Modularity-Fair Deep Community Detection

Christos Gkartzios
Department of Computer Science & Engineering
University of Ioannina
Ioannina, Greece
chgartzios@cs.uoi.gr

Evaggelia Pitoura
Department of Computer Science & Engineering
University of Ioannina
Ioannina, Greece
pitoura@uoi.gr

Panayiotis Tsaparas
Department of Computer Science & Engineering
University of Ioannina
Ioannina, Greece
tsap@uoi.gr

***Abstract*—Detecting meaningful communities in networks is essential for understanding complex social, biological, and information systems. Modularity effectively captures the quality of communities by comparing the observed and expected edge densities, but it often overlooks fairness with regards to the connectivity of different groups of nodes within the communities. In this work, we address this limitation by proposing fairness-aware community detection algorithms that incorporate group-sensitive connectivity into the modularity framework. Our approach is based on optimizing distinct sub-matrices of the modularity matrix that isolate intra-group and inter-group connections. We introduce two algorithmic families: (a) *Input-based* methods, including fair spectral and deep learning algorithms that directly operate on these sub-matrices; and (b) *Loss-based* methods, which integrate fairness-aware sub-matrix information into the learning objective of deep community detection models. Our experiments on synthetic and real-world networks demonstrate that our algorithms significantly improve group connectivity fairness without compromising community quality.**

***Index Terms*—Community Detection, Spectral Clustering, Deep Clustering, Social Networks, Fairness-aware community detection, Graph Neural Networks, Group Modularity**

I. INTRODUCTION

Networks capture relationships between entities across diverse domains, including social platforms, scientific collaboration, and citation systems. In many such networks, nodes tend to form communities, i.e., subsets of nodes that exhibit higher internal connectivity relative to the rest of the network [1], [2]. These communities play a critical role in determining how information spreads and how opinions are shaped [3], [4].

Traditional community detection algorithms aim to maximize quality, typically optimizing metrics that capture the intra-community connectivity compared to the inter-community connectivity. Modularity is a commonly used such metric. However, such algorithms often neglect fairness considerations. In many real-world networks, nodes carry sensitive attributes such as gender, age, or ethnicity, which naturally partition the network into groups. Recent research in network algorithmic fairness has emphasized the importance of equitable treatment, particularly at the group level [5]–[8]. In this work, we focus on the fairness of community detection algorithms on networks.

Prior work on group fairness in community detection has mainly focused on balanced representation, requiring that the proportion of each group within each community reflects the global distribution [9]–[11]. More recently, a structural notion of fairness based on connectivity was introduced, requiring that different groups are equally well connected within the communities [12]. For example, in a collaboration network, do women form as many intra-community connections as men? Connection strength within communities is crucial for minority groups to remain visible, influential, and well integrated into the network.

This notion was formalized through group modularity [12]. Modularity is a standard measure of community quality in networks, comparing the observed edge density within communities to the expected density under a random graph null model [13], [14]. The group modularity framework extends this idea by considering subsets of edges associated with specific groups. In addition, a diversity-based modularity measure was defined to capture inter-group connectivity. The work in [12] addressed connectivity fairness through fairness-aware variants of the Louvain algorithm [15], [16].

In this work, we extend these ideas in two directions: (a) we introduce new formulations of the problem based on decomposing the modularity matrices into sub-matrices that capture intra-group and inter-group connectivity, and (b) we propose fairness-aware algorithms that make use of these group modularity sub-matrices. Specifically, we build on Deep Modularity Networks (DMoN) [17], a GNN-based model that learns node embeddings and forms communities by optimizing a modularity-based objective, and we extend it to incorporate group modularity fairness into the learning objective. We call this approach *loss-based*. This is the first deep community detection framework that integrates group modularity directly into the training objective. In addition, we propose algorithms that directly operate on the group modularity matrices for both DMoN and spectral-based clustering. We call this approach *input-based*. The resulting communities are both structurally meaningful and balanced in terms of connectivity.

To evaluate our approach, we perform a comprehensive experimental study on both synthetic and real-world networks. Our goal is to assess whether the proposed algorithms effectively improve connectivity fairness while maintaining strong modularity. We systematically analyze trade-offs controlled by a tunable loss parameter and compare our methods against classical and fairness-aware baselines.

The contributions of this paper are threefold:

1) We revisit group modularity, by decomposing the adjacency matrix and the modularity matrix according to

group membership into submatrices that isolate intra-group and inter-group connectivity. These matrices enable fairness-aware spectral clustering and provide the foundation for fairness-aware loss functions.

2) We propose novel fairness-aware spectral and deep community detection algorithms that operate on the group-modularity matrices, and novel deep community detection algorithms that extend modularity-based GNN clustering with group-sensitive loss functions. Each algorithm targets a distinct fairness objective and enables tunable trade-offs between structural quality and group fairness.
3) We conduct extensive experiments on real and synthetic datasets, demonstrating that our methods improve fairness-modularity trade-offs compared to both classical and fairness-aware baselines.

The remainder of this paper is structured as follows. In Section II, we present the group-aware modularity framework and introduce fairness-sensitive variants of the modularity matrix. Section III presents our proposed fairness-aware community detection algorithms. Section IV provides the experimental evaluation. Section V reviews related work, and Section VI concludes the paper.

II. Group-Aware Modularity Matrices

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 are partitioned into groups, defined by the value of a sensitive attribute. For simplicity, we assume a binary attribute with two values: red and blue. The red group, denoted by $V_R \subseteq V$, contains the nodes with the red value. The blue group, denoted by $V_B \subseteq V$, contains the remaining nodes, such that $V_R \cup V_B = V$ and $V_R \cap V_B = \emptyset$.

Most prior work on group fairness in community detection emphasizes node-level balance, aiming to ensure that each community reflects the global group distribution [9], [11]. However, since network dynamics such as information flow and influence propagation are driven by edge interactions [3], [4], recently, there is work on defining fairness from a connectivity perspective [12], [18]. In this work, we adopt the modularity-based framework introduced in [12] that evaluates fairness in terms of how well each group is connected within the detected communities, using *modularity*. This approach defines group-aware variants of the popular modularity measure, introducing the concepts of group-modularity, modularity unfairness and modularity diversity. We will now reformulate these group modularity metrics, through a decomposition of the graph adjacency matrix, and the modularity matrix.

Let $A \in \mathbb{R}^{n \times n}$ denote the adjacency matrix of the graph G. We partition A into four disjoint sub-matrices:

$$A = \begin{bmatrix} A_{RR} & A_{RB} \\ A_{BR} & A_{BB} \end{bmatrix}$$

where

- A_{RR}: edges between Red nodes,
- A_{RB}: edges from Red nodes to Blue nodes,
- A_{BR}: edges from Blue nodes to Red nodes, and
- A_{BB}: edges between Blue nodes.

Since the graph is undirected, $A_{RB} = A_{BR}^\top$.

