Acta Univ. Sapientiae Informatica **13,** 2 (2021) 220–250

DOI: 10.2478/ausi-2021-0010

Incremental methods for community detection in both fully and growing dynamic networks

Fariza BOUHATEM
Mouloud Mammeri University
of Tizi-Ouzou, Algeria
email: fariza.bouhatem@ummto.dz

Fatiha SOUAM
Mouloud Mammeri University
of Tizi-Ouzou, Algeria
email: fatiha.souam@ummto.dz

Ali AIT EL HADJ
Mouloud Mammeri University
of Tizi-Ouzou, Algeria
email: ali.aitelhadj.@ummto.dz

Abdelhakim DAFEUR
Mouloud Mammeri University
of Tizi-Ouzou, Algeria
email: abdelhakim.dafeur@ummto.dz

Abstract. In recent years, community detection in dynamic networks has received great interest. Due to its importance, many surveys have been suggested. In these surveys, the authors present and detail a number of methods that identify a community without taking into account the incremental methods which, in turn, also take an important place in dynamic community detection methods. In this survey, we provide a review of incremental approaches to community detection in both fully and growing dynamic networks. To do this, we have classified the methods according to the type of network. For each type of network, we describe three main approaches: the first one is based on modularity optimization; the second is based on density; finally, the third is based on label propagation. For each method, we list the studies available in the literature and state their drawbacks and advantages.

Computing Classification System 1998: G.2.2
Mathematics Subject Classification 2010: 68R15
Key words and phrases: community detection, incremental methods, fully dynamic networks, growing dynamic networks

1 Introduction

In real world that represents complex networks, nodes and edges change over time making the network dynamic. In reality there are two types of dynamic networks: fully dynamic networks [46, 70, 6] and growing dynamic networks [10, 65]. In the first case, like social networks, individuals (nodes) and their relationships (edges) can appear and/or disappear at any time. In the second type like citation networks, nodes represent articles and the links represent citations between the articles. Articles and citations can only be added to the network and they cannot be deleted later. In such type of network, the discovery of so-called community [27] is very hard problem and a number of methods have been proposed to solve it.

Among these methods proposed in the literature to explore community in dynamic networks we find [34, 11, 15] These methods apply static community detection algorithms multiple times to snapshots of the network. These methods are more expensive and it is more effective to incrementally review the community structure of the old network and update the community structure in a timely manner [40, 21, 36].

Several surveys have been proposed in the literature that focus on classifying existing methods designed for detecting community in dynamic networks ignoring incremental methods. Note that these methods , known as incremental methods, came to solve the problems of the traditional approaches while no researcher has classified them in any paper. In addition, in the existing surveys, the authors only deal with fully dynamic networks and do not take into account the growing dynamic networks in which incremental methods are best used for discovering communities.

In this paper, we present a survey of incremental methods for community detection in both fully and growing dynamic networks. This survey can be helpful for understanding of several incremental methods for choosing appropriate method. It is organized as follows. In Section 2, we present a history of community discovery through the listing and classification of methods used in the literature. In Section 3, we talk about dynamic networks and mention the events that can occur in community. We present in Section 4 our classification of various incremental approaches for community detection in both fully and growing dynamic networks as we present in Section 5 a discussion study that explores the advantages and the weaknesses of each approach. Finally, the conclusions section comes to wrap up the paper.

2 Historic for community detection

The graph partitioning methods [7, 38, 24] are originally the first solutions solving the problem of community detection. Graph partitioning relates to supervised methods[1] [31], it allows the classification of large sets and imposes the number of groups to be identified.

In real networks like social networks, the number of communities is not known in advance. In these networks, the number of groups is in itself an important result. A new problem thus posed is to decompose the network into a set of interconnected subgraphs, each constituting what is called a community[2] without knowing a priori the number of communities. Hence the birth of unsupervised methods like static community detection methods [60, 44, 2] and dynamic community detection methods [10, 34, 15].

It was after the publication of the work of Girvan-Newman [28] and Fortunato [27] that the problem of community detection gained interest. [44, 59] use spectral clustering which takes into account the spectral properties based on the eigenvalues of the Laplacian matrix of the data considered [56]. Another type of clustering has been used by the authors of [60] is hierarchical clustering. This approach are devided into two categories: the first category [9, 17, 43] groups the pairs of nodes until all nodes are in the same community [67]. Conversely compared to the first category, the second class [28, 49, 5] consists in dividing the graph into several communities [66] by iteratively eliminating the edges between the nodes until obtaining a singleton node [22, 1]. The authors of [20, 3] use density-based methods which locate regions of high density separated from each other by regions of low density. The DBSCAN[3] algorithm [20] uses parameters as input, the setting of these parameters is difficult to determine. In order to overcome this difficulty, the same authors [3] proposed the method called OPTICS[4] with a basic idea similar to DBSCAN. Another algorithm called DENCLUE[5] [33] measures the effect of each object in its neighborhood. The algorithm is faster than [20] and [3]. In addition to density-based methods, there are other, diffusion-based methods. These consist of the propagation of information to all nodes of the network. Nodes with the same information are grouped together in the same community. Raghavan [50]

[1] In supervised methods, the number and size of communities are known in advance.

[2] Community, cluster, group means the same.

[3] DBSCAN is the abbreviation of Density-Based Spatial Clustering of Applications with Noise.

[4] OPTICS is the abbreviation of Ordering Points to Identify the Clustering Structure.

[5] DENCLUE is the abbreviation of DENsity-based CLUstEring.

and others [61] have proposed the LPA[6] algorithm based on label propagation. As an extension of these algorithms [29] and [62] have presented respectively the COPRA[7] and SLPA[8] algorithms to detect overlapping communities.

In fact, real networks are dynamic, not static, as they evolve over time. This dynamic requires the use of effective methods to detect communities in the network. Among these ways we find, a two-stage approach [34, 11] is proposed to uncover dynamic networks. It independently detects the communities for each snapshot and then matches them. Since the two-stage approach is not entirely satisfactory, dependent community detection methods are introduced [15]. These methods used also snapshots when identifying groups, taking into account the communities found in the previous snapshot but avoiding the need to match them. Other methods designed for detecting community in dynamic networks work directly on temporal networks are the incremental approaches [10, 42, 57]. They start with an initial community, and then update for each incoming change the community structure. We are interested in our survey to dynamic community detection, precisely to incremental methods. Figure 1 summarizes the historic for community detection. The incremental approaches is our aims, so we have detailed it in Section 4.

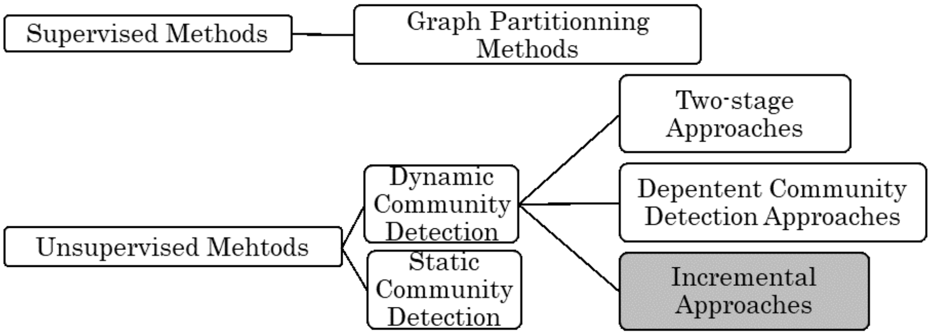

Supervised Methods
Graph Partitionning Methods
Two-stage Approaches
Dynamic Community Detection
Depentent Community Detection Approaches
Unsupervised Mehtods
Static Community Detection
Incremental Approaches

Figure 1: Organization diagram of the historic for community detection

[6] LPA is the abbreviation of Label Propagation Algorithm.

