetti et al. 2009], in which a node may belong to more than one cluster. Each node i associates with a community according to a *belonging factor* (i.e., soft assignment or membership) $[a_{i1}, a_{i2}, \ldots, a_{ik}]$ [Nepusz et al. 2008], in which a_{ic} is a measure of the strength of association between node i and cluster c. Without loss of generality, the following constraints are assumed to be satisfied

$$0 \le a_{ic} \le 1 \quad \forall i \in V, \forall c \in C \tag{1}$$

and

$$\sum_{c=1}^{|C|} a_{ic} = 1,$$

where $|C|$ is the number of clusters. However, the belonging factor is often solely a set of artificial weights. It may not have a clear or unambiguous physical meaning [Shen et al. 2009b].

In general, algorithms produce results that are composed of one of two types of assignments, *crisp* (nonfuzzy) assignment or *fuzzy* assignment [Gregory 2011]. With crisp assignment, the relationship between a node and a cluster is *binary*. That is, a node i either belongs to cluster c or does not. With fuzzy assignment, each node is associated with communities in proportion to a belonging factor. With a threshold, a fuzzy assignment can be easily converted to a crisp assignment. Most detection algorithms output crisp community assignments.

3. ALGORITHMS

In this section, algorithms for overlapping community detection are reviewed and categorized into five classes which reflect how communities are identified.

3.1. Clique Percolation

The Clique Percolation Method (CPM) is based on the assumption that a community consists of overlapping sets of fully connected subgraphs and detects communities by searching for adjacent cliques. It begins by identifying all cliques of size k in a network. Once these have been identified, a new graph is constructed such that each vertex represents one of these k-cliques. Two nodes are connected if the k-cliques that represent them share $k-1$ members. Connected components in the new graph identify which cliques compose the communities. Since a vertex can be in multiple k-cliques simultaneously, overlap between communities is possible. CPM is suitable for networks with dense connected parts. Empirically, small values of k (typically between 3 and 6) have been shown to give good results [Palla et al. 2005; Lancichinetti and Fortunato 2009; Gregory 2010]. CFinder[1] is the implementation of CPM, whose time complexity is polynomial in many applications [Palla et al. 2005]. However, it also fails to terminate in many large social networks.

CPMw [Farkas et al. 2007] introduces a subgraph intensity threshold for weighted networks. Only k-cliques with intensity larger than a fixed threshold are included into a community. Instead of processing all values of k, SCP [Kumpula et al. 2008] finds clique communities of a given size. In the first phase, SCP detects k-cliques by checking all the $(k-2)$-cliques in the common neighbors of two endpoints when links are inserted to the network sequentially in order of decreasing weights. In the second phase, the k-community is detected by finding the connected components in the $(k-1)$-clique projection of the bipartite representation, in which one type of node represents a k-clique and the other denotes a $(k-1)$-clique. Since each k-clique is processed exactly twice, the running time grows linearly as a function of the number of cliques. SCP allows multiple weight thresholds in a single run and is faster than CPM.

Despite their conceptual simplicity, one may argue that CPM-like algorithms are more like pattern matching rather than finding communities since they aim to find specific, localized structure in a network.

3.2. Line Graph and Link Partitioning

The idea of partitioning links instead of nodes to discover community structure has also been explored. A node in the original graph is called overlapping if links connected to it are put in more than one cluster.

In Ahn et al. [2010][2], links are partitioned via hierarchical clustering of edge similarity. Given a pair of links e_{ik} and e_{jk} incident on a node k, a similarity can be computed via the Jaccard index defined as

$$S(e_{ik}, e_{jk}) = \frac{|N_i \cap N_j|}{|N_i \cup N_j|},$$

[1]http://www.cfinder.org.

[2]https://github.com/bagrow/linkcomm.

where N_i is the neighborhood of node i including i. Single-linkage hierarchical clustering is then used to build a link dendrogram. Cutting this dendrogram at some threshold yields link communities. The time complexity is $O(nk_{max}^2)$, where k_{max} is the maximum node degree in the network.

Evans and Lambiotte [2009, 2010] projected the network into a weighted line graph, whose nodes are the links of the original graph. Then disjoint community detection algorithms can be applied. The node partition of a line graph leads to an edge partition of the original graph. CDAEO [Wu et al. 2010] provides a postprocessing procedure to determine the extent of overlapping. Once the preliminary partitioning on the line graph is done, for a node i with $|E_{icmin}|/|E_{icmax}|$ below some predefined threshold, where $E_{icmin(cmax)}$ is the set of edges in the community with which i has the minimum (maximum) number of connections, links in E_{icmin} of the line graph are removed. This essentially reduces node i to a single membership.

Kim and Jeong [2011] extended the map equation method (also known as Infomap [Rosvall 2008]) to the line graph, which encodes the path of the random walk on the line network under the Minimum Description Length (MDL) principle.

Line graph has been extended to clique graph [Evans 2010], wherein cliques of a given order are represented as nodes in a weighted graph. The membership strength of a node i to community c is given by the fraction of cliques containing i which are assigned to c.

Although the link partitioning for overlapping detection seems conceptually natural, there is no guarantee that it provides higher-quality detection than node-based detection does [Fortunato 2010] because these algorithms also rely on an ambiguous definition of community.

Note that a link-based extended modularity is also proposed by Nicosia et al. [2009]. This measure is built on the belonging coefficients of *links*. Let a link $l(i, j)$ connecting i to j for community c be $\beta_{l(i,j),c} = F(a_{ic}, a_{jc})$, then the expected belonging coefficient of any possible link $l(i, j)$ from node i to a node j in community c can be defined as $\beta^{out}_{l(i,j),c} = \frac{1}{|V|}\sum_{j\in V} F(a_{ic}, a_{jc})$. Accordingly, the expected belonging coefficient of any link $l(i, j)$ pointing to node j in community c is defined as $\beta^{in}_{l(i,j),c} = \frac{1}{|V|}\sum_{i\in V} F(a_{ic}, a_{jc})$. The preceding belonging coefficients are used as weights for the probability of an observed link (first term in (2)) and the probability of a link starting from i to j in the null model (second term in (2)), respectively, resulting in the new modularity defined as

$$Q_{ov}^{Ni} = \frac{1}{m}\sum_{c}\sum_{i,j\in V}\left[\beta_{l(i,j),c}A_{i,j} - \beta^{out}_{l(i,j),c}\beta^{in}_{l(i,j),c}\frac{k_i^{out}k_j^{in}}{m}\right], \tag{2}$$

where $k_i^{out(in)}$ is the number of outgoing (incoming) links of i and m is the total number of edges. Note that Q_{ov}^{Ni} depends on the link belonging coefficient $F(a_{ic}, a_{jc})$, which could be the product, average, or maximum of a_{ic} and a_{jc}.

3.3. Local Expansion and Optimization

Algorithms utilizing local expansion and optimization are based on growing a *natural* community [Lancichinetti et al. 2009] or a partial community. Most of them rely on a local benefit function that characterizes the quality of a densely connected group of nodes.

Baumes et al. [2005] proposed a two-step process. First, the algorithm RankRemoval is used to rank nodes according to some criterion. Then the process iteratively removes highly ranked nodes until small, disjoint cluster cores are formed. These cores serve as seed communities for the second step of the process, Iterative Scan (IS), that expands the cores by adding or removing nodes until a local density function cannot be improved.

The proposed density function can be formally given as

$$f(c) = \frac{w_{in}^c}{w_{in}^c + w_{out}^c},$$

where w_{in}^c and w_{out}^c are the total internal and external weight of the community c. The worst-case running time is $O(n^2)$. The quality of discovered communities depends on the quality of seeds. Since the algorithm allows vertices to be removed during the expansion, IS has been shown to produce disconnected components in some cases. For this reason, a modified version called CIS was introduced in Kelley [2009], wherein the connectedness is checked after each iteration. In the case that the community is broken into more than one part, only the one with the largest density is kept. CIS also develops a new fitness function

$$f(c) = \frac{w_{in}^c}{w_{in}^c + w_{out}^c} + \lambda e_p$$

incorporating the edge probability e_p. The parameter λ controls how the algorithm behaves in sparse areas of the network. The addition of a node needs to strike a balance between the change in the internal degree density and the change in edge density.

LFM [Lancichinetti et al. 2009] expands a community from a random seed node to form a natural community until the fitness function

$$f(c) = \frac{k_{in}^c}{\left(k_{in}^c + k_{out}^c\right)^\alpha} \tag{3}$$

is locally maximal, where k_{in}^c and k_{out}^c are the total internal and external degree of the community c, and α is the resolution parameter controlling the size of the communities. After finding one community, LFM randomly selects another node not yet assigned to any community to grow a new community. LFM depends significantly on the resolution parameter α. The computational complexity for a fixed α-value is roughly $O(n_c s^2)$, where n_c is the number of communities and s is the average size of communities. The worst-case complexity is $O(n^2)$.

MONC [Havemann et al. 2011] uses the modified fitness function of LFM

$$f(c) = \frac{k_{in}^c + 1}{\left(k_{in}^c + k_{out}^c\right)^\alpha},$$

which allows a single node to be considered a community by itself. This avoids violation of the principle of *locality*. The proposed fitness function enables MONC to find the range of αs (resolution parameter as in LFM) for which a set of nodes is locally optimal. Rather than numerical exploration of these α values, MONC calculates the next lowest value of α which results in further expansion and continues to expand the community. In the case that the natural community of a node i is a subset of another node, the analysis of i stops. In this way, MONC merges communities during processing and, as a result, uncovers the network faster than LFM.

OSLOM[3] [Lancichinetti et al. 2011] tests the statistical significance of a cluster [Bianconi et al. 2008] with respect to a global null model (i.e., the random graph generated by the configuration model [Molloy and Reed 1995]) during community expansion. To grow the current community, the r value is computed for each neighbor, which is the cumulative probability of having the number of internal connections equal or larger

[3] `http://www.oslom.org`.

than the number of connections from a neighbor into this community in the null model. If the cumulative distribution of the smallest r value is smaller than a given tolerance, it is considered to be significant and the corresponding node is added to the community. Otherwise, the second smallest r is checked and so on. OSLOM usually results in a significant number of outliers or singleton communities. The worst-case complexity in general is $O(n^2)$, while the exact complexity depends on the community structure of the underlying network being studied.

Rather than considering the original network, UEOC [Jin et al. 2011] unfolds the community of a node based on the l-step transition probability of the random walk on the corresponding *annealed* network [Newman et al. 2001], which represents an ensemble of networks. After sorting nodes according to the transition probabilities in descending order, the natural community is extracted with some proper cutoff. The dominating time complexity is for calculating the transition matrix, which is $O(ln^2)$.

OCA [Padrol-Sureda et al. 2010] is based on the idea of mapping each node to a d-dimensional vector. Each *subset* of nodes S is then defined as the sum of individual vectors in this set. The fitness function is defined as the directed Laplacian on function O, where O is the squared Euclidean length of a subset vector. Like LFM, starting from some initial seeds, OCA tries to remove or add a node that results in the largest increase in the value of the fitness function. OCA requires finding the most negative eigenvalue of the adjacency matrix.

Chen et al. [2010a] proposed selecting a node with maximal node strength based on two quantities $B(u, c)$ (called belonging degree) and the modified modularity Q_{ov} for weighted networks. Q_{ov} is defined as

$$Q_{ov}^{C} = \frac{1}{2m} \sum_{c} \sum_{i,j \in V} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \beta_{ic} \beta_{jc}, \tag{4}$$

where $\beta_{ic} = k_{ic} / \sum_{c'} k_{ic'}$ is the strength with which node i belongs to community c, and $k_{ic} = \sum_{j \in c} w_{ij}$ is the total weight of links from i into community c. $B(u, c)$ measures how tightly a node u connects to a given community c compared to the rest of the network. Given two thresholds B^U and B^L, when expanding a community c, neighboring nodes with $B(u, c) > B^U$ are included in c. For nodes with $B^L \leq B(u, c) \leq B^U$, if Q_{ov} increases after adding such a node, u is added to c. The drawbacks of this algorithm are the rather arbitrary selection of the B^U and B^L thresholds and the expensive computation of Q_{ov} whose complexity is $O(kn^2)$, where k is the number of communities.

iLCD[4] [Cazabet et al. 2010] is capable of detecting both static and temporal communities. Given a set of edges created at some time step, iLCD updates the existing communities by adding a new node if its number of second neighbors and number of robust second neighbors are greater than expected values. New edges are also allowed to create a new community if the minimum pattern is detected. Defining the similarity between two communities as the ratio of nodes in common, a merging procedure is performed to improve the detection quality if the similarity is high. iLCD relies on two parameters for adding a node and merging two communities. The complexity of iLCD is $O(nk^2)$ in general, whose precise quantity depends on community structures and its parameters.

Seeds are very important for many local optimization algorithms. A clique has been shown to be a better alternative over an individual node as a seed, serving as the basis for a wide range of algorithms. EAGLE [Shen et al. 2009a] uses the agglomerative framework to produce a dendrogram. First, all maximal cliques are found and made to be the initial communities. Then, the pair of communities with maximum similarity is

[4] http://cazabetremy.fr/Cazabet_remy/iLCD.html.

merged. The optimal cut on the dendrogram is determined by the extended modularity with a weight based on the number of overlapping memberships in Shen et al. [2009b]. Even without taking into account the time required to find all the maximal cliques, EAGLE is still computationally expensive with complexity $O(n^2+(h+n)s)$, where s is the number of maximal cliques whose upper bound is $3^{n/3}$ (i.e., theoretically exponential) [Moon and Moser 1965], and h is the number of pairs of maximal cliques which are neighbors. This paper also defines an extended modularity that uses the number of communities to which a node belongs as a weight for Q as

$$Q_{ov}^{E} = \frac{1}{2m}\sum_{c}\sum_{i,j\in c}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\frac{1}{O_i O_j}, \tag{5}$$

where O_i is the number of communities to which node i belongs. This measure is in the same form as (8), but with a coefficient defined based on the maximal clique. One may argue that they are identical as in Gregory [2011].