We now define the sub-matrices A_R, A_B, and A_{div} as follows:

$$A_R = \begin{bmatrix} A_{RR} & A_{RB} \\ A_{BR} & \mathbf{0} \end{bmatrix} \quad A_B = \begin{bmatrix} \mathbf{0} & A_{RB} \\ A_{BR} & A_{BB} \end{bmatrix} \quad A_{\text{div}} = \begin{bmatrix} \mathbf{0} & A_{RB} \\ A_{BR} & \mathbf{0} \end{bmatrix}$$

where A_R, A_B, and A_{div} correspond to edges incident to Red nodes, edges incident to Blue nodes, and inter-group edges, respectively.

Given these matrices, we can define the corresponding subgraphs G_R, G_B and G_{div}. We will use these matrices to decompose the modularity matrix, and define the clustering objectives that we use throughout this work.

Modularity is a commonly used metric for evaluating the quality of a community. For a community $C \subseteq V$, modularity is defined as

$$Q(C) = \frac{1}{2m} \sum_{u,v \in C} \left(A_{uv} - \frac{d_u d_v}{2m} \right)$$

where d_u and d_v are the degrees of nodes u, v and the sum is over all pairs of nodes in the community C. Modularity compares the density of edges in the community C to the expected number of edges in a random graph where edges are generated at random, while preserving the degrees. If $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ is a partition of the nodes in V into k communities, the modularity of the partition is defined as $Q(\mathcal{C}) = \sum_{C_j \in \mathcal{C}} Q(C_j)$.

Classical modularity optimization builds upon the *modularity matrix*, introduced by Newman [19]:

$$B = A - \frac{dd^\top}{2m}$$

where A is the adjacency matrix, d is the degree vector with d_i being the degree of node i, and m is the number of edges in the graph.

The modularity score for the partition $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$ can be computed using the modularity matrix B. Let $S \in \{0,1\}^{n \times k}$ be the binary community assignment matrix, where $S_{ij} = 1$ if node $i \in C_j$, and 0 otherwise. Then, the modularity of the partition $\mathcal{C}$ defined by the assignment matrix S is given by:

$$Q(\mathcal{S}) = \frac{1}{2m} \operatorname{Tr}(S^\top B S) \tag{1}$$

where $\operatorname{Tr}(\cdot)$ denotes the matrix trace [19].

We can now use the decomposition of the adjacency matrix to define the group-aware variants of modularity defined in [12], by decomposing the modularity matrix. Specifically, we define the *red modularity matrix* B_R using the red adjacency matrix A_R as follows:

$$B_R = A_R - \frac{d_R d_R^\top}{2m_R}$$

where d_R is the degree vector for the graph G_R and m_R is the number of edges in the graph G_R. The modularity score for red group connectivity, denoted Q_{R}, is then given by:

$$Q_{\text{R}}(S) = \frac{1}{2m}\operatorname{Tr}(S^\top B_R S). \tag{2}$$

Analogously, we define the *blue modularity matrix* B_B using the blue adjacency matrix A_B and the degree vector d_B of the graph G_B. We also define the *diversity modularity matrix* B_{div} using the matrix A_{div} and the corresponding degree vector d_{div}:

$$B_{\text{div}} = A_{\text{div}} - \frac{d_{\text{div}} d_{\text{div}}^\top}{2m_{\text{div}}}$$

$$Q_{\text{div}}(S) = \frac{1}{2m}\operatorname{Tr}(S^\top B_{\text{div}} S). \tag{3}$$

Building on the group-sensitive modularity formulations, we explore their spectral properties and demonstrate that these objectives can be effectively approximated through spectral embedding of the respective modularity matrix. This extends the spectral modularity approach of Newman [19] to the group-aware setting.

Proposition 1. *Let B_R be the red modularity matrix defined as above. Then the optimization of the red modularity*

$$\max_S Q_R(S) = \frac{1}{2m}\operatorname{Tr}(S^\top B_R S)$$

can be approximated by computing the top-k eigenvectors of B_R and applying k-means clustering in the resulting spectral space.

Proof (sketch). We begin with the case of two communities.

Let $\mathbf{s} \in \{-1, +1\}^n$ be a vector indicating a bipartition of the nodes, with $s_i = +1$ if node i belongs to community 1, and $s_i = -1$ otherwise. Then, the red modularity objective becomes:

$$Q_{\text{R}}(\mathbf{s}) = \frac{1}{4m}\mathbf{s}^\top B_R \mathbf{s}.$$

Maximizing this expression over discrete vectors $\mathbf{s} \in \{-1, +1\}^n$ is computationally difficult.

We instead consider the problem over the vectors $\mathbf{s} \in \mathbb{R}^n$ with $\mathbf{s}^\top \mathbf{s} = 1$, yielding the continuous problem:

$$\max_{\mathbf{s}^\top \mathbf{s}=1} \mathbf{s}^\top B_R \mathbf{s}.$$

Note that since A_R is symmetric and real, so is B_R, and all eigenvalues are real.

The objective is maximized when $\mathbf{s}$ is the eigenvector corresponding to the largest eigenvalue of B_R. Let $\mathbf{u}_1$ be this leading eigenvector. Then:

$$\max_{\mathbf{s}^\top \mathbf{s}=1} \mathbf{s}^\top B_R \mathbf{s} = \mathbf{u}_1^\top B_R \mathbf{u}_1.$$

To obtain a discrete partition, we assign nodes based on the sign of the corresponding entry in $\mathbf{u}_1$:

$$s_i = \begin{cases} +1 & \text{if } (\mathbf{u}_1)_i \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

This yields a bipartition that approximately maximizes Q_{R}.

For multiple communities, let $\{\mathbf{u}_1, \ldots, \mathbf{u}_p\}$ be the top p eigenvectors of B_R, and form the spectral embedding matrix:

$$Z = [\mathbf{u}_1 \mid \ldots \mid \mathbf{u}_p] \in \mathbb{R}^{n \times p}$$

We apply k-means clustering on the rows of Z, assigning each node to a community. Let S be the resulting community matrix. Then, the clustering approximately maximizes the red modularity objective:

$$Q_{\text{R}}(S) = \frac{1}{2m}\operatorname{Tr}(S^\top B_R S).$$

Thus, spectral clustering on the red modularity matrix approximates the maximization of red modularity. □

These formulations allow us to measure and optimize modularity with respect to both group-specific and inter-group connectivity patterns. By incorporating group constraints directly into the adjacency structure, our modularity matrix variants enable fairness-aware spectral optimization, and fairness-aware loss functions.

III. Fairness-Aware Community Detection

We now describe our algorithms for optimizing group modularity. We consider two classes of algorithms. Spectral community detection algorithms that make use of the eigenvectors of the modularity matrices, and deep community detection algorithms that use the modularity matrices to redefine the loss function.