[7] COPRA is the abbreviation of Community Overlap PRopagation Algorithm.

[8] SLPA is the abbreviation of Speaker-listener Label Propagation Algorithm.

3 Dynamic networks

In social networks, the interaction between individual changes over time due to the change in subjects of interest and relationships. The dynamic nature of these networks makes difficult the detection of communities. These dynamic networks can be represented by:

- A time series of static networks, called snapshots (Figure 2) that poorly supports the dynamic aspect. Each snapshot corresponding to interactions derived from a daily, weekly, or monthly collection of data.
- Gathering information in real time as a stream of edges that integrates directly the evolutionary aspect of networks in a two-dimensional space. The nodes are classified on the axis of ordinates and their temporal links between two are represented by arcs over time (Figure 3).

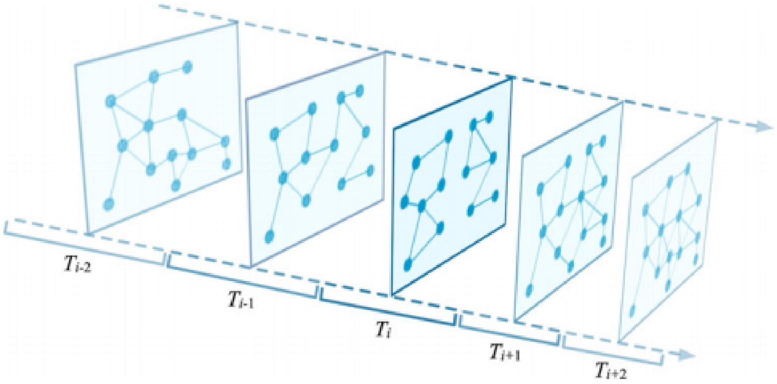


Figure 2: A temporal social network consisting of five timeframes [13]

3.1 Community events

With dynamic networks, nodes and edges can be added or removed at any time. These operations are called "events" and several of them are likely to appear in the life cycle of a community [55, 13, 48, 58]. The examples are shown in Figure 4 and a brief description can be found below:

- Continuing: the community continues its existence in the next time window.

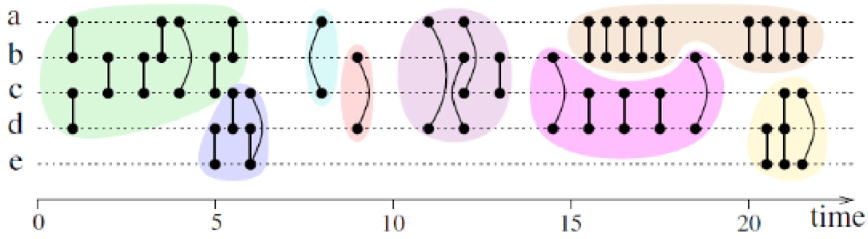


Figure 3: Diagram of a stream of edges with 5 vertices interacting with each other for 20 minutes (communities are colored differently) [32]

- Shrinking: the community shrinks or contracts when it loses some of its members.
- Growing: the community grows when some nodes (member) acquire.
- Splitting: the community is divided into several new communities.
- Merging: the community is divided into several new communities.
- Dissolving: when a community disappears (ends its life). All nodes belonging to this community disappear because they are dispersed between other communities.
- Forming: A community is formed or given birth when it appears at a given time.

3.2 Examination of dynamic community detection

Each algorithm or method designed for community detection must test its performance in a set of networks. These networks can be artificial networks or real networks. In this section, we will present most of the network models proposed in the literature to evaluate the result of the algorithm and the quality of the community structure.

In artificial networks or synthetic network generators, network data (benchmarks) are produced to model real-world networks and used to compare and assess the results of the algorithms. The best known of them are GN benchmark [28] and LFR benchmark [39] but there are others like ABCD [35] and [53] dedicated to community detection. These benchmarks are listed in Table 1. For each benchmark, we present its reference, its name, brief description and the type of generator.

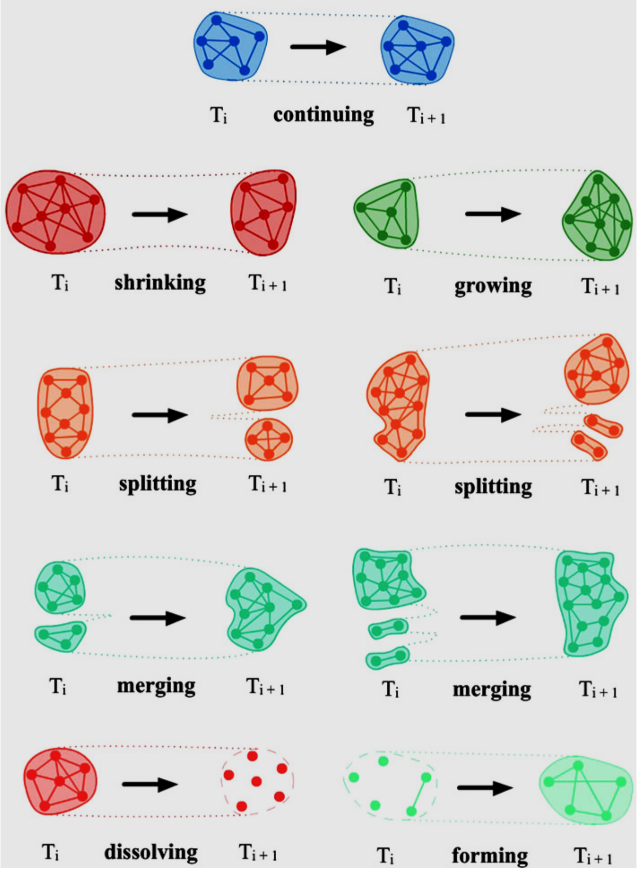


Figure 4: Seven possible type of events in the group evolution [13]

To evaluate the quality of a detected community, one method is to consider ground truth communities. In ground truth, community structure of the network may or may not be known in advance.

- If community structure of the network is known at advance, supervised measures are used. These measures compare the divergence between the community structure produced by community detection algorithms and the effective one. The most famous supervised measure used for the evaluation of quality of detected communities is NMI [39] which represent the degree of dependence between to partition. Equal to 1 if the two partitions are identical or to 0 otherwise. Rand proposed another metric

Reference	Name	Description	Type of generator
[5]	GN (Girvan-Newman benchmark)	- Based on the input of two values: mix parameter and average degree	Static (used also to evaluate dynamic community detection at each snapshot).
[62]	LFR (Lancichinetti Fortunato-Radicchi benchmark	- Community known in advance. - Depend on the mixing parameter.	Static (used also to evaluate dynamic community detection at each snapshot).
[63]	ABCD (Artificial Benchmark for Community Detection)	- Controls the fraction of edges between communities.	Static
[64]	A dynamic graph generator with overlapping community	- The node degrees, the community sizes, and the number of communities per node all follow power law distributions.	Dynamic graph generator with overlapping communities which simulate community scale events.

Table 1: Some benchmarks used in literature [28, 39, 35, 53]

similar to NMI called ARI [51] to measure the similarity between the detected community structure and a gold standard.

- If the ground-truth communities are unknown, unsupervised measures are often used. These measures estimate the quality of the partition without knowing the veritable partitions. Modularity proposed by Newman et al [43] is the most widely used metric quality (or quality function) to compare different community detection algorithm (cf. Section 4.2.1). However, it is not the only one: Kanna et al suggested Conductance metric [37] based on the density of community and the number of links outside. Radicchi et al defined the Internal Density measure [49], it is the ratio between the number of internal edges of the cluster and the number of its all possible internal edges.

