Universiteit
Leiden

Master Computer Science

Analyzing the Performance of Community Detection Methods: A Group Fairness Approach

Name:	Elze de Vink
Student ID:	s1708058
Date:	21/11/2024
Specialization:	Artificial Intelligence
1st supervisor:	Dr. Akrati Saxena
2nd supervisor:	Dr. Frank Takes

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Community detection (CD) offers a deep insight into network structure and nodes' behavior. In social networks, communities represent densely connected social groups that vary in size, density, and level of connectivity. CD methods identify communities by optimizing community quality measures, utilizing spectral properties, or employing alternative approaches. Real-world networks have structural inequalities which influence how well a CD method is able to identify communities. CD methods that overlook structural inequalities produce biased outcomes; however, the types of communities that are disadvantaged and the extent of this disadvantage have not yet been researched. This thesis introduces a novel group fairness metric to measure bias with respect to community properties size, density, and conductance. We define fairness as equity and propose community-based performance metrics that measure how well each ground-truth community is detected by the CD method. Trends in community-wise performance are quantified by our fairness metric, giving valuable insights that can help design fair CD methods, improve current CD methods, and set the parameters of CD methods. Experiments on real-world and synthetic networks show that CD methods have a bias towards large, dense, and low-conductance communities. This is not the case for certain modularity optimization methods, which perform better at identifying sparse communities.

Contents

Chapter 1

Introduction

Networks are a way in which complex systems can be studied, describing the relationships between network components in biological [31, 73], technological [1, 16], information [78], and social networks [21, 47, 79, 94]. In networks, nodes are connected by edges to form groups, also called communities [86]. Although there is no clear definition of what a community is, the literature agrees that communities are groups of connected nodes that are more likely to connect to each other than to other nodes in the network [4, 19].

Social networks have structural inequalities. Network evolution is strongly driven by people's ethnicity, gender, race, age, or wealth because people form connections based on similarity [48]. The homophilic principle, where nodes are more likely to connect to similar nodes, influences community structure. Structural inequalities in networks arise because communities in the real world also have different properties. In a population, there may be ethnic minority communities that are smaller in size, villages where everyone knows each other and are densely connected, and migrant communities that are less integrated into other parts of the network. If these structural inequalities are not taken into account, biased outcomes can develop for minorities or other groups. Algorithms that do not take biases into account are called fairness-oblivious algorithms. For example, in influence maximization, where bias is shown to exist [84, 90], communities that are smaller or less connected throughout the network are at risk of not being reached. Similarly, in community detection (CD), minority communities could not be properly detected, or assimilated into larger communities by some CD method. There are measures that compare the predicted partition to the ground-truth partition, some of which are based on node overlap. A CD method that is able to detect majority communities well will score highly for such measures, as the majority communities have a larger share of nodes than the minority communities. It is important to develop fairness-aware CD methods that perform well for all types of communities.

Fairness can be described from different perspectives in social network analysis (SNA) [77]. In the context of community detection, we define group fairness based on equity. We declare a CD method to be fair if all communities are found equally well. Because networks contain communities that vary in size, density, and connectivity within the network, we can look at group fairness with respect to these community attributes. Fair community detection is especially important because it is used in many other fair SNA problems. These include fairness-aware methods proposed for influence maximization [5, 17, 90], influence minimization [74], link prediction [75, 76], and centrality ranking [85, 91, 92]. These fairness-aware methods make use of community membership to arrive at their final fair outcome. Fair community detection aims to mitigate bias against individuals and groups, regardless of community properties.

How can bias in community detection be measured? Biases related to community properties have not yet been studied. In order to measure these biases and quantify their extent, this thesis introduces a group fairness metric for communities. For this, we use metrics that measure how well a ground-truth community is detected. We name these community-wise performance metrics (CPM). We introduce several community-wise performance measures, looking at both nodes and edges. The CPMs are divided into two categories: mapped and global metrics. Mapped metrics work by mapping ground-truth communities to predicted communities. This is done iteratively; we map the communities based on Jaccard similarity. The mapped metrics measure how well the mapped-to predicted community describes the ground-truth community. Our proposed mapped metrics include: (i) the Fraction of Correctly Classified Nodes (FCCN), (ii) the F1 score, (iii) the Fraction of Correctly Classified Edges (FCCE), and (iv) FCCE+. Global metrics compare a single ground-truth community to the entire predicted partition. We propose these methods because a difference between the number of ground-truth communities and the number of predicted communities leads to loss of information or noise. The proposed global metrics include: (i) the Average F1-score (AF1), (ii) the Sum of Weighted F1 (SWF1), (iii) the Sum of Weighted FCCE (SWFCCE), and (iv) the Sum of Weighted FCCE+ (SWFCCE+). These community-wise performance metrics and ground-truth community property values require ground-truth knowledge of the community structure.

Using the scores of the community-wise performance metrics, we create a group fairness metric Φ_p^{CPM}. The metric Φ_p^{CPM} measures the bias in relation to the community properties (p) size, density, and conductance, using one of the CPMs that we propose. A regression line is drawn through CPM scores. These scores are plotted against the min-max normalized values of community property p. The angle of a regression line is then calculated; this is scaled to create results of Φ that fall in the range (-1, 1). The sign of Φ signifies the type of communities that are favored by a CD method. E.g., when analyzing bias regarding community size, a negative value of Φ would mean the CD method favors minority communities. A positive value would mean the CD method favors majority communities. A Φ value of 0 would mean the regression line is horizontal; no bias trend is found and the method is fair. All communities, irrespective of type, are detected with equal accuracy, whether that accuracy is high or low.

Together with metrics that measure the quality of the predicted community partition, such as normalized mutual information (NMI) [20] and variation of information (VI) [51], the fairness metric Φ can be used to do the performance-fairness trade-off. We use the group fairness metric Φ to compare 24 CD methods in terms of fairness. We divide the CD methods into six categories: (i) Optimization, (ii) Spectral, (iii) Propagation, (iv) Dynamics, (v) Representation Learning, and (vi) Miscellaneous. Experiments are performed on real-world networks and synthetic networks, these are generated using the LFR benchmark [40], ABCD [32], and the HICH-BA model [74]. These models generate undirected, unweighted, connected networks with non-overlapping communities and the CD methods we analyze are suitable for such networks.

1.1 Contributions

Our contributions are summarized as follows:

- We compare eight community-wise performance metrics, separated into mapped and global metrics. These metrics quantify how well a ground-truth community is identified by the mapped predicted community (mapped metrics) or by the entire predicted community partition (global metrics).

- We introduce the group fairness metric Φ_p^{CPM}, which measures the bias regarding ground-truth community property p using community-wise performance metric *CPM*.

- We provide insights into community detection methods and on how they perform with respect to both performance and fairness, showcasing the performance-fairness trade-off. Our experiments and results are used to get a better understanding of existing community detection methods, which will facilitate the improvement and design of fair community detection methods.

- We provide guidance on what specific variant of the fairness metric Φ_p^{CPM} to prioritize, depending on the specific social network problem being addressed.

Using the group fairness metric Φ_p^{CPM} we find CD methods that are both fair and have high performance. We find that with LFR networks, most CD methods favor large, dense, and lowly-connected communities; this bias is increased as the community structure is less defined throughout the network. A group of modularity-based CD methods favor less-dense communities. This group includes Combo, Leiden, Louvain, RB-C, and RB-ER. We find that the trend between high performance and a larger bias toward dense, low-conductance communities is smaller for the ABCD networks. This may be because the overall performance on these networks is higher.

1.2 Thesis Overview

The rest of this work follows the following structure: Chapter 3 gives preliminary information and Chapter 2 present the relevant works; these chapters required background knowledge on networks, communities, and algorithmic fairness in SNA. Chapter 4 introduces the proposed fairness metric Φ and multiple community-wise performance metrics. The experimental setup in Chapter 5 describes the 24 analyzed community detection methods and network datasets including both real-world and synthetic networks. Chapter 6 displays and explains our results, and Chapter 7 offers a discussion of our results and proposes future directions of similar research in this area.

Chapter 2

Related Work

Algorithmic bias based on "sensitive" or "protected" attributes such as gender, ethnicity, race, and sexual orientation has been found in many systems. Examples include gender [35], language [44], and political bias [98] in language models, gender bias in ad delivery systems [37]. Fairness can be defined in different ways, some of which even contradict each other [22]. Fairness definitions have been described in [93] and in surveys on fairness in machine learning [9, 50, 63].

This chapter will primarily survey related work on fairness in SNA. Saxena et al. [77] review state-of-the-art for multiple research topics in SNA, going over the specific fairness constraint for each topic and highlighting areas where fairness has not been addressed. Community detection is one of these areas highlighted, and we hope that this work helps develop more research on fairness in community detection.

2.1 Fairness in Social Network Analysis

Fairness in SNA can be addressed in different ways, depending on the task at hand and the approach. The following sections will explain the fairness constraints and fair approaches of works in the respective SNA topics. These works use groups or communities in their fair approaches, highlighting the importance of fair community detection as well.

2.1.1 Fair Community Detection

Mehrabi et al. [49] state that low-degree nodes are excluded by CD algorithms, particularly those that optimize modularity. The authors propose a method Communities with Lowly-connected Attributed Nodes (CLAN) that works on networks with attributed nodes. This method uses a supervised learning step to place attributed nodes from smaller predicted communities into larger communities. While the authors remove these smaller predicted communities with the assumption that they will be used in downstream tasks, actual minority communities will be dissolved using this method instead of being correctly classified. Although this paper does not provide a definition for fairness, it proposed a method against an observed bias.

Manolis et al. [46] introduce two metrics that measure fairness for communities: balance fairness and modularity fairness. The analyzed networks have two disjunct groups of nodes, depicted by blue (B) and red (R), where red nodes are the protected group. Balance fairness measures how much the fraction of red nodes in a community c_i differs from the fraction of red nodes in the entire network, which is given by $\phi = \frac{|R|}{|V|}$. The set of red nodes in community c_i is given by $R(c_i)$ so balance fairness is given by:

$$f_{balance}(c_i) = \frac{|R(c_i)|}{|C_i|} - \phi \tag{2.1}$$

The second fairness metric compares how well red and blue are connected within a community. The connectedness of a group is given by the modularity of the group: $Q^B_{c_i}$ for blue and $Q^R_{c_i}$ for red [46]. Modularity fairness is given by:

$$f_{modularity}(c_i) = \frac{Q^R_{c_i} - Q^B_{c_i}}{abs(Q_{c_i})} \tag{2.2}$$

These methods measure whether nodes from the protected group are sufficiently represented and connected within each community. Using synthetic networks, they find that group size imbalance is the largest influence on both balance fairness and modularity fairness.

Detectability of communities could be a concern for fairness in community detection. Certain detectability thresholds [15, 19, 52] have been proposed to define communities that are undetectable under some conditions. Radicchi [65] finds that heterogeneity in degree distribution allows CD algorithms optimizing modularity to correctly recover the community structure of the network. In complex networks, like those generated using the LFR benchmark and ABCD, it is not clear whether there exists a non-trivial threshold for detectability [19]. Undetectability due to community properties could lead to a biased outcome in community detection.

Oostenbach [58] wrote his thesis on the fairness analysis of CD methods. We improve on his work by introducing a fairness metric that does not require splitting the set of ground-truth communities into "smaller" and "larger" groups by setting a threshold. We also propose a different way of mapping ground-truth communities to predicted communities, using an iterative approach instead of the greedy algorithm employed by Oostenbach. We expand upon Oostenbach's list of mapped community-wise performance metrics and introduce global metrics which look at the entire predicted community partition. This global approach removes noise introduced by fully misclassified ground-truth communities and the omitting of predicted communities that are not mapped to. Lastly, our approach not only measures the extent of bias, but also shows what property values the community detection method favor, like minority/majority when analyzing community size.

2.1.2 Fair Influence Maximization

A node embedding algorithm that is based on random walks like Node2Vec [24] and DeepWalk [62] can be enhanced by the CrossWalk method introduced by Khajehnejad et al. [34]. These node representations can not only be used for influence maximization, but also for link prediction, and community detection. This method works by adding additional weights to edges that are close to the community's periphery or are intercommunity edges. This paper aims to lower the performance disparity between communities. This can be formulated as:

$$disparity = Var(\{Q_i\} : i \in C) \tag{2.3}$$

Here, Q_i the performance of community $c_i \in C$, the set of communities. For influence maximization, this Q is the fraction of nodes that are infected at the end of the simulation. With experiments on real-world and synthetic networks, the authors find that at a small cost of performance, the CrossWalk method enhances fairness for multiple social network tasks including influence maximization.

Stoica et al. [83] describe two fairness constraints for influence maximization: fairness in seeding and fairness in outreach; both are instances of statistical parity. They employ a greedy approach and a degree-based approach to select seed nodes. They find that when the top-k nodes in each community are selected as seed nodes, by having differentiated thresholds per community, this results in higher outreach with more diversity.

2.1.3 Fair Influence Minimization

Influence minimization, or influence blocking maximization, can be approached in two ways: (i) by identifying users that, when blocked or immunized, will minimize the spread of misinformation, and (ii) by "truth-campaigning". Truth-campaigning is the spread of counter-truth information to help people find the correct information. This is the route taken by Saxena et al. [74], who created a method called FWRRS (Fairness-aware Weighted Reversible Reachable System), which approximates the blocking power of each node to select those that achieve the best and fairest outcome. Fairness is considered as the maximin fairness constraint, which aims to maximize the number of saved nodes in the community with the lowest fraction of saved nodes. Saved nodes are those that believe in the true information while they would have believed the false information if they had not been saved.

2.1.4 Fair Link Prediction

Communities play a large role in link prediction. Saxena et al. [75] explain the intuition that the probability of two nodes being connected, is dependent on the nodes' community membership. They find that intercommunity links have lower structural similarities than intracommunity links and their link prediction framework, called HM-EIICT (Heuristic Method-Extended using Intra and Inter Community Thresholds), takes this into account by setting different threshold values for inter and intracommunity edges based on the structural properties of the network. The method outperforms baseline methods for intercommunity edge prediction and accuracy.

The authors of NodeSim [76] have developed a network embedding method that captures similarities between nodes and the community structure. It does so by performing random walks where moving to another node is not only decided based on node similarity, but also by community membership. The authors proposed a link prediction method that predicts inter and intracommunity links with high accuracy by training a logistic regression model using node pair embedding and community membership information. While this method outperforms baseline methods for both inter and intracommunity edges, further prioritizing intercommunity edges when creating random walks improves the prediction of intercommunity edges even further.

2.1.5 Fair Centrality Ranking

PageRank [8] is a link analysis algorithm that determines the relative importance of nodes. It produces weights based on random walks, which can be used to rank nodes to perform tasks like search result ranking. PageRank has been modified by Tsioutsiouliklis et al. [92] to create two fair approaches: fairness-sensitive PageRank, and locally fair PageRank. The authors define a fairness constraint ϕ-fairness that is satisfied when the fraction of the total weight of the protected group is ϕ. By setting ϕ as the fraction of protected nodes in the graph, it requires that protected nodes have a proportional share of the total weights. Fairness-sensitive PageRank tries to adjusts the jump vector of the random walker, specifically to achieve ϕ fairness. The locally fair PageRank works by using the fairness-ratio ϕ on a node level. The nodes distribute their own pagerank score to protected and unprotected nodes according to ϕ.

Chapter 3

Preliminaries

We have a network $G(V, E)$, with V as the set of nodes and E the set of edges. These terms are often referred to by other names, depending on the context. Nodes can be referred to by vertices and points, edges can be called links, connections, and ties, and a network can be referred to by a graph. In this work, we look at undirected, unweighted, and connected networks with nonoverlapping, fully covering communities. An undirected network has edges that have no direction and unweighted means that edges have no associated weight or value. In a connected network, there exists a path between every pair of nodes. Nonoverlapping, fully covering communities mean that each node belongs to exactly one community, and together, all communities create a complete partition of the network.

3.1 Community Attributes

For this research, we analyze network data with ground-truth information on the community structure where each node is labeled with its community membership. We evaluate the bias of CD methods with respect to communities' properties, including community size, (internal) density, and conductance. These are defined as follows for a community c_i [10]:

- Size: number of nodes in the community, $|c_i|$.
- Internal density: the fraction of actual intracommunity edges out of the number of possible intracommunity edges.

$$density(c_i) = \frac{E_{c_i}^{in}}{\frac{1}{2}|c_i|(|c_i| - 1)} \tag{3.1}$$

- Conductance: the fraction of intercommunity edges out of the community's total edge volume.

$$conductance(c_i) = \frac{E_{c_i}^{out}}{2E_{c_i}^{in} + E_{c_i}^{out}} \tag{3.2}$$

Density gives a measure of how strongly connected members of a community are. If a community has a density of 1 it means that it is fully connected, and lower values indicate that it is more sparsely connected. Conductance gives insight into how connected the community is to other parts

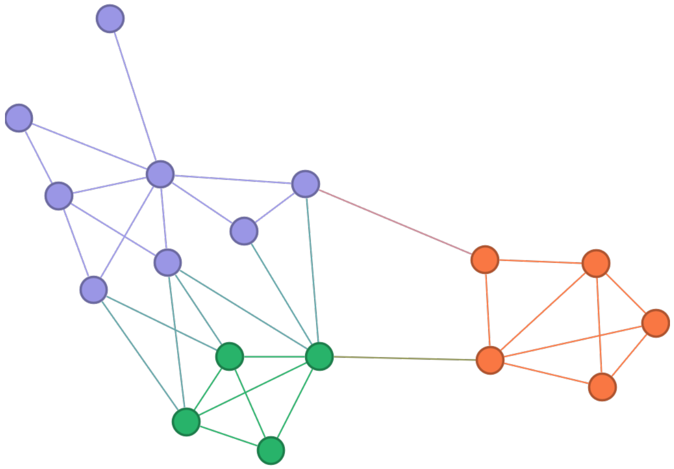


Figure 3.1: Example network with three communities. Blue nodes are in c_1, orange nodes in c_2, and green nodes in c_3.

of the network. Lower conductance indicates that the community is more separated from the rest of the network, and higher conductance means a higher level of connectivity.

Looking at the example in Figure 3.1, we can gather the size, density, and conductance values for the three communities, c_1 (blue), c_2 (orange), and c_3 (green). These values are shown in Table 3.1. We see that c_3 is fully connected and has the highest density, it also has the highest conductance by having the highest share of intercommunity edges out of its edge volume. The least connected community is c_2 with a conductance value of 0.11.

3.2 Notation

Table 3.2 gives a summary of the notations used throughout this work.

Community	$E_{c_i}^{in}$	$E_{c_i}^{out}$	Size	Density	Conductance
c_1	11	8	8	0.37	0.27
c_2	8	2	5	0.8	0.11
c_3	6	8	4	1	0.4

Table 3.1: Attribute values for the three communities in the example network from Figure 3.1.

Notation	Definition
$G(V, E)$	Network G with nodes V and edges E
V	Set of nodes
E	Set of edges, $(u, v) \in E$
u, v	Nodes in V
(u, v)	An edge in E, between nodes u and v
N	Number of nodes, $N = \|V\|$
C	Set of ground-truth communities
P	Set of predicted communities
c_i	A set of nodes that represent the ground-truth community i, $c_i \in C$
p_j	A set of nodes that represent the predicted community j, $p_j \in P$
$\|c_i\|$	Number of nodes in community c_i
m	Number of ground-truth communities: $m = \|C\|$
k	Number of predicted communities: $k = \|P\|$
p	Community property. Size, density, or conductance

Table 3.2: Notation for graphs and communities used in this work.

Chapter 4

The Proposed Metrics

In this chapter, we introduce metrics to compare group fairness of different CD methods. Several methods exist to assess the quality of predicted partitions, such as NMI [20] and VI [51]. However, no metric has yet been proposed to measure bias in the detection of ground-truth communities. To analyze group fairness, we first aim to measure how well a community is detected by the predicted partition, and we name these metrics community-wise performance metrics (CPM). These metric scores are used to create a group fairness method Φ_p^{CPM}, which is able to analyze the bias of CD methods toward communities based on properties (p) such as size, density, and conductance. These metrics help us understand to what extent the algorithm favors certain types of communities over others, providing a comprehensive assessment of its fairness.

To start, we will describe the proposed CPMs that measure CD performance on a per-community basis. These are categorized in mapped and global metrics. Then we describe how these metrics are used to compute the fairness metric Φ.

4.1 Community-wise Performance Metrics

Using ground-truth community data, we can measure how well a CD method has found the entire community structure with metrics such as NMI [20], ARI [29], VI [51], and others [10]. Here, we want to measure how well each individual ground-truth community is detected by the CD method, looking at both the community nodes and edges.

We have the given network $G(V, E)$ that has m ground-truth communities, defined as $C = \{c_1, c_2, ..., c_m\}$. We apply a CD method to G that gives a set of predicted communities P of size k defined as $P = \{p_1, p_2, ..., p_k\}$. One way of gathering per-community metrics is by mapping ground-truth communities to predicted communities, which is most effective when the number of ground-truth communities matches the number of predicted communities. The ground-truth community c_i is compared to its mapped-to predicted community p_j. The second method, which we call the global approach, evaluates how well each ground-truth community is represented by the entire set of predicted communities. In this case, each community c_i is compared to P.

All metrics should be normalized to fall within the range from 0 to 1. Receiving a score of 1 means that for a ground-truth community c_i there exists a predicted community p_j for which $c_i = p_j$ holds.

4.1.1 Community Mapping

Mapping ground-truth communities to predicted communities takes place in the following iterative process:

1. Similarity is calculated for each pair of ground-truth and predicted communities ($C \times P$).
2. The highest similarity score is chosen and the corresponding pair of ground-truth and predicted community is mapped. To break ties of equal similarity scores, a pair is randomly chosen.
3. As this mapping is one-to-one, all of the similarity scores of the mapped ground-truth and predicted communities are no longer considered.
4. If there are still both unmapped ground-truth and unmapped-to predicted communities, return to step 2.

After all ground-truth communities have a mapping or if there are no more predicted communities to map to, the mapping process is stopped. Ground-truth communities that have no mapping are marked as completely misclassified. The Jaccard Similarity Coefficient [67] is chosen as the similarity scoring function and is defined in Equation 4.1.

$$J(c_i, p_j) = \frac{|c_i \cap p_j|}{|c_i \cup p_j|} \tag{4.1}$$

4.1.2 Problems with Mapping

Mapping ground-truth communities to predicted communities allows for simple comparisons and metrics but leads to several issues when the number of predicted communities does not perfectly align with the number of ground-truth communities. Let us look at the three scenario's relating to the (in)equality of $m = |C|$ and $k = |P|$:

- $m = k$: All ground-truth communities have a mapped-to community in the set of predicted communities. All predicted communities are mapped to. There is a potential for perfect community detection.
- $m < k$: If the number of predicted communities is greater, that means that one or more predicted communities are not mapped to, causing all information contained in those predicted communities to be disregarded.
- $m > k$: If the number of predicted communities is smaller, one or more ground-truth communities have no mapping and are fully misclassified. The scores of these ground-truth communities would all be 0, which drastically changes the CDMs bias analysis.

To remedy these issues, we can set the correct number of ground-truth communities for several CDMs for which the number of communities is a parameter. Modifications could be made to other CDMs for which this information is not yet given as a parameter. However, providing this information significantly impacts the performance of the CDMs, but typically this information is not known beforehand. A way to solve this issue is to stop mapping ground-truth communities to predicted communities and instead consider all predicted communities while analyzing how well each ground-truth community is discovered. For this, we propose global metrics.

4.1.3 Mapped Metrics

We propose four mapped CPMs: FCCN, F1, FCCE, and FCCE+.

1. **Fraction of Correctly Classified Nodes (FCCN):** A simple metric to see how well a predicted community represents the ground-truth community is the fraction of ground-truth nodes present in the predicted community. This metric shares the definition of recall [81] and is defined as follows:

   $$FCCN(c_i, p_j) = recall(c_i, p_j) = \frac{|c_i \cap p_j|}{|c_i|} \tag{4.2}$$

   Note that FCCN can equal 1 while $c_i \neq p_j$, this is the case when c_i is a proper subset of p_j.

2. **F1 Score:** The FCCN focuses solely on the common nodes in the ground-truth and predicted communities. Additionally, we are interested in the nodes present in the predicted community but absent in the ground-truth community, which are considered false positive predictions. The performance metric precision [81], which accounts for these false positives, is defined as follows:

   $$precision(c_i, p_j) = \frac{|c_i \cap p_j|}{|p_j|} \tag{4.3}$$

   The F$_\beta$ score combines precision and recall, with the factor β allowing more weight to be given to either precision or recall. The F1 score [11] evenly balances the two, representing the harmonic mean of precision and recall. It is defined as follows:

   $$F1(c_i, p_j) = \frac{2(precision \cdot recall)}{precision + recall} = \frac{2|c_i \cap p_j|}{|c_i| + |p_j|} \tag{4.4}$$

3. **Fraction of Correctly Classified Edges (FCCE):** Community structure is primarily driven by edges and a comprehensive examination of a CD method's performance includes analysis of community edges. To provide this the following metric is proposed: the Fraction of Correctly Classified Edges (FCCE). It measures how many of the edges in the ground-truth community are found by the mapped-to predicted community. The set of edges of a subset of nodes, here community $c_i \in C$, is defined in Equation 4.5. These edges are also known as the intra-community edges for c_i.

   $$E_{c_i}^{in} = \{(u, v) \in E \mid u \in c_i \text{ and } v \in c_i\} \tag{4.5}$$

   We use this definition to define FCCE in the following manner:

   $$FCCE(c_i, p_j) = \frac{|E_{c_i}^{in} \cap E_{p_j}^{in}|}{|E_{c_i}^{in}|} \tag{4.6}$$

4. **Fraction of Correctly Classified Edges+ (FCCE+):** FCCE identifies edges as correctly classified if for edge (u, v) both nodes u and v are present in the predicted community. To expand FCCE, we also consider the edges where only one of its nodes is present in the predicted community. We refer to these edges as "false bridges" (FBs) because they act as a

bridge between predicted communities when they should be intracommunity edges. The set of these FBs is defined as follows:

$$FB(c_i, p_j) = \{(u, v) \in E^{in}_{c_i} \mid (u \in p_j \text{ and } v \notin p_j) \text{ or } (u \notin p_j \text{ and } v \in p_j)\} \tag{4.7}$$

FBs are used in the alternative metric to FCCE, termed FCCE+. Note that FCCE+ is always greater than or equal to FCCE.

$$\mathit{FCCE+}(c_i, p_j) = \frac{|FB(c_i, p_j)| + 2|E^{in}_{c_i} \cap E^{in}_{p_j}|}{2|E^{in}_{c_i}|} \tag{4.8}$$

4.1.4 Global Metrics

The global approach implements analysis between individual ground-truth community and the entire community prediction instead of solely the mapped-to predicted community. For a ground-truth community, we look how well every predicted community describes the ground-truth community using some metric. These metrics are combined to obtain a measure of how well the set of predicted communities depicts the ground-truth community.

A good global metric should take into account that a prediction is worse if elements of the same ground-truth community are spread out over multiple predicted communities. It should penalize such a distribution in some manner.

1. **Average of F1:** Scores can be combined by averaging non-zero scores of a metric. The calculation of the average F1 score is shown in Equation 4.9 which uses the definition of F1 from Equation 4.4.

$$\mathit{Avg\ F1}(c_i, P) = \frac{\sum_{p_j \in P} F1(c_i, p_j)}{\sum_{p_j \in P} 1_{\{F1(c_i, p_j) \neq 0\}}} \tag{4.9}$$

   Consider the following example: ground-truth community c_1 has been compared to the set of predicted communities $\{p_1, p_2, p_3, p_4\}$, resulting in the list of F1 scores $[0.7, 0.3, 0, 0.2]$ indexed accordingly. We combine the F1 scores by averaging the non-zero values, resulting in the following: $\mathit{Avg\ F1} = \frac{0.7+0.3+0.2}{3} = 0.4$.

   Though this seems a promising global metric, there is an intuitive bias against large communities. Because larger sized communities have more nodes, CDMs are more likely to spread these nodes across multiple predicted communities. This increases the denominator in the formula above, decreasing the average F1 score. This is the case for all global metrics that would utilize averaging some score over the number of non-zero predicted communities scores.

2. **Sum of Weighted F1:** Besides averaging non-zero scores, scores from comparisons between a single ground-truth community and the set of predicted communities can be aggregated by computing a sum of the weighted scores. the weight is set as the fraction of ground-truth nodes that are in the compared-to predicted community. The weight will be:

$$\mathit{weight}(c_i, p_j) = \frac{|c_i \cap p_j|}{|c_i|} \tag{4.10}$$

   The Sum of Weighted F1 (SWF1) can then be formulated, using the definition of F1 in Equation 4.4.

$$\mathit{SWF1}(c_i, P) = \sum_{p_j \in P} \mathit{weight}(c_i, p_j) \cdot F1(c_i, p_j) \tag{4.11}$$

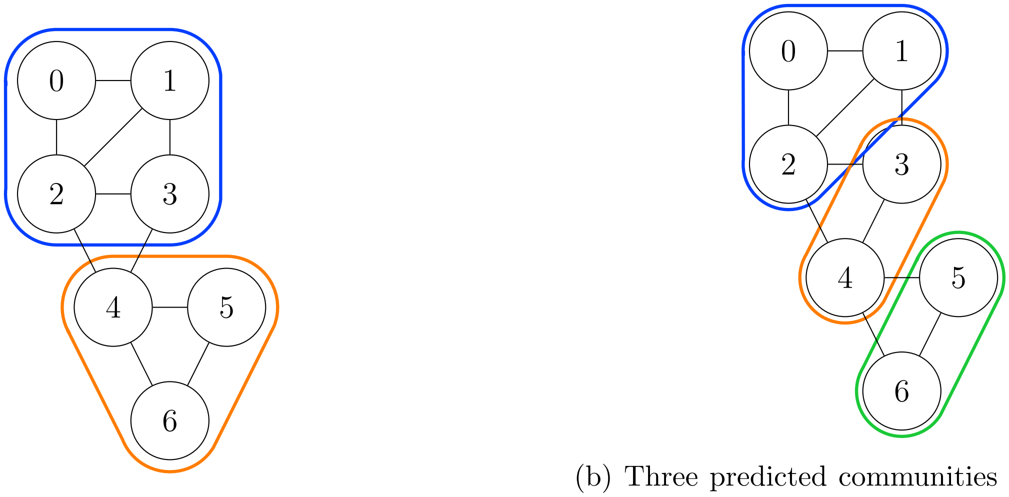
3. **Sum of Weighted FCCE:** Similarly, the Sum of Weighted FCCE (SWFCCE) can be defined using the same weight definition and the formula for FCCE found in Equation 4.6.

$$SWFCCE(c_i, P) = \sum_{p_j \in P} weight(c_i, p_j) \cdot FCCE(c_i, p_j) \tag{4.12}$$

4. **Sum of Weighted FCCE+:** Lastly, we define the Sum of Weighted FCCE+ (SWFCCE+) in the same manner. The definition of FCCE+ can be found in Equation 4.8.

$$SWFCCE{+}(c_i, P) = \sum_{p_j \in P} weight(c_i, p_j) \cdot FCCE{+}(c_i, p_j) \tag{4.13}$$

The following example illustrates the application of the global CPMs. Figure 4.1a shows the ground-truth communities for this example and Figure 4.1b shows the predicted communities by some CDM. Table 4.1 shows the intermediate results of several metrics and Equations 4.14 - 4.17 show the resulting global metric scores for community c_1 from Figure 4.1a.



(a) Two ground-truth communities shown in example graph. The blue community is c_1, the orange community c_2.

(b) Three predicted communities shown in example graph. The blue community is p_1, the orange community p_2, and the green community p_3.

Figure 4.1: Example graph with ground-truth and predicted communities shown by colored groupings.

Predicted Community	$c_1 \cap p_j$	weight	F1	FCCE	FCCE+
p_1	3	$\frac{3}{4}$	$\frac{2 \cdot 3}{2 \cdot 3+0+1} = \frac{6}{7}$	$\frac{3}{5}$	$\frac{2 \cdot 3+2}{2 \cdot 5} = \frac{4}{5}$
p_2	1	$\frac{1}{4}$	$\frac{2 \cdot 1}{2 \cdot 1+1+3} = \frac{1}{3}$	0	$\frac{2 \cdot 0+2}{2 \cdot 5} = \frac{1}{5}$
p_3	0	0	0	0	0

Table 4.1: Intermediate calculations for the global CPMs, calculated for c_1 and the corresponding predicted community p_j found in each row.

$$\begin{aligned}
\mathit{Avg\ F1}(c_1, P) &= \frac{6/7 + 1/3}{2} && \approx 0.60 && (4.14)\\
\mathit{SWF1}(c_1, P) &= \frac{3}{4} \cdot \frac{6}{7} + \frac{1}{4} \cdot \frac{1}{3} && \approx 0.73 && (4.15)\\
\mathit{SWFCCE}(c_1, P) &= \frac{3}{4} \cdot \frac{3}{5} && = 0.45 && (4.16)\\
\mathit{SWFCCE+}(c_1, P) &= \frac{3}{4} \cdot \frac{4}{5} + \frac{1}{4} \cdot \frac{1}{5} && = 0.65 && (4.17)
\end{aligned}$$

4.2 Group Fairness Metric Φ

Our aim is to investigate whether a given CD method performs better for communities of specific properties such as size, density, or conductance. By highlighting disparities in detection accuracy across different communities, the group fairness metric offers a understanding of biases, ensuring that CD methods are evaluated not just on their overall performance, but also on their fairness toward community attributes like size, density, and conductance.

In the previous section, we introduced several community-wise metrics that evaluate a CD method's performance on a per-community basis. These performance scores are plotted against the corresponding community attribute values, as illustrated in Figure 4.2. Here, FCCN scores are plotted against the community size for a dummy network. Because we fairness in respect to multiple community attributes that have values that differ in orders of magnitude, the values of the community attributes are min-max normalized. In Figure 4.2, the dotted line shows a linear least squares approximation [26] of the FCCN scores, to depict the potential disparity in CPM scores. If the approximation is a horizontal line, the algorithm is defined as fair, as it performs consistently across all attribute values. However, when the line is skewed, it suggests that the algorithm favors certain attributes. In Figure 4.2, the positive slope of the regression line indicates that the algorithm performs better for larger communities.