A. Input-Based Fair Spectral Community Detection

Spectral community detection using the modularity matrix B was first introduced in [19]. The approach is similar to that of spectral clustering using the Laplacian matrix. The algorithm computes the top-k eigenvectors of the matrix B, which are viewed as a continuous approximation of the discrete binary assignment matrix S. These vectors are then used to obtain the communities, typically by applying k-means clustering on the extracted k-dimensional vectors of the nodes.

To enable fairness-aware spectral clustering, we define a modified modularity matrix:

$$B_X^{(\lambda)} = (1-\lambda)B + \lambda B_X$$

where B is the standard modularity matrix, and $B_X \in \{B_R, B_B, B_{\text{div}}\}$ is a fairness-aware modularity matrix chosen according to the fairness criterion (e.g., group or diversity), with the parameter $\lambda \in [0, 1]$ controlling the emphasis on structural versus fairness-aware connectivity.

Specifically, we propose two new spectral community detection algorithms that rely on the different modularity matrices we have defined:

GroupSpectral: The algorithm assumes a protected group, usually the minority one, for which we want to achieve strong internal connectivity. It uses the modified modularity matrix $B_X^{(\lambda)} = (1-\lambda)B + \lambda B_X$, where $B_X \in \{B_R, B_B\}$ is the group modularity matrix corresponding to the selected protected group (red or blue). Then it applies the spectral clustering process: It extracts the k largest eigenvectors of $B_X^{(\lambda)}$ and performs k-means clustering to obtain the communities. The goal is to discover communities that simultaneously preserve structural quality and enhance group connectivity for the protected group.

DIVERSITYSPECTRAL: This algorithm uses $B_{\text{div}}^{(\lambda)} = (1 - \lambda)B + \lambda B_{\text{div}}$ matrix to extract the eigenvectors, and as before applies k-means to obtain the communities. The goal of the algorithm is to obtain communities with high diversity modularity.

B. Input-Based Fair Deep Community Detection

For this class of algorithms, we extend the Deep Modularity Network (DMoN) framework introduced in [17] to incorporate fairness-aware objectives. DMoN uses a Graph Convolutional Network (GCN) on the normalized adjacency matrix A to obtain k-dimensional node embeddings. Then it applies soft-max on the embeddings to obtain a soft assignment matrix S of the nodes to clusters. This soft assignment matrix is used to define the loss function $\mathcal{L}_{\text{DMoN}}$, which is defined as follows:

$$\mathcal{L}_{\text{DMoN}} = -\frac{1}{2m}\operatorname{Tr}(S^\top B S) + \gamma \mathcal{R}_{\text{collapse}}.$$

The first term corresponds to the modularity Q, while the second term is a regularization term that is defined as:

$$\mathcal{R}_{\text{collapse}} = \left(\frac{\sqrt{k}}{n} \|S^\top\|_F - 1 \right).$$

The regularization term discourages degenerate clustering solutions, such as assigning all nodes to a single community.

Building on the fairness-aware modularity formulations used in the spectral setting, we extend this idea to the deep clustering framework by modifying the input adjacency matrix. Specifically, we define:

$$A_X^{(\lambda)} = (1 - \lambda)A + \lambda A_X$$

where $A_X \in \{A_R, A_B, A_{\text{div}}\}$ is selected based on the desired connectivity objective, either group or diversity based. This modification retains the global structure of the graph while increasing the influence of A_X. The parameter $\lambda \in [0, 1]$ controls the trade-off between preserving global structure and increasing the influence of group-based connectivity.

As before we propose two variants relying on different modularity matrices.

GROUPDMON: This algorithm promotes strong connectivity within a protected group by using the modified adjacency matrix $A_X^{(\lambda)} = (1-\lambda)A + \lambda A_X$, where $A_X \in \{A_R, A_B\}$ retains only edges involving nodes from the target group. The model is trained on $A_X^{(\lambda)}$ and learns cluster assignments by optimizing a trade-off between structural and group modularity, with A_X contributing to the modularity objective for the target group.

DIVERSITYDMON: This algorithm promotes diversity by using the modified adjacency matrix $A_{\text{div}}^{(\lambda)} = (1-\lambda)A + \lambda A_{\text{div}}$, which retains only red-blue edges. The model is trained on $A_{\text{div}}^{(\lambda)}$ and learns cluster assignments by optimizing a trade-off between structural and diversity-based modularity, with A_{div} contributing to the diversity objective.

TABLE I: Summary of the proposed algorithms.

Method	Type	Objective
GROUPSPECTRAL	Input-based, Spectral	Group modularity
DIVERSITYSPECTRAL	Input-based, Spectral	Diversity
GROUPDMON	Input-based, Deep	Group modularity
DIVERSITYDMON	Input-based, Deep	Diversity
DEEPGROUP	Loss-based, Deep	Group modularity
DEEPDIVERSITY	Loss-based, Deep	Diversity
DEEPFAIRNESS	Loss-based, Deep	Fairness

Algorithm 1 Fairness-Aware Community Detection

Input: Graph $G = (V, E)$; number of clusters k; method M; fairness objective $\mathcal{F}$
Output: Community assignments $\{c_1, \ldots, c_n\}$
Construct fairness-aware matrices based on $\mathcal{F}$
Compute node representations H via M (spectral for Spectral, GNN for Deep)
Cluster nodes into k communities using H
return Community assignments $\{c_1, \ldots, c_n\}$

C. Loss-Based Fair Deep Community Detection

The input-based algorithms enforce fairness by modifying the input adjacency matrix, which influences the learned node representations. We now present an alternative strategy that incorporates fairness directly into the loss function, without changing the input graph structure. We define three such fairness-aware deep community detection algorithms, each based on a distinct group modularity objective.

DEEPGROUP: The algorithm enhances the connectivity of a particular group (red or blue) by increasing its group modularity Q_X, where $Q_X \in \{Q_R, Q_B\}$, in addition to the overall modularity. Therefore, the loss function is:

$$\mathcal{L}_{\text{DEEPGROUP}} = -\lambda\, Q_X \;-\; (1-\lambda)Q \;+\; \gamma\, \mathcal{R}_{\text{collapse}}$$

The parameter λ controls the tradeoff between modularity and group modularity.

DEEPDIVERSITY: The algorithm promotes the formation of communities with high diversity within the communities, maximizing diversity modularity Q_{div} in addition to the overall modularity. Therefore, the loss function is defined as:

$$\mathcal{L}_{\text{DEEPDIVERSITY}} = -\lambda\, Q_{\text{div}} \;-\; (1-\lambda)Q \;+\; \gamma\, \mathcal{R}_{\text{collapse}}$$

where λ is a parameter that controls the tradeoff between modularity and diversity.