All of these metrics used to evaluate the quality of a partition have some drawbacks. We cannot say that one metric is better than another, but certain selection criteria can be used for the choice of the metric. Some drawbacks and advantages of these measures are summarized in Table 2.

4 Incremental approaches for community detection in dynamic networks

Before describing the incremental approaches, we first list some notations and definitions used in this section.

Measure name	Measure type	Drawbacks	Advantages
NMI (Normalized Mutual Information)	Supervised	-You must know the basic community structure while it is unknown in many real dynamic networks. -High computational complexity -Impracticable for large-scale networks.	-Handle small networks.
ARI (Adjusted Rand Index)	Supervised	-Suffer from the problem of deciding if a pair of nodes should be linked or remain unlinked.	-Avoid the problem of unknown cluster. -ARI has the advantage of generating block model data comparing to other link prediction methods.
Modularity	Unsupervised	-Small communities may be not found.	-Handle large-scale network.
Conductance	Unsupervised	-Partition is good if it has both high values of intra and inter community conductance at the same time.	-Efficient in assessing community structure of disjoint communities.
Internal Density	Unsupervised	-Just considers internal relations between nodes of a community without any attention to external ones.	-Possibility to identify small and large communities.

Table 2: Some metric quality used to evaluate community detection algorithm (Drawbacks and advantages)

4.1 Notations and definitions

In this sub-section, we list out notations (Table 3) and introduce some definitions.

Notations	Description
C/C_S	Community / Communities
D	Disjoint communities
O	Overlap communities
In	Intrinsic communities
$V/v/V^{+}_{-}$	Number of vertices / Number of changed (new) vertices / Number of vertices added and deleted
E/ϵ	Number of edges / Number of changed edges
$\langle E\rangle/E^{+}_{-}$	Number of edges in each community / Number of edges added and deleted
I/i	Number of linear iterations in the number of edges E / Number of iterations
k_V	Number of degrees of the updated vertex
$d/d_t/d_v$	Average vertex degree / Average vertex degree at time slice t / the average degree of all vertices
$NbCinf$	Number of communities affected by the change in the network
S	Number of incremental sub-graphs
K	Number of communities
L	$L \ll V$ Maximum of vertices between the infected communities and incremental sub-graph

Table 3: Some notations used in this paper

Definition 1. Community
A group of nodes with similar characteristics or function that interact more densely among themselves.

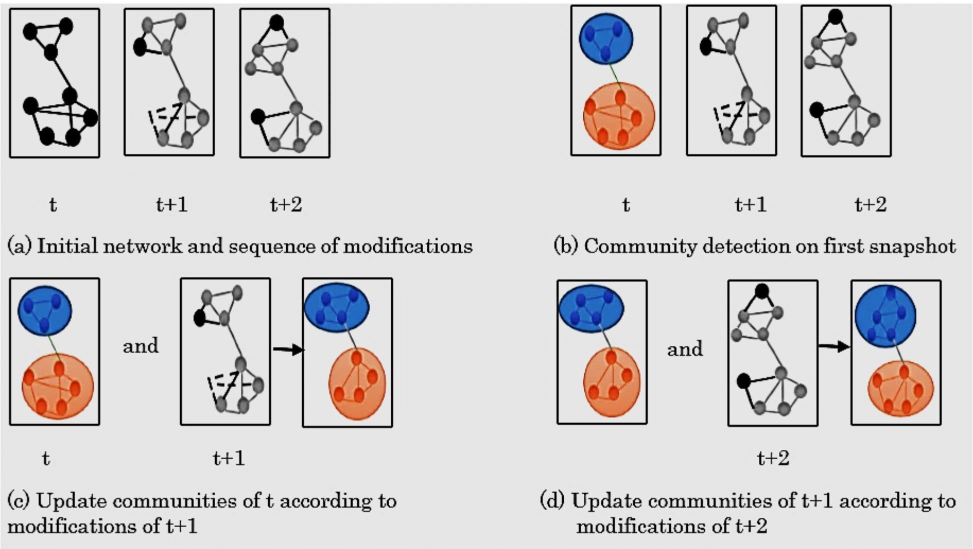
Definition 2. Disjoint community
A subset of members that belong to one and only one community.

Definition 3. Overlap community
Communities that share common members (interests) across different communities.

Definition 4. Intrinsic (embedded) community
Sub-community in which members are strongly connected to each other forming a compact group within the parental community to which they belong.

4.2 Incremental approaches

The incremental approach considers the evolution of network as a series of modification in the network. The input in the network is a sequence of events such as addition and suppression of nodes and edges. The approach starts by finding communities for the initial state of the network. These communities are generally obtained by applying static methods [9, 17], then update, for each incoming change, the community structure by using community structure of the previous time step. As a consequence, the most topologies of network remained unchanged except for a small portion. Moreover, the incremental method often has a memory of what happened and take into account the history and the dynamic of network. The principle of incremental approaches is illustrated in Figure 5. Figure 5a shows the initial network at snapshot t and the sequence of modifications at snapshot t+1 and t+2. In snapshot t+1, a node and its links are added/deleted to / from the initial network. At snapshot t+2, two nodes and its links are added to the network of the snapshot t+1. In Figure 5b, the incremental approaches detect the first communities at t, two communities. After having detected communities at t, the incremental methods update the initial network according to the modification of network at t+1 (Figure 5c) and detect two communities. In Figure 5d, the same process in Figure 5c is applied to detect community at t+2 after updating communities of t+1 based on network changes at t+2.



(a) Initial network and sequence of modifications

(b) Community detection on first snapshot

(c) Update communities of t according to modifications of t+1

(d) Update communities of t+1 according to modifications of t+2

Figure 5: Principle of incremental approaches

Below is a classification of the existing methods for community detection according to the type of network and their based method. We present the set of existing methods by adopting the following hierarchical organization (Figure 6).

– Community detection in fully dynamic networks
- Modularity based optimization methods
- Density based methods
- Label propagation-based methods

– Community detection in growing dynamic networks
- Modularity based optimization methods
- Density based methods
- Label propagation-based methods
- Other methods

4.2.1 Incremental community detection methods in fully dynamic networks

In fully dynamic networks, communities can change or evolve over time by adding and removing nodes and edges. For example, in social networks, indi-

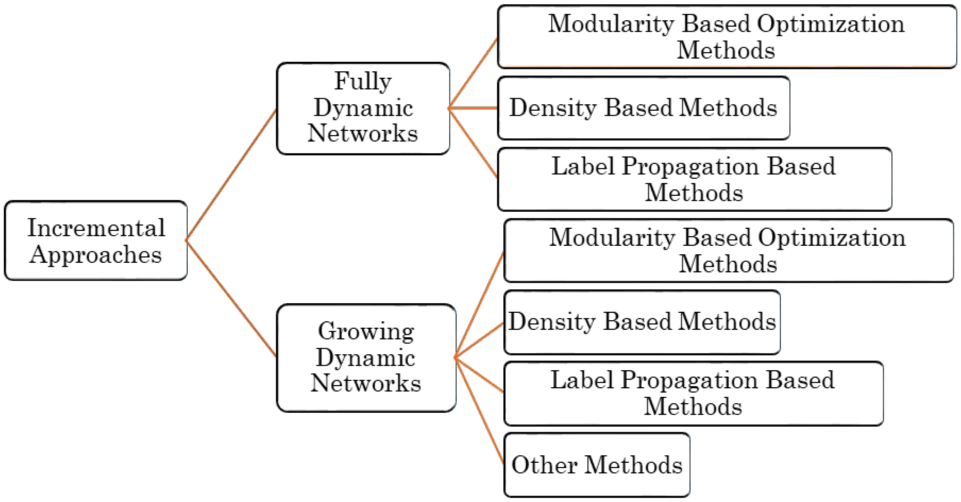

Incremental Approaches
Fully Dynamic Networks
Modularity Based Optimization Methods
Density Based Methods
Label Propagation Based Methods
Growing Dynamic Networks
Modularity Based Optimization Methods
Density Based Methods
Label Propagation Based Methods
Other Methods