We use the angle of the linear regression to quantify the bias. There are three possible scenarios for the slope: (i) the line is straight, (ii) the slope of the line is positive, and (iii) the slope of the line is negative. We can use the arctangent to calculate the angle of the slope using Δx and Δy, and because we have normalized the property values of all communities in the network, we know that $\Delta x = 1$.

$$\theta = \mathit{arctan}\left(\frac{\Delta y}{\Delta x}\right) = \mathit{arctan}(\Delta y) \tag{4.18}$$

The angle θ, given in radians, is within $(-\frac{\pi}{2}, \frac{\pi}{2})$. We multiply θ by $\frac{2}{\pi}$ to get a result in the range $(-1, 1)$. This leaves us with the equation for the group fairness metric Φ_p^{CPM} shown in Equation 4.19.

$$\Phi_p^{CPM} = \frac{2}{\pi} \cdot \mathit{arctan}(\Delta y) \tag{4.19}$$

Using the example in Figure 4.2 once again, we calculate Φ given the $\Delta y \approx 0.69$. We get $\Phi = \frac{2}{\pi} \cdot \mathit{arctan}(0.69) \approx 0.38$. The sign of Φ is equal to the sign of the slope, meaning that if $\Phi_p^{CPM} > 0$ the slope is positive and the CD method favors higher values of community property p for CPM CPM. If $\Phi_p^{CPM} < 0$, the CD method is biased for lower values of p, and if $\Phi_p^{CPM} = 0$ it means the CD methods are considered fair and find all communities equally well or equally poor.

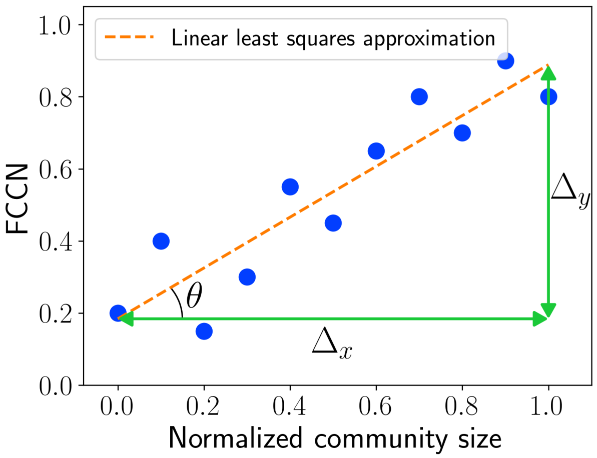


Figure 4.2: An example showing the score for each community by its size. The line is plotted to show the trend of the CDM's performance. In this example, it performs better on communities with a larger size.

This method is undefined when $\Delta x = 0$, e.g., when all communities are the same size. Min-max normalization for Δx is not applicable as this would require division by zero. The denominator of min-max normalization is $x_{max} - x_{min}$ which would equal 0.

4.3 Metric Behavior

To illustrate how the proposed metrics behave we draw up several toy examples. For the mapped and global CPMs we have generated a graph consisting of one community and we will see what values the metrics take as the prediction partiton changes in the examples.

For the mapped metrics a community of 1024 nodes is taken and the prediction is perfect at the start, there is one predicted community that is equal to the ground-truth community. Then, we change the prediction by removing an increasing number of nodes. The resulting metric scores can be found in Figure 4.3. As not all nodes have an equal degree, the scores for FCCE and FCCE+ are different based on the nodes that are removed. The figure shows the area between the highest and lowest recorded values in 20 repetitions, and the average is marked.

It can be seen that the average FCCE+ value is very close to the FCCN value. This is because, over many repetitions, the degree of the removed nodes averages out. In fact, given a complete graph, FCCN will equal FCCE+ for any number of removed nodes, see Appendix A.1 for the proof. Increasing the number of repetitions will average out the degrees of removed nodes further and the value of FCCE+ will approach FCCN even further.

For analysis on the global metrics we use the same 1024 node graph as before. Figure 4.4 shows what happens when we go from a perfect prediction for the single ground-truth community to a prediction that has information from the community, its nodes and edges, spread across multiple predicted communities. Figure 4.4b splits the length of the predicted communities evenly in two at each step. The average F1 score and the sum of weighted F1 are equal here, which is expected since the weighting factor matches the number of communities over which the F1 values are averaged.

Figure 4.4a cuts off a group of 100 nodes from the largest predicted community and uses

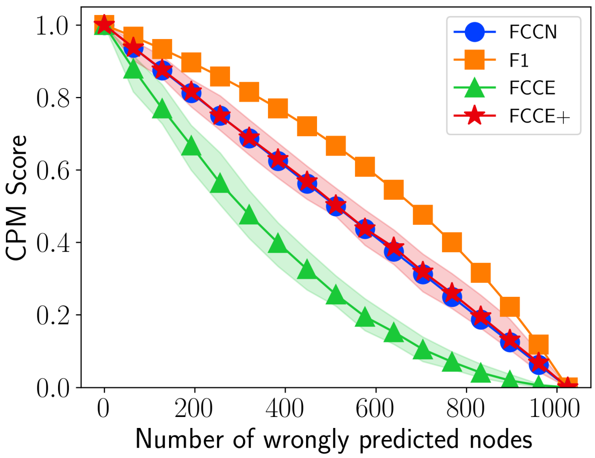
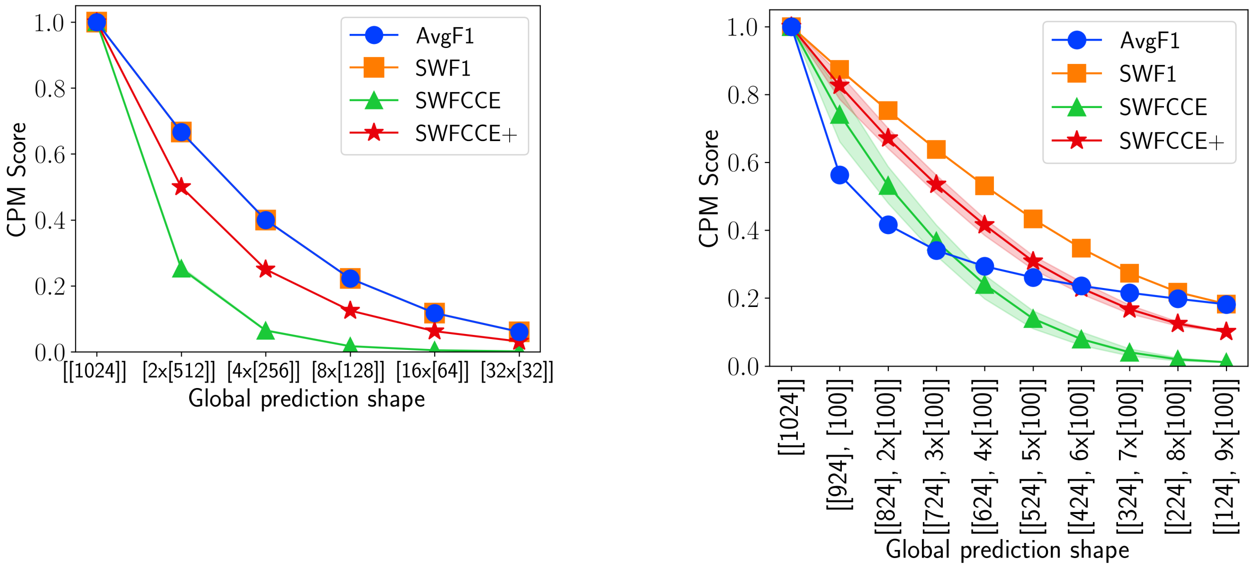


Figure 4.3: Mapped metric behavior with a decreasing number of correctly found nodes in the predicted community. The average values of FCCE and FCCE+ are plotted after 20 repetitions as well as their highest and lowest value at each point.



(a) Predicted communities are split evenly at each step.

(b) From the largest predicted community, 100 nodes are separated at each step.

Figure 4.4: Global metric behavior in two situations. Both start with a perfect prediction that has its nodes separated in different predicted communities. The average values of SWFCCE and SWFCCE+ are plotted after 20 repetitions as well as their highest and lowest value at each step.

those nodes as a predicted community. Here, as well as in Figure 4.4b, we gather results from 20 repetitions. Figure 4.4a shows more fluctuations for SWFCCE and SWFCCE+ scores, as a large part of its value comes from the number of edges correctly found in the largest predicted community due to its weight. Fluctuations in SWFCCE and SWFCCE+ are much less noticeable when the weights are the same as in Figure 4.4b.

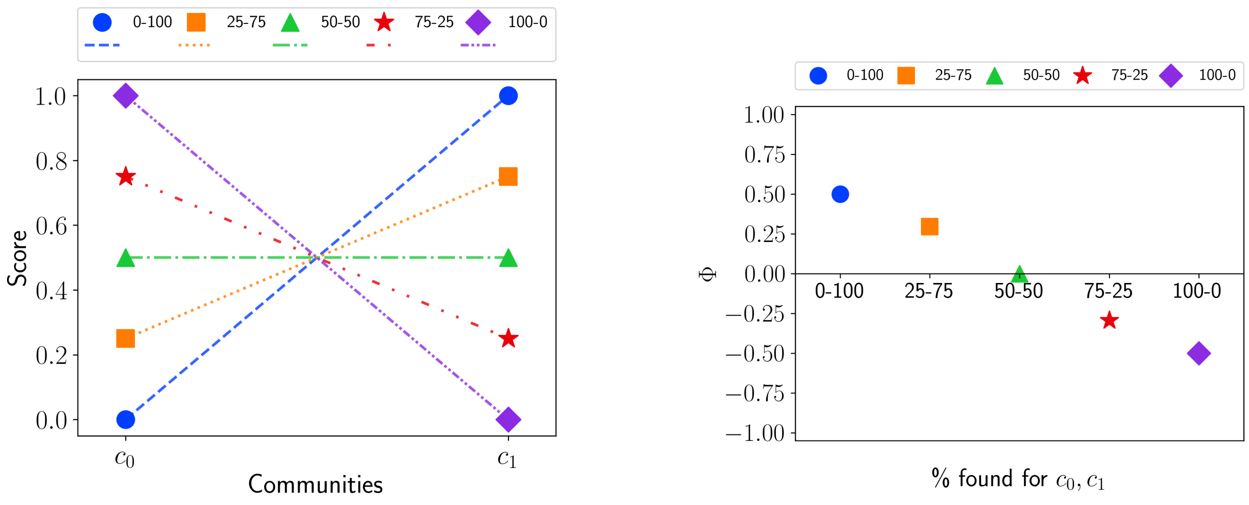
4.3.1 Fairness Metric Behavior

To highlight how the CPM values relate to the group fairness metric Φ we show an example of two ground-truth communities c_1 and c_2 that differ in some community property, e.g. size, density, or conductance. Note that c_1 has a lower value in the community attribute than c_2, as this has an effect on whether Φ is negative or positive. Five scenario's for individual scores are created and the regression line is drawn up through these points. This is shown in Figure 4.5a with the values of the fairness metric Φ shown in Figure 4.5b.

Let us look at a more concrete case with four communities of different sizes: 100, 200, 300, and 400. Figure 4.6 shows this example where, at the start, each community has a mapped-to predicted community that perfectly describes that community ($c_i = p_j$). Then, 50 and 100 nodes are removed from the perfect prediction and each time the regression line is drawn in Figure 4.6a. For communities of varying sizes, both the FCCN and F1 scores tend to favor larger communities because a constant number of misclassified nodes has a smaller relative impact on them. This bias is quantified by the fairness metric Φ, as shown in Figure 4.6b, which becomes more biased toward larger communities as more nodes are not correctly classified.

4.3.2 Mapping Behavior

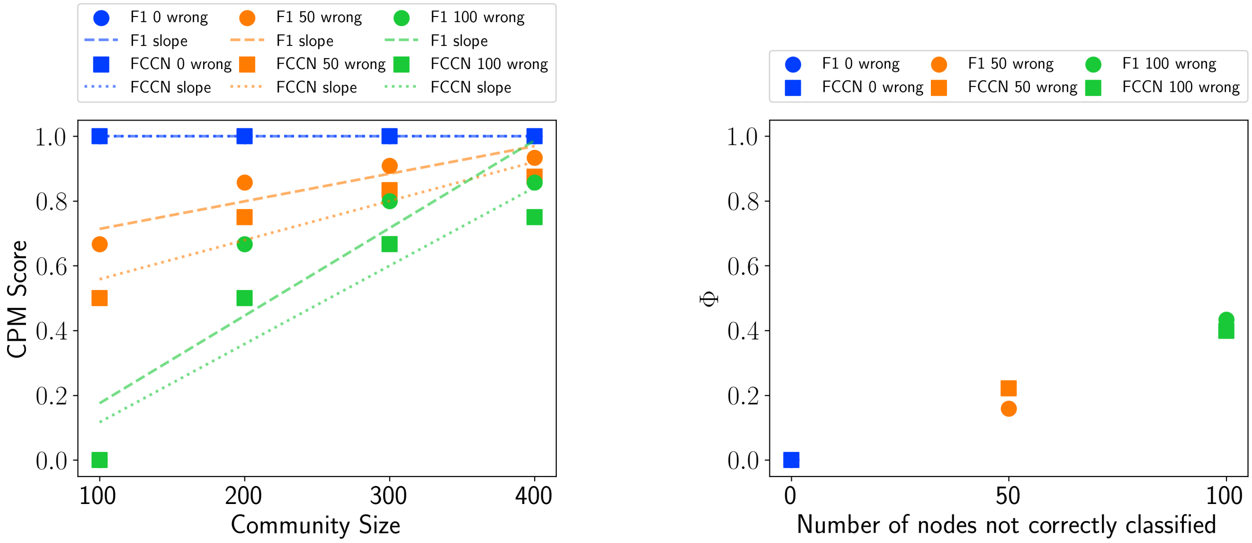
In order to show how the mapping step influences the CPMs and the resulting group fairness metric Φ, we make changes to predictions on a network with two communities. The homophilic network is generated using the HICH-BA model [74] with the homophilic factor set to 0.9. The network consists of a majority community (70 nodes) and a minority community (40 nodes) with ~ 900 edges. Initially, c_{maj} and c_{min} are perfectly detected by some CD method and are mapped to



(a) Individual community performance metric example with two communities. The regression line is drawn through the points in five scenarios given by: (performance in c_1) - (performance in c_2).

(b) Fairness Metric scores gathered from the five scenarios in Figure 4.5a.

Figure 4.5: Figures showing the relationship between individual community performance metrics, the regression line drawn from those points, and the FM.



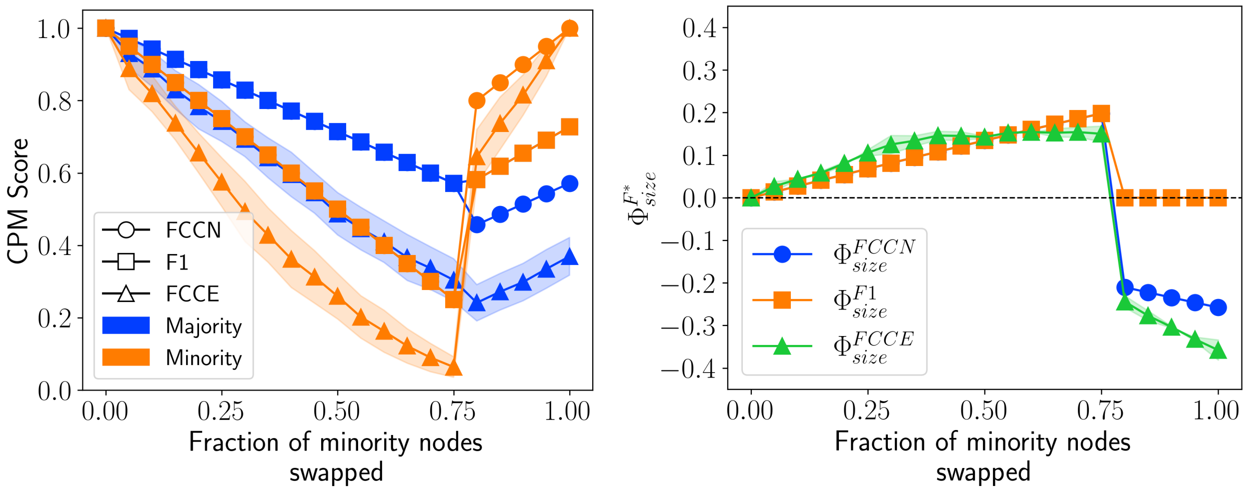
(a) Four communities with varying size. F1 and FCCN are plotted with different number of wrongly predicted nodes.

(b) Fairness Metric scores derived from values in Figure 4.6a.

Figure 4.6: Example of individual community performance metrics turning into FM scores. The bias is shown by the FM when an equal number of nodes are wrong for each community.

p_1 and p_2, respectively. We change the prediction by swapping nodes between the initial perfectly predicted communities p_1 and p_2. We change 0 to 40 nodes and map the ground-truth nodes based on the method described in Section 4.1.3. In Figure 4.7a, you can see the CPM scores against the number of swapped nodes. The resulting group fairness metric score are shown in Figure 4.7b.

The FCCE score varies based on which nodes are swapped, and therefore, we show the average and standard deviation over 20 iterations. FCCE is lower than FCCN and F1 because of the homophilic characteristic of the network; nodes are mostly connected to nodes in their own community. CPM scores are higher for the majority community, that is, until around 75% of the nodes have been swapped. At that point, c_{maj} no longer maps to p_1 and c_{min} no longer maps to p_2, and the mapping changes because a higher Jaccard similarity score is achieved. At that point, the bias, which was in favor of the majority community, swings the other way to favor the minority community for FCCN and FCCE based fairness, see Figure 4.7b. F1 is fair because the F1 scores for the majority and minority communities are equal, this is because F1 also considers false positives in the predicted community.



(a) Community-wise performance behavior

(b) Group fairness metric behavior

Figure 4.7: Analyzing behavior of community-wise fairness metric and group fairness on a network having minority and majority community.

Chapter 5

Experimental Setup

This chapter will give an overview of the experimental setup. This includes a description of the collection of CD methods used in our analysis, along with a categorization of these methods. Furthermore, we will describe the datasets used in our study, including both synthetic and real-world datasets. Lastly, this chapter describes the different metrics we use to determine the quality of the identified communities. We use them to study the performance-fairness trade-off.

5.1 Community Detection Methods

There is no agreed definition of community; yet, finding groups in network structures is valuable in several domains. Therefore, many methods have been proposed to address the problem of community detection; Li et al. [43] offer a comprehensive review of a large number of CD methods. We have assembled a set of 24 CD methods which we will analyze in terms of performance and fairness using our proposed group fairness metric Φ. Table 5.1 shows six categories of CD methods in which the 24 analyzed CD methods are placed. These categories classify CD methods based on their approach to creating community partitions.

Optimization methods aim to optimize a quality function that describes the quality of the partition or community. The modularity function Q [54, 56] is widely used as a standard measure of community quality and is used by all methods in this category, but significance [87]. Since optimizing modularity is NP-hard, these are heuristic methods.

Spectral methods create predictions based on spectral properties of matrices that describe the network, such as the adjacency matrix or the Laplacian matrix [19]. These methods analyze the eigenvalues and eigenvectors of these matrices, which provide insights into the network's structure. For example, Spectral Clustering [27] makes use of the Fiedler vector, the second smallest eigenvalue, to construct communities.

Representational methods work by transforming a network into vector space, often called a network embedding. Network embedding refers to the approach of learning latent low-dimensional feature representations for the nodes or links in a network" [3]. Most of the methods in this category create the network embedding by modeling random walks and use k-means in the network embedding space to create a partition of the network.

Dynamics methods aim to learn the community structure by looking at the traversal of the network, often by random walks. Network traversal could reveal communities by the intuition that

Optimization	Spectral Properties	Representational
• Clauset-Newman-Moore Algorithm (CNM) [12] • Combo [80] • Leiden [88] • Louvain [6] • Paris [7] • Reichardt-Bornholdt - configuration null model (RB-C) [68] • Reichardt-Bornholdt - Erdős-Rényi null model (RB-ER) [68] • Significance [87]	• Eigenvector [53] • Regularized Spectral Clustering with k-means (RSC-K) [97] • RSC sklearn Spectral Embedding (RCS-SSE) [97] • RSC - Vanilla (RSC-V) [97] • Spectral Clustering [27]	• Deepwalk [62] • Fairwalk [66] • Node2Vec [24]

Dynamics	Propagation	Miscellaneous
• Infomap [72] • Spinglass [68] • Walktrap [64]	• Fluid [59] • Label Propagation [13]	• Expectation-Maximization (EM) [57] • Stochastic Block Model (SBM) [60] • SBM - Nested [61]

Table 5.1: Overview of CDMs used in experimentation, categorized into 6 groups.

random walks often stay within communities as these are more densely connected than the rest of the graph.

Propagation methods work by assigning community labels to nodes. Community labels are iteratively updated based on the labels neighboring nodes have. The goal is to reach a stable configuration that reflects the underlying community structure. The first of these methods was the Label Propagation algorithm (LPA) [13] that updates a node's community membership based on the label that the majority of its neighbors have. In the case of a tie, a label is picked at random. LPA is fast and scalable, making the method suitable for large networks. LPA has been the foundation for multiple propagation methods, such as FLPA [89], LLPA [28], LPA-MNI [42], WSSLPA [45], DCC [14] and more.

The **Miscellaneous** category includes methods that could not be added to any of the categories mentioned above. This is because their underlying method differs too much from any of the categories, and therefore they are placed in the miscellaneous group.

5.2 Data

To perform our experiments, we used real-world networks and synthetic networks generated using the LFR benchmark [40], ABCD [32], and the HICH-BA model [74]. All analyzed networks have ground-truth information on the community structure, allowing us to calculate the CPMs. All networks are undirected, unweighted, connected networks with nonoverlapping communities that fully cover the network.

5.2.1 Real-World Networks

Table 5.2 shows a summary of the gathered real-world networks. Because the CPMs require a known community structure, there are not many networks available, especially of larger sizes. Below is a short description of the networks:

Polbooks [36]: This is a network of books on US politics. Data was collected around the time of the 2004 presidential election, and edges were recorded by frequent co-purchasing of a pair of books. The set of books are placed in three communities based on their political affiliation: conservative, liberal, and non-partisan.

Football [23]: US college (American) football is divided into 12 conferences. A network has been created to represent the regular-season games of the 2000 NCAA Division I-A football season. Nodes represent the football teams, connections represent the matches, and communities are derived from the conferences the teams are a part of.

Eu-core [41, 96]: This communication network was built using e-mails sent between employees of a European research center. The departments the employees work in form the community structure. We transform this originally directed network into an undirected network and use only its largest connected component (largest weakly connected component in the original network). The largest connected component makes up 98% of the nodes in the original network.

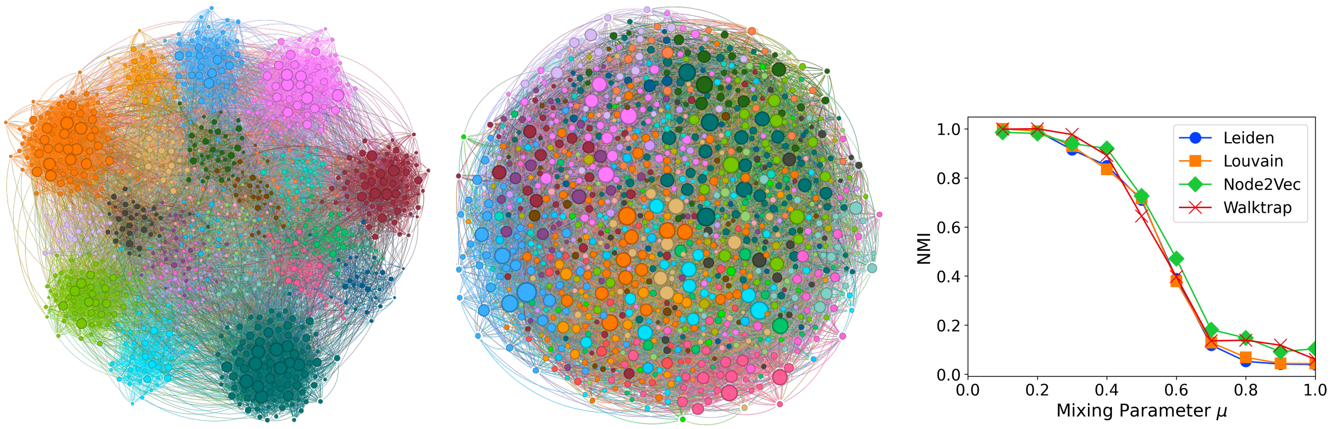
5.2.2 The LFR Benchmark

A popular method for generating synthetic networks with built-in community structure is the Lancichinetti–Fortunato–Radicchi (LFR) benchmark [40]. The LFR benchmark improved on the Girvan-Newman benchmark [23] by making the degree distribution and community size distribution following power laws, thereby creating networks that more closely resemble real-world networks [2, 82]. The LFR benchmark model enables the control of a mixing parameter μ, which gives the fraction of intercommunity connections. If this value is set to 0, all edges are placed within communities, and if the value is set to 1, no edges are within communities. Differences in the values of μ have a large impact on the generated networks. Figures 5.1a and 5.1b illustrate this difference; when μ is lower, communities are a lot more cohesive. The value of μ also influences the ability of CD methods to perform well, as shown in Figure 5.1c, where performance, measured in NMI, decreases with an increase in μ. Other LFR settings include the number of nodes, the average degree of nodes, the maximum degree of nodes, and the minimum community size.

The values we set for our experiments mostly follow the values used by Lancichinetti et al. [39]: The number of nodes $n = 10,000$, the power-law exponent of the degree distribution $\tau_1 = 2$, the power-law exponent of the community size distribution $\tau_2 = 2.5$ (average of values used in [39]),

Dataset name	$\|V\|$	$\|E\|$	*Avg deg*	deg_{max}	$\|C\|$	$\|c_{max}\|$	$\|c_{min}\|$
Polbooks [36]	105	441	8.40	25	3	49	13
Football [23]	115	613	10.66	12	12	13	5
Eu-core [41, 96]	986	16,687	33.85	347	42	107	1

Table 5.2: Real-world dataset summary. $|V|$: number of nodes, $|E|$: number of nodes, *Avg deg*: average degree, deg_{max}: maximum degree, $|C|$: number of communities, $|c_{max}|$: size of largest community, $|c_{min}|$: size of smallest community.



(a) LFR graph with 1000 nodes, $\mu = 0.2$.

(b) LFR graph with 1000 nodes, $\mu = 0.6$.

(c) NMI of predicted partitions with LFR graphs with differing mixing parameter μ values.

Figure 5.1: Two LFR networks with $\mu \in \{0.2, 0.6\}$ and a graph showing how μ affects the quality of the CD methods' prediction. The networks are visualized using ForceAtlas2 [30], node colors are given by ground-truth community, and node size is determined by degree.

the mixing parameter $\mu \in [0.2, 0.4, 0.6]$, the average degree $Avg\ deg = 20$, the maximum degree $deg_{max} = 100$, the minimum community size $|c_{min}| = 20$. Because the value of μ influences the performance of the CD methods, as can be seen in Figure 5.1, we generate 5 networks using the LFR model for every value of μ. For our results, we used the scores of our 5 LFR networks to compute the average and standard deviation. The values of the power-law exponents fall in the range that is found in real-world networks: $2 \geq \tau_1 \geq 3$ and $1 \geq \tau_2 \geq 3$ [18].

5.2.3 ABCD

The Artificial Benchmark for Community Detection (ABCD) model [32] is similar to the LFR but improves on it by addressing scalability issues and the interpretability of the mixing parameter. It generates networks with degree and community size distributions that follow power laws like LFR, but ABCD runs in the order of 100 times faster. ABCD uses ξ as its mixing parameter, which is a more intuitive measure of community strength. When $\xi = 0$, all edges lie within a community, as is also the case when $\mu = 0$. When $\xi = 1$, edges are randomly placed throughout the network, edges are no longer more likely to fall within a community than between communities. With LFR networks with $\mu = 1$, the number of intracommunity edges is 0. With ABCD, the number of intracommunity edges is dependent on the size of the community. The ABCD parameters we use were set to mimic the graphs generated by the LFR benchmark. The values of ξ are slightly different from the μ values $[0.2, 0.4, 0.6]$. We calculated the appropriate ξ values using the global method described in [32]. We get $\xi \in [0.201, 0.402, 0.603]$.

5.2.4 HICH-BA

HIgh Clustering Homophily Barabási-Albert (HICH-BA) [74] is an extension to the homophily BA model [33]. The LFR benchmark and the ABCD algorithm work by creating the degree and communities as a first step. HICH-BA iteratively adds nodes and edges to the network based on probabilities provided by the user. HICH-BA takes the following parameters:

- n: the number of nodes in the network.
- r: This is a list of probabilities. The likelihood of assigning a node to community c_i is given by the i-th element in r.
- h: The homophily factor gives the probability of an edge being within the community.
- p_N: The probability of adding a node to the graph. The probability of adding a edge in an iteration is $1 - p_N$.
- p_t: This gives the probability to form a close triad connection.
- p_{PA}: This gives the probability for a new edge to be placed using preferential attachment (PA) weights.

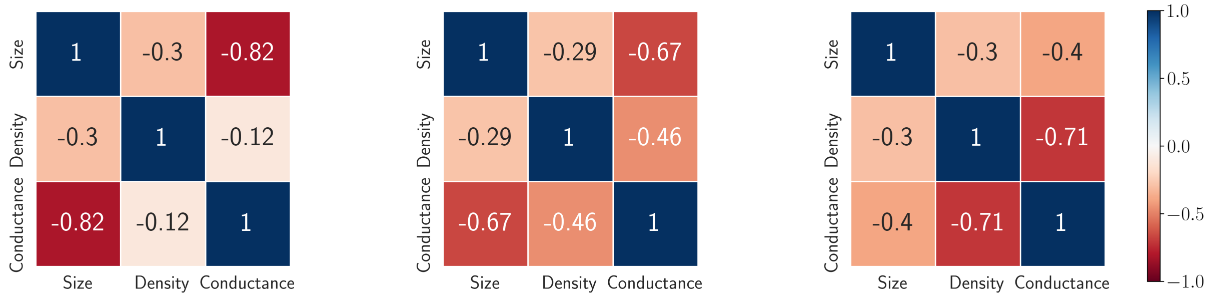
HICH-BA allows for a custom community size distribution by setting r. We set r to create two network cases: (i) Multiple minority communities with one majority community (MMin), and (ii) multiple majority communities (MMaj). The parameter r was set as [0.005, 0.005, 0.005, 0.01, 0.01, 0.01, 0.02, 0.02, 0.02, 0.9] in the MMin scenario, and r is set as [0.003, 0.003, 0.003, 0.03, 0.03, 0.03, 0.3, 0.3, 0.3] in the MMaj case. Note that r does not necessarily need to add up to 1. The other parameters are the same for both cases: $n = 10,000$, $h = 0.9$, $p_N = 0.1$, $p_T = 0.3$, and $p_{PA} = 0.8$. For our results, we average the results gathered over 5 HICH-BA networks, per community structure.

5.2.5 Synthetic Data Analysis

Because synthetic data is generated programmatically, it is important to look at some aspects of the networks because bias can also be found in data. For example, it might be possible that in synthetic networks, large communities are always dense. In such a case, we would like to know whether we can separate Φ_{size} and $\Phi_{density}$. We created correlation matrices between the community properties size, density, and conductance. These were created using all communities from the multiple generated networks, per generation model. Because they provide different results, the correlation matrices for LFR and ABCD are separated by their mixing parameters. Specifically, by correlation, we refer to the Pearson correlation coefficient.

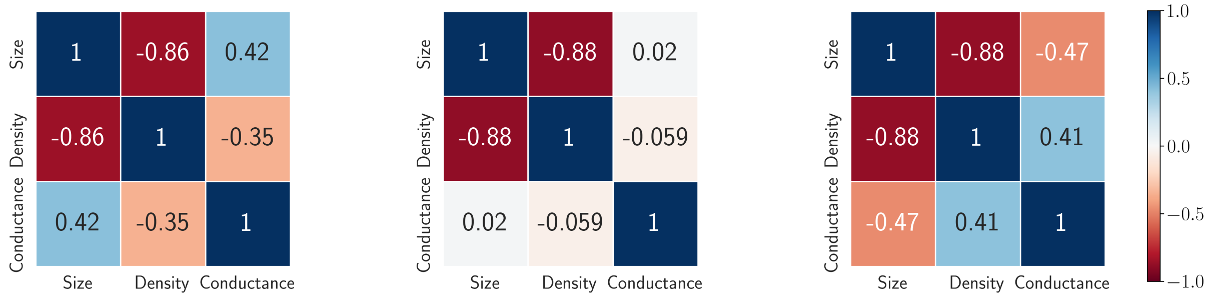
The correlation matrices comparing community properties from LFR networks are shown in Figures 5.2. Here we can see that the correlation between size and conductance is quite high when $\mu = 0.2$. With a negative value, this means that when community size gets bigger, the conductance is smaller. This correlation diminishes as μ gets larger, but in its place a correlation between density and conductance appears.

Figure 5.3 shows the correlation matrices for ABCD networks. Here, we can see a correlation between density and size. This correlation is constant for all values of ξ. For HICH-BA, we also find considerable correlation, see Figure 5.4. In MMaj (Figure 5.4a), we find correlation between density and size. In MMin (Figure 5.4b, correlation is particularly found between conductance and size, and between conductance and density. Table 5.3 gives a summary for the generated synthetic networks. Network entries are not separated by mixing parameter for LFR and ABCD because their values are almost the same.



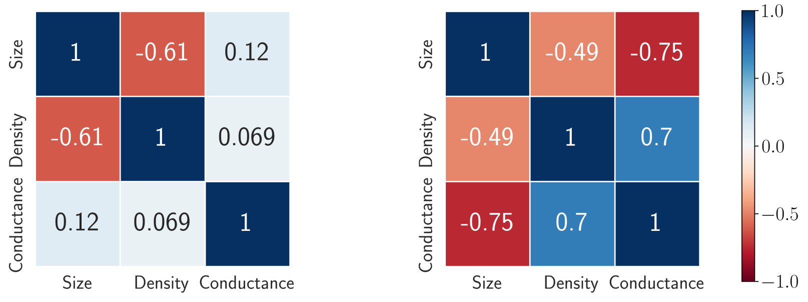
(a) $\mu = 0.2$ (b) $\mu = 0.4$ (c) $\mu = 0.6$

Figure 5.2: Correlation matrices for community properties size, density, and conductance. Community information is gathered from LFR networks.



(a) $\xi = 0.2$ (b) $\xi = 0.4$ (c) $\xi = 0.6$

Figure 5.3: Correlation matrices for community properties size, density, and conductance. Community information is gathered from ABCD networks.



(a) MMaj (b) MMin

Figure 5.4: Correlation matrices for community properties size, density, and conductance. Community information is gathered from HICH-BA networks.

