IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 September 2022, accepted 9 October 2022, date of publication 12 October 2022, date of current version 24 October 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3213980

SURVEY

Local Community Detection: A Survey

GEORGIA BALTSOU[1], KONSTANTINOS CHRISTOPOULOS[2], AND KONSTANTINOS TSICHLAS[2]
[1]Department of Informatics, Aristotle University of Thessaloniki, 54636 Thessaloniki, Greece
[2]Department of Computer Engineering and Informatics, University of Patras, 26504 Patras, Greece

Corresponding author: Georgia Baltsou (georgipm@csd.auth.gr)

This work was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "2nd Call for H.F.R.I. Research Projects to support Faculty Members and Researchers" under Project 3480. The work of Georgia Baltsou was supported by the Greece and the European Union (European Social Fund-ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning" in the context of the project "Strengthening Human Resources Research Potential via Doctorate Research-2nd Cycle" implemented by the State Scholarships Foundation (IKY) under Grant MIS-5000432.

ABSTRACT Community detection is a flourishing research field with a plethora of applications ranging from biology to sociology. Local community detection has emerged as a promising subfield of research concerned with community identification around a set of seeding nodes. The practical significance of local community detection is important for numerous real-world applications such as protein interactions and targeted advertising. Since 2005, when the first research paper on local community detection appeared, the literature has been vast and difficult to navigate, as each method works best under certain conditions and assumptions regarding the seed nodes and the identification of their community. For this reason, and motivated by the many real-world applications of local community detection, in this paper we provide a comprehensive overview and taxonomy of local community detection algorithms. There are quite a lot of surveys on community detection that make a compendious reference to local community detection. However, they do not achieve a systematic and comprehensive coverage of this particular field. Since the research area of local community detection is quite extensive, it is necessary to categorize and discuss the various methods, techniques, and assumptions used to address the problem. This survey aims to fill this gap and help researchers get a clear overview of the local community detection problem. To this end, we have also gathered the best documented tools and the most commonly used datasets in the local community detection literature to help researchers identify the tools they can use to prove their methods.

INDEX TERMS Algorithms, community detection, local, survey.

I. INTRODUCTION

Various real world systems [1], [2], [3], [4] are often described by networks because it is a convenient way to represent data. In particular, community detection can reveal the hidden structures and functions in networks [5], [6], and therefore attracts the attention of researchers from various fields [7].

As a result, there are many algorithms in the literature which aim at identifying communities in networks. Most of them focus on detecting the partitioning of the entire network into communities, i.e., *global community detection* [8], which implies a global knowledge of the structure of the entire network. Girvan and Newman's (GN) [9] method was the first proposed method for global community detection. Since then, the problem of community detection has attracted the attention of a large part of the scientific community, and a very large number of articles have already been published. More importantly, this large number of articles spans a variety of different disciplines, from computer science to physics, to biology, and to social sciences.

In reality, however, it is more common to know only part of a network, either because of its size or because it is dynamic (e.g., the WWW network) or because one is interested in a particular part of the network (e.g., particular neurons in a brain network). In such cases, it is very difficult or even impossible to obtain the information of the whole network [10]. As a result, *local community detection (LCD)* has attracted the attention of researchers. In general, LCD is applied to find one or a few communities starting from

The associate editor coordinating the review of this manuscript and approving it for publication was Yang Tang.

certain nodes of the network, called seeds. This problem is also known as seed set expansion [11]. The practical importance of LCD is very significant for numerous real-world applications, such as analysing terrorist activities or determining the placement of advertisements in social media [12]. Furthermore, LCD can be applied in web search, where with a few known pages containing similar information, one can generate a larger set of web pages that contain the relevant content related to a particular search query. Besides, in product networks, expanding the starting set allows an analyst to automatically categorize products that are in the same community as the tagged items [13]. In addition, LCD is also critical in biological networks analysis like protein-protein interactions, where LCD helps determine additional members of a protein complex [14], or in human brain co-activation networks, where the discovered communities provide spatial and functional meanings to the analyst. Nowadays, people pay great attention to the flavor profiles of culinary ingredients in order to customize their diets. The network of flavors where nodes represent ingredients and edges are between two ingredients if they have similar flavor compounds, can be used to create new recipes from existing ones by considering the local communities discovered in this network for key ingredients [15]. What is more, LCD is effectively used in collaboration networks where an analyst can determine the membership of a particular person in a working group by detecting local communities [16].

Clauset A. [17] was the first to suggest an LCD method. Since then, several LCD methods have been proposed by researchers. These approaches are not all appropriate for every LCD problem. Many factors need to be evaluated to select the most appropriate [18].

The motivation of this survey stems mainly from the fact that the literature for local community detection is scattered mainly between Physics and Computer Science and there is not a single point of entrance for a new researcher in this area. This results in raising the difficulty of entrance to this area of research. Indeed, to the surprise of the authors when they started working on local community detection, there was no publication (or a survey to be more specific) that would allow us to get to know this research area faster. This was the starting point for envisaging such a survey that could help researchers quickly review existing work and decide which of them fits their own problem.

Additionally, motivated by the numerous practical applications of LCD in real world networks, and with a view to help scientists choose quickly the best fitting method for particular problems, we present here a classification of existing local community detection techniques. While there are some surveys on community detection in general, as far as the authors are aware, there is no specific survey on the important problem of LCD. Therefore, the innovation of our paper lies on the fact that it attempts to fill this gap by providing an assistive taxonomy of existing LCD methods in the literature. Furthermore, the fundamental difference between our survey and other existing surveys, is that our work is dedicated to local community detection techniques and not generally to community detection.

A. CONTRIBUTIONS

The purpose of this review is to organize and categorize the existing approaches to LCD in networks that have been proposed in the literature especially over the last decade. Taking into account past, present, and future trends, the present work aims to help researchers and practitioners understand the field of local community detection with respect to the following aspects:

- **Taxonomy and Review:** We propose a systematic taxonomy of existing local community detection techniques. For each class, we summarize and review representative work.
- **Source Code, Tools and Datasets:** We provide statistics on the availability of the source code of the papers studied and the datasets used. We have also compiled and compared the available and best documented tools that help researchers detect communities in networks, by type, platform, and license.
- **Trends and Future Directions:**As research in local community detection has seen an upsurge in recent years, we discuss current trends in the field and open problems for the future.

The rest of the survey is organized as follows. Section II contains basic definitions related to the current survey domain. In Section III, related surveys on the general problem of community detection are discussed. The classification of the existing LCD approaches is presented in Section IV. In Section V, well-documented tools for community detection are presented. Besides, commonly used datasets in LCD experimental evaluations are presented in the same section. The authors' insights over the Local Community Detection literature as well as future directions are discussed in Section VI. Finally, we conclude in Section VII.

II. DEFINITIONS

In the present work, we assume that a network $G = (V, E)$ consists of a node set $V = \{1, \ldots, n\}$ where $n = |V|$ are the nodes, and an edge set $E \subseteq V^2$ where $m = |E|$ are the edges. In Table 1 we summarize main mathematical symbols used throughout the paper.

Definition 1: A *Community C* is defined as a sub-network of G in which the nodes inside C are more densely connected to nodes within C than to nodes outside C.

Definition 2: *Seed Nodes* or *Seeds* or *Source Nodes* are the nodes that define the community one wishes to discover.

Definition 3: A *Local Community LC* is defined as the community to which the seed nodes belong. Thus, a network G can be divided into LC and the rest of the network $G{-}LC = U$. Figure 1 shows the commonly accepted definition of the local community in a network G.

Definition 4: *Community Detection* is the procedure to find all communities C_i with $i \in \mathbb{Z}$ forming the network G

TABLE 1. Main mathematical symbols used throughout the paper along with their description.

Symbol	Description
G	network
V	node set of a network
n	the number of nodes in a network
E	edge set of a network
m	the number of edges in a network
C	community
E_{in}	the number of edges with two endpoints in C
E_{out}	the number of edges with one endpoint in C
M	local modularity metric
$\phi(C)$	conductance metric of C
s	the seed node
$d(x)$	the shortest path distance between nodes x and s
$n(x)$	the neighbour nodes of node x
d_x	the degree of node x
A	the adjacency matrix of a network

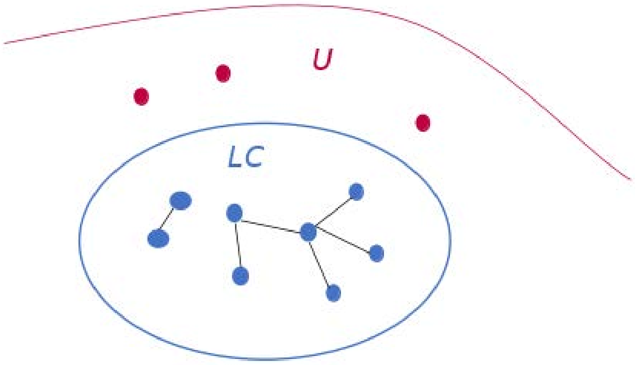

FIGURE 1. General scheme of local community definition.

and $\bigcup C_i = V$. *Local Community Detection (LCD)* is the process of finding only a subset of the communities which constitute the network. Thus, only a part of G is known at a local level and not the whole community structure of it.

At this point it should be noted that LCD is a term used in the literature for two similar problems. On the one hand, LCD refers to finding the community to which a seed node (or group of seed nodes) belongs. On the other hand, LCD refers to a method that uses local information to discover all communities in the network [19]. In the taxonomy of a following section, algorithms of both categories are presented, as they can both be used to uncover the community of particular node(s).

III. RELATED WORK

There are many surveys in the literature on the problem of community detection. However, there is no survey that focuses exclusively on methods for detecting local communities. Nevertheless, in some of these works there is a subsection describing the techniques of LCD, but this is not comprehensive. More specifically, Table 2 lists in chronological order some of the community detection surveys published in the last decade. In particular, Papadopoulos et al. [20] investigated the computational and memory performance of community detection algorithms specifically in social media. They studied six different local methods, focusing mainly on their scalability compared to global methods. In addition, Dhouioui Z. and Akaichi J. [21] reviewed community detection literature that can be used to detect overlapping communities. Under these circumstances, they dedicated one of the five algorithm categories to local expansion algorithms. Specifically, eleven different local community detection algorithms for overlapping communities were reviewed. Furthermore, Harenberg et al. [22] evaluated community detection algorithms with datasets that have ground-truth communities. Two of the thirteen algorithms evaluated focus on local expansion. Fortunato and Hric [7] gave a comprehensive overview of the problem of community detection in networks. However, this review did not specifically address local methods, but only mentioned five of them. Besides, Maivizhi et al. [23] investigated another aspect of the community detection problem, that of the tools used. In particular, the authors provided an overview of the tools available for community detection and mining. In addition, Javed et al. [24] surveyed community detection approaches and their related real-world applications. The authors categorized community detection algorithms into four different classes. Among these surveyed algorithms, the authors have selected some local algorithms that fall into the first four classes, so no section is devoted to local approaches. Moreover, Ma et al. [25] presented an experimental evaluation of local community metrics. More specifically, they divided the metrics into degree-based and similarity-based ones and conducted experiments on eight different metrics. Besides, Dakiche et al. [26] presented a classification of community evolution tracking methods in dynamic social networks into four main categories. The authors discussed three approaches for detecting local communities that fall into these categories. Furthermore, Dilmaghani et al. [19] proposed a scheme to explore the concept of locality at each step of a community detection process. The authors suggested a four-step community detection process (flow) and reviewed works that follow this flow. More specifically, they reviewed nineteen different works that use the concept of locality in one or more steps of their flow. Huang et al. [27] presented a survey of approaches to community detection in multilayer networks. The work discussed included three local approaches for multilayer networks. Souravlas et al. [18] proposed a three-class categorization of community detection methods in social networks. Among the discussed methods there were some local approaches. Furthermore, Meena et al. [28] survey also presented community detection techniques applied to social networks. They also reported other applications where community detection is used. The methods discussed are divided into three categories. A few local approaches that fall into these categories were presented. More recently, Su et al. [29] presented a survey of deep learning methods for community detection. These methods were divided into three categories. Popular datasets and evaluation metrics were also discussed.

The overview includes a discussion of a corresponding local method.

All this related literature present indicative methods for local community detection but they are very far from covering it completely as intended in this survey. Therefore, there is a big gap with respect to the extent and depth of the coverage of local community detection methods from the existing survey papers on community detection that this survey aspires to fill. Of course, there is an undoubtable connection between the general concept of community detection discussed in the related work and LCD, but there are also important applications and specific methods that exploit aspects of local approaches. Thus, there is a need for an overview of methods related to local community detection.

To the best of the authors' knowledge, the present study is the first comprehensive study dealing exclusively with local community detection techniques in networks.

IV. LOCAL COMMUNITY DETECTION TAXONOMY

It has been quite a while since Clauset A. [17] firstly proposed the formal definition of local community detection in networks in 2005. Subsequently, many approaches have been proposed until today making LCD a very active research problem in networks. In this survey, we provide an overview of the work that has been proposed on this problem since the introduction of LCD in 2005. However, we mainly focus on works from the last decade (2012-2022), since they are the most widely used today. We also present the most popular of the earlier approaches (2005-2011).

These methods can be classified into different groups considering various criteria. Therefore, the classification of the approaches is not absolute, but differs with respect to the aim of the taxonomy. Algorithms proposed to detect local communities are classified here according to the type of networks in terms of static, temporal/dynamic or fully streaming ones. Besides, we classify the methods according to the type of approach to detect local communities: greedy or non-greedy ones.

The three categories of local community detection algorithms in terms of network type are: 1) the methods used in static networks, 2) the techniques used in dynamic/temporal networks, and 3) those used in fully streaming networks. Static networks are those whose components, i.e., nodes and edges, remain the same. Dynamic or temporal networks are networks that change over time. Thus, both nodes and edges can be added or removed, resulting in evolving communities. For both forms of networks, the initial or past information about them is always available. More precisely, for the static networks the former is self-explanatory. For the dynamic networks, this means that all past information about the network is available when the network changes. In contrast, edges in graph streams are only available once. More specifically, when an edge e_i is processed in stream S, the past edges cannot be accessed again, and the subsequent edges in the stream are unknown, i.e. $\forall j \neq i$, e_j is not accessible when e_i arrives in stream S.

TABLE 2. Surveys on community detection in the literature.

Survey	Year	Focus On
Papadopoulos S., Kompatsiaris Y., Vakali, Ath. et al. [20]	2012	Performance in terms of complexity and memory of community detection in the context of social media
Dhouioui Z., Akaichi J. [21]	2013	Overlapping communities
Harenberg St., Bello G., Gjeltema, L.R. et al. [22]	2014	Overlapping and disjoint communities in large-scale real networks with ground truth communities
Fortunato S., Hric D. [7]	2016	General aspects of community detection problem and discussion on popular methods used
Maivizhi R., Sendhilkumar S., Mahalakshmi G. S. [23]	2016	Survey of tools for detection and mining of communities
Javed M., Aqib Y., Muhammad Sh. L. et al. [24]	2018	Overlapping and disjoint communities in various domains of real life and corresponding applications
Ma L., Chiew K. et al. [25]	2018	Experimental evaluation of local community detection metrics
Dakiche N., Benbouzid-Si T., Fatima S. et al. [26]	2019	Tracking community evolution over time in dynamic social networks
Dilmaghani S., Brust M., Danoy G. et al. [19]	2021	Scheme to investigate the concept of locality at each stage of a community detection workflow
Huang X., Chen D., Ren T. et al. [27]	2021	Multilayer networks
Souravlas S., Sifaleras A., Tsintogianni M. et al. [18]	2021	Classification of community detection approaches into: (a) bottom-up, (b) top-down, and (c) data structure- based, concerning social media
Meena P., Pawar M. et al. [28]	2021	Comparative analysis of community detection methods and Applications used in community detection algorithms
Su X., Xue S. et al. [29]	2022	Survey on community detection with deep learning

Figure 2 shows the general taxonomy of local community detection techniques that the present survey suggests.

Table 3 briefly presents all works reviewed for the survey, classified according to the proposed taxonomy of Figure 2. We have used the same colors for each class as in Figure 2 to help the reader understand in which class each work is classified. The arrows show the flow of further classification.

A. TECHNIQUES FOR STATIC NETWORKS

Techniques related to static or dynamic networks can be divided into two main groups of algorithms in terms of the approach they follow to expand a community. These groups are *greedy* and *non-greedy*. Regarding greedy algorithms, these methods focus on the technique used to add nodes: A node is selected for inclusion in a community if and only if

TABLE 3. Works reviewed in the present survey classified according to their class. We use the same class colors as in 2 for reader convenience.