In addition to enhancing intra-group connectivity or diversity, we propose a third approach that directly incorporates fairness into the loss function. According to [12], unfairness is defined as the difference between the group modularity scores, *Unfairness* $= Q_R - Q_B$. A network partition is considered fair when this difference is close to zero, indicating that

TABLE III: Real dataset characteristics. $|R|, |B|$: red and blue group sizes; $\bar{d}_X$: average degree in the graph G_X; p_h^R, p_h^B: red, blue homophily.

Network	Nodes	Edges	Attribute	$\|R\|$	$\|B\|$	$\bar{d}_R$	$\bar{d}_B$	$\bar{d}_{div}$	p_h^R	p_h^B	ρ
Deezer	28,281	92,752	Gender	12,538	15,743	6.34	6.73	2.79	0.972	1.07	0.443
Facebook-g	4,039	88,234	Gender	1,533	2,506	45.75	42.42	15.45	1.236	0.995	0.378
Facebook-c	4,039	88,234	Education	367	3,672	31.38	44.92	2.54	1.481	1.066	0.090
Twitch	168,114	6,797,557	Maturity	79,033	89,081	88.25	74.31	34.69	1.292	0.924	0.470

both groups are equally well connected within the discovered communities.

DEEPFAIRNESS: The algorithm enforces group-level modularity balance by minimizing the difference between red and blue group modularities, in addition to maximizing overall modularity. Therefore, the loss function is defined as:

$$\mathcal{L}_{\text{DEEPFAIRNESS}} = -Q \; + \; \phi \, |Q_{\text{R}} - Q_{\text{B}}| \; + \; \gamma \, \mathcal{R}_{\text{collapse}}$$

where ϕ is a parameter that controls the importance of the fairness term in the loss function.

All spectral algorithms have complexity $\mathcal{O}(mk)$ for sparse graphs [20]. All deep algorithms retain DMoN's per-epoch complexity, $\mathcal{O}(k^2 n + m)$ [17].

Algorithm 1 gives the outline of our fairness-aware community detection framework, while Table I summarizes the proposed algorithms.

IV. EVALUATION

We evaluate the proposed fairness-aware community detection methods across a range of real and synthetic networks. Our goal is to understand how different fairness objectives, such as improving minority group connectivity, increasing inter-group diversity, or reducing group modularity disparities, affect the resulting community structures and their trade-offs with structural quality.

We analyze (i) the impact of the parameters λ and ϕ on fairness and modularity, (ii) the relative performance of deep and spectral methods compared to traditional and fairness-aware baselines, and (iii) the extent to which the discovered communities reflect the intended fairness optimization goals. The code is available on GitHub[1].

A. Datasets

1) Synthetic Datasets: To systematically examine how structural and attribute-driven biases influence fairness in communities, we construct synthetic networks using a variant of the stochastic block model (SBM) [11], [12], [21]. Each node is assigned a color (red or blue), and communities are formed according to parameters that regulate group balance, homophily, and structural cohesion as shown below:

- ρ: the proportion of red nodes in the network, controlling group size imbalance;
- p_c: the probability of intra-community edges, controlling the strength of community structure;
- p_h^R (p_h^B): the probability that red (blue) nodes of the same color connect, controlling homophily.

TABLE II: Synthetic Dataset Parameters

Parameter	Description	Value
N	Number of nodes	1,000
ρ	Fraction of red nodes	0.2
ℓ	Edges per node	5
k	Initial number of communities	5
p_c	Intra-community edge probability	0.9
p_h^R, p_h^B	Red, Blue Homophily	0.5, 0.9

The default settings for our synthetic data are shown in Table II. We introduce group imbalance by setting the red node proportion to $\rho = 0.2$, making the red group the minority. To model structural bias, we assign higher homophily to the blue group ($p_h^B = 0.9$) than to the red group ($p_h^R = 0.5$). This configuration results in one group forming denser intra-group connections, leading to disparities in connectivity across communities. For each experiment, we generate 10 random datasets and report the results averaged over them.

2) Real Datasets: We evaluate our methods on three real-world networks, also used in prior work [12]. Each dataset contains node-level sensitive attributes, such as gender, which we use to define red and blue groups. The datasets are:

- **Deezer**[2]: A social network of users from European countries that mutually follow each other on the Deezer music platform. We use the gender attribute to define groups.
- **Facebook**[3]: A friendship social network derived from ego-network graphs from Facebook. We use the gender attribute (**Facebook-g**) and the academic concentration attribute (**Facebook-c**) to define groups.
- **Twitch**[4]: A mutual-follow graph of Twitch users. The sensitive attribute is the maturity level of the user's stream content.

We summarize the dataset characteristics in Table III, including the number of nodes and edges, group size imbalance (ρ), and homophily. We quantify homophily separately for red and blue groups. The red homophily score (p_h^R) is defined as the ratio of observed red-red edges to their expected count under random mixing, estimated by ρ^2. Similarly, blue homophily (p_h^B) measures the ratio of blue-blue edges, normalized by $(1-\rho)^2$. Values above 1 suggest strong within-group connectivity (homophily), while values below 1 reflect a tendency toward inter-group connections (heterophily). We

[1] https://github.com/gartzis/Modularity-Fair-Deep-Community-Detection/tree/main

[2] https://snap.stanford.edu/data/feather-deezer-social.html

[3] http://snap.stanford.edu/data/ego-Facebook.html

[4] https://snap.stanford.edu/data/twitch_gamers.html

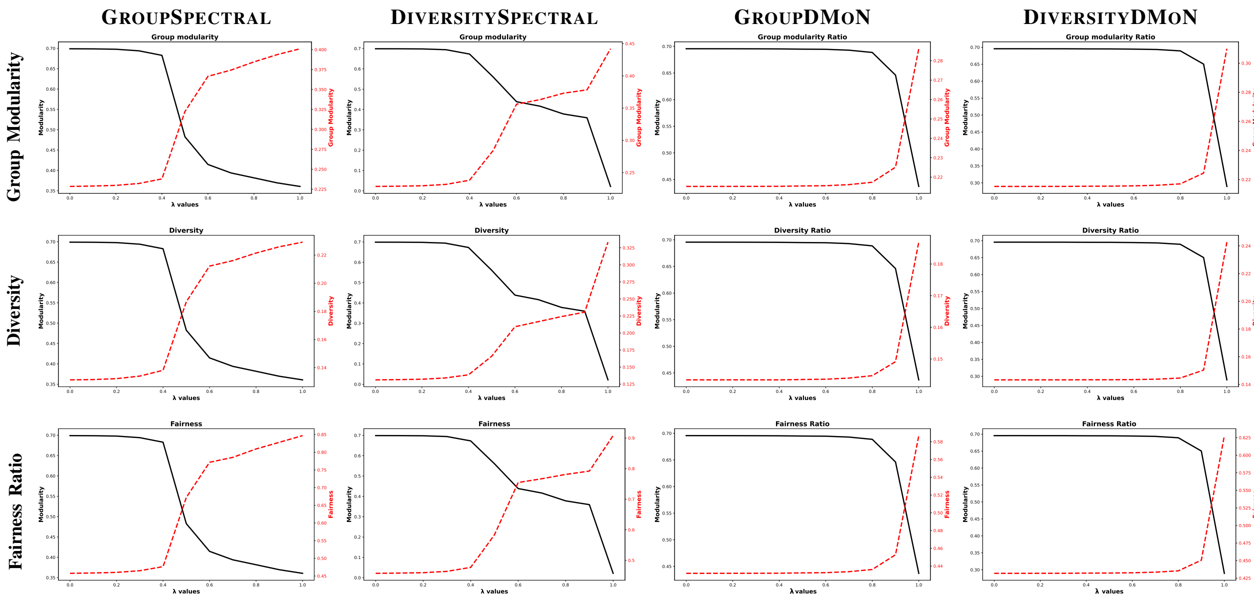

Fig. 1: Trade-offs curves: Modularity vs fairness metrics under varying λ (input-based algorithms).

focus specifically on the red group, which represents the minority group in all datasets.