Dataset name	$\|V\|$	$\|E\|$	*Avg deg*	deg_{max}	deg_{min}	$\|C\|$	$\|c_{max}\|$	$\|c_{min}\|$
LFR	10,000	136,334.4	27.3	111.8	8	282.0	98.4	20
ABCD	10,000	105,063.1	21.0	100.3	8	264.9	116.2	20
HICH-BA MMaj	10,000	98,054.4	19.6	1021.4	1.0	9	3038.8	27.4
HICH-BA MMin	10,000	92,597.4	18.5	1283.6	1.0	10	8964.6	42.0

Table 5.3: Synthetic dataset summary. $|V|$: number of nodes, $|E|$: number of nodes, *Avg deg*: average degree, deg_{max}: maximum degree, deg_{min}: minimum degree, $|C|$: number of communities, $|c_{max}|$: size of largest community, $|c_{min}|$: size of smallest community.

5.3 Measuring Quality of the Prediction Partition

There exist multiple metrics that can quantify the similarity of the predicted partition to the ground-truth community structure. Normalized mutual information (NMI) [20] is a metric that has been often used in papers comparing CD methods [38, 40, 95]. Our experiments include measuring NMI, as well as other metrics, defined below. All these metrics range from 0 to 1, where 1 means that the predicted partition is identical to the ground-truth community structure. This is not the case for variation of information (VI), which can give scores larger than 1. For variation of information, 0 indicates that the community structure is perfectly predicted.

5.3.1 Normalized Mutual Information

NMI [20] is, as the name suggests, a normalized variant of mutual information (MI). MI is a information theoretic measure, defined as:

$$MI(C, P) = \sum_{i=1}^{m} \sum_{j=1}^{k} P(c_i \cap p_j) \log \frac{P(c_i \cap p_j)}{P(c_i)P(p_j)} \tag{5.1}$$

The notation for the predicted partition P is not to be confused with the function representing probability. In Equation 5.1, $P(c_i)$ gives the probability that a randomly chosen node belongs to c_i, and $P(c_i \cap p_j)$ gives the probability that a randomly chosen node belongs to both c_i and p_j. NMI is defined as:

$$NMI(C, P) = \frac{MI(C, P)}{\sqrt{H(C)H(P)}} \tag{5.2}$$

Here, $H(C)$ and $H(P)$ are the entropies of the ground-truth and predicted partitions, respectively.

$$H(C) = -\sum_{i=1}^{m} P(c_i) \log P(c_i) \tag{5.3}$$

$$H(P) = -\sum_{j=1}^{k} P(p_j) \log P(p_j) \tag{5.4}$$

5.3.2 Reduced Mutual Information

The Reduced Mutual Information (RMI) [55] was proposed to correct a flaw with MI. Namely, that the measure should return a value of 0 in the case where the predicted partitions has n communities as this predicted partition communicates nothing about the community structure. However, instead of 0, MI returns $H(C)$ [55]. To combat this, Newman et al. added a correction term to define RMI as:

$$RMI(C, P) = MI(C, P) - \frac{1}{n} \log \Omega(a, b) \tag{5.5}$$

Here, $\Omega(a, b)$ is the number of $C \times R$ non-negative integer matrices with row sums $a = \{|c_i|\}$ and column sums $b = \{|p_j|\}$. For our experiments, we make use of the CDlib [70] implementation that normalizes the RMI score.

5.3.3 Variation of Information

Variation of Information (VI) [51], also known as shared information distance, is another information theoretic metric. For VI, a score of 0 represents the case where the predicted partition and ground-truth partition are equal. This makes sense given the name, the prediction is perfect given zero variation. The VI is calculated using the definition of MI (see Equation 5.1) and definitions of entropy of a partition (see Equations 5.3 and 5.4):

$$VI(C,P) = H(C) + H(P) - 2MI(C,P) \tag{5.6}$$

The maximum value of VI is equal to $H(C) + H(P)$. When the predicted partition and the ground-truth partition are completely independent, MI is zero. VI is then maximized.

5.3.4 Adjusted Rand Index

The Adjusted Rand Index (ARI) [29] is a chance-adjusted version of the Rand Index (RI). RI is given by:

$$RI(C,P) = \frac{TP + TN}{TP + FP + FN + TN} \tag{5.7}$$

Here, TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives, and FN is the number of false negatives. These terms come from the confusion matrix; they measure instances in prediction tasks. In this context, the terms are defined as follows:

- TP: the number of pairs of nodes that are in the same community in C and in the same community in P.
- TN: the number of pairs of nodes that are in a different community in C and in a different community in P.
- FP: the number of pairs of nodes that are in a different community in C and in the same community in P.
- FN: the number of pairs of nodes that are in the same community in C and in a different community in P.

Hubert et al. formulated a way to adjust for chance in any measure M as follows:

$$adjusted\ M = \frac{M - E(M)}{M_{max} - E(M)} \tag{5.8}$$

$E(M)$ is the expected value for some null model. Hubert formulated that if partitions are generated randomly, the expected number of pairs in a community intersection $c_i \cap p_j$ is given by:

$$E\binom{|c_i \cap p_j|}{2} = \binom{|c_i|}{2}\binom{|p_j|}{2} \Big/ \binom{N}{2} \tag{5.9}$$

With these definitions, we can get the ARI:

$$ARI(C,P) = \frac{\sum_{ij}\binom{|c_i \cap p_j|}{2} - \sum_i \binom{|c_i|}{2}\sum_j \binom{|p_j|}{2} \Big/ \binom{N}{2}}{\frac{1}{2}\left(\sum_i \binom{|c_i|}{2} + \sum_j \binom{|p_j|}{2}\right) - \sum_i \binom{|c_i|}{2}\sum_j \binom{|p_j|}{2} \Big/ \binom{N}{2}} \tag{5.10}$$

ARI is upper bounded by 1, this is given when the predicted partition exactly matches the ground-truth partition. It is lower bounded by -1. Negative ARI values indicate that the similarity between the two partitions is less than the expected value from two random partitions.

5.3.5 Average F1 Score: PF1

Rossetti et al. [71] introduced an approach to measure the quality of predicted communities. It works by mapping predicted communities to ground-truth communities. Ground-truth community membership is labeled. Using this, a predicted community p_j is mapped to the ground-truth community c_i with the highest number of labels in p_j. Many predicted communities can map to one ground-truth community.

When mapping is complete, we can compare the mapped communities just as we did with our proposed CPM F1. F1 is the harmonic mean of recall and precision, these are defined in Equations 4.2 and 4.3, respectively. F1 combines these measures, as described in Equation 4.4. These F1 scores measure similarity between pairs of mapped communities. These scores can be averaged and this value, along with its standard deviation, can be used to compare predicted partitions. To distinguish this measure from our proposed CPM, we refer to this partition-based approach as PF1

5.3.6 Normalized F1 Score

Rossetti later built upon PF1 by introducing a normalized version [69]. This variant takes into account two newly introduced measures. These measures come from the many-to-one mapping from predicted communities to ground-truth communities. It is possible that ground-truth communities are not mapped to. This is the case when a ground-truth community's label is not most frequent in any predicted community. We call C_{id} the set of ground-truth communities that are mapped to. The first introduced measure is "coverage", this gives the percentage of ground-truth communities that are mapped to.

$$coverage = \frac{|C_{id}|}{|C|} \in [0,1] \tag{5.11}$$

The second measure is "redundancy". This quantifies how many times larger the set of predicted communities P is compared to C_{id}. This given by:

$$redundancy = \frac{|P|}{|C_{id}|} \in [1,+\infty) \tag{5.12}$$

These measures are combined using the definition of PF1 to create the normalized version of PF1: NF1.

$$NF1 = \frac{PF1 \cdot coverage}{redundancy} \in (0,1] \tag{5.13}$$

5.4 Experimental Details

We perform our experiments using Python libraries NetworkX [25] and CDlib [70]. To create the LFR networks we use NetworkX's implementation. We use the CDlib implementation of most of the CD methods we analyze and when a method requires certain parameters to be set, we use the CDlib's default parameters. Several methods require the number of communities to be predicted, and we provide it if required. Note that the number of ground-truth communities is not usually known and these methods will have an advantage over other methods that have to infer the number of ground-truth communities themselves. The methods that require the number of predicted communities to be given are RSC-K, RSC-SSE, RSC-V, Spectral Clustering, Deepwalk, Fairwalk, Node2Vec, Fluid, and EM. All CD method parameters and their used values are reported in Appendix B. Our code and data can be found in our GitHub repository.

Chapter 6

Results

In this chapter, we present our results for the proposed fairness metric Φ_p^{CPM}. Our fairness metric examines bias with respect to three community properties: size, density, and conductance. CPM stands for community-wise performance metrics; these measure how well ground-truth communities are detected by CD methods. Our proposed CPMs are explained in further detail in Section 4.1 and are divided into two groups: mapped and global metrics. These results are gathered using real-world networks and synthetic networks, generated using the LFR benchmark, ABCD, and HICH-BA method. Refer to Section 5.2 for more details on the datasets.

This chapter is structured as follows: Section 6.1 contains the results of the mapped CPMs and Section 6.2 contains the results of the global CPMs. These sections will include figures where NMI is plotted against the group fairness measure Φ_p^{CPM}. NMI is the most popular partition quality measure, often used to compare CD methods [38, 40, 95]. Section 6.3 will give the results of other partition quality measures; these include RMI, VI, ARI, F1, and NF1. When we speak of "performance" in this chapter, we mean these partition quality metrics.

CD methods have to figure out the number of communities themselves. How successfully they identify number of communities influences the community-wise performance heavily; this is especially the case with mapped CPMs, as discussed in Section 4.1.2. For synthetic networks, we measure the average difference between the number of predicted communities $|P|$ and the number of ground-truth communities $|C|$. Table 6.1 has these values. Note that several CD methods were provided with the correct number of ground-truth communities. These are marked with a * in the table.

We have a lot of results to present, where it may be challenging to distinguish individual points in figures where marks overlap. Therefore, we present our synthetic results of Φ_p^{CPM} with NMI in tabular form in Appendix C: Appendices C.1 to C.24. Scores from the performance measures are given in tables in Appendices C.25 to C.32.

6.1 Results for Mapped CPMs

Here we present the results of Φ_p^{CPM} using the mapped CPMs. NMI is plotted against Φ_p^{CPM} in these figures to give insight into the performance-fairness trade-off.

	LFR			ABCD			HICH-BA	
CD Method	$\mu = 0.2$	$\mu = 0.4$	$\mu = 0.6$	$\xi = 0.2$	$\xi = 0.4$	$\xi = 0.6$	MMaj	MMin
CNM	-261.6	-269.6	-261.6	-237.8	-253	-237.8	16.2	151.6
Combo	-171.4	-208.4	-171.4	-166.8	-192	-166.8	0.2	6
Leiden	-167.2	-198.6	-167.2	-163.2	-186.8	-163.2	15.2	31.4
Louvain	-169.6	-197	-169.6	-162.4	-184.2	-162.4	14.8	19.6
Paris	-56.6	-192.4	-56.6	-54.4	-2.8	-54.4	-2.6	-5
RB-C	-167.2	-198	-167.2	-161.8	-184.8	-161.8	15.4	18.6
RB-ER	-132	-191.6	-132	-139	-172	-139	3974.6	3974.2
Significance	8	193.8	8	8.6	23.2	8.6	3870.4	3724.8
Eigenvector	-207.7	-258.2	-207.7	-198.8	-225.8	-198.8	27.7	-3
RSC-K*	0	0	0	0	0	0	0	0
RSC-SSE*	0	0	0	0	0	0	0	0
RSC-V*	0	-0.2	0	-2.2	-2	-2.2	0	0
Spectral*	-280.4	-275.6	-280.4	-265.8	-264.8	-265.8	-6.6	-8
Deepwalk*	0	0	0	0	0	0	0	0
Fairwalk*	0	0	0	0	0	0	0	0
Node2Vec*	0	0	0	0	0	0	0	0
Infomap	0	-6.2	0	0	0	0	463	635.8
Spinglass	-258.2	-254	-258.2	-242.8	-242.8	-242.8	15.6	14.4
Walktrap	-10	-46.2	-10	0	0	0	344	3128.8
Fluid*	0	0	0	0	0	0	0	0
Label Propagation	-37.2	-108	-37.2	-29.6	-86	-29.6	97	108
EM*	-51.8	-66.6	-51.8	0	0	0	0	0
SBM	-81	-86.6	-81	-97.6	-88.8	-97.6	15	4.8
SBM - Nested	45	70.6	45	13.4	1.8	13.4	12.4	7

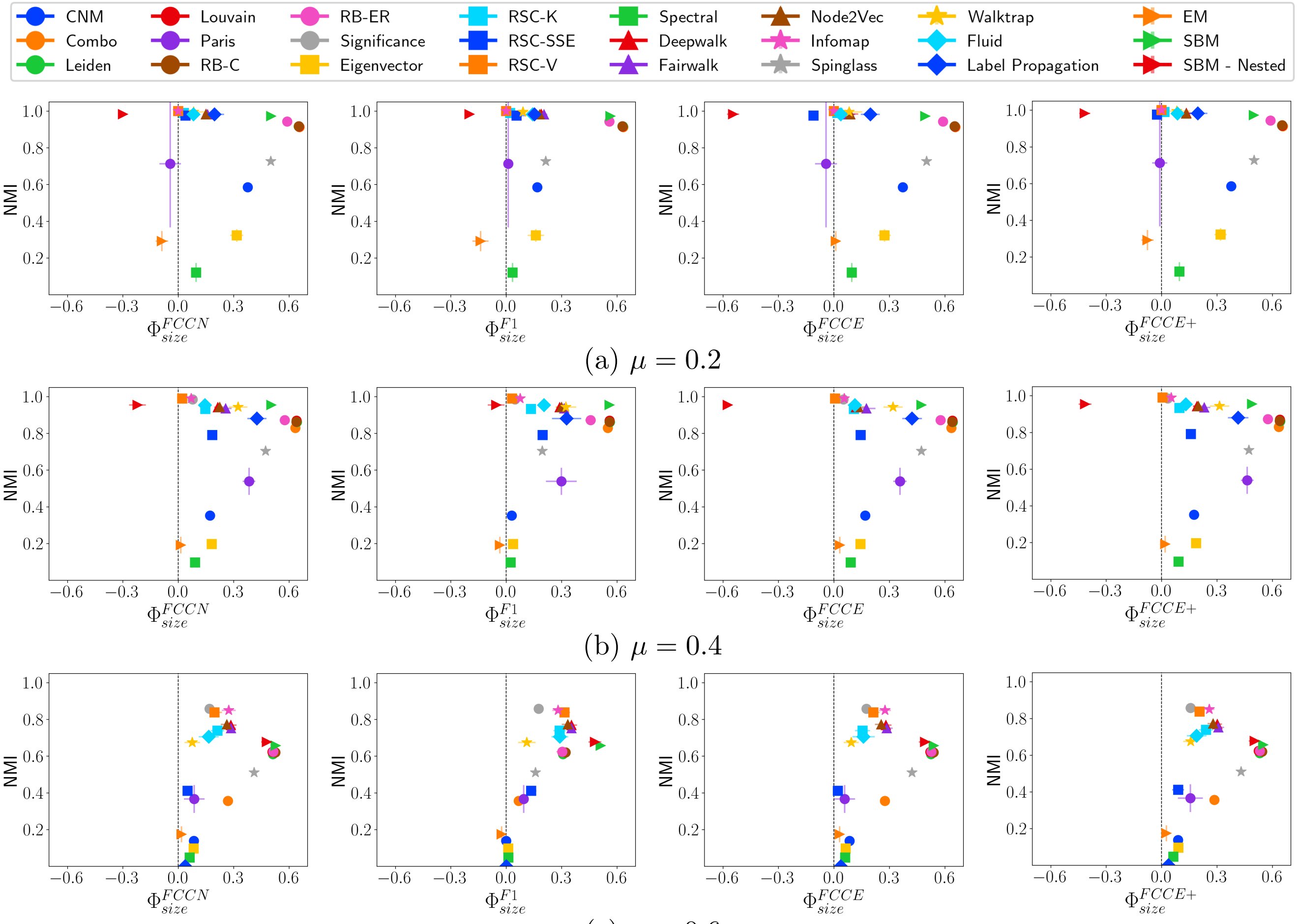
Table 6.1: This table gives an overview of the difference between the number of predicted communities and the number of ground-truth communities. This difference is averaged over the number of available networks. So, a negative value of x indicates that the CD method has predicted an average of x communities too few. CD methods that are marked with * have been given the number of ground-truth communities in order to work.

6.1.1 Results on LFR for Mapped CPMs

Fairness - Size

First, we analyze how well different CD methods identify communities of different sizes. In Figure 6.1 you can find the results on LFR networks with mixing parameters $\mu = 0.2$ (Figure 6.1a), $\mu = 0.4$ (Figure 6.1b), and $\mu = 0.6$ (Figure 6.1c). NMI is plotted against the mapped variants of the group fairness metric Φ: Φ^{FCCN}_{size}, Φ^{F1}_{size}, Φ^{FCCE}_{size}, and Φ^{FCCE+}_{size}. Detailed results can be found in tables in Appendices C.1, C.2, and C.3.

For $\mu = 0.2$, larger communities are favored by all CD methods except EM, Paris, SBM-Nested, and RSC-SSE. This is the case for almost all CPMs. The methods that have high performance (NMI $\approx$ 1) and are fair include RSC-K, RSC-V, RSC-SSE, Infomap, Fluid, Walktrap, and Sig-



(a) $\mu = 0.2$

(b) $\mu = 0.4$

(c) $\mu = 0.6$

Figure 6.1: NMI and Φ^{CPM} scores in regard to **size** with mapped CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

nificance. SBM-Nested is an outlier that favors minority communities. This is because this is a hierarchical variant of the SBM model, which returns the lowest level as its prediction. This is no longer the case when μ is set to 0.6. Then, no CD method favors minority communities regardless of CPM. The only exception is EM with Φ^{F1}_{size}. Communities that are found with $\mu = 0.6$ are larger, meaning all CD methods favor majority communities.

As illustrated by Figure 5.1, CD performance decreases with a higher mixing parameter. However, several CD methods already perform badly with a low mixing parameter. These are Paris, Spinglass, CNM, Eigenvector, EM, and Spectral.

When $\mu = 0.4$, there are still several methods that are both fair and have high performance: Significance, RSC-V, RSC-K, Infomap. Spectral is among the most fair for FCCN, FFCE, and FCCE+. However, we can see that its performance is really bad, meaning that all communities have been evenly misclassified. Looking at Table 6.1, we can see that this is the case because Spectral has predicted a lot fewer communities than ground truth had. Another CD method that has predicted far too few communities and has bad performance is CNM. However, there are some CD methods that have predicted a comparable amount of communities but still score well in terms of performance. These include Leiden, Louvain, RB-C, and RB-ER. We do see that these favor larger communities heavily, across all the different mixing parameter values. These methods are less biased when you look at Φ^{F1}. This is because the large predicted communities have a lot of false positives; these are taken into account by F1. This decreases the community-wise performance score for F1, bringing it in line with smaller communities.

FCCE and FCCE+ have almost identical fairness scores. However, when there is a difference, FCCE+ is almost always more biased toward larger communities than FCCE. This is consistently the case for all mixing parameter values.

Fairness - Density

Figure 6.2 presents results from LFR networks illustrating fairness with respect to density. For $\mu = 0.2$ and $\mu = 0.4$ the CD methods are more divided than was the case with Φ_{size}. This is still the case when $\mu = 0.6$. There, we find a group slightly favoring low-density communities while still performing well: Leiden, Louvain, RB-C, and RB-ER. Other CD methods that have high performance have an opposite bias that is also much stronger. This cluster is made up of Significance, RSC-K, RSC-V, Deepwalk, Fairwalk, Node2Vec, Infomap, Walktrap, Fluid, SBM, and SBM-Nested. These methods are more effective at detecting denser communities compared to less dense ones. This is also more pronounced when $\mu = 0.6$ compared to when $\mu = 0.4$.

As we saw with bias with respect to size, we can see that FCCE+ is very similar to FCCE. However now, whenever there is a deviation, it shows that FCCE+ favors low-density communities slightly more than FCCE.

Fairness - Conductance

Figure 6.3 shows the results of $\Phi_{conductance}$ for mapped CPMs. Communities with low conductance are more separated from the network. Already when $\mu = 0.2$, it is evident that most CD methods favor low-conductance communities. This is most prevalent with the group of modularity-based methods: Combo, Leiden, Louvain, RB-C, and RB-ER. SBM is another method that strongly favors low-conductance communities.

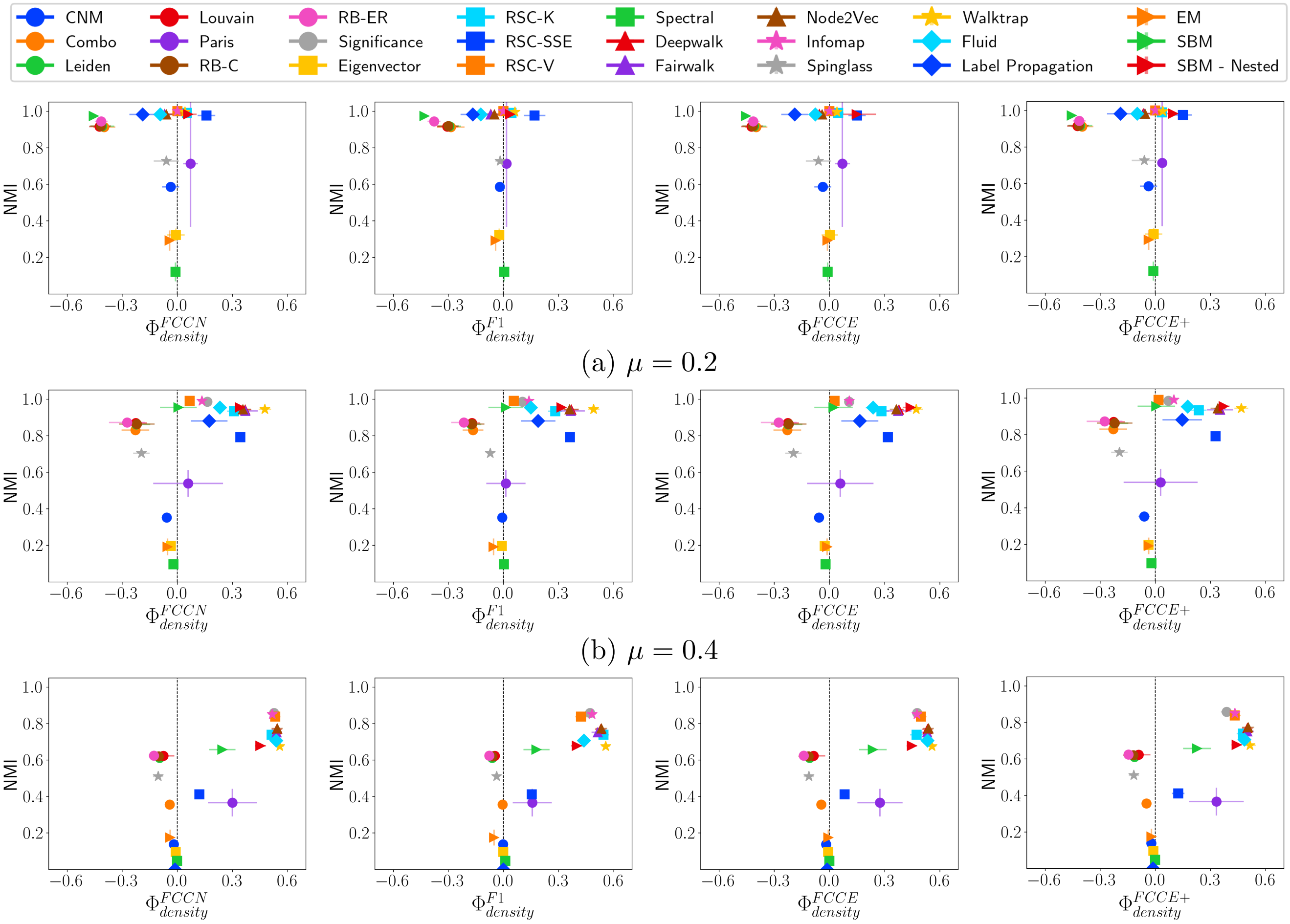
When μ increases to 0.4, almost all CD methods are biased toward low-conductance communities. Infomap, RSC-V, Significance, and SBM-Nested are also relatively fair when community-wise performance is measured in FCCN and F1. SBM-Nested moves to favor high-conductance communities for FCCE and FCCE+.

When μ increases even further, all CD methods are biased toward low-conductance communities. Our results show that high-performing CD metrics are more biased than lower-performing CD methods. This is the case for all CPMs.

6.1.2 ABCD Results Mapped CPMs

Fairness - Size

Now we look at the results we gathered when applying the CD methods to networks that were generated using the ABCD algorithm. Figure 6.4 shows the fairness metric Φ_{size} with respect to community size. The results with $\xi = 0.2$ look very similar to the results we got for LFR networks with $\mu = 0.2$, see Figure 6.1a. There are several CD methods that have an unusually low performance: Paris, Spinglass, CNM, Eigenvector, EM, and Spectral. Most CD methods have



(a) $\mu = 0.2$

(b) $\mu = 0.4$

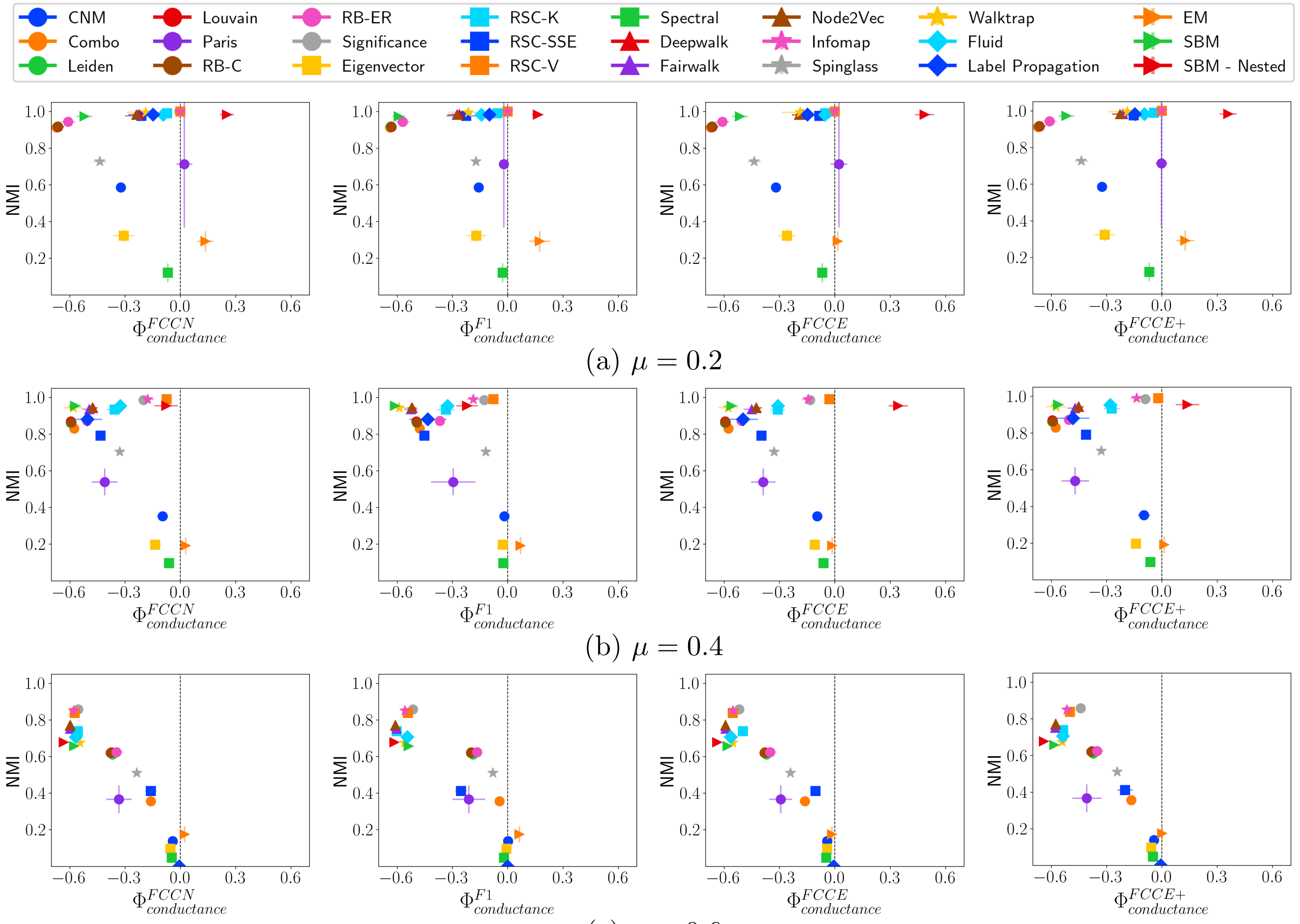
(c) $\mu = 0.6$

Figure 6.2: NMI and Φ^{CPM} scores in regard to **density** with mapped CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

created very successful predictions, scoring high performance values and low bias ratings. There is also, once again, a group that is already very biased toward majority communities, composed of Combo, Leiden, Louvain, RB-C, and SBM.

With $\xi = 0.4$, we get different fairness results. Five CD methods outperform the best performing CD method in the LFR network ($\mu = 0.4$) in terms of NMI. These are: Significance, RSC-V, Infomap, Walktrap, and SBM-Nested. Of these, Significance, RSC-V, Infomap, and Walktrap are also among the most fair CD methods meaning all communities have been detected well. Other CD methods have slight biases toward both majority and minority communities.

The group of Combo, Leiden, Louvain, RB-C, RB-ER, Spinglass and SBM favor larger communities heavily, also when $\xi = 0.6$. With this mixing parameter, the largest bias not from this group is found in Fairwalk with Φ^{F1}. There is a large group of fair, well-performing CD methods. This is quite remarkable for such a high mixing parameter value. Notable are Infomap, RSC-V, SBM-Nested, and Significance. These CD methods all score at or above 0.987. With $\mu = 0.6$, most CD methods favor larger communities, although most do only slightly. There is not a clear trend between Φ_{size} and performance as was evident from the LFR results.



(a) $\mu = 0.2$

(b) $\mu = 0.4$

(c) $\mu = 0.6$

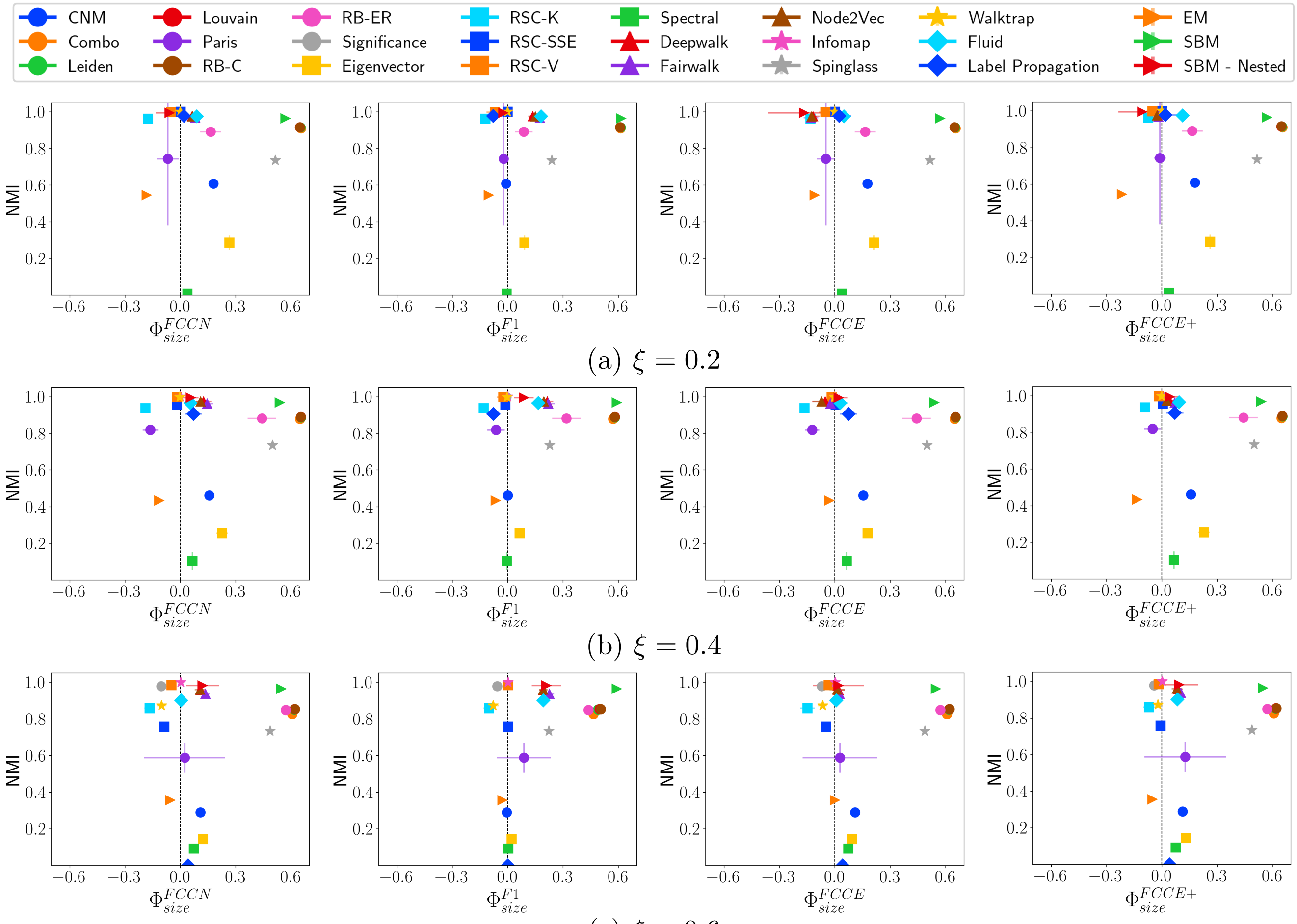
Figure 6.3: NMI and Φ^{CPM} scores in regard to **conductance** with mapped CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

Fairness - Density

Figure 6.5 shows the fairness metric $\Phi_{density}$ for CD methods performed on ABCD networks. Combo, Leiden, Louvain, RB-C, RB-ER, and SBM favor lower density communities for all values of ξ. They, along with most CD methods score well, even with high mixing. Contrary to what was found for LFR networks, we see that fairness does not increase as the mixing parameter increases. Also, more CD methods favor lower density communities instead of high density, especially for $\Phi^{F1}_{density}$.

Fairness - Conductance

Figure 6.6 shows $\Phi_{conductance}$ for results gathered from ABCD networks. Once again, we see high NMI scores for most CD methods and the group of Combo, Leiden, Louvain, RB-C, and SBM are deviating from other CD methods in terms of fairness. With $\xi = 0.2$, these methods favor communities with higher conductance. When $\xi = 0.4$, these methods appear fair but when $\xi = 0.6$ they favor low-conductance communities. Most CD methods perform well and are very fair.



