Fair Network Communities through Group Modularity

Christos Gkartzios
University of Ioannina
Ioannina, Greece
chgartzios@cs.uoi.gr

Evaggelia Pitoura
University of Ioannina
Ioannina, Greece
pitoura@uoi.gr

Panayiotis Tsaparas
University of Ioannina
Ioannina, Greece
tsap@uoi.gr

Abstract

Communities in networks are groups of nodes that are more densely connected to each other than to the rest of the network, forming clusters with strong internal relationships. When nodes have sensitive attributes, such as demographic groups in social networks, a key question is whether nodes in each group are equally well-connected within each community. We model connectivity fairness using group modularity, an adaptation of modularity that accounts for group structures. We introduce two versions of group modularity, each grounded on a different null model, and propose fairness-aware community detection algorithms. Finally, we provide experimental results on real and synthetic networks, evaluating both the connectivity fairness of community structures in networks and the performance of our fairness-aware algorithms.

CCS Concepts

• **Information systems → Data mining**.

Keywords

algorithmic fairness, community detection, clustering, social networks, group modularity

ACM Reference Format:
Christos Gkartzios, Evaggelia Pitoura, and Panayiotis Tsaparas. 2025. Fair Network Communities through Group Modularity. In *Proceedings of the ACM Web Conference 2025 (WWW '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3696410.3714625

1 Introduction

Networks are essential for representing and analyzing interconnected systems across different domains, such as in social, collaboration, and citation settings. Nodes in networks often form communities, i.e., subsets of nodes that are more tightly connected with each other than with nodes outside the community [15, 25]. Connections in networks play a pivotal role in shaping opinions and influencing decision-making processes [14, 39]. In this paper, we study the *fairness* of connections within network communities.

Algorithmic fairness has been the center of much current research [12, 29, 31, 32]. In a broad sense, fairness is addressed either at the level of individuals, or at the level of groups of individuals [13, 36]. In most networks, nodes have attributes, forming groups, where nodes have the same values in one or more of their attributes. For example, in a social network, groups of nodes often correspond to demographic groups formed based on gender, age, or race. We consider fairness at the level of such groups.

Most previous research in group fairness of communities asks that the representation of groups within each community is balanced [8, 10, 24]. In this paper, we shift the focus from nodes to connections. We ask the key question, whether each group is equally well-connected within each community. For example, consider a collaboration network. Do women in the network participate in an equitable number of connections within the formed communities? The strength of connections within each community is vital for minorities to be heard, and influence others.

To model fairness of connections, we use *modularity*. Modularity is a measure of the quality of community structures in networks that quantifies the strength of the division of a network into communities by comparing the density of edges within communities to the expected density in a random graph [11, 30]. We introduce a variation of modularity, termed *group modularity*, that considers the density of edges of nodes belonging to a specific group. We consider two different random graph models. One agnostic to the group each node belongs to, and one that takes into account group membership. In addition, we propose a *diversity-based* variation of modularity that looks only at connections between nodes belonging to different groups and we address its relationship to group fairness. Diversity of connections is important in addressing filter bubbles, and echo chambers, i.e., cases where individuals in a network are exposed only to opinions similar to their own often leading to reinforcing confirmation bias and polarization [14, 21, 28].

To locate fair community structures in networks, we propose a fairness-aware community detection algorithm. The algorithm builds on the Louvain algorithm [7, 33], an agglomerative hierarchical method, where sets of nodes are successively merged to form larger communities such that modularity increases. In the proposed fairness-aware algorithms, the criterion for merging communities takes into account the fairness and diversity of the communities.

To evaluate our approach, we present experimental results using both synthetic and real networks. The goal of our experimental evaluation is multifold. First, we ask whether community structures in networks are fair and diverse and what are the factors that affect fairness and diversity. Then, we evaluate the trade-off between the quality and the fairness and diversity of communities found by our community detection algorithms and compare the efficacy of the proposed approaches.

The remainder of the this paper is structured as follows. In Section 2, we introduce our model for fairness in communities, and in Section 3, we present the fairness-aware Louvain algorithm. Experimental results are reported in Section 4, related work in Section 5, while Section 6 concludes the paper.

WWW '25, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1274-6/25/04
https://doi.org/10.1145/3696410.3714625

2 Group-based Modularity Fairness

Let $G = (V, E)$ be an undirected graph, where V is the set of nodes and $E \subseteq V \times V$ is the set of edges. We assume that nodes in V belong to groups based on the value of one of their *sensitive attributes*. For simplicity, we assume two values, red and blue, with red being the sensitive one. The red group, denoted by R, $R \subseteq V$, is the subset of nodes with red value. The blue group, denoted by B, $B \subseteq V$, $B \cup R = V$ and $B \cap R = \emptyset$, contains the remaining nodes. We will use ϕ to denote the fraction of the red nodes in the overall population, that is, $\phi = \frac{|R|}{|V|}$.

Let us assume that the nodes of the graph are partitioned into k communities: $C = \{C_1, C_2, \ldots C_k\}$. We will use C_i^B and C_i^R respectively for the blue and red nodes in community C_i.

Most previous research on group fairness focuses on node-based notions of community fairness [10, 24] that seek to maintain a balanced representation of the groups in each community, where the *red balance* of a community $C_i \in C$ is defined as: $B^R(C_i) = \frac{|C_i^R|}{|C_i|} - \phi$.

Given that network processes, including opinion formation, information propagation, and diffusion, primarily occur through interactions along the edges of the network [14, 39], in this paper, we look into group fairness from the edge perspective. To this end, we adopt a modularity-based approach.

Modularity measures the divergence between the number of intra-community edges and the expected such number assuming a null model [11, 30]. The most commonly used null model is a random graph where the expected degree of each node within the graph is equal to the actual degree of the corresponding node in the real network. Specifically, the modularity of community C_i, $Q(C_i)$, is defined as [30]:

$$Q(C_i) = \frac{1}{2m}\left(\sum_{u \in C_i}\sum_{v \in C_i} A_{uv} - \frac{k_u\, k_v}{2m}\right) \tag{1}$$

where A is the adjacency matrix of G, m the number of edges in G and k_u, k_v the degree of node u, and v respectively. Modularity provides a measure of how well nodes in a community are connected with each other. Negative values indicate less connections than expected, while positive values indicate more connections.

2.1 Group Modularity

Our goal is to ensure that red nodes are well connected within each community. Thus, for each red node u in C_i we take the difference between the actual number of its intra-community edges and the expected such number. We call this measure *red modularity*.

As before, the expected number of connections is estimated assuming as null model a random graph that preserves the degrees of nodes in G. Using this null model, red modularity, $Q^R(C_i)$ is defined as:

$$Q^R(C_i) = \frac{1}{2m}\sum_{u \in C_i^R}\sum_{v \in C_i}\left(A_{uv} - \frac{k_u\, k_v}{2m}\right). \tag{2}$$

We define similarly the *blue modularity* $Q^B(C_i)$. We refer to red and blue modularity collectively as *group modularity*.

Note that if we consider the whole graph as a single community both the red and the blue modularity are zero. In general, positive values in a community mean that the nodes with the corresponding color are more connected in the community than expected.

We define *(group) modularity unfairness* by comparing the red and blue modularity.

Definition 2.1. For a community $C_i \in C$, the modularity unfairness of C_i, $u(C_i)$, is defined as:

$$u(C_i) = Q^R(C_i) - Q^B(C_i).$$

Negative values of $u(C_i)$ indicate unfairness towards the red group meaning that the red nodes are less well-connected within the community than the blue ones. Positive values indicate the opposite, while a zero value indicates lack of unfairness towards any of the groups.

We also consider diversity within each community by looking at the edges that connect nodes of different color. Let us call these edges *diverse edges*. Note that the expected number of diverse edges cannot be estimated using the same null model, since we need to know the color of both endpoints of each edge. Instead, in this case, we estimate the expected number of diverse edges using as null model a random bipartite graph, with edges only between nodes of different color, that preserves the degrees of the nodes in the original graph G.

