Information Sciences 662 (2024) 120271

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A hierarchical overlapping community detection method based on closed trail distance and maximal cliques

Pavla Dráždilová *, Petr Prokop, Jan Platoš, Václav Snášel

Department of Computer Science, VSB - Technical University of Ostrava, 708 00 Ostrava-Poruba, Czech Republic

A R T I C L E I N F O

Dataset link: http://www-personal.umich.edu/%7Emejn/netdata/

Dataset link: https://anonymous.4open.science/r/graph_hierarchical_agglomerative_clustering-C946/README.md

Keywords:
Overlapping community detection
Clique percolation
Closed trail distance
And hierarchical agglomerative clustering

A B S T R A C T

An important feature of real networks is their hierarchy and the existence of overlapping communities. Hierarchical agglomerative clustering is one way to determine the hierarchy of a network. To ensure the existence of overlapping communities, it is appropriate to choose the base elements for clustering – edges, cliques, etc. These base elements can then have common vertices and naturally provide the possibility of overlap. The proposed community detection method uses hierarchical agglomerative clustering on the 2-edge-connected component of the graph. Communities are constructed from maximal cliques as base elements. Novel dissimilarities for hierarchical agglomerative clustering were introduced for the merging of cliques. The dissimilarities use the size of the overlapped cliques and closed trail distance to express dissimilarity between communities in networks. The single linkage approach contains and extends the results of k-CPM. The proposed algorithm utilizing deterministic dissimilarity achieves comparable or superior outcomes compared to standard algorithms used for hierarchical or overlapping community detection.

1. Introduction

Using graph representation and network analysis tools can be beneficial for studying relationships between objects. A general description of community is a set of different objects connected more frequently among themselves in comparison to the rest of the network. In case of the possible belonging of objects to multiple communities, we are focusing here on overlapping communities [1]. Yang and Leskovec [2,3] noticed that the community overlaps are dense. In some real-world datasets, while most clustering algorithms cannot handle such dense overlapping structures, one vertex may belong to tens of communities simultaneously [4].

The representative method of overlapping clustering is the clique percolation method (CPM or k-CPM) by Palla et al. [5,6]. The detection of communities is realized via finding maximal cliques, construction of a clique graph with maximal cliques as vertices, and weighted edges representing the size of cliques' overlap that are bigger than or equal to a specified $k-1$. Communities are connected components in the clique graph where detected communities may not form a network cover. Some algorithms for overlapping community detection are based on clustering of more complex base elements than vertices – edges [7], cliques [8], weak-cliques [9], etc.

The second point of view on algorithms for community detection can be focused on the hierarchy of communities [1]: "Communities are nested within each other as many times as there are hierarchical levels." Algorithms for hierarchical community detection

* Corresponding author.
E-mail addresses: pavla.drazdilova@vsb.cz (P. Dráždilová), petr.prokop@vsb.cz (P. Prokop), jan.platos@vsb.cz (J. Platoš), vaclav.snasel@vsb.cz (V. Snášel).

https://doi.org/10.1016/j.ins.2024.120271
Received 31 July 2023; Received in revised form 30 January 2024; Accepted 30 January 2024

Available online 5 February 2024
0020-0255/Â© 2024 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Table 1
Notation used in the paper.

Symbol	Description
$V(G)$	Set of vertices of graph G
$E(G)$	Set of edges of graph G
n, m	Number of vertices and edges of graph
$deg(u)$	Degree of vertex u
$\langle deg \rangle$	Average degree of vertices
HAC	Hierarchical agglomerative clustering
$GHAC$	Graph hierarchical agglomerative clustering
SL, CL, AL	Single, complete and average linkage approach
A, A_{ij}	Adjacency matrix; one element from adjacency matrix
Q, Q_k	Clique and clique with k vertices
$SP(x_i, x_j)$	The shortest path between vertices x_i, x_j
$CT(x_i, x_j)$, $CT(u, v, w, u)$	The shortest closed trail containing vertices x_i, x_j; closed trail from u via v and w to u
C_i	i-th community
$\|C_i\|$	Size of i-th community
d_{SP}, d_{CT}	Shortest path and closed trail distance
d^{SL}_{GHAC}, d^{CL}_{GHAC}, d^{AL}_{GHAC}	Graph hierarchical agglomerative clustering dissimilarity with single, complete, and average linkage approach
$N(u)$, $N^+(u)$	$N(u) = \{v \in V(G); d_{SP}(u, v) = 1\}$, $N^+(u) = \{v \in V(G); d_{SP}(u, v) \leq 1\}$
λ	Level of cut in dendrogram

are often based on hierarchical agglomerative clustering (HAC). Ahn et al. [7] used a single linkage approach with Jaccard similarity, Shen et al. [10] agglomerate communities with the maximum similarity, and Blondel et al. [11] used the hierarchical approach based on modularity optimization. Another agglomerative method uses node influence and the similarity of nodes to detect non-overlapping communities [12]. Alternatively, divisive clustering is used in [13] in the form of a recursive partitioning algorithm, starting with a single community and separating the nodes into two communities by spectral clustering repeatedly.

The main motivation for developing a new community detection method was to combine the search for overlapped communities and the creation of their hierarchical structure. HAC is a commonly used procedure for detecting overlapping communities when the cliques are used as bases, e.g. EAGLE [10]. In this current work, we are introducing a novel dissimilarity for the HAC based on the structural closeness of cliques and their neighborhood in a graph. We designed new dissimilarities between communities that are deterministic and we used the closed trail distance between graph nodes and the size of communities that overlap. Closed trail distance (CT-distance) between vertices u, v in the unweighted, undirected, connected graph without bridges (2-edge-connected) is defined in the article [14] as the length of the shortest closed trail that contains vertices u, v (Table 1). The extension of CT-distance for undirected and weighted graphs was also listed in article [14]. The processing steps in the proposed community detection method are indicated in the graphical abstract. The CT-distance matrix among pairs of vertices is calculated for the use in dissimilarities in the proposed algorithm. All maximal cliques are detected in the source network and are used as bases in the HAC. The proposed dissimilarities serve for the agglomeration of bases and the creation of a hierarchy (dendrogram). The value of modularity for each possible level is monitored. The best value of modularity indicates the level of cut in the hierarchy. This result represents the network cover.

We would like to highlight the main contribution of this paper as follows:

- A hierarchical overlapping community detection method was proposed. The proposed method uses the HAC and maximal cliques as base elements for clustering.
- The relation between the well-known k-CPM and the proposed method was discussed. Due to the extended hierarchy, the proposed method allows the detection of better communities than the k-CPM.
- The proposed method is not focused on efficient computation for large graphs. Instead, it uses maximal cliques as building blocks. The dissimilarities are based on CT-distance and the size of cliques in the overlap which allows the study of the hierarchical structure of communities in the network.
- Resulting overlapping community structure depends on the sequential (greedy) merging of all maximal cliques and the already-found communities.

This article is organized as follows. Section 2 introduces the related work to community detection from hierarchical and agglomerative perspectives. Section 3 is focused on the relation between the CPM and HAC used for community detection with a single linkage approach. The section describes our motivation behind the proposed method. Section 4 presents the algorithm for community

detection and dissimilarities between clusters of vertices based on the CT-distance between vertices that are not in the intersection of clusters and takes into account the size of this intersection. The applied idea of clustering of more complex base elements than vertices – maximal cliques – enables the overlap between communities. Section 5 contains the experiments demonstrating the selection of cuts in the dendrogram where the hierarchy of communities for the proposed method is shown in a real-world network. The empirical evaluation of the method is also included and the results are compared with the selected well-known methods. The advantages of the proposed methods and future work are discussed in the conclusion in section 6.

2. Related work

There are currently many methods that perform hierarchical community detection. Some methods are algorithmically hierarchical [10,11] and create a hierarchy as a result of the applied algorithm. Another class of methods involves fitting a hierarchical model to the analyzed network. Schaub et al. [15] introduced a definition of hierarchy based on the concept of stochastic externally equitable partitions and their relation to probabilistic models, such as the stochastic block model. They focused on an agglomerative procedure that relies on accurately detecting the finest level in the hierarchy.

The result of HAC is represented by the proximity dendrogram [16] which is a tree-like structure where each node represents a cluster or a data point, and the branches show the merging of clusters during the clustering process.

The natural overlap can be constructed by partitioning links [7] instead of nodes. A node in the original graph is called overlapping if the links connected to it are put in more than one cluster. The authors use HAC with the SL approach and the similarity between links to build a dendrogram where each leaf is a link from the original network and the branches represent clusters of the links.

A different approach to community detection is applied in [17,5]. The authors developed the k-clique percolation method (k-CPM) for community detection. The community is created from k-cliques (Q_k) that are reached only from the k-cliques of the same community through a series of adjacent k-cliques. Two k-cliques are adjacent if they share $k-1$ vertices.

The extension of the CPM to the weighted network was proposed in [8] as CPMw. The authors introduced a module identification technique for weighted networks based on k-cliques having a subgraph intensity higher than a certain threshold and allowing shared nodes (overlaps) between modules.

A Sequential Clique Percolation (SCP) algorithm [18] was proposed for fast clique percolation detecting k-clique communities in a network by sequentially inserting its edges and keeping track of the emerging community structure. This algorithm has specifically been designed for (dense) weighted networks, where weight-based thresholding of either the links or the cliques formed by them is necessary for obtaining meaningful information on the structure. Reid et al. [19] analyzed SCP and stated: "However, these improved methods often perform poorly on networks with the kind of pervasively overlapping community structure we see in many real worlds social networks – an area of increasing interest in the applied study of community structure – and particularly poorly when performing percolation with high values of k."

The authors in [20] proposed the clique-based Louvain algorithm that classifies the non-classified node obtained after finding cliques in one of the communities by applying the Louvain algorithm.

One of the first algorithms for the detection of the overlapping and hierarchical community structure in complex networks is described in [21]. The method is based on the local optimization of a fitness function (ratio of the internal degree to the total degree of a module). The method corresponds to a sort of greedy optimization of the fitness function. It creates natural communities around vertices and the result is vertices' cover, i.e., it depends on the resolution parameter for scale.

Algorithm EAGLE [10] detects overlapping and hierarchical community structures in networks with a hierarchical agglomerative method. The similarity between communities is based on modularity and the maximal cliques are its base elements. This approach confirms that HAC is applicable for the hierarchy detection among communities, and maximal cliques (as base elements) ensure the overlap of communities.