(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

Figure 6.4: NMI and Φ^{CPM} scores in regard to **size** with mapped CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.

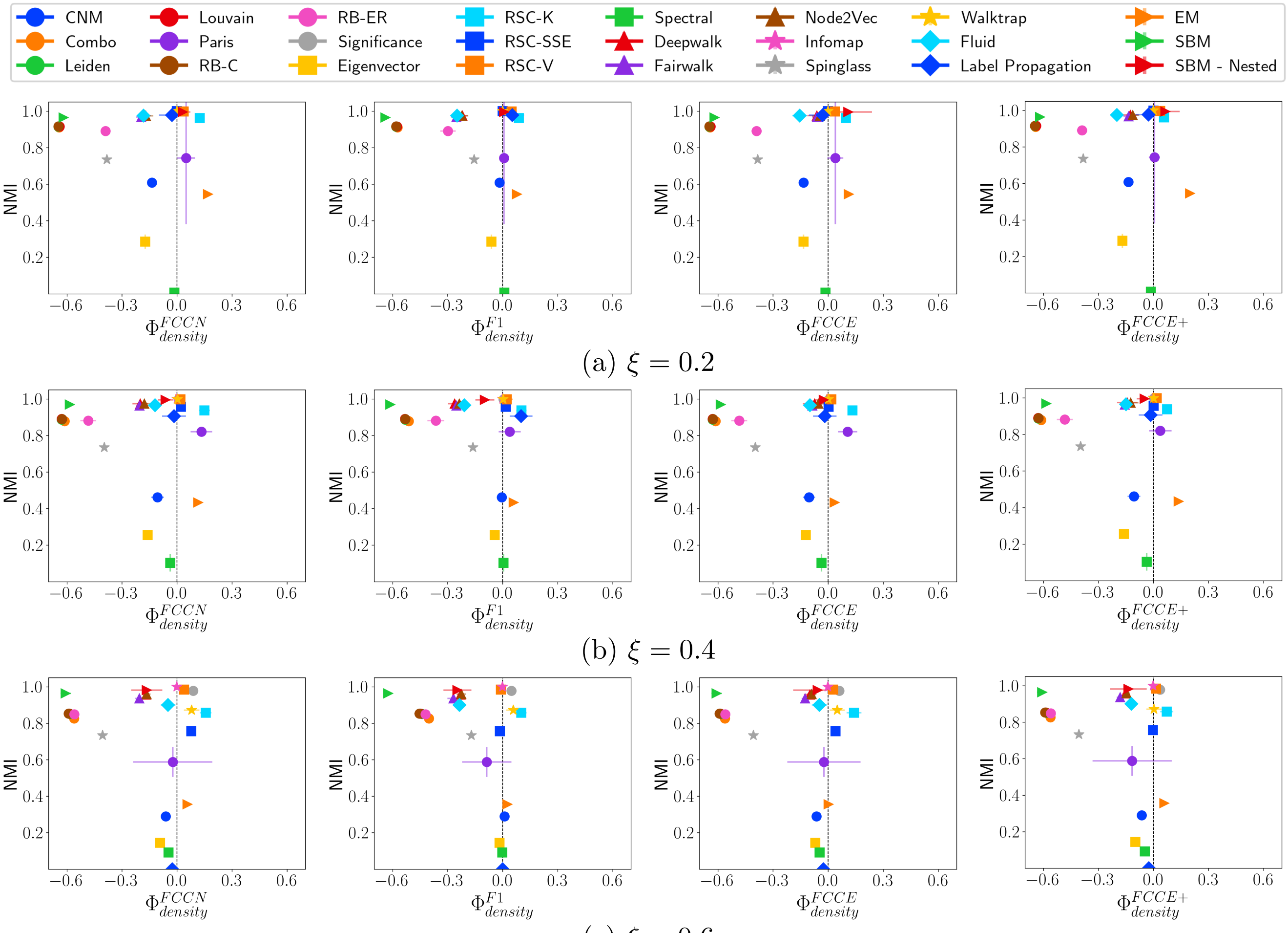
6.1.3 HICH-BA Results Mapped CPMs

Multiple Majority HICH-BA

Results for CD methods performed on the MMaj variant of the HICH-BA networks can be seen in Figure 6.7. Figure 6.7a shows Φ_{size}, Figure 6.7b shows $\Phi_{density}$, and Figure 6.7c shows $\Phi_{conductance}$. Table 6.1 shows that some CD methods predicted far too many communities. RB-ER and Significance predict more than 3000 communities too many. CD methods that score high in terms of NMI are Combo, Leiden, Louvain, RB-C, Spinglass, SBM, and SBM-Nested. Notably, all these methods have generated more than 21 predicted communities when 9 was the correct number. They were still able to score well in terms of performance.

For Φ_{size}, most CD methods favor larger communities, across all CPMs. This includes the high-performing methods that were previously mentioned. SBM and SBM-Nested are notably biased toward smaller communities for Φ^{FCCE}_{size} and Φ^{FCCE+}_{size}.

For $\Phi_{density}$, methods are spread with different fairness scores. The five best-performing CD methods favor sparse communities. SBM and SBM-Nested are also seen to be more biased for Φ^{FCCE}_{size} and Φ^{FCCE+}_{size}, again showing this unusual shift. The standard deviation for $\Phi_{density}$ is



(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

Figure 6.5: NMI and Φ^{CPM} scores in regard to **density** with mapped CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.

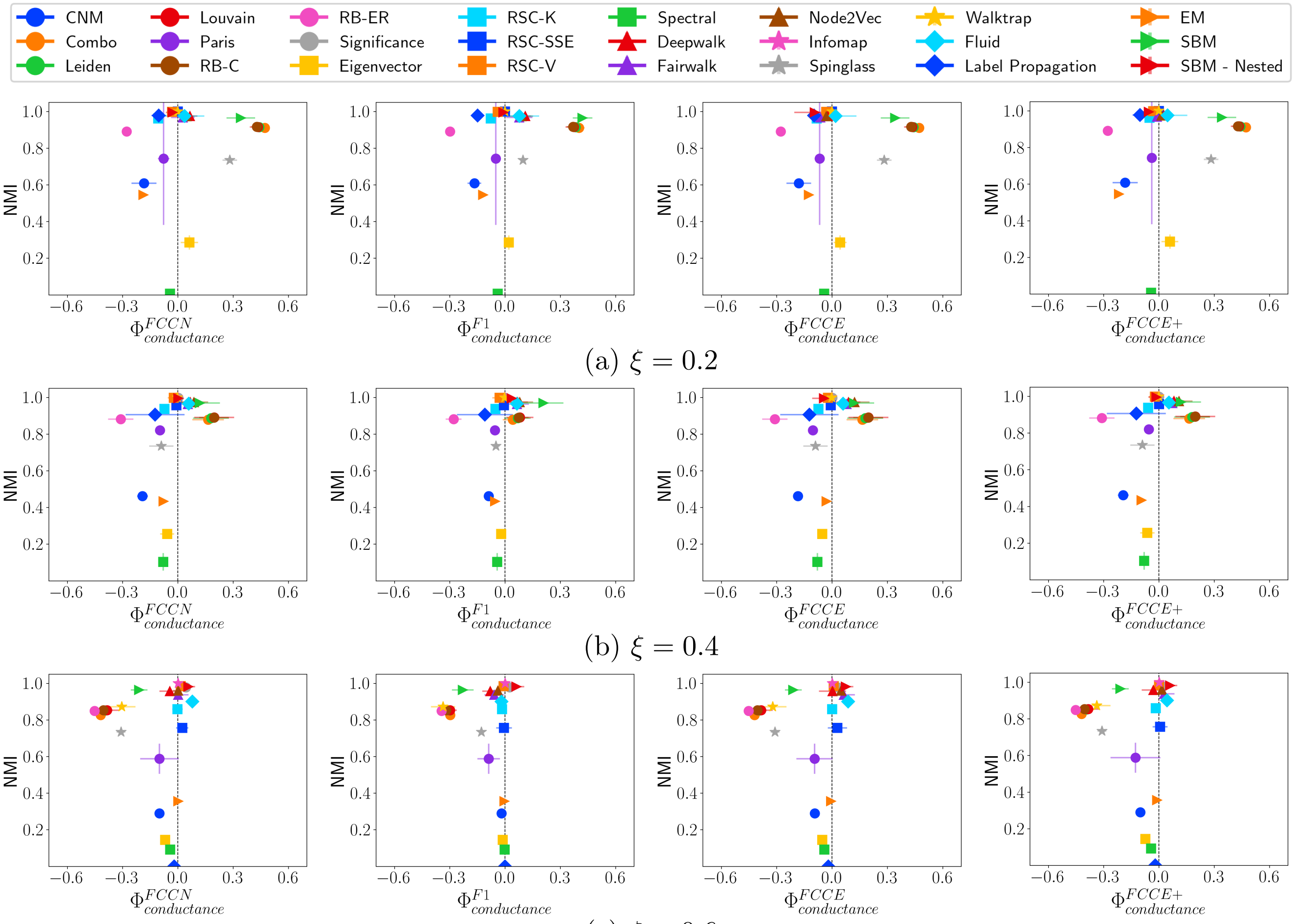
higher than we find for LFR and ABCD networks.

For $\Phi_{conductance}$, the standard deviation increases even further for all mapped CPMs. This indicates a very volatile prediction and mapping process, as the standard deviation for NMI is not as high. Almost all CD methods have a slight bias toward low-conductance communities. The best performing CD methods are all reasonably fair.

Multiple Minority HICH-BA

The results gathered from applying CD methods to HICH-BA networks of the MMin variant can be seen in Figure 6.8. Here, we can see that most CD methods were not able to score a high NMI value. This probably depended on whether the method was able to detect the largest ground-truth community. Methods that score high in terms of performance are Paris, RSC-V, Label Propagation, SBM and SBM-Nested. Three methods achieve scores within 0.4 and 0.6, but all others score lower than 0.29. These are spread in terms of Φ, regardless of the property of the analyzed community.

Fairness in regard to size, Φ_{size}, does not reflect the idea that well-performing CD methods



(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

Figure 6.6: NMI and Φ^{CPM} scores in regard to **conductance** with mapped CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.

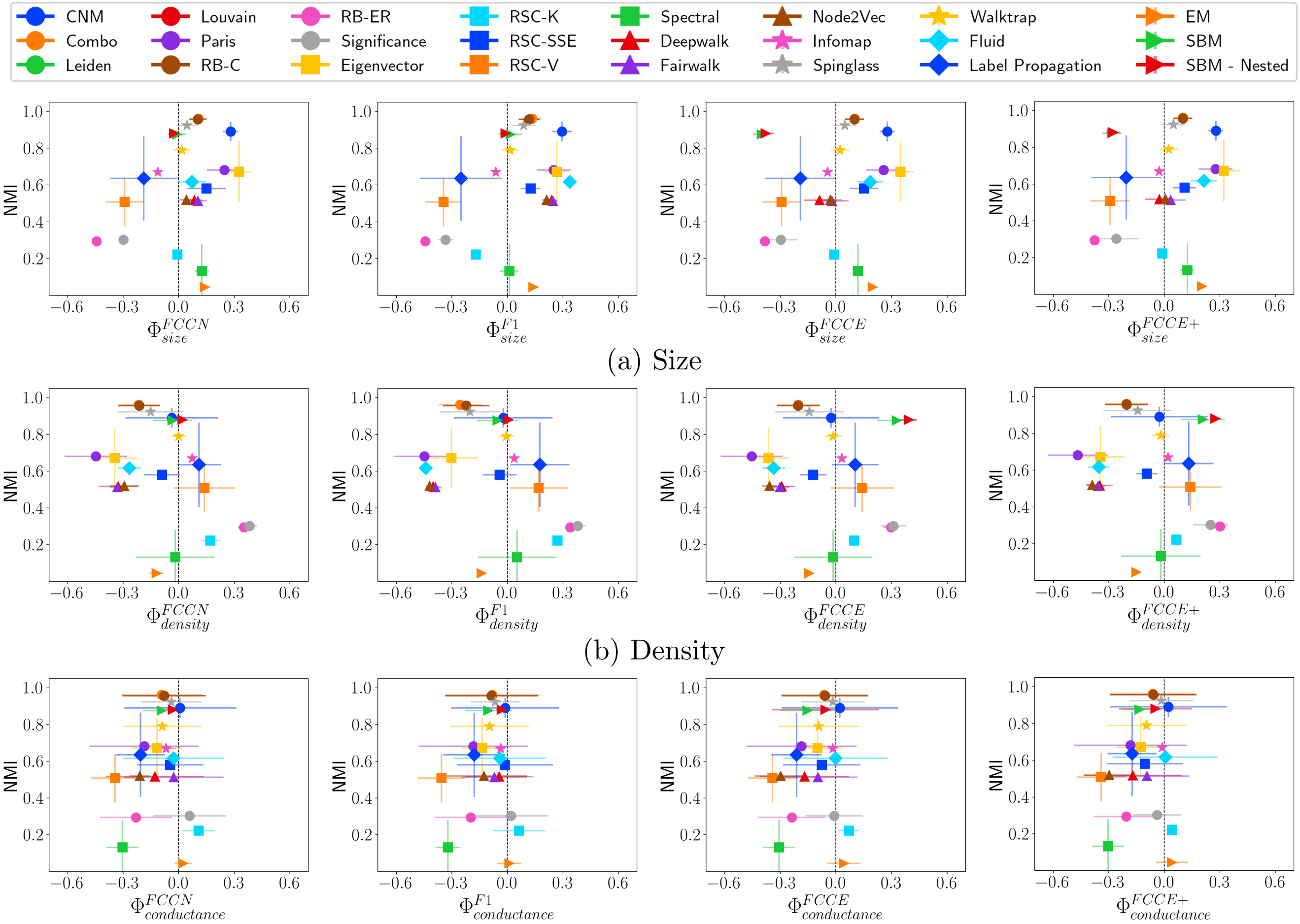
must have detected the largest community. If this had been the case, these methods would have large values of Φ_{size}, which is not the case.

6.1.4 Real-World Results Mapped CPMs

We have performed experiments on three real-world networks: Polbooks, Football, and Eu-core. Results for ϕ_{size} can be seen in Figure 6.9. For Polbooks (Figure 6.9a), all CD methods score similar NMI scores, around 0.5. Most methods favor the larger communities, except RSC-K, RSC-SSE, and Significance. These exceptions with fair Φ_{size} are not present when the CPM is F1.

The bias toward larger communities is also evident in the results gathered from the Football network, as seen in Figure 6.9b. Performance is much higher for most CD methods. All CD methods, except EM and Spectral, score higher for all NMI scores found by the results on Polbooks.

Figure 6.9c shows the results from Eu-core. Again, most CD methods favor larger communities. This is not the cases for Deepwalk, Node2Vec, RSC-SSe, and RSC-K when the CPM is FCCE. Notable is Significance, which is very fair and scores high in terms of NMI.



(a) Size

(b) Density

(c) Conductance

Figure 6.7: NMI and Φ_p^{CPM} scores with mapped CPMs for CDs methods applied on **HICH-BA** networks with $n = 10,000$, containing multiple majority communities (**MMaj**).

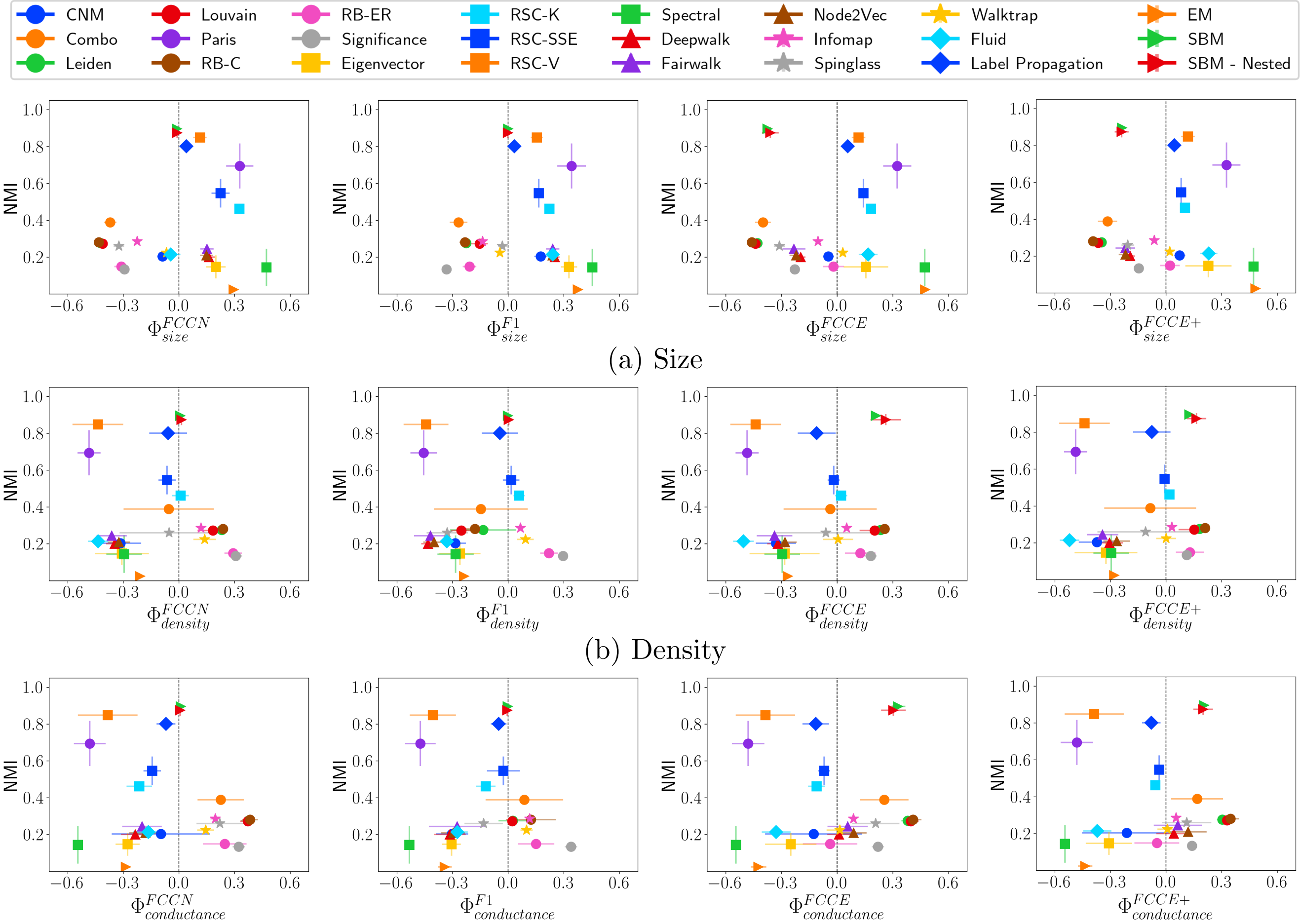
6.2 Results Global CPMs

Figures 6.1 to 6.8 show values of Φ_p^{CPM} and NMI for global CPMs. NMI is plotted against Φ_p^{CPM} in these figures to give insight into the performance-fairness trade-off.

6.2.1 LFR Results Global CPMs

Fairness - Size

Figure 6.10 shows NMI and Φ_{size}^{CPM} results for global CPMs. When the CPM is SWFCCE and SWFCCE+, we find that the CD methods appear more fair than when fairness was analyzed for their mapped counterparts. This is most evident for $\mu = 0.2$ and $\mu = 0.4$. Fairness with AvgF1 and SWF1 mimics the mapped F1 fairness measure closely, favoring larger communities. For $\mu = 0.6$, we see that almost all CD methods appear fair when the CPM is AvgF1.



(a) Size

(b) Density

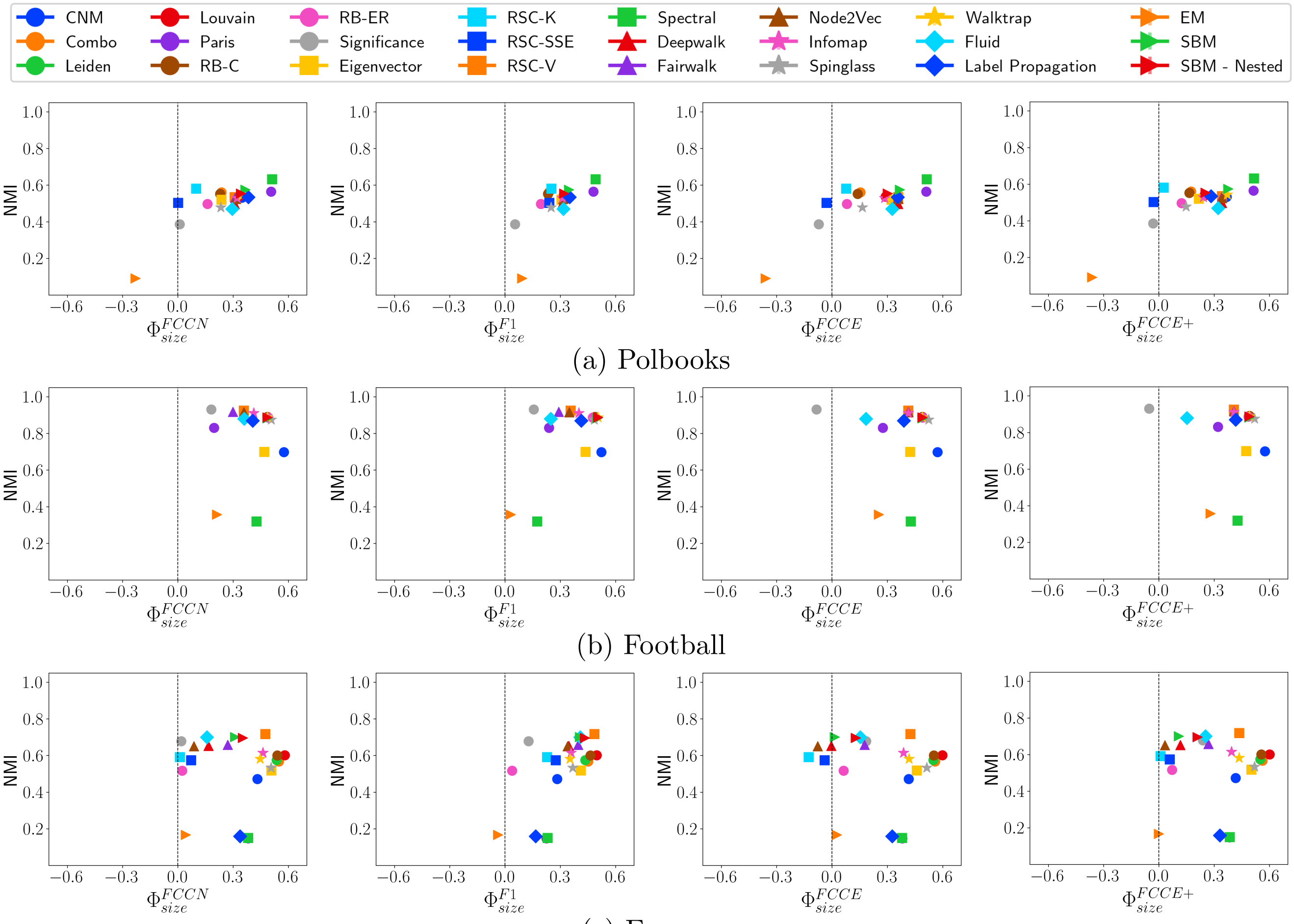
(c) Conductance

Figure 6.8: NMI and Φ_p^{CPM} scores with mapped CPMs for CDs methods applied on **HICH-BA** networks with $n = 10,000$, containing multiple minority communities with one majority community (**MMin**).

Fairness - Density

Figure 6.11 shows the fairness in regards to density for global CPMs. When $\mu = 0.2$, most CD methods perform well in terms of NMI and are all mostly fair. When $\mu = 0.4$, the bias toward dense communities is evident. As was the case with mapped CPMs, several CD methods stand out here. Combo, Leiden, Louvain, RB-C, and RB-ER are less biased toward dense communities and, with SWF1 as the CPM, this group is biased toward low-density communities.

When $\mu = 0.6$, CD methods are really unfair when the CPM is SWFCCE and SWFCCE+. SBM and SBM-Nested are much more fair than other well-performing CD methods. With higher values μ, AvgF1 and SWF1 show a clear correlation between high NMI scores and bias toward dense communities. The group of modularity-based CD methods that have often been found to be biased toward sparse networks fall in between this line, striking a balance between performance and fairness.



(a) Polbooks

(b) Football

(c) Eu-core

Figure 6.9: NMI and Φ^{CPM} scores in regard to **size** with mapped CPMs for CDs methods applied on the real-world networks Polbooks, Football, and Eu-core.

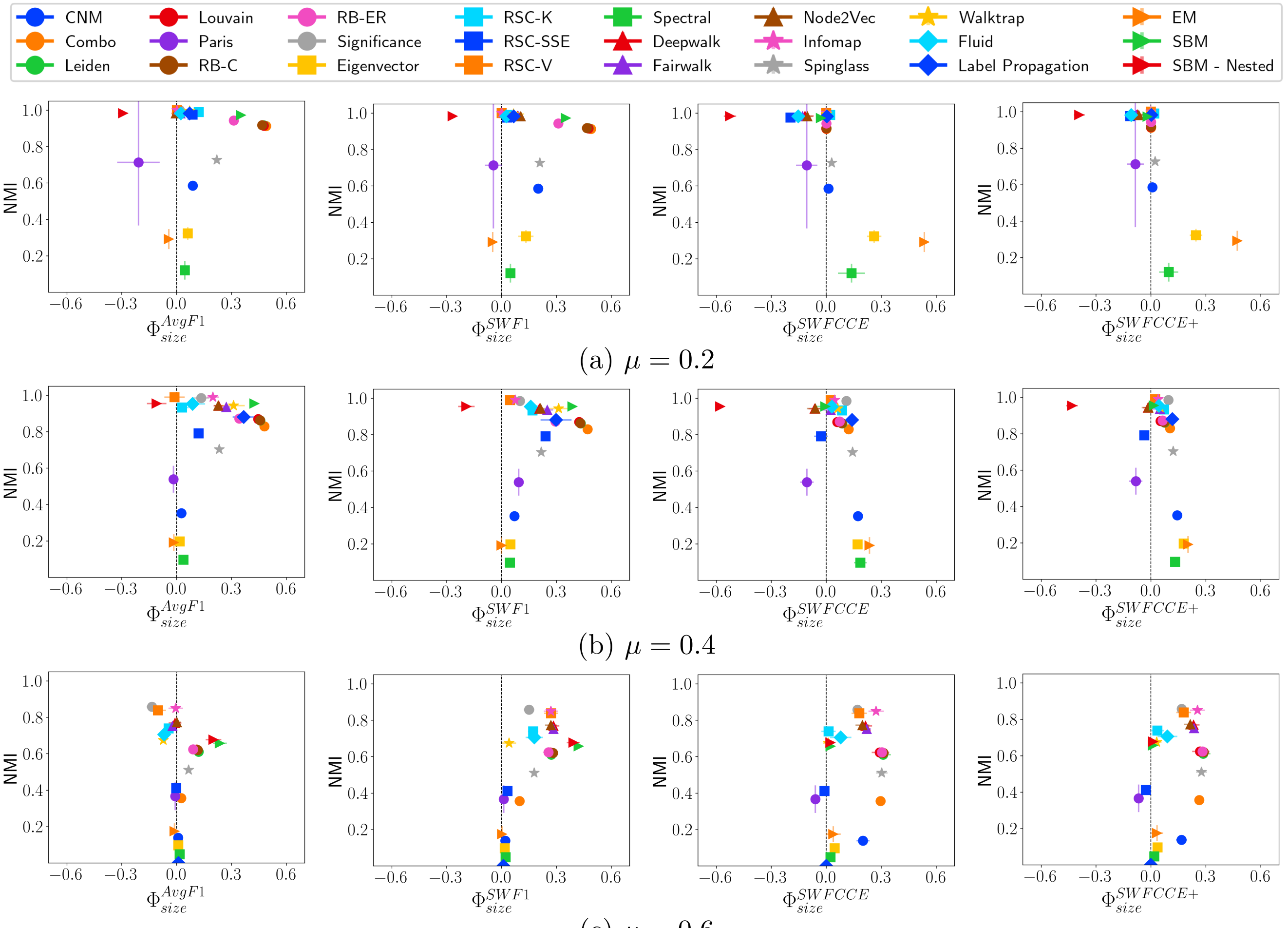
Fairness - Conductance

Figure 6.12 shows results for $\Phi_{conductance}^{CPM}$ with global CPMs. Similarly to the results we found with mapped CPMs (see Figure 6.3) we find that all CD methods favor low-conductance communities when $\mu = 0.6$. There is a connection between performance and bias toward low-conductance communities, though this is not straightforward with SWFCCE and SWFCCE+ as CPMs. Here SBM and SBM-nested are more fair than other methods while performing well in terms of NMI. For lower values of μ we see similar results to the mapped CPM variants.

6.2.2 ABCD Results Global CPMs

Fairness - Size

Figure 6.13 shows Φ_{size}^{CPM} results gathered from ABCD networks with global CPMs. As mentioned before, most CD methods perform well on ABCD networks, even with ξ set to 0.6. We see that Φ_{size}^{SWFCCE} and Φ_{size}^{SWFCCE} favor smaller communities more than their mapped counterparts. The LFR results suggested they became more fair, by moving toward $\Phi_{size}^{SWFCCE(+)} = 0$, here



(a) $\mu = 0.2$

(b) $\mu = 0.4$

(c) $\mu = 0.6$

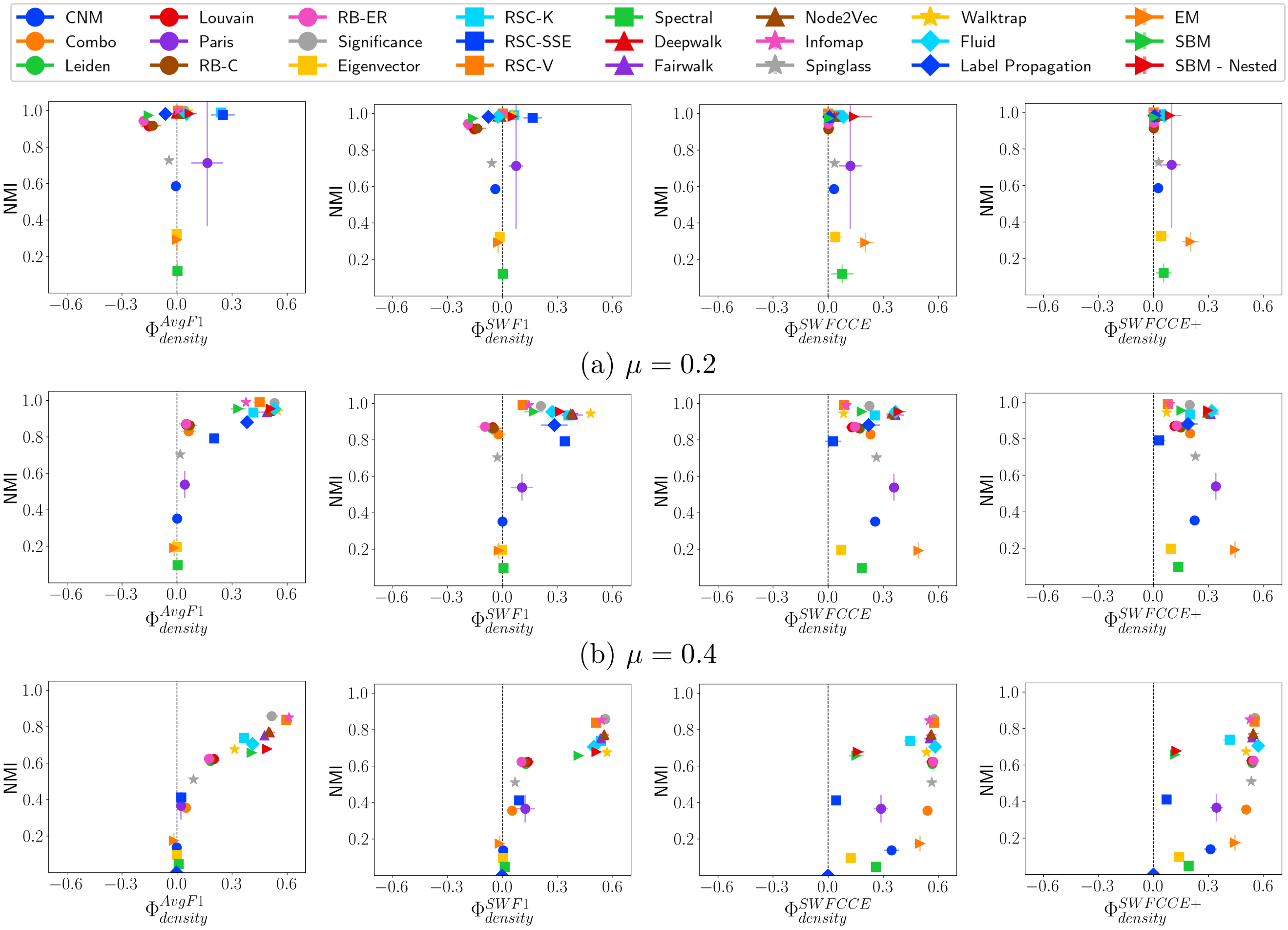
Figure 6.10: NMI and Φ^{CPM} scores in regard to **size** with global CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

$\Phi_{size}^{SWFCCE(+)}$ becomes negative for many CD methods. For Φ_{size}^{AvgF1} and Φ_{size}^{SWF1}, we find that most of the CD methods are fair or in favor of small communities. However, the modularity-based group of Combo, Leiden, Louvain, RB-C, and RB-ER favors large communities.

Fairness - Density

Figure 6.14 shows $\Phi_{density}^{CPM}$. What we saw with $\Phi_{size}^{SWFCCE(+)}$ on ABCD networks, where bias moved toward favoring small communities, we now see with $\Phi_{density}^{SWFCCE(+)}$ shifting to more bias toward high density communities. Aside from this, these results of $\Phi_{density}^{CPM}$ are very similar to the mapped variants of the CPMs.

When the CPM is AvgF1 or SWF1, for $\xi = 0.2$ and $\xi = 0.4$, CD methods range from favoring sparse networks, to being fair, to favoring dense networks. The modularity-based group favors sparse networks for all values of ξ, while RSC-V, Significance, Fluid, and Walktrap favor dense communities heavily only when $\xi = 0.6$.