For a community C_i, the *diversity modularity*, or simply *diversity*, is defined as:

$$D^{RB}(C_i) = \frac{1}{2m}\sum_{u \in C_i^R}\sum_{v \in C_i^B}\left(A_{uv} - \frac{k_u\, k_v}{m}\right). \tag{3}$$

If we consider the whole graph as a single community, then diversity takes a non positive value. The larger the value of D^{RB} the more diverse the network.

Simplifying the forms. Let $X \in \{R, B\}$ and $Y \in \{R, B\}$. We use In_i to denote the number of intra-community edges in C_i, In_i^X, the number of intra-community edges in C_i with at least one endpoint belonging to group X, and In_i^{XY} the number of intra-community edges with one endpoint in group X and one endpoint in group Y. We also use K_i for the sum of degrees of all nodes in C_i, and K_i^X for the sum of degrees of all nodes in C_i that belong to group X. With simple manipulations, we get:

$$Q^R(C_i) = \frac{1}{2m}\left(2In_i^{RR} + In_i^{RB} - \frac{K_i K_i^R}{2m}\right) \tag{4}$$

$$u(C_i) = \frac{In_i^{RR} - In_i^{BB}}{m} - \frac{(K_i^R)^2 - (K_i^B)^2}{(2m)^2} \tag{5}$$

$$D^{RB}(C_i) = \frac{1}{2m}\left(In_i^{RB} - \frac{K_i^R K_i^B}{m}\right) \tag{6}$$

2.2 Labeled Group Modularity

We now consider a null model which is not agnostic of the color of edge endpoints. For a node u, let k_u^R be the number of edges of u to red nodes and k_u^B be the number of edges of u to blue nodes,

$k_u^R + k_u^B = k_u$. In the following, k_u^R and k_u^B are respectively called the *red degree* and *blue degree* of node u.

We consider as null model a random graph where the expected red degree and the expected blue degree of each node is equal to the actual red degree and blue degree of the corresponding node in the real graph G. Formally, let P_{uv} be the probability of creating an edge between nodes u and v. Let m_{RR} be the number of red-red edges, m_{RB} the number of red-blue edges and m_{BB} the number of blue-blue edges in the graph. We have that $P_{uv} = k_u^R\, k_v^R / 2m_{RR}$, for red nodes $u, v \in R$, $P_{uv} = k_u^B\, k_v^B / 2m_{BB}$ for blue nodes $u, v \in B$, and $P_{uv} = k_u^B\, k_v^R / m_{RB}$ for red-blue nodes $u \in R$ and $v \in B$. For any node u, it holds that $\sum_{v \in R} P_{uv} = k_u^R$ and $\sum_{v \in B} P_{uv} = k_u^B$.

We define the *labeled red modularity* $Q_L^R(C_i)$ by taking again the difference between the actual number of intra-community edges involving red nodes, and the expected such number, but now considering the color (or, in general, label) of both endpoints.

$$Q_L^R(C_i) = \frac{1}{2m}\Big(\sum_{u \in C_i^R} \sum_{v \in C_i^B} (A_{uv} - \frac{k_u^B\, k_v^R}{m_{RB}}) + \sum_{u \in C_i^R} \sum_{v \in C_i^R} (A_{uv} - \frac{k_u^R\, k_v^R}{2m_{RR}}) \Big). \tag{7}$$

We define similarly the *labeled blue modularity* $Q_L^B(C_i)$. We refer to labeled red and labeled blue modularity collectively as *labeled group modularity*. Again, if we consider the whole graph as a single community both the labeled red and the labeled blue modularity are zero.

We define the *labeled modularity unfairness* by comparing the red and blue labeled modularity.

Definition 2.2. For a community $C_i \in C$, the labeled modularity unfairness of C_i, $u_L(C_i)$, is defined as:

$$u_L(C_i) = Q_L^R(C_i) - Q_L^B(C_i).$$

Negative values of $u_L(C_i)$ indicate unfairness towards the red group, positive values indicate unfairness towards the blue group, and a zero value lack of unfairness.

We define *labeled diversity modularity*, or simply *labeled diversity*, as follows:

$$D_L^{RB}(C_i) = \frac{1}{2m} \left(\sum_{u \in C_i^R} \sum_{v \in C_i^B} \left(A_{uv} - \frac{k_u^B\, k_v^R}{m_{RB}} \right) \right). \tag{8}$$

The labeled diversity of the whole graph is zero, while positive diversity values in a community indicate that the community contains more diverse edges than expected.

Simplifying the forms. We use K_i^{XY}, with $X, Y \in \{R, B\}$, to denote the sum of the degrees of all nodes of color X that belong to C_i to any node of color Y. With simple manipulations, we get:

$$Q_L^R(C_i) = \frac{1}{2m} \left(2\, In_i^{RR} + In_i^{RB} - \frac{K_i^{RB} K_i^{BR}}{m_{RB}} - \frac{(K_i^{RR})^2}{2m_{RR}} \right) \tag{9}$$

$$u_L(C_i) = \frac{In_i^{RR} - In_i^{BB}}{m} - \frac{(K_i^{RR})^2 - (K_i^{BB})^2}{4\, m\, m_{RB}} \tag{10}$$

$$D_L^{RB}(C_i) = \frac{1}{2m} \left(In_i^{RB} - \frac{K_i^{RB} K_i^{BR}}{m_{RB}} \right) \tag{11}$$

Discussion. Note that both diversity and labeled diversity are symmetric, that is, it holds that $D^{RB}(C_i) = D^{BR}(C_i)$, and $D_L^{RB}(C_i) = D_L^{BR}(C_i)$. Furthermore, communities whose edges are all diverse have zero labeled modularity unfairness. However, the opposite does not necessarily hold: communities can be fair without necessarily being diverse.

3 Fairness-Aware Community Detection

In this section, we present our fairness-aware community detection algorithm. Our algorithm is based on the well-known Louvain algorithm that identifies communities in networks by optimizing modularity [7, 11, 33].

The fairness-aware Louvain algorithm (Algorithm 1) follows a hierarchical agglomerative approach, starting with each node forming its own community. The original Louvain algorithm joins together two communities whose merge produces the largest increase in modularity Q (Eq. 1). The fairness-aware algorithm uses two-criteria: two communities are joined if (1) modularity increases and (2) a group fairness criterion (FC) is met.

For FC, we consider different approaches using either the non-labeled and the labeled group modularity, namely:

(a) the *fairness-gain* approach where we ask that unfairness in absolute value decreases (Eq. 5, or 10),
(b) the *group-increase* approach, where we ask that the group modularity of the group towards which the network is unfair increases (Eq. 4, or 9), and
(c) the *diversity-increase* approach where we ask that diversity increases (Eq. 6, or 11).

The algorithm operates in two phases that are repeated iteratively. In the first phase, the algorithm computes for each node u the gain in modularity and the fairness criterion FC when removing u from its current community and placing it to each of its neighboring communities. This process is applied repeatedly and sequentially and stops when a local maxima is reached, i.e., when no individual move can both increase modularity and satisfy the FC criterion.

In the second phase, the algorithm constructs a new graph whose nodes are now the communities found during the first phase. The algorithm operates on a weighted graph; each edge e is associated with a weight, $w(e)$, initially set to 1. Upon merging, the weights of the edges between the two nodes are set equal to the sum of the weights of the edges in the corresponding two communities. Edges between the nodes inside each community are modeled with a self-loop whose weight is the sum of the weights of these edges. For the node degrees, it holds $k(u) = \sum_{v,(u,v) \in E} w(u, v)$ and $k^X(u) = \sum_{(u,v) \in E, v \in X} w(u, v)$, for $X \in \{R, B\}$.

Once the graph is constructed, the first and second phase are repeated on the new graph. The iterations continue until there are no changes. The computational complexity of Algorithm 1 is $O(L|E|)$, the same with the original Louvain, where $O(|E|)$ is the complexity of the two phases, and L the number of iterations.