Maximal cliques are used in [22]. This paper proposes a Maximal Clique-based Multiobjective Evolutionary Algorithm (MCMOEA) for overlapping community detection. The representation scheme is based on the maximal clique and the algorithm can provide hierarchical partitions of the given network.

A DOCNA [23] is an algorithm for detecting overlapping communities in networks based on maximal cliques where an improved version of the Bron-Kerbosch Algorithm is adopted.

The request for maximal cliques is quite restrictive. Therefore, in [9] they used weak cliques as the base elements. The authors proposed a weak-CPM for overlapping community detection in a large-scale network.

The greedy coupled-seeds expansion method [24] for the overlapping community detection used a fitness function that is based on the size of a common neighbor of two vertices – similar to a weak clique percolation.

The authors in [25] propose an algorithm MOKP that uses k-plexes to generate community seeds from the whole network and assigns the remaining nodes by modularity optimization. This algorithm does not detect overlapping communities.

To identify the overlapping community structure, the authors in [26] constructed a maximal clique network from the original network, and proved that the optimization of their metric on the original network is equivalent to the optimization of Newman's modularity on the maximal clique network.

A useful approach for overlapping community detection is based on Nonnegative Matrix Factorization (NMF). Yang and Leskovec [27] used the NMF approach to find the overlapping communities in large-scale networks, Wang et al. [28] proposed the Modularized Nonnegative Matrix Factorization (MNMF) model to incorporate the community structure into network embedding, and Ye et al. [29] proposed a model called Deep Autoencoder-like NMF (DANMF) for community detection, inspired by the unique feature representation learning capability of the deep autoencoder.

The overview of community detection methods with hierarchical agglomerative clustering or overlapping community detection can be found in [21,30,31,1,32]. The survey of community detection using nonnegative matrix factorization (NMF) is in [33]. The comprehensive survey of community detection focused on deep learning is mentioned in [34].

The evaluation of the appropriateness of the detected community structure is a very important part of community analysis. Metrics related to all the classes of community structures (disjoint, overlapping, local, hierarchical, etc.) are presented in a survey [35] of the state-of-the-art metrics used for the detection and evaluation of community structure in networks. The article [36] focuses on the quality of community structure and contains a broad overview and classification of methods for the evaluation of detected communities.

The result of HAC is a dendrogram that represents the hierarchical structure of communities. The determination of the dendrogram's cut level is a crucial aspect that plays a pivotal role in uncovering an optimal community structure. To assess the efficacy of the community structure derived through HAC, the modularity metric serves as a valuable tool for evaluation. One of the modularities for overlapped communities can be found in [10]. This work introduces a belonging coefficient. The belonging coefficient of a node i for a given community is redefined as the number of communities O_i to which it belongs.

3. Relation between HAC and clique percolation

We would like to discuss a generalization of CPM to HAC. This generalization leads us to the theoretical grounding for the proposed dissimilarities. The idea about the clique's hierarchy detected by hierarchical clustering was stated in [37]. The authors used a co-clique matrix as an input for hierarchical clustering. This co-clique matrix corresponds to the adjacency matrix of the weighted graph of overlapped maximal cliques.

As far as complexity is concerned, the CPM was designed for selected k, very often $k = 3$ [17]. Derenyi et al. in [17] and Yuan et al. in [38] use for k-clique graph a different terminology and they named it as, "k-clique adjacency graph."

The standard k-CPM [17] can be described via a k-clique graph as follows:

1. Detect k-cliques in the source network and create a k-clique graph, where vertices are k-cliques and the edges exist between k-cliques which have $(k-1)$ vertices in the overlap in the source network.
2. Find the connected components in the clique graph. These connected components in the clique graph correspond to communities in the source network.

An effective algorithm based on maximal cliques in k-CPM [19] builds a minimal spanning forest over the maximal cliques, using a simple data structure to reduce unnecessary clique intersection tests. The Yuan et al. aim in [38] to find the densest clique percolation community which contains a given set of query nodes. They use a maximal clique adjacency graph and a maximal clique adjacency spanning tree with the maximum total weight of edges where the weight of the edge in the maximal clique adjacency graph is equal to the size of overlap between maximal cliques in the source graph.

Generalized CPM inspired by the article [38] introduces a connection between CPM and graph hierarchical clustering with a single linkage approach:

1. Detect maximal cliques in the source network and create a weighted maximal clique graph. The weights of edges in the maximal clique graph correspond to the size of the overlap between maximal cliques in the source network.
2. Use the HAC with the SL approach on the maximal cliques for the creation of a dendrogram. The weight of an edge is the similarity between vertices in the maximal clique graph.
3. For a specified k, obtain a level of cut in the dendrogram that represents the same result as in the standard k-CPM. Clusters from the dendrogram are the connected components in the maximal clique graph with the edge's weight bigger or equal to $k-1$. These represent overlapped communities in the source network. The minimal weight in the maximal clique graph equals 1 for $k = 2$ and in this situation, all vertices of the connected source graph are in one community.

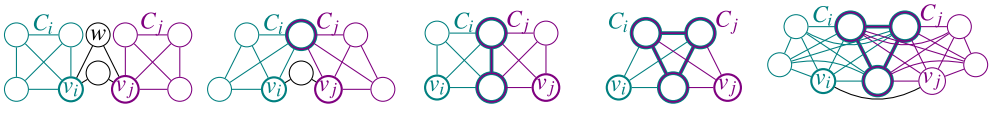
3.1. From k-CPM to a novel dissimilarity for GHAC

The formation of communities in the CPM can be naturally described with the SL approach (minimal distance or maximal similarity between two elements which are in different clusters) in a hierarchical agglomerative clustering on the graph (GHAC). The size of the overlap of the maximal cliques determines the degree of similarity and thus allows the hierarchization of the obtained communities with an overlap greater or equal to two. The other condition is merging the adjacent cliques, therefore the CT-distance between a pair of nodes (the length of the shortest closed trail containing a pair of nodes) is the smallest. Further extension of this hierarchy can be achieved by introducing a new dissimilarity based on a CT-distance.

At first, we define the dissimilarity between the subgraphs C_i and C_j based on the CT-distance and the SL approach in the GHAC:

$$d_{CT}^{SL}(C_i, C_j) = min_{(v_i \in C_i \setminus C_j),(v_j \in C_j \setminus C_i)} d_{CT}(v_i, v_j).$$

The next theorem shows the connection between the proposed dissimilarity and the percolation of the two adjacent cliques in the CPM.



(a) $d_{CT}(v_i,v_j)=4$ $d^{SL}_{GHAC}(C_i,C_j)=\frac{4}{1}$ (b) $d_{CT}(v_i,v_j)=4$ $d^{SL}_{GHAC}(C_i,C_j)=\frac{4}{2}$ (c) $d_{CT}(v_i,v_j)=4$ $d^{SL}_{GHAC}(C_i,C_j)=\frac{4}{3}$ (d) $d_{CT}(v_i,v_j)=4$ $d^{SL}_{GHAC}(C_i,C_j)=\frac{4}{4}$ (e) $d_{CT}(v_i,v_j)=3$ $d^{SL}_{GHAC}(C_i,C_j)=\frac{3}{4}$

Fig. 1. Dissimilarity SL GHAC between communities depends directly proportional to CT-distance of vertices out of overlap (in different communities) and is inversely proportional to the size of overlap.

Theorem 1. *The dissimilarity $d^{SL}_{CT}(Q,Q')$ between the two adjacent k-cliques in graph G with $k \geq 3$ based on the CT-distance and the SL approach equal to* 3 *or* 4.

Proof. Let Q, Q' be two adjacent k-cliques and $N(u) = \{v \in V(G); d_{SP}(u,v) = 1\}$ is a neighborhood of a vertex u. Two k-cliques are adjacent if they share $k-1$ vertices [5]. The amalgamation (gluing) of the two cliques that share $k-1$ vertices for $k \geq 3$ create a $4-CT$ component [39] because for all $u \in Q$ and for all $u' \in Q'$, the following situations may occur:

- $u, u' \in Q \cap Q'$ and $(u = u') \Rightarrow d_{CT}(u,u') = 0$,
- $u, u' \in Q \cap Q'$ and $u \neq u' \Rightarrow \exists v \in Q \cap Q'$ such that $v \in N(u)$ and $v \in N(u') \Rightarrow |CT(u,v,u',u)| = 3 = d_{CT}(u,u')$,
- $u, u' \notin Q \cap Q'$ and $u' \notin N(u) \Rightarrow \exists v, w \in Q \cap Q'$ such that $v, w \in N(u)$ and $v, w \in N(u') \Rightarrow |CT(u,v,u',w,u)| = 4 = d_{CT}(u,u')$,
- $u, u' \notin Q \cap Q'$ and $u' \in N(u) \Rightarrow \exists v \in Q \cap Q'$ such that $v \in N(u)$ and $v \in N(u') \Rightarrow |CT(u,v,u',u)| = 3 = d_{CT}(u,u')$. □

The above description of dissimilarity (d^{SL}_{CT}) does not distinguish between the smaller and the bigger overlap of cliques. The modification of the dissimilarity that incorporates the size of overlap better describes the relation between the overlapping cliques and the formation of hierarchical structure detected communities.

The size of overlap between communities C_i and C_j is defined as the number of vertices in the maximal shared clique of communities C_i, C_j. It is a different definition than that in [6], where the size of overlap is defined as the number of shared nodes in C_i and C_j.

The newly proposed dissimilarity $d^{SL}_{GHAC}(C_i,C_j)$ between communities C_i and C_j is directly proportional to the node distance (CT-distance) outside of the overlap of C_i and C_j, inversely proportional to overlap size, uses the SL approach, and then captures the hierarchy of the detected communities as well as the hierarchy of the percolated cliques:

$$d^{SL}_{GHAC}(C_i,C_j) = \frac{min_{(v_i \in C_i \setminus C_j),(v_j \in C_j \setminus C_i)} d_{CT}(v_i,v_j)}{1 + argmax_{Q \in C_i \cap C_j} |Q|}.$$

The SL approach for GHAC and the incorporation of the size of the overlap on base elements (cliques) ensures that the cliques with the biggest overlap are amalgamated at first – they have the smallest dissimilarity.