(a) $\mu = 0.2$

(b) $\mu = 0.4$

(c) $\mu = 0.6$

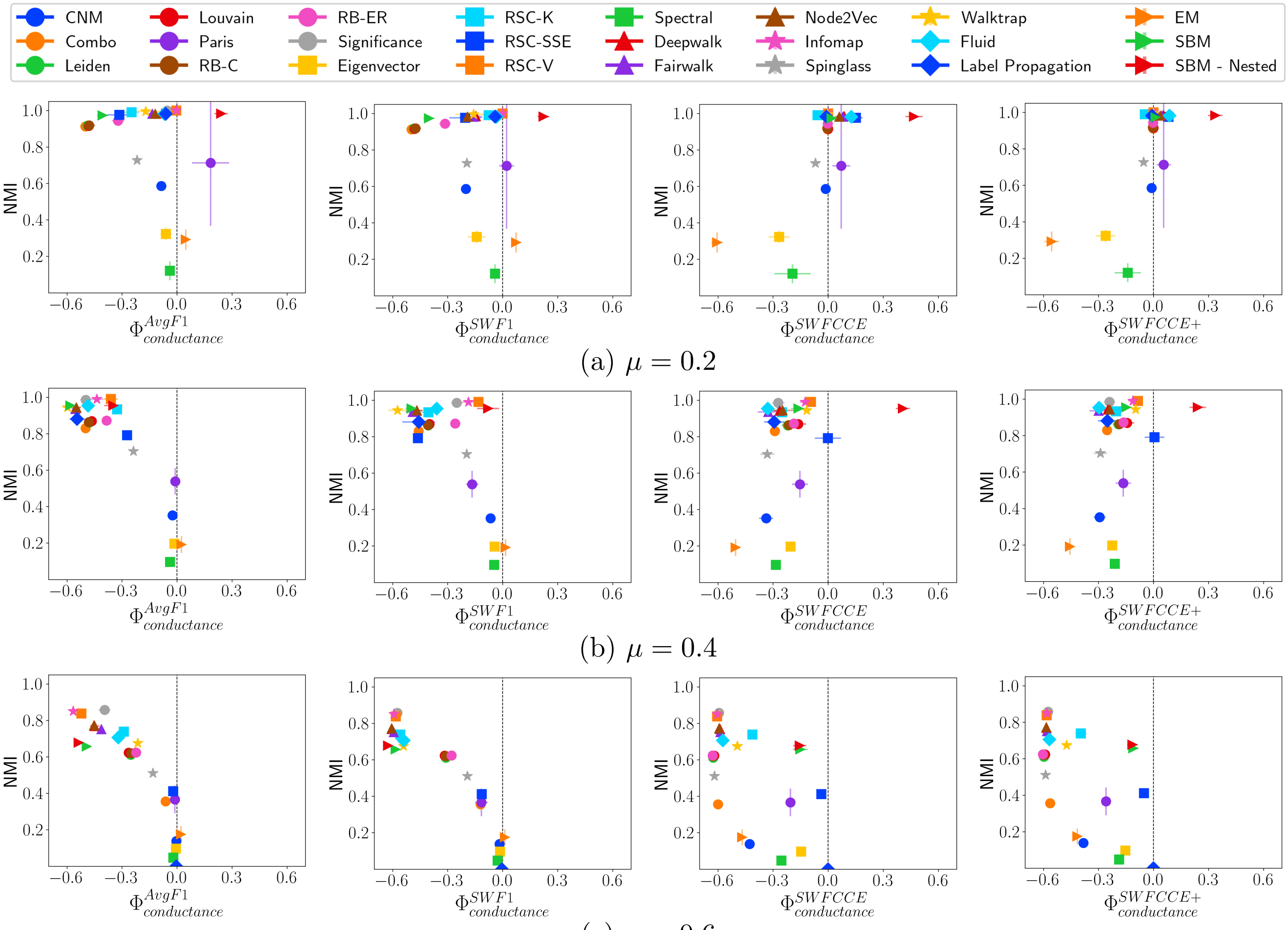
Figure 6.11: NMI and Φ^{CPM} scores in regard to **density** with global CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

Fairness - Conductance

Figure 6.15 shows fairness in regard to conductance for global CPMs. Here, all CD methods are fair in terms of Φ^{SWFCCE}_{size} for all values of ξ. Most CD methods are fair for Φ^{AvgF1}_{size} and Φ^{SWF1}_{size} as well, but when $\xi = 0.6$ the modularity-based group of Combo, Leiden, Louvain, RB-C, and RB-ER is joined by Walktrap and Spinglass in a group that is more biased toward low-conductance communities.

6.2.3 Real-World Results Global CPMs

Overal, the results gathered using global CPMs to analyze Φ_{size} is very similar to the results that were gathered using mapped CPMs. This can be seen by comparing the global results (Figure 6.16) with the mapped results (Figure 6.9). For Polbooks, a difference is that CD methods get more similar values for Φ^{SWFCCE}_{size} and $\Phi^{SWFCCE+}_{size}$. They all have shifted toward a more fair scores. This is also the case for the Football network. For Eu-core, values of Φ^{SWFCCE}_{size} and $\Phi^{SWFCCE+}_{size}$ have also decreased in value and for multiple CD methods, this meant that they now have negative



(a) $\mu = 0.2$

(b) $\mu = 0.4$

(c) $\mu = 0.6$

Figure 6.12: NMI and Φ^{CPM} scores in regard to **conductance** with global CPMs for CDs methods applied on **LFR** networks with $n = 10,000$.

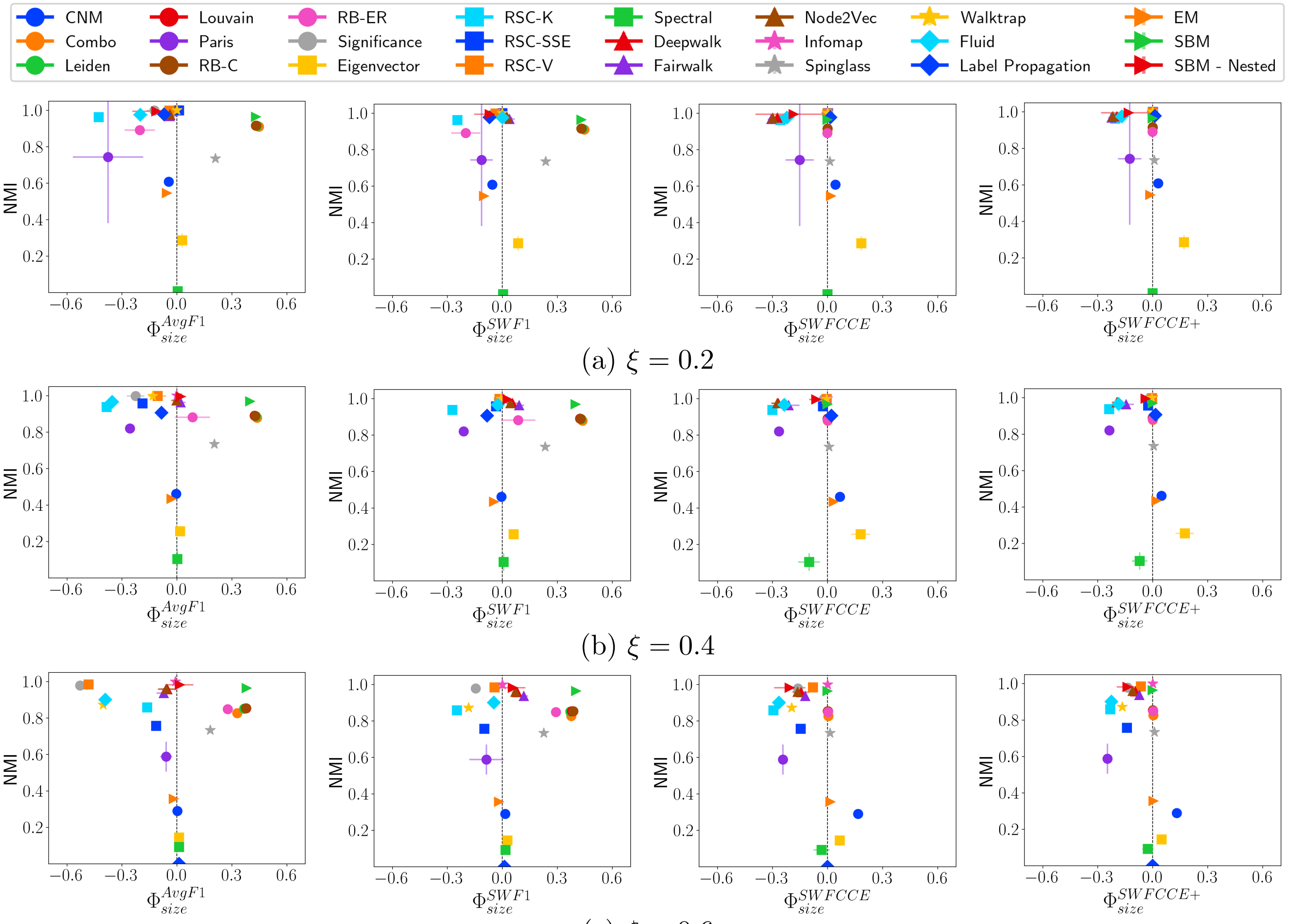
values and show biased results toward smaller communities.

Results from AvgF1 and SWF1 resemble the mapped F1 results across the real-world networks. The Φ^{AvgF1}_{size} results includes many fair methods and some that appear to favor minority communities, such as Significance and RB-ER.

6.3 Results Partition Quality Measures

As described in Section 5.3, other measures than NMI exist that also quantify how well a predicted partition describes the ground-truth community structure. These include RMI, VI, ARI, PF1, and NF1. To show the differences between these metrics, we show the performance scores obtained from applying the 24 CD methods on the set of LFR networks with $\mu = 0.4$. Figure 6.17 shows the performance scores plotted against Φ^{CPM}_{size} with the mapped CPMs, and these performance scores can be seen in tabular form in Appendix C.26. All partition quality measures can be seen for the synthetic networks in Appendices C.25 to C.32.

As described in Section 5.3, better predictions get higher scores for these partition quality



(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

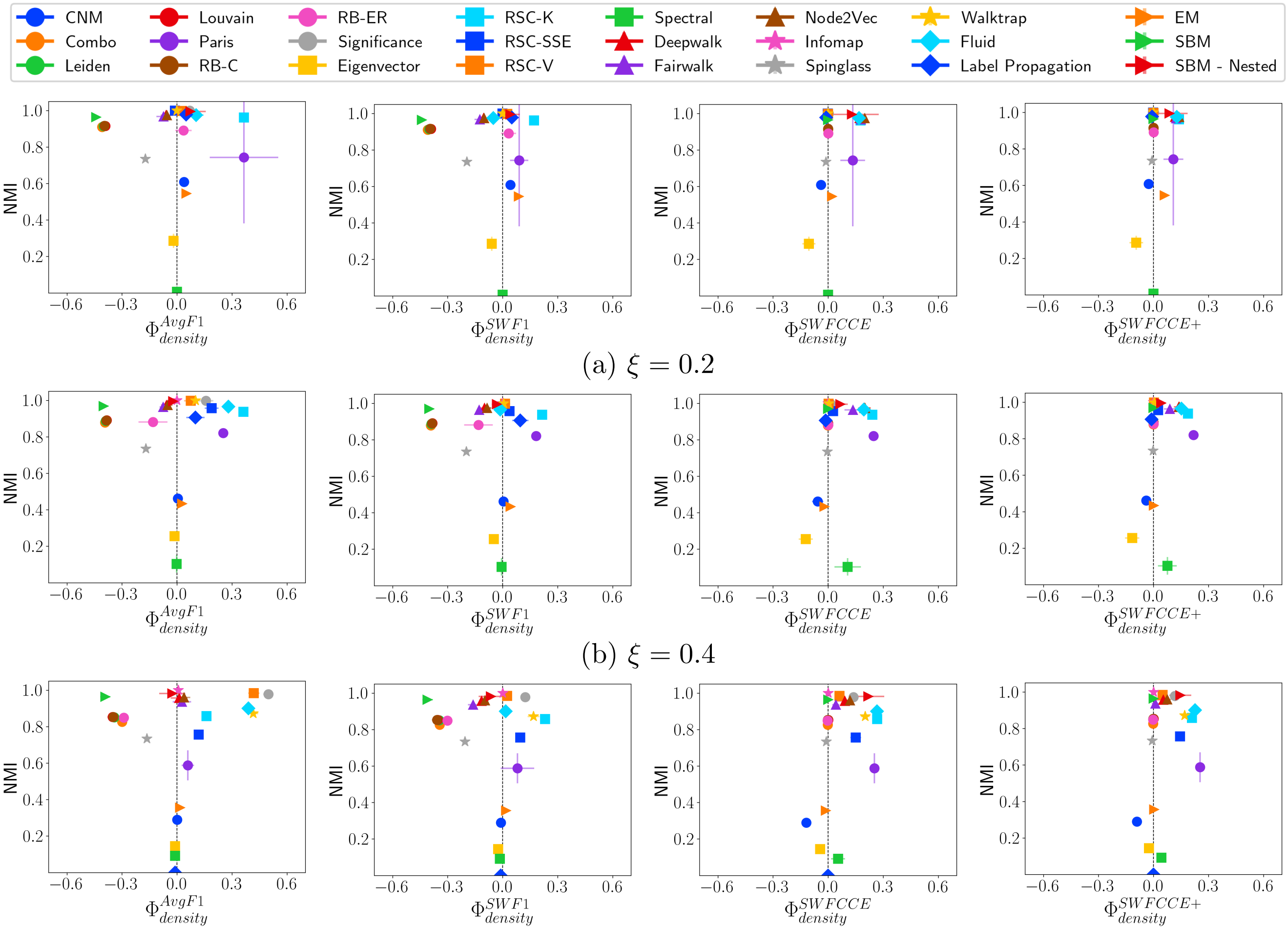
Figure 6.13: NMI and Φ^{CPM} scores in regard to **size** with global CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.

metrics. This is not the case for VI, where an ideal prediction gets a VI score of 0. VI can also exceed 1 and we have several CD methods that are not visible in the VI plots as they exceed the set y-limit of 3. These are Eigenvector (9.1), EM (8.4), Spectral (8.2), CNM (6.7), Paris (6.4), and Spinglass (3.7). All these also the lowest performing CD methods in terms of NMI, PF1, and NF1. Although these methods agree on the worst CD methods, they do not agree on the best performing methods.

Two CD methods place in the top-5 best methods across all partition quality metrics: Infomap and RSC-V. Significance and SBM place in three of the six top-5's. However, Significance is also ranked the second-worst in terms of RMI. Table 6.1 shows why this is the case. Significance has predicted ∼193 too many communities for this network, and RMI penalizes predictions that overestimate the true number of communities.

Unsurprisingly, just as NMI and RMI, PF1 and NF1 are often in agreement about the quality of a predicted partition. This is because both pairs of partition similarity measures share baselines: mutual information for RMI and NMI, and many-to-one F1-scores for PF1 and NF1.

We do have to note that the implementation for RMI might not be correct. We used CDlib's implementation [70] which offers a normalized variant of RMI such that this measure does not



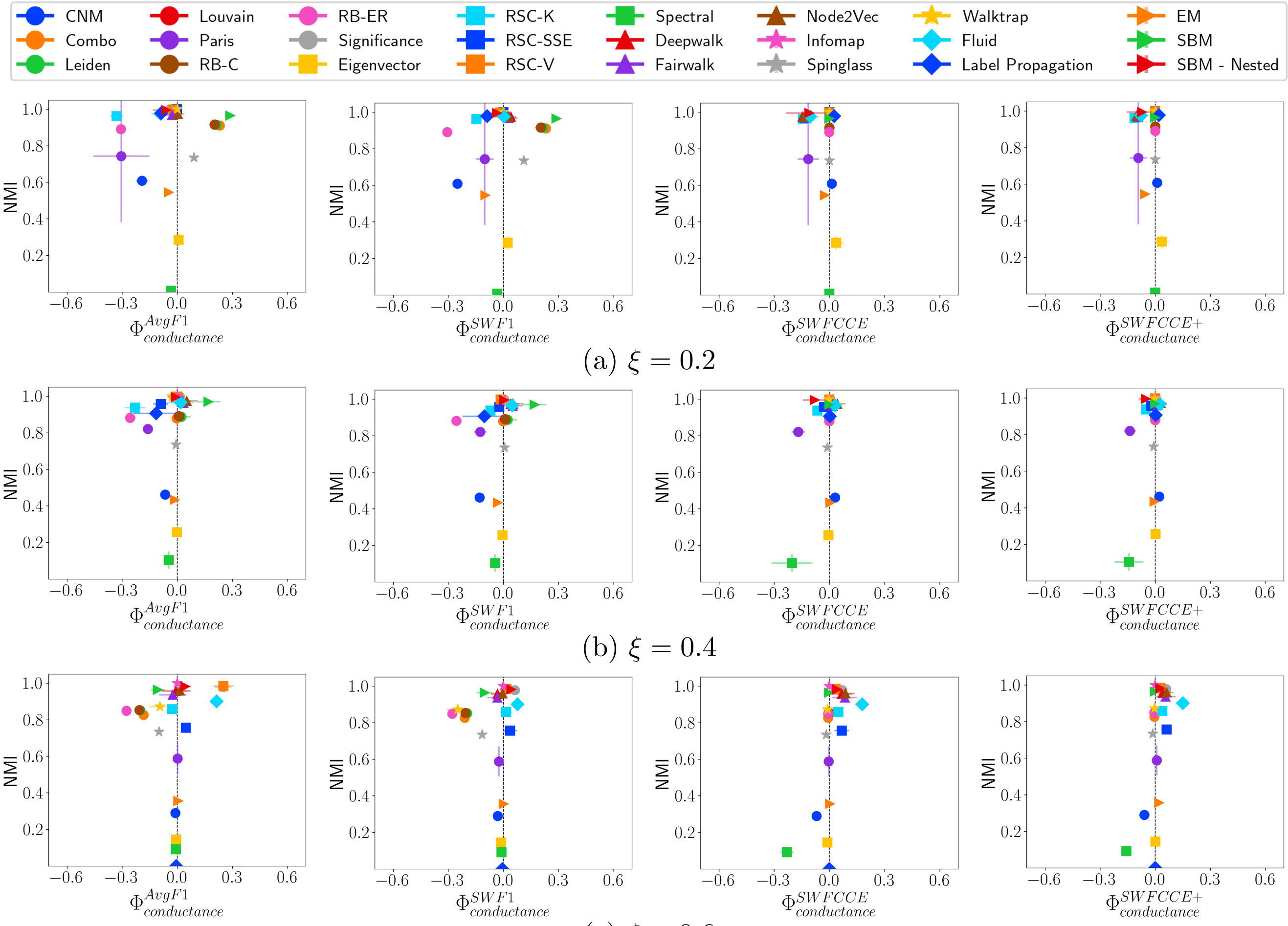
(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

Figure 6.14: NMI and Φ^{CPM} scores in regard to **density** with global CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.

exceed 1. However, we find that on several occasions RMI is incorrectly calculated or not properly normalized. For example, in Appendix C.27 read the RMI score for Walktrap as 190. In total, we find results that exceed 1 three times.

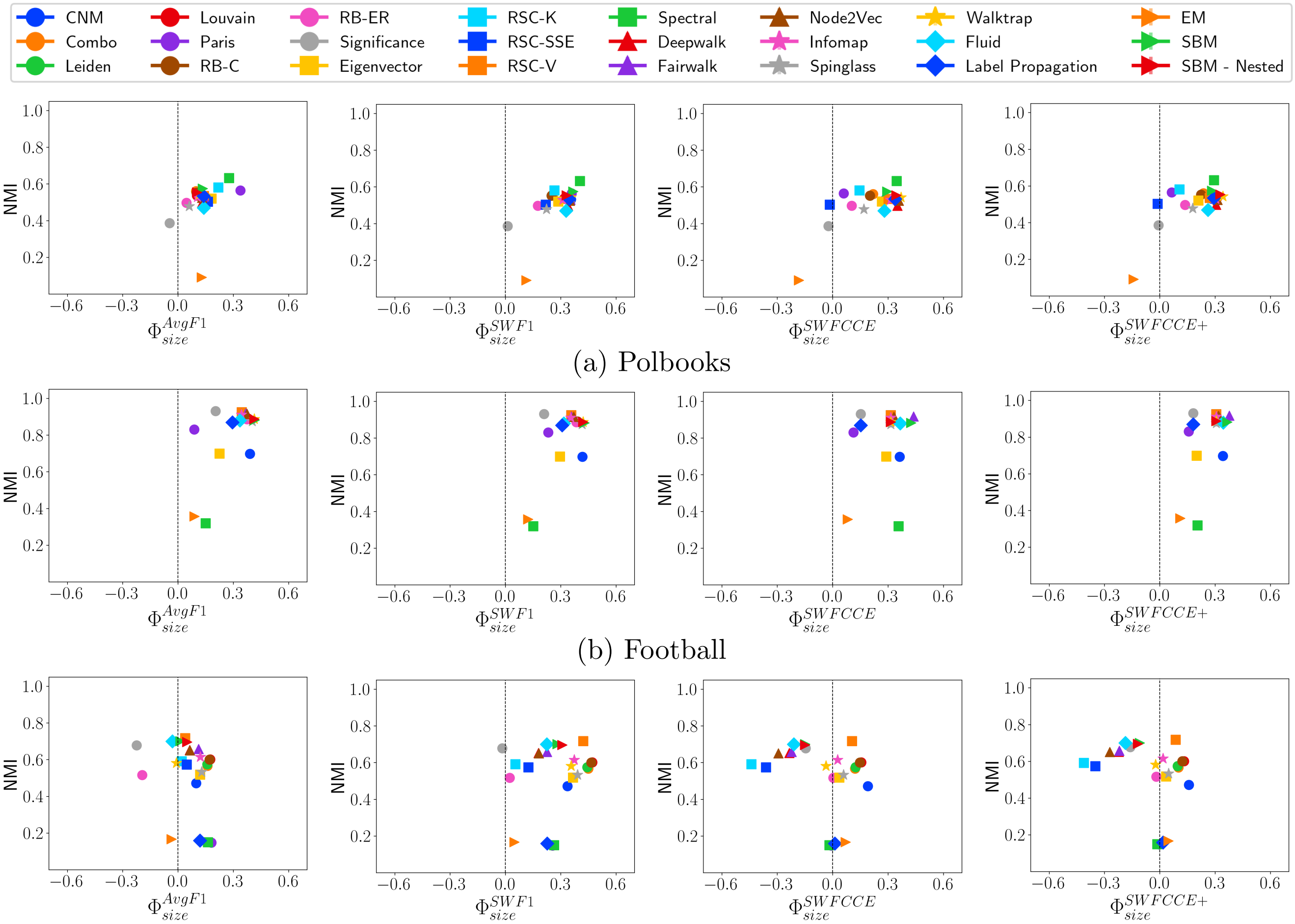


(a) $\xi = 0.2$

(b) $\xi = 0.4$

(c) $\xi = 0.6$

Figure 6.15: NMI and Φ^{CPM} scores in regard to **conductance** with global CPMs for CDs methods applied on **ABCD** networks with $n = 10,000$.



(a) Polbooks

(b) Football

(c) Eu-core

Figure 6.16: NMI and Φ^{CPM} scores in regard to **size** with global CPMs for CDs methods applied on the real-world networks Polbooks, Football, and Eu-core.

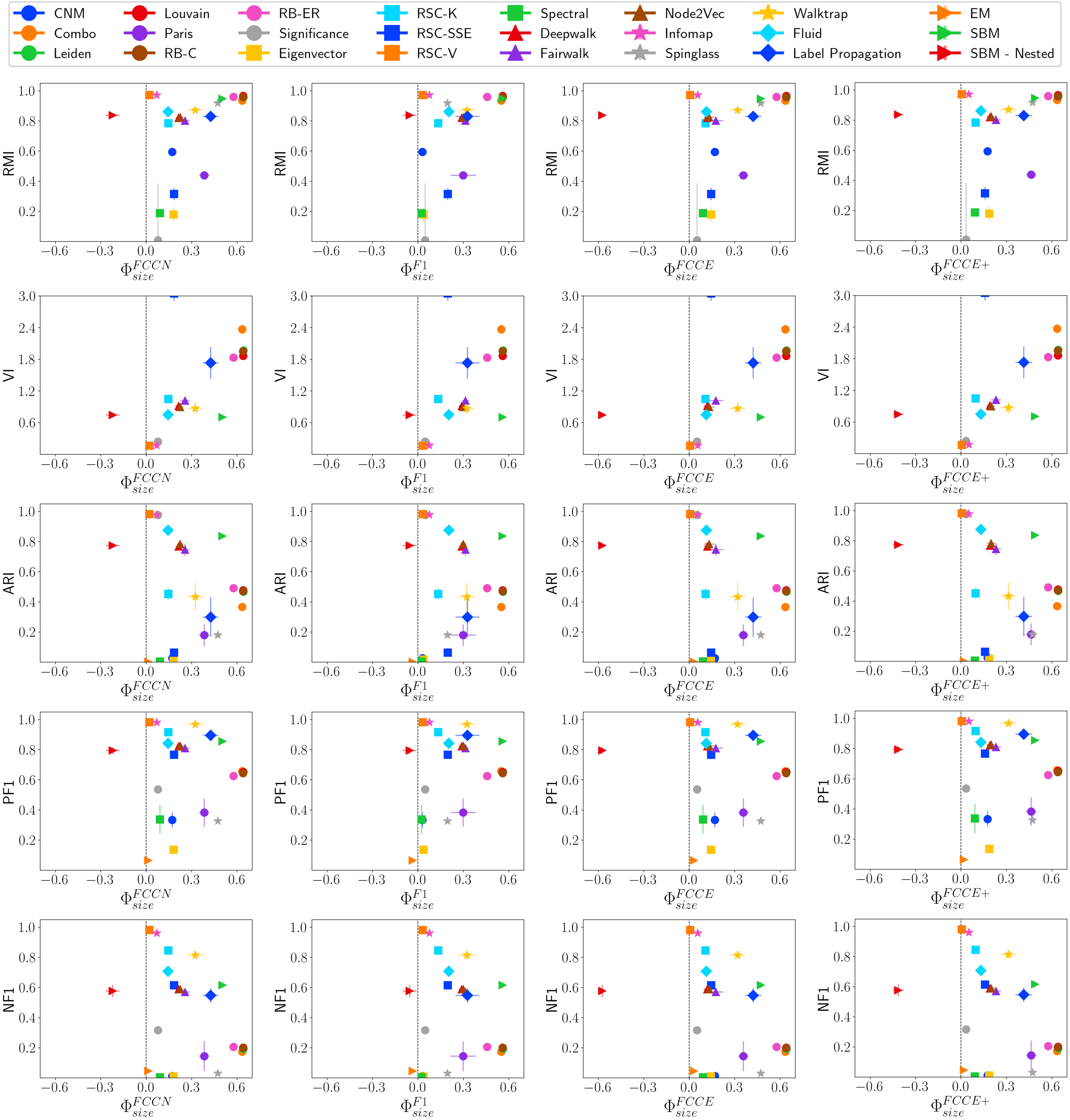


Figure 6.17: Partition quality metrics and Φ^{CPM}_{size} scores with mapped CPMs for CDs methods applied on **LFR** networks with $n = 10,000$, $\mu = 0.4$.

Chapter 7

Conclusions

Community detection (CD) offers help in understanding network structure and node behavior. Communities have different sizes, densities, and levels of connectivity throughout the network. Most CD algorithms do not take structural inequalities of communities into account which leads to biased outcomes for community detection and downstream SNA tasks. Certain types of communities cannot be properly identified by CD methods. In this thesis, we have introduced the group-fairness metric Φ_p^{CPM} that measure bias with respect to a community property p. This measure Φ is based on the fairness definition that all communities should be detected equally well. This works by comparing different scores of community-wise performance metrics (CPMs), which we introduce in this work. We propose mapped and global CPMs to quantify how well a community is identified. The proposed measure gives valuable insights that can be used to design fair community detection methods, or in what aspects current CD methods should be improved, or CD parameters should be set. We have performed experiments with 24 CD methods, which we have applied to real-world networks and synthetic networks. Synthetic networks were generated using the LFR benchmark, the ABCD algorithm, and the HICH-BA model. For LFR and ABCD, we have analyzed networks with varying degrees of community mixing. HICH-BA networks were generated using one of two structures that decide the number of majority communities and the overall community structure.

Next we discuss insights gained by analyzing 24 CD methods based on their fairness by Φ_p^{CPM} and their performance. NMI is the most prominently presented measure of prediction quality. Other measures include RMI, VI, ARI, PF1, and NF1. We find that well-performing CD methods include Significance, RSC-K, Infomap, and SBM(-Nested). These CD methods perform well for LFR and ABCD networks; these networks mimic real networks to a high degree. These methods were biased, as were other high-performing methods, toward large, dense, and low-conductance communities. However, we find that in many instances high performance goes hand in hand with fairness. This was especially evident for ABCD networks.

CD methods were able to perform very well for ABCD networks, even when ξ was set to 0.6. Because many communities were well detected, these CD methods achieve high fairness scores. For LFR networks, there is a clear correlation between Φ_p^{CPM} and performance, especially when the mixing parameter μ is higher. Methods that perform better are more biased toward large, dense, low-conductance communities.

A group containing several modularity-based approaches containing Combo, Leiden, Louvain, RB-C, and RB-ER received very similar scores in terms of performance and fairness. Although these methods achieved high performance, they were often found to be biased, but not in the way other CD methods were. These methods favored less dense communities and were much more biased in ABCD networks than other CD methods.

Performance and bias results from the HICH-BA networks differ from the LFR and ABCD results. With the community structure with multiple majority communities, the CD methods CNM, Combo, Leiden, Louvain, RB-C, and Spinglass stood out in terms of performance. Although we found an overall bias toward low-conductance communities in both HICH-BA structures, as we did for LFR and ABCD networks, the CD methods varied in their bias regarding community size and density. In the HICH-BA structure with a single large majority community, performance was found to be lower. Only seven CD methods achieved NMI scores higher than 0.4.

Based on our results, we advise people to use the CD methods Significance and Infomap. These methods consistently achieved high performance, and unlike other high-performing methods, these do not require the user to set the number of communities to predict. These methods detected many communities well and often scored highly in fairness, with Significance outperforming other methods in real-world networks. The selection of CD methods should not be based solely on our fairness metrics, as a fair method could also detect all communities equally poorly. Other CD methods could outperform these metrics given optimal parameter settings. We have used the default settings of CDlib; better parameters could be found for synthetic networks but this requires more experimentation. Significance and Infomap require no parameters to be given, making their achieved results in terms of fairness and performance more impressive. Interestingly, Significance does not join the similar scoring group of modularity-based methods. While Significance works by optimizing a goodness score like other methods from this class does, it optimizes for Significance instead of modularity.

Depending on what SNA task is undertaken after identifying communities, it can be advantageous to look at fairness scores derived from certain CPMs. Analyzing the bias with regard to conductance is important for influence maximization. If one aims to spread information to remote parts of the network, the decision could be made to select a CD method that is heavily biased toward finding low-conductance communities. When precision of communities needs to be taken into account, select a fairness metric based on a CPM that uses the F1 measure.

Regarding the proposed global metrics, we find that these metrics do not offer substantial additional information on the fairness of a CD method. Additionally, mapped CPMs reflect the one-to-one nature of community detection better than the one-to-many structure of global CPMs.

As expected, the CPMs FCCE and FCCE+ offer similar fairness results. The difference that can be found between them is that with Φ_{size}, FCCE+ is consistently more biased toward larger communities: $\Phi_{size}^{FCCE+} \geq \Phi_{size}^{FCCE}$ and $\Phi_{size}^{SWFCCE+} \geq \Phi_{size}^{SWFCCE}$. This is the case across CD methods and different networks. This happens because larger communities have more nodes, and thus there is a larger number of nodes that can be identified as false bridges. This improves the FCCE+ score relative to FCCE.

7.1 Future Work

This work has analyzed undirected, unweighted, connected networks with non-overlapping communities. However, there are various other type of complex networks conveying more information, such as overlapping communities, hyperedge networks and so on. CD methods have been proposed to apply to other types of networks and communities. The CPMs and fairness metric Φ work by comparing ground-truth communities with predicted communities. This basic principle can apply to other types of community, such as overlapping communities, hierarchical communities, or communities with outliers. Future work can also include further fairness analysis toward other community attributes. Looking at the number of intracommunity edges (community volume) could yield results that fall between the results we have gathered by looking at community size and density.

Our method could be improved by changing the approach we use to map ground-truth communities to predicted communities. This is an instance of the linear assignment problem. We have used an iterative approach that chooses the pair of communities that has the highest Jaccard similarity at each step, but one can explore other approaches. Furthermore, future work could focus on a different definition of fairness. Many different definitions exist within SNA [77]. By applying linear regression, we derive a metric to identify bias; however, this approach abstracts away from the actual community-wise performance values. In-depth research into community-wise performance could yield interesting results.

Lastly, we have found that there exists some level of correlation between community properties in synthetic networks. Because not many large real-world networks have ground-truth community information, we are reliant on these network generation models. More analysis needs to be done on the communities that are produced by LFR, ABCD, and HICH-BA. Additionally, more real-world networks need to be annotated with community information for CD methods to be compared on results gathered from the real world. It will also be helpful if one could propose group fairness metrics that do not require ground-truth information of the community structure.

References

[1] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Diameter of the world-wide web. *Nature*, 401(6749):130–131, September 1999.

[2] Alex Arenas, Leon Danon, Albert Diaz-Guilera, Pablo Gleiser, and Roger Guimerà. Community analysis in social networks. *European Physical Journal B*, 38, 12 2003.

[3] Nino Arsov and Georgina Mirceva. Network embedding: An overview, 2019.

[4] Albert-László Barabási and Márton Pósfai. *Network science*. Cambridge University Press, Cambridge, 2016.

[5] Ruben Becker, Gianlorenzo D'angelo, Sajjad Ghobadi, and Hugo Gilbert. Fairness in influence maximization through randomization. *Journal of Artificial Intelligence Research*, 73:1251–1283, 2022.

[6] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, October 2008.

[7] Thomas Bonald, Bertrand Charpentier, Alexis Galland, and Alexandre Hollocou. Hierarchical graph clustering using node pair sampling, 2018.

[8] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[9] Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ACM Computing Surveys*, 56(7):1–38, 2024.

[10] Tanmoy Chakraborty, Ayushi Dalmia, Animesh Mukherjee, and Niloy Ganguly. Metrics for community analysis: A survey. *ACM Computing Surveys (CSUR)*, 50(4):1–37, 2017.

[11] Nancy Chinchor. Muc-4 evaluation metrics. In *Proceedings of the 4th Conference on Message Understanding*, MUC4 '92, page 22–29, USA, 1992. Association for Computational Linguistics.

[12] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6), December 2004.

[13] Gennaro Cordasco and Luisa Gargano. Community detection via semi-synchronous label propagation algorithms. In *2010 IEEE international workshop on: business applications of social network analysis (BASNA)*, pages 1–8. IEEE, 2010.