Figure 6: Classification chart of the existing incremental approaches for community detection

viduals (nodes) and their relationships (edges) can be added and/or removed at any time. From this explanation of fully dynamic networks, we present the existing incremental approaches for community detection on this field of network. For each category of these methods, we present a list of studies available in the literature. We provide tables (Table 4, 5, 6) in each table we present the following information: the algorithm name and its reference, the type of network in which the approach operates (weighted, unweighted, directed, undirected), the technique used by the algorithm to discover communities when changes occur in the network, the algorithmic complexity (if the algorithmic complexity is not available in the original paper we present it by ——— or by ....), the community type that the algorithm can detect (disjoint, overlap, intrinsic) and the algorithm used by the study during the initial phase to detect initial communities.

- *Modularity based optimization methods*

  The first approach is viewed as one major approach to incremental community detection. It consists on optimization of the metric modularity to detect communities on the network. This metric is proposed in [45] and used to measure the community structure in large scale networks. It is a difference between the fraction of edges inside the community and the fraction of edges expected by random version of the network.

In general methods, the process of maximization of modularity stars by assigning each node in network to different communities and then merge the nodes together until no gain of modularity is possible. Table 4 gives a summary of papers belonging to this category.

[6] introduced a Fast Community Detection for Dynamic Complex Network (FCDDCN) that is a real-time online community detection method in which network changes by the addition and the deletion of edges. The method is based on the modularity optimization using heuristic search and on the greedy agglomerative technique CNM [17]. It starts with an initial community structure obtained by applying CNM[9] algorithm and updates the network structure by using adjacency lists to add and delete elements to the lists.

[46] proposed a Quick Community Adaptation (QCA) for identifying and tracking community structure of dynamic network. The network changes are a collection of simple events such as newVertex, newEdge, removeVertex, and removeEdge. QCA requires an initial community structure that is obtained by performing a static community detection method "Louvain" [9] and the community assignments of neighbor nodes are adjusted by maximizing the modularity.

[18] introduced a new method which is a modification of the original Louvain method [9]. The proposed method keeps the community structure always updated after adding or removing edges and nodes by maximizing the local modularity gain function only for those communities that are affected by modifications to the network.

Contrary to the rule-based[10] methods [46] which considers each network change as an independent event, [16] proposed an Incremental Batch (InBatch) Technique for Community Detection. The method considers the network changes as a batch of network changes that appear in the same network snapshot. The changes that can undergo to the network are vertices/edges addition/deletion and communities can merge, split due to intra edge deletion/addition respectively. To update the community structure, this method initializes all of the new and changed nodes of the current snapshot as singleton communities, then applies Louvain technique until no further increase modularity is possible.

[9] CNM is the abbreviation of the names of the authors Clauset, Newman and Moore, who proposed this algorithm.

[10] Uses predefined rules to specify how to revise vertices' assignment.

Algorithm name and Reference	Network type	Technique used	Algorithmic complexity	Community type		Algorithm used during the initial stage
				D	O	
[41] FCDDCN	Undirected and unweighted	-List of modified edges over time steps.	$O(E d \log V)$	✓		CNM [3].
[35] QCA	Undirected and unweighted	-Considers each network change as an independent event.	$O(\epsilon)$	✓		Louvain [1].
[36] Dynamic Louvain	Undirected and unweighted	-Continuous arrival of information on singular events.	....	✓		Louvain [1].
[39] IncBatch	Undirected and unweighted	-Considers a network changes as a batch appearing in the same network snapshot.	$O(\ (v+\epsilon) * \frac{E}{V} + \lvert V\rvert^*\ )$ $\lvert V\rvert^*$: number of unique vertices after initialization phase	✓		InBatch initialization phase.
[40] LBTR	Undirected and unweighted	-Predict vertices new community assignments after each round of network change.	$O(\ v * \frac{drR}{(1-drR)P}\ )$ R and P: the recall and precision of the classifier r: the probability that an examined vertex actually needs community assignment revision.	✓		Louvain [1] or any static community detection algorithm.
[38] DynaMo	Undirected and weighted	-Processing a set of network changes as a batch.	$O(\epsilon + \lvert E\rvert^*)$ $\lvert E\rvert^*$: number of unique edges after initialization phase	✓		Louvain [1].

Table 4: Incremental community detection methods in fully dynamic networks based on modularity optimization [46, 18, 70, 16, 55, 6]

[55] proposed a Learning-Based Target Revision (LBTR) approach that uses machine learning classifiers and historical community structure information to identify vertices whose community assignment needs to be revised and filters unchanged vertices. To build the vertex classifier LBTR uses Logistic Regression (LR) and Support Vector Machine (SVM), namely LBTR-LR and LBTR-SVM. Similar to previous study [16], this method put the new vertices into singleton communities and move the community assignment of changed vertices to maximize the modularity gain.

The authors of [70] proposed DynaMo[11], a new method designed for maximizing the modularity gain while updating the community structures. The dynamic network is modelled as a sequence of incremental changes: intra-community edge addition/weight increase, cross-community edge addition/weight increase, intra-community edge deletion/weight decrease, cross-community edge deletion/weight decrease, vertex addition, and vertex deletion. For each incremental network change, DynaMo maximizes the modularity in two steps. The first one initializes an intermedi-

[11] DynaMo: Dynamic Community Detection by Incrementally Maximizing Modularity.

ate community structure, depending on the incremental network changes and on the previous network community structure. In the second step, the last two phases of Louvain algorithm [9] are repeated on the intermediate community structure until no gain of modularity is possible.

- *Density based methods*

  The basic idea of density-based approach is to form a group that is dense enough and separated by sparse or low-density region. In other words, the connection of nodes inside a group is greater than the connection of nodes outside the group. Table 5 gives a summary of studies in the literature for community detection based on density.

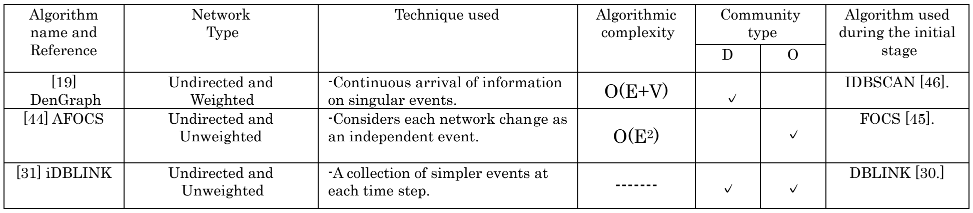
  In [23] the graph mining algorithm DenGraph[12] is proposed which is inspired by Incremental DBSCAN [21] the incremental version of DBSCAN [20]. The author's transfer the basic concepts of these algorithms to graph mining by defining proximity for graph nodes. In others words, they transfer the idea of density based incremental clustering of spatial data to social network structures. The algorithm adjusted locally the vertices (Border vertices, noise vertices) whose distances to the core vertices were changed when the network updates. The same authors proposed DenGraph-HO[13][52] based on the concepts of the DenGraph algorithm to detect overlapping communities.

  [47] used a density [25] as an objective function to Adaptively Finding Overlapping Community Structures (AFOCS) and tracing their evolution over time. The framework identifies the basic overlapped community structure in a network as a collection of dense parts of the network using FOCS[14] and then combines overlapping communities if they share significant substructure and if they are highly overlapped. The changes that can occur to the network are summarized into four events: adding and deleting nodes, adding and deleting edges.