The following lemma (proof in the Appendix) estimates the change in red modularity $\Delta Q^R_{u \to C_i}$, blue modularity $\Delta Q^B_{u \to C_i}$ and diversity $\Delta D^{RB}_{u\to C_i}$ when an isolated red node u moves to a community C_i.

Lemma 3.1. *When an isolated red node $u \in R$ is moved to community C_i, the difference $\Delta Q^R_{u\to C_i}$ in red modularity is:*

$$\Delta Q^R_{u\to C_i} = \frac{1}{2m}\left(2\sum_{v\in C_i, v\in R} w(u,v) + \sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u(K_i+K_i^R)}{2m}\right)$$

the difference $\Delta Q^B_{u\to C_i}$ in blue modularity is:

$$\Delta Q^B_{u\to C_i} = \frac{1}{2m}\left(\sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u K_i^B}{2m}\right)$$

and the difference $\Delta D^{RB}_{u\to C_i}$ in diversity is:

$$\Delta D^{RB}_{u\to C_i} = \frac{1}{2m}\left(\sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u K_i^B}{m}\right)$$

Similar formulas hold when moving a blue node u to C_i.

The following lemma (proof in the Appendix) estimates the change in labeled red $\Delta Q^R_{L,u\to C_i}$, labeled blue modularity $\Delta Q^B_{L,u\to C_i}$ and labeled diversity $\Delta D^{RB}_{L,u\to C_i}$ when an isolated red node u moves to a community C_i.

Lemma 3.2. *When an isolated red node $u \in R$ is moved to community C_i, the difference $\Delta Q^R_{L,u\to C_i}$ in labeled red modularity is:*

$$\Delta Q^R_{L,u\to C_i} = \frac{1}{2m}(2\sum_{v\in C_i, v\in R} w(u,v) + \sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u^B K_i^{BR}}{m_{RB}} + \frac{k_u^R K_i^{RR}}{m_{RR}})$$

the difference $\Delta Q^B_{L,u\to C_i}$ in labeled blue modularity is:

$$\Delta Q^B_{L,u\to C_i} = \frac{1}{2m}\left(\sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u^B K_i^{BR}}{m_{RB}}\right)$$

and the difference $\Delta D^{RB}_{L,u\to C_i}$ in labeled diversity is:

$$\Delta D^{RB}_{L,u\to C_i} = \frac{1}{2m}\left(\sum_{v\in C_i, v\in B} w(u,v) - \frac{k_u^B K_i^{BR}}{m_{RB}}\right).$$

Similar formulas hold when moving a blue u node to C_i. In the Appendix, we also present formulas applicable when merging communities.

4 Experiments

The goal of our experiments is to address the following research questions (RQ):

- RQ1 What are the characteristics of a network that contribute to unfairness and lack of diversity within communities?
- RQ2 Under which network conditions and through what modifications of fairness-aware Louvain algorithms can improvement in both notions of fairness be attained?
- RQ3 How effective are the two definitions of unfairness and diversity in quantifying their respective measure and consecutively improving fair community detection?

Algorithm 1 Fairness-Aware Louvain

Input: Graph $G(V, E, A)$ where V is the set of nodes, E is the set of edges, and A are the node colors
Output: List of communities detected.
repeat
  Assign every node $v \in V$ to a singleton community
  Calculate modularity Q
  for each node $v \in V$ **do**
    for each u in neighbors of v **do**
      Calculate the modularity gain ΔQ and fairness criterion FC from the removal of v from its current community and placement in the community of each neighbor.
      if modularity increases and FC is met **then**
        Move v to neighboring community
      end if
    end for
  end for
  Create new "super-nodes" from the communities found in previous step. The new V set consists of these "super-nodes".
  Recalculate the weight of the edges between these new "super-nodes".
until there is no improvement

To address these questions we conducted experiments on both synthetic and real datasets. To account for the randomness in the order of considering nodes, we selected the best communities from 10 runs. The code is available on GitHub[1].

Table 1: Synthetic Dataset Characteristics

Parameter	Meaning	Default
N	Number of nodes	1000
ϕ	Ratio of red nodes	0.5
l	Avg node degree	5
k	Initial number of communities	5
p_h, p_h^R, p_h^B	Homophily	0.5
p_c	Prob. of intra-community edge	0.9

4.1 Datasets

4.1.1 Synthetic Datasets. To study the factors that may lead to unfairness, we use a model based on the stochastic block model [20, 24] to create networks with nodes of different colors and connectivity behavior. The model has three important parameters: (1) Parameter ϕ controls the size imbalance between the different groups. In a perfectly node-balanced network, $\phi = 0.5$; smaller values make the red group the minority one. (2) Parameter p_c controls the probability of intra-community edges. In a random network with no community structure, $p_c = 0.5$; communities appear as p_c increases. (3) Parameter p_h controls the probability of same color edges, i.e., homophily. Values of p_h larger than 0.5 result in homophily, while values smaller than 0.5 result in heterophily. When $p_h = 0.5$, we have neutrality.

[1] https://github.com/gartzis/Fair-Network-Communities-through-Group-Modularity.git

We start by an initial assignment of nodes in k communities and then generate edges between the nodes. Note that the actual number of communities created may differ from k, depending on the values of the other parameters. An edge (u, v) is generated with probability $p(u, v)$ defined as follows:

$$p(u,v) = \begin{cases} p_c\, p_h, & \text{if } u \text{ and } v \text{ are in the same cluster} \\ & \text{and have the same color,} \\ (1-p_c)\, p_h, & \text{if } u \text{ and } v \text{ are in different clusters} \\ & \text{and have the same color,} \\ p_c\,(1-p_h), & \text{if } u \text{ and } v \text{ are in the same cluster} \\ & \text{and have different colors,} \\ (1-p_c)\,(1-p_h), & \text{if } u \text{ and } v \text{ are in different clusters} \\ & \text{and have different colors.} \end{cases}$$

We also consider an asymmetric case, with different homophily probabilities, p_h^R and p_h^B, for the red and the blue nodes respectively. When generating edge $p(u, v)$, we use p_h^R if $u \in R$, and p_h^B if $u \in B$.

Table 1 summarizes the parameters. We study the influence of size imbalance (ϕ) and homophily (p_h). In each case, we vary one of the parameters and use the default values for the other. In each case, we create 10 random networks and report average values.

4.1.2 Real datasets. We use the following real datasets[2]:

- **Pokec.** Nodes are users of Pokec, a Sloval social network, and edges are friendship relationships between them. We consider the gender attribute (**Pokec-g**) and the age attribute (**Pokec-a**) as sensitive ones. For age, we create two groups based on whether the user is over 30 years old or not.
- **Deezer.** Nodes are users of Deezer, a music streaming platform, from European countries and edges are mutual follower relationships between them. The sensitive attribute is gender.
- **Facebook.** The dataset consists of friends list from Facebook. We consider the gender attribute **Facebook-g** and the concentration attribute **Facebook-c**. The concentration attribute is the specialized field of study the users chose as their major.
- **Twitch.** Nodes are users of Twitch, a live streaming platform for gamers, and edges are mutual follower relationships between them. The sensitive attribute is gender.

The dataset characteristics are summarized in Table 2. We also report homophily values that indicate the tendency of nodes to connect with nodes with the same sensitive attribute. We report separately the homophily of the red nodes (Rh) and the homophily of the blue nodes (Bh). Red homophily (Rh) is computed as the ratio of the number of the actual edges connecting two red nodes and the expected number of such edges (estimated as ϕ^2). $Rh > 1$ indicate homophily, while $Rh < 1$ heterophily (tendency to connect with nodes of the opposite color). Similarly, we compute the blue homophily (Bh) as the ratio of the number of the actual edges between two blue nodes and the expected number of such edges (estimated as $(1-\phi)^2$).

[2] https://snap.stanford.edu/data/

4.2 Evaluation Results

To evaluate our approach on synthetic data, we create both symmetric and asymmetric datasets. Our goal is to examine how different group distributions and homophily patterns impact unfairness and diversity.

We created datasets based on three distinct values of ϕ:

- Red minority $\phi = 0.2$, where the red group is underrepresented relative to the blue group.
- Balanced groups $\phi = 0.5$, where the red and blue groups are evenly represented.
- Red majority group $\phi = 0.8$, where the red group forms the majority.