[14] Soumita Das, Anupam Biswas, and Akrati Saxena. Dcc: A cascade-based approach to detect communities in social networks. In *International Conference on Computer Vision, High-Performance Computing, Smart Devices, and Networks*, pages 381–392. Springer, 2022.

[15] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6), December 2011.

[16] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On power-law relationships of the internet topology. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '99, page 251–262, New York, NY, USA, 1999. Association for Computing Machinery.

[17] Golnoosh Farnad, Behrouz Babaki, and Michel Gendreau. A unifying framework for fairness-aware influence maximization. pages 714–722, 04 2020.

[18] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3–5):75–174, February 2010.

[19] Santo Fortunato and Darko Hric. Community detection in networks: A user guide. *Physics Reports*, 659:1–44, November 2016.

[20] Ana L.N. Fred and Anil K Jain. Robust data clustering. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–II. IEEE, 2003.

[21] Linton Freeman. The development of social network analysis. *A Study in the Sociology of Science*, 1(687):159–167, 2004.

[22] Sorelle A. Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. The (im)possibility of fairness: different value systems require different mechanisms for fair decision making. *Commun. ACM*, 64(4):136–143, March 2021.

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

[24] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[25] Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

[26] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer series in statistics. Springer, 2009.

[27] Desmond J. Higham, Gabriela Kalna, and Milla Kibble. Spectral clustering and its use in bioinformatics. *Journal of Computational and Applied Mathematics*, 204(1):25–37, 2007. Special issue dedicated to Professor Shinnosuke Oharu on the occasion of his 65th birthday.

[28] Bin Hu, Wenmin Li, Xuesong Huo, Ye Liang, Minghui Gao, and Pei Pei. Improving louvain algorithm for community detection. In *2016 International Conference on Artificial Intelligence and Engineering Applications*, pages 110–115. Atlantis Press, 2016.

[29] Lawrence J. Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[30] Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. Forceatlas2, a continuous graph layout algorithm for handy network visualization designed for the gephi software. *PLOS ONE*, 9(6):1–12, 06 2014.

[31] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, October 2000.

[32] Bogumił Kamiński, Paweł Prałat, and François Théberge. Artificial benchmark for community detection (abcd)—fast random graph model with community structure. *Network Science*, 9(2):153–178, 2021.

[33] Fariba Karimi, Mathieu Génois, Claudia Wagner, Philipp Singer, and Markus Strohmaier. Homophily influences ranking of minorities in social networks. *Scientific reports*, 8(1):11077, 2018.

[34] Ahmad Khajehnejad, Moein Khajehnejad, Mahmoudreza Babaei, Krishna P Gummadi, Adrian Weller, and Baharan Mirzasoleiman. Crosswalk: Fairness-enhanced node representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11963–11970, 2022.

[35] Hadas Kotek, Rikker Dockum, and David Sun. Gender bias and stereotypes in large language models. In *Proceedings of The ACM Collective Intelligence Conference*, CI '23, page 12–24, New York, NY, USA, 2023. Association for Computing Machinery.

[36] V. Krebs. Political book networks. Unpublished, found at http://www.orgnet.com/.

[37] Anja Lambrecht and Catherine Tucker. Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads. *Management science*, 65(7):2966–2981, 2019.

[38] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: A comparative analysis. *Physical Review E*, 80(5), November 2009.

[39] Andrea Lancichinetti and Santo Fortunato. Limits of modularity maximization in community detection. *Physical Review E*, 84(6), December 2011.

[40] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4), October 2008.

[41] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[42] Huan Li, Ruisheng Zhang, Zhili Zhao, and Xin Liu. Lpa-mni: an improved label propagation algorithm based on modularity and node importance for community detection. *Entropy*, 23(5):497, 2021.

[43] Jiakang Li, Songning Lai, Zhihao Shuai, Yuan Tan, Yifan Jia, Mianyang Yu, Zichen Song, Xiaokang Peng, Ziyang Xu, Yongxin Ni, et al. A comprehensive review of community detection in graphs. *Neurocomputing*, page 128169, 2024.

[44] Queenie Luo, Michael J. Puett, and Michael D. Smith. A "perspectival" mirror of the elephant: Investigating language bias on google, chatgpt, youtube, and wikipedia. *Queue*, 22(1):23–47, March 2024.

[45] Deepanshu Malhotra, Ralucca Gera, and Akrati Saxena. Community detection using semilocal topological features and label propagation algorithm. In *Computational Data and Social Networks: 10th International Conference, CSoNet 2021, Virtual Event, November 15–17, 2021, Proceedings 10*, pages 255–266. Springer, 2021.

[46] Konstantinos Manolis and Evaggelia Pitoura. Modularity-based fairness in community detection. In *Proceedings of the 2023 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '23, page 126–130, New York, NY, USA, 2024. Association for Computing Machinery.

[47] John McLevey, Peter J Carrington, and John Scott. The sage handbook of social network analysis. 2023.

[48] Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(Volume 27, 2001):415–444, 2001.

[49] Ninareh Mehrabi, Fred Morstatter, Nanyun Peng, and Aram Galstyan. Debiasing community detection: the importance of lowly connected nodes. In *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*, pages 509–512, 2019.

[50] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021.

[51] Marina Meilă. Comparing clusterings—an information based distance. *Journal of Multivariate Analysis*, 98(5):873–895, 2007.

[52] Raj Rao Nadakuditi and M. E. J. Newman. Graph spectra and the detectability of community structure in networks. *Physical Review Letters*, 108(18), May 2012.

[53] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3), September 2006.

[54] M. E. J. Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, June 2006.

[55] M. E. J. Newman, George T. Cantwell, and Jean-Gabriel Young. Improved mutual information measure for clustering, classification, and community detection. *Physical Review E*, 101(4), April 2020.

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

[57] M. E. J. Newman and E. A. Leicht. Mixture models and exploratory analysis in networks. *Proceedings of the National Academy of Sciences*, 104(23):9564–9569, June 2007.

[58] Rens Oostenbach. *Fairness-Aware Analysis of Community Detection.* Master's thesis, Eindhoven University of Technology, 2023.

[59] Ferran Parés, Dario Garcia Gasulla, Armand Vilalta, Jonatan Moreno, Eduard Ayguadé, Jesús Labarta, Ulises Cortés, and Toyotaro Suzumura. Fluid communities: A competitive, scalable and diverse community detection algorithm. In *Complex Networks & Their Applications VI: Proceedings of Complex Networks 2017 (The Sixth International Conference on Complex Networks and Their Applications)*, pages 229–240. Springer, 2018.

[60] Tiago P. Peixoto. Efficient monte carlo and greedy heuristic for the inference of stochastic block models. *Physical Review E*, 89(1), January 2014.

[61] Tiago P. Peixoto. Hierarchical block structures and high-resolution model selection in large networks. *Physical Review X*, 4(1), March 2014.

[62] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '14. ACM, August 2014.

[63] Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Comput. Surv.*, 55(3), feb 2022.

[64] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. In *Computer and Information Sciences-ISCIS 2005: 20th International Symposium, Istanbul, Turkey, October 26-28, 2005. Proceedings 20*, pages 284–293. Springer, 2005.

[65] Filippo Radicchi. Detectability of communities in heterogeneous networks. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 88(1):010801, 2013.

[66] Tahleen Rahman, Bartlomiej Surma, Michael Backes, and Yang Zhang. Fairwalk: towards fair graph embedding. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, page 3289–3295. AAAI Press, 2019.

[67] Raimundo Real and Juan M. Vargas. The Probabilistic Basis of Jaccard's Index of Similarity. *Systematic Biology*, 45(3):380–385, 09 1996.

[68] Jörg Reichardt and Stefan Bornholdt. Statistical mechanics of community detection. *Physical Review E*, 74(1), July 2006.

[69] Giulio Rossetti. Rdyn: Graph benchmark handling community dynamics. *Journal of Complex Networks*, 5:893–912, 12 2017.

[70] Giulio Rossetti, Letizia Milli, and Rémy Cazabet. CDlib: a python library to extract, compare and evaluate communities from complex networks. *Applied Network Science Journal*, 2019.

[71] Giulio Rossetti, Luca Pappalardo, and Salvatore Rinzivillo. A novel approach to evaluate community detection algorithms on ground truth. In *Complex Networks VII: Proceedings of the 7th Workshop on Complex Networks CompleNet 2016*, pages 133–144. Springer, 2016.

[72] Martin Rosvall and Carl T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, January 2008.

[73] Pratha Sah, Lisa O Singh, Aaron Clauset, and Shweta Bansal. Exploring community structure in biological networks with random graphs. *BMC bioinformatics*, 15:1–14, 2014.

[74] Akrati Saxena, Cristina Bierbooms, and Mykola Pechenizkiy. Fairness-aware fake news mitigation using counter information propagation. *Applied Intelligence*, 53:1–22, 09 2023.

[75] Akrati Saxena, George Fletcher, and Mykola Pechenizkiy. Hm-eiict: Fairness-aware link prediction in complex networks using community information. *Journal of Combinatorial Optimization*, 44(4):2853–2870, 2022.

[76] Akrati Saxena, George Fletcher, and Mykola Pechenizkiy. Nodesim: node similarity based network embedding for diverse link prediction. *EPJ Data Science*, 11(1):24, 2022.

[77] Akrati Saxena, George Fletcher, and Mykola Pechenizkiy. Fairsna: Algorithmic fairness in social network analysis. *Association for Computing Machinery*, 56(8), April 2024.

[78] Akrati Saxena, Yulong Pei, Jan Veldsink, Werner van Ipenburg, George Fletcher, and Mykola Pechenizkiy. The banking transactions dataset and its comparative analysis with scale-free networks. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 283–296, 2021.

[79] Akrati Saxena, Pratishtha Saxena, Harita Reddy, and Ralucca Gera. A survey on studying the social networks of students. *arXiv preprint arXiv:1909.05079*, 2019.

[80] Stanislav Sobolevsky, Riccardo Campari, Alexander Belyi, and Carlo Ratti. General optimization technique for high-quality community detection in complex networks. *Physical Review E*, 90(1), July 2014.

[81] Marina Sokolova and Guy Lapalme. A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4):427–437, 2009.

[82] Clara Stegehuis, Remco van der Hofstad, and Johan S. H. van Leeuwaarden. Power-law relations in random networks with communities. *Physical Review E*, 94(1), July 2016.

[83] Ana-Andreea Stoica and Augustin Chaintreau. Fairness in social influence maximization. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, page 569–574, New York, NY, USA, 2019. Association for Computing Machinery.

[84] Ana-Andreea Stoica, Jessy Xinyi Han, and Augustin Chaintreau. Seeding network influence in biased networks and the benefits of diversity. In *Proceedings of The Web Conference 2020*, WWW '20, page 2089–2098, New York, NY, USA, 2020. Association for Computing Machinery.

[85] Ana-Andreea Stoica, Nelly Litvak, and Augustin Chaintreau. Fairness in link analysis ranking algorithms. In *The Web Conference 2024*, 2024.

[86] Lei Tang and Huan Liu. *Community detection and mining in social media.* Springer Nature, 2022.

[87] V. A. Traag, G. Krings, and P. Van Dooren. Significant scales in community structure. *Scientific Reports*, 3(1), October 2013.

[88] V. A. Traag, L. Waltman, and N. J. van Eck. From louvain to leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1), March 2019.

[89] Vincent A. Traag and Lovro Šubelj. Large network community detection by fast label propagation. *Scientific Reports*, 13(1), February 2023.

[90] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. Group-fairness in influence maximization. *CoRR*, abs/1903.00967, 2019.

[91] Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Panayiotis Tsaparas, Ilias Kleftakis, and Nikos Mamoulis. Fairness-aware link analysis. *CoRR*, abs/2005.14431, 2020.

[92] Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Panayiotis Tsaparas, Ilias Kleftakis, and Nikos Mamoulis. Fairness-aware pagerank. In *Proceedings of the Web Conference 2021*, WWW '21, page 3815–3826, New York, NY, USA, 2021. Association for Computing Machinery.

[93] Sahil Verma and Julia Rubin. Fairness definitions explained. In *Proceedings of the International Workshop on Software Fairness*, FairWare '18, page 1–7, New York, NY, USA, 2018. Association for Computing Machinery.

[94] Stanley Wasserman and Katherine Faust. Social network analysis: Methods and applications. 1994.

[95] Zhao Yang, René Algesheimer, and Claudio J. Tessone. A comparative analysis of community detection algorithms on artificial networks. *Scientific Reports*, 6(1), August 2016.

[96] Hao Yin, Austin R. Benson, Jure Leskovec, and David F. Gleich. Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 555–564, New York, NY, USA, 2017. Association for Computing Machinery.

[97] Yilin Zhang and Karl Rohe. Understanding regularized spectral clustering via graph conductance. *Advances in Neural Information Processing Systems*, 31, 2018.

[98] Karen Zhou and Chenhao Tan. Entity-based evaluation of political bias in automatic summarization. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10374–10386, Singapore, December 2023. Association for Computational Linguistics.

Appendix A

Proofs

A.1 FCCN equals FCCE+ in a complete graph

Claim: In a complete graph, FCCE+ equals FCCN.

Given a complete, undirected graph $G(V, E)$ with $n = |V|$ nodes and $\frac{n(n-1)}{2}$ edges, let m be the number of misclassified nodes in a mapped predicted community p_j. The number of false bridges (FBs) is given by $m(n-m)$ and the number of correctly classified edges is given by $\frac{1}{2}(n-m)(n-m-1)$.

FCCE+ (see Eq. 4.8) can be described as:

$$
\begin{aligned}
FCCE{+} &= \frac{|FB| + 2|\mathit{classified\ edges}|}{2|\mathit{edges\ in}\ c_i|} && \text{(A.1)}\\
&= \frac{m(n-m) + 2 \cdot \frac{1}{2}(n-m)(n-m-1)}{2 \cdot \frac{1}{2}n(n-1)} && \text{(A.2)}\\
&= \frac{nm - m^2 + n^2 - nm - n - nm + m^2 + m}{n(n-1)} && \text{(A.3)}\\
&= \frac{n^2 - n - nm + m}{n(n-1)} && \text{(A.4)}\\
&= \frac{(n-1)(n-m)}{n(n-1)} && \text{(A.5)}\\
&= \frac{n-m}{n} && \text{(A.6)}
\end{aligned}
$$

Which is the definition of FCCN.

Appendix B

Community Detection Method Parameters

CDMs	CDlib Alg.	Parameters
CNM	greedy_modularity	-
Combo	pycombo	- weight: “weight” - max_coms: None - modularity_resolution: 1 - num_split_attempts: 0 - start_separate: False - treat_as_modularity: False - random_seed: 42
Leiden	leiden	- initial_membership: None - weights: None
Louvain	louvain	- weight: “weight” - resolution: 1 - randomize: None
Paris	paris	-
RB-C	rb_pots	- initial_membership: None - weights: None - resolution_parameter: 1
RB-ER	rber_pots	- initial_membership: None - weights: None - node_sizes: None - resolution_parameter: 1
Significance	significance_coms	- initial_membership: None - node_sizes: None

Table B.1: CDM settings used in experiments. Settings are in bold if they deviate from the CDlib defaults or have to be set manually.

CDMs	CDlib Alg.	Parameters
Eigenvector	eigenvector	-
RSC-K	r_spectral_clustering	- **n_clusters**: Number of ground-truth coms - **method**: “regularized_with_kmeans” - percentile: None
RSC-SSE	r_spectral_clustering	- **n_clusters**: Number of ground-truth coms - **method**: “sklearn_spectral_embedding” - percentile: None
RSC-V	r_spectral_clustering	- **n_clusters**: Number of ground-truth coms - **method**: “vanilla” - percentile: None
Spectral	spectral	- **kmax**: Number of ground-truth coms - projection_on_smaller_class: True - scaler: None
Deepwalk	Own implementation	- **n_clusters**: Number of ground-truth coms - **dimensions**: 128 - **walk_length**: 80 - **num_walks**: 10
Fairwalk	GitHub implementation	- **n_clusters**: Number of ground-truth coms - **dimensions**: 128 - **walk_length**: 80 - **num_walks**: 10
Node2Vec	GitHub implementation	- **n_clusters**: Number of ground-truth coms - **dimensions**: 128 - **walk_length**: 80 - **num_walks**: 10
Infomap	infomap	- flags: “ ”
Spinglass	spinglass	- spins: 25
Walktrap	walktrap	-
Fluid	async_fluid	- **k**: Number of ground-truth coms
Label Propagation	label_propagation	-
EM	em	- **k**: Number of ground-truth coms
SBM	sbm_dl	-
SBM - Nested	sbm_dl_nested	-

Table B.2: CDM settings used in experiments. Settings are in bold if they deviate from the CDlib defaults or have to be set manually.

Appendix C

Full Results

The results are presented in tabular form here. Appendices C.1 to C.24 give the NMI and fairness metrics Φ_p^{CPM} from the predictions produced by the 24 CD methods. Appendices C.25 to C.32 give their performance measures. Cells in the tables are colored according to how good the value in the cell is. The greener the color, the better the value. For fairness metrics Φ_p^{CPM}, the ideal value is 0. A cell's color becomes more white the further away it is from 0, regardless of sign. For the partition quality measures NMI, RMI, ARI, PF1, and NF1, the value of 1 is ideal. The closer to 1 a cell's value is, the greener it is colored. VI is different, and in this case, a value of 0 is ideal.

C.1 Results: LFR Size mu 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.586	0.376	0.168	0.373	0.377	0.089	0.200	0.012	0.008
Combo	0.912	0.657	0.634	0.657	0.657	0.489	0.489	**0.001**	**0.001**
Leiden	0.917	0.652	0.632	0.652	0.652	0.477	0.476	0.001	0.001
Louvain	0.915	0.653	0.630	0.653	0.653	0.475	0.475	**0.000**	**0.000**
Paris	0.713	-0.043	**0.011**	-0.043	**-0.010**	-0.209	-0.045	-0.107	-0.085
RB-C	0.918	0.653	0.629	0.653	0.653	0.467	0.466	0.001	0.001
RB-ER	0.943	0.590	0.559	0.590	0.590	0.312	0.310	**0.001**	**0.001**
Significance	**1.000**	**0.002**	**0.001**	**0.001**	**0.001**	**0.017**	**0.003**	0.003	0.003
Eigenvector	0.323	0.316	0.161	0.273	0.319	0.061	0.133	0.262	0.247
RSC-K	**0.990**	**0.031**	**0.021**	**0.019**	**0.016**	0.121	**0.031**	0.022	0.019
RSC-SSE	0.976	**0.039**	0.056	-0.110	-0.024	0.087	0.031	-0.195	-0.113
RSC-V	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.002**	**0.000**	**0.000**	**0.000**
Spectral	0.120	0.097	0.035	0.096	0.097	0.045	0.049	0.138	0.097
Deepwalk	0.984	0.151	0.190	0.088	0.134	**-0.003**	0.087	-0.116	-0.074
Fairwalk	0.982	0.158	0.205	0.076	0.133	0.021	0.094	-0.129	-0.080
Node2Vec	0.982	0.149	0.182	0.089	0.133	**-0.005**	0.105	-0.103	-0.066
Infomap	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.002**	**0.000**	**0.000**	**0.000**
Spinglass	0.727	0.501	0.214	0.501	0.501	0.220	0.210	0.030	0.023
Walktrap	**0.995**	0.083	0.091	0.083	0.083	0.071	0.065	0.001	0.001
Fluid	0.982	0.082	0.144	0.037	0.086	0.024	**0.026**	-0.153	-0.107
Label Propagation	0.982	0.197	0.153	0.197	0.196	0.071	0.066	0.004	0.003
EM	0.292	-0.089	-0.139	**0.010**	-0.076	-0.042	-0.049	0.535	0.472
SBM	0.973	0.504	0.564	0.493	0.499	0.352	0.351	-0.029	-0.018
SBM - Nested	0.983	-0.300	-0.198	-0.545	-0.416	-0.293	-0.267	-0.525	-0.391

Table C.1: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.2$. The 5 best (unrounded) scores are in bold.

C.2 Results: LFR Size mu 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.352	0.172	**0.030**	0.169	0.176	**0.026**	**0.070**	0.173	0.144
Combo	0.830	0.634	0.551	0.634	0.634	0.480	0.470	0.122	0.104
Leiden	0.862	0.642	0.561	0.642	0.642	0.457	0.433	0.087	0.075
Louvain	0.869	0.641	0.559	0.641	0.641	0.446	0.424	0.060	0.053
Paris	0.539	0.383	0.299	0.357	0.464	**-0.017**	0.094	-0.105	-0.082
RB-C	0.863	0.640	0.559	0.640	0.640	0.456	0.431	0.084	0.073
RB-ER	0.872	0.576	0.457	0.576	0.576	0.342	0.295	0.074	0.064
Significance	**0.985**	**0.078**	0.048	**0.051**	**0.033**	0.134	0.100	0.110	0.096
Eigenvector	0.197	0.181	**0.037**	0.144	0.187	**0.016**	**0.049**	0.171	0.179
RSC-K	0.933	0.146	0.134	0.108	0.097	0.030	0.169	0.086	0.072
RSC-SSE	0.791	0.185	0.197	0.144	0.159	0.121	0.241	**-0.029**	**-0.036**
RSC-V	**0.990**	**0.021**	**0.032**	**0.007**	**0.005**	**-0.012**	**0.047**	**0.025**	**0.023**
Spectral	0.096	**0.091**	**0.024**	**0.091**	**0.092**	0.038	**0.046**	0.186	0.133
Deepwalk	0.942	0.214	0.290	0.119	0.192	0.230	0.206	-0.063	**-0.016**
Fairwalk	0.936	0.257	0.314	0.176	0.231	0.271	0.250	**0.024**	0.050
Node2Vec	0.943	0.225	0.300	0.131	0.199	0.227	0.214	-0.059	**-0.014**
Infomap	**0.989**	**0.071**	0.076	**0.056**	**0.053**	0.198	0.075	0.045	0.041
Spinglass	0.703	0.473	0.195	0.473	0.473	0.233	0.217	0.144	0.122
Walktrap	0.943	0.324	0.324	0.319	0.314	0.311	0.312	0.063	0.052
Fluid	0.953	0.145	0.205	0.113	0.132	0.088	0.160	**0.031**	0.044
Label Propagation	0.880	0.426	0.327	0.423	0.415	0.367	0.298	0.141	0.117
EM	0.191	**0.014**	**-0.034**	**0.031**	**0.019**	**-0.015**	**-0.001**	0.236	0.203
SBM	**0.955**	0.504	0.560	0.473	0.490	0.425	0.387	**-0.005**	**0.016**
SBM - Nested	**0.954**	-0.221	-0.054	-0.574	-0.414	-0.109	-0.192	-0.578	-0.431

Table C.2: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.4$. The 5 best (unrounded) scores are in bold.

C.3 Results: LFR Size mu 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.138	0.084	**-0.000**	0.084	**0.089**	**0.008**	**0.020**	0.200	0.167
Combo	0.356	0.269	0.068	0.276	0.286	0.025	0.099	0.296	0.264
Leiden	0.611	0.513	0.307	0.525	0.531	0.120	0.271	0.310	0.286
Louvain	0.623	0.509	0.304	0.521	0.526	0.110	0.266	0.292	0.268
Paris	0.366	0.087	0.095	**0.058**	0.156	**-0.008**	**0.012**	-0.059	-0.067
RB-C	0.620	0.526	0.322	0.539	0.544	0.115	0.281	0.312	0.289
RB-ER	0.624	0.514	0.304	0.527	0.533	0.090	0.256	0.304	0.282
Significance	**0.858**	0.169	0.176	0.175	0.157	-0.135	0.151	0.170	0.168
Eigenvector	0.096	0.083	**0.011**	0.063	0.091	0.010	**0.018**	0.046	0.036
RSC-K	0.739	0.213	0.289	0.155	0.241	-0.044	0.173	**0.013**	0.036
RSC-SSE	0.411	**0.051**	0.135	**0.022**	**0.090**	**-0.002**	0.033	**-0.010**	**-0.028**
RSC-V	**0.838**	0.196	0.316	0.214	0.206	-0.101	0.271	0.180	0.179
Spectral	0.047	**0.062**	**0.012**	**0.061**	**0.064**	0.016	0.022	0.023	**0.019**
Deepwalk	**0.769**	0.285	0.353	0.281	0.302	-0.009	0.284	0.214	0.233
Fairwalk	0.751	0.286	0.354	0.286	0.307	-0.023	0.284	0.221	0.236
Node2Vec	**0.773**	0.263	0.333	0.255	0.279	**0.001**	0.270	0.197	0.215
Infomap	**0.850**	0.274	0.281	0.276	0.258	**-0.005**	0.270	0.272	0.255
Spinglass	0.510	0.411	0.158	0.423	0.430	0.065	0.179	0.303	0.277
Walktrap	0.674	**0.076**	0.112	0.094	0.158	-0.073	0.041	**0.012**	0.032
Fluid	0.706	0.165	0.290	0.160	0.190	-0.068	0.180	0.078	0.090
Label Propagation	0.000	**0.038**	**0.001**	**0.038**	**0.038**	0.010	**0.010**	**-0.000**	**-0.000**
EM	0.174	**0.018**	**-0.025**	**0.031**	**0.026**	-0.011	**0.002**	0.038	0.034
SBM	0.657	0.528	0.511	0.540	0.550	0.236	0.421	0.024	**0.011**
SBM - Nested	0.677	0.480	0.481	0.493	0.504	0.201	0.393	**0.022**	**0.012**

Table C.3: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.6$. The 5 best (unrounded) scores are in bold.

C.4 Results: LFR Density mu 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.586	-0.036	-0.020	-0.036	-0.037	-0.007	-0.040	0.031	0.025
Combo	0.912	-0.398	-0.286	-0.398	-0.398	-0.150	-0.154	**0.002**	**0.001**
Leiden	0.917	-0.410	-0.295	-0.410	-0.410	-0.143	-0.148	0.002	0.002
Louvain	0.915	-0.423	-0.306	-0.423	-0.423	-0.155	-0.157	**0.001**	**0.000**
Paris	0.713	0.072	**0.017**	0.071	0.037	0.165	0.073	0.121	0.098
RB-C	0.918	-0.416	-0.301	-0.416	-0.416	-0.133	-0.139	0.003	0.003
RB-ER	0.943	-0.414	-0.378	-0.414	-0.414	-0.182	-0.189	0.002	0.002
Significance	**1.000**	**0.004**	**0.002**	**0.002**	**0.001**	0.043	**0.006**	0.006	0.005
Eigenvector	0.323	**-0.008**	-0.023	**0.002**	**-0.009**	**-0.002**	-0.016	0.039	0.043
RSC-K	**0.990**	0.053	0.044	0.047	0.036	0.242	0.062	0.055	0.042
RSC-SSE	0.976	0.159	0.169	0.151	0.151	0.250	0.163	0.008	0.008
RSC-V	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.004**	**0.000**	**0.000**	**0.000**
Spectral	0.120	**-0.010**	**0.005**	**-0.010**	**-0.010**	**0.002**	**0.001**	0.077	0.056
Deepwalk	0.984	-0.069	-0.069	-0.055	-0.065	-0.006	-0.028	0.032	0.022
Fairwalk	0.982	-0.069	-0.070	-0.039	-0.060	0.032	-0.019	0.057	0.037
Node2Vec	0.982	-0.060	-0.049	-0.040	-0.056	0.016	**-0.007**	0.048	0.032
Infomap	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.004**	**0.000**	**0.000**	**0.000**
Spinglass	0.727	-0.060	-0.018	-0.060	-0.060	-0.043	-0.060	0.035	0.028
Walktrap	**0.995**	0.042	0.063	0.041	0.041	0.060	0.052	**0.001**	**0.001**
Fluid	0.982	-0.093	-0.124	-0.076	-0.098	0.051	-0.025	0.082	0.060
Label Propagation	0.982	-0.188	-0.167	-0.189	-0.190	-0.063	-0.078	0.006	0.005
EM	0.292	-0.041	-0.043	-0.011	-0.036	**0.000**	-0.025	0.203	0.202
SBM	0.973	-0.455	-0.431	-0.454	-0.454	-0.155	-0.161	0.005	0.003
SBM - Nested	0.983	0.059	0.037	0.148	0.097	0.068	0.054	0.140	0.089

Table C.4: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.2$. The 5 best (unrounded) scores are in bold.

C.5 Results: LFR Density mu 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.352	**-0.058**	**-0.007**	-0.057	-0.060	**0.000**	**-0.001**	0.256	0.224
Combo	0.830	-0.228	-0.166	-0.229	-0.229	0.064	**-0.024**	0.230	0.200
Leiden	0.862	-0.223	-0.175	-0.224	-0.224	0.055	-0.059	0.173	0.150
Louvain	0.869	-0.226	-0.172	-0.226	-0.227	0.048	-0.054	0.130	0.114
Paris	0.539	0.059	**0.013**	0.059	**0.029**	0.042	0.105	0.358	0.340
RB-C	0.863	-0.221	-0.172	-0.221	-0.221	0.069	-0.049	0.168	0.147
RB-ER	0.872	-0.273	-0.216	-0.276	-0.276	0.050	-0.096	0.146	0.127
Significance	**0.985**	0.163	0.103	0.107	0.071	0.531	0.208	0.224	0.197
Eigenvector	0.197	**-0.035**	**-0.009**	**-0.026**	-0.037	**-0.001**	**-0.004**	**0.071**	**0.094**
RSC-K	0.933	0.308	0.282	0.283	0.237	0.416	0.358	0.254	0.200
RSC-SSE	0.791	0.343	0.362	0.318	0.328	0.203	0.338	**0.026**	**0.031**
RSC-V	**0.990**	0.068	0.056	**0.028**	**0.018**	0.450	0.109	**0.088**	**0.078**
Spectral	0.096	**-0.021**	**0.003**	**-0.021**	**-0.021**	**0.004**	**0.005**	0.183	0.135
Deepwalk	0.942	0.370	0.370	0.378	0.351	0.506	0.382	0.365	0.308
Fairwalk	0.936	0.370	0.367	0.373	0.353	0.490	0.380	0.366	0.312
Node2Vec	0.943	0.357	0.361	0.364	0.340	0.508	0.372	0.345	0.293
Infomap	**0.989**	0.135	0.139	0.108	0.101	0.376	0.143	**0.099**	**0.089**
Spinglass	0.703	-0.195	-0.072	-0.195	-0.195	**0.017**	-0.029	0.263	0.228
Walktrap	0.943	0.477	0.491	0.472	0.468	0.549	0.479	**0.085**	**0.072**
Fluid	0.953	0.233	0.149	0.238	0.177	0.531	0.269	0.367	0.318
Label Propagation	0.880	0.174	0.188	0.165	0.147	0.382	0.282	0.220	0.188
EM	0.191	**-0.053**	-0.054	**-0.013**	**-0.036**	**-0.015**	**-0.022**	0.492	0.445
SBM	**0.955**	**0.006**	**0.013**	**0.022**	**0.006**	0.335	0.170	0.183	0.152
SBM - Nested	**0.954**	0.341	0.316	0.441	0.372	0.510	0.313	0.384	0.298

Table C.5: Average NMI and average $\Phi_{density}^{CPM}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.4$. The 5 best (unrounded) scores are in bold.

C.6 Results: LFR Density mu 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.138	**-0.019**	**-0.003**	**-0.018**	**-0.022**	**-0.002**	**0.002**	0.345	0.310
Combo	0.356	-0.042	**-0.005**	-0.044	-0.048	0.049	0.052	0.541	0.505
Leiden	0.611	-0.096	-0.061	-0.107	-0.113	0.182	0.126	0.567	0.538
Louvain	0.623	-0.076	-0.047	-0.086	-0.090	0.201	0.136	0.565	0.535
Paris	0.366	0.300	0.157	0.275	0.333	0.021	0.123	0.287	0.343
RB-C	0.620	-0.101	-0.060	-0.114	-0.120	0.183	0.119	0.570	0.542
RB-ER	0.624	-0.127	-0.077	-0.140	-0.146	0.173	0.103	0.572	0.545
Significance	**0.858**	0.527	0.471	0.477	0.389	0.516	0.560	0.576	0.553
Eigenvector	0.096	**-0.009**	**-0.002**	**-0.007**	**-0.011**	**-0.001**	**0.002**	**0.123**	**0.140**
RSC-K	0.739	0.514	0.545	0.474	0.478	0.366	0.536	0.447	0.415
RSC-SSE	0.411	0.120	0.154	0.082	0.125	0.023	0.090	**0.044**	**0.071**
RSC-V	**0.838**	0.533	0.423	0.497	0.433	0.596	0.509	0.578	0.551
Spectral	0.047	**-0.001**	**0.010**	**-0.001**	**-0.001**	**0.009**	**0.011**	0.261	0.191
Deepwalk	**0.769**	0.541	0.529	0.533	0.501	0.506	0.550	0.557	0.541
Fairwalk	0.751	0.540	0.514	0.532	0.501	0.476	0.535	0.553	0.538
Node2Vec	**0.773**	0.544	0.533	0.539	0.506	0.500	0.554	0.564	0.546
Infomap	**0.850**	0.517	0.483	0.476	0.433	0.610	0.539	0.551	0.526
Spinglass	0.510	-0.104	-0.037	-0.111	-0.116	0.090	0.067	0.565	0.533
Walktrap	0.674	0.558	0.557	0.558	0.515	0.314	0.569	0.536	0.505
Fluid	0.706	0.539	0.438	0.533	0.485	0.412	0.496	0.583	0.571
Label Propagation	0.000	**-0.012**	**-0.000**	**-0.012**	**-0.012**	**-0.003**	**-0.003**	**-0.000**	**-0.000**
EM	0.174	**-0.039**	-0.051	**-0.006**	**-0.021**	**-0.017**	**-0.016**	0.500	0.444
SBM	0.657	0.245	0.180	0.236	0.228	0.407	0.414	**0.154**	**0.117**
SBM - Nested	0.677	0.453	0.400	0.448	0.443	0.492	0.512	**0.162**	**0.125**

Table C.6: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.6$. The 5 best (unrounded) scores are in bold.