Fig. 1 shows the different situations for two communities (cliques C_i, C_j) that differ in the size of the overlap.

The maximal overlapping cliques in Fig. 1a are edges $(v_i,w),(v_j,w)$ with a common vertex w. Fig. 1b represents the situation out of the clique percolation but with two 4-cliques with an overlap. Figs. 1c and 1d correspond to the 3-clique percolation (the size of overlap equal to 2) and the 4-clique percolation (the size of overlap equal to 3). Fig. 1e corresponds to the 6-cliques amalgamation with a size of the overlap equal to 3 and the denser neighborhood ($d_{CT}(v_i,v_j) = 3$).

The k-CPM uses only the information about the size of the overlap of the two adjacent k-cliques. This size is equal to $(k-1)$. The CT-distance between vertices in the adjacent clique (that are not in overlap) is mostly equal to 4. These two aspects of the suggested dissimilarity, yield identical outcomes to the clique percolation. However, in denser network regions (refer to Fig. 1e), the CT-distance can be reduced to 3 (Theorem 1). In such cases, the proposed dissimilarity employing the SL approach captures additional information regarding the network's density at the intersection of the two cliques. This dissimilarity shows to be more accurate in extremely dense parts of the network than the CPM does. The HAC with the proposed dissimilarity finds a hierarchical structure over the communities detected by CPM with an arbitrary k. The usage of CT-distance in the proposed dissimilarity reflects the relation, not only among cliques, but also for the not-so-dense communities.

4. Hierarchical clustering based on CT-distance

In this article, we propose new CT-distance based dissimilarities for hierarchical agglomerative clustering on graphs. We have formalized the basic idea for our method as an extension of generalized CPM from the previous section. The idea of the use of novel dissimilarity d^{SL}_{GHAC} with GHAC (therefore SL GHAC) and the specification of the level of cut in the dendrogram can be summarized as a sequence of processing steps:

1. Detect the maximal cliques as base elements in the source network.
2. Calculate the CT-distance among vertices in the source network.
3. Use the SL GHAC for cluster construction.

4. The lower part of a resulting dendrogram mostly corresponds to the dendrogram in generalized CPM. Then, the hierarchy continues and connects the communities with the size of the overlap (the size of the biggest common clique) equal to one.
5. Use evaluation for quality of community detection to determine the best level of cut in the dendrogram or specify the number of communities from the dendrogram.

The SL GHAC at the cut level with a value equal to $4/k$ corresponds to the result of the k-CPM. It is a consequent of the Theorem 1 where $d_{CT}^{SL}(Q,Q')$ for adjacent cliques is mostly equal to four and, in an extremely dense part of the network, can be equal to three. The situation with $d_{CT}^{SL}(Q,Q')=3$ is incorporated into the level of the cut with the value $d_{GHAC}^{SL}(Q,Q')=4/(1+(k-1))$, where $(k-1)$ is the size of the overlap between the two cliques.

The SL GHAC framework will be further extended in the following subsection. The introduction of other dissimilarities will modify the idea of the percolation of adjacent cliques when the single linkage approach is used.

4.1. Additional dissimilarities for the GHAC based on the CT-distance

The standard linkage methods of the HAC [40] have different properties. The SL HAC tends to produce unbalanced and straggly clusters (chaining), especially in large data sets. It does not take into account the cluster structure. The CL HAC tends to find compact clusters with equal diameters (maximum distance between objects). It does not take into account the cluster structure. The AL HAC tends to join clusters with small variances and takes into account the cluster structure.

The evaluation of the other HAC methods [41] shows that more successful methods than the SL are CL, AL, or Ward's methods.

Our experiments empower the GHAC with multiple dissimilarities based on the CT-distance, and the size of overlap. Apart from the SL, we have defined the approaches based on the CL and the AL as:

$$d_{GHAC}^{CL}(C_i,C_j)=\frac{max_{(v_i\in C_i\setminus C_j),(v_j\in C_j\setminus C_i)}d_{CT}(v_i,v_j)}{1+argmax_{Q\in C_i\cap C_j}|Q|},$$

and

$$d_{GHAC}^{AL}(C_i,C_j)=\frac{\sum_{(v_i\in C_i\setminus C_j),(v_j\in C_j\setminus C_i)}d_{CT}(v_i,v_j)}{|(C_i\cup C_j)\setminus(C_j\cap C_i)|(1+argmax_{Q\in C_i\cap C_j}|Q|)}.$$

For the current work, we have denoted the use of the GHAC method with dissimilarity d_{GHAC}^{SL} as SL GHAC. The other markings of AL GHAC and CL GHAC are applied as well.

4.2. Community detection computation procedure

The proposed community detection methods as GHAC framework are summarized in computation steps in Algorithm 1. The proposed methods differ in the used dissimilarities.

Algorithm 1: Proposed community detection method based on the GHAC and dissimilarity leveraging CT-distance.

Input : The biggest connected component of a network without bridges
Output: Network cover

Step 1: Calculate CT-distance matrix among vertices in a input graph.
Step 2: Find maximal cliques (Bron-Kerbosch alg.).
Step 3: Graph hierarchical agglomerative clustering:
  Step 3.1: Agglomerate communities according to proposed dissimilarity with maximal cliques as base elements.
  Step 3.2: Map merged clusters of base elements to source graph vertices.
  Step 3.3: Evaluate the structural quality of network cover by modularity.
  Step 3.4: Repeat the algorithm from Step 3.1 until all the clusters are merged.
Step 4: Choose the best level of a cut of a dendrogram.

Suurballe's algorithm [42] is used to calculate the CT-distances among vertices. The CT-distances are one part of used dissimilarities in the GHAC. Another part of dissimilarities takes the size of the overlap into account.

The Maximal cliques are used as bases in the GHAC and the merged clusters of maximal cliques are mapped to vertices with a few post-processing steps. Firstly, the non-agglomerated bases of size 2 (edges) are not considered in the final communities and they are filtered out. The vertices, that are not part of any community, are added as separate communities. These post-processing steps allow us to compare the network cover of the different community detection methods with respect to the need for some modularity measure to contain all vertices.

The modularities of the *overlapping* network covers are used as quality evaluation criteria for the selection of the level of the cut in a dendrogram.

Table 2
Characteristics of the giant connected component of network used in experiments. Number of nodes (n), number of edges (m), density ($dens$), clustering coefficient (CC), average degree ($\langle deg \rangle$), maximal degree (deg_{max}), shortest path distance diameter ($diam_{SP}$), closed trail distance diameter ($diam_{CT}$), and number of maximal cliques ($\#cliques$).

Network	n	m	$dens$	CC	$\langle deg \rangle$	deg_{max}	$diam_{SP}$	$diam_{CT}$	$\#cliques$
Zachary's karate club	33	77	0.15	0.26	4.7	17	5	11	35
American college football	115	613	0.09	0.41	10.7	12	4	8	281
Coauthorships in network science	340	865	0.015	0.45	5.1	32	16	39	169
High-energy theory collaborations	4557	12399	0.001	0.30	5.4	50	16	41	3976

The implementation of the GHAC method is written in Python 3.11. The agglomeration method is re-implemented for the proposed dissimilarities calculation. The commonly used libraries are used for the graph-related operations.[1]

5. Experiments

The suggested community detection methods are compared to other known methods over the selection of real-world networks. According to the CT-distance definition requirement, only the giant connected component of each network after bridge removal was used in the experiments. A summary of the pre-processed networks is given in Table 2.

The proposed community detection methods based on the SL (CL, AL) GHAC offer several different levels of dendrogram cut to provide community detection results. There is a need to evaluate the quality of the detected communities in agglomerative structures to select the best cut in the dendrogram. Modularity can be used for that, which explains the reason for the EAGLE algorithm to employ it [10]. To ensure the corresponding modularity evaluation of the detected components for various community detection methods and the modularity measures used in this paper, every node has to be assigned to at least one community; hence, a node not assigned to any community is treated as a community of a single node.

5.1. Quality evaluation of overlapping communities

Three different definitions of modularity [10,43,44] for overlapping communities are applied in the agglomerative process of the GHAC method to determine the best network cover when using the proposed community detection method. The extensions of modularity to overlapping communities are based on the traditional Newman approach in [45].

Shen's modularity extension [10] for overlapping communities considers vertex membership in multiple communities. It is directly equivalent to Newman's modularity when vertices belong to just one community. This is defined as follows:

$$M^e = \frac{1}{2m}\sum_{k=1}^{c}\sum_{i,j\in C_k}\frac{1}{O_i O_j}\left[A_{ij} - \frac{deg(i)deg(j)}{2m}\right],$$

where O_v is the number of communities to which vertex v belongs, c is the number of communities.

The other measure for quantifying cluster structures in graphs was introduced by Lazar in [43]. It is based on two assumptions: one, the edges of a node should be primarily inside the community, and two, the clusters (communities) should be dense. The measure is defined as:

$$M^{ov} = \frac{1}{c}\sum_{k=1}^{c}\left(\sum_{i\in C_k}\frac{\sum_{j\in C_k, i\neq j} a_{ij} - \sum_{j\notin C_k} a_{ij}}{deg(i)O_i}\frac{n^e_{C_k}}{|C_k|\binom{|C_k|}{2}}\right),$$