Static Networks	Greedy	Focus on starting Node(s)		Bagrow J. et al [30],Xu B. et al [12],Zhang T. et al [31],Chen Q. Et al [32],Moradi F. et al [33],Xia S. et al [10],Fanrong M. et al [34],Wang P. et al [35],Hamann M. et al [36],Ding X. et al [37],Tasgin M. et al [38],Xu X. Et al [39],Guo K. et al [8],Luo W. et al [40],Liu J. et al [16],Aghaalizadeh et al [41],Hu Y. et al [42],Ji P. et al [43]
		Focus on Quality metric		Clauset A. [17],Luo F. et al [44],Chen J. et al [45],Huang J. et al [46],Branting L. et al [47],Wu Y. et al [48],Ngonmang B. et al [49],Kudelka M. et al [50],Ma L. et al [51],Cui W. et al [52],Fagnan J. et al [53],Wu Y. et al [54],Chang C. et al [55],Xia Z. et al [56],Kanawati R. et al [57],Zhao W. et al [58],Liu J. et al [59],Interdonato R. et al [60],Luo W. et al [61],Ni L. et al [62],Li X. et al [63],Guo K. et al [64],Baltsou G. et al [65],Shang R. et al [66]
	Non-Greedy	Flow Propagation		Orecchia L. et al [67],Panagiotakis P. et al [68],Veldt N. Et al [69],Veldt N. Et al [70]
		Random Walk	PageRank	Andersen R. et al [71],Gharan S. et al [72],Zhu Z. et al [73],Kloumann I. et al [74],Yin H. Et al [14],Hollocou A. et al [75]
			Heat kernel	Chung F. et al [76],Kloster K. et al [77],Shun J. et al [78]
			Local spectral	Mahoney M. et al [79],Li Y. et al [13],He K. et al [80],Laarhoven T. et al [81],Bian Y. et al [82],Jeub L. et al [83],Bian Y. et al [15],Freitas S. et al [84],He K. et al [85],Zhang Y. et al [86],Fan X. et al [87],Shi P. et al [88],Luo D. et al [89]
Dynamic Networks	Snapshot model			Takaffoli M. et al [90],Zakrzewska A. et al [11],Gao W. et al [91],Nathan E. et al [92],DiTursi D. et al [93],Javadi S. et al [94],Guo K. et al [95],Papadopoulos A. et al [96],Liu J. et al [97]
	Temporal smoothness			Hu Y. et al [98],Rigi M. et al [99],Fu D. et al [100],Hu Y. et al [42]
Graph Streams				Liakos P. et al [101],Baltsou G. et al [102]

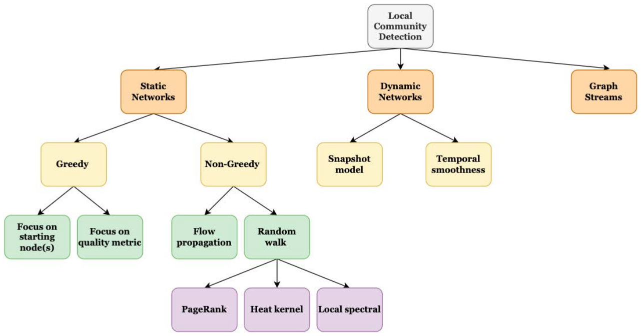

FIGURE 2. Proposed taxonomy of local community detection techniques.

that node has the maximum value of a quality function that assigns a value to each node [36]. The majority of existing algorithms belong to this category. The group of non-greedy algorithms includes techniques that do not attempt to greedily expand a community, but use various methods to find local communities. Most algorithms in this category use random walks.

1) GREEDY TECHNIQUES

In greedy techniques there is a general scheme that the algorithms follow and describes the knowledge of the graph at each step. This scheme consists of three main steps and is depicted in Figure 3. LC represents the local community under investigation in a network G, and is usually initialised with the starting node(s). The set N contains nodes adjacent to at least one node of LC. Nodes that are not adjacent to any element of LC form the set U, the unexplored nodes. Apart from the first set classification, LC can be further divided into two subsets: LC_{core}, which contains nodes that have no neighbour in the set N, and $LC_{boundary}$, which contains nodes with at least one edge to a node in the set N. Thus, $LC_{core} = LC - LC_{boundary}$. Greedy local community detection methods are also called greedy seed expansion methods. In the following, we provide the description of the basic greedy scheme for LCD based on Chen et al [45].

1) Initialize LC with seed node n_0
2) Initialize N with the neighbours of n_0
3) The value of the quality function Q of the initial community is set to 0
4) Find the node $n \in N$ which maximizes the quality function used in the algorithm
5) If the insertion of the above node to the set LC increases the quality criterion, then the node is moved from N to LC and the subset N is updated
6) The former process is repeated until there is not any node n whose inclusion to LC increases the quality function
7) *(Optional step)* Check LC for nodes that need to be removed according to a filtering method
8) Now LC contains the local community of n_0

The former process can be generalized to more than one seed nodes.

In general, there are two main problems to solve in greedy techniques: 1) starting node selection and 2) the quality criterion [47]. The first describes the problem of which node

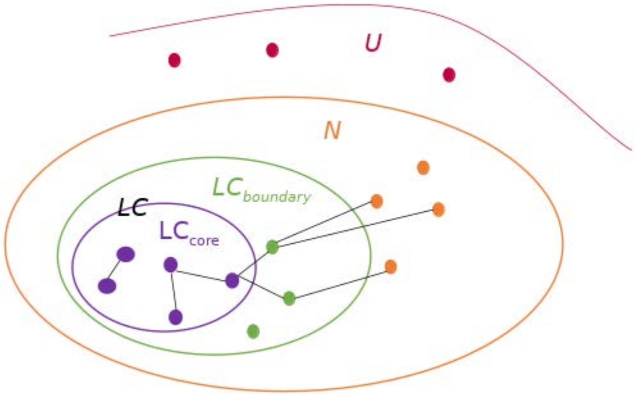

FIGURE 3. Scheme of local community definition for greedy techniques.

should be selected as the starting node (seed) to begin the process of discovering a local community. The second concerns the quality criterion that is checked after each node is added to the community. This criterion is also crucial for the termination of the process, i.e. when to stop adding nodes to the community.

Based upon the two main problems arising in greedy techniques for LCD, we categorize them in two groups; a) greedy methods for LCD which try to improve the node selection process IV-A1.a, and b) greedy methods for LCD which try to optimize the quality function. IV-A1.b

a: METHODS FOCUSING ON NODE SELECTION

This category contains algorithms that focus mainly on optimizing the selection of the seed node(s). There are two main problems that arise when using greedy seed expansion, and both are related to seed node(s) selection. The first problem is the *seed-dependent* problem. It means that the location of the seed node(s) affects the quality of the detected local community. The second problem is known as *seed-invalid* problem, which means that many LCD algorithms cannot ensure that the seed node(s) participate in the final detected local community [37]. The methods in this category follow different approaches to selecting the starting node, with a view to overcome the above problems. Most of these approaches focus on finding important nodes that are good representatives of the community to which they belong.

Bagrow et al. [30] aim to better represent the actions that members of a network would take to identify their own communities. To achieve this, they propose to start the LCD process from an $l - shell$ where l is the distance from starting node to all shell nodes N. The suggested method is based on a parameter, α, which controls when to stop the spread of the $l - shell$. The performance of their approach highly depends on the parameter l as well as on the starting node, since the resulting communities differ a lot if the algorithm starts from border nodes. The speed of the algorithm is also very low [10]. Experiments are conducted only on small networks, i.e. the largest one is the polbooks dataset [103] with 105 nodes.

Xu et al. [12], acknowledge the problem of the limited ability to deal with large-scale networks suggesting a method of community discovery based on the seed expansion, which is inspired by the ego-centered theory. More specifically, they suggest dividing the network into k communities and find the k-top leader nodes. These k-top leaders are then selected as the seed nodes. In each iteration the extension can start with these new seed nodes. Nodes are added in the community if its similarity (calculated based on common neighbors) with the community is above a threshold σ. They only experiment with real networks, and besides discovering the local community, their method also finds the linked nodes that are loosely related to the community. A conclusion made by the authors about their method is that threshold σ has a great impact on the community size, i.e. the largest the threshold, the smaller the community. The largest dataset (Sina micro-blog [104]) that they experiment with has 310.000 nodes.

Another method that focuses on the starting node is the one proposed by Zhang et al. [31]. The authors' motivation is the fact that if the position of the starting node is at the boundary of the communities, the local community structures detected have low accuracy. Thus, they propose to count the degree centrality of each node in the communities and select the nodes with maximum degree as core nodes. The first step is to find the communities of the network and then identify the core nodes of each community. From these nodes, they start the LCD process to obtain the new communities of core nodes. They choose to test their method on real networks. However, sometimes their proposed method fails to detect communities with high accuracy. The largest dataset that they experiment with is Political blogosphere [105] with 1733 nodes.

Moradi et al. [33] propose a different parameter-free method for finding better seed nodes. In particular, the authors suggest a seeding algorithm that first computes the similarity score of each node based on link prediction, and then it uses a biased graph coloring algorithm to improve the seeding process. There are various proposed similarity scores but they prefer using metrics that require only access to the node's neighborhood. The experiments performed on real networks of products, collaboration and social networks, have shown that the proposed method successfully detects communities of good quality in less time than without their proposed seeding procedure. The largest dataset that they use to test their method is a dataset collected by the authors from Soundcloud with 5187722 nodes.

Furthermore, Xia et al. [10] suggest the ILCDSP algorithm with a view to overcome the seed-invalid problem. This algorithm uses a selection probability for the candidate nodes at each step, so that the nodes with a high selection probability are more likely to be included in the community. They experiment with both synthetic and real networks. The largest network that the authors use to evaluate their method is a synthetic one generated by the LFR method [106] with 5000 nodes. Compared to Clauset [17] and Luo et al. [44] the proposed method achieves better overall F1 score, precision and recall values for the small number of real networks that were used. Although their suggested algorithm improves the accuracy of community detection, it is not stable enough in terms of not achieving the best precision values in all datasets.

Moreover, Fanrong et al. [34] discuss the seed-invalid problem, where the seed node sometimes does not participate in the detected community. Thus, they propose to start community expansion not from a single node, but from a maximal clique containing the seed node and suggest an algorithm named Local Community Detection algorithm based on Maximum Cliques extension (LCD-MC). Their experiments with both synthetic and real networks show that the proposed method can detect high-quality communities. The largest datasets that the authors use are synthetic generated by the LFR method [106] with 5000 nodes each. More precisely, their method compared to Clauset A. [17] and Wu et al. [48] presents better results in terms of F1 score and NMI (Normalized Mutual Information) [107].

Hamann et al. [36] also suggest to start the expansion of the community from the largest clique in the neighbourhood of the seed node. Besides, they introduce Triangle-based Community Expansion (TCE) as an alternative strategy for greedy community expansion. Their method utilizes the fact that edges inside communities are usually embedded in triangles. Thus, the authors propose triangle-based edge evaluation to decide which node should be added to the community in each step. The proposed approach is applicable to both unweighted and weighted networks and overlapping communities. They conduct experiments with synthetic as well as real networks and compare their method with other algorithms, including Infomap [108], a global community detection algorithm. They are comparing their method to seven different local community detection approaches. Four of them are density based ([45], [71], [109], [110]) and the others ([34], [46], [53]) start the LCD process from a maximal clique (e.g., triangles). Their largest datasets are from Facebook [111] with at most 41536 nodes. They conclude that any algorithm which assumes a maximal clique rather than a single node, leads to higher quality communities in terms of F1 score metric, especially in real networks. However, the choice of the expansion algorithm should always depend on the network type. In addition, the source code of their proposed method is publicly available.

Besides, Ding et al. [37] aim to overcome both the seed-dependent and the seed-invalid problems, and suggest replacing the seed node with the core member of the target community. The core node is a node that has the strongest relationship with the starting node and also higher centrality than the starting node. Next, the core member is taken as the initial community and then they extend it based on community relation strength. Their method is called RTLCD. The authors conduct experiments on both synthetic and real networks and find that their suggested method is more robust to the seed-dependent problem and the seed-invalid problem compared to [17], [44], [45], [48], [34], and [112] in terms of F1 score, NMI and NCR (Node Coverage Rate). Furthermore, their method tends to find more ground-truth community members. The largest dataset that this method is tested on is the DBLP network [113] with 317080 nodes.

Tasgin et al. [38] focus on the boundary nodes of the communities and propose an LCD method based on label propagation. In this method, after identifying the boundary nodes between communities and ranking them, a node with the highest score among its neighbors propagates its label and finally, the communities are extracted from the network. The suggested method is better than existing ones in terms of NMI only when the communities are subtle. However, experiments with synthetic and real networks (co-purchase, collaboration and social) revealed the main weaknesses of this method, namely instability in determining the final communities and weak performance in identifying a reasonable number of communities. The largest dataset that this method is tested on is the youtube network [113] with 1134890 nodes. The source code of the proposed method is publicly available.

In addition, Xu et al. [39] present a distributed LCD algorithm name DLCD-CCE, based on community center extension for large complex (weighted) networks. Their motivation is to provide an algorithm to an as wider as possible scientific and analytic community. The algorithm is evaluated using a Spark-based prototype system to verify its accuracy and scalability. The results of experiments on co-purchase, social and collaboration networks, show that DLCD-CCE has better accuracy, stability, and scalability compared to typical local community detection algorithms like [17], [44], and [32], in terms of precision, recall and F1 score, and effectively overcomes the problem that existing algorithms are sensitive to the location of initial seeds. The largest dataset that this method is tested on is the youtube network [113] with 1134890 nodes.

Guo et al. [8], propose the InfoNode algorithm aiming at overcoming quality and stability deficiencies in overlapping community detection. It uses local degree central nodes and Jaccard coefficient to detect core members in the network. The detected core members are then treated as seed nodes, which guarantee that they are central nodes of the communities. Next, the node with the highest degree among the seed nodes applies the fitness function strategy for pre-expansion. In the last step, the top k nodes with the best performance in the pre-expansion process are expanded by the fitness function with internal force between nodes. Three parameters have to be defined to use InfoNode: α (used to control the scale of communities detected), k (used to select the top k nodes that have the best performance in the pre-expansion process), and ϵ (used to select seeds in the network). Experiments with synthetic and real networks have shown that InfoNode can accurately discover communities. The largest dataset that this method is tested on is the CA-GrQc network [114] with 14845 nodes. More precisely, InfoNode was compared to several methods like [109], [32], and [75], upon many different datasets (social, co-purchase, power, collaboration), and was found that it can accurately uncover local communities in terms of EM (extension of Modularity) [115] and NMI.

Furthermore, Liu et al. [16] propose the algorithm HqsMLCD to detect multiple overlapping communities for a given starting node. Their motivation is to define an LCD method that is not sensitive to the position of the seed node but it is sensitive to the local structure of the seed node. HqsMLCD first finds seeds of higher quality than the given seed node, and then expands these seeds sequentially to obtain multiple communities that are likely to overlap. They test their method on real networks and find that it successfully discovers high-quality communities (in terms of F1 score), compared to [75], [80], and [116]. They experiment with three real datasets (co-purchase, collaboration and social), the largest of which is the LiveJournal network [113] with 3997962 nodes.

In addition, a recent approach by Aghaalizadeh et al. [41] proposes a local and deterministic method named LCDR for detecting communities in social unweighted and weighted networks using core nodes. The authors are motivated by the fact that the quality of the detected communities depends on the selected important nodes as community cores. In this method, core nodes of the networks that are responsible for influencing other remaining nodes are detected and then communities are formed around these nodes using a Sorensen similarity index. The core nodes are selected according to their importance, which is calculated considering the first and second degree neighbors of each node. The experiments performed on both synthetic and real networks (co-purchase, collaboration, social) show that the proposed method is stable and detects high-quality communities quite quickly, compared to other approaches like [108] and [38], in terms of modularity and NMI. The largest dataset that this method is tested on is the youtube network [113] with 1134890 nodes.

Finally, Hu et al. [42] propose three hybrid local centrality measures which combine degree and denseness of node neighborhood to evaluate how good a node is as a seed, and then apply them to a local algorithm for selecting seeds in networks. They focus on local seeding because of its efficiency. Experimental results with the proposed method, called S-LM, in both synthetic and real networks show that their method performs well. However, it is inferior to using the conductance centrality in networks where the difference between degrees is small and the community structure is clear. The largest dataset that this method is tested on is the LiveJournal network [113] with 3997962 nodes. The results of the experiments were evaluated with F1 score and overlapping modularity [117].