In addition to the group ratio, we adjust the homophily parameter p_h, which controls the likelihood of nodes within the same group to connect. We explore values of p_h in the range from 0.1 to 0.9, capturing the range from strongly heterophilic ($p_h = 0.1$) to strongly homophilic ($p_h = 0.9$) networks. We create symmetric datasets, where the same homophily is assigned to both groups, allowing us to evaluate networks where both groups follow similar connectivity patterns. We also create asymmetric datasets, where the homophily parameter between the red and the blue group is different. In this case, we fix the homophily of the red group to be neutral ($p_h^R = 0.5$), and we vary the homophily of the blue group p_h^B, from 0.1 (heterophilic) to 0.9 (homophilic).

To address RQ1, we apply the original Louvain algorithm on each of the synthetic networks; the results are shown in the first row of Figures 1 (symmetric) and 2 (asymmetric). Our analysis reveals that unfairness and diversity are correlated with the homophily and the group size of the network. Specifically, higher homophily is associated with increased unfairness and reduced diversity, as nodes tend to connect with nodes of the same type, which reinforces group isolation within communities.

The most favorable outcomes for both network fairness and diversity occur when group sizes are balanced and homophily is moderate, allowing cross-group interactions (Figure 2(c)). The highest levels of unfairness are observed when both groups exhibit high homophily. As both red and blue homophily increase, the unfairness metric moves further from zero, which indicates that both groups are predominantly forming internal connections, leading to high segregation between groups.

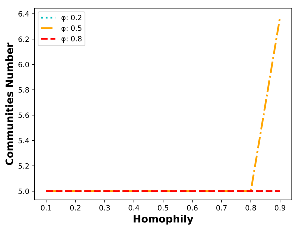
In cases of asymmetry (Figure 2), we observe distinct patterns based on the homophily levels of the blue group. When the blue group is the majority group ($\phi = 0.2$), fairness declines as the group becomes more homophilic, with unfairness values becoming increasingly negative. In contrast, when the blue group is the minority group ($\phi = 0.8$), communities become less unfair. Diversity is higher when the blue group is the smaller one.

Similar patterns are also evident when Louvain is applied to the real networks. The results for the proposed diversity and fairness metrics are presented in Table 3 (Louvain). In particular, we find that networks with high group size disparities, such as Pokec-a and Facebook-c, exhibit high levels of unfairness within the detected communities.

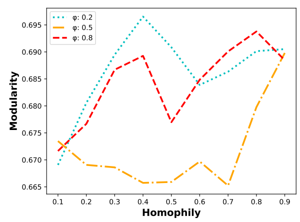
To investigate RQ2, we evaluate the various fairness-aware approaches, i.e, the fairness-gain, group-increase, and diversity-increase

Table 2: Real dataset characteristics, $\bar{K}^X$: average degrees, Rh (Bh): red (blue) homophily.

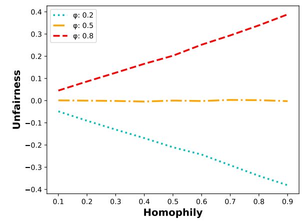
Network	#Nodes	#Edges	Attribute	#Red nodes	#Blue nodes	$\bar{K}^R$	$\bar{K}^B$	$\bar{K}^{RR}$	$\bar{K}^{BB}$	$\bar{K}^{RB}$	Rh	Bh	ϕ
Pokec-g	1,632,636	22,301,602	Gender	804,335	828,301	26.33	28.28	5.18	6.39	15.49	0.770	0.922	0.492
Pokec-a	1,632,636	22,301,602	Age	239,785	1,392,851	15.95	29.27	0.79	13.40	2.47	0.394	1.149	0.146
Deezer	28,281	92,752	Gender	12,538	15,743	6.34	6.73	1.41	1.96	2.79	0.972	1.07	0.443
Facebook-g	4,039	88,234	Gender	1,533	2,506	45.75	42.42	10.24	13.48	15.45	1.236	0.995	0.378
Facebook-c	4,039	88,234	Education	367	3,672	31.38	44.92	2.94	21.18	2.54	1.481	1.066	0.090
Twitch	168,114	6,797,557	Maturity	79,033	89,081	88.25	74.31	24.57	19.80	34.69	1.292	0.924	0.470



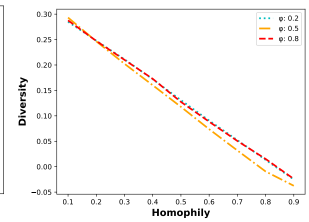
(a) Louvain Communities



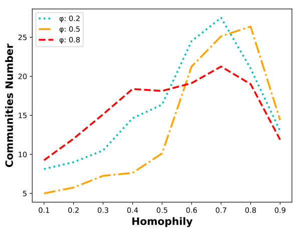
(b) Louvain Modularity



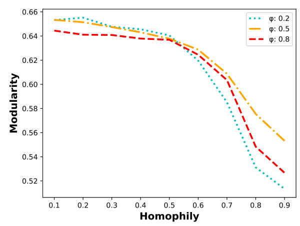
(c) Louvain Unfairness



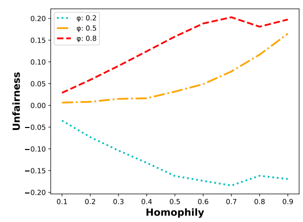
(d) Louvain Diversity



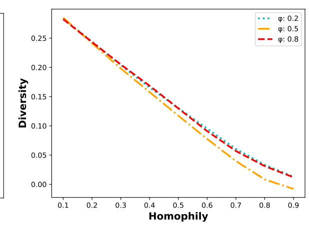
(e) Group-Increase Communities



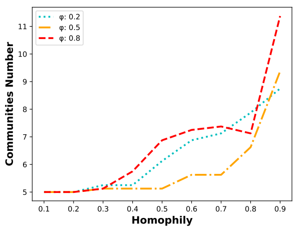
(f) Group-Increase Modularity



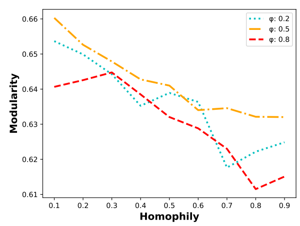
(g) Group-Increase Unfairness



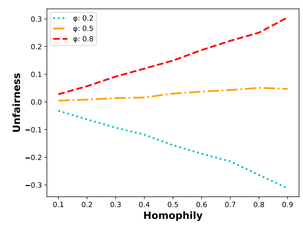
(h) Group-Increase Diversity



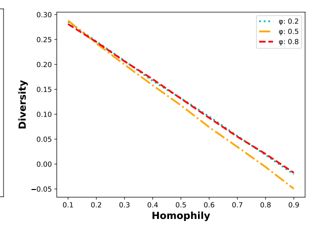
(i) L-Group-Increase Communities



(j) L-Group-Increase Modularity



(k) L-Group-Increase Unfairness



(l) L-Group-Increase Diversity

Figure 1: Symmetric Datasets

approaches introduced in Section 3. We use the prefix *L* for the approaches that use the labeled group modularity. The group-increase approach, where we increase the group modularity of the group towards which there is unfairness, proved consistently the most effective one. For the synthetic datasets, this is the red group when $\phi = 0.2$ and the blue group when $\phi = 0.8$. Rows 2 and 3 of Figures 1 and 2 show the results of the group-increase and L-group-increase approaches on the synthetic networks. Both methods reduce unfairness, but at the cost of an increase in the number of communities. This pattern is also observed in the real-world datasets (Table 3 (Group-Increase) and (L-Group-Increase)), particularly in networks with high group size imbalance, such as Pokec-a.

In terms of improving diversity, in Table 4, we report results for the real datasets using the approach that achieves the highest diversity gain. Interestingly, diversity-increase is the best one only for Deezer. Also, for Pokec-g, the approach that increases the group fairness of the benefited blue group performs best probably due to the high number of diverse edges in this dataset. In general, while there is improvement in diversity in most networks, such as Deezer (0.157) and Twitch (0.053), this also leads to a notable increase in the number of communities. This further underscores the challenge of achieving fairness and diversity while maintaining cohesive network structures. Additional results with other approaches are found in the Appendix.