Similar to EAGLE, GCE[5] [Lee et al. 2010] identifies maximum cliques as seed communities. It expands these seeds by greedily optimizing a local fitness function. GCE also removes communities that are similar to previously discovered ones using distance between communities c_1 and c_2 defined as

$$1 - \frac{|c_1 \cap c_2|}{min(|c_1|, |c_2|)}.$$

If this distance is shorter than a parameter ϵ, the communities are similar. The time complexity for greedy expansion is $O(mh)$, where m is the number of edges and h is the number of cliques.

In COCD [Du et al. 2008], cores are a set of independent maximal cliques induced on each vertex. Two maximal cliques are said to be dependent if their *closeness* function is positive. This function is a product of the differences between the size of internal links between two maximal cliques and the number of links connecting nodes appearing only in one of the two maximal cliques. Once the cores are identified, the remaining nodes are attached to cores with which they have maximum connections. COCD runs in $O(C_{max} \cdot Tri^2)$ in the worst case, where C_{max} is the maximum size of the detected communities, and Tri is the number of triangles, whose lower bound is $\frac{9mn-2n^3-2(n^2-3m)^{3/2}}{27}$ [Fisher 1989] or $O(n^3)$ for a dense enough graph.

3.4. Fuzzy Detection

Fuzzy community detection algorithms quantify the strength of association between all pairs of nodes and communities. In these algorithms, a soft membership vector, or belonging factor [Gregory 2010], is calculated for each node. A drawback of such algorithms is the need to determine the dimensionality k of the membership vector. This value can be either provided as a parameter to the algorithm or calculated from the data.

Nepusz et al. [2008] modeled the overlapping community detection as a nonlinear constrained optimization problem which can be solved by simulated annealing methods. The objective function to minimize is

$$f = \sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}(\tilde{s_{ij}} - s_{ij})^2, \tag{6}$$

[5] https://sites.google.com/site/greedycliqueexpansion.

where w_{ij} denotes the predefined weight, $\tilde{s_{ij}}$ is the *prior* similarity between nodes i and j, and the similarity s_{ij} is defined as

$$s_{ij} = \sum_{c} a_{ic} a_{jc}, \tag{7}$$

where the variable a_{ic} is the fuzzy membership of node i in community c, subject to the total membership degree constraint in (1) and a nonempty community constraint. To determine the number of communities k, the authors increased the value of k until the community structure does not improve as measured by a modified fuzzy modularity, which, by weighting Q with the product of a node's belonging factor, is defined as

$$Q_{ov}^{Ne} = \frac{1}{2m} \sum_{c} \sum_{i,j \in c} \left[A_{ij} - \frac{k_i k_j}{2m} \right] a_{ic} a_{jc}, \tag{8}$$

where a_{ic} is the degree of membership of node i in the community c.

Zhang et al. [2007a] proposed an algorithm based on the spectral clustering framework [Newman 2006; White and Smyth 2005]. Given an upper bound on the number of communities k, the top $k-1$ eigenvectors are computed. The network is then mapped into a d-dimensional Euclidean space, where $d \leq k-1$. Instead of using k-means, Fuzzy C-Means (FCM) is used to obtain a soft assignment. Both detection accuracy and computation efficiency rely on the user-specified value k. With running time $O(mkh + nk^2h + k^3h) + O(nk^2)$, where m is the number of edges, n is the number of nodes, the first term is for the implicitly restarted Lanczos method, and the second term is for FCM, it is not scalable for large networks. An extended modularity that used the average of the belonging factor is also proposed as

$$Q_{ov}^{Z} = \sum_{c} \left[\frac{A(V_c', V_c')}{A(V, V)} - \left(\frac{A(V_c', V)}{A(V, V)} \right)^2 \right],$$

where V_c' is the set of nodes in a community c, w_{ij} is the weight of the link connecting nodes i and j, $A(V_c', V_c') = \sum_{i,j \in V_c'} w_{ij}(a_{ic} + a_{jc})/2$, $A(V_c', V) = A(V_c', V_c') + \sum_{i \in V_c', j \in V \setminus V_c'} w_{ij}(a_{ic} + (1 - a_{jc}))/2$, and $A(V, V) = \sum_{i,j \in V} w_{ij}$.

Due to their probabilistic nature, mixture models provide an appropriate framework for overlapping community detection [Newman and Leicht 2007]. In general, the number of mixture models is equal to the number of communities, which needs to be specified in advance. In SPAEM[6] [Ren et al. 2009], the mixture model is viewed as a generative model for the links in the network. Suppose that π_r is the probability of observing community r and community r selects node i with probability $B_{r,i}$. For each r, $B_{r,i}$ is a multinomial across elements $i = 1, 2, \ldots, n$, where n is the number of nodes. Therefore, $\sum_{i=1}^{n} B_{r,i} = 1$. The edge probability e_{ij} generated by such finite mixture model is given by

$$p(e_{ij} | \pi, B) = \sum_{r=1}^{k} \pi_r B_{r,i} B_{r,j}.$$

The total probability over all the edges present in the network is maximized by the Expectation-Maximization (EM) algorithm. As in Kim and Jeong [2011], the optimal number of communities k is identified based on the minimum description length. There is another algorithm called FOG[7] [Davis and Carley 2008] also trying to infer groups based on link evidence.

[6] http://www.code.google.com/p/spaem.

[7] http://www.casos.cs.cmu.edu/projects/ora.

Similar mixture models can also be constructed as a generative model for nodes [Fu and Banerjee 2008]. In SSDE[8] [Magdon-Ismail and Purnell 2011], the network is first mapped into a d-dimensional space using the spectral clustering method. A Gaussian Mixture Model (GMM) is then trained via the Expectation-Maximization algorithm. The number of communities is determined when the increase in log-likelihood of adding a cluster is not significantly higher than that of adding a cluster to random data which is uniform over the same space.

Stochastic Block Model (SBM) [Nowicki and Snijders 2001] is another type of generative model for groups in the network. Fitting an empirical network to an SBM requires inferring model parameters similar to GMM. In OSBM [Latouche et al. 2011], each node i is associated with a latent vector (i.e., community assignment) Z_i with K independent Boolean variables $Z_{ik} \in \{0, 1\}$, where K is the number of communities, and Z_{ik} is drawn from a multivariate Bernoulli distribution. Z is inferred by maximizing the posterior probability conditioned on the present of edges as in Ren et al. [2009]. OSBM requires more efforts than mixture models because the factorization in the observed condition distribution for edges given Z is in general intractable. MOSES[9] [McDaid and Hurley 2010] combines OSBM with the local optimization scheme in which the fitness function is defined based on the observed condition distribution. MOSES greedily expands a community from edges. Unlike OSBM, no connection probability parameters are required as input. The worst-case time complexity is $O(en^2)$, where e is the number of edges to be expanded.

Nonnegative Matrix Factorization (NMF) is a feature extraction and dimensionality reduction technique in machine learning that has been adapted to community detection. NMF approximately factorizes the feature matrix V into two matrices with the nonnegativity constraint as $V \approx WH$, where V is $n \times m$, W is $n \times k$, H is $k \times m$, and k is the number of communities provided by users. W represents the data in the reduced feature space. Each element $w_{i,j}$ in the normalized W quantifies the dependence of node i with respect to community j. In Zhang et al. [2007b], V is replaced with the diffusion kernel, which is a function of the Laplacian of the network. In Zarei et al. [2009], V is defined as the correlation matrix of the columns of the Laplacian. This results in better performance than Zhang et al. [2007b]. In Zhao et al. [2010], redundant constraints in the approximation are removed, reducing NMF to a problem of symmetrical Nonnegative Matrix Factorization (s-NMF). Psorakis et al. [2011] proposed a hybrid algorithm called Bayesian NMF[10]. The matrix V, where each element v_{ij} denotes a count of the interactions that took place between two nodes i and j, is decomposed via NMF as part of the parameter inference for a generative model similar to OSBM and GMM. Traditionally, NMF is inefficient with respect to both time and memory constraints due to the matrix multiplication. In the version of Psorakis et al. [2011], the worst-case time complexity is $O(kn^2)$, where k denotes the number of communities.

Wang et al. [2009] combined disjoint detection methods with local optimization algorithms. First, a partition is obtained from any algorithm for disjoint community detection. Communities attempt to add or remove nodes. The difference, called variance, of two fitness scores on a community, either including a node i or removing node i, is computed. The normalized variances form a fuzzy membership vector of node i.

Ding et al. [2010] employed the affinity propagation clustering algorithm [Frey and Dueck 2007] for overlapping detection, in which clusters are identified by representative exemplars. First, nodes are mapped as data points in the Euclidean space via the commute time kernel (a function of the inverse Laplacian). The similarity

[8] `http://www.cs.rpi.edu/~purnej/code.php`.

[9] `http://sites.google.com/site/aaronmcdaid/moses`.

[10] `http://www.robots.ox.ac.uk/~parg/software.html`.

between nodes is then measured by the cosine distance. Affinity propagation reinforces two types of messages associated with each node, the responsibility $r(i, k)$ and the availability $a(i, k)$. The probability for assigning node i into the cluster represented by exemplar node k is computed by equation $p(i, k) = e^{\hat{r}(i,k)}$, where $\hat{r}$ is the normalized responsibility as in Geweniger et al. [2009].

3.5. Agent-Based and Dynamical Algorithms

The label propagation algorithm [Raghavan et al. 2007; Xie and Szymanski 2011], in which nodes with same label form a community, has been extended to overlapping community detection by allowing a node to have multiple labels. In COPRA[11] [Gregory 2010], each node updates its belonging coefficients by averaging the coefficients from all its neighbors at each time step in a synchronous fashion. The parameter v is used to control the maximum number of communities with which a node can associate. The time complexity is $O(vm \log(vm/n))$ per iteration.

SLPA[12] [Xie et al. 2011; Xie and Szymanski 2012] is a general speaker-listener-based information propagation process. It spreads labels between nodes according to pairwise interaction rules. Unlike Raghavan et al. [2007] and Gregory [2010], where a node forgets knowledge gained in the previous iterations, SLPA provides each node with a memory to store received information (i.e., labels). The probability of observing a label in a node's memory is interpreted as the membership strength. SLPA does not require any knowledge about the number of communities, which is determined by the clustering of labels in the network. The time complexity is $O(tm)$, linear in the number of edges m, where t is a predefined maximum number of iterations (e.g., $t \geq 20$). SLPA can also be adapted for weighted and directed networks by generalizing the interaction rules, known as SLPAw.

A game-theoretic framework is proposed in Chen et al. [2010b], in which a community is associated with a Nash local equilibrium. A gain function and a loss function are associated with each agent. The game assumes that each agent is selfish and selects to join, leave, and switch communities based on its own utility. An agent is allowed to join multiple communities to handle overlapping, so long as it results in increased utility. The time complexity to find the best local operation for an agent i is $O(|L_i| \cdot |L(N_i)| \cdot k_i)$, where L_i is the communities that agent i wants to join, $L(N_i)$ is the set of communities that i's neighbors want to join, and k_i is the node degree. The time it takes to reach a local equilibrium is bounded by $O(m^2)$, where m is the number of edges.

A process in which particles walk and compete with each other to occupy nodes is presented in Breve et al. [2009]. Particles represent different communities. Each node has an instantaneous ownership vector (similar to belonging factor) and a long-term ownership vector. At each iteration, each particle takes either a random walk or a deterministic walk to one of its neighbors with some probability. If the random walk is performed, the visited neighbor updates its instantaneous ownership vector; otherwise, the long-term ownership vector is updated. At the end of the process, the long-term ownership vector is normalized to produce a soft assignment. Different from SLPA and COPRA, this algorithm takes a semisupervised approach. It requires at least one labeled node per class.

Multistate spin models [Reichardt and Bornholdt 2004; Lu et al. 2009], in which a spin is assigned to each node, can also be applied to community detection. One of such models is the q-state Potts model [Blatt et al. 1996; Reichardt and Bornholdt 2004], where q is the number of states that a spin may take, indicating the maximum number of communities. The community detection problem is equivalent to the problem of

[11] http://www.cs.bris.ac.uk/~steve/networks/software/copra.html.
[12] https://sites.google.com/site/communitydetectionslpa.

minimizing the Hamiltonian of the model. In the ground states (i.e., local minima of the Hamiltonian), the set of nodes with the same spin state form a community. The overlap of communities is linked to the degeneracy of the minima of the Hamiltonian [Reichardt and Bornholdt 2006a]. Although a co-appearance matrix keeps track of how frequently nodes i and j have been grouped together over multiple runs, it is not clear how to aggregate this information into overlapping communities when analyzing large networks. Another Potts model-like approach was proposed in Ronhovde and Nussinov [2009] to evaluate the hierarchical or multiresolution structure of a graph via information-based replica correlations.

Synchronization of a system that consists of coupled phase oscillators is able to uncover community structures. In such a model (e.g., the Kuramoto model) the phase of each unit evolves in time according to the predefined dynamics. The set of nodes with the same phase or frequency can be viewed as a community [Arenas et al. 2006] while nodes that do not match any observed dynamic behaviors can be considered overlapping nodes [Li et al. 2008]. Like methods utilizing a Potts model, such algorithms are parameter dependent.

3.6. Others

CONGA[13] [Gregory 2007] extends Girvan and Newman's divisive clustering algorithm (GN) [Girvan and Newman 2002] by allowing a node to split into multiple copies. Both *splitting betweenness*, defined by the number of shortest paths on the imaginary edge, and the conventional edge betweenness are considered. CONGA inherits the high computational complexity of GN. In a more refined version, CONGO [Gregory 2008] uses local betweenness to optimize the speed. Gregory [2009] also proposed to perform disjoint detection algorithms on the network produced by splitting the node into multiple copies using the split betweenness.

Zhang[14] [Zhang et al. 2009] proposed an iterative process that reinforces the network topology and *propinquity* that is interpreted as the probability of a pair of nodes belonging to the same community. The propinquity between two vertices is defined as the sum of the number of direct links, number of common neighbors, and the number of links within the common neighborhood. Given the topology, propinquity is computed. Propinquity above a certain threshold is then used to redistribute links, updating the topology. If the propinquity is large, a link is added to the network; otherwise, the link is removed. The propinquity can be used to perform microclustering on each vertex to allow overlap.