  Lastly, [40] proposed iDBLINK an incremental Density Based LINK clustering algorithm which is an extended version of the static algorithm DBLINK[15] [41]. The algorithm can effectively update the current community structure according to the change of the community structure and network topology a moment before. It focuses on the change of link

[12] DenGraph: A Density based Community Detection Algorithm.
[13] DenGraph-HO: A Density-based Hierarchical Graph Clustering Algorithm.
[14] FOCS is the abbreviation of Finding Overlapping Community Structures.
[15] DBLINK is the abbreviation of Density Based LINK clustering algorithm.

communities through the change of similarity between the edges at the adjacent moments. The changes of link communities are divided into positive changes and negative changes. The network topology updates with positive changes mean that communities grow, merge or create and negative changes mean that communities delete, split or decline. The overlapping community detection quality was improved despite the fact that this algorithm kept the advantage of DBLINK algorithm.



Algorithm name and Reference	Network Type	Technique used	Algorithmic complexity	Community type		Algorithm used during the initial stage
				D	O	
[19] DenGraph	Undirected and Weighted	-Continuous arrival of information on singular events.	$O(E+V)$	✓		IDBSCAN [46].
[44] AFOCS	Undirected and Unweighted	-Considers each network change as an independent event.	$O(E^2)$		✓	FOCS [45].
[31] iDBLINK	Undirected and Unweighted	-A collection of simpler events at each time step.	-------	✓	✓	DBLINK [30.]

Table 5: Incremental community detection methods in fully dynamic networks based on density [52, 40, 47]

- *Label/information propagation-based methods*

  In label propagation methods, each node has its own label which changes by interaction with its neighborhood. The nodes can share the same label to identify a disjoint group or allowing multiple labels to detect overlapping community. Instead of propagating of labels, the information is propagated and exchanged between nodes in the network. The propagation of information between nodes with similar interests tends to be more frequent, as a result, node in the same community have almost the same quantity of information, whereas those in various communities have different quantity of information. Table 6 gives the studies that have exploited these ideas.

  In their work, [63] proposed LabelRankT[16] to detect evolving communities in large scale dynamic network. The algorithm is based on the generalized LabelRank[17] [64] to incorporate important network features such as edge weights and directions. Since each node requires only local information during label propagation process. During the evolution,

[16] LabelRankT: Incremental Community Detection in Dynamic Networks via Label Propagation.

[17] LabelRank: A Stabilized Label Propagation Algorithm for Community Detection in Networks.

nodes and edges are added or removed and only the nodes that are changed between two consecutive snapshots are updated in the algorithm. To give an example, when a new edge is added in the stream, the algorithm updates only the nodes that are attached to this edge. Indeed, the communities can split, merge and dissolve.

The ALPA[18] algorithm was introduced by [30] to detect communities through a local label propagation process. The algorithm consists in adding (or removing) edges/nodes as it appears (or disappears). The edge can be intra-community edge or inter-community edge and the node as an isolated or with its adjacent edges. To update the community structure and avoid unnecessary updates ALPA apply local label propagation process [50]. ALPA perform the warm-up step to propagate labels only inside the target communities, then apply local label propagation process to involve some nodes outside of the target communities.

Other work belonging to this category is [4] that proposed an unsupervised machine learning algorithm based on SLPA [62]. The algorithm used an extended SLPA to detect communities at the first time slot. The first list of labels is obtained and is used to assign the labels to unlabeled nodes at the next time slot. To execute the algorithm, three input arguments are necessary: an input graph, a stop criterion and a threshold.

Lastly, [57] proposed a new framework to detect communities in dynamic networks based on information dynamics. The framework uses information dynamic model to identify initial communities and to incrementally discover community structure of the network. The nodes and edges are added and deleted to the network that leads to the merging, division, expansion, contraction, birth and death of communities.

[18] ALPA is the abbreviation of Adaptive Label Propagation Algorithm.

Algorithm name and Reference	Network type	Technique used	Algorithmic complexity	Community type		Algorithm used during the initial stage
				D	O	
[32] LabelRankT	Directed and weighted undirected and unweighted	-Considers a network changes as a batch appearing in the same network snapshot.	$O(IE)$	✓		LabelRank [33]
[29] ALPA	Undirected and unweighted	-Considers each network change as an independent event.	$O(t*i < E >)$ t: time step	✓		Any available static methods or LPA [25].
[34] ISLPA	Directed and undirected	-A network changes is a batch of nodes and edges that are removed or added over time.	$O(TE)$ T: a stop criterion	✓	✓	ESLPA [34].
[54] DCDID	Undirected	-Batch processing technique.	(1) In initial community partition: $O(i*V*d)$ (2) In incremental community detection $O(E^{+}_{-} + i * V^{+}_{-} * d_t)$	✓		CDID [54].

Table 6: Incremental community detection methods in fully dynamic networks based on label/information propagation [30, 63, 4, 57]

Some abbreviations in Table 6 described in footnote: ISLPA[19], ESLPA[20], DCDID[21], CDID[22].

4.2.2 Incremental community detection methods in growing dynamic networks

We agree that all networks change over time are dynamic networks. In this type of network, there are networks that evolve only by adding nodes and their links and cannot be removed later. For example, in research article citations networks, most articles cite the previous work by other authors on the same topic. These citations form a network in which nodes represent articles and edges represent the oriented link from article A to article B which indicates that B is cited by A. These networks are a special case of dynamic networks and for this reason we have adopted the name "growing dynamic networks". Tables 7, 8, 9, and 10 list the studies belonging to this category, we present in each table : the algorithm name and its reference, the type of network in which the approach works (weighted, unweighted, directed, undirected), changes that refer to

[19] ISLPA is the abbreviation of Incremental Speaker-Listener Propagation Algorithm.

[20] ESLPA is the abbreviation of Extended Speaker-Listener Label Propagation Algorithm.

[21] DCDID is the abbreviation of Dynamic Community Detection based on Information Dynamics

[22] CDID is the abbreviation of Community Detection based on Information Dynamics.

the review strategies that an algorithm applies when updating networks (community events), the technique used by the algorithm to discover communities when changes occur in the network, the algorithmic complexity, if it is available in the original paper, the community type that the algorithm can detect (disjoint, overlap, intrinsic) and the algorithm used by the study during the initial phase to detect initial communities.

- *Modularity based optimization methods*

  Authors of [54] proposed GreMod[23], a real time detecting algorithm to track community structure of growing dynamic networks. Their method consists in adding of new edges between: (1) two nodes already exist and belong to the same community, (2) two nodes already exist and belong to different communities, (3) new node and old node, (4) two new nodes. It starts with initial communities calculated by Louvain [9] algorithm, then apply their incremental strategies in a way to increase the modularity if possible.

  In [65], Authors introduced NGI[24], a novel approach for node-grained streaming network in which changes arrive sequentially and frequently. The method adds a single node with its connecting edges in the network simultaneously and all the edges at the same time. The node added to the network can come: (1) without any neighbor, (2) with multiple neighbors belonging to an isolating singleton community, (3) with multiple neighbors belonging to different communities.

  [68] proposed Δ–screening technique to quickly identify the relevant parts of the graph that are potentially impacted by a batch of changes ((1) adding edges). The approach assigns firstly all vertices to a distinct community using any static community detection algorithm [9, 17]. After an iterative process, the method identifies all vertices whose community affiliation could potentially change and retrain the previous community assignments to the remaining vertices. The vertex that changes his affiliation migrates to a neighboring community that maximizes the modularity gain of that vertex. Table 7 lists the works talk over in this section.