B. Evaluation Metrics

To evaluate the quality and fairness of the detected communities, we consider the following metrics:

- **Modularity:** Defined as in Eq. (1) that measures the structural quality of the partitions.
- **Group Modularity:** Measures how well nodes from a specific group are connected within the communities. We focus on the red group and use Eq. (2).
- **Diversity:** Defined as in Eq. (3) that evaluates the inter-group connectivity.
- **Fairness Ratio:** Defined as

$$f_L = 1 - \left| \frac{Q_R - Q_B}{Q} \right|.$$

This metric adjusts the unfairness by the overall modularity, producing a normalized score in $[0, 1]$, where higher values indicate fairer group connectivity relative to the structural quality of the clustering.

These metrics assess both structural cohesion and group-level fairness of the detected communities. We also report all group-aware connectivity metrics as ratios with respect to modularity. This strategy allows us to evaluate fairness in relation to the overall structural quality of the communities. In some cases, modularity may decrease due to fairness constraints, which can also reduce the absolute values of group modularity or diversity. However, their ratio to total modularity may increase, indicating that a larger proportion of the network structure reflects fairness-aware connectivity. Likewise, a decrease in absolute unfairness may not always imply better fairness if modularity declines more sharply, highlighting the importance of using relative measures such as the fairness ratio.

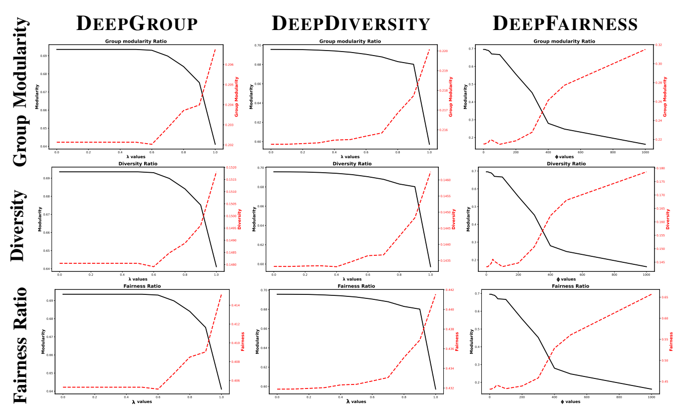

Fig. 2: Trade-offs curves: Modularity vs fairness metrics under varying λ, or ϕ (loss-based algorithms).

C. Algorithms

We compare our methods with the following baselines:

- Louvain [15]: A modularity-based method that greedily merges nodes into communities.
- Spectral [19]: A classical method that clusters nodes based on the eigenvectors of the modularity matrix.
- DMoN [17]: A deep clustering method that uses a GNN and optimizes modularity.
- GroupLouvain [12]: A fairness-aware variant of Louvain that considers both modularity and group modularity. Specifically, communities are merged only if the change improves both the overall group modularity and the modularity of the minority group.

- BALANCESPECTRAL [11]: A spectral clustering method that includes linear constraints to ensure that each community contains a balanced mix of groups.

For the algorithms that require the number of communities as input, including our algorithms, we estimate it using eigenvalue gap analysis [22] of the modularity matrix. All deep models are trained using the same configuration as DMoN [17].

D. Evaluation Results

1) Fairness metrics - modularity tradeoff: In the first experiment, we consider the proposed community detection algorithms, and we study the tradeoff between cluster quality, as captured by modularity, and the modularity fairness metrics (i.e., group modularity, diversity, fairness), as we vary the parameter λ, or ϕ. For this experiment, we generate synthetic datasets, using the parameters in Table II that correspond to a setting that is structurally unfair towards the red group. The results are shown in Figures 1, 2, where each column corresponds to one of the fair algorithms, and each row to one of the modularity fairness metrics. In all plots, modularity is shown on the left y-axis (marked with the solid black line in the plots), and the fairness metric on the right y-axis (marked with the dashed red line).

As expected, we observe a consistent trade-off between modularity and fairness as the trade-off parameter increases. As each algorithm shifts toward its objective, fairness-to-modularity ratios increase, while modularity typically declines, reflecting the structural compromises required to achieve group-level equity.

We observe that the spectral algorithms (Fig. 1, left) demonstrate a sharp trade-off. As λ increases both the group modularity and diversity metrics improve, but this comes at the cost of a steep decline in modularity, especially in DIVERSITYSPECTRAL where modularity collapses to zero at $\lambda = 1$. This outcome reflects the strict nature of the fairness-aware matrices, which prioritize their respective connectivity objective (group modularity or diversity) over preserving structural quality. In particular, in the case of networks with high homophily (Fig. 3, at $p_h^R = 0.9$), the diversity-based matrices become sparse, making it difficult to form meaningful communities, especially at $\lambda = 1$. In contrast, the DMON-based variants GROUPDMON, DIVERSITYDMON (Fig. 1, right) preserve modularity across λ including at high values where spectral methods fail. However, improvements in group modularity and diversity in these models only become evident at higher values of λ (typically above 0.8). This suggests that although the DMON-based models achieve more modest fairness improvements, they maintain higher structural quality (0.45 and 0.3 modularity respectively at $\lambda = 1$).