C.7 Results: LFR Conductance mu 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.586	-0.323	-0.157	-0.320	-0.325	-0.086	-0.200	-0.013	-0.010
Combo	0.912	-0.672	-0.638	-0.672	-0.672	-0.500	-0.497	**-0.002**	**-0.002**
Leiden	0.917	-0.663	-0.635	-0.663	-0.663	-0.488	-0.485	-0.003	-0.003
Louvain	0.915	-0.664	-0.631	-0.664	-0.664	-0.481	-0.480	**-0.001**	**-0.000**
Paris	0.713	**0.021**	**-0.021**	**0.022**	**-0.002**	0.183	**0.021**	0.071	0.055
RB-C	0.918	-0.666	-0.632	-0.666	-0.666	-0.479	-0.475	-0.005	-0.004
RB-ER	0.943	-0.609	-0.570	-0.609	-0.609	-0.322	-0.315	**-0.002**	**-0.002**
Significance	**1.000**	**-0.006**	**-0.003**	**-0.003**	**-0.002**	**-0.054**	**-0.009**	-0.009	-0.008
Eigenvector	0.323	-0.307	-0.171	-0.259	-0.310	-0.061	-0.142	-0.267	-0.261
RSC-K	**0.990**	-0.072	-0.056	-0.052	**-0.043**	-0.248	-0.076	-0.058	-0.048
RSC-SSE	0.976	-0.211	-0.225	-0.084	-0.153	-0.314	-0.204	0.151	0.082
RSC-V	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.004**	**-0.001**	**-0.000**	**-0.000**
Spectral	0.120	**-0.068**	**-0.028**	-0.068	-0.068	**-0.039**	-0.042	-0.194	-0.140
Deepwalk	0.984	-0.224	-0.262	-0.173	-0.211	-0.083	-0.146	0.087	0.054
Fairwalk	0.982	-0.234	-0.276	-0.171	-0.213	-0.135	-0.156	0.081	0.045
Node2Vec	0.982	-0.239	-0.272	-0.194	-0.227	-0.118	-0.192	0.061	0.036
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.004**	**-0.001**	**-0.000**	**-0.000**
Spinglass	0.727	-0.437	-0.172	-0.437	-0.437	-0.217	-0.195	-0.069	-0.054
Walktrap	**0.995**	-0.188	-0.213	-0.187	-0.187	-0.170	-0.159	-0.002	-0.002
Fluid	0.982	-0.093	-0.143	-0.053	-0.095	-0.069	**-0.033**	0.126	0.087
Label Propagation	0.982	-0.148	-0.099	-0.148	-0.146	-0.062	-0.042	-0.009	-0.008
EM	0.292	0.136	0.174	**0.015**	0.127	**0.047**	0.073	-0.605	-0.555
SBM	0.973	-0.520	-0.591	-0.514	-0.517	-0.406	-0.402	0.019	0.011
SBM - Nested	0.983	0.253	0.163	0.486	0.360	0.240	0.223	0.468	0.337

Table C.7: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.2$. The 5 best (unrounded) scores are in bold.

C.8 Results: LFR Conductance mu 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.352	**-0.097**	**-0.018**	**-0.095**	**-0.097**	**-0.025**	**-0.066**	-0.337	-0.294
Combo	0.830	-0.576	-0.478	-0.576	-0.576	-0.499	-0.459	-0.290	-0.253
Leiden	0.862	-0.595	-0.494	-0.595	-0.595	-0.477	-0.406	-0.220	-0.192
Louvain	0.869	-0.594	-0.495	-0.594	-0.594	-0.465	-0.399	-0.163	**-0.144**
Paris	0.539	-0.411	-0.295	-0.388	-0.470	**-0.009**	-0.166	**-0.154**	-0.165
RB-C	0.863	-0.591	-0.492	-0.591	-0.591	-0.481	-0.407	-0.213	-0.186
RB-ER	0.872	-0.506	-0.368	-0.505	-0.505	-0.384	-0.259	-0.187	-0.163
Significance	**0.985**	-0.200	-0.127	-0.134	**-0.089**	-0.498	-0.251	-0.272	-0.240
Eigenvector	0.197	-0.136	**-0.027**	**-0.109**	-0.141	**-0.015**	**-0.043**	-0.204	-0.225
RSC-K	0.933	-0.357	-0.335	-0.309	-0.272	-0.327	-0.405	-0.252	-0.204
RSC-SSE	0.791	-0.433	-0.452	-0.398	-0.411	-0.272	-0.462	**-0.001**	**0.006**
RSC-V	**0.990**	**-0.075**	**-0.077**	**-0.030**	**-0.020**	-0.361	-0.132	**-0.094**	**-0.085**
Spectral	0.096	**-0.062**	**-0.024**	**-0.061**	**-0.062**	**-0.038**	**-0.046**	-0.284	-0.211
Deepwalk	0.942	-0.474	-0.517	-0.424	-0.450	-0.548	-0.469	-0.264	-0.249
Fairwalk	0.936	-0.494	-0.525	-0.450	-0.471	-0.555	-0.490	-0.326	-0.299
Node2Vec	0.943	-0.477	-0.521	-0.426	-0.453	-0.549	-0.468	-0.248	-0.238
Infomap	**0.989**	-0.177	-0.187	-0.144	-0.136	-0.437	-0.187	**-0.125**	**-0.112**
Spinglass	0.703	-0.329	-0.118	-0.329	-0.329	-0.237	-0.196	-0.331	-0.289
Walktrap	0.943	-0.581	-0.588	-0.577	-0.573	-0.596	-0.573	**-0.116**	**-0.097**
Fluid	0.953	-0.326	-0.326	-0.308	-0.280	-0.485	-0.359	-0.328	-0.297
Label Propagation	0.880	-0.504	-0.433	-0.497	-0.482	-0.545	-0.458	-0.294	-0.251
EM	0.191	**0.029**	**0.070**	**-0.015**	**0.013**	**0.024**	**0.016**	-0.503	-0.456
SBM	**0.955**	-0.571	-0.612	-0.559	-0.563	-0.581	-0.497	-0.161	-0.149
SBM - Nested	**0.954**	**-0.077**	-0.220	0.343	0.139	-0.348	**-0.079**	0.407	0.242

Table C.8: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.4$. The 5 best (unrounded) scores are in bold.

C.9 Results: LFR Conductance mu 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.138	**-0.042**	**0.002**	**-0.042**	**-0.043**	**-0.004**	**-0.017**	-0.428	-0.382
Combo	0.356	-0.160	**-0.045**	-0.162	-0.166	-0.062	-0.121	-0.600	-0.564
Leiden	0.611	-0.367	-0.188	-0.370	-0.372	-0.253	-0.310	-0.626	-0.598
Louvain	0.623	-0.375	-0.198	-0.378	-0.380	-0.263	-0.316	-0.620	-0.591
Paris	0.366	-0.334	-0.211	-0.293	-0.408	**-0.012**	-0.116	-0.205	-0.259
RB-C	0.620	-0.378	-0.199	-0.382	-0.383	-0.250	-0.310	-0.627	-0.600
RB-ER	0.624	-0.346	-0.165	-0.350	-0.351	-0.223	-0.279	-0.629	-0.603
Significance	**0.858**	-0.554	-0.514	-0.518	-0.440	-0.395	-0.574	-0.594	-0.575
Eigenvector	0.096	**-0.054**	**-0.007**	**-0.041**	**-0.058**	**-0.006**	**-0.015**	**-0.147**	**-0.154**
RSC-K	0.739	-0.556	-0.602	-0.498	-0.536	-0.290	-0.558	-0.412	-0.396
RSC-SSE	0.411	-0.161	-0.254	-0.104	-0.200	-0.021	-0.114	**-0.037**	**-0.052**
RSC-V	**0.838**	-0.573	-0.541	-0.552	-0.499	-0.522	-0.583	-0.605	-0.583
Spectral	0.047	**-0.047**	**-0.020**	**-0.046**	**-0.048**	**-0.020**	**-0.027**	-0.254	-0.187
Deepwalk	**0.769**	-0.599	-0.611	-0.593	-0.578	-0.449	-0.604	-0.589	-0.583
Fairwalk	0.751	-0.599	-0.602	-0.592	-0.579	-0.412	-0.594	-0.587	-0.581
Node2Vec	**0.773**	-0.597	-0.609	-0.592	-0.576	-0.453	-0.605	-0.593	-0.584
Infomap	**0.850**	-0.579	-0.557	-0.552	-0.515	-0.566	-0.593	-0.604	-0.580
Spinglass	0.510	-0.237	-0.081	-0.240	-0.242	-0.131	-0.192	-0.618	-0.588
Walktrap	0.674	-0.545	-0.559	-0.553	-0.540	-0.214	-0.541	-0.494	-0.473
Fluid	0.706	-0.567	-0.544	-0.563	-0.537	-0.320	-0.540	-0.574	-0.568
Label Propagation	0.000	**-0.006**	**-0.000**	**-0.006**	**-0.006**	**-0.004**	**-0.004**	**0.000**	**0.000**
EM	0.174	**0.024**	0.063	**-0.015**	**0.001**	0.022	**0.012**	-0.468	-0.415
SBM	0.657	-0.576	-0.538	-0.580	-0.583	-0.493	-0.583	**-0.152**	**-0.111**
SBM - Nested	0.677	-0.633	-0.614	-0.637	-0.640	-0.537	-0.625	**-0.155**	**-0.115**

Table C.9: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\mu = 0.6$. The 5 best (unrounded) scores are in bold.

C.10 Results: ABCD Size xi 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.608	0.180	-0.008	0.177	0.179	-0.044	-0.055	0.043	0.030
Combo	0.910	0.654	0.612	0.654	0.654	0.448	0.448	**-0.000**	**-0.000**
Leiden	0.915	0.652	0.613	0.652	0.652	0.438	0.438	**-0.000**	**-0.000**
Louvain	0.915	0.648	0.610	0.648	0.648	0.431	0.431	**-0.000**	**-0.000**
Paris	0.743	-0.068	-0.022	-0.048	**-0.011**	-0.375	-0.113	-0.152	-0.125
RB-C	0.916	0.649	0.610	0.649	0.649	0.431	0.431	**-0.000**	**-0.000**
RB-ER	0.891	0.165	0.087	0.165	0.165	-0.202	-0.200	-0.001	-0.001
Significance	**0.999**	**-0.004**	**-0.002**	**-0.002**	**-0.001**	-0.123	-0.006	-0.006	-0.005
Eigenvector	0.286	0.266	0.091	0.214	0.262	**0.028**	0.086	0.184	0.171
RSC-K	0.962	-0.175	-0.121	-0.132	-0.074	-0.426	-0.245	-0.256	-0.204
RSC-SSE	**1.000**	**0.001**	**-0.000**	**0.001**	**0.001**	**0.012**	**0.001**	0.003	0.002
RSC-V	**0.998**	-0.051	-0.070	-0.051	-0.051	-0.041	-0.037	-0.000	-0.000
Spectral	0.006	0.038	**-0.006**	0.038	0.038	**0.005**	**0.005**	**-0.000**	**-0.000**
Deepwalk	0.975	0.063	0.136	-0.118	-0.023	-0.039	0.025	-0.273	-0.195
Fairwalk	0.968	0.080	0.175	-0.129	-0.018	-0.039	0.039	-0.306	-0.224
Node2Vec	0.974	0.066	0.150	-0.125	-0.026	-0.036	0.028	-0.298	-0.217
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Spinglass	0.735	0.516	0.239	0.516	0.516	0.211	0.237	0.013	0.009
Walktrap	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.001**	**0.000**	0.000	0.000
Fluid	0.975	0.089	0.181	0.049	0.113	-0.199	**0.000**	-0.222	-0.169
Label Propagation	0.978	**0.020**	-0.078	**0.024**	0.020	-0.070	-0.070	0.016	0.012
EM	0.545	-0.183	-0.103	-0.110	-0.217	-0.054	-0.100	0.019	-0.013
SBM	0.964	0.570	0.615	0.570	0.570	0.431	0.431	**-0.000**	**-0.000**
SBM - Nested	0.995	-0.058	-0.024	-0.166	-0.107	-0.113	-0.069	-0.185	-0.131

Table C.10: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.2$. The 5 best (unrounded) scores are in bold.

C.11 Results: ABCD Size xi 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.461	0.157	**0.001**	0.154	0.157	**-0.004**	**-0.005**	0.068	0.048
Combo	0.878	0.649	0.572	0.649	0.649	0.437	0.437	0.001	0.000
Leiden	0.887	0.655	0.581	0.655	0.655	0.431	0.430	**0.000**	**0.000**
Louvain	0.890	0.654	0.582	0.654	0.654	0.424	0.424	**-0.000**	**-0.000**
Paris	0.820	-0.163	-0.063	-0.122	-0.050	-0.256	-0.211	-0.265	-0.236
RB-C	0.889	0.653	0.581	0.653	0.653	0.427	0.427	**0.000**	**0.000**
RB-ER	0.881	0.443	0.319	0.443	0.443	0.085	0.086	-0.000	-0.000
Significance	**0.998**	**-0.013**	**-0.007**	**-0.007**	**-0.004**	-0.225	-0.019	-0.020	-0.017
Eigenvector	0.255	0.226	0.065	0.178	0.229	0.018	0.062	0.180	0.175
RSC-K	0.937	-0.189	-0.130	-0.164	-0.091	-0.383	-0.272	-0.300	-0.237
RSC-SSE	0.957	**-0.018**	-0.013	**0.005**	**0.005**	-0.187	-0.034	-0.025	-0.025
RSC-V	**0.998**	**-0.018**	-0.023	**-0.017**	**-0.016**	-0.104	**-0.016**	-0.004	-0.003
Spectral	0.103	0.065	**-0.005**	0.064	0.066	**0.002**	**0.007**	-0.100	-0.071
Deepwalk	0.974	0.125	0.215	-0.052	0.042	**0.001**	0.056	-0.268	-0.193
Fairwalk	0.963	0.145	0.220	-0.024	0.070	0.020	0.092	-0.211	-0.146
Node2Vec	0.974	0.109	0.196	-0.072	0.026	**-0.002**	0.044	-0.274	-0.196
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Spinglass	0.734	0.500	0.229	0.501	0.501	0.205	0.234	0.008	0.005
Walktrap	**0.998**	**-0.003**	**-0.003**	**-0.002**	**-0.001**	-0.131	**-0.006**	-0.005	-0.004
Fluid	0.966	0.054	0.166	0.029	0.093	-0.353	-0.025	-0.236	-0.186
Label Propagation	0.906	0.071	-0.078	0.075	0.071	-0.084	-0.083	0.021	0.016
EM	0.434	-0.115	-0.065	-0.030	-0.134	-0.029	-0.045	0.033	0.020
SBM	0.969	0.538	0.591	0.538	0.538	0.399	0.399	**-0.000**	**-0.000**
SBM - Nested	**0.995**	0.056	0.088	0.021	0.043	0.019	0.029	-0.061	-0.039

Table C.11: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.4$. The 5 best (unrounded) scores are in bold.

C.12 Results: ABCD Size xi 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.290	0.109	-0.004	0.110	0.113	**0.002**	**0.016**	0.166	0.132
Combo	0.826	0.607	0.465	0.607	0.607	0.329	0.376	0.005	0.004
Leiden	0.851	0.618	0.490	0.618	0.618	0.367	0.370	0.003	0.002
Louvain	0.853	0.620	0.499	0.620	0.620	0.378	0.380	**0.000**	**0.000**
Paris	0.588	**0.024**	0.088	0.028	0.127	-0.059	-0.087	-0.243	-0.247
RB-C	0.853	0.622	0.505	0.622	0.622	0.376	0.388	**0.002**	**0.002**
RB-ER	0.849	0.572	0.438	0.572	0.572	0.278	0.292	0.003	0.002
Significance	**0.978**	-0.104	-0.056	-0.071	-0.042	-0.528	-0.145	-0.161	-0.137
Eigenvector	0.144	0.123	0.021	0.094	0.131	**0.012**	0.028	0.067	0.048
RSC-K	0.858	-0.167	-0.100	-0.148	-0.069	-0.162	-0.247	-0.294	-0.231
RSC-SSE	0.757	-0.086	**0.003**	-0.047	**-0.006**	-0.113	-0.098	-0.146	-0.140
RSC-V	**0.984**	**-0.048**	**0.003**	-0.033	**-0.017**	-0.482	-0.043	-0.079	-0.064
Spectral	0.092	0.073	**0.003**	0.073	0.075	**0.012**	**0.018**	-0.032	-0.027
Deepwalk	0.957	0.109	0.193	0.020	0.089	-0.052	0.082	-0.144	-0.095
Fairwalk	0.937	0.137	0.227	0.026	0.103	-0.072	0.117	-0.122	-0.072
Node2Vec	0.960	0.107	0.193	**0.012**	0.082	-0.058	0.071	-0.163	-0.109
Infomap	**0.999**	**0.002**	**0.002**	**0.002**	**0.002**	**-0.008**	**0.001**	**0.000**	**0.000**
Spinglass	0.733	0.488	0.224	0.488	0.489	0.183	0.227	0.013	0.010
Walktrap	0.872	-0.102	-0.079	-0.065	**-0.021**	-0.402	-0.183	-0.195	-0.165
Fluid	0.901	**0.005**	0.193	**0.009**	0.085	-0.391	-0.046	-0.264	-0.225
Label Propagation	0.000	**0.042**	**0.001**	0.042	**0.042**	**0.012**	**0.012**	**0.000**	**0.000**
EM	0.356	-0.055	-0.028	**0.001**	-0.051	-0.017	**-0.018**	0.016	0.004
SBM	**0.964**	0.548	0.592	0.548	0.548	0.381	0.402	**-0.000**	**-0.000**
SBM-Nested	**0.982**	0.121	0.210	**0.019**	0.094	0.016	0.058	-0.206	-0.137

Table C.12: Average NMI and average Φ^{CPM}_{size} scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.6$. The 5 best (unrounded) scores are in bold.

C.13 Results: ABCD Density xi 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.608	-0.137	-0.017	-0.134	-0.137	0.039	0.042	-0.039	-0.027
Combo	0.910	-0.642	-0.574	-0.642	-0.642	-0.409	-0.409	**-0.000**	**-0.000**
Leiden	0.915	-0.646	-0.581	-0.646	-0.646	-0.400	-0.400	**-0.000**	**-0.000**
Louvain	0.915	-0.640	-0.576	-0.640	-0.640	-0.390	-0.390	**-0.000**	**-0.000**
Paris	0.743	0.050	**0.007**	0.040	**0.005**	0.365	0.090	0.134	0.108
RB-C	0.916	-0.647	-0.581	-0.647	-0.647	-0.395	-0.395	**-0.000**	**-0.000**
RB-ER	0.891	-0.390	-0.298	-0.390	-0.390	0.035	0.033	0.001	0.001
Significance	**0.999**	**0.002**	**0.001**	**0.001**	**0.001**	0.069	**0.003**	0.003	0.003
Eigenvector	0.286	-0.173	-0.061	-0.133	-0.170	**-0.019**	-0.059	-0.105	-0.094
RSC-K	0.962	0.124	0.088	0.097	0.058	0.365	0.171	0.176	0.139
RSC-SSE	**1.000**	**-0.001**	**-0.000**	**-0.001**	**-0.001**	**-0.011**	**-0.001**	-0.003	-0.002
RSC-V	**0.998**	0.036	0.048	0.036	0.036	0.027	0.025	0.000	0.000
Spectral	0.006	**-0.014**	0.009	**-0.014**	-0.014	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Deepwalk	0.975	-0.180	-0.219	-0.062	-0.126	-0.060	-0.104	0.171	0.117
Fairwalk	0.968	-0.195	-0.253	-0.061	-0.135	-0.074	-0.125	0.178	0.123
Node2Vec	0.974	-0.169	-0.222	-0.044	-0.113	-0.055	-0.101	0.199	0.140
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Spinglass	0.735	-0.382	-0.155	-0.383	-0.383	-0.172	-0.195	-0.012	-0.008
Walktrap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.001**	**-0.000**	-0.000	-0.000
Fluid	0.975	-0.183	-0.249	-0.155	-0.200	0.104	-0.051	0.168	0.127
Label Propagation	0.978	-0.027	0.052	-0.030	-0.027	0.050	0.050	-0.010	-0.008
EM	0.545	0.170	0.077	0.112	0.199	0.052	0.088	0.022	0.061
SBM	0.964	-0.618	-0.639	-0.618	-0.618	-0.440	-0.440	**-0.000**	**-0.000**
SBM - Nested	0.995	0.031	0.009	0.110	0.064	0.073	0.042	0.130	0.087

Table C.13: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.2$. The 5 best (unrounded) scores are in bold.

C.14 Results: ABCD Density xi 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.461	-0.106	**-0.005**	-0.103	-0.107	**0.004**	**0.004**	-0.057	-0.039
Combo	0.878	-0.613	-0.512	-0.613	-0.613	-0.392	-0.392	-0.001	-0.001
Leiden	0.887	-0.629	-0.532	-0.629	-0.629	-0.389	-0.389	**-0.000**	**-0.000**
Louvain	0.890	-0.629	-0.533	-0.629	-0.629	-0.383	-0.383	**-0.000**	**-0.000**
Paris	0.820	0.133	0.039	0.106	0.037	0.252	0.183	0.247	0.218
RB-C	0.889	-0.627	-0.531	-0.627	-0.627	-0.385	-0.385	**-0.000**	**-0.000**
RB-ER	0.881	-0.484	-0.364	-0.484	-0.484	-0.131	-0.132	0.000	0.000
Significance	**0.998**	**0.009**	**0.004**	**0.005**	**0.003**	0.158	**0.013**	0.013	0.011
Eigenvector	0.255	-0.160	-0.044	-0.122	-0.161	**-0.013**	-0.049	-0.122	-0.116
RSC-K	0.937	0.149	0.102	0.132	0.074	0.362	0.215	0.240	0.188
RSC-SSE	0.957	0.021	0.017	**0.002**	**0.001**	0.189	0.037	0.027	0.026
RSC-V	**0.998**	**0.019**	0.022	**0.018**	0.018	0.075	0.013	0.003	0.002
Spectral	0.103	-0.037	**0.004**	-0.036	-0.037	**-0.002**	**-0.006**	0.106	0.076
Deepwalk	0.974	-0.200	-0.262	-0.074	-0.143	-0.057	-0.099	0.198	0.138
Fairwalk	0.963	-0.205	-0.252	-0.086	-0.156	-0.075	-0.128	0.135	0.089
Node2Vec	0.974	-0.178	-0.237	-0.053	-0.124	-0.051	-0.085	0.201	0.141
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Spinglass	0.734	-0.396	-0.162	-0.397	-0.397	-0.169	-0.198	-0.004	-0.002
Walktrap	**0.998**	**0.003**	**0.003**	**0.002**	**0.001**	0.101	**0.005**	0.004	0.004
Fluid	0.966	-0.120	-0.209	-0.099	-0.149	0.280	-0.013	0.194	0.152
Label Propagation	0.906	**-0.016**	0.101	-0.018	**-0.016**	0.100	0.096	-0.014	-0.010
EM	0.434	0.115	0.060	0.035	0.137	0.028	0.043	-0.020	0.002
SBM	0.969	-0.585	-0.613	-0.585	-0.585	-0.400	-0.400	**-0.000**	**-0.000**
SBM - Nested	**0.995**	-0.063	-0.097	-0.024	-0.049	**-0.021**	-0.032	0.067	0.042

Table C.14: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.4$. The 5 best (unrounded) scores are in bold.

C.15 Results: ABCD Density xi 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.290	-0.061	**0.010**	-0.064	-0.064	**0.000**	**-0.010**	-0.119	-0.090
Combo	0.826	-0.561	-0.402	-0.562	-0.562	-0.300	-0.344	-0.003	-0.002
Leiden	0.851	-0.583	-0.438	-0.583	-0.583	-0.342	-0.346	-0.002	-0.002
Louvain	0.853	-0.587	-0.448	-0.587	-0.587	-0.351	-0.354	**0.000**	**0.000**
Paris	0.588	**-0.023**	-0.087	**-0.023**	-0.117	0.059	0.081	0.252	0.254
RB-C	0.853	-0.591	-0.454	-0.591	-0.591	-0.348	-0.358	**-0.002**	**-0.002**
RB-ER	0.849	-0.560	-0.421	-0.560	-0.560	-0.290	-0.301	-0.003	-0.002
Significance	**0.978**	0.088	0.047	0.060	0.036	0.498	0.123	0.138	0.117
Eigenvector	0.144	-0.093	-0.017	-0.070	-0.099	**-0.011**	-0.025	-0.044	-0.025
RSC-K	0.858	0.156	0.102	0.141	0.072	0.160	0.231	0.267	0.209
RSC-SSE	0.757	0.078	-0.015	0.041	**-0.003**	0.118	0.096	0.150	0.145
RSC-V	**0.984**	**0.038**	**-0.010**	**0.027**	**0.014**	0.419	0.025	0.062	0.050
Spectral	0.092	**-0.046**	**-0.002**	-0.046	-0.047	**-0.011**	**-0.016**	0.054	0.043
Deepwalk	0.957	-0.168	-0.230	-0.099	-0.153	0.012	-0.115	0.091	0.053
Fairwalk	0.937	-0.205	-0.269	-0.125	-0.182	0.027	-0.161	0.042	0.010
Node2Vec	0.960	-0.165	-0.225	-0.090	-0.147	0.038	-0.096	0.119	0.075
Infomap	**0.999**	**-0.002**	**-0.001**	**-0.002**	**-0.002**	**0.008**	**0.000**	**0.000**	**0.000**
Spinglass	0.733	-0.406	-0.170	-0.406	-0.407	-0.164	-0.205	-0.010	-0.007
Walktrap	0.872	0.079	0.058	0.050	**0.003**	0.415	0.168	0.202	0.170
Fluid	0.901	-0.049	-0.236	-0.047	-0.120	0.390	**0.016**	0.266	0.226
Label Propagation	0.000	**-0.026**	**-0.001**	**-0.026**	**-0.026**	**-0.011**	**-0.011**	**0.000**	**0.000**
EM	0.356	0.058	0.026	**0.003**	0.059	0.017	0.018	-0.013	0.003
SBM	**0.964**	-0.607	-0.624	-0.607	-0.607	-0.391	-0.409	**0.001**	**0.001**
SBM - Nested	**0.982**	-0.165	-0.247	-0.057	-0.136	-0.026	-0.064	0.219	0.146

Table C.15: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.6$. The 5 best (unrounded) scores are in bold.

C.16 Results: ABCD Conductance xi 0.2

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.608	-0.183	-0.166	-0.180	-0.183	-0.192	-0.251	0.013	0.010
Combo	0.910	0.475	0.403	0.475	0.475	0.232	0.232	**-0.000**	**-0.000**
Leiden	0.915	0.444	0.385	0.444	0.444	0.214	0.214	**-0.000**	**-0.000**
Louvain	0.915	0.428	0.370	0.428	0.428	0.203	0.203	**-0.000**	**-0.000**
Paris	0.743	-0.077	-0.050	-0.067	-0.039	-0.305	-0.102	-0.115	-0.092
RB-C	0.916	0.433	0.371	0.433	0.433	0.204	0.204	**-0.000**	**-0.000**
RB-ER	0.891	-0.274	-0.296	-0.274	-0.274	-0.307	-0.306	-0.001	-0.001
Significance	**0.999**	**-0.001**	**-0.001**	**-0.001**	**-0.000**	-0.038	**-0.002**	-0.002	-0.002
Eigenvector	0.286	0.063	0.020	0.044	0.060	**0.007**	0.024	0.037	0.035
RSC-K	0.962	-0.106	-0.077	-0.083	-0.051	-0.330	-0.147	-0.144	-0.113
RSC-SSE	**1.000**	**0.000**	**-0.000**	**0.000**	**0.000**	**-0.002**	**-0.000**	0.000	0.000
RSC-V	**0.998**	-0.032	-0.040	-0.031	-0.031	-0.021	-0.019	-0.000	-0.000
Spectral	0.006	-0.043	-0.040	-0.043	-0.043	-0.034	-0.034	**-0.000**	**-0.000**
Deepwalk	0.975	0.066	0.110	-0.023	0.027	-0.008	0.040	-0.146	-0.099
Fairwalk	0.968	**0.029**	0.078	-0.062	**-0.009**	-0.029	0.028	-0.135	-0.093
Node2Vec	0.974	0.043	0.087	-0.035	0.010	**0.003**	0.031	-0.144	-0.102
Infomap	**1.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**	**-0.000**
Spinglass	0.735	0.282	0.098	0.283	0.283	0.091	0.111	0.000	-0.000
Walktrap	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.001**	**0.000**	0.000	0.000
Fluid	0.975	0.035	0.078	**0.019**	0.046	-0.092	**0.003**	-0.104	-0.077
Label Propagation	0.978	-0.102	-0.147	-0.095	-0.101	-0.088	-0.089	0.026	0.019
EM	0.545	-0.186	-0.119	-0.125	-0.217	-0.045	-0.098	-0.024	-0.055
SBM	0.964	0.350	0.425	0.350	0.350	0.288	0.288	**-0.000**	**-0.000**
SBM - Nested	0.995	-0.032	**-0.012**	-0.097	-0.058	-0.059	-0.035	-0.109	-0.072

Table C.16: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.2$. The 5 best (unrounded) scores are in bold.

C.17 Results: ABCD Conductance xi 0.4

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.461	-0.192	-0.089	-0.185	-0.194	-0.066	-0.130	0.031	0.021
Combo	0.878	0.162	0.041	0.162	0.162	**-0.004**	**-0.002**	-0.001	-0.001
Leiden	0.887	0.179	0.069	0.179	0.179	0.023	0.023	**-0.000**	**-0.000**
Louvain	0.890	0.199	0.082	0.199	0.199	0.008	0.008	**0.000**	**0.000**
Paris	0.820	-0.097	-0.055	-0.103	-0.055	-0.160	-0.126	-0.169	-0.139
RB-C	0.889	0.196	0.075	0.196	0.196	0.011	0.011	0.000	0.000
RB-ER	0.881	-0.304	-0.276	-0.304	-0.304	-0.257	-0.256	-0.000	-0.000
Significance	**0.998**	**0.002**	**0.001**	**0.001**	**0.001**	0.014	**0.003**	0.003	0.003
Eigenvector	0.255	-0.058	**-0.021**	-0.053	-0.063	**-0.002**	-0.005	-0.005	0.002
RSC-K	0.937	-0.072	-0.052	-0.073	-0.058	-0.230	-0.071	-0.066	-0.050
RSC-SSE	0.957	**-0.008**	**-0.006**	**-0.007**	**0.000**	-0.090	-0.023	-0.029	-0.022
RSC-V	**0.998**	-0.021	-0.029	**-0.022**	-0.022	**0.000**	-0.016	**0.000**	**0.000**
Spectral	0.103	-0.080	-0.042	-0.079	-0.080	-0.046	-0.046	-0.204	-0.143
Deepwalk	0.974	0.067	0.069	0.089	0.081	0.028	0.040	0.019	0.013
Fairwalk	0.963	0.055	0.064	0.078	0.073	0.034	0.050	0.020	0.013
Node2Vec	0.974	0.094	0.084	0.129	0.116	0.051	0.054	0.043	0.030
Infomap	**1.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**
Spinglass	0.734	-0.090	-0.049	-0.090	-0.090	**-0.007**	0.007	-0.011	-0.009
Walktrap	**0.998**	**-0.002**	**-0.002**	**-0.001**	**-0.001**	-0.028	**-0.004**	-0.003	-0.003
Fluid	0.966	0.058	0.064	0.058	0.053	0.019	0.048	0.031	0.025
Label Propagation	0.906	-0.118	-0.106	-0.117	-0.118	-0.115	-0.104	0.003	0.003
EM	0.434	-0.077	-0.054	-0.029	-0.094	-0.013	-0.030	0.005	-0.004
SBM	0.969	0.116	0.207	0.116	0.116	0.170	0.170	**0.000**	**0.000**
SBM - Nested	**0.995**	**0.004**	0.037	-0.042	**-0.013**	-0.008	**0.004**	-0.079	-0.050

Table C.17: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.4$. The 5 best (unrounded) scores are in bold.

C.18 Results: ABCD Conductance xi 0.6

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.290	-0.100	-0.019	-0.094	-0.101	-0.011	-0.031	-0.070	-0.059
Combo	0.826	-0.420	-0.298	-0.421	-0.421	-0.184	-0.211	-0.007	-0.005
Leiden	0.851	-0.392	-0.300	-0.392	-0.392	-0.202	-0.198	**-0.003**	-0.003
Louvain	0.853	-0.384	-0.298	-0.384	-0.384	-0.206	-0.207	**-0.000**	**-0.000**
Paris	0.588	-0.100	-0.089	-0.095	-0.127	**0.002**	-0.025	-0.004	0.008
RB-C	0.853	-0.403	-0.317	-0.403	-0.403	-0.206	-0.207	-0.004	-0.003
RB-ER	0.849	-0.447	-0.341	-0.447	-0.447	-0.277	-0.279	-0.007	-0.006
Significance	**0.978**	0.044	0.024	0.031	0.018	0.250	0.062	0.069	0.059
Eigenvector	0.144	-0.069	-0.012	-0.053	-0.073	**-0.006**	-0.013	-0.011	**0.001**
RSC-K	0.858	**-0.002**	-0.016	**0.000**	-0.017	-0.028	0.014	0.049	0.038
RSC-SSE	0.757	0.024	**-0.006**	0.029	**0.006**	0.047	0.037	0.067	0.062
RSC-V	**0.984**	0.015	-0.008	**0.009**	**0.001**	0.251	0.017	0.034	0.028
Spectral	0.092	-0.042	**-0.002**	-0.041	-0.042	-0.008	**-0.011**	-0.230	-0.158
Deepwalk	0.957	-0.046	-0.082	**0.000**	-0.033	-0.006	-0.032	0.068	0.040
Fairwalk	0.937	**0.001**	-0.059	0.063	0.028	-0.025	-0.033	0.084	0.056
Node2Vec	0.960	**0.003**	-0.038	0.049	**0.016**	0.014	**-0.005**	0.088	0.062
Infomap	**0.999**	**0.002**	**0.001**	**0.002**	**0.002**	**-0.000**	**-0.001**	**-0.001**	**-0.001**
Spinglass	0.733	-0.308	-0.128	-0.309	-0.309	-0.099	-0.116	-0.017	-0.014
Walktrap	0.872	-0.304	-0.336	-0.321	-0.337	-0.094	-0.248	-0.010	-0.004
Fluid	0.901	0.077	-0.020	0.086	0.042	0.214	0.077	0.178	0.150
Label Propagation	0.000	-0.020	**-0.000**	-0.020	-0.020	**-0.006**	**-0.006**	**-0.000**	**-0.000**
EM	0.356	**0.003**	**-0.003**	**-0.005**	**-0.009**	**0.006**	**0.002**	0.004	0.021
SBM	**0.964**	-0.209	-0.229	-0.209	-0.209	-0.106	-0.101	**-0.002**	**-0.002**
SBM - Nested	**0.982**	0.050	0.057	0.068	0.054	0.044	0.043	0.042	0.027

Table C.18: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 5 LFR networks with $n = 10,000$, $\xi = 0.6$. The 5 best (unrounded) scores are in bold.