Kovács et al. [2010] proposed an approach focusing on centrality-based influence functions. Community structures are interpreted as hills of the influence landscape. For each node i, the influence over each link $f_i(j,k)$ is computed. Links within a community should have higher influence than those linking distant areas of the network. The influence on a given link $c(j,k)$ is the sum of $f_i(j,k)$ over all nodes. The function $c(j,k)$ over each link defines the community landscape, wherein the communities are determined by local maxima and their surrounding regions.

Rees and Gallagher [2010] proposed an algorithm to extract the overlapping communities from the *egonet*, which is a subgraph including a center node, its neighbors, and the links around them. When all egonets are induced, each center is removed, creating small connected components among neighbors. Then, the center node is added back to each of these components to form a so-called *friendship group*. Clearly, each center node can be in multiple friendship groups. The overlapping communities are determined by merging all friendship groups.

[13] http://www.cs.bris.ac.uk/~steve/networks/software/conga.html.

[14] http://dbgroup.cs.tsinghua.edu.cn/zhangyz/kdd09.

Inspired by OPTIC [Ankerst et al. 1999], an algorithm based on techniques from visualization was proposed in Chen et al. [2009]. Nodes are ordered according to the Reachability Score (RS) with respect to a starting node. The reachability is based on the probability of the existence of a link between two nodes. By scanning through the obtained sequence of nodes, a community containing consecutive nodes with RS larger than a community threshold is found. Clearly, this algorithm is hard to apply to large networks and requires the introduction of a community threshold.

4. EVALUATION CRITERIA

Evaluating the quality of a detected partitioning or cover is nontrivial, and extending evaluation measures from disjoint to overlapping communities is rarely straightforward. Unlike disjoint community detection, where a number of measures have been proposed for comparing identified partitions with the known partitions [Danon et al. 2005; Leskovec et al. 2010], only a few measures are suitable for a set of overlapping communities. Two most widely used measures are the Normalized Mutual Information (NMI) and Omega index.

4.1. Normalized Mutual Information

Lancichinetti et al. [2009] has extended the notion of normalized mutual information to account for overlap between communities. For each node i in cover C', its community membership can be expressed as a binary vector of length $|C'|$ (i.e., the number of clusters in C'). $(x_i)_k = 1$ if node i belongs to the k^{th} cluster C'_k; $(x_i)_k = 0$ otherwise. The k^{th} entry of this vector can be viewed as a random variable X_k, whose probability distribution is given by $P(X_k = 1) = n_k/n$, $P(X_k = 0) = 1 - P(X_k = 1)$, where $n_k = |C'_k|$ is the number of nodes in the cluster C'_k and n is the total number of nodes. The same holds for the random variable Y_l associated with the l^{th} cluster in cover C''. Both the empirical marginal probability distribution $P(X_k)$ and the joint probability distribution $P(X_k, Y_l)$ are used to further define entropy $H(X)$ and $H(X_k, Y_l)$.

The conditional entropy of a cluster X_k given Y_l is defined as $H(X_k|Y_l) = H(X_k, Y_l) - H(Y_l)$. The entropy of X_k with respect to the entire vector Y is based on the best matching between X_k and any component of Y given by

$$H(X_k|Y) = min_{l\in\{1,2,\ldots,|C''|\}} H(X_k|Y_l).$$

The normalized conditional entropy of a cover X with respect to Y is

$$H(X|Y) = \frac{1}{|C'|}\sum_k \frac{H(X_k|Y)}{H(X_k)}.$$

In the same way, one can define $H(Y|X)$. Finally the NMI for two covers C' and C'' is given by

$$NMI(X|Y) = 1 - [H(X|Y) + H(Y|X)]/2. \tag{9}$$

The extended NMI is between 0 and 1, with 1 corresponding to a perfect matching. Note that this modified NMI does not reduce to the standard formulation of NMI when there is no overlap.

4.2. Omega Index

Omega index [Collins and Dent 1988] is the overlapping version of the Adjusted Rand Index (ARI) [Hubert and Arabie 1985]. It is based on pairs of nodes in agreement in two covers. Here, a pair of nodes is considered to be in *agreement* if they are clustered in exactly the same number of communities (possibly none). That is, the Omega index considers how many pairs of nodes belong together in no clusters, how many are placed together in exactly one cluster, how many are placed in exactly two clusters, and so on.

Let K_1 and K_2 be the number of communities in covers C_1 and C_2, respectively, the Omega index is defined as [Gregory 2011; Havemann et al. 2011]

$$\omega(C_1, C_2) = \frac{\omega_u(C_1, C_2) - \omega_e(C_1, C_2)}{1 - \omega_e(C_1, C_2)}. \tag{10}$$

The unadjusted Omega index ω_u is defined as

$$\omega_u(C_1, C_2) = \frac{1}{M} \sum_{j=0}^{max(K_1,K_2)} |t_j(C_1) \cap t_j(C_2)|,$$

where M equals to $n(n-1)/2$ represents the number of node pairs, and $t_j(C)$ is the set of pairs that appear exactly j times in a cover C. The expected Omega index in the null model ω_e is given by

$$\omega_e(C_1, C_2) = \frac{1}{M^2} \sum_{j=0}^{max(K_1,K_2)} |t_j(C_1)| \cdot |t_j(C_2)|.$$

The subtraction of the expected value in (10) takes into account agreements resulting from chance alone. The larger the Omega index, the better the matching between two covers. A value of 1 indicates perfect matching. When there is no overlap, the Omega index reduces to the ARI.

In addition to NMI and Omega, some other measures have been proposed, such as the generalized external indexes [Campello 2007, 2010] and the fuzzy rand index [Hüllermeier and Rifqi 2009].

5. BENCHMARKS

It is necessary to have good benchmarks to both study the behavior of a proposed community detection algorithm and to compare the performance across various algorithms. In order to accurately perform these two analyses, networks in which the ground truth is known are needed. This requirement implies that real-world networks, which are often collected from online or observed interactions, do not paint a clear enough picture due to their lack of "ground truth". In light of this requirement, we begin our discussion with synthetic networks. In the GN benchmark [Girvan and Newman 2002], equal size communities are embedded into a network for a given expected degree and a given mixing parameter μ that measures the ratio of internal connections to outgoing connections. One drawback of this benchmark is that it fails to account for the heterogeneity in complex networks. Another is that it does not allow embedded communities to overlap. A few benchmarks have been proposed for testing overlapping community detection algorithms, all of which are special cases of the planted l-partition model [Condon and Karp 2001] just like GN.

Sawardecker et al. [2009] proposed an extension of GN, in which the probability p_{ij} of an edge being present in the network is a nondecreasing function based solely on the set of comemberships of nodes i and j. With the definition $p_{ij} = p_k$, parameter p_k is the connection probability of nodes i and j that cooccur k times, subject to $p_0 < p_1 \leq p_2 \leq \dots$.

The LFR[15] benchmark proposed in Lancichinetti et al. [2008] introduces heterogeneity into degree and community size distributions of a network. These distributions are

[15] http://sites.google.com/site/andrealancichinetti/files.

Table I. Algorithms Included in the Experiments

Algorithm	Reference	Complexity	Imp
CFinder	[Palla et al. 2005]	-	C++
LFM	[Lancichinetti et al. 2009]	$O(n^2)$	C++
EAGLE	[Shen et al. 2009a]	$O(n^2 + (h+n)s)$	C++
CIS	[Kelley 2009]	$O(n^2)$	C++
GCE	[Lee et al. 2010]	$O(mh)$	C++
COPRA	[Gregory 2010]	$O(vm \log(vm/n))$	Java
Game	[Chen et al. 2010b]	$O(m^2)$	C++
NMF	[Psorakis et al. 2011]	$O(kn^2)$	Matlab
MOSES	[McDaid and Hurley 2010]	$O(en^2)$	C++
Link	[Ahn et al. 2010]	$O(nk_{max}^2)$	C++
iLCD	[Cazabet et al. 2010]	$O(nk^2)$	Java
UEOC	[Jin et al. 2011]	$O(ln^2)$	Matlab
OSLOM	[Lancichinetti et al. 2011]	$O(n^2)$	C++
SLPA	[Xie et al. 2011; Xie and Szymanski 2012]	$O(tm)$	C++

governed by power laws with exponents τ_1 and τ_2, respectively. To generate overlapping communities O_n, the fraction of overlapping nodes is specified and each node is assigned to $O_m \geq 1$ communities. The generating procedure is equivalent to generating a bipartite network where the two classes are the communities and nodes subject to the requirement that the sum of community sizes equals the sum of node memberships. LFR also provides a rich set of parameters to control the network topology, including the mixing parameter μ, the average degree $\bar{k}$, the maximum degree k_{max}, the maximum community size c_{max}, and the minimum community size c_{min}.

The LFR model brings benchmarks closer to the features observed in real-world networks. However, requiring that overlapping nodes interact with the same number of embedded communities is unrealistic in practice. A simple generalization where each overlapping node may belong to different number of communities has been considered in McDaid and Hurley [2010].

In Gregory [2011], crisp communities from LFR are converted to fuzzy associations by adding a belonging coefficient to the occurrence of nodes. This coefficient can be defined as

$$p_{ij} = s_{ij} p_1 + (1 - s_{ij}) p_0,$$

where p_k is the same as in Sawardecker et al.'s model and $s_{ij} = \sum_{c \in C} \alpha_{ic} \alpha_{jc}$ is the similarity of node i and j as defined in (6). In other words, the probability of an edge being present depends not only on the number of communities in which nodes i and j appear together but also on their degree of belonging to these communities.

6. TESTS ON SYNTHETIC NETWORKS

In this section, we empirically compare the performance of different algorithms on LFR networks. We focus on algorithms which produce a crisp assignment of vertices to communities. In total, 14 algorithms were collected and tested. They are listed in Table I. Note that the time complexity given is for the worst case.

For algorithms with tunable parameters, the results with the best setting are reported. For LFM, we varied α from 0.8 to 1.6 with an interval 0.1, within which good results have previously been reported [Lancichinetti et al. 2009; Lee et al. 2010]. For iLCD, *fRatio* is from $\{0.75, 0.5, 0.35\}$ and *bThreshold* is from $\{0.5, 0.3, 0.2\}$ as suggested by the authors. For GCE, the minimum clique size k ranges from 3 to 8. For CFinder, k ranges from 3 to 8. For OSLOM, we considered the first two levels. For Link, the threshold varies from 0.1 to 0.9 with an interval 0.1. For COPRA, parameter v is taken from the range [1,10]. For SLPA, parameter r varies from 0.05 to 0.5 with an interval 0.05. Since COPRA and SLPA are nondeterministic, we repeated each of them 10 times on each network instantiation. For NMF, which returns a fuzzy assignment, we applied the same threshold as SLPA to convert it to a crisp assignment.

For each parameter set generated via LFR, we generated 10 instantiations. We used networks with sizes $n \in \{1000, 5000\}$. The average degree is kept at $\bar{k} = 10$, which is of the same order as most large real-world social networks[16]. The rest of the parameters of the LFR generator are set similar to those in Lancichinetti and Fortunato [2009]: node degrees and community sizes are governed by power law distributions with exponents $\tau_1 = 2$ and $\tau_2 = 1$, respectively, the maximum degree is $k_{max} = 50$, and community sizes vary in both small range $s = (10, 50)$ and large range $b = (20, 100)$. The mixing parameter μ is from $\{0.1, 0.3\}$, which is the expected fraction of links through which a node connects to other nodes in the same community.

The degree of overlap is determined by two parameters. O_n is the number of overlapping nodes, and O_m is the number of communities to which each overlapping node belongs. O_n is set to 10% and 50% of the total number of nodes, indicating low and high *overlapping density*, respectively. Instead of fixing O_m [Lancichinetti and Fortunato 2009; Gregory 2010], we also allow it to vary from 2 to 8 indicating the *overlapping diversity* of overlapping nodes. By increasing the value of O_m, we create harder detection tasks. This also allow us to look in greater detail at the detection accuracy at node level.

Two previously discussed measures, Omega and NMI, are used to quantify the quality of the cover discovered by an algorithm.

6.1. Effects of μ, n and O_m

We first examined how the performance, measured by NMI, changes as the number of memberships O_m varies from small to large values (i.e., 2 to 8) for different network sizes (n) and intra-community strength (μ) in Figure 1.

In general, changes in the network topology, especially the mixing value μ, have a similar impact for disjoint community detection. That is, the larger the value of μ, the poorer the results produced by detection algorithms (i.e., red curve $<$ blue curve in Figure 1) due to the fact that the connection inside communities is weak for larger μ. This is true for the majority of algorithms with the only exception NMF (see the 5000b case). On the other hand, increasing network size from 1000 to 5000 typically results in slightly better performance (i.e., square $>$ dot in Figure 1), with prominent exceptions for EAGLE, NMF, and UEOC. Slight fluctuations in performance are observed for iLCD and Link. Detection performance typically decays at a moderate rate as the diversity of overlapping increases (i.e., O_m getting larger), except for OSLOM and UEOC.

6.2. Effects of Community Size Range and Overlapping Density O_n

We evaluated the effects of O_n and community size ranges on each individual algorithm on networks with $n = 5000$ and $\mu = 0.3$. Results for NMI are shown in Figure 2.

As expected, the performance of detection consistently and significantly drops in the case where there are many overlapping nodes for all algorithms (i.e., red curves ($O_n = 50\%$) $<$ blue curves ($O_n = 10\%$)). However, the difference in performance between small and large community size ranges (gaps between two curves with the same color) is more prominent in the case of low overlapping density.

Interestingly, the NMI's for networks with small community size $s = (10, 50)$ are typically higher than those for networks with large community size range $b = (20, 100)$

[16] snap.stanford.edu/data.