Further research papers that belong in this category include Chen et al. [32] (LMD method), Wang et al. [35] (MAGA-LC method), Luo et al. [40] (LCDNN method) and Ji et al. [43] (CAELCD method) which are not analyzed due to the similarity with the previously discussed methods.

b: METHODS FOCUSING ON QUALITY FUNCTION

The local community approaches of the present category focus mainly on community valuation. More precisely, the starting node of the local community we want to discover is already given. The goal is to extend the community beyond the starting node by optimizing a quality function.

Clauset A. [17] was the first to propose an approach for LCD, motivated by the fact that in many cases there is a lack of global knowledge of a graph's topology. Thus, the author propose a measure of community structure independent of global properties, the local modularity R. This metric represents the fraction of boundary edges which are internal to the local community. The proposed algorithm merges the neighbour nodes one by one, that increase the value of R, until the community size reaches a predefined size. This means that it is not a parameter-free approach. The suggested method is used in both synthetic and real networks (co-purchase, collaboration). The largest dataset that this method is tested on is the Amazon.com recommender network with 409687 nodes. The discovered communities usually have high recall but low accuracy (compared to the ground-truth communities), as the algorithm is sensitive to the position of the source node. It is also possible for this method to identify multiple communities for the seed node as it explores the network. Furthermore, the resulting community may contain many outliers.

Chen et al. [45] propose the metric L to decide whether a new node is going to be added in the local community, aiming at avoiding outliers. This metric is the ratio between the community internal relation (based on node degrees), to the community external relation (based on node degrees). Based on this metric, a two-phase algorithm which does not require initial parameters, is suggested to identify the local community of starting nodes. Experiments conducted on real networks (football, co-purchase) prove the accuracy and the effectiveness of the proposed method compared to the ground-truth communities in terms of precision, recall and F1 score. The largest dataset that this method is tested on is the Amazon.com network collected by [44] with 585283 nodes. However, when the starting node is on the boundary of another community, the proposed method may not recognize its proper community.

LTE algorithm is suggested by Huang et al. [46] based on a quality metric, named similarity-based tightness. The authors focus on avoiding a priori assumptions of network properties and predefined parameters like other methods. Tightness metric consists an adaptation of the cosine similarity function and computes the internal similarity between nodes in a community. The proposed method can be used in both unweighted and weighted network as well as for overlapping and non-overlapping communities. Experiments on both real (social, football, co-purchase) and synthetic networks showed that the suggested approach achieves good performance in efficient time compared to methods [17], [44], [109] in terms of precision, recall, NMI and Generalized Normalized Mutual Information (GNMI) [109]. The largest dataset that this method is tested on is the Amazon.com network collected by [44] with 585283 nodes. However, the suggested method is seed-dependent in networks without a clear community structure.

Branting et al. [47], focus on improving node selection in the community expansion phase independently to the termination and filtering stages. The authors compare different local community detection methods, which are classified into two categories, xenophobic and non-xenophobic: the former try to maximize (or minimize) internal (or external) connectivity, while the non-xenophobic algorithms discard external connectivity. They also propose an evaluation criterion for LCD algorithms that takes into account the relative centrality of nodes within the target community, Normalized Utility-Weighted Recall (NUWR). After an experimental evaluation with synthetic and real networks (power, collaboration, football, social), the authors conclude that there is no single LCD algorithm, but instead algorithms should be selected based on the properties of the graph and the nature of the community to be detected. The largest dataset that this method is tested on is the Western US Power Grid [118] with 4941 nodes.

Moreover, Wu et al. [48], propose a method to determine the local community structure by analyzing the link similarity between the community and a node, motivated by the fact that most existing methods are dependent on seed node and impose too strict a policy in the expansion phase. Link similarity of a node is regarded as the ratio between the number of common edges of the node and the community and the number of neighbors of the node. Inspired by the fact that elements in the same community are more likely to have common edges, the community structure is explored heuristically by prioritizing nodes that have high link similarity with the community. To improve the quality of the community structure, a three-step process is also used. Experiments with both synthetic and real networks (football, citation) show that the suggested method detects stable and high-quality communities compared to [17], [44], and [45] in terms of F1 score, although it is only suitable for undirected networks. The largest dataset that this method is tested on is the Cora Citation Network [119] with about 17000 nodes.

Ngonmang et al. [49] try to optimize a well-known quality metric, modularity, in order to detect local overlapping communities, overcoming the problem of failure when the starting node is at the boundary of community. More specifically they are based on the work of Chen et al. [45], which suggests a metric that considers the densities of intra-community edges and outer edges rather than their number, and try to optimize it. The problem with the former method is that it may not recognize the local community of a starting node that is on the boundary of another local community. The method that is suggested here is called IOLoCo. Experiments with synthetic and real networks (collaboration, co-purchase, social) show that their proposed method can be successfully used to improve recommendation systems in social networks, compared to [17], [44], and [45] and evaluated by performance index [120]. The largest dataset that this method is tested on is the Amazon.com network collected by [121] with 548552 nodes.

Furthermore, a method called GMAC for LCD is presented by Ma et al. [51], focusing on the seed-depended problem. This method estimates the similarity between nodes via examining the neighborhood of nodes and reveals a local community by maximizing its internal similarity while minimizing its external similarity. The Compactness-Isolation (CI) Metric is used to decide the inclusion of new nodes in the local community. According to this metric, the nodes of a good local community should have high internal similarities and low external similarities, resulting in a high CI value. The authors conducted experiments on synthetic and real networks (social, football, collaboration, power, co-purchase) and proved that their method is more insensitive to seeds than approaches like [46] and [17]. In addition, the suggested method discovered local communities more accurately in terms of precision, recall and F1 score. The largest datasets that the authors use are synthetic by the LFR method [106] with 10000 nodes each.

Moreover, Fagnan et al. [53] focus on accurately identifying communities, outliers, and hubs in social networks. The main component of the algorithm is the T-metric, which evaluates the relative quality of a community by considering the number of internal and external triads (i.e., 3-node cliques/triangles) it contains. Intuitively, this metric favours nodes that form many triads with nodes inside the community and few triads with nodes outside the community. Experiments on real networks with ground truth communities (co-purchase, football, social, politics) verify that the proposed method performs as well as, or in some cases better than, state-of-the-art methods like [17], [44], and [45], in terms of Adjusted Rand Index (ARI) [122]. The largest dataset that they experiment with is Political blogosphere [105] with 1733 nodes. The authors consider their method applicable to directed networks too, even though they did not experiment on such networks.

Besides, Wu et al. [54] examine an interesting aspect of LCD methods: the free-rider effect. This effect is related to the observation that most existing metrics tend to include irrelevant subgraphs in the detected local community. More precisely, they study the existing goodness metrics and provide theoretical explanations for the causes of the free-rider effect. They also develop a query-oriented node weighting scheme to reduce this effect, and a corresponding algorithm for LCD. Experimental results in synthetic and real networks (co-purchase, social, collaboration) show that the proposed method efficiently reduces the free-rider effect and discover ground-truth communities more accurately in terms of F1 score when compared to other approaches like [17], [44], [71], [79]. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes. Besides, the authors propose two variants of their suggested method for overlapping and multiple communities respectively.

Xia et al. [56] propose a new local modularity metric G and a two-stage algorithm for LCD based on it, considering networks lacking global information. The method they follow adds nodes to the community until G stops increasing. G is defined as the ratio between the number of internal edges in the community and the sum of internal edges in the community, edges with only one endpoint in the community, and the number of edges between the neighboring nodes' set of the community. The suggested method was tested on synthetic and real networks (football) and proved to be quite effective in terms of precision, recall and F1 score compared to [17] and [44] methods. The largest dataset that this method is tested on is NCAA football network [9] with 179 nodes.

In addition, Kanawati et al. [57] explore and evaluate the combination of different local modularity functions to identify the node-centric communities in weighted networks. Various ensemble-based approaches are proposed and implemented, including approaches based on ensemble rankings and ensemble clustering. Experiments are conducted on synthetic and real networks (social, co-purchase, football) with ground-truth communities and show that ensemble ranking-based approaches outperform other approaches considering ARI and NMI values. The largest dataset that the authors use is a synthetic one by the LFR method [106] with 2000 nodes.

Zhao et al. [58] propose an approach for LCD via edge weighting. More specifically, they first design a new measure of node similarity considering the weights of neighbouring nodes. They also develop an edge weighting method based on this similarity measure. Then, they define a new goodness metric, Closeness-Isolation (CI), to quantify the quality of the local community by integrating the edge weights. In this algorithm, they discover the local community by giving priority to those shell nodes that have maximum similarity with the current local community. The CI metric is defined as the ratio between the sum of the weights of all edges in the community and the sum of the weights of all edges with one node in the community and the other in the shell. The suggested approach has been evaluated with both synthetic and real networks (social, football, co-purchase) and achieves good performance in terms of precision, recall and F1 score, compared to [17], [44], and [51]. However, the real networks used are quite small. The largest dataset that this method is tested on is NCAA football network [9] with 179 nodes.

Liu et al. [59] are motivated by the fact that there is no algorithm that guarantees finding an optimal local community structure and develop a method for LCD with a given starting node. This method has as its first step finding the most similar node adjacent to the starting node and forming the initial local community C together with the starting node. The similarity is calculated based on the number of common neighbours between the two nodes. Then, the connectivity degree of the nodes belonging to the neighbours of C is calculated and the node whose connectivity degree is maximum to C is added when the modified local modularity measure is increased. The modification refers to the exception when the external edges of a community are zero. More precisely, the local modularity proposed by Luo F. et al [44] is defined as follows:

$$M = \frac{E_{in}}{E_{out}}$$

where E_{in} is the number of edges with two endpoints in C and E_{out} is the number of edges with one endpoint in C. Liu J. et al [59] modify the local modularity as:

$$M' = \frac{E_{in}}{E_{in} + E_{out}}$$

The proposed method was tested on synthetic and real networks (social, football, co-purchase, politics) and, in terms of precision, recall and F1 score, exhibited good results compared to [17] and [44]. The largest dataset that this method is tested on is the Amazon.com network collected by Liu et al. [44] with 585283 nodes.

Besides, Interdonato et al. [60] propose the first method for LCD in multilayer networks (ML-LCD). Multilayer networks model multiple but different interactions among nodes. For example, in social computing, a person often has multiple accounts in different social networks, and in fact, nowadays it has become important to link distributed user profiles of the same user from multiple platforms. Therefore, in the presented work, a greedy algorithm is proposed to find a community shared by all networks. The authors provide three definitions of the objective function of the ML-LCD problem, corresponding to different ways to incorporate intra-layer and cross-layer topological features. They also provide the source code of the method, which is also tested on seven real networks (transportation, biological, collaboration and social). The suggested approach was compared with both single-layer LCD methods, ([45] and [13]), as well as multilayer global community detection methods. The results show that the method successfully detects multilayer local communities. The largest dataset that this method is tested on is the DBLP network [123] with 83901 nodes.

Moreover, Luo et al. [61] propose two LCD algorithms, *DMF_M* and *DMF_R*, based on the local modularity metrics as suggested by Luo et al. [44] and Clauset [17], respectively. The former methods consider the characteristics of the local community during its creation process. These two algorithms divide the detection process into three stages and employ different dynamic membership functions for each stage to find local communities. The results after experiments on synthetic and real networks (social, co-purchase, football) show that the detected local communities are of high quality compared to methods of Luo et al. [44] and Clauset [17], in terms of precision, recall and F1 score. However, the proposed method is only applicable to undirected networks, and the algorithm is sensitive to seed node's position. The largest dataset that this method is tested on is the Amazon.com network [113] with 334,863 nodes.

Given a node that belongs to multiple communities, the approach of Ni et al. [62], focuses on finding the communities to which it belongs according to local information. The authors propose a framework named LOCD, for overlapping communities in three steps: first, a group of nodes in different communities to which the node belongs is identified, then the representative members of the group are selected, and finally the communities to which they belong are determined.

Experiments with synthetic and real networks (co-purchase) lead to high quality communities considering precision, recall and F1 score, compared with methods like MULTICOM [75]. LOCD is proved to be simpler and more efficient than MULTICOM. The largest dataset that this method is tested on is the Amazon.com network [113] with 334863 nodes.

Furthermore, Li et al. [63] investigate the LCD problem in multi-layer networks concerning the trust relation. The framework proposed by the authors is called MTLCD and is based on the selection of nodes to be included in social network communities according to the trust relation. Users who are trusted can influence others according to their preferences, since they want to connect with influential and trustworthy people. Through experiments with real multilayer networks (biological, satelite, social, mobile network), the results show that the proposed algorithm is highly competitive for multilayer social networks compared to other methods in terms of modularity. However, in non-social networks, where the trust relation between nodes cannot be computed well, the results deteriorate as the social trust increases. The largest dataset that this method is tested on is the Mobile QQ Zone network [124] with 562062 nodes.

An LCD algorithm based on local modularity density (LCDMD) is proposed by Guo et al. [64], with a view to limit sensitivity to seed node selection and unstable communities problems. More specifically, the algorithm divides the process of local community formation into a core area detection phase and a local community expansion phase according to the density of the community tightness based on the Jaccard coefficient. In the core area detection phase, the modularity density is used to ensure the quality of the communities. In the local community expansion phase, the influence of the nodes and the similarity between the nodes and the local community are used to determine the boundary nodes to reduce the sensitivity to the selection of the starting nodes. The source code of the proposed algorithm is publicly available. Experiments with synthetic and real networks (social, co-purchase, word, collaboration, transportation) lead to high accuracy and stable communities in terms of precision, recall, f1 score and conductance, compared to several methods like [14], [17], [44], [116], and [75]. The largest dataset that this method is tested on is the roadNet-CA [125] with 1965206 nodes.

More recently, the Hint Enhancement Framework (HEF) has been proposed by Baltsou et al. [65]. In particular, this framework provides an efficient method for detecting better quality local communities of predefined important nodes called hints. In many cases the choice of seed(s) incorporates external knowledge that attaches to these nodes an additional importance for their community. This knowledge, may be derived from an expert on the domain, or may arise from the network's side information and it constitutes the motivation for the this work. HEF applies a two-step procedure to discover the community of hints: 1) it modifies the network by enhancing the hints using reweighting or rewiring strategies to materialize the preference, and 2) it applies local community detection algorithms to the modified network from step 1. The proposed method is applied on both unweighted and weighted networks. Extensive experiments with synthetic and real networks (social, collaboration, biological) prove the accuracy of the framework to detect local communities compared with local methods like [36], [46], [126] and a global one [108], considering precision, recall and F1 score. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes.

Shang et al. [66] propose an interesting approach to extend local communities, with a view to solve the problem of poor algorithm results caused by low-quality seeds. The algorithm suggested is called HSEI and is based on higher-order structure and edge information. More specifically, first, different ways of selecting the first node to join a local community are used depending on the motif degree of the seed. Second, a new motif-based modularity function is proposed to extend the local community so that the extended community is more tightly connected. A new motif-based central node of the community is defined to extend the central part of the local community. For the edge of the community and the area with sparse connections, the edge information is used to mine the membership strength between nodes and communities to obtain more complete members of the local community. They experimented on various real networks including social, collaboration, co-purchase and www networks. The largest dataset that this method is tested on is the LiveJournal network [113] with 3997962 nodes. The approach was compared with others like [13], [37], [61], and [40] in terms of precision, recall and F1 score, and outperformed the other methods even when the seed nodes were considered of low quality.

Further research papers that belong to this class include Kudvelka et al. [50], Cui et al. [52], Chang et al. [55] which are not analyzed due to the similarity with other, previously discussed works.

2) NON-GREEDY TECHNIQUES

This group includes algorithms that find local communities in ways other than greedily adding a node at each step. Such techniques require a stopping criterion to define the boundary of the community. Conductance is a widely used metric for determining the boundary of a community. Methods that use conductance try to minimize it, since it measures the fraction of edges that leave the community. Conductance is defined for a community C as [101]:

$$\phi(C) = \frac{adj(C, V) \setminus C}{\min\{adj(C, V), adj(V \setminus C, V)\}}$$

where $adj(C, V) = |\{(u, v) \in E : u \in C, v \in V\}|$. The conductance minimization is an NP-hard problem [127] thus non-greedy techniques try to approximate a solution (e.g., [81]). The techniques used in non-greedy LCD are categorized into Flow-Propagation (FP) and Random-Walk (RW) based ones.

a: FLOW-PROPAGATION BASED TECHNIQUES

The methods of this category work as follows: the seed node emits a stream that shares flow with the adjacent nodes, i.e. its neighbours. Each of these nodes, stores a flow to pass on to its neighbors but may also return a part of the flow to the first node. The probability $p(x)$ of a node to belong to the same community as the seed node s, is analogous to the stored flow of the node x. This probability is:

$$p(x) = \rho^{-d(x)},$$
$$p(x) \leqslant \frac{\sum\limits_{y \in n(x)} p(x)}{|n(x)|}$$

where $x \in V$, $d(x)$ the shortest path distance between node x and s, ρ the average ratio of local links to node degree value and $n(x)$ the neighbours of node x.