For the last research question, RQ3, we found that while all approaches successfully improve their respective fairness objectives, they often lead to an increase in the number of communities (second and third rows of Figures 1 and 2). Notably, we find that the group-increase modification is more effective at decreasing unfairness and ensuring equitable distribution of edges across groups, particularly in high homophily networks with group size disparity. However this improvement comes at the cost of an increase in the number

Table 3: Communities formed by different approaches.

Network	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Louvain						
Pokec-g	41	0.716	-0.031	-0.031	0.180	0.192
Pokec-a	39	0.713	-0.589	-0.589	0.050	0.055
Deezer	89	0.683	-0.103	-0.101	0.141	0.160
Facebook-g	16	0.834	-0.167	-0.175	0.152	0.181
Facebook-c	16	0.834	-0.725	-0.722	0.038	0.042
Twitch	23	0.420	0.001	-0.004	0.043	0.090
Group-Increase Approach						
Pokec-g	58,369	0.695	-0.019	-0.019	0.178	0.191
Pokec-a	179,082	0.616	-0.490	-0.490	0.052	0.056
Deezer	2,945	0.593	0	0	0.146	0.162
Facebook-g	204	0.818	-0.149	-0.157	0.154	0.181
Facebook-c	1,462	0.592	-0.480	-0.479	0.040	0.043
Twitch	5,396	0.394	0.005	0	0.042	0.087
L- Group-Increase Approach						
Pokec-g	1,440	0.691	-0.024	-0.023	0.172	0.188
Pokec-a	7,645	0.658	-0.531	-0.531	0.057	0.058
Deezer	323	0.649	-0.061	-0.059	0.142	0.164
Facebook-g	17	0.830	-0.162	-0.169	0.152	0.182
Facebook-c	21	0.822	-0.711	-0.707	0.039	0.043
Twitch	526	0.384	0.004	-0.015	0.022	0.084

Table 4: Communities formed by the fairness-aware Louvain to improve diversity (various approaches used).

Network	Approach	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Pokec-g	Group-Increase (blue group)	41,273	0.705	-0.038	-0.038	0.181	0.194
Pokec-a	Group-Increase (red group)	179,082	0.616	-0.490	-0.0490	0.052	0.057
Deezer	Diversity-Increase	5,366	0.567	-0.063	-0.062	0.157	0.173
Facebook-g	Group-Increase (red-group)	204	0.818	-0.149	-0.157	-0.154	0.182
Facebook-c	Group-Increase (red-group)	1,462	0.592	-0.480	-0.479	0.041	0.043
Twitch	Group-Increase (blue-group)	1,299	0.401	-0.005	-0.007	0.053	0.088

of communities and a reduction in modularity. In contrast, the L-group-increase approaches are more effective in preserving high modularity while limiting the number of communities, especially in cases of group size disparity. However, they achieve a more moderate improvement in fairness.

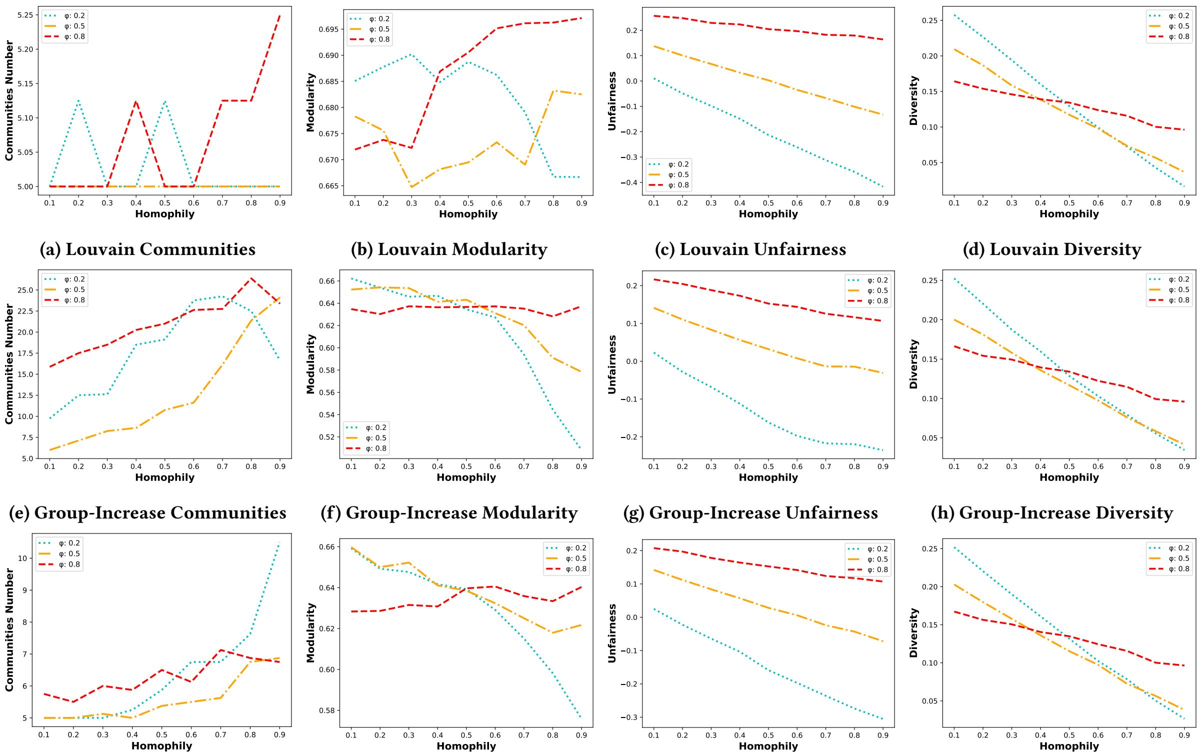
5 Related Work

Fairness in machine learning has received considerable attention [12, 29, 31, 32, 35]. At a high-level, fairness models are distinguished based on whether fairness is addressed at the level of individuals or at the level of groups of individuals [13, 36]. In this paper, we study the specific problem of group fairness of communities in networks. Community detection is similar to the more general problem of clustering defined as the task of grouping a set of objects in clusters such that the objects in the same cluster are more similar to each other than to those in other clusters [22]. In the case of communities, nodes are grouped so that nodes inside each community, i.e, cluster, are more tightly connected with each other than with nodes outside the community [15]. Next, we place our work in the context of previous work on fairness in community detection and clustering.

As opposed to our approach that defines fairness based on node connections, most group fairness definitions are based on balancing the representation of each group within each cluster. The balanced approach to fairness in clustering was introduced in the seminal work of fairlets [10] to ensure that each protected group must have approximately equal representation in each cluster. The approach has been extended along various directions, such as to support scalability and distributed processing [3, 6, 8], more than one protected group [5] and parametric fair representation [4].

A different model of proportionality fairness does not assume protected groups but seeks fair treatment for any subset of points [9]. A related notion but for individual fairness was studied in [26] based on a previous formulation of the fair facility allocation problem. A general definition of individual fairness in graph mining is that similar nodes should receive similar output. Applying this definition to graph clustering means that similar nodes should receive similar cluster assignments [23]. Given a similarity matrix that encodes the pair-wise similarity between nodes, this definition results in each node having most of its neighbors in this graph in the same cluster. The approach was extended in [38] for multiview graph clustering. Yet another approach assumes the existence of a representation graph between nodes and asks that the neighbors of each node are proportionately represented in each cluster [18, 19].