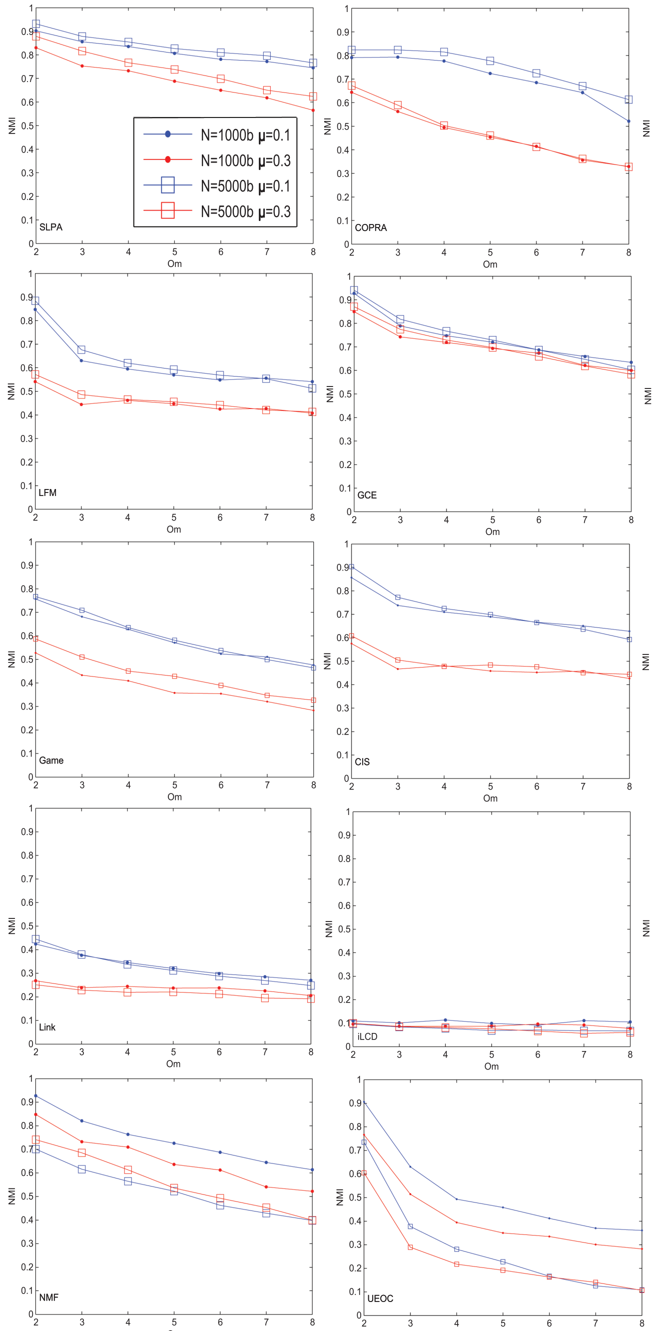

Fig. 1. The effects of network size n and mixing parameter μ on LFR networks. Plots show NMI's for networks with large community size range and $O_n = 10\%$. The order of subplots is from general behaviors to exceptions (see text).

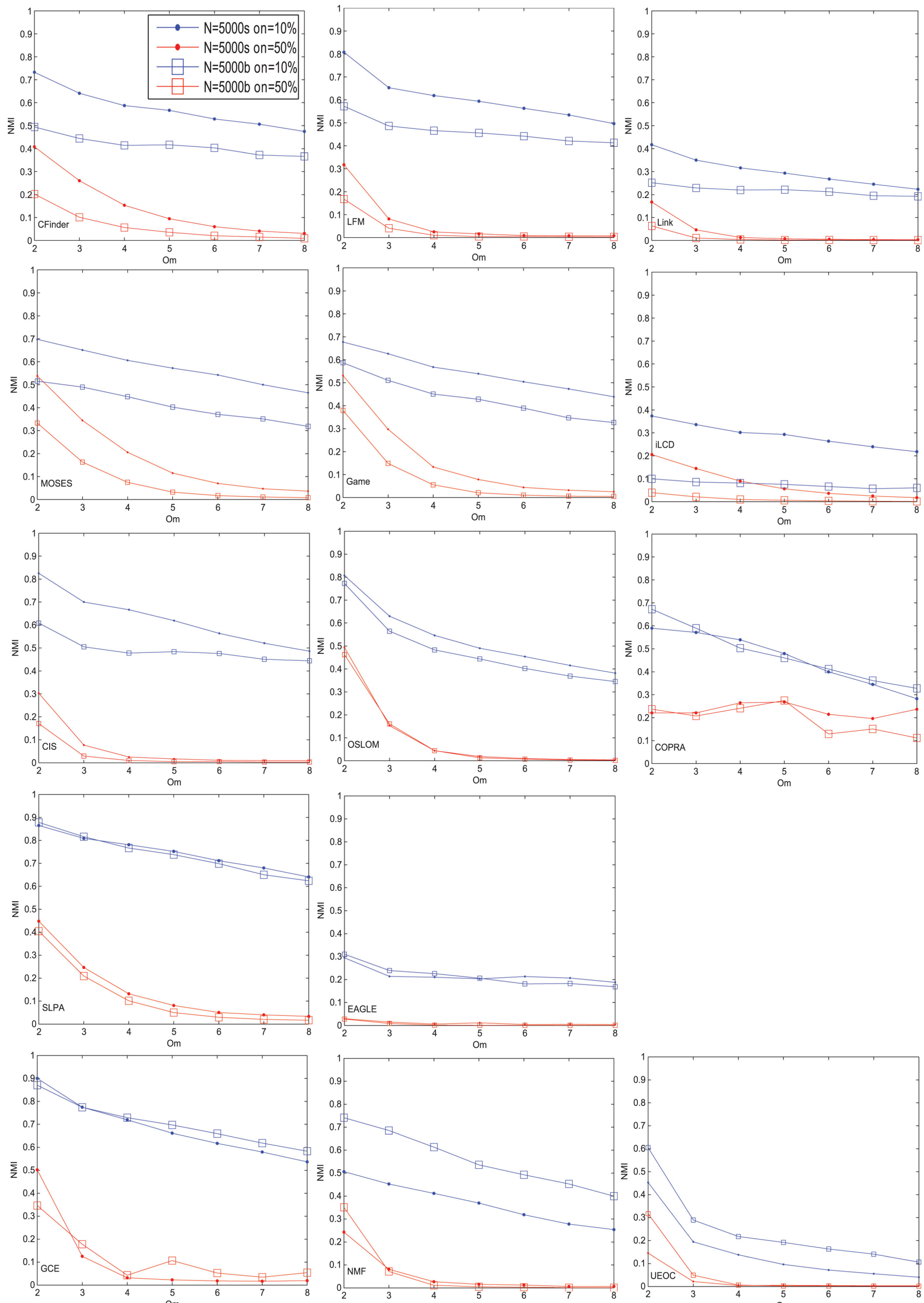

Fig. 2. The effects of community size range and overlapping density O_n on LFR networks. Plots show NMI's for networks with $n = 5000$ and $\mu = 0.3$. The order of subplots is from general behaviors to exceptions (see text).

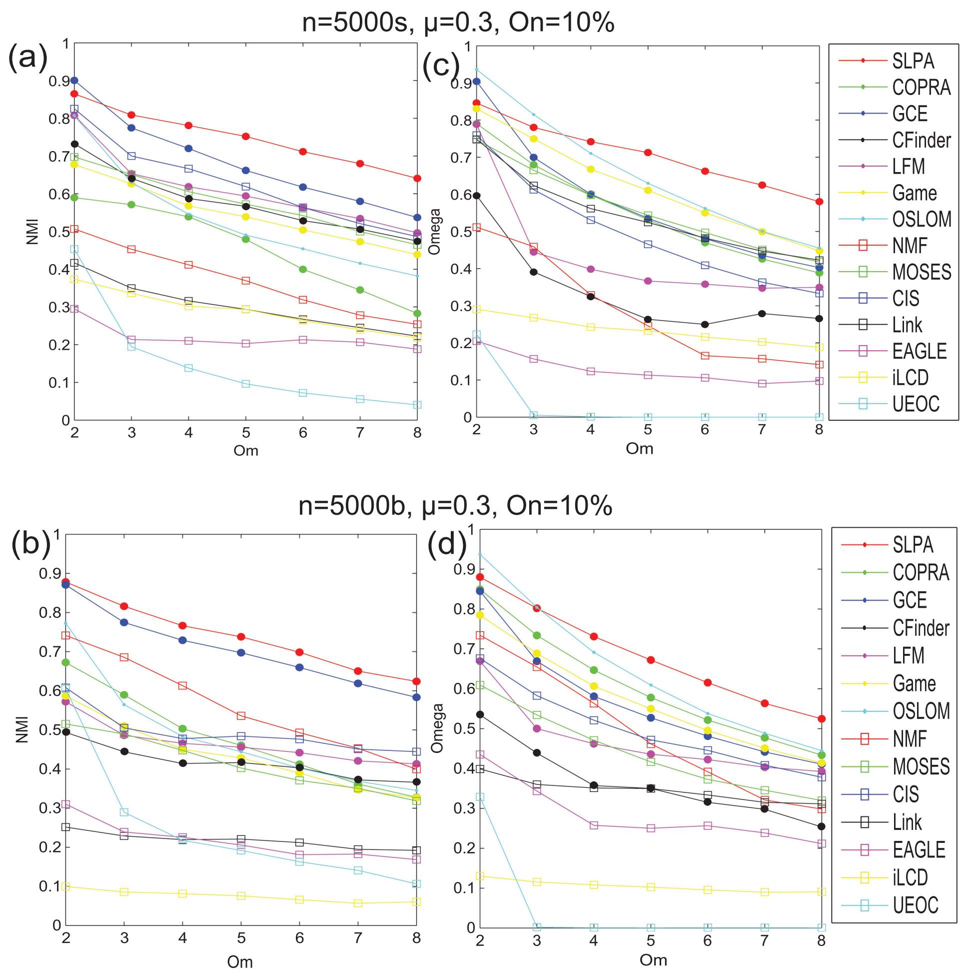

Fig. 3. Evaluations of overlapping community detection on LFR networks with low overlap density O_n = 10%. Left column: NMI as a function of the number of memberships O_m; Right column: Omega as a function of the number of memberships O_m. Results for small community size range are shown in top row (i.e., (a) and (c)), and results for large community size range are shown in bottom row (i.e., (b) and (d)). All resutls are from networks with $n = 5000$ and $\mu = 0.3$.

(i.e., dot > square in Figure 2). It appears that the well-known resolution limit does not play a role here since all the tested algorithms are neither based on a modularity nor an extended modularity. This is evidenced by algorithms including CFinder, LFM, Link, MOSES, Game, iLCD, CIS, and OSLOM that have a significant performance gap between small and large community size ranges. Given only small variances in performance in two tested ranges, we also conclude that the community size range has limited impact on algorithms including SLPA, COPRA, EAGLE, and GCE.

6.3. Ranking for Community Detection

Extensive comparisons have been conducted over different overlapping densities and community size ranges. Performance measured by NMI and Omega for $n = 5000$ and $\mu = 0.3$ is shown in Figures 3 and 4.

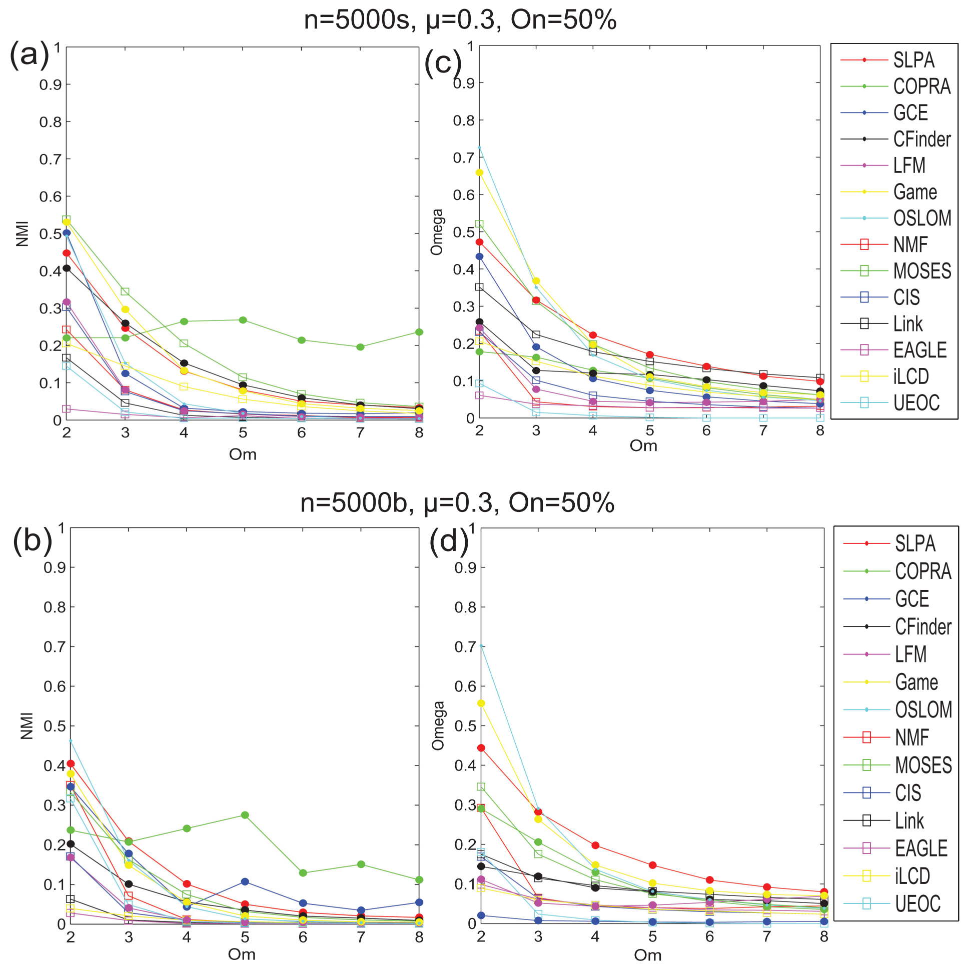

Fig. 4. Evaluations of overlapping community detection on LFR networks with high overlap density $O_n = 50\%$. Left column: NMI as a function of the number of memberships O_m; Right column: Omega as a function of the number of memberships O_m. Results for small community size range are shown in top row (i.e., (a) and (c)), and results for large community size range are shown in bottom row (i.e., (b) and (d)). All resutls are from networks with $n = 5000$ and $\mu = 0.3$.