The loss-based deep clustering methods shown in Figure 2, optimize their respective connectivity metric through the loss function in contrast to the SPECTRAL and DMON-based models, which rely on fairness-aware modifications to the modularity matrix. We first focus on DEEPGROUP and DEEPDIVERSITY, which share the same connectivity objectives as

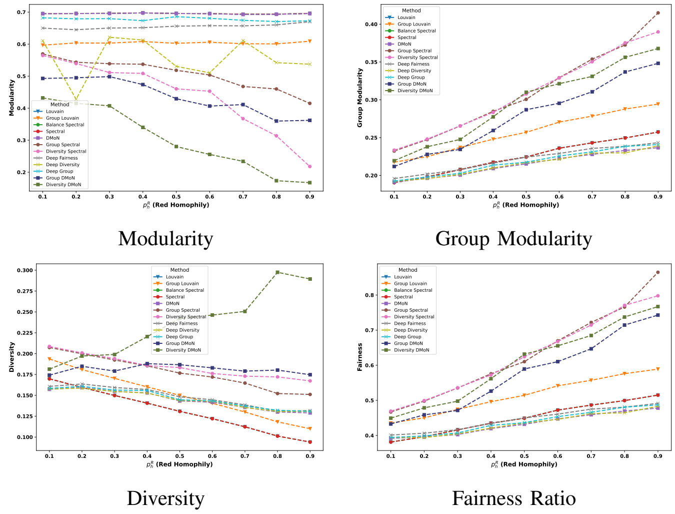

Fig. 3: Comparison of methods across red group homophily levels.

GROUPSPECTRAL, DIVERSITYSPECTRAL, GROUPDMON, and DIVERSITYDMON. Compared to the spectral methods, the loss-based variants maintain significantly higher modularity across the full range of λ, while achieving more modest improvements in their respective connectivity metric, visible in the smaller scale of the right y-axis. DEEPFAIRNESS directly targets fairness through a fairness-specific loss term, weighted by the parameter ϕ. Among all deep models, it achieves the highest fairness score (Fig. 2, row 3, column 3), approximately 0.65, but this comes at the cost of a steady decline in modularity. However, both GROUPSPECTRAL and DIVERSITYSPECTRAL achieve superior fairness performance (Fig. 1 row 3, cols. 1 and 2), and in some cases, retain higher modularity. These results suggest that although DEEPFAIRNESS fulfills its design objective, input-based spectral methods yield superior fairness outcomes by altering the modularity matrix, which can result in less stable clustering outcomes.

These findings confirm that λ and ϕ serve as tunable hyperparameters that enable flexible control over the trade-off between structural quality and different fairness goals. Furthermore, the input-based algorithms are more effective at improving their respective fairness objective.

2) Algorithm Comparison: We evaluate our algorithms and compare against baselines on both synthetic and real datasets. We use the synthetic datasets to understand the role of homophily. We use the parameters in Table II, keeping the blue group strongly homophilous ($p_h^B = 0.9$) but varying the homophily of the red group, with p_h^R taking values in $[0.1, 0.9]$ (heterophilous to homophilous). For each algorithm, we set the trade-off parameter $\lambda = 1$ to isolate the effect of optimizing the respective connectivity objective. For DEEPFAIRNESS, we use $\phi = 200$, a value chosen to produce visible fairness improvements without causing modularity to collapse. Figure 3 shows our results for the four different metrics.

Note that as the homophily of the red group increases, we expect that the group modularity also increases. On the other

TABLE IV: Communities formed by different approaches.

Method	Communities	Modularity	Group Modularity	Diversity	Fairness Ratio
Facebook-g					
LOUVAIN	16	**0.834**	0.389	0.217	0.790
GROUPLOUVAIN	17	0.830	0.390	0.219	0.797
SPECTRAL	31	0.784	0.392	0.218	0.799
BALANCESPECTRAL	31	0.722	0.373	0.211	0.759
GROUPSPECTRAL	31	0.750	**0.411**	0.226	**0.836**
DIVERSITYSPECTRAL	31	0.770	0.400	0.224	0.814
DMON	31	0.755	0.392	0.234	0.788
GROUPDMON	31	0.714	0.398	**0.237**	0.799
DIVERSITYDMON	31	0.730	0.395	0.236	0.794
DEEPGROUP	26.6	0.125	0.408	0.232	0.816
DEEPDIVERSITY	27.9	0.253	0.399	**0.237**	0.806
DEEPFAIRNESS	29.1	0.633	0.400	**0.237**	0.801
Facebook-c					
LOUVAIN	16	**0.834**	0.061	0.050	0.134
GROUPLOUVAIN	21	0.822	0.063	0.052	0.134
SPECTRAL	31	0.784	0.060	0.050	0.131
BALANCESPECTRAL	31	0.722	0.066	0.054	0.148
GROUPSPECTRAL	31	0.538	0.085	0.067	0.200
DIVERSITYSPECTRAL	31	0.491	**0.092**	0.073	**0.219**
DMON	31	0.758	0.077	0.069	0.164
GROUPDMON	31	0.477	0.084	**0.075**	0.179
DIVERSITYDMON	31	0.486	0.084	0.074	0.160
DEEPGROUP	30.3	0.223	0.085	0.072	0.171
DEEPDIVERSITY	30.6	0.514	0.078	0.070	0.164
DEEPFAIRNESS	25.8	0.729	0.079	**0.075**	0.169
Deezer					
LOUVAIN	89	**0.683**	0.424	0.234	0.779
GROUPLOUVAIN	323	0.649	0.453	0.252	0.909
SPECTRAL	25	0.358	0.382	0.234	0.774
BALANCESPECTRAL	25	0.362	0.383	0.234	0.776
GROUPSPECTRAL	25	0.011	0.472	0.397	0.947
DIVERSITYSPECTRAL	25	0.008	0.483	0.447	0.967
DMON	25	0.593	0.440	0.244	0.880
GROUPDMON	25	0.489	0.442	0.243	0.882
DIVERSITYDMON	25	0.452	0.438	0.243	0.877
DEEPGROUP	25	0.495	0.442	0.244	0.887
DEEPDIVERSITY	25	0.470	0.440	0.245	0.881
DEEPFAIRNESS	25	0.469	0.439	0.243	0.883
Twitch					
LOUVAIN	23	**0.420**	0.480	0.214	0.990
GROUPLOUVAIN	526	0.384	0.458	0.218	0.961
SPECTRAL	55	0.141	0.460	0.198	0.937
BALANCESPECTRAL	55	0	0	0	0
GROUPSPECTRAL	55	0.218	0.496	0.228	**0.998**
DIVERSITYSPECTRAL	55	0.123	0.495	0.223	0.994
DMON	55	0.301	0.501	0.245	0.993
GROUPDMON	55	0.268	**0.511**	**0.250**	0.986
DIVERSITYDMON	55	0.251	0.502	0.247	0.993
DEEPGROUP	51	0.022	0.500	0.272	1.0
DEEPDIVERSITY	40	0.020	0.500	0.250	1.0
DEEPFAIRNESS	55	0.272	0.500	0.246	0.997

hand, diversity is expected to decrease as both groups become strongly homophilous.