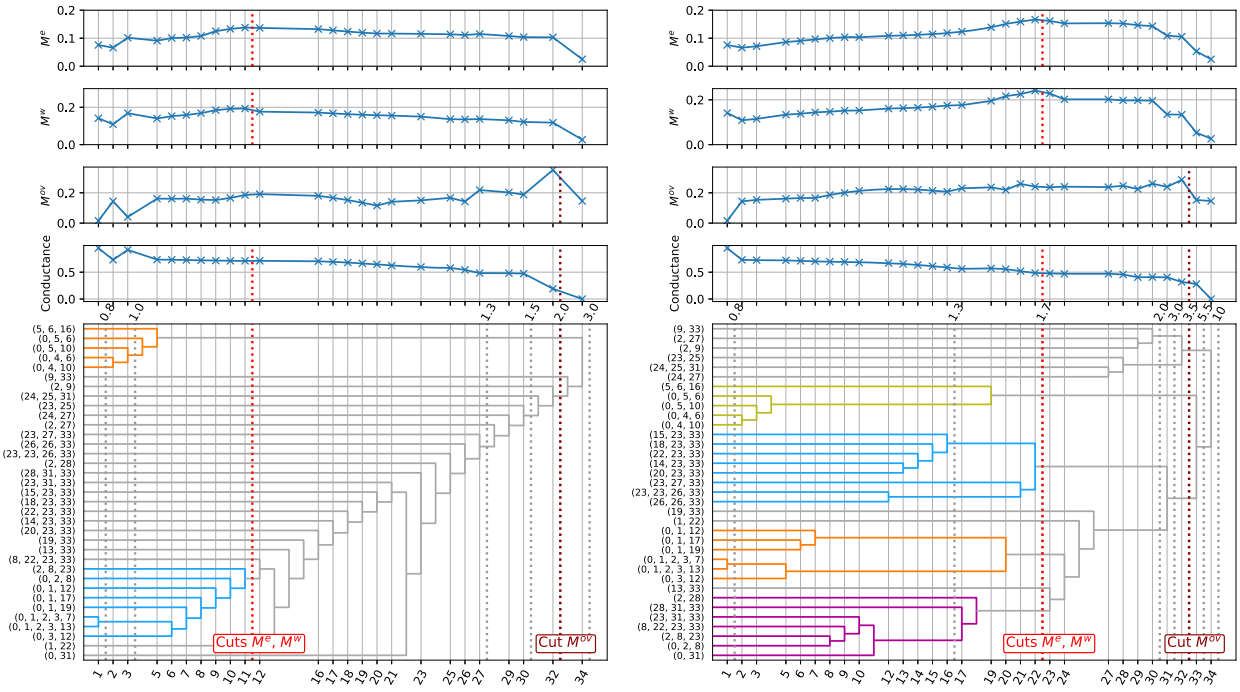
where c is the number of clusters, O_i is number of clusters the i belongs to, where $|C_k|$ is the number of nodes and $n^e_{C_k}$ is the number of edges that the kth cluster C_k contains.

The third modularity for the overlapping communities is defined by Cao in [44] and leverages the weighted edges by cosine similarity of the node's neighborhood to tackle the problem of resolution limit. Resolution limit means favoring large communities by a modularity measure. This disadvantage can be limited by properly weighted edges [44]. The modularity is defined as follows:

$$M^w = \frac{1}{2W}\sum_{k=1}^{c}\sum_{i,j\in V}(w_{ij} - \frac{s_i s_j}{2W})u_{ki}u_{kj},$$

where $U = [u_{ki}]$ and the value represents the degree to which node v_i is in the kth community, the edge weight is $w_{ij} = \frac{|N(i)\cap N(j)|}{\sqrt{|N(i)||N(j)|}}$, the strength of node v_i is $s_i = \sum_{j\in N(i)} w_{ij}$ and W is the total weight of the edges.

[1] The code for the method is available online https://anonymous.4open.science/r/graph_hierarchical_agglomerative_clustering-C946/README.md.



(a) SL GHAC ($k = 2$) (b) CL GHAC ($k = 2$)

Fig. 2. Dendrograms of GHAC method with the proposed dissimilarities d^{SL}_{GHAC} and d^{CL}_{GHAC} for Zachary's karate club network, where all maximal cliques are the input bases for the GHAC. The network cover in each agglomerative step is evaluated and the values are given in line plots. The bottom axis illustrates the agglomerative steps in the GHAC method, and the top axis portrays the respective dissimilarities.

5.2. Indication of the cut level in dendrogram by modularity

Zachary's karate club network is used in this section to demonstrate the process of choosing the best level of a cut in a dendrogram.

Fig. 2 presents a dendrogram that visually represents the agglomerative process of the proposed method, revealing the agglomeration of maximal cliques and clusters, and providing insights into the underlying structure. The bottom axis of the dendrogram demonstrates the agglomerative steps of the algorithm, while the top axis corresponds to the level of cuts based on dissimilarity value.

Throughout the agglomeration process, the quality of network cover is assessed using modularities as a structural quality measure. The progress of the modularities values during the agglomerative stage of the GHAC method can be observed in Fig. 2 through the three line plots displayed on the top.

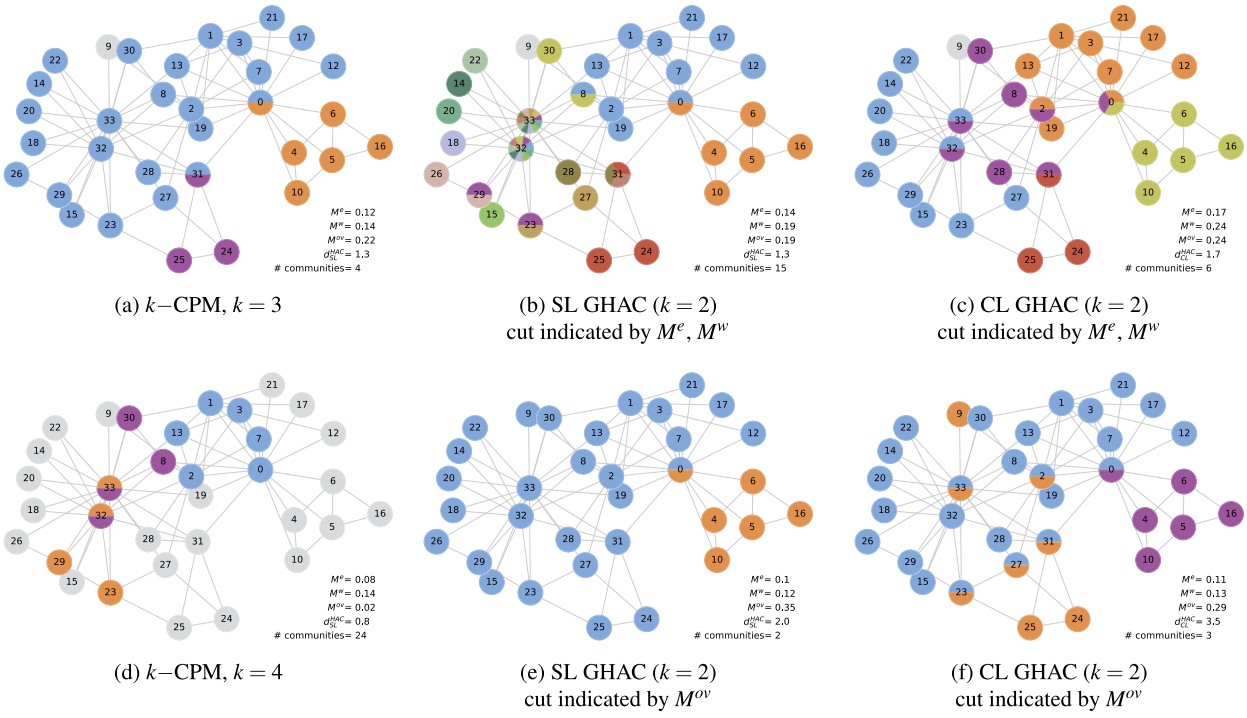
The highest modularity value for each line plot is selected and depicted as a dotted line, indicating a cut in the dendrogram. In our proposed community detection algorithm, this dotted line represents the cut in the dendrogram, where the bases are merged and mapped to nodes of the network. It is noteworthy that the M^e and M^w exhibit the same cut level in the dendrogram, while M^{ov} indicates a different cut level.

By comparing the corresponding dendrograms (illustrated in Fig. 2) obtained for methods SL GHAC and CL GHAC, we observe significant differences in the hierarchical structure. The higher overall modularity values M^e and M^w are achieved as seen in Fig. 2b, while Fig. 2a shows a higher modularity value for M^{ov}.

Each highlighted cut within the dendrogram corresponds to a network cover that is further visualized in Fig. 3 in the original network.

During the analysis of the SL GHAC method, we can observe the equivalent result to the k-CPM method (with $k = 3$) as the cut in the dendrogram occurred at step 27, represented by a grey dotted line with a dissimilarity value of a cut $d^{SL}_{HAC} = 4/3$ on the top axis. SL GHAC continues with the detection of larger clusters - over and above the clusters detected via clique percolation. The modularity measures M^e and M^w indicated the best cut for the agglomerative step 11. Different optimal cut at level 32 was identified by modularity M^{ov}. The network covers obtained from the SL GHAC method can be observed in Fig. 3b and Fig. 3e. It is worth mentioning that none of these network covers exhibit intuitively meaningful communities for the main actors of the social network.

The author in [46] discussed the difficulties with the k-CPM in Zachary's karate network, where the community detection method is not able to distinguish between communities associated with two key individuals (node 0 and 33) who played pivotal roles in the division of the karate club. The obtained communities for the k-CPM method can be observed in Fig. 3a and Fig. 3d.



(a) $k-$CPM, $k = 3$ (b) SL GHAC ($k = 2$) cut indicated by M^e, M^w (c) CL GHAC ($k = 2$) cut indicated by M^e, M^w

(d) $k-$CPM, $k = 4$ (e) SL GHAC ($k = 2$) cut indicated by M^{ov} (f) CL GHAC ($k = 2$) cut indicated by M^{ov}

Fig. 3. Illustration of the network covers detected by different methods for Zachary's karate club network. The grey color is used for communities with single nodes.

The dendrogram in Fig. 2b shows a different hierarchical structure for the CL GHAC method. The corresponding network cover, as displayed in Fig. 3c, reveals the presence of separate communities specifically formed for nodes 0 and 33 within the karate club. These communities are visually represented by the colors blue and orange, respectively. Additionally, there is a purple community that exhibits overlaps and shared vertices between both main actors in the karate club.