[23] GreMod: A Real-Time Detecting Algorithm for Tracking Community Structure of Dynamic Networks.

[24] NGI is the abbreviation of Node-Grained Incremental community detection algorithm.

Algorithm name and Reference	Network type	Changes	Technique used	Algorithmic complexity	Community type		Algorithm used during the initial stage
					D	O	
[47] GreMod	Undirected and Weighted	-(1)-> No change. -(2)->No change/merge C_S. -(3)->assigning node to an existing C / create new C. -(4)-> assigning node to an existing C / create new C.	-The edges added to the network are processed one by one.	$O(\epsilon)$	✓		Louvain [1].
[57] NGI	Undirected and unweighted	-(1)- > New isolated C. -(2) ->Aggregate all nodes in the adjacent C_s with newly arrived node into a community. -(3) -> Insert a new node in one of its adjacent C_s or merge similar C_s with new coming node.	-Node with its connecting edges is added into network simultaneously and all edges arrive at the same time.	$O(E * d_v)$	✓		Detect the initial communities after the arriving of the first node in the network.
[43] Δ-screening	Undirected and weighted or unweighted	(1) (not specified the type of edges)->reevaluate all vertices whose community affiliation could potentially change.	-Considers the addition of edges as a batch appearing in the same network snapshot.	________	✓		Any static community detection algorithm.

Table 7: Incremental community detection methods in growing dynamic networks based on modularity optimization [68, 54, 65]

- *Density based methods*

To detect and track the evolution of hierarchical and overlapping communities in evolving networks [8] proposed a novel approach called HOC-Tracker. The approach identifies a cluster by detecting the neighborhood of each node in the network using a distance function. Then for each new event, it classifies the active nodes that have caused the network to change in order to track evolutionary events like birth, death, merge, split, growth and shrink of communities.

[42] proposed InDEN (Intrinsic Community Detection in Evolving Networks) method to find intrinsic community in growing dynamic networks. The method starts with the initialization of community containing the first two nodes that are incoming with the first edge. Then, InDEN uses the membership score to assign any new incoming node into the community with maximum score. To detect intrinsic communities, the approach analyses the density variation in community at each time.

In their work, [10] proposed a density-based approach with dual optimization to track and identify the community structure. The proposed method starts from an initial community obtained by Louvain [9], then for each event that occurs in the network, a new node and their links are added in the best community after having passed by two levels of density optimization. This optimization is calculated only in the community affected by the changes. Table 8 lists the studies discussed in this section.

Algorithm name and Reference	Network Type	Changes	Technique used	Algorithmic complexity	Community type			Algorithm used during the initial stage
					D	O	In	
[53] HOCTracker	Directed/ undirected and Weighted/ unweighted	-New core node emerges ->expansion/birth/merge of C_S. -A core node becomes non-core->split/shrink of C_S. - A core node gains nodes/ loses nodes but remains a core->merge, growth, split, shrink of C_S.	-Considers the active node and their direct neighbors at each new state of the network.	O(E)		✓		Initial community obtained by HOCTracker [53].
[48] InDEN	Undirected and unweighted	-New edges between two existing nodes or a new edge between one existing node and new node ->movement of node from one C to another/ merge of C_S / create new C.	-Adds edges to the community independently.	—	✓		✓	—
[37]	Undirected and unweighted	-New isolated node -> create new C. -New node with its links in the same C -> reinforce the infected C. -New node with its links in different C_S-> split/merge /birth of infected C_S.	-For each time stamp adding node and its links simultaneously.	O(nbCinf)	✓			Louvain [1] or Any static community detection algorithm [3].

Table 8: Incremental community detection methods in growing dynamic networks based on density [10, 42, 8]

Some abbreviations in Table 8 described in footnote: HOCTracker[25], core node[26]

- *Label propagation-based methods*

  Hiroki et al. [36] proposed a new algorithm to detect communities for high volume graphs stream based on DEMON[27] [19] algorithm. It used three functions for this issue, two incremental functions ((1) Ego minus ego network, (2) Label propagation) and (3) optimized merge function. Table 9 summarizes the papers belonging to this class.

[25] HOCTracker: Tracking the Evolution of Hierarchical and Overlapping Communities in Dynamic Social Networks.

[26] Core node is a node having non-zero reciprocated interactions with any of its neighbor(s) in a set of nodes of the network.

[27] DEMON: a local-first discovery method for overlapping communities.

Algorithm name and Reference	Network Type	Changes	Technique used	Algorithmic complexity	Community type			Algorithm used during the initial stage
					D	O	In	
[51] Incremental DEMON	Undirected and unweighted	-Adding nodes and his edges/Adding edges. between existing nodes-> merge of C_S.	-Batch technique.	(1): O(V+E)*d (2): O(V+d^2) (3): $\vert K_V \vert$*V	✓	✓		DEMON [52]

Table 9: Incremental community detection methods in growing dynamic networks based on label propagation [36]

- *Other methods*

Algorithm name and Reference	Network type	Changes	Technique used	Algorithmic complexity	Community type			Algorithm used during the initial stage
					D	O	In	
iLCD [50]	Undirected and unweighted	-(1) and (2)->updating of existing C_S /create new C /merge similar C_S.	-For each time stamped adding a set of edges simultaneously.	It's nearly impossible to determine the complexity of iLCD.		✓	✓	————
[49]	Undirected and unweighted	-(1) ->birth of new C. -(2) ->keep or not the edges in their C. -(3) ->birth of new C/ enlarging of the previous C_S. -(4) ->birth of new C/strengthening of previous C_S.	-Joining simultaneously incremental subgraph in the network.	O(V + S + K + L)	✓			————

Table 10: Some other incremental community detection methods in growing dynamic networks [69, 14]

A part from that, [14] proposed iLCD (intrinsic Longitudinal Community Detection) algorithm to find intrinsic and overlapping communities. The events that can occur to the network are: (1) adding edges between existing nodes, (2) adding edges between new nodes and existing nodes. The community are detected by doing re-evaluation at each new iteration according to the path lengths between each node and its adjacent communities.

[69] proposed a new incremental method capable of handling subgraphs (including nodes and edges) addition. To do this, the method proposes four types of incremental elements: (1) complete independent, (2) complete contained, (3) mixed with new and old nodes and (4) multiple contained. It then applies its own update strategies based on the edge weights to determine the impact of historical relationship. Finally, it outputs the latest communities and updates

the evolutionary path accordingly. Table 10 reviews some other methods for community detection in growing dynamic networks.

5 Discussion study

We discuss in this section the weaknesses and strengths of each incremental approach based on modularity optimization, density and label propagation. An overview is summarized in Table 11. We present then in general terms the main advantages and disadvantages of using of incremental approaches for community detection in fully dynamic networks and in growing dynamic networks. The advantages and disadvantages are listed in Table 12.