In terms of fairness in clustering, another view considers the cluster quality for each group. This approach is taken in the *socially fair* k-means clustering approach that seeks to minimize the maximum



(a) Louvain Communities (b) Louvain Modularity (c) Louvain Unfairness (d) Louvain Diversity

(e) Group-Increase Communities (f) Group-Increase Modularity (g) Group-Increase Unfairness (h) Group-Increase Diversity

(i) L-Group-Increase Communities (j) L-Group-Increase Modularity (k) L-Group-Increase Unfairness (l) L-Group-Increase Diversity

Figure 2: Asymmetric Datasets

of the average k-means objective applied to each group [16] and in *equitable* clustering that seeks to minimize the distance of each point to its nearest center [1]. In a sense, group modularity follows this view, since its goal is maintaining good clustering quality in terms of intra-cluster connectivity for each group.

In contrast to previous work, in this paper, we define community fairness using group modularity. Modularity has been refined to promote *mixed links*, i.e., links connecting nodes of different color in link recommendations [28]. This refinement is similar to our definition of diversity under the first null model. Red modularity using the first null model was introduced in a short paper [27].

Finally, in terms of algorithms for graph clustering, in this paper, we propose a modularity-based algorithm. Most previous work on addressing fairness considers algorithms based on spectral clustering, that add fairness constraints. Work in [24] adds constraints to spectral clustering for balancing nodes and is extended in [37] for scalability. Work in [23, 38] imposes individual fairness using spectral clustering of the adjacency matrix combined with the similarity matrix. Spectral-based approaches are also followed in [17–19].

6 Conclusions

In many real-world networks, communities are formed, where nodes in each community are more tightly connected with the nodes inside their community than with the nodes outside their community. In this paper, we studied the fairness of such communities. Specifically, given that the nodes in a network belong to different groups, we examine whether the nodes of each group are equally well connected within the communities. To capture connectivity fairness, we proposed group modularity, an adaptation of modularity. We also used modularity to study the diversity of communities, i.e., the percentage of inter-group edges within each community. We proposed a fairness-aware Louvain-based algorithm that detects communities with good modularity, and fairness, or diversity. Our experimental evaluation showed the effects of homophily and size discrepancy in the fairness and diversity of the formed communities.

Our modularity-based metrics of fairness and diversity are orthogonal to the community detection algorithms used. As future work, we plan to investigate alternative approaches to community detection. Another directions for future work are understanding the evolution of community fairness and diversity through time and investigating approaches for improving the fairness and diversity of communities by link recommendations, as for example in previous work on improving Pagerank fairness [34], and fighting opinion formation [2].

Acknowledgments

This work has been implemented within the framework of the H.F.R.I call “Basic Research Financing" (H.F.R.I. Project Number: 016636), under the National Recovery and Resilience Plan "Greece 2.0," funded by the European Union - NextGenerationEU.

References

[1] Mohsen Abbasi, Aditya Bhaskara, and Suresh Venkatasubramanian. 2021. Fair Clustering via Equitable Group Representations. In *FAccT*. 504–514.
[2] Victor Amelkin and Ambuj K. Singh. 2019. Fighting Opinion Control in Social Networks via Link Recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. ACM, 677–685.
[3] Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. 2019. Scalable Fair Clustering. In *ICML (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 405–413.
[4] Suman Bera, Deeparnab Chakrabarty, Nicolas Flores, and Maryam Negahbani. 2019. Fair algorithms for clustering. *Advances in Neural Information Processing Systems* 32 (2019).
[5] Suman K. Bera, Deeparnab Chakrabarty, Nicolas J. Flores, and Maryam Negahbani. 2019. Fair Algorithms for Clustering. In *NeurIPS*. 4955–4966.
[6] Suman K Bera, Syamantak Das, Sainyam Galhotra, and Sagar Sudhir Kale. 2022. Fair k-center clustering in MapReduce and streaming settings. In *Proceedings of the ACM Web Conference 2022*. 1414–1422.
[7] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment* 10 (2008).
[8] Matteo Ceccarello, Andrea Pietracaprina, and Geppino Pucci. 2024. Fast and Accurate Fair k-Center Clustering in Doubling Metrics. In *Proceedings of the ACM on Web Conference 2024*. 756–767.
[9] Xingyu Chen, Brandon Fain, Liang Lyu, and Kamesh Munagala. 2019. Proportionally Fair Clustering. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 1032–1041.
[10] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. 2017. Fair Clustering Through Fairlets. In *NeurIPS*. 5029–5037.
[11] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. 2004. Finding community structure in very large networks. *Phys. Rev. E* 70 (2004). Issue 6.
[12] Yushun Dong, Jing Ma, Song Wang, Chen Chen, and Jundong Li. 2023. Fairness in Graph Mining: A Survey . *IEEE Transactions on Knowledge & Data Engineering* 35, 10 (2023), 10583–10602.
[13] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Rich Zemel. 2012. Fairness through awareness. In *ITCS*. ACM, 214–226.
[14] David A. Easley and Jon M. Kleinberg. 2010. *Networks, Crowds, and Markets - Reasoning About a Highly Connected World*. Cambridge University Press.
[15] Santo Fortunato. 2010. Community detection in graphs. *Physics reports* 486, 3-5 (2010), 75–174.
[16] Mehrdad Ghadiri, Samira Samadi, and Santosh Vempala. 2021. Socially Fair k-Means Clustering. In *FAccT*. ACM, 438–448.
[17] Siamak Ghodsi, Seyed Amjad Seyedi, and Eirini Ntoutsi. 2024. Towards Cohesion-Fairness Harmony: Contrastive Regularization in Individual Fair Graph Clustering. In *Advances in Knowledge Discovery and Data Mining - 28th Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD 2024, Taipei, Taiwan, May 7-10, 2024, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 14645)*. Springer, 284–296.
[18] Shubham Gupta and Ambedkar Dukkipati. 2021. Protecting Individual Interests across Clusters: Spectral Clustering with Guarantees. *CoRR* abs/2105.03714 (2021). arXiv:2105.03714 https://arxiv.org/abs/2105.03714
[19] Shubham Gupta and Ambedkar Dukkipati. 2022. Consistency of Constrained Spectral Clustering under Graph Induced Fair Planted Partitions. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.
[20] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. 1983. Stochastic blockmodels: First step. *Social Networks* 5 (1983), 109–137.
[21] Ruben Interian, Ruslán G. Marzo, Isela Mendoza, and Celso C. Ribeiro. 2023. Network polarization, filter bubbles, and echo chambers: an annotated review of measures and reduction methods. *Int. Trans. Oper. Res.* 30, 6 (2023), 3122–3158.
[22] Anil K Jain, M Narasimha Murty, and Patrick J Flynn. 1999. Data clustering: a review. *ACM computing surveys (CSUR)* 31, 3 (1999), 264–323.
[23] Jian Kang, Jingrui He, Ross Maciejewski, and Hanghang Tong. 2020. InFoRM: Individual Fairness on Graph Mining. In *KDD*. ACM, 379–389.
[24] Matthäus Kleindessner, Samira Samadi, Pranjal Awasthi, and Jamie Morgenstern. 2019. Guarantees for Spectral Clustering with Fairness Constraints. In *ICML*, Vol. 97. 3458–3467.
[25] Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman. 2014. *Mining of Massive Datasets, 2nd Ed.* Cambridge University Press.
[26] Sepideh Mahabadi and Ali Vakilian. 2020. Individual Fairness for k-Clustering. In *ICML*, Vol. 119. 6586–6596.
[27] Konstantinos Manolis and Evaggelia Pitoura. 2023. Modularity-Based Fairness in Community Detection. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2023, Kusadasi, Turkey, November 6-9, 2023*. ACM, 126–130.
[28] Farzan Masrour, Tyler Wilson, Heng Yan, Pang-Ning Tan, and Abdol-Hossein Esfahanian. 2020. Bursting the Filter Bubble: Fairness-Aware Network Link Prediction. In *AAAI*. AAAI Press, 841–848.
[29] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2022. A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* 54, 6 (2022), 115:1–115:35.
[30] M. E. J. Newman. 2004. Fast algorithm for detecting community structure in networks. *Phys. Rev. E* 69 (2004).
[31] Evaggelia Pitoura, Kostas Stefanidis, and Georgia Koutrika. 2022. Fairness in rankings and recommendations: an overview. *VLDB J.* 31, 3 (2022), 431–458.
[32] Akrati Saxena, George Fletcher, and Mykola Pechenizkiy. 2024. Fairsna: Algorithmic fairness in social network analysis. *Comput. Surveys* 56, 8 (2024), 1–45.
[33] Vincent A. Traag, Ludo Waltman, and Nees Jan van Eck. 2019. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports* 9, 1 (2019), 5233.
[34] Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Konstantinos Semertzidis, and Panayiotis Tsaparas. 2022. Link Recommendations for PageRank Fairness. In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*. ACM, 3541–3551.
[35] Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Panayiotis Tsaparas, Ilias Kleftakis, and Nikos Mamoulis. 2021. Fairness-Aware PageRank. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. ACM / IW3C2, 3815–3826.
[36] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018*. ACM, 1–7.
[37] Ji Wang, Ding Lu, Ian Davidson, and Zhaojun Bai. 2023. Scalable Spectral Clustering with Group Fairness Constraints. In *International Conference on Artificial Intelligence and Statistics, 25-27 April 2023, Palau de Congressos, Valencia, Spain (Proceedings of Machine Learning Research, Vol. 206)*. PMLR, 6613–6629.
[38] Yian Wang, Jian Kang, Yinglong Xia, Jiebo Luo, and Hanghang Tong. 2022. iFiG: Individually Fair Multi-view Graph Clustering. In *IEEE International Conference on Big Data, Big Data 2022, Osaka, Japan, December 17-20, 2022*. IEEE, 329–338.
[39] Reza Zafarani, Mohammad Ali Abbasi, and Huan Liu. 2014. *Social Media Mining: An Introduction*. Cambridge University Press.