The work of Orecchia et al. [67] presents the first local graph-partitioning algorithm that combines flow and spectral methods. The authors propose this combination to achieve better results when searching for low conductance cuts. In particular, they show how to locally find a $\tilde{O}(\frac{1}{\gamma})$ approximation to the conductance given a starting set that overlaps the cut by a γ fraction. Their approach can be generalised to weighted networks. Although this method is theoretically efficient, it relies on a complicated variation of Dinic's algorithm [128], which is difficult to implement in practice.

The FlowPro algorithm is proposed by Panagiotakis et al. [68] with a view to provide a useful community detection tool for a simple user of a social network, which is impossible to know the entire graph structure. More precisely, in each iteration of the main process of the algorithm, the starting node propagates a flow that is shared among its neighbors. Each node can store a portion of the received flow, propagate it to its neighbors, and send it back to the starting node. When the algorithm converges, the flow stored in the nodes belonging to the community of the initial node is usually higher than the flow stored in the rest of the nodes of the graph, since the stored flow of a node is analogous to $p(x)$, and thus forms the desired community. The algorithm does not require access to the entire network to proceed. FlowPro has the additional ability to remove and add edges to s to increase $d(x)$ for nodes x that do not belong to $C(s)$ (e.g., removing bridges) or decrease $d(x)$ for nodes x that do belong to $C(s)$. This feature increases the convergence and performance of FlowPro, as experiments with many synthetic (208 synthetic networks in total) and real networks (collaboration social, www) have shown, compared to [109]. FlowPro detected communities more accurately and efficiently than [109]. Besides, FlowPro seems to perform better against the seed-invalid problem than methods like [109]. The source code of the proposed method is also provided. The largest dataset that this method is tested on is the WWW network with 325730 nodes. The proposed method was tested also on networks with low overlapping communities and can be extended to weighted networks.

Veldt et al. [69] introduces the SimpleLocal algorithm that begins with a reference set reflecting an important part of the network and seeks for a better conductance set nearby. Their method is simple to implement and can take advantage of many maximum flow algorithms [129]. Experiments were conducted on real networks (collaboration, biological) and accurately discovered local communities considering precision and recall. However, the suggested method relies on repeatedly solving numerous maximum flow problems, without taking advantage of the similarity between consecutive flow problems. An improvement on the SimpleLocal algorithm, called FlowSeed, is presented in Veldt et al. [70] by the same authors. In this work, the authors develop a framework that allows users to place strict constraints and soft penalties on excluding specified seeds from the result set, depending on the user's level of confidence for whether or not each node should belong to the result set.Experiments on real networks (social, co-purchase) show the robustness of the proposed method compared to others like [71], [77], and [69], in terms of F1 score and execution time. The largest dataset that these methods were tested on is an MRI network [130] with approximately 18000000 nodes.

b: RANDOM-WALK BASED TECHNIQUES

The general idea of RW-based methods is that if a given network has a community structure, a random walker should be trapped in a community for a relatively long time before leaving it. This arises from the high density of edges within communities and the sparse connections between communities. In the algorithms in this category, the paths of random walkers are repeatedly sampled around the starting node(s) until a convergent probability distribution for the visits of random walkers to the nodes is obtained. Then, this probability distribution is considered as a measure of the similarity between the starting node and its neighbors. Nodes with higher similarity values indicate that they are more likely to be placed in the seed community [131]. Most algorithms in this category can be classified into three main groups: 1) PageRank, 2) Heat-kernel and 3) Local spectral based.

c: PAGERANK BASED TECHNIQUES

Personalized PageRank (PPR) is based on a random walk with restart and is the most commonly used score for non-greedy LCD. Given a parameter $\alpha \in (0, 1)$, it is assumed that a random walk $X_0, X_1, \ldots$ starts uniformly at random from the starting set S and at each step moves from a node u to a node v with probability $\alpha \frac{A_{uv}}{d_u}$ and restarts from a uniformly at random selected node of S with probability $1 - \alpha$. A is the adjacency matrix of the network and d is the degree of a node. For all $t \geq 0$ and $v \in V$ it holds:

$$Pr(X_{t+1} = v) = (1 - \alpha)\frac{1s(v)}{|S|} + \alpha \sum_{u=1}^{n} \frac{A_{uv}}{d_u} Pr(X_t = u).$$

There is a unique stationary distribution p for the Markov chain $(X_t)_{t \geq 0}$, which is the limiting distribution of X_t when

$t \to \infty$. This distribution satisfies the linear system:

$$p(v) = (1-\alpha)\frac{1s(v)}{|S|} + \alpha \sum_{u=1}^{n} \frac{A_{uv}}{d_u} p(u). \tag{1}$$

The vector p is known as the Personalized PageRank (PPR) associated with the seed set S. However, solving the above linear system is computationally inefficient [75]. Therefore, methods for approximating PPR have been proposed in the literature. In practice, experimental results show that a few iterations of the above fixed point equation are sufficient to obtain a very good ordering of the nodes [74]. In general, the methods in this category attempt to solve the conductance minimization problem locally, where the running time depends only on the volume of the output set.

Andersen et al. [71] suggested a method called PageRank Nibble that improves the work of Spielman et al. [132]. More specifically, they suggest a local partitioning algorithm using a variation of PageRank with a specified starting distribution and show that the nodes' ordering produced by a PageRank vector, reveals a cut with small conductance.

An evaluation of different variants of the PageRank method can be found in Kloumann et al. [74] focusing on heuristics that one can control in practice. In particular, the authors show that standard PageRank performs better than degree-normalized, personalized PageRank (DN PageRank). DN PageRank is adopted by several competing PageRank-based community detection methods, including the PageRank Nibble method by Andersen et al. [71]. Their approach is evaluated using real networks with ground-truth communities (collaboration, co-purchase, social) and the authors conclude that almost all PageRank performance improvements result from only two or three iterations of the PageRank update rule. The largest dataset that this method is tested on is the youtube network [113] with 1134890 nodes. However, the highest value of recall is achieved when a large proportion of target community nodes is used as seeds, e.g., 10% in the case of the collaboration network.

A different approach is proposed by Yin et al [14] where the Motif-based Approximate Personalized PageRank (MAPPR) algorithm is presented as an adaptation of the classical Approximate Personalized PageRank (APPR) method. The aim of this work is to find the local community that the seed node belongs, which has the minimal motif conductance, a generalization of the conductance metric tailored to network motifs. The authors consider as motif any small connected graph (such as a triangle). Experiments on both synthetic and real networks (collaboration, co-purchase, social) show that the proposed framework MAPPR successfully detect local communities compared to the ground truth, in terms of precision, recall and F1 score. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes. The suggested method can be applied to directed networks too.

Moreover, Hollocou et al. [75] propose MULTICOM, a method to detect multiple local communities around a given seed set that may overlap. MULTICOM finds multiple local communities by iteratively finding new seed sets and determining local communities on that basis. The algorithm, whose implementation is publicly available, consists of three main steps. In the first step, the local algorithm is used to find a community for each node in the seed set. In the second step, the scoring function is used to assign a vector to each node in the network. The obtained vectors are clustered using the DBSCAN algorithm to obtain multiple communities. The algorithm is tested on both synthetic and real networks (social, co-purchase, collaboration) and compared to other approaches like [71], [133], and [13] in terms of F1 score. The largest dataset that this method is tested on is the youtube network [113] with 1134890 nodes. However, in the suggested method the number of communities detected is highly related to the input parameters, making it not so efficient as other methods like the approach of Ni et al. [62].

Similar research papers to the ones described above include Gharan et al. [72] and Zhu et al. [73].

d: HEAT-KERNEL BASED TECHNIQUES

The heat kernel is another type of graph diffusion that is useful for discovering communities near a starting node. The heat kernel method uses the Taylor series expansion of the exponential function of the transition matrix.

After theoretically analyzing the heat kernel diffusion property in a previous work [133], Chung et al. [76] propose a randomized Monte Carlo method to estimate the diffusion. More specifically, they present an algorithm that takes as input a graph and a boundary condition and outputs a vector that is a good approximation to the solution of the linear SDD (Symmetric Diagonally Dominant) system with boundary conditions. The suggested method can be generalised to weighted networks too.

Besides, Kloster et al. [77] are the first to propose a deterministic method for computing a heat kernel diffusion in a graph. More precisely, the authors present a relaxation method (*hk*-relax) to solve a linear system for estimating the heat kernel diffusion, where the heat value of each node represents the probability of association. Thus, it consists a useful approach compared to similar ones like [76] They show, after an experimental evaluation with synthetic and real networks (collaboration, social, co-purchase), that the heat kernel outperforms the personalized PageRank of [71], with higher values in the F1 score. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes. They also provide the source code of their approach.

A parallel version of heat kernel algorithms, among others, is presented by Shun et al. [78], which besides being tested on the large network of Friendster [113] with 65608366 nodes, is also tested on a very large network from Yahoo [134] with 1413511391 nodes.

e: LOCAL SPECTRAL BASED TECHNIQUES

Algorithms in this category apply spectral techniques to detect local communities. In particular, these methods require first obtaining the embedding of the nodes and then applying the vector clustering method to the embedding matrix.

Mahoney et al. [79] suggest a locally-biased analogue of the second eigenvector that can be used to compute an approximation to the best partition near an input seed set. They also provide an empirical evaluation of their method called LocalSpectral, on synthetic and real networks (social, collaboration) and show that it can be applied to finding locally-biased sparse cuts around an input node set in small social and information networks. The largest dataset that this method is tested on is a collaboration network [135] with 379 nodes.

In addition, the LEMON (Local Expansion via Minimum One Norm) algorithm is presented by Li et al. [13]. The algorithm is based on the extraction of a sparse vector y in the span of the so-called local spectral subspace of the graph around the seed set S. Its source code is also publicly available for use. The previously mentioned vector y is used as a scoring function. The LEMON algorithm is an iterative algorithm, since the nodes with the highest scores are used to expand the seed set S and find a new vector y. The iteration stops when the conductance starts to increase. Experiments with both synthetic and real networks (co-purchase, collaboration, social) prove the effectiveness and efficiency of LEMON in finding communities with high accuracy, compared to local methods like [77], [136], and [74], and certain global community detection methods, considering the F1 score. The largest dataset that this method is tested on is the orkut network [113] with 3072441 nodes. In interesting insight concerns how the size of the seed set affects the algorithm's performance. In general, as the seed set size increases, the seed expansion algorithm performs better. Nevertheless, in LEMON only a small proportion of the target community nodes is needed in order to accurately detect a local community, i.e. 2 to 3 nodes.

Moreover, a method for local overlapping community detection is proposed by He et al. [80]. The method is called LOSP and its source code is publicly available. More precisely, the authors suggest extracting the local community by searching a sparse vector from the local spectral subspaces using $l1$ norm optimization. Experiments with real-world networks (co-purchase, collaboration, social) show that LOSP outperforms in terms of F1 score methods like [77] and [13]. In contrast to LEMON [13], which needs more than one seed node to execute, LOSP can achieve high F1 score values even with a single seed node. The largest dataset that this method is tested on is the LiveJournal network [113] with 3997962 nodes. An extension of this work is also proposed by the same authors in He et al. [85]. However, the quality of detected communities of the suggested method is sensitive to the position of the seed node in the community.

Besides, an LCD method based on network motifs (LCD-Motif) is proposed by Zhang et al. [86], which incorporates higher-order network information. LCD-Motif uses the local expansion of a seed set to identify the local community with minimal motif conductance, a generalization of the conductance metric for network motifs. Different from PageRank-like diffusion methods, LCD-Motif finds the community by searching for a sparse vector in the span of the local spectra such that the seeds lie in its support vector. The authors evaluate their approach on both synthetic and real networks (citation, biological, social, food chain) with good and comparable results to several state-of-the-art approaches like [13], [74], [77], [136], and [14] in terms of precision, recall and F1 score. The largest dataset that this method is tested on is the LiveJournal network [113] with 3997962 nodes.

Finally, Shi et al. [88] propose a Locally-Biased Spectral Approximation (LBSA) approach to identify all latent members of a local community from very few seed members, which improves the suggested algorithm by Li et al. [13]. In addition, this method is the first to use the Lanczos iteration [137] for the local community detection problem, a classical method for computing eigenvalues (the spectra of the matrix). They also use the heat kernel as a sampling method instead of detecting communities directly. Experimental results with several synthetic and real networks (co-purchase, collaboration, social) show that the proposed method outperforms state-of-the-art LCD algorithms like [71], [77], [80], and [13] in terms of F1 score and Jaccard index. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes. The source code of their method is provided.

Similar random walk based algorithms can be found in Luo et al. [15], [82], [83], [84], [87], [89].

B. TECHNIQUES FOR DYNAMIC/TEMPORAL NETWORKS

The methods for LCD in dynamic/temporal networks can be divided into two groups: 1) Snapshot model and 2) Temporal Smoothness. In the first group, snapshots of the evolving network are available at different points in time. At each of these points in time (timestamps), several changes (additions/deletions of nodes/edges) may occur. Because of these changes, the communities may also change. The algorithms belonging to this category focus on updating the communities in each snapshot considering the community structure of the previous snapshot. Temporal smoothing methods focus on finding local communities over time and updating them after each atomic change, i.e., addition/deletion of nodes/edges.

1) SNAPSHOT MODEL

Takaffoli et al. [90] propose an incremental community mining approach called incremental L metric, in order to detect more stable communities compared to previous approaches. The main contribution of their work is to adopt the static L-metric approach [138] to compute dynamic communities. Community mining starts at each snapshot with the communities found in the previous one. The communities found in different snapshots are then matched based on their similarity (i.e., the L-metric) and grouped as instances of the communities evolving over time. The main assumption of the L-metric is that a community has fewer connections from its edge nodes to the unknown part of the graph. Although the experiments were only performed with a real network, they resulted in meaningful communities compared to other methods like [139] in terms of a modified modularity metric. The largest dataset that this method is tested on, is Enron email network [125] with 87273 nodes.

Furthermore, in Zakrzewska et al. [11] a dynamic algorithm is proposed to expand the seed set by updating the fitness score of each snapshot incrementally, focusing on efficiency. The work focuses on maintaining a community centered around the seed. Therefore, it is necessary to track the order in which the nodes were added. Thus, the proposed method ensures that the order of fitness scores remains monotonically increasing and then restarts the algorithm. The process is finished when there is no node whose addition to the community increases the fitness score. The choice of fitness score metric is left to the user. Experiments with real networks (social, email) show that the proposed method is faster than re-computation approaches and that the performance improvement is greatest when low-latency updates are required in terms of precision, recall, ratio of the fitness scores in the dynamic algorithm vs. those obtained by re-computation and the ratio of the size of the community output by the former methods. However, it holds that the suggested algorithm is not time-efficient. The largest dataset that this method is tested on is the DBLP network [113] with 317080 nodes.

An interesting approach is presented by Nathan et al. [92]. Their main goal is to tie together community detection and centrality by studying how personalized centrality metrics can be used for LCD in dynamic networks. The authors are the first to update the personalized centrality vector each time the network changes, and then determine the new local community based on the updated centrality vector. More specifically, they combine Katz and PageRank centrality values to obtain local communities of a network. Katz centrality values count the number of weighted walks in a network starting from a node, penalizing longer walks with a user-selected parameter α. PageRank, as mentioned earlier, is also a walk-based centrality metric that assigns a high score to nodes visited by a large number of random walks in the network. Once the personalized Katz or PageRank centrality scores are calculated, the local community is formed from the nodes with the highest centrality scores. The main drawback of this method is that it requires a priori knowledge about the size of the community. Experiments with synthetic and real networks (social, email) show that the proposed method results in similar high-quality communities as static re-computation approaches, but is more efficient in terms of both time and number of iterations required. The largest dataset that this method is tested on is the youtube network [140] with 3223589 nodes.