5.3. Hierarchical aspects of the proposed methods

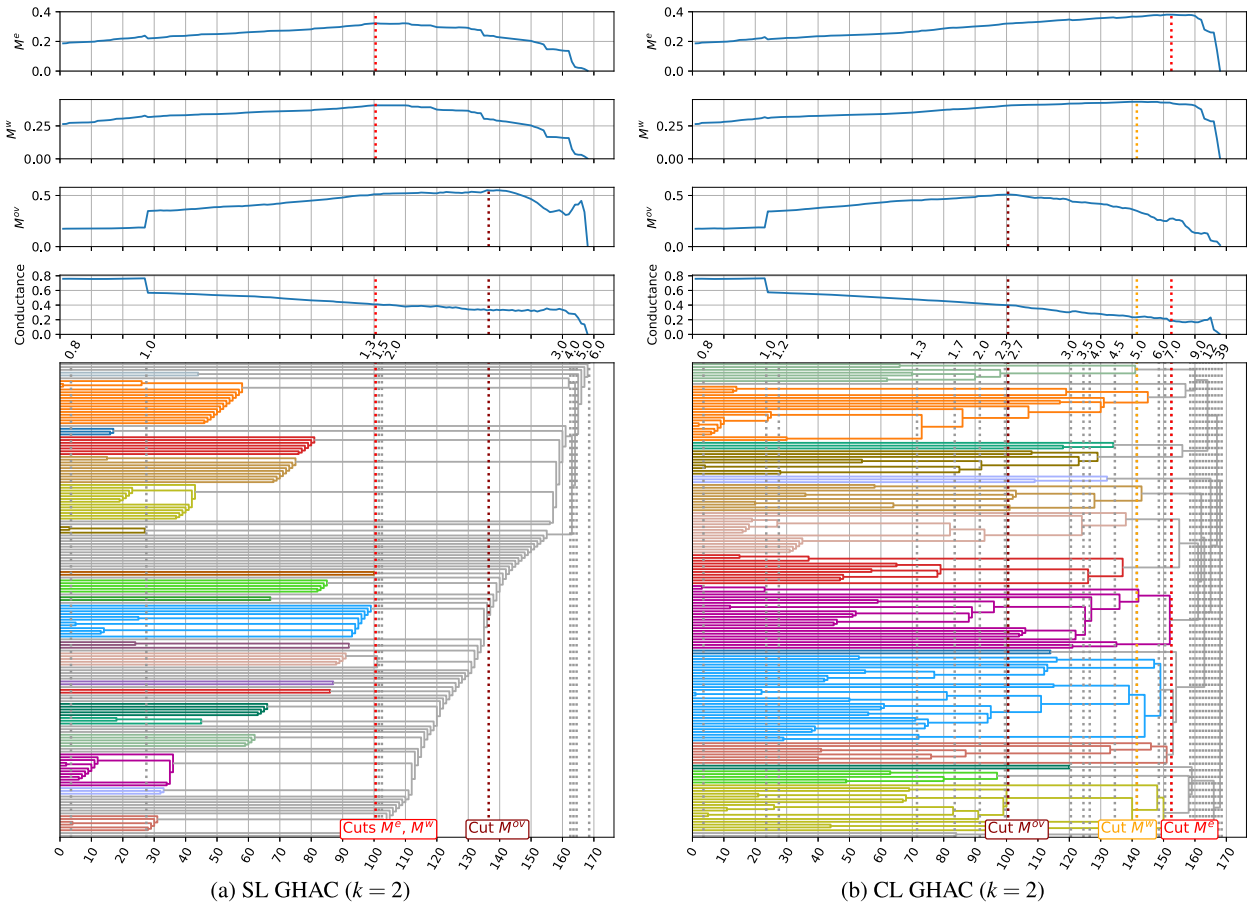
The hierarchical aspects of the proposed methods are studied for Coauthorships in network science. The different hierarchical structures for the proposed methods are visualized by dendrograms. The relation between the hierarchy and the detected overlapping communities is discussed for SL GHAC and CL GHAC.

One of the primary advantages of the proposed method is its ability to reveal the hierarchical structure of maximal cliques within a network. A single cut in a dendrogram yields communities but offers a limited perspective of the community structure. Nevertheless, the sequence of the network covers displayed in Figs. 5b, 5d, and 5f demonstrate the agglomerative process of the CL GHAC across various steps and reveal a hierarchy of some communities. For instance, focusing on the node representing M. Newman in the network (denoted in the top right corner), we observe the node's assignment to five communities in Fig. 5b. His two communities in step 141 of Fig. 4b are colored in light khaki and light purple in Fig. 5d, which are subsequently merged into a single purple community as portrayed in Fig. 5f.

The cut level indicated by modularities M^e and M^w for the SL GHAC equates to k-CPM, due to the dissimilarity value $d_{GHAC}^{SL} = 4/3$. The complete hierarchical structure created by SL GHAC is visualized as a dendrogram in Fig. 4a. This dendrogram allows us to examine the outcomes of the method beyond the percolation of the k-CPM method for $k = 3$. This dendrogram also provides a visual representation of the hierarchical structure, enabling manual inspection. The formation of long and connected structures is evident in the SL GHAC dendrogram in Fig. 4a. At level 136, approximately 50% of the bases are incorporated into a single cluster during agglomeration which is the effect of chaining characteristic for the SL approach. This cut's outcome is visualized in Fig. 5c, where the community highlighted by blue color corresponds to the agglomeration in the bottom half of the dendrogram. In contrast, the hierarchical community structure for CL GHAC is more balanced, merging similar numbers of bases to form clusters at different hierarchical levels. The network covers for CL GHAC contain more locally-centered communities compared to SL GHAC. Figs. 5c and 5d illustrate similar numbers of detected communities, but with markedly distinct community structures.

5.4. Empirical evaluation of the proposed community detection methods for selected real-world networks

We have examined the outcomes of our proposed methods by applying them to a selection of real-world networks. The subsequent sections will provide a detailed analysis of the results obtained for selected networks.



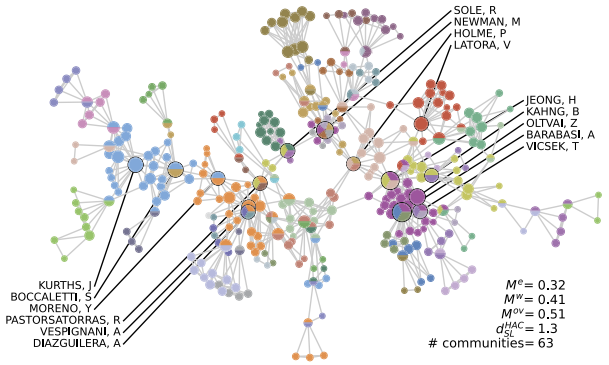
(a) SL GHAC ($k = 2$) (b) CL GHAC ($k = 2$)

Fig. 4. Dendrograms of GHAC method with the proposed dissimilarities d^{SL}_{GHAC} and d^{CL}_{GHAC} for Coauthorships in networks science. The network cover in each agglomerative step is evaluated and values are given in line plots. The bottom axis illustrates the agglomerative steps in the GHAC method, and the top axis portrays the respective dissimilarities.

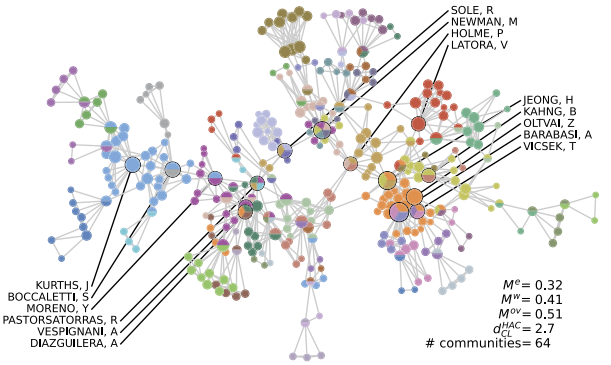
According to the definition of the proposed dissimilarities, the GHAC method merges communities with the smallest dissimilarity first during the agglomerative process. The other hierarchical community detection method used for comparison is the EAGLE algorithm [10]. This algorithm uses the HAC. The linkage type during the agglomerative process is given by the similarity $M = \frac{1}{2m}\sum_{v\in C_1, w\in C_2, v\neq w}\left[A_{vw} - \frac{deg(v)deg(w)}{2m}\right]$; therefore, the HAC method merges communities with the greatest similarity. Other well-known community detection methods selected for quality comparison are the k-CPM method, and the two algorithms based on label propagation used in the experiments for comparison: Speaker-listener Label Propagation algorithm (SLPA) [47] and Democratic Estimate of the Modular Organization of a Network (DEMON) [48]. The SLPA and DEMON methods are not deterministic, the 10 iterations of these algorithms were performed for each network during the community detection quality evaluation, and the best modularity is reported in Tables 3, 4, 5.

This empirical evaluation also includes a comparative analysis using the NMF approach, specifically focusing on the MNMF [28] and DANMF [29] algorithms. These algorithms generate embeddings that reflect the network's community structure. They incorporate multiple input parameters, including the number of detected communities, which is crucial, yet not predetermined in real-world networks. To address this, we used an optimization process for quality evaluation through hyperparameter search. The NMF-inspired algorithms were applied in both non-overlapping and overlapping regimes. In the non-overlapping regime, nodes are assigned to communities based on the highest belonging values from the algorithms' membership matrix. In the overlapping regime, a threshold value for the membership matrices [49] was introduced as an additional parameter, with the optimal threshold identified through hyperparameter search. The number of hyperparameter search iterations was set to 120 for the non-overlapping and 600 for the overlapping regime. The best modularity values are reported in Tables 3, 4, 5.