To provide an intuitive way for both comparing two measures and also summarizing the vast volume of experiment results, we propose $RS_M(i)$, the averaged ranking score for a given algorithm i with respect to some measure M as:

$$RS_M(i) = \sum_{j=1} rank(i, O_m^j), \tag{11}$$

where O_m^j is the number of memberships (diversity) in $\{2, 3, \ldots, 8\}$, and function *rank* returns the ranking of algorithm i for the given O_m. Sorting RS_M in increasing order gives the final ranking among algorithms. Whenever it is clear from context, we use the term *ranking* to refer to the final rank without the actual score value.

The results for low overlapping density case in Figure 3 are summarized as four rankings including RS^s_{NMI}, RS^b_{NMI}, RS^s_{Omega}, and RS^b_{Omega} in Table II, where $RS^{s(b)}_{NMI(Omega)}$

Table II. The Community Detection Ranking for $n = 5000$, $\mu = 0.3$ and Low Overlapping Density $O_n = 10\%$

Rank	RS^s_{NMI}	RS^s_{Omega}	RS^b_{NMI}	RS^b_{Omega}	$RS^*_{NMI,Omega}$	RS^*_F
1	SLPA	SLPA	SLPA	SLPA	SLPA	SLPA
2	GCE	OSLOM	GCE	OSLOM	GCE	CFinder
3	CIS	Game	NMF	COPRA	OSLOM	Game
4	LFM	GCE	CIS	Game	CIS	OSLOM
5	MOSES	MOSES	COPRA	GCE	Game	MOSES
6	CFinder	COPRA	OSLOM	CIS	COPRA	COPRA
7	Game	Link	LFM	NMF	LFM	iLCD
8	OSLOM	CIS	Game	LFM	MOSES	Link
9	COPRA	LFM	CFinder	MOSES	NMF	LFM
10	NMF	CFinder	MOSES	CFinder	CFinder	UEOC
11	Link	NMF	Link	Link	Link	EAGLE
12	iLCD	iLCD	EAGLE	EAGLE	EAGLE	GCE
13	EAGLE	EAGLE	UEOC	iLCD	iLCD	CIS
14	UEOC	UEOC	iLCD	UEOC	UEOC	NMF

Table III. The Community Detection Ranking for $n = 5000$, $\mu = 0.3$ and High Overlapping Density $O_n = 50\%$

Rank	RS^s_{NMI}	RS^s_{Omega}	RS^b_{NMI}	RS^b_{Omega}	$RS^*_{NMI,Omega}$	RS^*_F
1	MOSES	SLPA	COPRA	SLPA	SLPA	Link
2	COPRA	Link	SLPA	Game	MOSES	UEOC
3	CFinder	Game	GCE	OSLOM	Game	SLPA
4	Game	MOSES	MOSES	Link	COPRA	Game
5	SLPA	CFinder	CFinder	MOSES	CFinder	LFM
6	GCE	OSLOM	Game	CFinder	OSLOM	CFinder
7	iLCD	COPRA	OSLOM	COPRA	Link	CIS
8	OSLOM	GCE	LFM	LFM	GCE	MOSES
9	CIS	iLCD	CIS	NMF	LFM	OSLOM
10	LFM	LFM	NMF	EAGLE	CIS	iLCD
11	NMF	CIS	UEOC	CIS	iLCD	GCE
12	Link	NMF	iLCD	iLCD	NMF	COPRA
13	EAGLE	EAGLE	Link	UEOC	EAGLE	EAGLE
14	UEOC	UEOC	EAGLE	GCE	UEOC	NMF

denotes the ranking based on NMI (or Omega) for networks with small (or large) community size range. Results for high overlapping density case in Figure 4 are summarized in Table III.

We first compared pairwise similarities of (RS^s_{NMI}, RS^s_{Omega}) and (RS^b_{NMI}, RS^b_{Omega}). As shown, among the top seven (half of the total fourteen) algorithms in two rankings, there are 4 pairs of matches (ignoring the exact order) for $O_n = 10\%$ and 5 pairs of matches for $O_n = 50\%$ for (RS^s_{NMI}, RS^s_{Omega}). Even more, there are 6 pairs of matches for both $O_n = 10\%$ and $O_n = 50\%$ for (RS^b_{NMI}, RS^b_{Omega}). This suggests that NMI and Omega provide similar overall evaluation to some extent.

Based on these four rankings, we further derive the average ranking $RS^*_{NMI,Omega}$ as the overall community detection performance. In this final ranking, the top seven algorithms are exclusively agent-based algorithms (SLPA, Game, and COPRA) and local expansion-based algorithms (GCE, OSLOM, CIS, and LFM), which significantly outperform the others for networks with low overlapping density (see Figure 3).

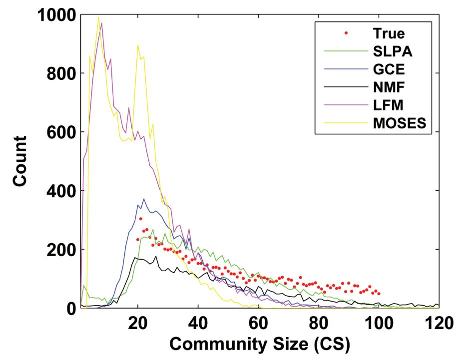

Fig. 5. Histogram of the detected community sizes for SLPA, GCE, NMF, LFM, and MOSES created from the results for LFR networks with $n = 5000$, $\mu = 0.3$, and $O_n = 10\%$.

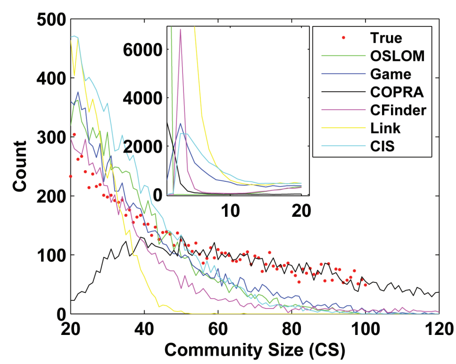

Fig. 6. Histogram of the detected community sizes for OSLOM, Game, COPRA, CFinder, Link, and CIS created from the results for LFR networks with $n = 5000$, $\mu = 0.3$, and $O_n = 10\%$.

For high overlapping density, agent-based algorithms remain in the top seven, together with MOSES representing fuzzy algorithms, CFinder representing clique algorithms, and Link representing link partitioning. However, given the fact that the performance is actually fairly low (most of them are less than 0.5 for $O_m > 2$) shown in Figure 4, it is fair to conclude that all the algorithms do not yet achieve satisfying performance for networks with high overlapping density and high overlapping diversity (e.g., for $O_n = 50\%$ and $O_m > 2$).

6.4. Comparing Detected Community Size Distribution in LFR

In order to provide insight into the behaviors of different algorithms and verify the ranking, we examined the discovered distribution (histograms) of Community Sizes (CS) and compared it with the known ground truth. Here we only provide analysis for $n = 5000$, $\mu = 0.3$, $O_n = 10\%$ (the corresponding ranking is RS^b_{NMI} in Table II). In this case, we expect the community size distribution to follow the power law with exponent $\tau_2 = 1$, a minimum of 20, and a maximum size of 100. Note that the histograms are created from communities over different $O'_m s$. As shown in Figure 5, SLPA, GCE, and NMF find communities whose sizes are distributed in a *unimodal* distribution with a single peak at $CS = 20$ in aggreement with the ground-truth distribution. This explains why they perform well with respect to ranking RS^b_{NMI}. LFM and MOSES have a peak around $CS = 5$, which lowers their performance. The prominent feature of Figure 6 (see the inset) is a *bimodal* distribution that has a peak at $CS = 1 \sim 5$. This means that

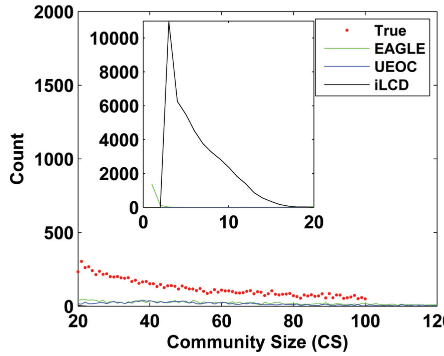

Fig. 7. Histogram of the detected community sizes for EAGLE, UEOC, and iLCD crated from the results for LFR networks with $n = 5000$, $\mu = 0.3$ and $O_n = 10\%$.

algorithms like OSLOM, Game, COPRA, CIS, and CFinder find significant numbers of small communities. In Figure 7, the distribution is shifted mostly outside the predefined range 20$\sim$100. Algorithms with such a distribution create relatively small communities and perform poorly with respect to this analysis. Here, we conclude that observations on the community size distribution can be used to explain the performance and ranking.

It is worth noticing that in Figure 6, excluding the range that contains the undesired peak ($CS = 1 \sim 20$), the distributions seem to agree well with the ground truth, especially for COPRA. The performances of OSLOM, Game, CIS, and COPRA with respect to NMI are still fairly stable. This demonstrates that NMI is, to some degree, not sensitive to small size communities (including outliers or singleton communities).

6.5. Identifying Overlapping Nodes in LFR

Community overlap manifests itself as the existence of the nodes with membership in multiple communities. Thus, we will refer to nodes with multiple membership as overlapping nodes. In real-world social networks, such nodes are important because they usually represent bridges (or messengers) between communities. For this reason, the ability to identify overlapping nodes, although often neglected, is essential for assessing the accuracy of community detection algorithms. Measures like NMI and Omega focus only on providing an overall measure of algorithmic accuracy. As we see in Section 6.4, these measures might not be sensitive enough to provide an accurate picture of what is happening at node level. In this section, we evaluate an algorithm's ability to identify overlapping nodes.

Similar to the definitions of O_n and O_m, we define the number of detected overlapping nodes O_n^d and detected memberships O_m^d. Note that the number of overlapping nodes O_n^d alone is insufficient to accurately quantify the detection performance, because it contains both true and false positive. Ideally, an algorithm should report as many true overlapping nodes as possible (i.e., a balance between quality and quantity). To provide more precise analysis, we consider the identification of overlapping nodes as a *binary classification* problem. A node is labeled as overlapping as long as $O_m > 1$ or $O_m^d > 1$ and labeled as nonoverlapping otherwise. Within this framework, one can use Jaccard index as in Ball et al. [2011] or F-score as a measure of detection accuracy. In this article, we use the latter that is defined as

$$F = \frac{2 \cdot precision \cdot recall}{precision + recall}, \tag{12}$$

where *recall* is the number of correctly detected overlapping nodes divided by the true number of overlapping nodes O_n, and *precision* is the number of correctly detected overlapping nodes divided by the total number of detected overlapping nodes O_n^d. F-score accounts for the balance between detection quantity and quality, and reaches its best and worst value at 1 and 0, respectively.

Figures 8 and 9 show the F-score, precision, and recall for different settings of the LFR benchmark. In general, an algorithm achieves better F-score on benchmarks with small community sizes for both low and high overlapping density. However, the behaviors are quite different for the various algorithms. For example, the gain in F-score on communities in the small size range for EAGLE is due to the increase in recall (i.e., detect more overlapping nodes shown in (c) and (f) in Figure 8), while the gain for iLCD is mainly due to the increase in precision (see (b) and (e) in Figure 8). Moreover, the F-score (performance) typically decays moderately as overlapping diversity O_m increases. It is evident that O_m has great impacts on OSLOM with a rapid drop for large O_m. SLPA is the only exception that has a positive correlation with O_m for the low overlapping density case.

In terms of the precision, half of the algorithms including SLPA, CFinder, Game, OSLOM, MOSES, EAGLE, and iLCD consistently outperform the expected random performance, 10% and 50% for low and high overlapping density respectively (see (b) and (e) in both Figures 8 and 9). The high precision of EAGLE (also CFinder and GCE for $O_m = 2$) shows that clique-like assumption of communities may help to identify overlapping nodes in the low overlapping density case. Link performs merely as well as the random classifier.

Experiments also reveal an imbalance in precision and recall for some algorithms, which is partially due to either overdetection where more overlapping nodes than there exists are claimed or underdetection where only very few overlapping nodes are identified. Extreme examples are EAGLE and Link. Although EAGLE achieves very high detection precision (e.g., (b) and (e) in Figure 8), it suffers underdetection (verified in Figure 10), which results in a low recall score ((c) and (f) in Figure 8). As a result, we observe a low F-score ((a) and (b) in Figure 8). For Link, the algorithm does not perform well in terms of F-score even though it has very high recall ((c) and (f) in Figure 8). This is due to the fact that Link claims way more overlapping nodes than excepted (verified in Figure 10).

6.6. Ranking for Overlapping Node Detection

The rankings with respect to F-scores for different community size ranges, RS_F^s and RS_F^b are shown in Tables IV and V for different overlapping density cases. RS_F^* is the average ranking over two community size ranges.

To facilitate comparison, we copy RS_F^* into Tables II and III. It is clearly shown, for example in Table II, that the community quality ranking $RS^*_{NMI,Omega}$ and node quality ranking RS_F^* might provide quite different pictures of the performance. For the low overlapping density case (as in Table II), algorithms with a low rank in detecting communities could actually have good performances when it comes to identifying overlapping nodes (e.g., CFinder, iLCD, and MOSES), while high-ranking algorithms, including GCE and CIS, might perform badly due to underdetection and overdetection, respectively. SLPA has very stable and good performance for the low overlapping density case. These observations suggest the need for a careful treatment of the algorithms with a high NMI or Omega score if the application of these algorithms is aimed at identifying nodes with multiple community memberships. Similar conclusions can be drawn for high overlapping density case.