C.19 Results: HICH-BA Multiple Majorities Size

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.890	0.281	0.295	0.277	0.279	-0.251	0.222	**-0.001**	**0.001**
Combo	**0.961**	0.107	0.134	0.101	0.101	-0.337	0.103	0.007	0.008
Leiden	**0.956**	0.103	0.117	0.100	0.100	-0.379	0.081	**0.003**	**0.003**
Louvain	**0.956**	0.103	0.117	0.100	0.101	-0.373	0.082	**0.003**	**0.003**
Paris	0.681	0.248	0.250	0.257	0.277	**-0.031**	0.194	-0.113	-0.091
RB-C	**0.956**	0.103	0.117	0.100	0.100	-0.379	**0.081**	**0.003**	**0.003**
RB-ER	0.294	-0.444	-0.443	-0.382	-0.377	-0.339	-0.430	-0.364	-0.356
Significance	0.302	-0.299	-0.334	-0.297	-0.259	-0.071	-0.269	-0.274	-0.279
Eigenvector	0.672	0.326	0.267	0.349	0.322	0.069	0.338	0.272	0.200
RSC-K	0.222	**-0.007**	-0.170	**-0.008**	**-0.010**	**0.054**	0.096	0.059	0.057
RSC-SSE	0.580	0.151	0.125	0.152	0.110	**0.036**	0.183	0.139	0.103
RSC-V	0.508	-0.292	-0.345	-0.293	-0.292	-0.398	-0.183	-0.010	-0.007
Spectral	0.132	0.125	**0.011**	0.119	0.125	0.146	0.165	-0.036	-0.028
Deepwalk	0.517	0.085	0.240	-0.088	**-0.027**	0.124	0.196	-0.218	-0.229
Fairwalk	0.513	0.104	0.246	**-0.021**	0.034	0.091	0.197	-0.167	-0.186
Node2Vec	0.519	**0.042**	0.212	**-0.030**	**0.007**	0.096	0.192	-0.184	-0.202
Infomap	0.669	-0.113	**-0.062**	**-0.045**	**-0.028**	-0.387	-0.159	-0.147	-0.134
Spinglass	**0.923**	0.045	0.088	0.049	0.050	-0.421	**0.064**	**-0.006**	**-0.006**
Walktrap	0.789	**0.018**	**0.015**	**0.023**	**0.024**	-0.231	**0.005**	-0.010	-0.009
Fluid	0.616	0.072	0.337	0.186	0.214	**-0.029**	0.210	-0.219	-0.197
Label Propagation	0.635	-0.189	-0.251	-0.192	-0.206	-0.282	-0.121	0.077	0.062
EM	0.044	0.141	0.141	0.200	0.204	**0.061**	0.173	-0.050	-0.054
SBM	0.875	**-0.008**	**0.017**	-0.399	-0.284	-0.387	**-0.027**	-0.413	-0.307
SBM - Nested	0.880	**-0.025**	**-0.009**	-0.379	-0.275	-0.344	**-0.032**	-0.373	-0.274

Table C.19: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple majority communities. The 5 best (unrounded) scores are in bold.

C.20 Results: HICH-BA Multiple Majorities Density

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.890	**-0.037**	**-0.022**	**-0.025**	**-0.026**	0.285	**-0.017**	**-0.012**	**-0.013**
Combo	**0.961**	-0.216	-0.256	-0.204	-0.204	0.122	-0.197	-0.018	-0.018
Leiden	**0.956**	-0.213	-0.222	-0.203	-0.203	0.197	-0.156	-0.014	-0.014
Louvain	**0.956**	-0.213	-0.222	-0.203	-0.203	0.207	-0.156	-0.014	**-0.014**
Paris	0.681	-0.447	-0.448	-0.454	-0.468	**-0.033**	-0.401	0.119	0.096
RB-C	**0.956**	-0.213	-0.222	-0.203	-0.203	0.197	-0.156	**-0.014**	**-0.014**
RB-ER	0.294	0.352	0.340	0.298	0.300	0.342	0.322	0.266	0.265
Significance	0.302	0.382	0.380	0.310	0.250	0.146	0.392	0.384	0.381
Eigenvector	0.672	-0.346	-0.302	-0.363	-0.344	-0.072	-0.364	-0.245	-0.179
RSC-K	0.222	0.171	0.271	**0.099**	**0.065**	**0.033**	0.090	0.102	0.069
RSC-SSE	0.580	-0.090	**-0.041**	-0.123	-0.094	**0.034**	-0.085	-0.064	-0.042
RSC-V	0.508	0.139	0.169	0.142	0.141	0.242	**0.036**	**0.005**	**0.003**
Spectral	0.132	**-0.018**	0.053	**-0.014**	**-0.018**	**-0.033**	-0.070	0.037	0.028
Deepwalk	0.517	-0.327	-0.401	-0.292	-0.348	-0.203	-0.306	0.108	0.165
Fairwalk	0.513	-0.331	-0.389	-0.302	-0.358	-0.184	-0.306	**0.013**	0.098
Node2Vec	0.519	-0.293	-0.420	-0.358	-0.389	-0.203	-0.320	0.062	0.137
Infomap	0.669	0.073	**0.038**	**0.033**	**0.020**	0.428	0.102	0.099	0.089
Spinglass	**0.923**	-0.151	-0.202	-0.143	-0.144	0.216	-0.159	**-0.008**	**-0.008**
Walktrap	0.789	**0.000**	**-0.005**	**-0.014**	**-0.018**	0.437	**0.020**	0.032	0.029
Fluid	0.616	-0.267	-0.440	-0.335	-0.353	-0.105	-0.362	0.166	0.148
Label Propagation	0.635	0.110	0.176	0.104	0.133	0.366	0.040	-0.120	-0.093
EM	0.044	-0.119	-0.138	-0.144	-0.151	**-0.057**	-0.150	0.129	0.126
SBM	0.875	**-0.033**	-0.053	0.329	0.209	0.302	**-0.005**	0.368	0.257
SBM - Nested	0.880	**0.020**	**0.006**	0.394	0.276	0.327	**0.022**	0.384	0.268

Table C.20: Average NMI and average $\Phi^{CPM}_{density}$ scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple majority communities. The 5 best (unrounded) scores are in bold.

C.21 Results: HICH-BA Multiple Majorities Conductance

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.890	**0.008**	**-0.012**	**0.022**	**0.022**	-0.142	**-0.007**	**-0.023**	**-0.022**
Combo	**0.961**	-0.092	-0.081	-0.063	-0.062	-0.260	-0.066	-0.044	-0.043
Leiden	**0.956**	-0.079	-0.085	-0.059	-0.059	-0.251	-0.054	-0.027	-0.027
Louvain	**0.956**	-0.079	-0.085	-0.059	-0.059	-0.237	**-0.054**	-0.027	-0.027
Paris	0.681	-0.186	-0.184	-0.185	-0.183	-0.103	-0.194	**-0.002**	**-0.001**
RB-C	**0.956**	-0.079	-0.086	-0.059	-0.059	-0.252	-0.054	-0.027	-0.027
RB-ER	0.294	-0.231	-0.197	-0.237	-0.205	-0.233	-0.251	-0.254	-0.228
Significance	0.302	0.059	**0.019**	**-0.009**	**-0.039**	0.047	0.097	0.093	0.081
Eigenvector	0.672	-0.117	-0.135	-0.099	-0.127	-0.066	-0.116	**-0.009**	**0.005**
RSC-K	0.222	0.108	0.064	0.070	0.042	0.063	0.135	0.112	0.085
RSC-SSE	0.580	-0.046	**-0.013**	-0.075	-0.105	**0.000**	**0.046**	0.146	0.114
RSC-V	0.508	-0.344	-0.356	-0.341	-0.341	-0.321	-0.308	**-0.006**	**-0.005**
Spectral	0.132	-0.303	-0.321	-0.306	-0.303	-0.283	-0.296	**-0.027**	**-0.020**
Deepwalk	0.517	-0.128	-0.044	-0.169	-0.170	**-0.015**	**-0.044**	-0.149	-0.099
Fairwalk	0.513	**-0.026**	-0.070	-0.096	-0.093	-0.056	-0.063	-0.099	-0.074
Node2Vec	0.519	-0.210	-0.126	-0.297	-0.298	-0.055	-0.072	-0.202	-0.137
Infomap	0.669	-0.068	-0.036	**-0.016**	**-0.011**	-0.092	-0.093	-0.079	-0.076
Spinglass	**0.923**	-0.039	-0.066	**-0.016**	**-0.016**	-0.276	-0.106	-0.030	-0.030
Walktrap	0.789	-0.087	-0.096	-0.091	-0.096	**0.028**	-0.081	0.035	0.030
Fluid	0.616	**-0.028**	-0.040	**-0.001**	**0.006**	**-0.039**	-0.069	-0.060	-0.055
Label Propagation	0.635	-0.206	-0.179	-0.212	-0.173	-0.129	-0.227	-0.182	-0.153
EM	0.044	**0.025**	**0.012**	0.044	0.042	**0.007**	**0.029**	0.057	0.050
SBM	0.875	-0.092	-0.102	-0.153	-0.133	-0.266	-0.100	-0.109	-0.093
SBM - Nested	0.880	**-0.031**	**-0.030**	-0.056	-0.047	-0.182	-0.058	-0.075	-0.069

Table C.21: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple majority communities. The 5 best (unrounded) scores are in bold.

C.22 Results: HICH-BA Multiple Minorities Size

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.204	-0.088	0.175	**-0.047**	**0.074**	-0.155	**0.010**	-0.450	-0.410
Combo	0.388	-0.371	-0.268	-0.399	-0.315	-0.409	-0.333	-0.475	-0.433
Leiden	0.275	-0.422	-0.226	-0.429	-0.347	-0.353	-0.318	-0.481	-0.452
Louvain	0.272	-0.411	-0.154	-0.440	-0.366	-0.285	-0.247	-0.482	-0.451
Paris	**0.694**	0.328	0.342	0.324	0.326	**-0.027**	0.307	**-0.002**	**-0.000**
RB-C	0.281	-0.432	-0.233	-0.459	-0.394	-0.338	-0.305	-0.484	-0.455
RB-ER	0.149	-0.311	-0.208	**-0.020**	**0.022**	-0.366	-0.369	-0.344	-0.342
Significance	0.134	-0.294	-0.332	-0.228	-0.147	-0.039	-0.260	-0.269	-0.282
Eigenvector	0.147	0.199	0.328	0.155	0.229	0.165	0.238	-0.362	-0.314
RSC-K	0.462	0.327	0.223	0.183	0.102	0.372	0.382	0.375	0.312
RSC-SSE	0.546	0.225	0.165	0.142	0.081	0.239	0.295	0.274	0.224
RSC-V	**0.848**	0.113	0.155	0.116	0.118	-0.353	0.129	**0.008**	**0.004**
Spectral	0.144	0.472	0.454	0.472	0.472	0.360	0.446	**-0.000**	**-0.000**
Deepwalk	0.200	0.162	0.252	-0.196	-0.193	**0.025**	0.162	-0.249	-0.277
Fairwalk	0.244	0.151	0.240	-0.232	-0.220	**0.015**	0.140	-0.289	-0.288
Node2Vec	0.209	0.149	0.240	-0.220	-0.218	**0.014**	0.145	-0.235	-0.260
Infomap	0.285	-0.225	-0.137	**-0.102**	**-0.063**	-0.427	-0.296	-0.280	-0.259
Spinglass	0.260	-0.324	**-0.031**	-0.311	-0.206	-0.242	-0.179	-0.472	-0.440
Walktrap	0.224	**-0.067**	**-0.044**	**0.031**	**0.020**	-0.053	-0.096	**-0.048**	**-0.058**
Fluid	0.214	**-0.045**	0.241	0.167	0.230	**-0.004**	**0.033**	-0.387	-0.365
Label Propagation	**0.801**	**0.041**	**0.034**	**0.057**	**0.045**	-0.290	**0.044**	**0.034**	**0.023**
EM	0.024	0.295	0.376	0.474	0.483	0.081	0.261	-0.084	-0.087
SBM	**0.896**	**-0.011**	**0.000**	-0.375	-0.238	-0.440	**-0.010**	-0.377	-0.242
SBM - Nested	**0.874**	**-0.010**	**-0.001**	-0.360	-0.239	-0.409	**-0.011**	-0.360	-0.242

Table C.22: Average NMI and average Φ^{CPM}_{size} scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple minority communities with one large majority community. The 5 best (unrounded) scores are in bold.

C.23 Results: HICH-BA Multiple Minorities Density

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.204	-0.316	-0.286	-0.330	-0.373	-0.119	-0.228	0.253	0.224
Combo	0.388	**-0.055**	-0.147	**-0.037**	-0.085	0.087	**-0.015**	0.267	0.237
Leiden	0.275	0.231	-0.135	0.234	0.182	**0.028**	-0.085	0.273	0.252
Louvain	0.272	0.183	-0.252	0.202	0.152	**-0.061**	-0.199	0.275	0.251
Paris	**0.694**	-0.485	-0.456	-0.485	-0.488	-0.141	-0.440	**0.023**	**0.017**
RB-C	0.281	0.237	-0.179	0.255	0.211	**-0.017**	-0.139	0.275	0.253
RB-ER	0.149	0.292	0.220	0.125	0.129	0.232	0.300	0.223	0.238
Significance	0.134	0.307	0.296	0.181	0.112	**0.085**	0.321	0.303	0.306
Eigenvector	0.147	-0.308	-0.260	-0.283	-0.324	-0.128	-0.221	0.139	0.131
RSC-K	0.462	**0.009**	**0.059**	**0.022**	**0.019**	-0.123	**-0.028**	**-0.040**	**-0.043**
RSC-SSE	0.546	-0.064	**0.016**	**-0.018**	**-0.008**	-0.087	**-0.061**	-0.089	-0.080
RSC-V	**0.848**	-0.437	-0.443	-0.439	-0.440	-0.106	-0.376	-0.095	-0.068
Spectral	0.144	-0.296	-0.283	-0.296	-0.296	-0.246	-0.292	**0.000**	**0.000**
Deepwalk	0.200	-0.346	-0.432	-0.319	-0.305	-0.154	-0.294	**-0.023**	0.063
Fairwalk	0.244	-0.363	-0.419	-0.337	-0.342	-0.110	-0.242	0.102	0.134
Node2Vec	0.209	-0.324	-0.400	-0.281	-0.266	-0.149	-0.268	**-0.037**	**0.048**
Infomap	0.285	0.119	0.067	**0.051**	**0.032**	0.431	0.158	0.151	0.139
Spinglass	0.260	**-0.053**	-0.329	-0.060	-0.110	-0.149	-0.235	0.266	0.244
Walktrap	0.224	0.139	0.094	**0.005**	**0.001**	0.128	0.196	0.138	0.143
Fluid	0.214	-0.436	-0.331	-0.505	-0.521	-0.104	-0.202	0.203	0.189
Label Propagation	**0.801**	-0.058	**-0.044**	-0.110	**-0.077**	0.277	-0.070	-0.091	**-0.058**
EM	0.024	-0.210	-0.238	-0.266	-0.280	**-0.050**	-0.152	0.139	0.135
SBM	**0.896**	**0.008**	**-0.001**	0.210	0.125	0.320	**0.006**	0.213	0.130
SBM - Nested	**0.874**	**0.014**	**0.003**	0.262	0.165	0.317	**0.016**	0.265	0.170

Table C.23: Average NMI and average Φ_{size}^{CPM} scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple minority communities with one large majority community. The 5 best (unrounded) scores are in bold.

C.24 Results: HICH-BA Multiple Minorities Conductance

CDMs	NMI	FCCN	F1	FCCE	FCCE+	AVGF1	SWF1	SWFCCE	SWFCCE+
CNM	0.204	**-0.097**	-0.303	-0.127	-0.212	**-0.026**	-0.183	0.403	0.364
Combo	0.388	0.225	0.088	0.254	0.168	0.289	0.193	0.424	0.384
Leiden	0.275	0.374	**0.025**	0.380	0.303	0.195	0.119	0.431	0.403
Louvain	0.272	0.371	**0.022**	0.397	0.329	0.197	0.107	0.432	0.402
Paris	**0.694**	-0.481	-0.474	-0.479	-0.481	-0.139	-0.441	**0.008**	**0.006**
RB-C	0.281	0.384	0.123	0.409	0.348	0.254	0.189	0.434	0.405
RB-ER	0.149	0.248	0.151	**-0.038**	**-0.050**	0.176	0.285	0.236	0.257
Significance	0.134	0.322	0.339	0.220	0.140	**0.062**	0.308	0.303	0.313
Eigenvector	0.147	-0.277	-0.305	-0.249	-0.308	-0.138	-0.225	0.235	0.220
RSC-K	0.462	-0.213	-0.121	-0.110	-0.058	-0.299	-0.265	-0.266	-0.222
RSC-SSE	0.546	**-0.144**	**-0.026**	**-0.070**	**-0.036**	-0.171	-0.143	-0.187	-0.158
RSC-V	**0.848**	-0.384	-0.406	-0.386	-0.388	**0.035**	-0.367	**-0.069**	**-0.049**
Spectral	0.144	-0.545	-0.533	-0.545	-0.545	-0.463	-0.528	**0.000**	**0.000**
Deepwalk	0.200	-0.236	-0.314	**0.010**	**0.042**	-0.069	-0.210	**0.060**	0.144
Fairwalk	0.244	-0.199	-0.275	**0.057**	0.064	**-0.036**	-0.146	0.222	0.243
Node2Vec	0.209	-0.181	-0.296	0.089	0.119	-0.070	-0.196	**0.075**	0.152
Infomap	0.285	0.197	0.116	0.088	**0.055**	0.420	0.258	0.246	0.227
Spinglass	0.260	0.221	-0.132	0.208	0.111	0.114	**0.014**	0.422	0.392
Walktrap	0.224	0.144	0.099	**0.014**	**0.007**	0.094	0.193	0.144	0.148
Fluid	0.214	-0.165	-0.274	-0.328	-0.371	**-0.031**	**-0.089**	0.345	0.324
Label Propagation	**0.801**	**-0.070**	-0.051	-0.115	-0.079	0.273	**-0.085**	-0.108	**-0.073**
EM	0.024	-0.286	-0.342	-0.423	-0.436	-0.073	-0.232	0.140	**0.135**
SBM	**0.896**	**0.010**	**-0.002**	0.328	0.205	0.413	**0.008**	0.330	0.209
SBM - Nested	**0.874**	**0.007**	**-0.005**	0.303	0.200	0.343	**0.002**	0.302	0.202

Table C.24: Average NMI and average $\Phi^{CPM}_{conductance}$ scores for the 24 CD methods. CD methods were used on 10 HICH-BA networks with $n = 10,000$, containing multiple minority communities with one large majority community. The 5 best (unrounded) scores are in bold.

C.25 Results: Performance LFR mu 0.2

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.586	0.846	4.819	0.068	0.524	0.041
Combo	0.912	0.988	1.289	0.594	0.737	0.299
Leiden	0.917	0.989	1.229	0.614	0.738	0.318
Louvain	0.915	**0.990**	1.252	0.608	0.732	0.305
Paris	0.713	0.543	3.056	0.591	0.718	0.661
RB-C	0.918	**0.991**	1.218	0.621	0.736	0.313
RB-ER	0.943	0.982	0.858	0.729	0.795	0.446
Significance	**1.000**	**1.000**	**0.005**	**0.999**	**0.974**	**0.948**
Eigenvector	0.323	0.133	8.041	0.020	0.262	0.064
RSC-K	**0.990**	0.967	**0.165**	**0.963**	**0.984**	**0.969**
RSC-SSE	0.976	0.916	0.388	0.744	0.950	0.858
RSC-V	**1.000**	**1.000**	**0.001**	**1.000**	**1.000**	**1.000**
Spectral	0.120	0.281	7.782	0.005	0.389	0.004
Deepwalk	0.984	0.944	0.255	0.905	0.905	0.762
Fairwalk	0.982	0.937	0.293	0.903	0.897	0.738
Node2Vec	0.982	0.936	0.290	0.850	0.914	0.768
Infomap	**1.000**	**1.000**	**0.001**	**1.000**	**1.000**	**1.000**
Spinglass	0.727	0.964	3.454	0.179	0.326	0.030
Walktrap	**0.995**	0.986	**0.076**	**0.954**	**0.993**	**0.961**
Fluid	0.982	0.937	0.291	0.911	0.903	0.758
Label Propagation	0.982	0.958	0.281	0.873	0.966	0.839
EM	0.292	-2.072	7.765	0.004	0.125	0.063
SBM	0.973	0.978	0.426	0.880	0.875	0.663
SBM - Nested	0.983	0.944	0.281	0.872	0.900	0.776

Table C.25: Scores of partition quality measures. CD methods were applied to LFR networks with 10,000 nodes and with $\mu = 0.2$.

C.26 Results: Performance LFR mu 0.4

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.352	0.594	6.746	0.025	0.333	0.012
Combo	0.830	0.934	2.368	0.366	0.657	0.173
Leiden	0.862	0.958	1.968	0.467	0.652	0.192
Louvain	0.869	**0.966**	1.863	0.476	0.652	0.201
Paris	0.539	0.438	6.438	0.179	0.383	0.144
RB-C	0.863	**0.958**	1.952	0.471	0.644	0.199
RB-ER	0.872	**0.960**	1.831	0.491	0.625	0.206
Significance	**0.985**	0.007	**0.240**	**0.976**	0.536	0.316
Eigenvector	0.197	0.178	9.146	0.016	0.135	0.010
RSC-K	0.933	0.784	1.047	0.451	**0.916**	**0.845**
RSC-SSE	0.791	0.314	3.047	0.063	0.766	0.614
RSC-V	**0.990**	**0.972**	**0.163**	**0.982**	**0.982**	**0.981**
Spectral	0.096	0.187	8.215	0.004	0.336	0.004
Deepwalk	0.942	0.820	0.914	0.767	0.822	0.588
Fairwalk	0.936	0.801	1.018	0.746	0.810	0.570
Node2Vec	0.943	0.824	0.895	0.783	0.825	0.588
Infomap	**0.989**	**0.971**	**0.168**	**0.977**	**0.980**	**0.960**
Spinglass	0.703	0.917	3.740	0.179	0.326	0.030
Walktrap	0.943	0.870	0.869	0.432	**0.968**	**0.814**
Fluid	0.953	0.861	0.748	**0.875**	0.842	**0.708**
Label Propagation	0.880	0.830	1.731	0.298	**0.895**	0.547
EM	0.191	-1.974	8.414	0.001	0.064	0.045
SBM	**0.955**	0.947	**0.704**	**0.836**	0.855	0.616
SBM - Nested	**0.954**	0.837	**0.744**	0.773	0.794	0.576

Table C.26: Scores of partition quality measures. CD methods were applied to LFR networks with 10,000 nodes and with $\mu = 0.4$.

C.27 Results: Performance LFR mu 0.6

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.138	0.161	8.652	0.008	0.079	0.012
Combo	0.356	0.493	7.421	0.042	0.234	0.012
Leiden	0.611	**0.685**	5.183	0.179	0.436	0.071
Louvain	0.623	**0.704**	5.011	0.181	0.445	0.072
Paris	0.366	0.031	9.133	0.058	0.214	0.092
RB-C	0.620	**0.690**	5.104	0.204	0.439	0.073
RB-ER	0.624	**0.693**	5.063	0.211	0.429	0.075
Significance	**0.858**	0.425	**2.440**	**0.712**	0.135	0.041
Eigenvector	0.096	0.076	9.749	0.004	0.069	0.002
RSC-K	0.739	0.144	3.924	0.142	**0.672**	**0.419**
RSC-SSE	0.411	-0.844	8.765	0.016	0.210	0.084
RSC-V	**0.838**	0.525	**2.595**	**0.703**	**0.569**	**0.530**
Spectral	0.047	0.075	8.622	0.002	0.328	0.003
Deepwalk	**0.769**	0.304	**3.671**	**0.525**	0.556	**0.389**
Fairwalk	0.751	0.252	3.944	0.497	0.512	0.357
Node2Vec	**0.773**	0.316	**3.621**	**0.532**	0.569	**0.403**
Infomap	**0.850**	0.596	**2.390**	**0.715**	**0.673**	**0.617**
Spinglass	0.510	0.631	6.161	0.112	0.293	0.027
Walktrap	0.674	**190.865**	5.036	0.074	0.191	0.111
Fluid	0.706	0.135	4.739	0.439	0.425	0.296
Label Propagation	0.000	0.000	8.015	0.000	0.020	0.000
EM	0.174	-2.471	8.638	0.001	0.051	0.037
SBM	0.657	0.682	4.291	0.027	**0.781**	0.240
SBM - Nested	0.677	0.640	4.102	0.028	**0.845**	0.335

Table C.27: Scores of partition quality measures. CD methods were applied to LFR networks with 10,000 nodes and with $\mu = 0.6$.

C.28 Results: Performance ABCD xi 0.2

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.608	0.811	4.534	0.075	0.546	0.065
Combo	0.910	0.978	1.302	0.608	0.714	0.281
Leiden	0.915	0.979	1.242	0.627	0.717	0.292
Louvain	0.915	0.979	1.231	0.631	0.715	0.298
Paris	0.743	0.650	2.524	0.645	0.794	0.722
RB-C	0.916	0.980	1.225	0.632	0.721	0.292
RB-ER	0.891	0.913	1.552	0.252	0.767	0.391
Significance	**0.999**	**0.999**	**0.009**	**0.998**	0.969	0.939
Eigenvector	0.286	0.108	8.288	0.022	0.176	0.040
RSC-K	0.962	0.886	0.592	0.600	**0.976**	**0.963**
RSC-SSE	**1.000**	**0.999**	**0.007**	**0.999**	**1.000**	**1.000**
RSC-V	**0.998**	**0.993**	**0.036**	**0.970**	**0.997**	**0.989**
Spectral	0.006	0.009	7.870	0.000	0.477	0.004
Deepwalk	0.975	0.926	0.389	0.837	0.896	0.722
Fairwalk	0.968	0.905	0.506	0.805	0.872	0.666
Node2Vec	0.974	0.922	0.403	0.835	0.883	0.696
Infomap	**1.000**	**1.000**	**0.000**	**1.000**	**1.000**	**1.000**
Spinglass	0.735	0.936	3.326	0.199	0.348	0.033
Walktrap	**1.000**	**1.000**	**0.000**	**1.000**	**1.000**	**1.000**
Fluid	0.975	0.925	0.395	0.880	0.877	0.704
Label Propagation	0.978	0.945	0.346	0.804	0.961	0.835
EM	0.545	-0.214	6.966	0.045	0.218	0.136
SBM	0.964	0.981	0.546	0.851	0.843	0.590
SBM - Nested	0.995	0.984	0.086	0.964	0.960	0.917

Table C.28: Scores of partition quality measures. CD methods were applied to ABCD networks with 10,000 nodes and with $\xi = 0.2$.

C.29 Results: Performance ABCD xi 0.4

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.461	0.701	5.740	0.039	0.420	0.028
Combo	0.878	0.973	1.711	0.495	0.646	0.186
Leiden	0.887	0.976	1.597	0.528	0.659	0.203
Louvain	0.890	0.976	1.559	0.537	0.666	0.212
Paris	0.820	0.540	2.840	0.576	0.780	0.762
RB-C	0.889	0.976	1.570	0.534	0.662	0.214
RB-ER	0.881	0.946	1.672	0.404	0.675	0.262
Significance	**0.998**	**1.000**	**0.032**	**0.992**	0.913	0.840
Eigenvector	0.255	0.177	8.554	0.023	0.138	0.019
RSC-K	0.937	0.815	0.972	0.419	0.953	**0.919**
RSC-SSE	0.957	0.877	0.673	0.693	**0.969**	**0.962**
RSC-V	**0.998**	**0.994**	**0.036**	**0.984**	**0.997**	**0.989**
Spectral	0.103	0.209	7.801	0.004	0.412	0.005
Deepwalk	0.974	0.923	0.403	0.861	0.872	0.682
Fairwalk	0.963	0.893	0.576	0.810	0.860	0.647
Node2Vec	0.974	0.922	0.410	0.859	0.878	0.685
Infomap	**1.000**	**1.000**	**0.000**	**1.000**	**1.000**	**1.000**
Spinglass	0.734	0.936	3.326	0.200	0.346	0.034
Walktrap	**0.998**	**0.996**	**0.027**	**0.995**	**0.999**	**0.999**
Fluid	0.966	0.901	0.545	0.859	0.860	0.688
Label Propagation	0.906	0.858	1.346	0.306	0.938	0.636
EM	0.434	-0.457	8.721	0.017	0.109	0.074
SBM	0.969	0.981	0.476	0.875	0.859	0.619
SBM - Nested	**0.995**	**0.985**	**0.079**	**0.980**	**0.960**	0.912

Table C.29: Scores of partition quality measures. CD methods were applied to ABCD networks with 10,000 nodes and with $\xi = 0.4$.

C.30 Results: Performance ABCD xi 0.6

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.290	0.433	7.169	0.021	0.244	0.012
Combo	0.826	0.942	2.325	0.355	0.553	0.110
Leiden	0.851	0.954	2.029	0.421	0.578	0.134
Louvain	0.853	**0.955**	2.006	0.422	0.587	0.138
Paris	0.588	0.257	6.074	0.222	0.449	0.340
RB-C	0.853	0.954	2.011	0.425	0.584	0.141
RB-ER	0.849	0.943	2.061	0.380	0.603	0.161
Significance	**0.978**	**2.689**	**0.350**	**0.923**	0.590	0.369
Eigenvector	0.144	0.110	9.296	0.009	0.084	0.005
RSC-K	0.858	0.616	2.111	0.133	**0.878**	**0.809**
RSC-SSE	0.757	0.362	3.574	0.100	0.775	0.686
RSC-V	**0.984**	**0.962**	**0.251**	**0.958**	**0.981**	**0.979**
Spectral	0.092	0.181	8.022	0.004	0.041	0.000
Deepwalk	0.957	0.889	0.672	0.847	0.865	0.667
Fairwalk	0.937	0.838	0.992	0.791	0.823	0.602
Node2Vec	0.960	0.895	0.632	0.855	**0.867**	0.668
Infomap	**0.999**	**0.998**	**0.010**	**0.996**	**0.999**	**0.996**
Spinglass	0.733	0.916	3.322	0.203	0.354	0.035
Walktrap	0.872	0.694	1.973	0.511	0.828	**0.724**
Fluid	0.901	0.746	1.566	0.712	0.739	0.552
Label Propagation	0.000	-0.000	7.828	0.000	0.020	0.000
EM	0.356	-0.505	9.888	0.004	0.055	0.045
SBM	**0.964**	**0.977**	**0.549**	**0.856**	0.849	0.569
SBM - Nested	**0.982**	0.952	**0.284**	**0.921**	**0.895**	**0.752**

Table C.30: Scores of partition quality measures. CD methods were applied to ABCD networks with 10,000 nodes and with $\xi = 0.6$.

C.31 Results: Performance HICH-BA MMaj

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.890	0.892	0.436	0.915	0.244	0.057
Combo	**0.961**	**0.962**	**0.166**	**0.977**	**0.865**	**0.720**
Leiden	**0.956**	**0.960**	**0.186**	**0.975**	0.328	0.109
Louvain	**0.956**	**0.960**	**0.186**	**0.975**	0.332	0.111
Paris	0.681	0.680	1.321	0.697	**0.915**	**0.652**
RB-C	**0.956**	**0.961**	**0.187**	**0.975**	0.326	0.108
RB-ER	0.294	0.001	7.237	0.051	0.002	0.000
Significance	0.302	0.001	8.951	0.020	0.003	0.000
Eigenvector	0.672	0.685	1.181	0.693	0.323	0.069
RSC-K	0.222	0.207	2.212	0.041	0.487	**0.487**
RSC-SSE	0.580	0.577	1.768	0.537	**0.711**	**0.594**
RSC-V	0.508	0.501	1.421	0.242	**0.649**	**0.500**
Spectral	0.132	0.129	2.016	0.077	**0.759**	0.207
Deepwalk	0.517	0.511	2.255	0.261	0.528	0.179
Fairwalk	0.513	0.507	2.273	0.263	0.512	0.173
Node2Vec	0.519	0.513	2.207	0.264	0.554	0.217
Infomap	0.669	-0.909	1.999	0.725	0.018	0.000
Spinglass	**0.923**	**0.930**	**0.333**	**0.959**	0.279	0.095
Walktrap	0.789	-2.897	1.030	0.859	0.026	0.001
Fluid	0.616	0.614	1.907	0.578	0.502	0.155
Label Propagation	0.635	0.692	1.151	0.473	0.069	0.006
EM	0.044	0.033	3.913	0.002	0.157	0.017
SBM	0.875	0.881	0.574	0.889	0.409	0.170
SBM - Nested	0.880	0.884	0.549	0.892	0.434	0.184

Table C.31: Scores of partition quality measures. CD methods were applied to HICH-BA networks with 10,000 nodes, containing multiple majority communities.

C.32 Results: Performance HICH-BA MMin

CDMs	NMI	RMI	VI	ARI	PF1	NF1
CNM	0.204	0.168	3.326	0.055	0.016	0.000
Combo	0.388	0.372	2.324	0.082	0.584	0.312
Leiden	0.275	0.253	3.452	0.044	0.184	0.024
Louvain	0.272	0.250	3.303	0.048	0.195	0.026
Paris	**0.694**	**0.694**	**0.394**	**0.759**	**0.890**	**0.450**
RB-C	0.281	0.260	3.367	0.045	0.265	0.050
RB-ER	0.149	0.003	7.812	0.036	0.002	0.000
Significance	0.134	0.004	10.089	0.003	0.004	0.000
Eigenvector	0.147	0.134	2.176	0.021	0.364	0.014
RSC-K	0.462	0.432	0.617	0.509	0.634	**0.634**
RSC-SSE	0.546	0.527	0.679	0.583	**0.713**	**0.662**
RSC-V	**0.848**	**0.845**	**0.243**	**0.905**	**0.782**	**0.557**
Spectral	0.144	0.139	0.724	0.123	**0.876**	0.161
Deepwalk	0.200	0.182	2.392	0.037	0.374	0.063
Fairwalk	0.244	0.227	2.263	0.088	0.418	0.096
Node2Vec	0.209	0.191	2.411	0.048	0.409	0.090
Infomap	0.285	-0.029	3.881	0.203	0.015	0.000
Spinglass	0.260	0.237	3.242	0.048	0.183	0.015
Walktrap	0.224	0.003	5.411	0.165	0.004	0.000
Fluid	0.214	0.198	2.975	0.063	0.169	0.002
Label Propagation	**0.801**	**1.150**	**0.366**	**0.858**	0.082	0.007
EM	0.024	0.002	2.888	-0.035	0.147	0.001
SBM	**0.896**	**0.894**	**0.180**	**0.907**	**0.674**	**0.458**
SBM - Nested	**0.874**	**0.872**	**0.224**	**0.880**	0.597	0.356

Table C.32: Scores of partition quality measures. CD methods were applied to HICH-BA networks with 10,000 nodes, containing multiple minority communities with one large majority community.