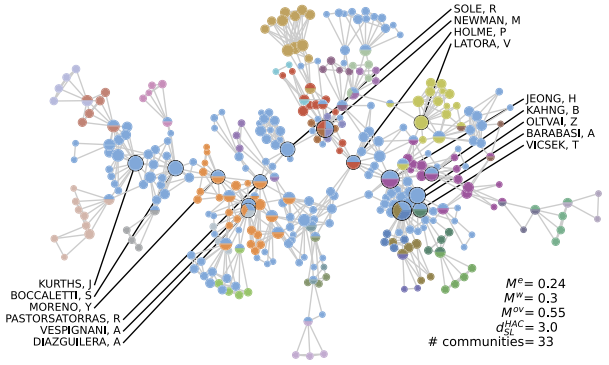
In the following experiments, we used the selection of input bases by minimal size filtration for the HAC method in the proposed methods (GHAC) and the EAGLE algorithm. In the experiments, the base's minimum size ranges from size $k = 2$ (all maximal cliques including edges) to size $k = 4$ (maximal cliques of size 4 or greater). These results are shown in Tables 3, 4, 5, where we display the maximal modularity value achieved by each community detection method and its corresponding parameters (k, r or ϵ), or the overlapping regime of the NMF-based methods. The network cover is expressed through the modularity values presented in the



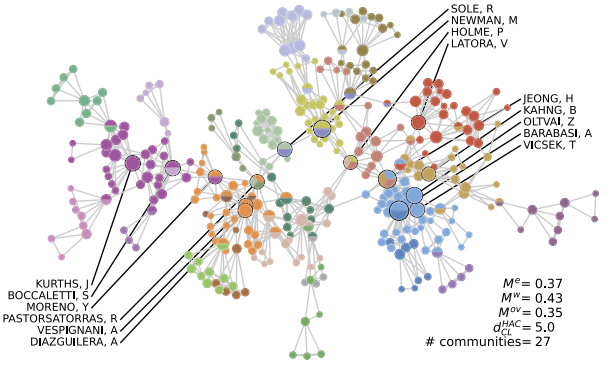
(a) SL GHAC ($k = 2$) cut in step 100 indicated by M^e, M^w



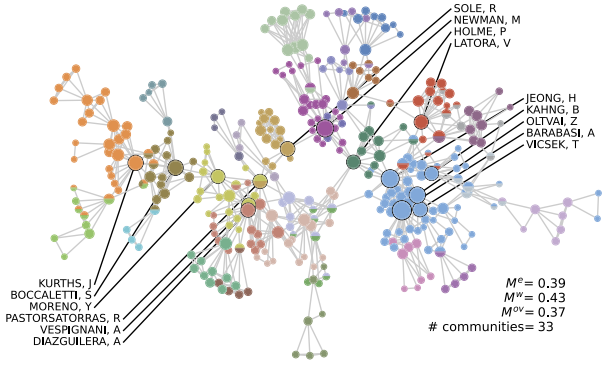
(b) CL GHAC ($k = 2$) cut in step 100 indicated by M^{ov}



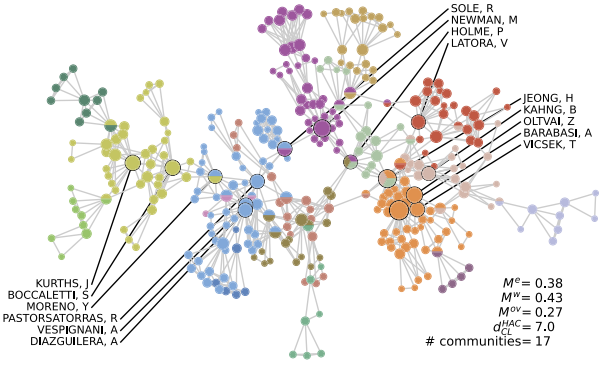
(c) SL GHAC ($k = 2$) cut in step 136 indicated by M^{ov}



(d) CL GHAC ($k = 2$) cut in step 141 indicated by M^w



(e) SLPA ($r = 0.05$)



(f) CL GHAC ($k = 2$) cut in step 152 indicated by M^e

Fig. 5. Illustration of network covers detected by different community detection methods for Coauthorships in science network.

columns, accompanied by information in brackets indicating the number of communities, conductance, and the average number of node memberships in the community. The three highest modularity values within each column have been highlighted for emphasis.

5.4.1. Assessing the structural quality of network cover by modularities

The three different modularities are used to evaluate the structural quality of the detected communities. The modularities are designed in different ways and the list of modularities is given in Section 5.1. Tables 3, 4 and 5 present the highest modularity values corresponding to the detected network covers for the selected measures.

Table 3 outlines the modularity values of the M^e, a metric originally proposed by the authors of the EAGLE method in [10], which is employed during their optimization process. In the context of Zachary's karate club network, the modularity values for the GHAC methods are found to be smaller compared to those achieved by EAGLE and SLPA. This observation is primarily due to the use of the M^e to assess network cover outcomes in the EAGLE algorithm, leading to consistently higher values across all networks.

Table 3
Modularity M^e and basic description of network cover detected for real-world networks.

Method	Zachary's karate club	American college football	Coauthorships in network science	High-energy theory collaborations
SL GHAC ($k=2$)	0.138 (15/0.71/1.73)	0.292 (9/0.32/1.04)	0.322 (63/0.41/1.19)	0.239 (1563/0.82/1.15)
SL GHAC ($k=3$)	0.138 (15/0.71/1.73)	0.292 (9/0.32/1.04)	0.326 (57/0.4/1.17)	0.239 (1563/0.82/1.15)
SL GHAC ($k=4$)	0.086 (23/0.95/1.0)	**0.295** (12/0.44/1.02)	0.308 (120/0.8/1.07)	0.211 (2326/0.89/1.11)
CL GHAC ($k=2$)	0.167 (6/0.49/1.18)	0.276 (14/0.39/1.1)	0.383 (17/0.18/1.07)	0.288 (122/0.28/1.27)
CL GHAC ($k=3$)	0.167 (6/0.49/1.18)	0.276 (14/0.39/1.1)	**0.394** (17/0.19/1.05)	**0.319** (285/0.55/1.14)
CL GHAC ($k=4$)	0.086 (23/0.95/1.0)	**0.295** (12/0.44/1.02)	0.334 (108/0.84/1.03)	0.252 (2015/0.94/1.04)
AL GHAC ($k=2$)	0.186 (5/0.47/1.15)	0.269 (17/0.48/1.1)	0.385 (19/0.18/1.07)	0.289 (166/0.3/1.27)
AL GHAC ($k=3$)	0.185 (5/0.47/1.12)	0.269 (17/0.48/1.1)	**0.394** (19/0.18/1.06)	0.316 (282/0.55/1.14)
AL GHAC ($k=4$)	0.086 (23/0.95/1.0)	**0.295** (12/0.44/1.02)	0.332 (107/0.84/1.03)	0.25 (2029/0.94/1.04)
EAGLE ($k=2$)	0.179 (7/0.54/1.48)	0.16 (3/0.24/1.46)	0.39 (17/0.1/1.06)	**0.35** (38/0.15/1.16)
EAGLE ($k=3$)	**0.198** (3/0.43/1.06)	0.258 (6/0.29/1.21)	**0.396** (19/0.15/1.05)	**0.358** (158/0.75/1.09)
EAGLE ($k=4$)	0.086 (23/0.95/1.0)	**0.295** (10/0.42/1.01)	0.341 (100/0.88/1.01)	0.268 (1920/0.97/1.02)
k-CPM ($k=3$)	0.115 (4/0.48/1.06)	0.099 (4/0.35/1.13)	0.322 (63/0.41/1.19)	0.244 (1174/0.56/1.34)
k-CPM ($k=4$)	0.076 (24/0.95/1.06)	0.284 (15/0.45/1.05)	0.239 (148/0.77/1.22)	0.168 (2557/0.87/1.22)
SLPA ($r=0.05$)	**0.189** (2/0.15/1.09)	0.287 (9/0.32/1.12)	0.389 (33/0.17/1.11)	0.303 (461/0.34/1.38)
SLPA ($r=0.1$)	**0.199** (2/0.13/1.09)	**0.297** (9/0.3/1.07)	0.387 (37/0.2/1.09)	0.311 (452/0.34/1.29)
DEMON ($\epsilon=0.25$)	0.1 (5/0.69/1.03)	0.153 (9/0.33/2.12)	0.243 (76/0.67/1.6)	0.16 (1315/0.84/2.11)
DEMON ($\epsilon=0.4$)	0.1 (5/0.69/1.03)	0.191 (11/0.36/1.77)	0.297 (87/0.65/1.22)	0.191 (1634/0.79/1.61)
MNMF nonoverlap	0.118 (6/0.33/1)	0.25 (10/0.25/1)	0.27 (15/0.11/1)	0.278 (43/0.14/1)
MNMF overlap	0.113 (5/0.28/1.09)	0.25 (12/0.28/1)	0.277 (27/0.15/1.07)	0.261 (174/0.53/1.14)
DANMF nonoverlap	0.125 (4/0.19/1)	0.244 (8/0.23/1)	0.285 (18/0.08/1)	0.268 (64/0.17/1)
DANMF overlap	0.114 (6/0.3/1.12)	0.244 (14/0.42/1)	0.262 (40/0.34/1.09)	0.198 (802/0.48/1.31)

Nonetheless, the proposed methods demonstrated competitive modularity values with the CL GHAC and the AL GHAC showing particularly notable performances for other networks.

Inspecting the results for the SL GHAC ($k=2$) and the $k-CPM$ ($k=3$) reveals an identical outcome for Coauthorships in network science. This similarity is further depicted in Fig. 4a, where the cut level, distinguished by a red highlight, corresponds to a dissimilarity value of $4/3$ — an equivalent measure for the SL GHAC and $k-CPM$ ($k=3$).

The M^w modularity metric is designed to address resolution limits by implementing appropriate weighting [44]. A review of the values in Table 4 unveils a trend of a higher number of detected communities within covers exhibiting high modularity. This modularity metric appears to encourage a greater number of communities, leading to network covers that not only possess a higher community count but also exhibit increased conductance values. This pattern is also observable in the progress of modularity values for the CL GHAC represented in Fig. 4b. The modularity's peak is reached earlier during the agglomerative process for the M^w than for the M^e. Both methods (CL GHAC and AL GHAC) outperform the SL GHAC method in achieving higher modularity values, as shown in Table 4. Interestingly, the EAGLE method also achieved high modularity values. In contrast, the NMF-based methods achieved low M^w values in comparison to other methods.

An evaluation of M^{ov} modularity values shows the dominance of the SL GHAC out of the proposed methods when all maximal cliques are used as a base for each real-world network (see Table 5). The highest modularity values were often achieved by the MNMF or DANMF methods, especially for the smaller networks. With the biggest network used in comparison (High-energy theory collaborations), the SL GHAC achieved a higher modularity value than the NMF-based method, where the input parameters values were searched by a hyperparameter tuning process.

The CL and AL GHAC methods not managing to achieve comparably high modularity values to the SL GHAC suggests that the M^{ov} modularity metric is particularly advantageous to the SL GHAC method. The elevated modularity values observed for the Coauthorships in network science and High-energy theory collaboration networks seem to suggest that the M^{ov} modularity metric finished with a higher community count for both the AL GHAC and CL GHAC methods when compared to previous modularities.

5.4.2. Does the minimum clique size for the HAC matter?

The choice of bases for the GHAC method significantly affects the outcomes of the proposed approaches. Notably, the impact of the minimal size of the maximal cliques is evident in Tables 3, 4, and 5. Utilizing a minimum base size of 4 or greater exhibited a positive impact exclusively in the case of the American college football network. For other real-world networks, the best cut finished with a high number of identified communities for $k=4$. This outcome is accountable to the post-processing procedure where nodes not assigned to any community are considered separate isolated communities.