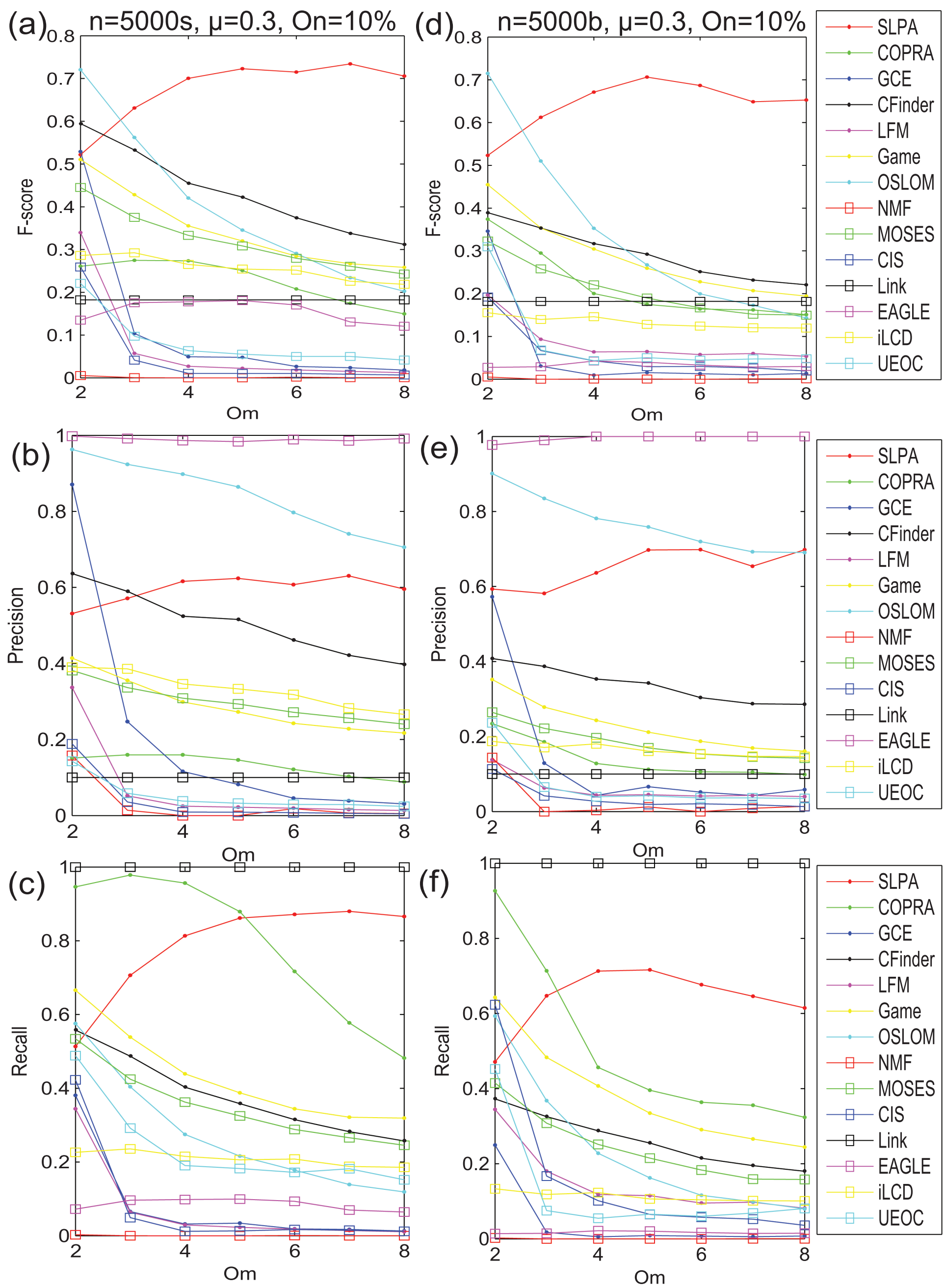

Fig. 8. Evaluations of overlapping node detection on LFR networks with low overlap density $O_n = 10\%$. Plots show F-score (together with precision and recall) as a function of the number of memberships for $n = 5000$ and $\mu = 0.3$. Results for small community size range are shown in the left column, and results for large community size range are shown in the righ column.

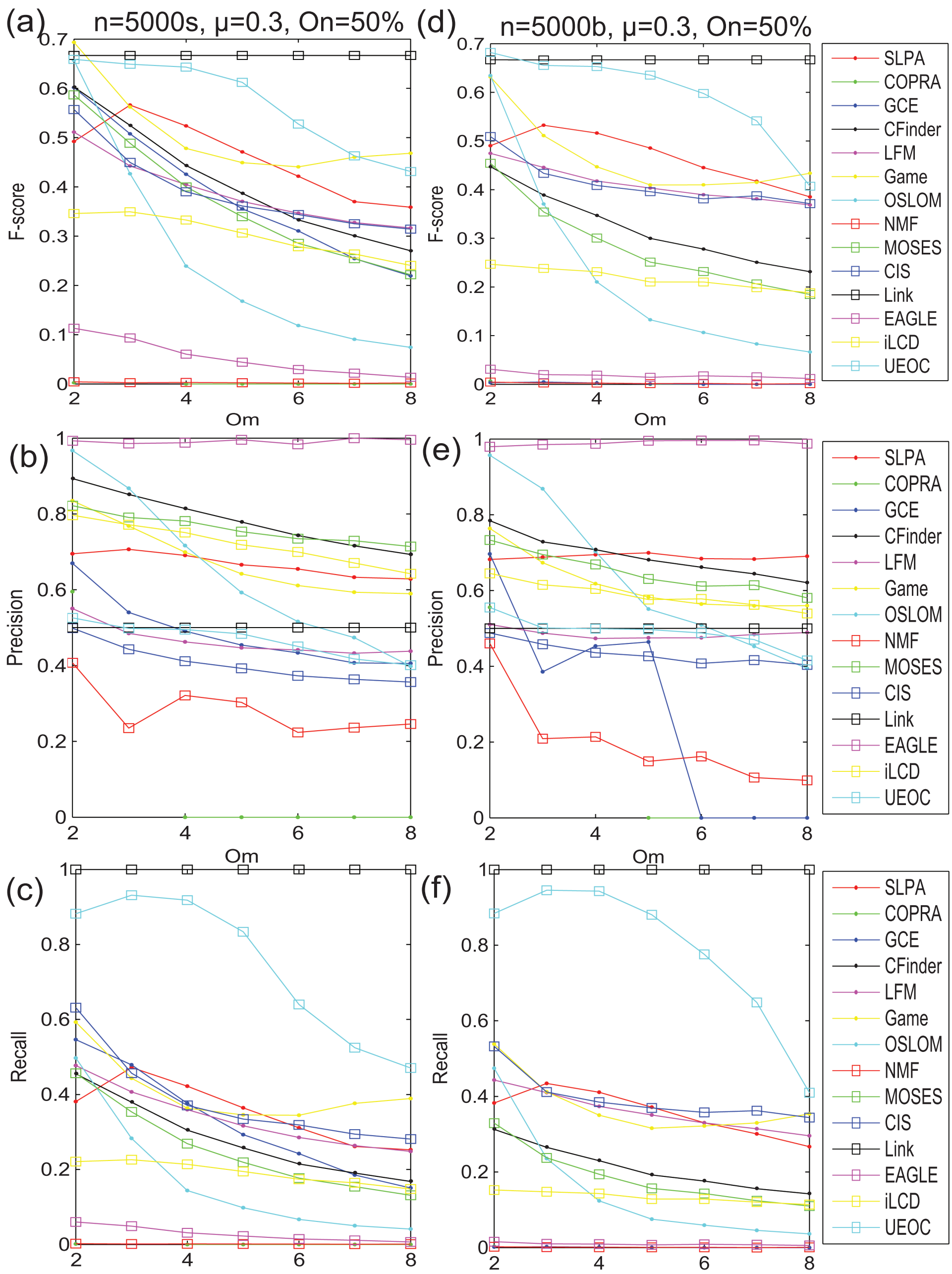

Fig. 9. Evaluations of overlapping node detection on LFR networks with high overlap density $O_n = 50\%$. Plots show F-score (together with precision and recall) as a function of the number of memberships for $n = 5000$ and $\mu = 0.3$. Results for small community size range are shown in the left column, and results for large community size range are shown in the righ column.

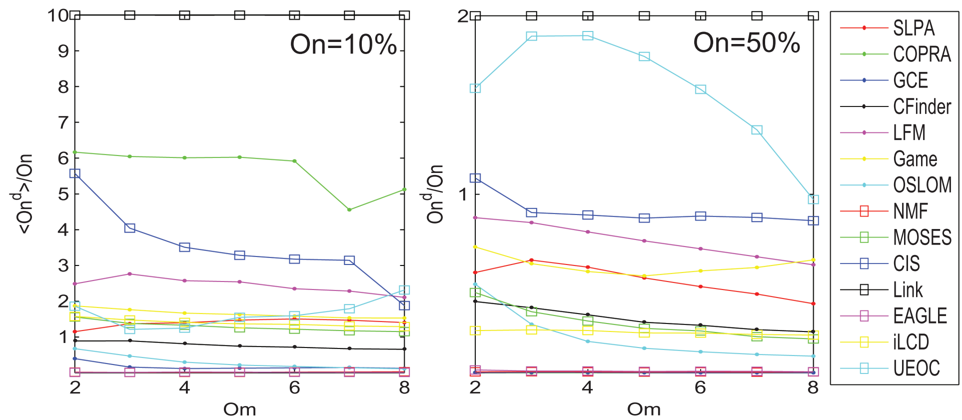

Fig. 10. The number of detected overlapping nodes (normalized by O_n) based on the results for LFR networks with $n = 5000b$ and $\mu = 0.3$. A value larger than 1 is possible.

Table IV. The Overlapping Node Detection Ranking for $n = 5000$, $\mu = 0.3$ and Low Overlap Density $O_n = 10\%$

Rank	RS_F^s	RS_F^b	RS_F^*
1	SLPA	SLPA	SLPA
2	CFinder	CFinder	CFinder
3	OSLOM	Game	Game
4	Game	OSLOM	OSLOM
5	MOSES	COPRA	MOSES
6	iLCD	MOSES	COPRA
7	COPRA	Link	iLCD
8	Link	iLCD	Link
9	EAGLE	LFM	LFM
10	GCE	UEOC	UEOC
11	UEOC	CIS	EAGLE
12	LFM	EAGLE	GCE
13	CIS	GCE	CIS
14	NMF	NMF	NMF

6.7. Final Ranking

Since two types of rankings provide complementary information, we conclude by considering algorithms that are consistently ranked in the top seven in both RS_F^* and $RS_{NMI,Omega}^*$: (a) For low overlapping density networks, SLPA, OSLOM, Game, and COPRA offer better performance than the other tested algorithms; (b) for high overlapping density networks, both SLPA and Game provide better performance. (Note that we do not include Link and UEOC because their high ranks are mainly due to the overdetection.)

7. TESTS ON REAL-WORLD SOCIAL NETWORKS

We first examined algorithm performance on a high school friendship network[17] where the ground truth is known. This social network from a high school is based on self-reporting from students. It is known that the true partitioning of the network roughly

[17] A project funded by the National Institute of Child Health and Human Development.

Table V. The Overlapping Node Detection Ranking for $n = 5000$, $\mu = 0.3$ and High Overlap Density $O_n = 50\%$

Rank	RS_F^s	RS_F^b	RS_F^*
1	Link	Link	Link
2	UEOC	UEOC	UEOC
3	Game	SLPA	SLPA
4	SLPA	Game	Game
5	CFinder	LFM	LFM
6	LFM	CIS	CFinder
7	CIS	CFinder	CIS
8	GCE	MOSES	MOSES
9	MOSES	OSLOM	OSLOM
10	iLCD	iLCD	iLCD
11	OSLOM	COPRA	GCE
12	COPRA	EAGLE	COPRA
13	EAGLE	NMF	EAGLE
14	NMF	GCE	NMF

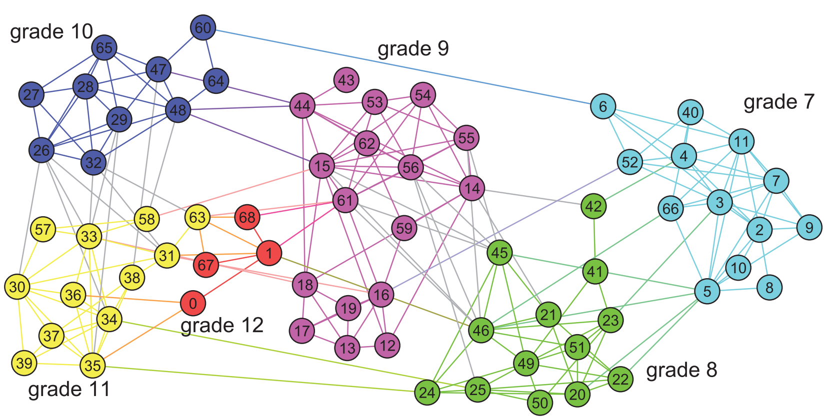

Fig. 11. High school network ($n = 69$, $\bar{k} = 6.4$). Colors represent known communities corresponding to grades ranging from 7 to 12. Grade 9 is separated into two subgroups that correspond to white (upper) and black (lower) students respectively. Numbers are the node id's.

corresponds to the grade (from 7 to 12) of students listed in the survey. The ground truth is a total of 6 communities (see Figure 11) together with two subgroups within grade 9 corresponding to a group of white and black students. Even though there are no overlapping nodes reported by the students, each algorithm reports some by its own. Results are shown in Table VI[18]. Discovered overlapping nodes are listed in the third column. For algorithms that discover more than 10 overlapping nodes, only the total number is shown. We also include NMI and the number of communities for reference.

It is easy to verify that all the overlapping nodes in Table VI are connected to at least two different groups. Some of them lie between different grades with strong connections to each individual one, for example, nodes 45, 46, 61, 26, 32, and 33. Some are boundary nodes between subgroups within a grade such as nodes 59, 12, and 18. Node 42 serves as a bridge between groups without having particular coherence to any group. However, it is still not clear whether these nodes are really meaningful or

[18]For each algorithm, we show results with parameters that output the best NMI score.