5.1 Modularity based optimization methods

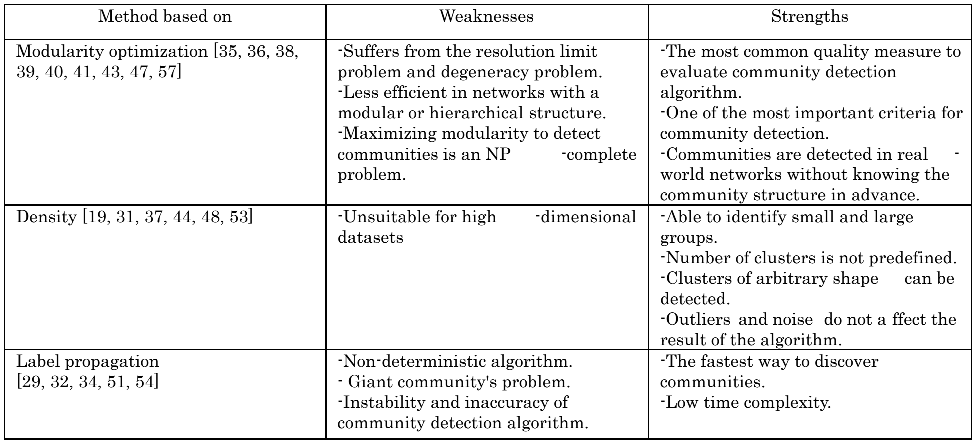
Most of these approaches aim to maximize modularity in order to assess the quality and robustness of their detected communities. This scale is one of the most important metrics because it has the unique advantage of being a universal standard at the same time for defining communities and it is a key component of the most common method of graph clustering. These methods are suitable for temporal networks whose stream changes or evolves over time and for a stochastic network that not expected to have a cluster structure. Despite these strengths, some limitations have been noted about its performance. First, small communities may not be found. This limitation is called resolution limit problem [12, 26] and is considered one of the most serious problems in detecting communities. Another problem known as degeneracy problem, when modularity maximization finds so many different partitions whose typical values are very close to each other. This problem is most grave when applied to networks with modular structure; it happens for weighted, directed, bipartite and multi-scale generalizations of modularity. Finally, maximizing modularity is NP-complete problem, so the solution cannot be found in polynomial-time.

5.2 Density based methods

Most of the algorithms belonging in this class are able to identify groups that are more connected inside and less connected outside. Its aim is to discover small and large clusters thus solving the resolution limit problem resulting from modularity optimization. Other strengths of these methods: is that the number of clusters is not required as an input to the algorithm but rather is disclosed in the algorithm based on data set characteristics, insensitive to noise[28] and outliers[29] and capable to identify clusters of arbitrary shapes[30]. Even though these advantages, these approaches still have weaknesses it is unsuitable for high-dimensional datasets because of the inherent shortage of the feature space, which in turn, reduces any clustering propensity.

5.3 Label propagation-based methods

Label propagation approaches are considered as one of the fastest methods of community detection with a near linear time complexity. At this speed, these methods are suitable for large-scale networks. One of the main weaknesses of these approaches is that in the general case the algorithm is non-deterministic, the multiplicity execution of the same algorithm produces no unique solution, but an aggregate of many solutions. This problem remains important, and all label propagation-based methods attempt to overcome it. Another potential issue is the uncertainty and randomness in the label propagation process. This problem may affect the stability and accuracy of detected communities. Also, the bad propagation of labels can lead to the discovery of huge communities without sense, this problem is known as "Giant community's problem".



Method based on	Weaknesses	Strengths
Modularity optimization [35, 36, 38, 39, 40, 41, 43, 47, 57]	-Suffers from the resolution limit problem and degeneracy problem. -Less efficient in networks with a modular or hierarchical structure. -Maximizing modularity to detect communities is an NP -complete problem.	-The most common quality measure to evaluate community detection algorithm. -One of the most important criteria for community detection. -Communities are detected in real - world networks without knowing the community structure in advance.
Density [19, 31, 37, 44, 48, 53]	-Unsuitable for high -dimensional datasets	-Able to identify small and large groups. -Number of clusters is not predefined. -Clusters of arbitrary shape can be detected. -Outliers and noise do not a ffect the result of the algorithm.
Label propagation [29, 32, 34, 51, 54]	-Non-deterministic algorithm. - Giant community's problem. -Instability and inaccuracy of community detection algorithm.	-The fastest way to discover communities. -Low time complexity.

Table 11: Some weaknesses and strengths of incremental methods based on modularity optimization, density, and label propagation [52, 30, 40, 63, 4, 46, 18, 10, 70, 16, 55, 6, 68, 47, 54, 42, 36, 8, 57, 65]

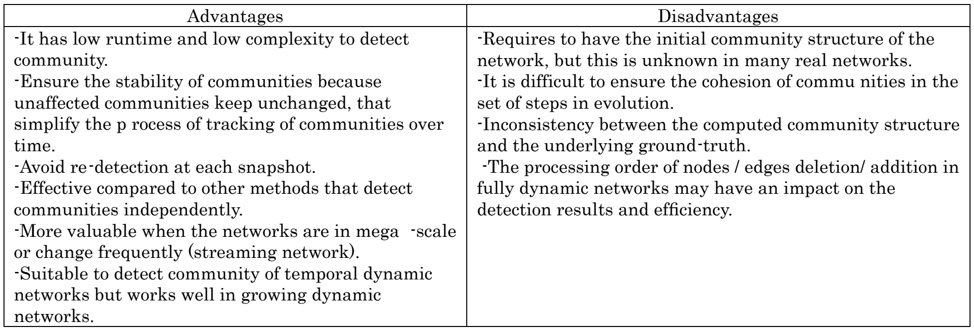
[28] Node with area of low density.

[29] Nodes which cannot be grouped into any of the communities.

[30] Non-convex shape.

5.4 Advantages and disadvantages of incremental methods for community detection in both fully and growing dynamic networks

We return to the first tree level of our classification (Figure 6) to discuss the main advantages and disadvantages of incremental community detection methods in fully and growing dynamic networks (Table 12). In general, the incremental approach has the advantage in runtime and computational complexity because it updates the current community structure according to the change of the network and the previous community structure, thus it can also avoid the hassles of re-detection. There are other advantages of ensuring stable communities across time steps which allows avoiding the instability problem taken from methods that detect communities at each time steps undependably. As a result, it deals with temporal networks[31] that change frequently and works well in growing dynamic networks. Incremental clustering has been used as one of the most efficient methods; however, it requires initial community structure of the network, but this is unknown in many real networks. Another defect is the accumulation of errors resulting from the partition error which will lead to a discrepancy between the computed community structure and the underlying ground-truth. As a result, it is difficult to ensure the cohesion of communities in the set of steps in evolution because tracking a community is only done when switching from one snapshot to another. Another problem is the possibility of neglecting of some changes in the associated community structure due to the detection of only local structure of the network.



Advantages	Disadvantages
-It has low runtime and low complexity to detect community. -Ensure the stability of communities because unaffected communities keep unchanged, that simplify the p rocess of tracking of communities over time. -Avoid re-detection at each snapshot. -Effective compared to other methods that detect communities independently. -More valuable when the networks are in mega -scale or change frequently (streaming network). -Suitable to detect community of temporal dynamic networks but works well in growing dynamic networks.	-Requires to have the initial community structure of the network, but this is unknown in many real networks. -It is difficult to ensure the cohesion of commu nities in the set of steps in evolution. -Inconsistency between the computed community structure and the underlying ground-truth. -The processing order of nodes / edges deletion/ addition in fully dynamic networks may have an impact on the detection results and efficiency.

Table 12: Major disadvantages and advantages of incremental approaches in fully and growing dynamic networks

[31] A temporal network also known as time-varying network is a dynamic network in which both nodes and edges may appear and disappear as time goes.

Abbreviation in Table 12 described in footnote: Unaffected communities (column one)[32].

6 Conclusions

An incremental approach is one way to discover communities in both fully and growing dynamic networks. The idea of these approaches is to build and maintain communities in a network, following a series of changes that occur to the network itself.

In this paper, a study of incremental methods for detecting communities in both fully and growing dynamic networks is presented and discussed. The aim of this survey was to classify the incremental methods and discuss their disadvantages and advantages. Both points will help the reader to define and find direction for his future research and choose the appropriate method depending on the type of networks, type of communities and the technique used to identify community.