A further comparison was drawn regarding the choice of the input bases in the GHAC - either including all cliques (including edges) or restricting to cliques where triangles represent the smallest base elements. Observations in Table 5 suggest that including edges generally yields more favorable results for modularity M^{ov}. This impact is most noticeable in the High-energy theory collabo-

Table 4
Modularity M^{w} and basic description of network cover detected for real-world networks.

Method	Zachary's karate club	American college football	Coauthorships in network science	High-energy theory collaborations
SL GHAC ($k=2$)	0.194 (15/0.71/1.73)	0.399 (9/0.32/1.04)	0.406 (63/0.41/1.19)	0.351 (1160/0.55/1.33)
SL GHAC ($k=3$)	0.194 (15/0.71/1.73)	0.399 (9/0.32/1.04)	0.408 (57/0.4/1.17)	0.351 (1160/0.55/1.33)
SL GHAC ($k=4$)	0.162 (23/0.95/1.0)	**0.425** (14/0.44/1.03)	0.381 (120/0.8/1.07)	0.312 (2327/0.89/1.11)
CL GHAC ($k=2$)	0.241 (6/0.49/1.18)	0.409 (14/0.39/1.1)	**0.434** (27/0.23/1.1)	0.402 (446/0.41/1.33)
CL GHAC ($k=3$)	0.241 (6/0.49/1.18)	0.409 (14/0.39/1.1)	**0.434** (25/0.22/1.08)	**0.41** (285/0.55/1.14)
CL GHAC ($k=4$)	0.162 (23/0.95/1.0)	**0.425** (14/0.44/1.03)	0.395 (109/0.83/1.04)	0.344 (2017/0.94/1.04)
AL GHAC ($k=2$)	**0.267** (5/0.47/1.12)	0.406 (17/0.48/1.1)	**0.435** (23/0.22/1.09)	0.4 (325/0.41/1.28)
AL GHAC ($k=3$)	**0.267** (5/0.47/1.12)	0.406 (17/0.48/1.1)	**0.435** (23/0.2/1.07)	0.409 (332/0.52/1.15)
AL GHAC ($k=4$)	0.162 (23/0.95/1.0)	**0.425** (14/0.44/1.03)	0.393 (111/0.82/1.04)	0.343 (2035/0.94/1.05)
EAGLE ($k=2$)	**0.261** (11/0.63/1.64)	0.381 (106/0.86/2.99)	0.427 (17/0.1/1.06)	**0.42** (177/0.43/1.19)
EAGLE ($k=3$)	0.249 (4/0.46/1.09)	0.381 (16/0.58/1.43)	0.429 (19/0.15/1.05)	**0.426** (214/0.61/1.1)
EAGLE ($k=4$)	0.162 (23/0.95/1.0)	**0.425** (14/0.44/1.03)	0.393 (108/0.83/1.03)	0.349 (1998/0.95/1.02)
k-CPM ($k=3$)	0.137 (4/0.48/1.06)	0.168 (4/0.35/1.13)	0.406 (63/0.41/1.19)	0.36 (1174/0.56/1.34)
k-CPM ($k=4$)	0.141 (24/0.95/1.06)	0.417 (15/0.45/1.05)	0.328 (148/0.77/1.22)	0.271 (2557/0.87/1.22)
SLPA ($r=0.05$)	0.249 (2/0.15/1.09)	0.42 (9/0.32/1.12)	0.427 (33/0.17/1.11)	0.383 (461/0.34/1.38)
SLPA ($r=0.1$)	0.249 (2/0.15/1.09)	0.419 (9/0.3/1.07)	0.423 (37/0.2/1.09)	0.388 (452/0.34/1.29)
DEMON ($\epsilon=0.25$)	0.121 (5/0.69/1.03)	0.245 (9/0.33/2.12)	0.31 (76/0.67/1.6)	0.239 (1318/0.84/2.11)
DEMON ($\epsilon=0.4$)	0.121 (5/0.69/1.03)	0.31 (11/0.36/1.77)	0.374 (87/0.65/1.22)	0.286 (1634/0.79/1.61)
MNMF nonoverlap	0.044 (13/0.49/1)	0.272 (12/0.28/1)	0.226 (23/0.11/1)	0.196 (171/0.17/1)
MNMF overlap	0.042 (7/0.34/1.33)	0.27 (13/0.35/1)	0.228 (27/0.11/1.05)	0.19 (295/0.59/1.08)
DANMF nonoverlap	0.047 (8/0.37/1)	0.259 (14/0.34/1)	0.228 (32/0.12/1)	0.196 (128/0.18/1)
DANMF overlap	0.059 (16/0.48/1.09)	0.264 (16/0.46.1)	0.213 (35/0.27/1.15)	0.16 (822/0.46/1.29)

Table 5
Modularity M^{ov} and basic description of network cover detected for real-world networks.

Method	Zachary's karate club	American college football	Coauthorships in network science	High-energy theory collaborations
SL GHAC ($k=2$)	0.351 (2/0.19/1.03)	0.303 (4/0.23/1.03)	**0.552** (33/0.33/1.15)	**0.505** (23/0.24/1.01)
SL GHAC ($k=3$)	0.236 (3/0.47/1.03)	0.303 (4/0.23/1.03)	0.524 (34/0.35/1.14)	**0.383** (775/0.53/1.18)
SL GHAC ($k=4$)	0.018 (23/0.95/1.0)	0.211 (14/0.44/1.03)	0.187 (150/0.76/1.24)	0.074 (2506/0.87/1.19)
CL GHAC ($k=2$)	0.286 (3/0.32/1.18)	0.189 (14/0.39/1.1)	0.511 (64/0.41/1.2)	0.296 (1227/0.55/1.51)
CL GHAC ($k=3$)	0.259 (7/0.52/1.24)	0.189 (14/0.39/1.1)	0.51 (64/0.41/1.19)	0.305 (1179/0.55/1.41)
CL GHAC ($k=4$)	0.018 (23/0.95/1.0)	0.211 (14/0.44/1.03)	0.188 (154/0.76/1.28)	0.073 (2562/0.87/1.24)
AL GHAC ($k=2$)	0.264 (5/0.43/1.24)	0.137 (17/0.48/1.1)	0.513 (63/0.41/1.19)	0.323 (1106/0.54/1.39)
AL GHAC ($k=3$)	0.248 (8/0.52/1.3)	0.161 (2/0.19/1.03)	0.512 (63/0.41/1.19)	0.33 (1080/0.54/1.33)
AL GHAC ($k=4$)	0.018 (23/0.95/1.0)	0.211 (14/0.44/1.03)	0.188 (154/0.76/1.28)	0.073 (2570/0.87/1.23)
EAGLE ($k=2$)	0.26 (3/0.24/1.3)	0.094 (1/0.0/1.0)	0.46 (29/0.22/1.1)	0.191 (43/0.16/1.16)
EAGLE ($k=3$)	0.18 (5/0.45/1.15)	0.163 (6/0.29/1.21)	0.467 (31/0.24/1.09)	**0.328** (601/0.5/1.19)
EAGLE ($k=4$)	0.018 (23/0.95/1.0)	0.226 (10/0.42/1.01)	0.173 (173/0.76/1.45)	0.066 (2503/0.87/1.21)
k-CPM ($k=3$)	0.218 (4/0.48/1.06)	0.1 (4/0.35/1.13)	0.512 (63/0.41/1.19)	0.312 (1174/0.56/1.34)
k-CPM ($k=4$)	0.016 (24/0.95/1.06)	0.187 (15/0.45/1.05)	0.186 (148/0.77/1.22)	0.073 (2557/0.87/1.22)
SLPA ($r=0.05$)	0.169 (2/0.15/1.09)	0.248 (9/0.32/1.14)	0.381 (35/0.18/1.15)	0.182 (470/0.34/1.41)
SLPA ($r=0.1$)	0.175 (2/0.13/1.09)	0.282 (7/0.27/1.04)	0.428 (37/0.2/1.09)	0.19 (448/0.33/1.3)
DEMON ($\epsilon=0.25$)	0.142 (5/0.69/1.03)	0.067 (9/0.32/2.35)	0.155 (78/0.65/1.7)	0.052 (1318/0.84/2.14)
DEMON ($\epsilon=0.4$)	0.142 (5/0.69/1.03)	0.084 (11/0.36/1.85)	0.247 (87/0.65/1.22)	0.119 (1634/0.79/1.61)
MNMF nonoverlap	0.378 (3/0.14/1)	0.305 (14/0.34/1)	**0.564** (71/0.26/1)	0.324 (503/0.26/1)
MNMF overlap	**0.432** (9/0.38/1.42)	**0.369** (12/0.28/1.01)	**0.567** (75/0.28/1.21)	0.272 (595/0.3/1.18)
DANMF nonoverlap	**0.392** (3/0.14/1)	**0.343** (9/0.25/1)	0.533 (56/0.23/1)	0.18 (256/0.27/1)
DANMF overlap	**0.43** (3/0.17/1.09)	**0.321** (12/0.32/1.02)	0.54 (78/0.34/1.24)	0.055 (444/0.4/1.74)

rations network for the SL GHAC method. The algorithm indicated a network covers consisted of 23 or 775 communities when edges were included or excluded from the bases, respectively. For context, the SLPA identified 448 communities within the network.

A comparison between CL GHAC and AL GHAC methods with parameters $k \in \{2,3\}$ revealed significant differences, especially in the largest networks. Particularly, methods with a parameter $k=3$ tended to achieve higher modularity M^{e} in Table 3.

6. Conclusion

This research introduces novel dissimilarities, proposed for graph hierarchical agglomerative clustering (GHAC), to detect the overlapping communities within a network. These proposed dissimilarities are based on the CT-distance between vertices and the size of the largest clique within the overlap. In this context, maximal cliques serve as the base elements for the GHAC method. The single linkage strategy employed within the GHAC yields a hierarchy that contains network covers derived from the k-CPM method and extends the k-CPM results with an additional hierarchical community structure.