Moreover, the work by DiTursi et al. [93] aims to find communities with stable membership over time, where members interact mainly with each other rather than with the rest of the network. To model this intuition, the authors propose the temporal conductance measure, an extension of the already well-known conductance metric. Thus, they suggest a method called PHASR to find the temporal community with the lowest conductance. Evaluations on synthetic and real networks (road, communication, www) show that the proposed algorithm scales better than existing alternatives like [71] and [90] and achieves significant runtime reduction. The largest dataset that this method is tested on, is a preferential attachment synthetic network [141] with 15000 nodes. PHASR, is related to [90], with the difference that PHASR achieves the discovery of communities with lower conductance since it considers the full timeline as opposed to considering only consecutive time steps.

Javadi et al. [94] suggest using the concept of leader nodes to detect dynamic communities, as they assume that communities are formed around them. They define the node with the highest degree of centrality as the leader. Experiments have shown that their method can effectively detect communities in both real (social, communication) and artificial dynamic networks. However, as the size of the dynamic networks increases, the time efficiency decreases dramatically. The largest dataset that this method is tested on, is the Enron email network [125] with 87273 nodes. Leader nodes have also been used by Gao et al. [91] to detect a network's community structure with their proposed method called EvoLeaders.

In addition, the method of Guo et al. [95] first finds the starting nodes of the community using a metric called the local fitness of the nodes. Next, a static algorithm is used to obtain communities in the initial snapshot of the network. Finally, node contribution is proposed to incrementally reveal communities in non-initial snapshots of the network with a suggested method called DyCDNC. The authors perform experiments on synthetic and real networks (email, routers) to prove the accuracy of their method for detecting local communities compared to methods like [142] and [143] considering modularity and NMI. The largest dataset that this method is tested on, is Enron email network [125] with 87273 nodes.

More recently, Papadopoulos et al. [96] extended the PHASR [93] algorithm to conform to the Apache Spark engine distributed processing standard. An approximation method for computing the personalised PageRank vector in the refinement step is also proposed. Performance evaluation results with synthetic and real networks (communication, web, road) have shown that the proposed approach is scalable by increasing the degree of parallelism.

Since a starting node can belong to multiple communities, in [97] Liu et al. extend their static algorithm [16] to the dynamic network to solve this problem. In fact, they are the first to propose a method for multiple LCD in dynamic networks, called HqsDMLCD. Their method achieves comparable results to the static method in real networks (co-purchase, collaboration, social), in terms of F1 score, conductance, community coverage and number of detected communities. The largest dataset that this method is tested on is the LiveJournal network [113] with 3997962 nodes.

2) TEMPORAL SMOOTHNESS

Hu et al. [98], propose a local algorithm (LDM-CET) for dynamic community detection that focuses only on the part of the network that changes at each time step, since network data usually does not change dramatically in a short time. First, a static community detection algorithm is executed, and then a personalized PageRank approach is applied. The starting nodes are selected based on a strategy that tracks the behavior of dynamic communities that construct a partial evolution graph. In general, the PageRank approach is chosen because it runs fast. However, the chosen strategy for finding seed nodes is impractical because it spends a lot of time checking whether a node is a local-minimal conductance node or not. Experimental results with synthetic and real networks (co-authorship) show that the proposed method tracks the community structure well when a network does not change dramatically, in a more efficient way than the compared methods like [139]. The largest network that the authors use to evaluate their method is a synthetic network generated by the LFR method [106] with 1000000 nodes.

An interesting approach is presented by Rigi et al. [99], where a method for detecting local communities inspired by geometric active contours is proposed. Geometric active contours are widely used in machine vision to detect objects in 2D images. They are known for their speed, autonomous and unsupervised nature, and ability to track dynamic objects. The proposed model introduces and uses the derivative-based concepts of curvature and gradient of the boundary of a connected subgraph in networks. Then, a velocity function based on curvature and gradient is proposed to determine whether the boundary of a community should evolve to include a neighbouring candidate. A framework is proposed to approximate derivatives in graphs. This framework is tested on real networks (social) and leads to local communities with high accuracy compared to [30] and [79] in terms of conductance, precision, recall and F1 score. The largest network that the authors use to evaluate their method is Facebook graph FB-JHK of John Hopkins University [111] with 5180 nodes.

Besides, method L-MEGA is suggested by Fu et al. [100], which is based on motif-based clustering. The authors use the multi-linear PageRank vector by edge filtering and motif push operation, and then apply an incremental sweep cut to obtain the local community. Experimental analysis of synthetic and real networks (rating, communication, contact) shows that the proposed method can detect high-quality communities in an efficient manner, compared to both static and dynamic state-of-the-art methods like [144] and [14] considering conductance and triangle density (i.e. the ratio of triangles in the returned local community). The largest network that the authors use to evaluate their method is the contact interaction network [145] with 10972 nodes.

Finally, based on static local seeding, Hu et al. [42] propose a dynamic local seeding algorithm to handle dynamic networks by considering only the nodes affected by network changes. More precisely, the authors suggest a technique to update the centrality values of nodes involved in network change. Experiments with synthetic networks show that the proposed method, called D-LM, is quite fast, but not when many nodes are affected by changes. The authors use the F1 score, coverage and modularity in order to evaluate the suggested method. The largest network that the authors use to evaluate their method is a synthetic one by the LFR method [106] with 1000000 nodes.

C. TECHNIQUES FOR GRAPH STREAMS

As far as the authors are aware, there are only a few papers in the literature on LCD in graph streams, since the specific problem has engaged the scientific community only in the last two years.

More specifically, Liakos et al. [101] are the first to propose a streaming graph community detection algorithm, which they call CoEuS (source code is publicly available). The algorithm aims to expand seed sets from nodes to communities under the constraints posed by the streaming model, which dictates that only a single access to the stream is possible and the working memory is limited. The proposed approach has been evaluated on synthetic and real networks (social, co-purchase, collaboration) and leads to comparable results with methods for detecting local communities without streaming like [13] and [80], but utilize the entire graph structure, considering F1 score. The largest dataset that they experiment with is Friendster [113] with 65608366 nodes.

Moreover, Baltsou et al. [102] and its extension Christopoulos et al. [146], describe a framework, which is used to strengthen the vicinity of the seed set (called anchors) by exploiting the fact that the seed set is of central importance for the evolving community. Informally, the anchors are considered the core of their community because they contribute in some way to its formation and definition, regardless of their topological properties within the network. The multi-step framework that is suggested, firstly applies a static algorithm to discover the initial anchor's community and then for each incoming edge change in the influence range of the anchor, the anchor's community is updated. With a view to discover the most stable anchor's community, the authors suggest using a node rewarding method. That is, for each update, the stable edges in the anchor's influence range are rewarded by a weight increase. Experiments are conducted on synthetic datasets along with three proposed rewarding methods, and the results are compared with the case where no rewarding method is used. The authors' findings indicate that all three dynamic methods with rewards outperform the dynamic method without rewards in terms of recall, precision and F1 score. The largest network that the authors use to evaluate their method is a synthetic one by the RDyn generator [147] with 5000 nodes.

Tables 4-12 contain all publications discussed in the present survey, separated by their class. Each Table contains the references of a class together with the name of the proposed algorithm (if any), some important features, and the availability of the source code.

V. TOOLS AND DATASETS USED FOR LOCAL COMMUNITY DETECTION

A. TOOLS

Since network analysis is a very important research topic in recent years, many tools have been developed to help researchers uncover important properties of networks. As can be inferred, these tools are not specifically designed for detecting local communities, but for detecting communities in general. Of course, researchers can use such tools for LCD, and here we present several that are either packages based on graphical user interfaces (GUIs) or libraries in scripting/programming languages [23], [148]. In Table 13 we present in alphabetical order the most popular tools used in network community detection, along with their type (GUI or library), the platform requirements to be installed and their licenses.

1) GRAPHICAL USER INTERFACES

As can be deduced, GUIs are easier to use than scripting/programming languages because they consist of a graphical interface that guides the user. Here we present the most commonly used ones, which are also well documented.

a: GEPHI

[149] It is a free open source platform for interactive network visualization. Users can interact and manipulate structures, colors, and shapes to discover hidden properties of the graph. Communities can be detected using the Louvain algorithm [150]. The Louvain algorithm attempts to greedily optimize modularity by randomly moving nodes in multiple layers from one community to another.

b: CYTOSCAPE

[151] This is an open source software platform for visualizing complex networks and integrating them with any type of attribute data.

c: GRAPHVIZ

[152] It is an open source graph visualization software. It contains various graph designs for displaying networks in interactive mode.

d: SocNetV

[153] It is a free open source software for social network analysis and visualization. This tool provides various metrics for graph and network cohesion, as well as numerous layout models. It also implements community detection algorithms such as triad and clique census. Finally, it offers famous datasets for social network analysis for use.

e: PAJEK

[154] It is a software program for analysis and visualization of very large networks, which is free for non-commercial use. It helps to calculate various network metrics and detect community structures. Communities are found using the Louvain and Visualization Of Similarities (VOS) methods. It also includes several network layouts.

f: CFinder

[155] It constitutes a free software for finding and visualizing overlapping network communities, based on the Clique Percolation Method (CPM) [156]. CFinder uses spring layouts to visualize graphs.

g: VISONE

[157] It is a free tool for social network analysis and visualization. It implements the Louvain algorithm for community discovery.

h: NetMiner

[158] It is not an open source tool. NetMiner can be used to analyze and visualize network data and implements several algorithms to detect community structures such as Edge betweenness, Blondel, Eigenvector, Label propagation and modularity. In addition, it supports a wide range of visualization layouts.

i: NodeXL

[159] It is a tool for displaying graph data, performing network analysis, and exploring networks visually. It supports multiple social network data providers that import graph data into a spreadsheet format.

2) SCRIPTING/PROGRAMMING LANGUAGES LIBRARIES

Scripting/programming language libraries require knowledge of the particular language to be used. Therefore, they may be more difficult to learn than GUIs, but they are much more powerful and extensible.

a: networkX

[160] It is a Python package for creating, editing and studying the structure, dynamics and functions of complex networks. It contains many community detection algorithms such as Louvain, k-cores, label propagation and others.

b: IGRAPH

[161] It is a library for network analysis and visualization in R, Python, and C/C++. It contains several algorithms for community detection, such as Edge betweenness, Infomap, Label propagation, Louvain method and Random walk method among others.

c: Neo4j

[162] It is a platform for graph data that also provides many tools for network analysis. More specifically, the Neo4j library includes several community detection algorithms such as Louvain, Label Propagation and others. It also includes many useful metrics such as centralities, similarities, etc.

TABLE 4. References of LCD on static networks. Greedy techniques focusing on starting node(s).

Static networks/ Greedy/ Focus on starting Node(s)			
Reference	Algorithm	Key features	Source Code
Bagrow J. et al [30]	-	use of $l - shell$ to find the starting nodes, parameter dependent	-
Xu B. et al [12]	-	use of k top leader nodes as starting nodes, community size is parameter dependent	
Zhang T. et al [31]	-	use nodes with maximum degree as starting ones, low accuracy	-
Chen Q. Et al [32]	LMD	local degree central node as seed	-
Moradi F. et al [33]	-	similarity score to define starting nodes, parameter-free	-
Xia S. et al [10]	ILCDSP	uses selection probability to find starting nodes, not very stable	-
Fanrong M. et al [34]	LCD-MC	start LCD from maximal clique containing the seed node	-
Wang P. et al [35]	MAGA-LC	tightest nodes as seeds	-
Hamann M. et al [36]	-	start LCD from maximal clique containing the seed node, TCE method, unweighted and weighted networks, overlapping communities	Provided
Ding X. et al [37]	RTLCD	use core node as starting one	-
Tasgin M. et al [38]	-	use of highest score node to start the label propagation method, instability in determining the final communities, weak performance in identifying a reasonable number of communities	Provided
Xu X. Et al [39]	DLCD-CCE	expands community centers, can run in a distributed environment, weighted networks	-
Guo K. et al [8]	InfoNode	local degree central nodes and Jaccard coefficient to detect starting nodes, overlapping communities, parameter dependent, more than one seeds	-
Luo W. et al [40]	LCDNN	nearest nodes with greater centrality as seeds, multiscale local communities	-
Liu J. et al [16]	HqsMLCD	multiple communities, higher quality starting nodes	-
Aghaalizadeh et al [41]	-	use core nodes as starting, weighted networks	-
Hu Y. et al [42]	S-LM	local centrality measurements used for selecting starting nodes	-
Ji P. et al [43]	CAELCD	finds core nodes as seeds and expand them	-

d: SNAP

[163] It is a library for network analysis. It contains interfaces for both Python and C++. It also provides a large number of different network datasets, including temporal ones.

e: GUNROCK

[164] It is currently the most powerful CUDA graph processing library designed specifically for the GPU. Many methods, useful for community detection, are implemented and executed in optimal time, such as Louvain, PageRank Nibble and others. However, it only supports static graph datasets and not dynamic ones.

f: SENTENCETRANSFORMERS

[165] It is a Python framework for state-of-the-art sentence, text and image embeddings. It can be used along with GPU for very fast implementation. Among others, it implements a local community detection algorithm that is tuned for large datasets (50k sentences in less than 5 seconds). The user can also specify the minimal size of a local community.

Among the works included in this survey, only a few provide their source code. More precisely, the 20% of the works make their algorithm available online for anyone to use. Figure 4 shows the source code availability of the works presented in this survey.

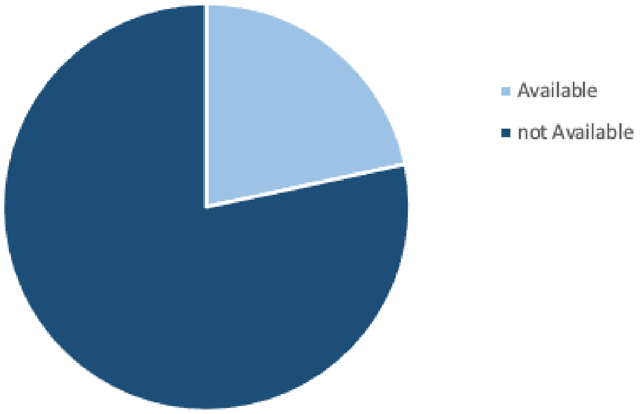

FIGURE 4. Source code availability in works of the present survey.

B. DATASETS

There is a plethora of datasets that are used to experiment with proposed approaches. However, there are certain datasets that are very commonly used by researchers. In particular, in the LCD-related literature, we found that most papers use the same network datasets for their experiments. The main reason for this is that it allows us to make a fair comparison with competing methods. Another reason is that there are not many well-described datasets that simultaneously share certain characteristics, e.g., ground truth communities. In Table 14 we present the most commonly used synthetic and real-world datasets in the works for LCD along with their usage percentages.

In most works, concerning LCD, researchers choose to experiment with both synthetic and real network datasets. More specifically, about 78% of the papers presented here use both synthetic and real networks, while 22% use only real networks. As can be deduced, there is no work that uses only synthetic networks.

As for synthetic datasets, about 59% use the LFR benchmark network [106]. The remaining works use one of the following: Dynamic Benchmark Network Generator [169], Girvan-Newman network generator [9], R-MAT generator [170], Stochastic Block Model [171], preferential attachment synthetic networks [141], dynamic network generator based on Markovian evolution [172], Ulam networks [173] and the RDYN graph benchmark [147].

TABLE 5. References of LCD on static networks. Greedy techniques focusing on quality metric.

Static networks/ Greedy/ Focus on Quality metric			
Reference	Algorithm	Key features	Source Code
Clauset A. [17]	-	local modularity R as quality metric, parameter dependent (community size), sensitive to the seed's position	-
Luo F. et al [44]	-	local modularity L as quality metric	-
Chen J. et al [45]	-	quality local metric L	-
Huang J. et al [46]	-	tightness as quality metric, seed dependent	-
Branting L. et al [47]	-	NUWR metric	-
Wu Y. et al [48]	-	link similarity	-
Ngonmang B. et al [49]	IOLoCo	[45] optimization to start LCD from boundary node	-
Kudelka M. et al [50]	-	weighted, dependency as quality metric	-
Ma L. et al [51]	GMAC	CI metric as quality	-
Cui W. et al [52]	-	goodness metric as quality	-
Fagnan J. et al [53]	-	T-metric (triads), directed	-
Wu Y. et al [54]	-	Query-oriented node weighting scheme to reduce free-rider effect, overlapping, multiple	-
Chang C. et al [55]	-	relative centrality metric	-
Xia Z. et al [56]	-	local modularity G	-
Kanawati R. et al [57]	-	Combination of different local modularity functions, weighted	-
Zhao W. et al [58]	-	Closeness-Isolation quality metric based on edge weighting	-
Liu J. et al [59]	-	modularity modification metric, similarity is calculated based on the number of common neighbours	-
Interdonato R. et al [60]	ML-LCD	multilayer	Provided
Luo W. et al [61]	DMF_M and DMF_R	based on the local modularity metrics as suggested by Luo J. et al [44] and Clauset A. [17]	-
Ni L. et al [62]	LOCD	overlapping	-
Li X. et al [63]	-	trust relation, multilayer, Suitable for social networks	-
Guo K. et al [64]	LCDMD	local modularity and community tightness based on the Jaccard coefficient used	Provided
Baltsou G. et al [65]	HEF	detecting better quality local communities of predefined important nodes called hints, reweighting and rewiring strategies used to materialize the preference	-
Shang R. et al [66]	HSEI	higher-order structure and edge information, motif-based modularity	-

TABLE 6. References of LCD on static networks. Non-greedy techniques using flow propagation.