Table VI. Test on a High School Friendship Network

Algorithm	Num. of communities	Overlapping nodes	NMI	
CFinder	2	{12, 18}	0.1679	
CIS	9	total 34	0.7495	
COPRA	6	total 14	0.7966	0.697
EAGLE	4	{18}	0.4962	
Game	10	total 14	0.4673	
GCE	6	{0, 21, 45, 46, 61}	0.8333	0.768
iLCD	7	{5, 21, 26, 29, 31, 32, 33, 46, 61}	0.3713	
LFM	7	{0, 45}	0.8134	
Link	20	total 31	0.3155	
MOSES	10	total 18	0.5037	
NMF	7	{0, 12, 18, 45}	0.643	
OSLOM	11	{45, 46}	0.4315	0.659
SLPA	6	{1, 42, 45, 59}	0.6788	
UEOC	7	{0, 12, 18, 26, 29, 45}	0.8148	

Table VII. Social Networks in the Tests

Network	n	$\bar{k}$	Network	n	$\bar{k}$
karate (KR)	34	4.5	PGP	10680	4.5
football (FB)	115	10.6	Email (EM)	33696	10.7
lesmis (LS)	77	6.6	P2P	62561	2.4
dolphins (DP)	62	5.1	Epinions (EP)	75877	10.6
CA-GrQc (CA)	4730	5.6	Amazon (AM)	262111	6.8

necessary to be considered as “overlapping”. This is one factor that makes the detection (and verification) even more challenging in real-life applications.

Next, we tested on a wider range of social networks listed in Table VII. More information about these networks can be found here[19]. Given that the ground truth is not available for most of these networks, we selected two overlapping modularities Q_{ov}^{E} in (5) and Q_{ov}^{Ni} in (2) as quality measures. The former is based on the node belonging factor, and the latter is based on the link belonging factor. For the arbitrary function in Q_{ov}^{Ni}, we adopted the one used in Gregory [2010], $f(x) = 60x - 30$.

In Figures 12 through 17, networks are shown in order of increasing number of edges along the x-axis. Lines connecting points are meant merely to aid the reader in differentiating points from the same algorithm. We removed CFinder, EAGLE, and NMF from the test due to either their memory or computation inefficiency in large networks. As a reference, we also performed disjoint community detection with the Infomap algorithm [Rosvall 2008], which has been shown quite accurate in Lancichinetti and Fortunato [2009].

Figures 12 and 13 show a positive correlation between the two quality measures. Typically, the disjoint partitioning achieves higher Q_{ov}^{E} than overlapping clusterings,

[19]CA-GrQc: a coauthorship network based on papers in General Relativity publishing in Arxiv [Leskovec et al. 2007b].
PGP: a network of users of the Pretty-Good-Privacy algorithm [Boguna et al. 2004].
Email: a communication network in Enron via emails [Leskovec et al. 2009].
Epinions: a who-trust-whom online social network of a consumer review site Epinions.com [Richardson et al. 2003].
P2P: the Gnutella peer-to-peer file sharing network from August 2002 [Ripeanu et al. 2002].
Amazon: a copurchase network of the Amazon Web site [Leskovec et al. 2007a].
Data are available at http://www-personal.umich.edu/~mejn/netdata and http://snap.stanford.edu/data.

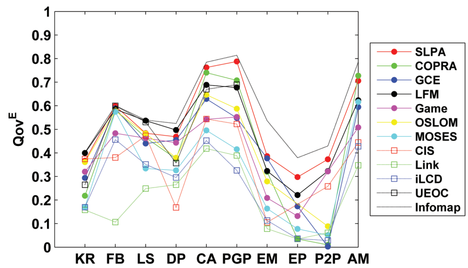

Fig. 12. Overlapping modularity Q_{ov}^{E} for social networks.

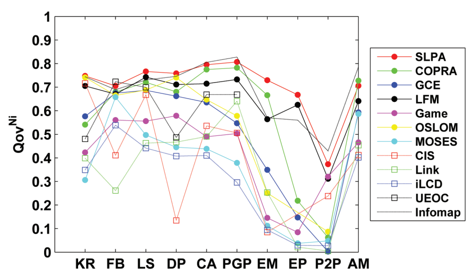

Fig. 13. Overlapping modularity Q_{ov}^{Ni} for social networks.

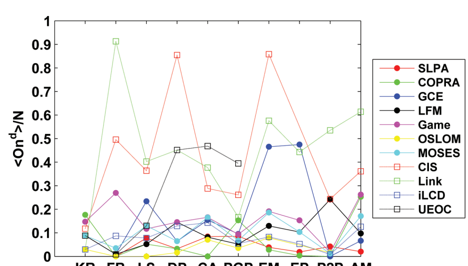

Fig. 14. The normalized number of detected overlapping nodes for social networks based on the clustering with the best Q_{ov}^{E}.

which empirically serves as a bound of the quality of detected overlapping communities. This also holds for Q_{ov}^{Ni} in general.

In general, Link and iLCD achieve lower Q_{ov}^{Ni} or Q_{ov}^{E} compared to others, while SLPA, LFM, COPRA, OSLOM, and GCE achieve higher performance on larger networks (e.g., last five networks). Moreover, an algorithm may not perform equally well on different types of network structures. Some of them are sensitive to specific structures. For example, only SLPA, LFM, CIS, and Game have satisfying performances in networks with highly sparse structure such as $P2P$, for which COPRA finds merely one single giant community and GCE also fails. Another issue is that some algorithms tend to overdetect the overlap, as was the case for LFR networks. CIS and Link fail in the test because they find too many overlapping nodes or memberships relative to the consensus shown by the other algorithms as seen in Figures 14 through 17. Such overdetection happens to other algorithms, including COPRA, GCE, and UEOC on specific networks, resulting in low performance for these algorithms.

Some interesting common features are observed from our tests. As shown in Figures 14 and 15, the fraction of overlapping nodes found by most of the algorithms

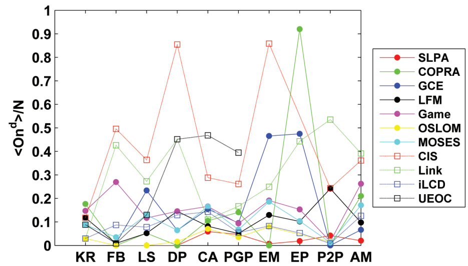

Fig. 15. The normalized number of detected overlapping nodes for social networks based on the clustering with the best Q_{ov}^{Ni}.

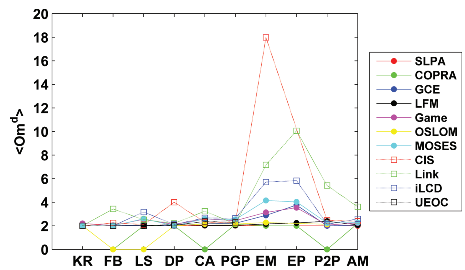

Fig. 16. The number of detected memberships for social networks based on the clustering with the best Q_{ov}^{E}.

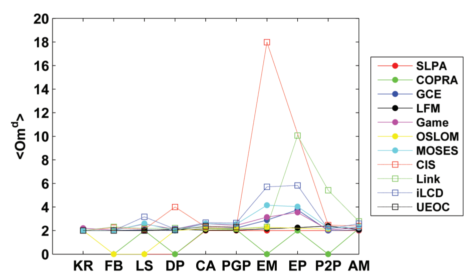

Fig. 17. The number of detected memberships for social networks based on the clustering with the best Q_{ov}^{Ni}.

is typically less than 30%. Results from SLPA, OSLOM, and COPRA, which offer good performances in the LFR benchmarks, show an even smaller fraction of overlapping nodes, less than 20%, in most real-world networks examined in this article. Moreover, Figures 16 and 17 confirm that the diversity (i.e., membership) of overlapping nodes in the tested social networks is relatively small as well, typically 2 or 3.

8. CONCLUSIONS AND DISCUSSIONS

In this article, we review a wide range of overlapping community detection algorithms along with quality measures and several existing benchmarks. A number of tests are performed on the LFR benchmarks, incorporating different network structures and various degree of overlapping. Quality evaluation is performed on both community and node levels to provide complementary information. Results show that the detection in networks with high overlapping density and high overlapping diversity still has space for improvements. The node-level evaluation reveals the problems of overdetection and underdetection which need to be considered when designing or evaluating detection algorithms. The results discovered in real-world social networks suggest the sensitivity of some algorithms to sparse networks. A common feature of social networks in view of agreement of different algorithms is the relatively small number of overlapping nodes, most of which belong just to a few communities. Moreover, the ambiguity in the definition of overlapping nodes imposes challenges in real-life applications as well.

Here, we review work that has been done mostly for unweighted networks. However, there are number of applications where a weight bears significant information (e.g., the correlation network in biological studies [Langfelder and Horvath 2008]). Algorithms that explicitly take weights into account and allow overlapping, such as CPMw and SLPAw, expect to have advantages over others.

Despite the large amount of work devoted to developing detection algorithms, there are a number of fundamental questions that have yet to be fully addressed. Two of the most prominent are when to apply overlapping methods and how significant the overlapping is.

It is natural to ask whether or not an application of the overlapping detection algorithms captures any additional information that a disjoint algorithm would necessarily miss. Unfortunately, measures like NMI and Omega do not offer a satisfying answer. The discussion on the necessity of overlap has largely been left unexplored. In Kelley [2009], the author empirically examined attributes of the vertices in a network representing commenting activity. The author suggests that, for a pair of communities A and B, the trait similarity between $A \cap B$ and the sets $A - B$ and $B - A$ be higher than the similarity between $A - B$ and $B - A$. Such a relationship might offer a way to estimate the validation of the overlap.

The significance of community structures has been previously explored only within the context of disjoint community detection and based on the notion of modularity [Reichardt and Bornholdt 2006b; Guimerà et al. 2004; Massen and Doye 2005]. The robustness and uniqueness of a discovered partitioning is also examined in Gfeller et al. [2005], Karrer et al. [2008], and Massen and Doye [2007]. Many of these techniques can be extended to assess the overlapping community structure. Interestingly, statistical significance has begun to be included in detection methodologies such as OSLOM [Lancichinetti et al. 2011].

ACKNOWLEDGMENTS

The authors would like to thank researchers who generously provided their software and helped setting up our experiments. We especially thank Steve Gregory for his helpful codes and discussions.

REFERENCES

Ahn, Y.-Y., Bagrow, J. P., and Lehmann, S. 2010. Link communities reveal multiscale complexity in networks. *Nature 466*, 761–764.

Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. 1999. Optics: Ordering points to identify the clustering structure. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. 49–60.

Arenas, A., Diaz-Guilera, A., and Perez-Vicente, C. J. 2006. Synchronization reveals topological scales in complex networks. *Phys. Rev. Lett. 96,* 11.

Ball, B., Karrer, B., and Newman, M. E. J. 2011. Efficient and principled method for detecting communities in networks. *Phys. Rev. E 84*, 3.

Baumes, J., Goldberg, M., Krishnamoorthy, M., Magdon-Ismail, M., and Preston, N. 2005. Finding communities by clustering a graph into overlapping subgraphs. In *Proceedings of the IADIS International Conference on Applied Computing*. 97–104.

Bianconi, G., Pin, P., and Marsili, M. 2008. Assessing the relevance of node features for network structure. *Proc. Natl. Acad. Sci. USA 106,* 28, 7.

Blatt, M., Wiseman, S., and Domany, E. 1996. Superparamagnetic clustering of data. *Phys. Rev. Lett. 76*, 3251–3254.

Boguna, M., Pastor-Satorras, R., Diaz-Guilera, A., and Arenas, A. 2004. Models of social networks based on social distance attachment. *Phys. Rev. E 70,* 5.

BREVE, F., ZHAO, L., AND QUILES, M. 2009. Uncovering overlap community structure in complex networks using particle competition. In *Proceedings of the International Conference on Artificial Intelligence and Computational Intelligence (AICI'09)*. 619–628.

CAMPELLO, R. J. G. B. 2007. A fuzzy extension of the rand index and other related indexes for clustering and classification assessment. *Pattern Recogn. Lett. 28*, 7, 833–841.

CAMPELLO., R. J. G. B. 2010. Generalized external indexes for comparing data partitions with overlapping categories. *Pattern Recogn. Lett. 31*, 9, 966–975.

CAZABET, R., AMBLARD, F., AND HANACHI, C. 2010. Detection of overlapping communities in dynamical social networks. In *Proceedings of the 2nd IEEE International Conference on Social Computing (SOCIALCOM'10).* 309–314.

CHEN, D., SHANG, M., LV, Z., AND FU, Y. 2010a. Detecting overlapping communities of weighted networks via a local algorithm. *Physica A 389,* 19, 4177–4187.

CHEN, J., ZAIANE, O. R., AND GOEBEL, R. 2009. A visual data mining approach to find overlapping communities in networks. In *Proceeding of the International Conference on Advances in Social Network Analysis and Mining (ASONAM'09).* 338–343.

CHEN, W., LIU, Z., SUN, X., AND WANG, Y. 2010b. A game-theoretic framework to identify overlapping communities in social networks. *Data Mining Knowl. Discov. 21*, 2, 224–240.

COLLINS, L. M. AND DENT, C. W. 1988. Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions. *Multivar. Behav. Res. 23,* 2, 231–242.

CONDON, A. AND KARP, R. M. 2001. Algorithms for graph partitioning on the planted bisection model. *Rand. Struct. Algor. 18*, 116–140.

DANON, L., DUCH, J., ARENAS, A., AND DIAZ-GUILERA, A. 2005. Comparing community structure identification. *J. Stat. Mech. Thoer. Exp. 2005,* 9.

DAVIS, G. B. AND CARLEY, K. 2008. Clearing the fog: Fuzzy, overlapping groups for social networks. *Soc. Netw. 30,* 3, 201–212.

DING, F., LUO, Z., SHI, J., AND FANG, X. 2010. Overlapping community detection by kernel-based fuzzy affinity propagation. In *Proceedings of the International Workshop on Indoor Spatial Awareness (ISA'10).* 1–4.

DU, N., WANG, B., AND WU, B. 2008. Overlapping community structure detection in networks. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM'08).* 1371–1372.