A Appendix

A.1 Fairness-Aware Modularity detection

Proof of Lemma 3.1.

Proof. Let C_u be the single node community the red node u belongs to before u is moved to C_i. Moving u only affects the modularity and diversity of C_u and C_i.

Before moving u, it holds:

$$Q^R(C_u) = -\frac{1}{2m}\left(\frac{k_u^2}{2m}\right)$$

Before moving u:

$$Q^R(C_i) = \frac{1}{2m}\left(2In_i^{RR} + In_i^{RB} - \frac{K_i K_i^R}{2m}\right)$$

After moving u:

$$Q^R(C_i) = \frac{1}{2m}(2\frac{In_i^{RR} + 2\sum_{v\in C_i, v\in R} w(u,v) + In_i^{RB} + \sum_{v\in C_i, v\in B} w(u,v)}{m} - \frac{(K_i + k_u)(K_i^R + k_u)}{2m})$$

Subtracting the before from the after values, we get $\Delta Q^R_{u\to C_i}$.

For the blue modularity of C_u, before moving u, we have $Q^B(C_u)$ is 0.

Before moving u:

$$Q^B(C_i) = \frac{1}{2m}\left(2In_i^{BB} + In_i^{BR} - \frac{K_i K_i^B}{2m}\right)$$

Table 5: Communities formed by the fairness-aware Louvain using the diversity increase (Diversity-Increase method).

Network	Approach	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Pokec-g	Diversity-Increase	110,564	0.678	-0.025	-0.024	0.178	0.191
Pokec-a	Diversity-Increase	309,777	0.580	-0.454	-0.455	0.063	0.062
Deezer	Diversity-Increase	5,366	0.567	-0.063	-0.062	0.157	0.173
Facebook-g	Diversity-Increase	159	0.821	-0.154	-0.163	0.153	0.181
Facebook-c	Diversity-Increase	229	0.728	-0.617	-0.615	0.040	0.43
Twitch	Diversity-Increase	7,011	0.367	0.004	0	0.052	0.093

Table 6: Communities formed by the fairness-aware Louvain using the fairness-gain (Fairness-Gain method).

Network	Approach	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Pokec-g	Fairness-Gain	42	0.713	-0.031	-0.031	0.179	0.179
Pokec-a	Fairness-Gain	36	0.709	-0.586	-0.586	0.049	0.055
Deezer	Fairness-Gain	68	0.681	-0.102	-0.101	0.140	0.140
Facebook-g	Fairness-Gain	17	0.823	-0.166	-0.171	0.146	0.179
Facebook-c	Fairness-Gain	12	0.794	-0.688	-0.684	0.034	0.041
Twitch	Fairness-Gain	26	0.422	0.002	0.001	0.039	0.091

After moving u:

$$Q^B(C_i) = \frac{1}{2m}\left(2In_i^{BB} + In_i^{BR} + \sum_{v\in C_i, v\in B} w(u,v) - \frac{(K_i + k_u)K_i^B}{2m}\right)$$

Subtracting the before from the after values, we get $\Delta Q^B_{u\to C_i}$.
The diversity of C_u before moving u is 0.
Before moving u:

$$D^{RB}(C_i) = \frac{1}{2m}\left(In_i^{RB} - \frac{K_i^R K_i^B}{m}\right)$$

After moving u:

$$D^{RB}(C_i) = \frac{1}{2m}\left(In_i^{RB} + \sum_{v\in C_i, v\in B} w(u,v) - \frac{(K_i^R + k_u)K_i^B}{m}\right)$$

Subtracting the before from the after values, we get $\Delta D^{RB}_{u\to C_i}$. □

Merging Communities. With similar manipulations, we get the red gain $\Delta Q^R_{C_i\to C_i}$. of merging two communities, community C_i and community C_j.

Let W_{XY} be the number of edges between nodes of color X in community C_i and nodes of color Y in community C_j. Let K_i be the sum of degrees of all nodes in community C_i, K_i^X be the sum of the degrees of all node of color X in community C_i and K_i^{XY} be the sum of the Y-colored degrees of the X-colored nodes in community C_i.

$$\Delta Q^R_{C_i\to C_j} = \frac{1}{2m}\left(2W_{RR} + W_{RB} - \frac{K_i K_j^R + K_j K_i^R}{2m}\right)$$

Proof of Lemma 3.2.

Proof. Again, let C_u be the single node community of the red node. Moving u only affects the modularity and diversity of C_u and C_i.

Before moving u, it holds:

$$Q_L^R(C_u) = -\frac{1}{2m}\left(\frac{(k_u^R)^2}{2m_{RR}}\right)$$

Before moving u, it holds:

$$Q_L^R(C_i) = \frac{1}{2m}\left(2\,In_i^{RR} + In_i^{RB} - \frac{K_i^{RB}K_i^{BR}}{m_{RB}} - \frac{(K_i^{RR})^2}{2m_{RR}}\right)$$

After moving u, it holds:

$$Q_L^R(C_i) = \frac{1}{2m}((2\,In_i^{RR} + In_i^{RB} + 2\sum_{v\in C_i, v\in R} w(u,v) + \sum_{v\in C_i, v\in B} w(u,v) - \frac{(K_i^{RB} + k_u^B)K_i^{BR}}{m_{RB}} - \frac{(K_i^{RR} + k_u^R)^2}{2m_{RR}}$$

Subtracting the before from the after values, we get $\Delta Q^R_{L,u\to C_i}$.
The blue modularity $Q_L^B(C_u)$ of C_u, before moving u is 0.
Before moving u, it holds:

$$Q_L^R(C_i) = \frac{1}{2m}\left(2\,In_i^{BB} + In_i^{BR} - \frac{K_i^{BR}K_i^{RB}}{m_{RB}} - \frac{(K_i^{BB})^2}{2m_{BB}}\right)$$

After moving u, it holds:

$$Q_L^R(C_i) = \frac{1}{2m}(2\,In_i^{BB} + In_i^{BR} + \sum_{v\in C_i, v\in B} w(u,v) - \frac{K_i^{BR}(K_i^{RB} + k_u^B)}{m_{RB}} - \frac{(K_i^{BB})^2}{2m_{BB}})$$

Subtracting the before from the after values, we get $\Delta Q^B_{L,u\to C_i}$.
The diversity of C_u before moving u is 0.
Before moving u:

$$D_L^{RB}(C_i) = \frac{1}{2m}\left(In_i^{RB} - \frac{K_i^{RB}K_i^{BR}}{m_{BB}}\right)$$

Table 7: Communities formed by the fairness-aware Louvain using the labeled diversity increase (L-Diversity-Increase method).