Static networks/ Non-Greedy/ Flow Propagation			
Reference	Algorithm	Key features	Source Code
Orecchia L. et al [67]	-	first to combine flow and spectral methods for LCD, (un)weighted	-
Panagiotakis P. et al [68]	FlowPro	(un)weighted, overlapping	Provided
Veldt N. Et al [69]	SimpleLocal	searches better conductance set, takes advantage of many maximum flow algorithms	-
Veldt N. Et al [70]	FlowSeed algorithm	improvement of [69] where strict constraints and soft penalties on excluding specified seeds from the result set allowed	Provided

Regarding the real-world datasets they are of different domains, although most of them are social networks. The DBLP [113] website contains an extensive list of research papers from the field of computer science. The dataset is a collaboration network where each node denotes a researcher and each edge represents the existence of a collaborative work. Amazon [44] represents a co-purchase network in which nodes refer to products sold on the Amazon website, and the edge between two nodes indicates that they are frequently purchased together. Football [9] describes American football games between division IA colleges during the Fall 2000 regular season. The nodes represent football teams, and an edge between two nodes indicates that a game occurred between the two teams. The remaining datasets are social networks. More specifically, Karate [166] is a network in which each node represents a member of Zachary's karate club, and an edge indicates that two members are friends. Youtube [167] is another social network where each node represents a user and each edge represents a friendship between two users. LiveJournal [113] is another social network where edges indicate that two users have formed a friendship. Polbooks [103] is a network of books about US politics. Each node represents a book, and an edge between two books indicates that they are often bought together. Dolphins [168] is a social network with frequent interactions between dolphins living in New Zealand. The nodes represent dolphins and the edges indicate that the corresponding dolphins have frequent contact. Besides, Orkut [113] is also a social network where users make friends with each other. Finally, Friendster [113] is an on-line gaming network while it started as a social networking site where users could be form friendships with other users. Figure 5 shows the usage percentage of real networks used in the literature of this survey.

VI. INSIGHTS AND FUTURE DIRECTIONS

To suggest the best performing methods to the reader, it is necessary to take into consideration several of their aspects. Thus, we have to consider if a method is tested along with state-of-the-art approaches and proved to perform better in terms of specific metrics. Besides, it would be a plus if a method is tested on several and of different real datasets. Furthermore, it is also important for a method to provide the source code, so as the researchers could compare their methods with the existing ones.

TABLE 7. References of LCD on static networks. Non-greedy, random-walk based techniques using PageRank.

Static networks/ Non-Greedy/ Random Walk/ PageRank			
Reference	Algorithm	Key features	Source Code
Andersen R. et al [71]	PageRank Nibble	computes a single personalized PageRank vector rather than a sequence of random walk distributions	-
Gharan S. et al [72]	-	Approximation guarantee close to the guarantee of the Cheeger inequalities, weighted	-
Zhu Z. et al [73]	-	focus on identification of well-connected communities, (un)weighted	-
Kloumann I. et al [74]	-	Evaluation of different variants of the PageRank method	-
Yin H. Et al [14]	(MAPPR) algorithm,	Motif-based Approximate Personalized PageRank, (un)directed	-
Hollocou A. et al [75]	MULTICOM	detects multiple local communities that might overlap	Provided

TABLE 8. References of LCD on static networks. Non-greedy, random-walk based techniques using heat kernel.

Static networks/ Non-Greedy/ Random Walk/ Heat kernel			
Reference	Algorithm	Key features	Source Code
Chung F. et al [76]	-	Randomized Monte Carlo method, (un)weighted	-
Kloster K. et al [77]	hk-relax	the first deterministic method for computing a heat kernel diffusion, parameter dependent	Provided
Shun J. et al [78]	-	Parallel version of heat kernel algorithm	-

TABLE 9. References of LCD on static networks. Non-greedy, random-walk based techniques using local spectral.

Static networks/ Non-Greedy/ Random Walk/ Local spectral			
Reference	Algorithm	Key features	Source Code
Mahoney M. et al [79]	-	locally-biased analogue of the second eigenvector, parameter dependent, suitable for small networks	-
Li Y. et al [13]	LEMON	few seeds needed, overlapping	Provided
He K. et al [80]	LOSP	works with single seed, overlapping	Provided
Laarhoven T. et al [81]	-	conductance minimization approximation	-
Bian Y. et al [82]	MWC	multi-walker instead of singe one, weighted	-
Jeub L. et al [83]	-	Analysis of the local behavior of different random walks on multiplex networks	Provided
Bian Y. et al [15]	MRW	memory-based random walk method, that can simultaneously identify multiple target local communities, weighted, overlapping	Provided
Freitas S. et al [84]	-	uses both network structure and attribute data, an application to predict the evolution of local communities, (un)weighted	Provided
He K. et al [85]	-	Extension of He K. et al [80]	Provided
Zhang Y. et al [86]	LCD-Motif	based on network motifs, parameter dependent	-
Fan X. et al [87]	-	gravity between each neighbor node to core nodes, overlapping	-
Shi P. et al [88]	LBSA	first to use the Lanczos iteration, few seeds needed	Provided
Luo D. et al [89]	LCDNN	multiscale, (non-)overlapping	-

The best performing methods that focus on node selection in terms of accurately identify local communities are those that propose starting the LCD process from maximal cliques rather than a single seed node proposed by Fanrong et al. [34] and Hamann et al [36]. However, the method proposed by Hamann et al. [36], might be the best overall approach, since it is extensively experimented on many different datasets and compared to several other state-of-the-art approaches. Furthermore, the suggested method is suitable for (un)weighted and overlapping networks. Moreover, the source code is publicly available.

Considering greedy methods that focus on a quality metric, probably the best performing method is the one proposed by Guo et al. [64]. With a view to limit sensitivity to seed node selection and the instability of communities, the authors suggest an algorithm that uses a simple quality metric based on Jaccard coefficient. The proposed method is publicly available and the authors conducted many experiments with both real and synthetic networks in order to compare with several existing methods.

As for non-greedy methods, the random-walk based method seems to perform better than flow based methods, especially when the seed set is small compared with the community it belongs. The random-walk based method proposed by Shi et al. [88] performs very well compared to other state-of-the-art methods. The authors also provide the source code of the proposed approach and conducted experiments with several real and synthetic datasets.

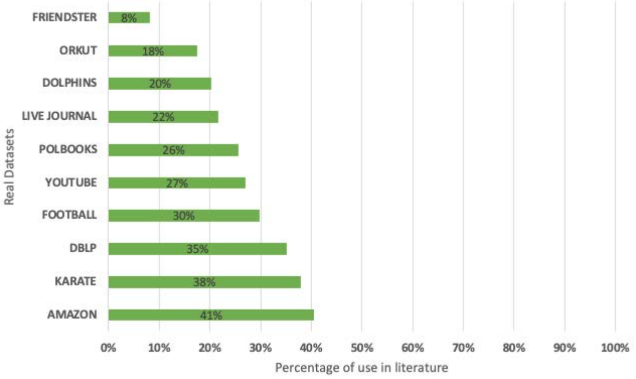

FIGURE 5. Real-world datasets used in works of the present survey.

Concerning the temporal networks, the methods proposed by DiTursi et al. [93] and Fu et al. [100] respectively, seem to perform more effectively and efficiently than

TABLE 10. References of LCD on dynamic networks. Techniques using snapshot model.

Dynamic networks/ Snapshot model			
Reference	Algorithm	Key features	Source Code
Takaffoli M. et al [90]	-	Adaptation of the static L-metric to dynamic	-
Zakrzewska A. et al [11]	-	focuses on maintaining the community centered around the seed	-
Gao W. et al [91]	EvoLeaders	leader nodes	-
Nathan E. et al [92]	-	combination of Katz and PageRank centrality, requires a priori knowledge about the size of the community	-
DiTursi D. et al [93]	PHASR	temporal conductance measure, weighted	-
Javadi S. et al [94]	-	leader nodes, degree centrality	-
Guo K. et al [95]	-	local fitness, node contribution, DyCDNC method	-
Papadopoulos A. et al [96]	-	Extension of DiTursi D. et al [93] PHASR algorithm to conform to the Apache Spark engine distributed processing standard	-
Liu J. et al [97]	HqsDMLCD	extension of Liu J. et al [16] to dynamic networks, first work for dynamic local multiple communities	-

TABLE 11. References of LCD on dynamic networks. Techniques using temporal smoothness.

Dynamic networks/ Temporal smoothness			
Reference	Algorithm	Key features	Source Code
Hu Y. et al [98]	LDM-CET	Personalized PageRank approach, overlapping	-
Rigi M. et al [99]	-	method inspired by geometric active contours	-
Fu D. et al [100]	L-MEGA	motif-based method, multi-linear PageRank vector by edge filtering	-
Hu Y. et al [42]	D-LM method	extension of their static version, not suitable when many nodes are affected by changes	-

TABLE 12. References of LCD on graph streams.

Graph Streams			
Reference	Algorithm	Key features	Source Code
Liakos P. et al [101]	CoEuS	uses community participation of nodes, conductance evaluation, first for graph streams	Provided
Christopoulos K. et al [146]	-	Three rewarding methods and three different quality metrics to strengthen the vicinity of the seed set called anchors	-

other state-of-the-art approaches. Both methods are extensively compared with others on synthetic and real networks of various types, and detect high quality communities (in terms of conductance and triangle density) in an efficient manner.

Another aspect that we discuss here concerns the scalability of LCD methods. In general, one of the reasons that local approaches are preferred over global ones, is the fact that the size of the network does not have an important role in the community detection process, since the researcher is interested in only a specific region of the network i.e. around the seed nodes. Thus, the time needed to detect a community is proportional to the size of the community [14], [33], [34]. However, mostly in diffusion approaches, their run-time as well as the quality of the detected local communities may depend on some other characteristics of the networks like density and community diameter [25], [39], [68], [100]. For instance, a simple method such as a short random walk may be efficient on a dataset with small diameter communities, but may fail to reach a large part of a community in a sparser graph with a larger diameter. Nonetheless, a more complex approach such as a local spectral method might maintain consistent community quality across these graphs, but as the graph size increases, orders of magnitude become slower.

Considering the future directions of local community detection, these may be determined, to some extent, by current trends in the field. More specifically, as the available amount of data becomes larger and more mutable, there is the need for effectively and efficiently analyze rapidly changing large networks. Thus, researchers of the domain should probably focus on techniques designed for dynamic networks and for graph streams. Another trend that was observed, concerns multilayer networks. Since social networks play a vital role in users data analysis, it is important to link distributed user profiles of the same user from multiple platforms. In such cases, multilayer networks are used as a modeling tool. We believe that the local community detection approaches concerning multilayer networks will attract more and more researchers' interest.

Finally, deep learning is gaining popularity in recent years as it can be used to solve many problems in various fields such as speech recognition, natural language processing, image classification, and so on. One important reason for this is the performance superiority of Deep Learning techniques in processing large amounts of data compared to traditional machine learning methods. In problems related to community detection, convolutional graphs are used as a semi-supervised method. More specifically, given several communities of a network as training data the aim is to discover more communities in the same network [174]. Today, there are few works on LCD in graph convolutional networks [175], [176], but in the future such methods may attract the interest of researchers, although they should focus at unsupervised methods.

TABLE 13. Tools used for network community detection in alphabetical order.

Tool	Type	Platform	License
CFinder	GUI	Windows, Linux, MacOS	Free
Cytoscape	GUI	Windows, Linux, MacOS	Open-source and Free
Gephi	GUI	Windows, Linux, MacOS	Open-source and Free
Graphviz	GUI	Windows, Linux, MacOS	Open-source and Free
Gunrock	library	Ubuntu 16.04,18.04 with CUDA 9+	Free
igraph	library	C/C++, Python, R	Open-source and Free
Neo4j	library	Windows, Linux, MacOS	Free
NetMiner	GUI	Windows	Trial and Commercial
networkX	library	Python	Open-source and Free
NodeXL	GUI	Microsoft Excel	Trial and Commercial
Pajek	GUI	Windows, Linux, MacOS	Free for Non-commercial use
Sentence Transformers	library	Python	Open-source and Free
SNAP	library	C++, Python	Open-source and Free
SocNetV	GUI	Windows, Linux, MacOS	Open-source and Free
Visone	GUI	Windows, Linux, MacOS	Free and Commercial

TABLE 14. Datasets mostly used in LCD papers of the present survey.

Type of Dataset	Name	Percentage of works that use it
Synthetic	LFR [106]	59%
Real	Amazon [44]	41%
Real	Karate [166]	38%
Real	DBLP [113]	35%
Real	Football [9]	30%
Real	Youtube [167]	27%
Real	Polbooks [103]	26%
Real	LiveJournal [113]	22%
Real	Dolphins [168]	20%
Real	Orkut [113]	18%
Real	Friendster [113]	8%

VII. CONCLUSION

Local community detection has been a very active area of research in recent years. There are a variety of proposals to achieve this goal, but they are not all applicable to any particular problem. Motivated by the abundance of practical applications of local community detection, we suggest a classification of the local community detection techniques proposed in the literature, focusing on the techniques of the last decade. This classification is not absolute, as there may be different separation criteria. Our goal is to help researchers decide which approach is the best for their particular problem, based on the type of network and local community detection method they wish to use. Besides, we discuss tools that are commonly used in the detection of local communities. These tools are either GUIs or scripting/programming languages. Furthermore, we present a list of the most commonly used synthetic and real datasets in the literature so that researchers can test their approaches and compare them to others.

As it can be concluded from the present survey, there is no LCD approach that can be said to be the best or the one that fits all. Rather, the researcher must choose the most appropriate method for the problem at hand, depending on the nature and characteristics of the network and the constraints that she/he might place on the setting.

In the future, it would be very interesting to extend the present study with an experimental comparison of local community detection algorithms. This would give, to any interested researcher, a more thorough and complete overview of the available options to LCD problems.

ACKNOWLEDGMENT

“Georgia Baltsou is co-financed by Greece and the European Union (European Social Fund-ESF) through the Operational Programme “Human Resources Development, Education and Lifelong Learning” in the context of the project “Strengthening Human Resources Research Potential via Doctorate Research - 2nd Cycle” (MIS-5000432), implemented by the State Scholarships Foundation (IKY).”

“This research was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the “2nd Call for H.F.R.I. Research Projects to support Faculty Members & Researchers” (Project Number: 3480).”

REFERENCES

[1] Y. Ding, “Scientific collaboration and endorsement: Network analysis of coauthorship and citation networks,” *J. Informetrics*, vol. 5, pp. 187–203, Jan. 2011.

[2] I. Himelboim, “Social network analysis (social media),” in *The International Encyclopedia of Communication Research Methods*. Wiley, 2017, pp. 1–15.

[3] S. Kharel, P. Shivananda, K. Ramesh, K. N. Jothi, and K. G. Raj, “Use of transportation network analysis for bus stop relocation, depiction of service area and bus route details,” *J. Geomatics*, vol. 13, no. 2, pp. 224–229, 2019.

[4] G. Mantini, A. M. Vallés, T. Y. S. Le Large, M. Capula, N. Funel, T. V. Pham, S. R. Piersma, G. Kazemier, M. F. Bijlsma, E. Giovannetti, and C. R. Jimenez, “Co-expression analysis of pancreatic cancer proteome reveals biology and prognostic biomarkers,” *Cellular Oncol.*, vol. 43, no. 6, pp. 1147–1159, Dec. 2020.

[5] R. Sharan, I. Ulitsky, and R. Shamir, “Network-based prediction of protein function,” *Mol. Syst. Biol.*, vol. 3, no. 1, p. 88, Jan. 2007.