We have further developed the CL GHAC and AL GHAC methods, which implement complete and average linkage strategies within the GHAC, respectively, and leverage the appropriate dissimilarities. The dendrogram of cliques is evaluated using several modularities to determine the optimal cut level that produces communities. The theoretical contributions of this paper are the generalization of the CPM to the hierarchical structure and the explanation of the connection between the k-CPM and the SL GHAC that leads us to the proposition of other GHAC methods.

When compared to other established algorithms (EAGLE, SLPA, DEMON) and NMF-based methods (MNMF and DANMF) which require extensive parameter tuning, the proposed approach registers higher values in terms of the selected modularities for detecting overlapping communities in certain instances. However, it is important to note that no single modularity measure consistently surpasses all others in effectively evaluating overlapping communities within real-world network contexts.

The study intentionally selected real-world networks to illustrate the hierarchical structures given by the proposed methods. Hierarchical information is one of the valuable insights into network structure. We argue that these networks provide a more accurate representation of practical applications than synthetic networks.

Recognizing the inherent limitations of modularity measures for quality assessment, our future work will include other possibilities for determining the best cut of the dendrogram that reflects the real community structure. In our future studies, we aim to use the LFR benchmark. This will help in evaluating how well our proposed methods can identify communities that match the ground truth, and in assessing the quality of the hierarchy structure. Additionally, we also aim to extend these methods to weighted networks.

CRediT authorship contribution statement

Pavla Dráždilová: Visualization, Resources, Methodology, Conceptualization. **Petr Prokop:** Visualization, Software, Investigation, Formal analysis. **Jan Platoš:** Methodology, Conceptualization. **Václav Snášel:** Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data are available from: http://www-personal.umich.edu/%7Emejn/netdata/. The code for is available https://anonymous.4open.science/r/graph_hierarchical_agglomerative_clustering-C946/README.md.

Acknowledgement

The authors are deeply grateful to the anonymous reviewers and editor for their helpful and constructive comments and suggestions, which assisted in improving this work. This work was supported by SGS, VŠB – Technical University of Ostrava, Czech Republic, under the grant No. SP2024/006 “Parallel processing of Big Data XI”.

References

[1] S. Fortunato, D. Hric, Community detection in networks: a user guide, Phys. Rep. 659 (2016) 1–44.
[2] J. Yang, J. Leskovec, Overlapping communities explain core–periphery organization of networks, Proc. IEEE 102 (12) (2014) 1892–1902.
[3] J. Yang, J. Leskovec, Community-affiliation graph model for overlapping network community detection, in: 2012 IEEE 12th International Conference on Data Mining, IEEE, 2012, pp. 1170–1175.
[4] Y. Jia, Q. Zhang, W. Zhang, X. Wang, Communitygan: community detection with generative adversarial nets, in: The World Wide Web Conference, 2019, pp. 784–794.
[5] G. Palla, D. Ábel, I.J. Farkas, P. Pollner, I. Derenyi, T. Vicsek, k-clique percolation and clustering, in: Handbook of Large-Scale Random Networks, Springer, 2008, pp. 369–408.
[6] G. Palla, I. Derényi, I. Farkas, T. Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, Nature 435 (7043) (2005) 814–818.
[7] Y.-Y. Ahn, J.P. Bagrow, S. Lehmann, Link communities reveal multiscale complexity in networks, Nature 466 (7307) (2010) 761–764.
[8] I. Farkas, D. Ábel, G. Palla, T. Vicsek, Weighted network modules, New J. Phys. 9 (6) (2007) 180.
[9] X. Zhang, C. Wang, Y. Su, L. Pan, H.-F. Zhang, A fast overlapping community detection algorithm based on weak cliques for large-scale networks, IEEE Trans. Comput. Soc. Syst. 4 (4) (2017) 218–230.
[10] H. Shen, X. Cheng, K. Cai, M.-B. Hu, Detect overlapping and hierarchical community structure in networks, Phys. A, Stat. Mech. Appl. 388 (8) (2009) 1706–1712.
[11] V.D. Blondel, J.-L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech. Theory Exp. 2008 (10) (2008) P10008.
[12] Y. Xu, T. Ren, S. Sun, Community detection based on node influence and similarity of nodes, Mathematics 10 (6) (2022) 970.

[13] T. Li, L. Lei, S. Bhattacharyya, K. Van den Berge, P. Sarkar, P.J. Bickel, E. Levina, Hierarchical community detection by recursive partitioning, J. Am. Stat. Assoc. 117 (538) (2022) 951–968.
[14] V. Snášel, P. Dráždilová, J. Platoš, Closed trail distance in a biconnected graph, PLoS ONE 13 (8) (2018) e0202181.
[15] M.T. Schaub, J. Li, L. Peel, Hierarchical community structure in networks, Phys. Rev. E 107 (5) (2023) 054305.
[16] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, Inc., 1988.
[17] I. Derényi, G. Palla, T. Vicsek, Clique percolation in random networks, Phys. Rev. Lett. 94 (16) (2005) 160202.
[18] J.M. Kumpula, M. Kivelä, K. Kaski, J. Saramäki, Sequential algorithm for fast clique percolation, Phys. Rev. E 78 (2) (2008) 026109.
[19] F. Reid, A. McDaid, N. Hurley, Percolation computation in complex networks, in: 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, IEEE, 2012, pp. 274–281.
[20] S.K. Gupta, D.P. Singh, Cbla: a clique based Louvain algorithm for detecting overlapping community, Proc. Comput. Sci. 218 (2023) 2201–2209.
[21] A. Lancichinetti, S. Fortunato, J. Kertész, Detecting the overlapping and hierarchical community structure in complex networks, New J. Phys. 11 (3) (2009) 033015.
[22] X. Wen, W.-N. Chen, Y. Lin, T. Gu, H. Zhang, Y. Li, Y. Yin, J. Zhang, A maximal clique based multiobjective evolutionary algorithm for overlapping community detection, IEEE Trans. Evol. Comput. 21 (3) (2016) 363–377.
[23] H. Boubaker, W. Karoui, Improved overlapping community detection in networks based on maximal cliques enumeration, Proc. Comput. Sci. 176 (2020) 858–867.
[24] K. Asmi, D. Lotfi, A. Abarda, The greedy coupled-seeds expansion method for the overlapping community detection in social networks, Computing 104 (2) (2022) 295–313.
[25] J. Zhu, B. Chen, Y. Zeng, Community detection based on modularity and k-plexes, Inf. Sci. 513 (2020) 127–142.
[26] H.-W. Shen, X.-Q. Cheng, J.-F. Guo, Quantifying and identifying the overlapping community structure in networks, J. Stat. Mech. Theory Exp. 2009 (07) (2009) P07042.
[27] J. Yang, J. Leskovec, Overlapping community detection at scale: a nonnegative matrix factorization approach, in: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, 2013, pp. 587–596.
[28] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, S. Yang, Community preserving network embedding, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, 2017.
[29] F. Ye, C. Chen, Z. Zheng, Deep autoencoder-like nonnegative matrix factorization for community detection, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 1393–1402.
[30] M.A. Porter, J.-P. Onnela, P.J. Mucha, Communities in networks, Not. Am. Math. Soc. 56 (9) (2009) 1082–1097.
[31] J. Xie, S. Kelley, B.K. Szymanski, Overlapping community detection in networks: the state-of-the-art and comparative study, ACM Comput. Surv. 45 (4) (2013) 1–35.
[32] M.A. Javed, M.S. Younis, S. Latif, J. Qadir, A. Baig, Community detection in networks: a multidisciplinary review, J. Netw. Comput. Appl. 108 (2018) 87–111.
[33] C. He, X. Fei, Q. Cheng, H. Li, Z. Hu, Y. Tang, A survey of community detection in complex networks using nonnegative matrix factorization, IEEE Trans. Comput. Soc. Syst. 9 (2) (2021) 440–457.
[34] S. Xing, X. Shan, L. Fanzhen, W. Jia, Y. Jian, Z. Chuan, H. Wenbin, P. Cecile, N. Surya, J. Di, et al., A comprehensive survey on community detection with deep learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[35] T. Chakraborty, A. Dalmia, A. Mukherjee, N. Ganguly, Metrics for community analysis: a survey, ACM Comput. Surv. 50 (4) (2017) 1–37.
[36] J. Yang, J. Leskovec, Defining and evaluating network communities based on ground-truth, in: Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, 2012, pp. 1–8.
[37] M.G. Everett, S.P. Borgatti, Analyzing clique overlap, Connections 21 (1) (1998) 49–61.
[38] L. Yuan, L. Qin, W. Zhang, L. Chang, J. Yang, Index-based densest clique percolation community search in networks, IEEE Trans. Knowl. Data Eng. 30 (5) (2017) 922–935.
[39] V. Snášel, P. Dráždilová, J. Platoš, Cliques are bricks for k-ct graphs, Mathematics 9 (11) (2021) 1160.
[40] S. Landau, M. Leese, D. Stahl, B.S. Everitt, Cluster Analysis, John Wiley & Sons, 2011.
[41] S. Sharma, N. Batra, Comparative study of single linkage, complete linkage, and ward method of agglomerative clustering, in: 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), IEEE, 2019, pp. 568–573.
[42] J.W. Suurballe, R.E. Tarjan, A quick method for finding shortest pairs of disjoint paths, Networks 14 (2) (1984) 325–336.
[43] A. Lázár, D. Abel, T. Vicsek, Modularity measure of networks with overlapping communities, Europhys. Lett. 90 (1) (2010) 18001.
[44] J. Cao, Z. Bu, G. Gao, H. Tao, Weighted modularity optimization for crisp and fuzzy community detection in large-scale networks, Phys. A, Stat. Mech. Appl. 462 (2016) 386–395.
[45] M.E. Newman, Analysis of weighted networks, Phys. Rev. E 70 (5) (2004) 056131.
[46] T.S. Evans, Clique graphs and overlapping communities, J. Stat. Mech. Theory Exp. 2010 (12) (2010) P12037.
[47] J. Xie, B.K. Szymanski, X. Liu, Slpa: uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process, in: 2011 IEEE 11th International Conference on Data Mining Workshops, IEEE, 2011, pp. 344–349.
[48] M. Coscia, G. Rossetti, F. Giannotti, D. Pedreschi, Demon: a local-first discovery method for overlapping communities, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2012, pp. 615–623.
[49] H. Zhang, I. King, M. Lyu, Incorporating implicit link preference into overlapping community detection, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 29, 2015.