EVANS, T. 2010. Clique graphs and overlapping communities. *J. Stat. Mech.-Theor. Exp. 2010*, 12.

EVANS, T. AND LAMBIOTTE, R. 2010. Line graphs of weighted networks for overlapping communities. *Euro. Phys. J. B 77*, 265.

EVANS, T. S. AND LAMBIOTTE, R. 2009. Line graphs, link partitions and overlapping communities. *Phys. Rev. E 80*, 1.

FARKAS, I., ABEL, D., PALLA, G., AND VICSEK, T. 2007. Weighted network modules. *New J. Phys. 9,* 6, 180.

FISHER, D. C. 1989. Lower bounds on the number of triangles in a graph. *J. Graph Theor. 13,* 4, 505–512.

FORTUNATO, S. 2010. Community detection in graphs. *Phys. Rep. 486*, 75–174.

FREY, B. J. AND DUECK, D. 2007. Clustering by passing messages between data points. *Sci. 315*, 972–976.

FU, Q. AND BANERJEE, A. 2008. Multiplicative mixture models for overlapping clustering. In *Proceedings of the International Conference on Data Mining (ICDM'08)*. 791–796.

GEWENIGER, T., ZUHLKE, D., HAMMER, B., AND VILLMANN, T. 2009. Fuzzy variant of affinity propagation in comparison to median fuzzy c-means. In *Proceedings of the 7th International Workshop on Self Organizing Maps (WSOM'09)*. 72–79.

GFELLER, D., CHAPPELIER, J.-C., AND DE LOS RIOS, P. 2005. Finding instabilities in the community structure of complex networks. *Phys. Rev. E 72*, 5.

GIRVAN, M. AND NEWMAN, M. E. J. 2002. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA 99,* 12, 7821–7826.

GREGORY, S. 2007. An algorithm to find overlapping community structure in networks. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases.* 91–102.

GREGORY, S. 2008. A fast algorithm to find overlapping communities in networks. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases.* Lecture Notes in Computer Science, vol. 5211, Springer, 408–423.

GREGORY, S. 2009. Finding overlapping communities using disjoint community detection algorithms. *CompleNet 207*, 47–61.

GREGORY, S. 2010. Finding overlapping communities in networks by label propagation. *New J. Phys. 12*, 10.

GREGORY, S. 2011. Fuzzy overlapping communities in networks. *J. Stat. Mech. 2011,* 2.

GUIMERA, R., SALES-PARDO, M., AND AMARAL, L. A. N. 2004. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E 70*, 2.

HAVEMANN, F., HEINZ, M., STRUCK, A., AND GLASER, J. 2011. Identification of overlapping communities and their hierarchy by locally calculating community-changing resolution levels. *J. Statist. Mech. 2011,* 1.

HUBERT, L. AND ARABIE, P. 1985. Comparing partitions. *J. Classif. 2*, 193–218.

HULLERMEIER, E. AND RIFQI, M. 2009. A fuzzy variant of the rand index for comparing clustering structures. In *Proceedings of the Joint International Fuzzy Systems Association World Congress and European Society of Fuzzy Logic and Technology Conference*. 1294–1298.

JIN, D., YANG, B., BAQUERO, C., LIU, D., HE, D., AND LIU, J. 2011. A markov random walk under constraint for discovering overlapping communities in complex networks. *J. Statist. Mech. 2011,* 5.

KARRER, B., LEVINA, E., AND NEWMAN, M. E. J. 2008. Robustness of community structure in networks. *Phys. Rev. E 77*, 4.

KELLEY, S. 2009. The existence and discovery of overlapping communities in large-scale networks. Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, NY.

KELLEY, S., GOLDBERG, M., MAGDON-ISMAIL, M., MERTSALOV, K., AND WALLACE, A. 2011. Defining and discovering communities in social networks. In *Handbook of Optimization in Complex Networks*, Springer, 139–168.

KIM, Y. AND JEONG, H. 2011. The map equation for link community (unpublished). http://stat.kaist.ac.kr/~hjeong/papers/2011_Map.pdf.

KOVACS, I. A., PALOTAI, R., SZALAY, M., AND CSERMELY, P. 2010. Community landscapes: An integrative approach to determine overlapping network module hierarchy, identify key nodes and predict network dynamics. *PLoS ONE 5,* 9.

KUMPULA, J. M., KIVELA, M., KASKI, K., AND SARAMAKI, J. 2008. Sequential algorithm for fast clique percolation. *Phys. Rev. E 78,* 2.

LANCICHINETTI, A. AND FORTUNATO, S. 2009. Community detection algorithms: A comparative analysis. *Phys. Rev. E 80*, 5.

LANCICHINETTI, A., FORTUNATO, S., AND KERTESZ, J. 2009. Detecting the overlapping and hierarchical community structure of complex networks. *New J. Phys. 11*, 3.

LANCICHINETTI, A., FORTUNATO, S., AND RADICCHI, F. 2008. Benchmark graphs for testing community detection algorithms. *Phys. Rev. E 78*, 4.

LANCICHINETTI, A., RADICCHI, F., RAMASCO, J. J., AND FORTUNATO, S. 2011. Finding statistically significant communities in networks. *PLoS ONE 6,* 4.

LANGFELDER, P. AND HORVATH, S. 2008. WGCNA: An r package for weighted correlation network analysis. *BMC Bioinf. 1*, 559.

LATOUCHE, P., BIRMELE, E., AND AMBROISE, C. 2011. Overlapping stochastic block models with application to the french political blogosphere. *Annals Appl. Statist. 5*, 309–336.

LEE, C., REID, F., MCDAID, A., AND HURLEY, N. 2010. Detecting highly overlapping community structure by greedy clique expansion. In *Proceedings of the 4th Workshop on Social Network Mining and Analysis held in Conjunction with the International Conference on Knowledge Discovery and Data Mining (SNA/KDD'10)*. 33–42.

LESKOVEC, J., ADAMIC, L. A., AND HUBERMAN, B. A. 2007a. The dynamics of viral marketing. *ACM Trans. Web 1*, 5.

LESKOVEC, J., KLEINBERG, J., AND FALOUTSOS, C. 2007b. Graph evolution: Densification and shrinking diameters. *ACM Trans. Knowl. Disc. Data 1*, 1.

LESKOVEC, J., LANG, K. J., DASGUPTA, A., AND MAHONEY, M. W. 2009. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Math. 6*, 29–123.

LESKOVEC, J., LANG, K. J., AND MAHONEY, M. W. 2010. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th Conference on World Wide Web (WWW'10)*. 631–640.

LI, D., LEYVA, I., ALMENDRAL, J., SENDINA-NADAL, I., BULDU, J., HAVLIN, S., AND BOCCALETTI, S. 2008. Synchronization interfaces and overlapping communities in complex networks. *Phys. Rev. Lett. 101*, 16.

LU, Q., KORNISS, G., AND SZYMANSKI, B. K. 2009. The naming game in social networks: Community formation and consensus engineering. *J. Econ. Interact. Coord. 4*, 221–235.

MAGDON-ISMAIL, M. AND PURNELL, J. 2011. Fast overlapping clustering of networks using sampled spectral distance embedding and gmms. Tech. rep., Rensselaer Polytechnic Institute, Troy, NY.

MASSEN, C. AND DOYE, J. 2005. Identifying communities within energy landscapes. *Phys. Rev. E 71*, 4.

MASSEN, C. AND DOYE, J. 2007. Thermodynamics of community structure. Preprint arXiv:con-mat/0610077v1.

MCDAID, A. AND HURLEY, N. 2010. Detecting highly overlapping communities with model-based overlapping seed expansion. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM'10)*. 112–119.

MOLLOY, M. AND REED, B. 1995. A critical point for random graphs with a given degree sequence. *Rand. Struct. Algor. 6*, 161–179.

MOON, J. AND MOSER, L. 1965. On cliques in graphs. *Israel J. Math. 3*, 23–28.

NEPUSZ, T., PETROCZI, A., NEGYESSY, L., AND BAZSO, F. 2008. Fuzzy communities and the concept of bridgeness in complex networks. *Phys. Rev. E 77*, 1.

NEWMAN, M. E. J. 2006. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E 74*, 3.

NEWMAN, M. E. J. AND LEICHT, E. A. 2007. Mixture models and exploratory analysis in networks. *Proc. Natl. Acad. Sci. USA 104*, 9564–9569.

NEWMAN, M. E. J., STROGATZ, S. H., AND WATTS, D. J. 2001. Random graphs with arbitrary degree distributions and their applications. *Phys. Rev. E 64,* 2.

NICOSIA, V., MANGIONI, G., CARCHIOLO, V., AND MALGERI, M. 2009. Extending the definition of modularity to directed graphs with overlapping communities. *J. Stat. Mech. 2009,* 3.

NOWICKI, K. AND SNIJDERS, T. A. B. 2001. Estimation and prediction for stochastic blockstructures. *J. Amer. Statist. Assoc. 96,* 455, 1077–1087.

PADROL-SUREDA, A., PERARNAU-LLOBET, G., PFEIFLE, J., AND MUNTS-MULERO, V. 2010. Overlapping community search for social networks. In *Proceedings of the 26th International Conference on Data Engineering (ICDE'10)*. 992–995.

PALLA, G., DERENYI, I., FARKAS, I., AND VICSEK, T. 2005. Uncovering the overlapping community structure of complex networks in nature and society. *Nature 435*, 814–818.

PSORAKIS, I., ROBERTS, S., EBDEN, M., AND SHELDON, B. 2011. Overlapping community detection using bayesian non-negative matrix factorization. *Phys. Rev. E 83,* 6.

RAGHAVAN, U. N., ALBERT, R., AND KUMARA, S. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E 76*, 3.

REES, B. AND GALLAGHER, K. 2010. Overlapping community detection by collective friendship group inference. In *Proceedings of the International Conference on Advances in Social Network Analysis and Mining (ASONAM'10)*. 375–379.

REICHARDT, J. AND BORNHOLDT, S. 2004. Detecting fuzzy community structures in complex networks with a potts model. *Phys. Rev. Lett. 93*, 2.

REICHARDT., J. AND BORNHOLDT, S. 2006a. Statistical mechanics of community detection. *Phys. Rev. E 74,* 1.

REICHARDT, J. AND BORNHOLDT, S. 2006b. When are networks truly modular? *Physica D224*, 20–26.

REID, F., MCDAID, A. F., AND HURLEY, N. J. 2011. Partitioning breaks communities. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining (ASONAM'11)*. 102–109.

REN, W., YAN, G., LIAO, X., AND XIAO, L. 2009. Simple probabilistic algorithm for detecting community structure. *Phys. Rev. E 79,* 3.

RICHARDSON, M., AGRAWAL, R., AND DOMINGOS, P. 2003. Trust management for the semantic web. In *Proceedings of the 2nd International Semantic Web Conference (ISWC'03)*. Lecture Notes in Computer Science, vol. 2870. Springer, 351–368.

RIPEANU, M., FOSTER, I., AND IAMNITCHI, A. 2002. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Comput. J. 6*, 1, 50–57.

RONHOVDE, P. AND NUSSINOV, Z. 2009. Multiresolution community detection for megascale networks by information-based replica correlations. *Phys. Rev. E 80*, 1.

ROSVALL, M. AND BERGSTROM, C. T. 2008. Maps of random walks on complex networks reveal community structure. *Proc. Natl. Acad. Sci. 105*, 1118–1123.

SAWARDECKER, E., SALES-PARDO, M., AND AMARAL, L. 2009. Detection of node group membership in networks with group overlap. *Euro. Phys. J. B67*, 277.

SHEN, H., CHENG, X., CAI, K., AND HU, M.-B. 2009a. Detect overlapping and hierarchical community structure. *Physica A388*, 1706.

SHEN, H., CHENG, X., AND GUO, J. 2009b. Quantifying and identifying the overlapping community structure in networks. *J. Stat. Mech. 2009,* 7, 9.

WANG, X., JIAO, L., AND WU, J. 2009. Adjusting from disjoint to overlapping community detection of complex networks. *Physica A388*, 5045–5056.

WHITE, S. AND SMYTH, P. 2005. A spectral clustering approach to finding communities in graphs. In *Proceedings of the SIAM International Conference on Data Mining*. 76–84.

WU, Z., LIN, Y., WAN, H., AND TIAN, S. 2010. A fast and reasonable method for community detection with adjustable extent of overlapping. In *Proceedings of the Conference on Intelligent Systems and Knowledge Engineering (ISKE'10)*. 376–379.

XIE, J. AND SZYMANSKI, B. K. 2011. Community detection using a neighborhood strength driven label propagation algorithm. In *Proceedings of the IEEE Network Science Workshop (NSW'11)*. 188–195.

XIE, J. AND SZYMANSKI, B. K. 2012. Towards linear time overlapping community detection in social networks. In *Proceedings of the 16th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD'12)*. 25–36.

XIE, J., SZYMANSKI, B. K., AND LIU, X. 2011. SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process. In *Proceedings of the 11th IEEE International Conference on Data Mining Workshops (ICDMW'11)*. 344–349.

ZAREI, M., IZADI, D., AND SAMANI, K. A. 2009. Detecting overlapping community structure of networks based on vertex-vertex correlations. *J. Stat. Mech. 2009,* 11.

ZHANG, S., WANG, R.-S., AND ZHANG, X.-S. 2007a. Identification of overlapping community structure in complex networks using fuzzy c-means clustering. *Physica A374*, 483–490.

ZHANG, S., WANG, R.-S., AND ZHANG, X.-S. 2007b. Uncovering fuzzy community structure in complex networks. *Phys. Rev. E 76,* 4.

ZHANG, Y., WANG, J., WANG, Y., AND ZHOU, L. 2009. Parallel community detection on large networks with propinquity dynamics. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'09)*. 997–1006.

ZHAO, K., ZHANG, S.-W., AND PAN, Q. 2010. Fuzzy analysis for overlapping community structure of complex network. In *Proceedings of the Chinese Control and Decision Conference (CCDC'10)*. 3976–3981.

Received October 2011; revised May 2012; accepted June 2012