[6] S. Chen, Z.-Z. Wang, L. Tang, Y.-N. Tang, Y.-Y. Gao, H.-J. Li, J. Xiang, and Y. Zhang, “Global vs local modularity for network community detection,” *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0205284.

[7] S. Fortunato and D. Hric, “Community detection in networks: A user guide,” *Phys. Rep.*, vol. 659, pp. 1–44, Nov. 2016.

[8] K. Guo, L. He, Y. Chen, W. Guo, and J. Zheng, "A local community detection algorithm based on internal force between nodes," *Int. J. Speech Technol.*, vol. 50, no. 2, pp. 328–340, Feb. 2020.

[9] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proc. Nat. Acad. Sci. USA*, vol. 99, no. 12, pp. 7821–7826, Jun. 2002.

[10] S. Xia, R. Zhou, Y. Zhou, and M. Zhu, "An improved local community detection algorithm using selection probability," *Math. Problems Eng.*, vol. 2014, pp. 1–10, Jan. 2014.

[11] A. Zakrzewska and D. A. Bader, "A dynamic algorithm for local community detection in graphs," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2015, pp. 559–564.

[12] B. Xu, Z. Liang, Y. Jia, B. Zhou, and Y. Han, "Local community detection using seeds expansion," in *Proc. IEEE Conf. Anthology*, Jan. 2013, pp. 557–562.

[13] Y. Li, K. He, D. Bindel, and J. E. Hopcroft, "Uncovering the small community structure in large networks: A local spectral approach," in *Proc. 24th Int. Conf. World Wide Web*, May 2015, pp. 658–668.

[14] H. Yin, A. R. Benson, J. Leskovec, and D. F. Gleich, "Local higher-order graph clustering," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 555–564.

[15] Y. Bian, Y. Yan, W. Cheng, W. Wang, D. Luo, and X. Zhang, "On multi-query local community detection," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 9–18.

[16] J. Liu, Y. Shao, and S. Su, "Multiple local community detection via high-quality seed identification," in *Proc. Asia–Pacific Web (APWeb) Web-Age Inf. Manage. (WAIM) Joint Int. Conf. Web Big Data*. Germany: Springer, 2020, pp. 37–52.

[17] A. Clauset, "Finding local community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 72, no. 2, Aug. 2005, Art. no. 026132.

[18] S. Souravlas, A. Sifaleras, M. Tsintogianni, and S. Katsavounis, "A classification of community detection methods in social networks: A survey," *Int. J. Gen. Syst.*, vol. 50, no. 1, pp. 63–91, Jan. 2021.

[19] S. Dilmaghani, M. R. Brust, G. Danoy, and P. Bouvry, "Community detection in complex networks: A survey on local approaches," in *Proc. Asian Conf. Intell. Inf. Database Syst.* Germany: Springer, 2021, pp. 757–767.

[20] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos, "Community detection in social media," *Data Mining Knowl. Discovery*, vol. 24, no. 3, pp. 515–554, May 2012.

[21] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Comput. Surv.*, vol. 45, no. 4, pp. 1–35, Aug. 2013.

[22] S. Harenberg, G. Bello, L. Gjeltema, S. Ranshous, J. Harlalka, R. Seay, K. Padmanabhan, and N. Samatova, "Community detection in large-scale networks: A survey and empirical evaluation," *Wiley Interdiscipl. Rev., Comput. Statist.*, vol. 6, no. 6, pp. 426–439, 2014.

[23] R. Maivizhi, S. Sendhilkumar, and G. Mahalakshmi, "A survey of tools for community detection and mining in social networks," in *Proc. Int. Conf. Informat. Analytics*, 2016, pp. 1–8.

[24] M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig, "Community detection in networks: A multidisciplinary review," *J. Netw. Comput. Appl.*, vol. 108, pp. 87–111, Apr. 2018.

[25] L. Ma, K. Chiew, H. Huang, and Q. He, "Evaluation of local community metrics: From an experimental perspective," *J. Intell. Inf. Syst.*, vol. 51, no. 1, pp. 1–22, Aug. 2018.

[26] N. Dakiche, F. B.-S. Tayeb, Y. Slimani, and K. Benatchba, "Tracking community evolution in social networks: A survey," *Inf. Process. Manage.*, vol. 56, no. 3, pp. 1084–1102, May 2019.

[27] X. Huang, D. Chen, T. Ren, and D. Wang, "A survey of community detection methods in multilayer networks," *Data Mining Knowl. Discovery*, vol. 35, no. 1, pp. 1–45, Jan. 2021.

[28] P. Meena, M. Pawar, and A. Pandey, "A survey on community detection algorithm and its applications," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 6, pp. 4807–4815, 2021.

[29] X. Su, S. Xue, F. Liu, J. Wu, J. Yang, C. Zhou, W. Hu, C. Paris, S. Nepal, D. Jin, Q. Z. Sheng, and P. S. Yu, "A comprehensive survey on community detection with deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Mar. 9, 2022, doi: 10.1109/TNNLS.2021.3137396.

[30] J. P. Bagrow and E. M. Bollt, "Local method for detecting communities," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 72, no. 4, Oct. 2005, Art. no. 046108.

[31] T. Zhang and B. Wu, "A method for local community detection by finding core nodes," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2012, pp. 1171–1176.

[32] Q. Chen, T.-T. Wu, and M. Fang, "Detecting local community structures in complex networks based on local degree central nodes," *Phys. A, Stat. Mech. Appl.*, vol. 392, no. 3, pp. 529–537, Feb. 2013.

[33] F. Moradi, T. Olovsson, and P. Tsigas, "A local seed selection algorithm for overlapping community detection," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2014, pp. 1–8.

[34] M. Fanrong, Z. Mu, Z. Yong, and Z. Ranran, "Local community detection in complex networks based on maximum cliques extension," *Math. Problems Eng.*, vol. 2014, pp. 1–12, Jan. 2014.

[35] P. Wang and J. Liu, "A multi-agent genetic algorithm for local community detection by extending the tightest nodes," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jul. 2016, pp. 3215–3221.

[36] M. Hamann, E. Röhrs, and D. Wagner, "Local community detection based on small cliques," *Algorithms*, vol. 10, no. 3, p. 90, Aug. 2017.

[37] X. Ding, J. Zhang, and J. Yang, "A robust two-stage algorithm for local community detection," *Knowl.-Based Syst.*, vol. 152, pp. 188–199, Jul. 2018.

[38] M. Tasgin and H. O. Bingol, "Community detection using boundary nodes in complex networks," *Phys. A, Stat. Mech. Appl.*, vol. 513, pp. 315–324, Jan. 2019.

[39] X. Xu, N. Hu, M. Trovati, J. Ray, F. Palmieri, and H. M. Pandey, "DLCD-CCE: A local community detection algorithm for complex IoT networks," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4607–4615, May 2019.

[40] W. Luo, N. Lu, L. Ni, W. Zhu, and W. Ding, "Local community detection by the nearest nodes with greater centrality," *Inf. Sci.*, vol. 517, pp. 377–392, May 2020.

[41] S. Aghaalizadeh, S. T. Afshord, A. Bouyer, and B. Anari, "A three-stage algorithm for local community detection based on the high node importance ranking in social networks," *Phys. A, Stat. Mech. Appl.*, vol. 563, Feb. 2021, Art. no. 125420.

[42] Y. Hu, Y. Zhang, X. Wang, J. Wu, and B. Duo, "A local seeding algorithm for community detection in dynamic networks," in *Proc. Int. Conf. Adv. Data Mining Appl.* Germany: Springer, 2022, pp. 97–112.

[43] P. Ji, K. Guo, and Z. Yu, "Local community detection algorithm based on core area expansion," in *Proc. CCF Conf. Comput. Supported Cooperat. Work Social Comput.* Germany: Springer, 2022, pp. 238–251.

[44] F. Luo, J. Z. Wang, and E. Promislow, "Exploring local community structures in large networks," *Web Intell. Agent Syst., Int. J.*, vol. 6, no. 4, pp. 387–400, 2008.

[45] J. Chen, O. Zaïane, and R. Goebel, "Local community identification in social networks," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, Jul. 2009, pp. 237–242.

[46] J. Huang, H. Sun, Y. Liu, Q. Song, and T. Weninger, "Towards online multiresolution community detection in large-scale networks," *PLoS ONE*, vol. 6, no. 8, Aug. 2011, Art. no. e23829.

[47] L. K. Branting, "Context-sensitive detection of local community structure," *Social Netw. Anal. Mining*, vol. 2, no. 3, pp. 279–289, Sep. 2012.

[48] Y.-J. Wu, H. Huang, Z.-F. Hao, and F. Chen, "Local community detection using link similarity," *J. Comput. Sci. Technol.*, vol. 27, no. 6, pp. 1261–1268, 2012.

[49] B. Ngonmang, M. Tchuente, and E. Viennet, "Local community identification in social networks," *Parallel Process. Lett.*, vol. 22, no. 1, Mar. 2012, Art. no. 1240004.

[50] M. Kudelka, P. Dráždilová, E. Ochodková, K. Slaninová, and Z. Horák, "Local community detection and visualization: Experiment based on student data," in *Proc. 3rd Int. Conf. Intell. Hum. Comput. Interact. (IHCI)* Prague, Czech Republic: Springer, Aug. 2011, pp. 291–303.

[51] L. Ma, H. Huang, Q. He, K. Chiew, J. Wu, and Y. Che, "GMAC: A seed-insensitive approach to local community detection," in *Proc. Int. Conf. Data Warehousing Knowl. Discovery*. Germany: Springer, 2013, pp. 297–308.

[52] W. Cui, Y. Xiao, H. Wang, and W. Wang, "Local search of communities in large graphs," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2014, pp. 991–1002.

[53] J. Fagnan, O. Zaiane, and D. Barbosa, "Using triads to identify local community structure in social networks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2014, pp. 108–112.

[54] Y. Wu, R. Jin, J. Li, and X. Zhang, "Robust local community detection: On free rider effect and its elimination," *Proc. VLDB Endowment*, vol. 8, no. 7, pp. 798–809, Feb. 2015.

[55] C.-S. Chang, C.-J. Chang, W.-T. Hsieh, D.-S. Lee, L.-H. Liou, and W. Liao, "Relative centrality and local community detection," *Netw. Sci.*, vol. 3, no. 4, pp. 445–479, 2015.

[56] Z. Xia, X. Gao, and X. Zhang, "Local community detection based on modularity metric G," *Modern Phys. Lett. B*, vol. 29, no. 33, Dec. 2015, Art. no. 1550215.

[57] R. Kanawati, "Empirical evaluation of applying ensemble methods to ego-centred community identification in complex networks," *Neurocomputing*, vol. 150, pp. 417–427, Feb. 2015.

[58] W. Zhao, F. Zhang, and J. Liu, "Local community detection via edge weighting," in *Proc. Asia Inf. Retr. Symp.* Germany: Springer, 2016, pp. 68–80.

[59] J. Liu, D. Wang, W. Zhao, S. Feng, and Y. Zhang, "A novel approach for discovering local community structure in networks," in *Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*. Germany: Springer, 2016, pp. 363–374.

[60] R. Interdonato, A. Tagarelli, D. Ienco, A. Sallaberry, and P. Poncelet, "Local community detection in multilayer networks," *Data Mining Knowl. Discovery*, vol. 31, no. 5, pp. 1444–1479, Sep. 2017.

[61] W. Luo, D. Zhang, H. Jiang, L. Ni, and Y. Hu, "Local community detection with the dynamic membership function," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 5, pp. 3136–3150, Oct. 2018.

[62] L. Ni, W. Luo, W. Zhu, and B. Hua, "Local overlapping community detection," *ACM Trans. Knowl. Discovery Data*, vol. 14, no. 1, pp. 1–25, 2019.

[63] X. Li, Q. Tian, M. Tang, X. Chen, and X. Yang, "Local community detection for multi-layer mobile network based on the trust relation," *Wireless Netw.*, vol. 26, no. 8, pp. 5503–5515, Nov. 2020.

[64] K. Guo, X. Huang, L. Wu, and Y. Chen, "Local community detection algorithm based on local modularity density," *Appl. Intell.*, vol. 52, no. 2, pp. 1–16, 2021.

[65] G. Baltsou, K. Tsichlas, and A. Vakali, "Local community detection with hints," *Appl. Intell.*, vol. 52, pp. 1–22, Jan. 2022.

[66] R. Shang, W. Zhang, J. Zhang, J. Feng, and L. Jiao, "Local community detection based on higher-order structure and edge information," *Phys. A, Stat. Mech. Appl.*, vol. 587, Feb. 2022, Art. no. 126513.

[67] L. Orecchia and Z. A. Zhu, "Flow-based algorithms for local graph clustering," in *Proc. 25th Annu. ACM-SIAM Symp. Discrete Algorithms*, Jan. 2014, pp. 1267–1286.

[68] C. Panagiotakis, H. Papadakis, and P. Fragopoulou, "Local community detection via flow propagation," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2015, pp. 81–88.

[69] N. Veldt, D. Gleich, and M. Mahoney, "A simple and strongly-local flow-based method for cut improvement," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1938–1947.

[70] N. Veldt, C. Klymko, and D. F. Gleich, "Flow-based local graph clustering with better seed set inclusion," in *Proc. SIAM Int. Conf. Data Mining*, 2019, pp. 378–386.

[71] R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using PageRank vectors," in *Proc. 47th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, 2006, pp. 475–486.

[72] S. O. Gharan and L. Trevisan, "Approximating the expansion profile and almost optimal local graph clustering," in *Proc. IEEE 53rd Annu. Symp. Found. Comput. Sci.*, Oct. 2012, pp. 187–196.

[73] Z. A. Zhu, S. Lattanzi, and V. Mirrokni, "A local algorithm for finding well-connected clusters," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 396–404.

[74] I. M. Kloumann and J. M. Kleinberg, "Community membership identification from small seed sets," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 1366–1375.

[75] A. Hollocou, T. Bonald, and M. Lelarge, "Multiple local community detection," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 45, no. 3, pp. 76–83, 2018.

[76] F. Chung and O. Simpson, "Solving linear systems with boundary conditions using heat kernel pagerank," in *Int. Workshop Algorithms Models for Web-Graph*, pp. 203–219. Springer, 2013.

[77] K. Kloster and D. F. Gleich, "Heat kernel based community detection," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2014, pp. 1386–1395.

[78] J. Shun, F. Roosta-Khorasani, K. Fountoulakis, and M. W. Mahoney, "Parallel local graph clustering," *Proc. VLDB Endowment*, vol. 9, no. 12, pp. 1041–1052, Aug. 2016.

[79] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi, "A local spectral method for graphs: With applications to improving graph partitions and exploring data graphs locally," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 2339–2365, 2012.

[80] K. He, Y. Sun, D. Bindel, J. Hopcroft, and Y. Li, "Detecting overlapping communities from local spectral subspaces," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2015, pp. 769–774.

[81] T. V. Laarhoven and E. Marchiori, "Local network community detection with continuous optimization of conductance and weighted kernel K-means," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 5148–5175, 2016.

[82] Y. Bian, J. Ni, W. Cheng, and X. Zhang, "Many heads are better than one: Local community detection by the multi-walker chain," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 21–30.

[83] L. G. S. Jeub, M. W. Mahoney, P. J. Mucha, and M. A. Porter, "A local perspective on community structure in multilayer networks," *Netw. Sci.*, vol. 5, no. 2, pp. 144–163, Jun. 2017.

[84] S. Freitas, N. Cao, Y. Xia, D. H. P. Chau, and H. Tong, "Local partition in rich graphs," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 1001–1008.

[85] K. He, P. Shi, D. Bindel, and J. E. Hopcroft, "Krylov subspace approximation for local community detection in large networks," *ACM Trans. Knowl. Discovery Data*, vol. 13, no. 5, pp. 1–30, Oct. 2019.

[86] Y. Zhang, B. Wu, Y. Liu, and J. Lv, "Local community detection based on network motifs," *Tsinghua Sci. Technol.*, vol. 24, no. 6, pp. 716–727, Dec. 2019.

[87] X. Fan, Z. Chen, F. Cai, J. Wu, S. Liu, Z. Liao, and Z. Liao, "Local core members aided community structure detection," *Mobile Netw. Appl.*, vol. 24, no. 4, pp. 1373–1381, Aug. 2019.