With respect to modularity, we observe that as expected, the baseline algorithms (LOUVAIN, SPECTRAL, DMON) achieve the best modularity. However, this comes at the cost of lower performance in fairness-related metrics. In terms of group modularity, the baseline methods show only moderate improvement as the red group becomes more homophilous. Although the red group modularity does increase slightly (more pronounced in LOUVAIN and SPECTRAL), the gain is limited. As the red group homophily increases, diversity exhibits a clear decreasing trend, especially in LOUVAIN and SPECTRAL, which form increasingly homogeneous communities as intra-group connectivity strengthens. This is expected as these methods optimize modularity without considering group connectivity, they tend to reinforce structural homophily, resulting in less inter-group mixing. DMON also shows a reduction in diversity, though the effect is less pronounced. In terms of fairness the baselines only show slight improvement as red group homophily increases. This trend is likely the result of increasing similarity in the connectivity of the two groups, rather than any fairness-aware behavior of the algorithms themselves.

Unlike traditional baselines, GROUPLOUVAIN and BALANCESPECTRAL integrate fairness into the community detection process. In terms of modularity, both methods underperform relative to their non-fairness-aware counterparts. GROUPLOUVAIN in particular shows a consistent decline in modularity, as it frequently generates a large number of small communities. This behavior is also observed in real datasets (Table. IV), where the method often produces hundreds of communities (Deezer 321, Twitch 526), reducing interpretability and structural coherence. BALANCESPECTRAL, by contrast, maintains modularity levels similar to SPECTRAL, indicating that its balancing objective has limited impact on the underlying clustering structure. Despite lower modularity, GROUPLOUVAIN achieves the highest group modularity and fairness among all baselines, particularly as the red group homophily increases. However, its performance in diversity is limited, especially in highly homophilous settings, where it tends to separate groups more strictly, reducing inter-group mixing. BALANCESPECTRAL, while promoting balanced attribute distributions, does not explicitly optimize group connectivity and therefore shows no substantial improvements in group modularity, diversity, or fairness. In terms of the fairness metric, GROUPLOUVAIN remains the strongest among the baselines, while BALANCESPECTRAL yields only marginal gains, reflecting its limited sensitivity to group-level structural disparities.

The trends observed in synthetic experiments largely carry over to real-world networks (Table IV). Traditional baselines retain high modularity but perform poorly in fairness (LOUVAIN: Facebook-c 0.134, Deezer, 0.779) and diversity (SPECTRAL: Facebook-g 0.218, Twitch 0.198). GROUPLOUVAIN shows strong fairness (Deezer, 0.909) and group modularity (Deezer, 0.453), but often produces hundreds of small communities, limiting interpretability. In contrast, our spectral and deep models offer better trade-offs between fairness and structure. GROUPSPECTRAL and DIVERSITYSPECTRAL achieve high fairness (Facebook-g, 0.836) and diversity (Facebook-c, 0.073) scores while maintaining a reasonable number of communities, though their modularity drops in datasets like Deezer and Twitch. This is most likely due to strong homophily and limited diversity in these networks, which may lead fairness-aware methods to override natural community boundaries. Deep models (GROUPDMON, DIVERSITYDMON, DEEPFAIRNESS) retain stronger modularity and competitive fairness. DEEPGROUP and DEEPDIVERSITY achieve their targeted objectives but are outperformed by spectral methods in fairness and overall trade-offs.

In summary, the traditional baselines (SPECTRAL. LOUVAIN, DMON) achieve good modularity but perform poorly on all other metrics. Among the fairness-aware baselines, BALANCESPECTRAL, which enforces balance on the group representation, does not achieve our modularity-specific objectives. GROUPLOUVAIN either performs worse compared to our algorithms or it produces a large number of communities. All proposed algorithms succeed in optimizing their respective objective, with the input-based ones generally being more effective. The results also depend on the network characteristics, for example, in highly homophilous networks, it is easier to achieve high group modularity but harder to improve diversity.

3) Consistency Across Objectives: Each model performs the best on the specific metric it is designed to optimize. At the same time, we often observe gains in secondary metrics. For instance, DEEPFAIRNESS, which minimizes unfairness, frequently increases group modularity and diversity (Fig. 1 and 2, third column). Similarly, methods focused on group modularity, such as DEEPGROUP, GROUPDMON and GROUPSPECTRAL, can also increase fairness (Table IV, Facebook-g 0.836, Twitch 0.998) and enhance diversity (Table IV, Facebook-g 0.237, Twitch 0.250) by reinforcing the structure of minority groups. While not consistent across all settings, these gains highlight dependencies among fairness objectives.

V. RELATED WORK

Fairness in machine learning has attracted significant attention across tasks such as classification, recommendation, and ranking [5]–[8], [23]. Fairness definitions are categorized as either individual or group-based [24], [25]. In this work, we focus on group fairness in the context of community detection, where nodes in a network are clustered into communities, and fairness is evaluated with respect to how different demographic groups are connected within those communities.

Community detection is a special case of clustering [26], where the goal is to identify subsets of nodes with dense internal connectivity [2]. Much of the prior work on fair clustering has focused on *representation fairness*, ensuring proportional group representation in each cluster. The fairlets framework [9] formalized this idea and has been extended to support scalability [10], [27], [28], multiple attributes and fair representation [29]. Proportionality fairness aims to ensure fair treatment for any subset of the population [30].

Beyond group fairness, *individual fairness* methods seek to ensure that similar nodes receive similar outputs. In clustering, this has inspired formulations based on fair resource allocation [31], as well as graph-based approaches that use similarity matrices [32], multiview clustering extensions [33], and models that enforce proportional neighbor representation [34]. Recently, [35] applies fairness-aware regularization to non-negative matrix factorization, to encourage similar treatment of nodes with shared attributes. However, these methods do not directly optimize connectivity or modularity, nor are they integrated into a deep learning framework.

A complementary perspective considers *group-specific clustering quality*, aiming to minimize disparities in cluster cohesion, as seen in socially fair k-means [36] and equitable clustering [37]. The group modularity framework aligns with this perspective by measuring intra-community connectivity separately for each group and optimizing structural inclusion.

Distinct from prior work, we adopt a *connectivity-based notion of fairness*, extending link recommendation approaches that promote inter-group connections [38]. The modularity-based fairness framework we build upon introduces three key metrics, *group modularity*, *unfairness*, and *diversity*, which quantify intra-group strength, disparity across groups, and cross-group interactions, respectively [12], [18]. Previously incorporated into Louvain, we apply this framework to spectral and deep clustering.

While several fairness-aware clustering methods leverage *spectral formulations with fairness constraints* [11], [39], including those targeting individual similarity preservation [32]–[35], our approach is the first to apply modularity-based group fairness principles in both *spectral* and *deep learning* settings, enabling a unified view of community quality and fairness.

VI. Conclusion

In this paper, we addressed the problem of fairness in deep community detection by building on the group modularity framework, which quantifies how well different demographic groups are connected within detected communities. While prior work introduced modularity-based fairness metrics and fairness-aware variants of the Louvain algorithm, we extended these ideas into the spectral and deep learning settings.