Network	Approach	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Pokec-g	L-Diversity-Increase	140,789	0.632	-0.022	-0.021	0.162	0.181
Pokec-a	L-Diversity-Increase	461,703	0.535	-0.414	-0.415	0.047	0.055
Deezer	L-Diversity-Increase	6,079	0.542	-0.058	-0.059	0.137	0.165
Facebook-g	L-Diversity-Increase	213	0.773	-0.146	-0.151	0.128	0.171
Facebook-c	L-Diversity-Increase	1,478	0.575	-0.469	-0.471	0.034	0.040
Twitch	L-Diversity-Increase	7,070	0.381	0	-0.005	0.015	0.085

Table 8: Communities formed by the fairness-aware Louvain using the labeled fairness-gain (L-Fairness-Gain).

Network	Approach	Communities	Modularity	Unfairness	L-Unfairness	Diversity	L-Diversity
Pokec-g	L-Fairness-Gain	34	0.713	-0.031	-0.032	0.179	0.191
Pokec-a	L-Fairness-Gain	36	0.714	-0.591	-0.591	0.051	0.055
Deezer	L-Fairness-Gain	98	0.687	-0.104	-0.102	0.140	0.162
Facebook-g	L-Fairness-Gain	15	0.834	-0.167	-0.175	0.152	0.181
Facebook-c	L-Fairness-Gain	16	0.834	-0.725	-0.722	0.038	0.042
Twitch	L-Fairness-Gain	22	0.423	0.001	-0.001	0.042	0.091

After moving u:

$$D_L^{RB}(C_i) = \frac{1}{2m}\left(In_i^{RB} + \sum_{v\in C_i, v\in B} w(u,v) - \frac{(K_i^{RB} + k_u^B)K_i^{BR}}{m_{RB}}\right)$$

Subtracting the before from the after values, we get $\Delta D^{RB}_{L,u\to C_i}$. □

Merging Communities (labeled modularity). With similar manipulations, we get the labeled red gain, $\Delta Q^R_{L,C_i\to C_i}$. of merging two communities, community C_i and community C_j.

$$\Delta Q^R_{L,C_i\to C_j} = \frac{1}{2m}\left(2W_{RR} + W_{RB} - \frac{K_i^{RB}K_j^{BR} + K_i^{BR}K_j^{RB}}{m_{RB}} - \frac{K_i^{RR}K_j^{RR}}{m_{RR}}\right)$$

A.2 Additional Experiments

This section presents the experimental evaluation of various fairness-aware modifications of the Louvain algorithm on both synthetic and real-world networks. The objective of these experiments is to assess the effectiveness of these modified algorithms in improving fairness and diversity within the detected communities, while preserving the overall quality of the community structure.

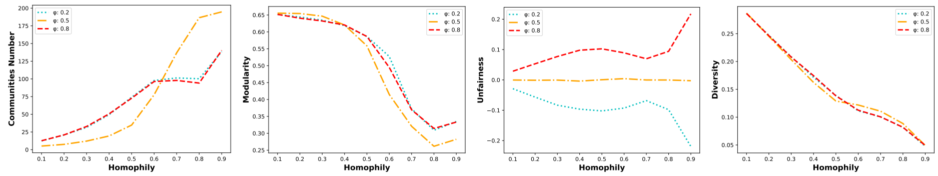
In Tables 5-8, we present results on the real datasets using additional fairness-aware Louvain algorithms.

In Figures 3 and 4 , we present results on synthetic datasets using the diversity gain and labeled diversity gain methods.

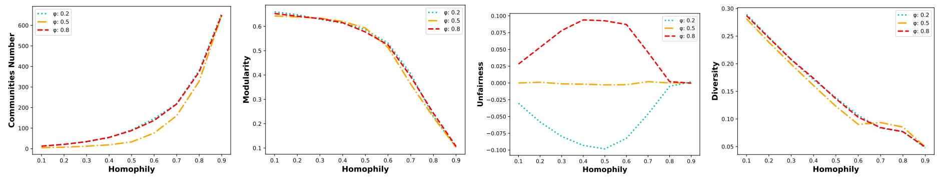
Figure 5 illustrates the trade-off between modularity and unfairness metrics for various levels of blue homophily when $\phi = 0.2$. In these figures, we plot the difference in each metric between the communities detected by the Louvain algorithm and those identified by our fairness-aware methods.

- Plots (a) and (c) display results for the communities obtained using the *Group-Increase Method* on symmetric and asymmetric datasets, respectively.
- Plots (b) and (d) show the corresponding results for the *labeled Group-Increase (L-Group-Increase) Method* .

Our findings indicate that networks with higher homophily tend to exhibit increased levels of unfairness. While our fairness-aware algorithms achieve lower unfairness compared to Louvain, this improvement comes at the expense of reduced modularity. This trade-off becomes particularly pronounced at high homophily levels, where the most significant gains in fairness are observed, but at the cost of a noticeable drop in modularity.

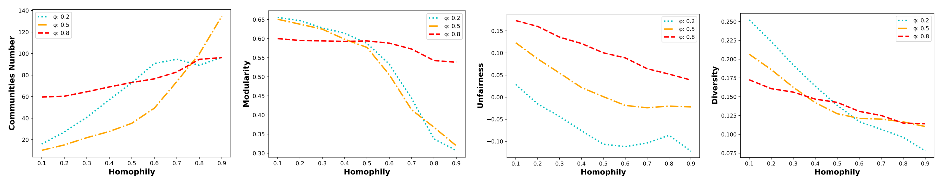


(a) Diversity-Increase Communities (b) Diversity-Increase Modularity (c) Diversity-Increase Unfairness (d) Diversity-Increase Diversity

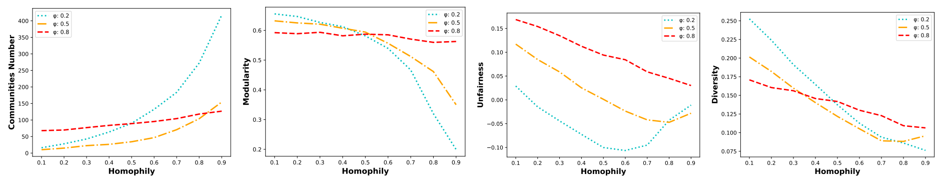


(e) L-Diversity-Increase Communities (f) L-Diversity-Increase Modularity (g) L-Diversity-Increase Unfairness (h) L-Diversity-Increase Diversity

Figure 3: Symmetric Datasets

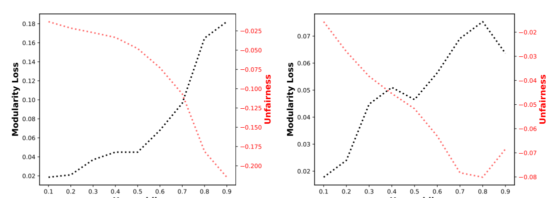


(a) Diversity-Increase Communities (b) Diversity-Increase Modularity (c) Diversity-Increase Unfairness (d) Diversity-Increase Diversity



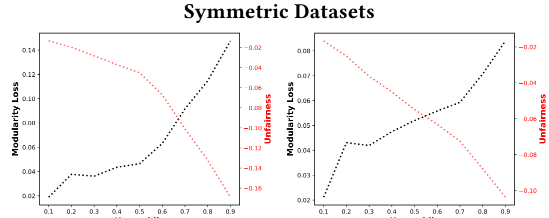
(e) L-Diversity-Increase Communities (f) L-Diversity-Increase Modularity (g) L-Diversity-Increase Unfairness (h) L-Diversity-Increase Diversity

Figure 4: Asymmetric Datasets



(a) Group-Increase (b) L-Group-Increase

Symmetric Datasets



(c) Group-Increase (d) L-Group-Increase

Asymmetric Datasets

Figure 5: Trade-off between Modularity Loss and Unfairness