[88] P. Shi, K. He, D. Bindel, and J. E. Hopcroft, "Locally-biased spectral approximation for community detection," *Knowl.-Based Syst.*, vol. 164, pp. 459–472, Jan. 2019.

[89] D. Luo, Y. Bian, Y. Yan, X. Liu, J. Huan, and X. Zhang, "Local community detection in multiple networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 266–274.

[90] M. Takaffoli, R. Rabbany, and O. R. Zaïane, "Incremental local community identification in dynamic social networks," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2013, pp. 90–94.

[91] W. Gao, W. Luo, and C. Bu, "Evolutionary community discovery in dynamic networks based on leader nodes," in *Proc. Int. Conf. Big Data Smart Comput. (BigComp)*, Jan. 2016, pp. 53–60.

[92] E. Nathan, A. Zakrzewska, J. Riedy, and D. Bader, "Local community detection in dynamic graphs using personalized centrality," *Algorithms*, vol. 10, no. 3, p. 102, Aug. 2017.

[93] D. J. DiTursi, G. Ghosh, and P. Bogdanov, "Local community detection in dynamic networks," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 847–852.

[94] S. H. S. Javadi, P. Gharani, and S. Khadivi, "Detecting community structure in dynamic social networks using the concept of leadership," in *Sustainable Interdependent Networks*. Germany: Springer, 2018, pp. 97–118.

[95] K. Guo, L. He, J. Huang, Y. Chen, and B. Lin, "A local dynamic community detection algorithm based on node contribution," in *Proc. CCF Conf. Comput. Supported Cooperat. Work Social Comput.* Germany: Springer, 2019, pp. 363–376.

[96] A. N. Papadopoulos and G. Tzortzidis, "Distributed time-based local community detection," in *Proc. 24th Pan-Hellenic Conf. Informat.*, Nov. 2020, pp. 390–393.

[97] J. Liu, Y. Shao, and S. Su, "Multiple local community detection via high-quality seed identification over both static and dynamic networks," *Data Sci. Eng.*, vol. 6, pp. 249–264, May 2021.

[98] Y. Hu, B. Yang, and C. Lv, "A local dynamic method for tracking communities and their evolution in dynamic networks," *Know. Based Syst.*, vol. 110, pp. 176–190, Oct. 2016.

[99] M. A. Rigi, I. Moser, M. M. Farhangi, and C. Lui, "Finding and tracking local communities by approximating derivatives in networks," *World Wide Web*, vol. 23, no. 3, pp. 1519–1551, May 2020.

[100] D. Fu, D. Zhou, and J. He, "Local motif clustering on time-evolving graphs," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 390–400.

[101] P. Liakos, K. Papakonstantinopoulou, A. Ntoulas, and A. Delis, "Rapid detection of local communities in graph streams," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 5, pp. 2375–2386, May 2020.

[102] G. Baltsou and K. Tsichlas, "Dynamic community detection with anchors," in *Proc. 10th Int. Conf. Complex Netw. Appl.*, 2021, pp. 1–4.

[103] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, Feb. 2004, Art. no. 026113.

[104] *Sina Micro-Blog*. Accessed: Apr. 21, 2022. [Online]. Available: http://weibo.com

[105] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 U.S. election: Divided they blog," in *Proc. 3rd Int. Workshop Link Discovery (LinkKDD)*, 2005, pp. 36–43.

[106] A. Lancichinetti, S. Fortunato, and F. Radicchi, "Benchmark graphs for testing community detection algorithms," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 78, no. 4, Oct. 2008, Art. no. 046110.

[107] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *J. Stat. Mech., Theory Exp.*, vol. 2005, no. 9, Sep. 2005, Art. no. P09008.

[108] M. Rosvall, D. Axelsson, and C. T. Bergstrom, "The map equation," *Eur. Phys. J. Special Topics*, vol. 178, no. 1, pp. 13–23, Nov. 2009.

[109] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New J. Phys.*, vol. 11, no. 3, Mar. 2009, Art. no. 033015.

[110] C. L. Staudt, Y. Marrakchi, and H. Meyerhenke, "Detecting communities around seed nodes in complex networks," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Oct. 2014, pp. 62–69.

[111] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of Facebook networks," *Phys. A, Stat. Mech. Appl.*, vol. 391, no. 16, pp. 4165–4180, Aug. 2012.

[112] Y. Yao, W. Wu, M. Lei, and X. Zhang, "Community detection based on variable vertex influence," in *Proc. IEEE 1st Int. Conf. Data Sci. Cyberspace (DSC)*, Jun. 2016, pp. 418–423.

[113] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," *Knowl. Inf. Syst.*, vol. 42, no. 1, pp. 181–213, 2015.

[114] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, p. 2, Mar. 2007.

[115] H. Shen, X. Cheng, K. Cai, and M.-B. Hu, "Detect overlapping and hierarchical community structure in networks," *Phys. A, Stat. Mech. Appl.*, vol. 388, no. 8, pp. 1706–1712, Nov. 2009.

[116] D. Kamuhanda and K. He, "A nonnegative matrix factorization approach for multiple local community detection," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2018, pp. 642–649.

[117] S. Gregory, "Fuzzy overlapping communities in networks," *J. Stat. Mech., Theory Exp.*, vol. 2011, no. 2, Feb. 2011, Art. no. P02017.

[118] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, Jun. 1998.

[119] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore, "Automating the construction of internet portals with machine learning," *Inf. Retr.*, vol. 3, no. 2, pp. 127–163, Jul. 2000.

[120] H. Papadakis, C. Panagiotakis, and P. Fragopoulou, "Local community finding using synthetic coordinates," in *Future Information Technology*. Germany: Springer, 2011, pp. 9–15.

[121] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The dynamics of viral marketing," *ACM Trans. Web*, vol. 1, no. 1, p. 5, 2007.

[122] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, 1985.

[123] B. Boden, S. Günnemann, H. Hoffmann, and T. Seidl, "Mining coherent subgraphs in multi-layer graphs with edge labels," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2012, pp. 1258–1266.

[124] H. Gao, K. Zhang, J. Yang, F. Wu, and H. Liu, "Applying improved particle swarm optimization for dynamic service composition focusing on quality of service evaluations under hybrid networks," *Int. J. Distrib. Sensor Netw.*, vol. 14, no. 2, Feb. 2018, Art. no. 1550147718761583.

[125] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters," *Internet Math.*, vol. 6, no. 1, pp. 29–123, Jan. 2009.

[126] Y. Zhou, G. Sun, Y. Xing, R. Zhou, and Z. Wang, "Local community detection algorithm based on minimal cluster," *Appl. Comput. Intell. Soft Comput.*, vol. 2016, pp. 1–11, Jan. 2016.

[127] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar, "On the hardness of approximating multicut and sparsest-cut," *Comput. Complex.*, vol. 15, no. 2, pp. 94–114, 2006.

[128] E. Dinic, "Algorithm for solution of a problem of maximum flow in a network with power estimation," *Soviet Math. Doll.*, vol. 11, no. 5, pp. 1277–1280, 1970.

[129] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

[130] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies (OASIS): Cross-sectional MRI data in young, middle aged, nondemented, and demented older adults," *J. Cogn. Neurosci.*, vol. 19, no. 9, pp. 1498–1507, Sep. 2007.

[131] Y. Yi, L. Jin, H. Yu, H. Luo, and F. Cheng, "Density sensitive random walk for local community detection," *IEEE Access*, vol. 9, pp. 27773–27782, 2021.

[132] D. A. Spielman and S.-H. Teng, "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems," in *Proc. 26th Annu. ACM Symp. Theory Comput. (STOC)*, 2004, pp. 81–90.

[133] F. Chung, "A local graph partitioning algorithm using heat kernel pagerank," *Internet Math.*, vol. 6, no. 3, pp. 315–330, Jan. 2009.

[134] *Web Graph*. Accessed: Apr. 21, 2022. [Online]. Available: http://webscope.sandbox.yahoo.com/

[135] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 74, no. 3, Sep. 2006, Art. no. 036104.

[136] J. J. Whang, D. F. Gleich, and I. S. Dhillon, "Overlapping community detection using seed set expansion," in *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2013, pp. 2099–2108.

[137] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *J. Res. Nat. Bur. Standards*, vol. 45, no. 4, pp. 1–29, 1950.

[138] J. Chen, O. R. Zaiane, and R. Goebel, "Detecting communities in large networks by iterative local expansion," in *Proc. Int. Conf. Comput. Aspects Social Netw.*, Jun. 2009, pp. 105–112.

[139] Y.-R. Lin, Y. Chi, S. Zhu, H. Sundaram, and B. L. Tseng, "Analyzing communities and their evolutions in dynamic social networks," *ACM Trans. Knowl. Discovery Data*, vol. 3, no. 2, pp. 1–31, 2009.

[140] J. Kunegis, "Konect: The Koblenz network collection," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 1343–1350.

[141] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[142] N. P. Nguyen, T. N. Dinh, S. Tokala, and M. T. Thai, "Overlapping communities in dynamic networks: Their detection and mobile applications," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2011, pp. 85–96.

[143] N. P. Nguyen, T. N. Dinh, Y. Shen, and M. T. Thai, "Dynamic social community detection and its applications," *PLoS ONE*, vol. 9, no. 4, Apr. 2014, Art. no. e91431.

[144] D. A. Spielman and S.-H. Teng, "A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning," *SIAM J. Comput.*, vol. 42, no. 1, pp. 1–26, Jan. 2013.

[145] R. Rossi and N. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 1–2.

[146] K. Christopoulos, G. Baltsou, and K. Tsichlas, "Dynamic community detection with anchors," in *Proc. 11th Int. Conf. Complex Netw. Appl.*, 2022, pp. 1–4.

[147] G. Rossetti, "RDyn: Graph benchmark handling community dynamics," *J. Complex Netw.*, vol. 5, no. 6, pp. 893–912, 2017.

[148] D. Joshi and T. Patalia, "Community detection methods and tools for various complex network," *Int. J. Eng. Res. Technol.*, vol. 13, no. 6, p. 1386, Jun. 2020.

[149] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: An open source software for exploring and manipulating networks," in *Proc. 3rd Int. AAAI Conf. Weblogs Social Media*, 2009, pp. 1–12.

[150] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech., Theory Exp.*, vol. 2008, no. 10, Oct. 2008, Art. no. P10008.

[151] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: A software environment for integrated models of biomolecular interaction networks," *Genome Res.*, vol. 13, no. 11, pp. 2498–2504, 2003.

[152] *Graphviz Graph Visualization Software*. Accessed: Apr. 20, 2022. [Online]. Available: https://graphviz.org

[153] D. Kalamaras, "Social network visualizer (SocNetV)," Social Netw. Anal. Vis. Softw., 2010. Accessed: Apr. 20, 2022. [Online]. Available: http://socnetv.org

[154] W. D. Nooy, A. Mrvar, and V. Batagelj, *Exploratory Social Network Analysis With Pajek*, vol. 46. Cambridge, U.K.: Cambridge Univ. Press, 2018.

[155] A. Barta, G. Palla, P. Pollner, and T. Vicsek, "CFinder," 2009. Accessed: Apr. 20, 2022. [Online]. Available: http://www.cfinder.org/

[156] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, Jun. 2005.

[157] U. Brandes and D. Wagner, "Analysis and visualization of social networks," in *Graph Drawing Software*. Germany: Springer, 2004, pp. 321–340.

[158] G.-H. Ghim, N. Cho, and J. Seo, *NetMiner*. New York, NY, USA: Springer, 2014, pp. 1025–1037.

[159] M. Smith, N. Milic-Frayling, B. Shneiderman, E. M. Rodrigues, J. Leskovec, and C. Dunne, "NodeXL: A free and open network overview, discovery and exploration add-in for excel 2007/2010," 2010. Accessed: Apr. 20, 2022. [Online]. Available: https://nodexl.com/

[160] A. Hagberg, P. Swart, and D. S. Chult, "Exploring network structure, dynamics, and function using NetworkX," Los Alamos Nat. Lab. (LANL), Los Alamos, NM, USA, Tech. Rep. LA-UR-08-05495; LA-UR-08-5495, 2008.

[161] G. Csárdi and T. Nepusz, "The igraph software package for complex network research," *InterJ., Complex Syst.*, vol. 1695, no. 5, pp. 1–9, 2006.

[162] *Neo4j Graph Data Platform*. Accessed: Apr. 20, 2022. [Online]. Available: https://neo4j.com

[163] J. Leskovec and R. Sosič, "SNAP: A general-purpose network analysis and graph-mining library," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 1, pp. 1–20, Oct. 2016.

[164] Y. Wang, Y. Pan, A. Davidson, Y. Wu, C. Yang, L. Wang, M. Osama, C. Yuan, W. Liu, A. T. Riffel, and J. D. Owens, "Gunrock: GPU graph analytics," *ACM Trans. Parallel Comput*, vol. 4, no. 1, pp. 1–49, 2017.

[165] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.

[166] W. Zachary, "An information flow model for conflict and fission in small groups," *J. Anthropolog. Res.*, vol. 33, no. 4, pp. 452–473, Nov. 1976.

[167] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas. (IMC)*, San Diego, CA, USA, Oct. 2007, pp. 29–42.

[168] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson, "The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations," *Behav. Ecol. Sociobiol.*, vol. 54, no. 4, pp. 396–405 2003.

[169] D. Greene, D. Doyle, and P. Cunningham, "Tracking the evolution of communities in dynamic social networks," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2010, pp. 176–183.

[170] D. Chakrabarti, Y. Zhan, and C. Faloutsos, "R-MAT: A recursive model for graph mining," in *Proc. SIAM Int. Conf. Data Mining*, Apr. 2004, pp. 442–446.

[171] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová, "Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, Dec. 2011, Art. no. 066106.

[172] V. Kawadia and S. Sreenivasan, "Sequential detection of temporal communities by estrangement confinement," *Sci. Rep.*, vol. 2, no. 1, pp. 1–10, Dec. 2012.

[173] D. L. Shepelyansky and O. V. Zhirov, "Google matrix, dynamical attractors, and Ulam networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 81, no. 3, Mar. 2010, Art. no. 036213.

[174] Y. Zhang, Y. Xiong, Y. Ye, T. Liu, W. Wang, Y. Zhu, and P. S. Yu, "SEAL: Learning heuristics for community detection with generative adversarial networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 1103–1113.

[175] X. Wang, J. Li, L. Yang, and H. Mi, "Unsupervised learning for community detection in attributed networks based on graph convolutional network," *Neurocomputing*, vol. 456, pp. 147–155, Oct. 2021.

[176] H. Zhou and Y. Zhang, "Unsupervised community detection algorithm based on graph convolution network and social media," *Mobile Inf. Syst.*, vol. 2022, pp. 1–10, Jun. 2022.

GEORGIA BALTSOU received the one tier degree (five years) in computer and communication engineering from the Department of Electrical and Computer Engineering, University of Thessaly, Volos, Greece, and the master's degree in Internet and World Wide Web from the Aristotle University of Thessaloniki, where she is currently pursuing the Ph.D. degree in network analysis with the Department of Informatics. She also worked as an Adult Instructor on informatics in many Vocational Training Centers. Her research interests include static and dynamic networks, community detection in networks, and the notion of causality cast in a network setting. She received her Ph.D. degree with full-time scholarship with the State Scholarships Foundation.

KONSTANTINOS CHRISTOPOULOS received the B.Sc. degree in mathematics and the M.Sc. degree in mathematics of computation and decision making. He is currently pursuing the Ph.D. degree with the Department of Computer Engineering and Informatics, University of Patras. His research interests include community and outlier detection in temporal networks.

KONSTANTINOS TSICHLAS was a Faculty Member with the School of Informatics, Aristotle University of Thessaloniki, from 2008 to 2002. He was a Research Assistant with the Department of Informatics, King's College London, from January 2004 to June 2005. He has been an Assistant Professor with the Department of Computer Engineering and Informatics, University of Patras, since 2020. He have worked in the past (and continue working) on fundamental data structures, like dictionaries and priority queues in various models of computations (PM, RAM, I/O model, and distributed), computational geometry problems, strings matching algorithms with applications to bioinformatics and music analysis, indexing problems of databases and combinatorial optimization problems in large-scale networks mainly with respect to query optimization. He is mainly working on decentralized algorithms and the notion of emergence in a "complex algorithms" setting. In addition, he is very interested and currently working with collaborators on virus propagation on networks, management of temporal networks, agent-based modeling for epidemics, network analysis (mainly community detection), natural algorithms, and the notion of causality cast in a network setting. His research interests include design and analysis of algorithms and data structures.