Specifically, we proposed fairness-aware variants of the modularity matrix that incorporate group structure into the clustering objective. We applied these modifications in both spectral and deep clustering settings. In the spectral case, we introduced GroupSpectral and DiversitySpectral, which optimize group-sensitive modularity objectives via spectral decomposition. In the deep setting, we integrated these variants into the Deep Modularity Network (DMoN) framework, resulting in five GNN-based models. GroupDMoN and DiversityDMoN follow the spectral design by modifying the input matrices to encode fairness objectives. In contrast, DeepFairness, DeepDiversity, and DeepGroup preserve the original input graph and enforce fairness via loss function. Each model targets a distinct objective: intra-group connectivity, diversity, or fairness.

Our experimental evaluation on synthetic and real-world datasets demonstrated that the proposed models can effectively improve connectivity fairness while maintaining competitive modularity. We also showed that trade-offs between modularity and fairness can be controlled through tunable parameters.

A promising direction for future work is to extend our framework to multi-valued and continuous attributes. Another line of work is to design link recommendation algorithms to improve community fairness (e.g., along the lines of [40]) and to study how fairness evolves over time.

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. Work was also supported by the Hellenic Foundation for Research and Innovation (HFRI) under the 5th Call for HFRI PhD Fellowships (Fellowship Number: 20770).

References

[1] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets, 2nd Ed.* Cambridge University Press, 2014.
[2] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.
[3] R. Zafarani, M. A. Abbasi, and H. Liu, *Social Media Mining: An Introduction*. Cambridge University Press, 2014.
[4] D. A. Easley and J. M. Kleinberg, *Networks, Crowds, and Markets - Reasoning About a Highly Connected World*. Cambridge University Press, 2010.
[5] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Comput. Surv.*, vol. 54, no. 6, pp. 115:1–115:35, 2022.
[6] E. Pitoura, K. Stefanidis, and G. Koutrika, "Fairness in rankings and recommendations: an overview," *VLDB J.*, vol. 31, no. 3, pp. 431–458, 2022.
[7] Y. Dong, Y. Ma, S. Wang, C. Chen, and J. Li, "Fairness in graph mining: A survey," *TKDE*, 2023.
[8] A. Saxena, G. Fletcher, and M. Pechenizkiy, "Fairsna: Algorithmic fairness in social network analysis," *CSUR*, vol. 56, no. 8, pp. 1–45, 2024.
[9] F. Chierichetti, R. Kumar, S. Lattanzi, and S. Vassilvitskii, "Fair clustering through fairlets," in *NeurIPS*, 2017.
[10] M. Ceccarello, A. Pietracaprina, and G. Pucci, "Fast and accurate fair k-center clustering in doubling metrics," in *WWW*, 2024.
[11] M. Kleindessner, S. Samadi, P. Awasthi, and J. Morgenstern, "Guarantees for spectral clustering with fairness constraints," in *ICML*, 2019.
[12] C. Gkartzios, E. Pitoura, and P. Tsaparas, "Fair network communities through group modularity," in *WWW*, 2025.
[13] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E*, vol. 70, 2004.
[14] M. E. J. Newman, "Fast algorithm for detecting community structure in networks," *Phys. Rev. E*, vol. 69, 2004.
[15] V. D. Blondel, J.-L. Guillaume, and E. L. R. Lambiotte, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 10, 2008.
[16] V. A. Traag, L. Waltman, and N. J. van Eck, "From louvain to leiden: guaranteeing well-connected communities," *Scientific Reports*, vol. 9, no. 1, p. 5233, 2019.
[17] A. Tsitsulin, J. Palowitch, B. Perozzi, and E. Müller, "Graph clustering with graph neural networks," *J. Mach. Learn. Res.*, vol. 24, no. 1, 2023.
[18] K. Manolis and E. Pitoura, "Modularity-based fairness in community detection," in *ASONAM*, 2023.
[19] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E*, p. 036104, 2006.
[20] D. Yan, L. Huang, and M. I. Jordan, "Fast approximate spectral clustering," in *SIGKDD*, 2009.
[21] P. W. Holland, K. Laskey, and S. Leinhardt, "Stochastic blockmodels: First step," *Social Networks*, vol. 5, pp. 109–137, 1983.
[22] U. von Luxburg, "A tutorial on spectral clustering," 2007.
[23] S. Tsioutsiouliklis, E. Pitoura, P. Tsaparas, I. Kleftakis, and N. Mamoulis, "Fairness-aware pagerank," in *WWW*, 2021.
[24] S. Verma and J. Rubin, "Fairness definitions explained," in *FairWare@ICSE*, 2018.
[25] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. S. Zemel, "Fairness through awareness," in *ITCS*, 2012.
[26] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *CSUR*, vol. 31, no. 3, pp. 264–323, 1999.
[27] A. Backurs, P. Indyk, K. Onak, B. Schieber, A. Vakilian, and T. Wagner, "Scalable fair clustering," in *ICML*, 2019.
[28] S. K. Bera, S. Das, S. Galhotra, and S. S. Kale, "Fair k-center clustering in mapreduce and streaming settings," in *WWW*, 2022.
[29] S. K. Bera, D. Chakrabarty, N. Flores, and M. Negahbani, "Fair algorithms for clustering," in *NeurIPS*, 2019.
[30] X. Chen, B. Fain, L. Lyu, and K. Munagala, "Proportionally fair clustering," in *ICML*, 2019.
[31] S. Mahabadi and A. Vakilian, "Individual fairness for k-clustering," in *ICML*, 2020.
[32] J. Kang, J. He, R. Maciejewski, and H. Tong, "Inform: Individual fairness on graph mining," in *KDD*, 2020.
[33] Y. Wang, J. Kang, Y. Xia, J. Luo, and H. Tong, "ifig: Individually fair multi-view graph clustering," in *IEEE Big Data*, 2022.
[34] S. Gupta and A. Dukkipati, "Consistency of constrained spectral clustering under graph induced fair planted partitions," in *NeurIPS*, 2022.
[35] S. Ghodsi, S. A. Seyedi, and E. Ntoutsi, "Towards cohesion-fairness harmony: Contrastive regularization in individual fair graph clustering," in *PAKDD*, 2024.
[36] M. Ghadiri, S. Samadi, and S. S. Vempala, "Socially fair k-means clustering," in *FAccT*, 2021.
[37] M. Abbasi, A. Bhaskara, and S. Venkatasubramanian, "Fair clustering via equitable group representations," in *FAccT*, 2021.
[38] F. Masrour, T. Wilson, H. Yan, P. Tan, and A. Esfahanian, "Bursting the filter bubble: Fairness-aware network link prediction," in *AAAI*, 2020.
[39] J. Wang, D. Lu, I. Davidson, and Z. Bai, "Scalable spectral clustering with group fairness constraints," in *AISTATS*, 2023.
[40] S. Tsioutsiouliklis, E. Pitoura, K. Semertzidis, and P. Tsaparas, "Link recommendations for pagerank fairness," in *WWW*, 2022.