Detecting communities is NP-hard problem. Therefore, defining the community accurately and effectively in dynamic networks remains a very interesting and challenging task and incremental approaches try to solve it.

References

[1] Y. Y. Ahn, J. P. Bagrow, S. Lehmann, Link communities reveal multi-scale complexity in networks, *Nature*, **466**, 7307 (2010) 761–764. ⇒222

[2] E. Akachar, B. Ouhbi, B. Frikh, A new algorithm for detecting communities in social networks based on content and structure information, *International Journal of Web Information Systems*, 2019. ⇒222

[3] M. Ankerst, M. M. Breunig, H. P. Kriegel, J. Sander, OPTICS: Ordering Points To Identify the Clustering Structure, *ACM Sigmod record*, **28**, 2 (1999) 49–60. ⇒222

[4] M. Asadi, F. Ghaderi, Incremental community detection in social networks using label propagation method, *23rd Conference of Open Innovations Association (FRUCT)*, 2018, pp. 39–47. ⇒236, 237, 243

[5] J. Bagrow, E. Bollt, A local method for detecting communities, *Physical review E*, **72**, 4 (2005) 46–108. ⇒222

[6] S. Bansal, S. Bhowmick, P. Paymal, Fast community detection for dynamic complex networks, *Complex Networks, Springer*, Berlin, Heidelberg, 2011, pp. 196–207. ⇒221, 232, 233, 243

[32] Unaffected communities are communities those unconcerned by the change that occur to the network.

[7] E. R. Barnes, An Algorithm for Partitioning the Nodes of a Graph, *SIAM Journal on Algebraic and Discrete Methods*,**3**, 4(1982) 541–550. ⇒222

[8] S. Y. Bhat, M. Abulaish, HOCTracker: Tracking the evolution of hierarchical and overlapping communities in dynamic social networks, *IEEE Transactions on Knowledge and Data engineering*, **27**, 4(2014) 1019–1013. ⇒239, 240, 243

[9] V. D. Blondel, J. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, *Journal of Statistical Mechanics: Theory and Experiment*, **2008**, 10 (2008) P10008. ⇒222, 229, 232, 234, 238, 239

[10] F. Bouhatem, A. A. Ait El Hadj, F. Souam, Density-based Approach with Dual Optimization for Tracking Community Structure of Increasing Social Networks, *International Journal on Artificial Intelligence Tools*, **29**, 01(2020) 2050002. ⇒ 221, 222, 223, 239, 240, 243

[11] L. Boujlaleb, A. Idarrou, D. Mammass, Tracking community evolution in social networks, *Journal of theoretical and applied information technology*, **95**, 22 (2017) . ⇒221, 223

[12] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, D. Wagner, On modularity clustering, *IEEE transactions on knowledge and data engineering*, **20**, 2 (2007) 172–188. ⇒242

[13] P. Bródka, S. Saganowski, P. Kazienko, GED: the method for group evolution discovery in social networks, *Social Network Analysis and Mining*, **3**, 1(2013), 1–14, ⇒224, 226

[14] R. Cazabet, F. Amblard, C. Hanachi, Detection of overlapping communities in dynamical social networks, *IEEE second international conference on social computing*, 2010, pp. 309–314. ⇒241

[15] D. Chakrabarti, R. Kumar, A. Tomkins, Evolutionary clustering, *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp.554–560, ⇒221, 222, 223

[16] W. H. Chong, L. N. Teow, An incremental batch technique for community detection, *Proceedings of the 16th international conference on information fusion*, 2013, pp. 750–757. ⇒232, 233, 243

[17] A. Clauset, M. E. J. Newman, C. Moore, Finding community structure in very large networks, *Physical review E*, **70**, 6 (2004) 066111. ⇒222, 229, 232, 238

[18] M. Cordeiro, R. P. Sarmento, J. Gama, Dynamic community detection in evolving networks using locality modularity optimization, *Social Network Analysis and Mining*, **6**, 1(2016) 15. ⇒232, 233, 243

[19] M. Coscia, G. Rossetti, F. Giannotti, D. Pedreschi, Demon: a local-first discovery method for overlapping communities, *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 615–623. ⇒240

[20] M. Ester, H. P. Kriegel, J. Sander, X. Xu, A density-based algorithm for discovering clusters in large spatial databases with noise, Proceedings of the 2nd International Conference on Knowledge Discovery and Data mining, 1996, pp. 226–231. ⇒222, 234

[21] M. Ester, R. Wittmann, Incremental generalization for mining in a data warehousing environment, *International Conference on Extending Database Technology, Springer*, Berlin, Heidelberg,1998, pp. 135–149. ⇒221, 234

[22] T. S. Evans, R. Lambiotte, Line graphs, link partitions and overlapping communities, *Physical review E*, **80**, 1 (2009) 016105. ⇒222

[23] T. Falkowski, A. Barth, M. Spiliopoulou, Dengraph: A density-based community detection algorithm, *IEEE/WIC/ACM International Conference on Web Intelligence (WI'07)*, 2007, pp. 112–115. ⇒234

[24] C. M. Fiduccia, R. M. Mattheyses, A Linear-Time Heuristic for Improving Network Partitions, *19th Design Automation Conference, IEEE*, 1982, pp. 175–181. ⇒222

[25] S. Fortunato, C. Castellano, Community structure in graphs, ArXiv preprint arXiv: 0712.2716, 2007. ⇒234

[26] S. Fortunato, M. Barthelemy, Resolution limit in community detection, *Proceedings of the national academy of sciences*, **104**, 1(2007) 36–41. ⇒242

[27] S. Fortunato, Community detection in graphs, *Physics Reports*, **486**, 3(2010) 75–174. ⇒221, 222

[28] M. Girvan, M. E. J. Newman, Community structure in social and biological networks, *Proceedings Of the National Academy Of Sciences* , **99**,12 (2002) 7821–7826. ⇒222, 225, 227

[29] S. Gregory, Finding overlapping communities in networks by label propagation, New Journal of Physics, **12**, 10 (2010) 103018. ⇒223

[30] J. Han, W. Li, L. Zhao, Z. Su, Y. Zou, W. Deng, Community detection in dynamic networks via adaptive label propagation, *PloS one*, **12**, 11 (2017) e0188655. ⇒236, 237, 243

[31] J. A. Hartigan, M. A. Wong, Algorithm AS 136: A K-Means Clustering Algorithm, *Journal of the Royal Statistical Society, Series C*, **28**, 1(1979) 100–108. ⇒222

[32] S. Heymann, *Exploratory link stream analysis for event detection*, Doctoral dissertation, Université Pierre et Marie Curie-Paris VI, Paris, 2013. ⇒225

[33] A. Hinneburg, D. A. Keim, An Efficient Approach to Clustering in Large Multimedia Databases with Noise, *Knowledge Discovery and Data Mining*, New York City, USA, 1998, pp. 58–65. ⇒222

[34] J. Hopcroft, O. Khan, B. Kulis, B. Selman,Tracking evolving communities in large linked networks, *Proceedings of the national academy of sciences of the United States of America*, 101 (2004) 5249–5253. ⇒221, 222, 223

[35] B. Kaminski, T. Olczak, Artificial Benchmark for Community Detection (ABCD), *Joint Mathematics Meetings (JMM), AMS*, 2021. ⇒225, 227

[36] H. Kanezashi, T. Suzumura, An incremental local-first community detection method for dynamic graphs, *IEEE International Conference on Big Data (Big Data)*, 2016, pp. 3318–3325. ⇒221, 240, 241, 243

[37] R. Kannan, S. Vempala, A. Vetta, On clusterings: Good, bad and spectral, *Journal of the ACM (JACM)*, **51**, 3(2004) 497–515. ⇒227

[38] B.W. Kernighan, S. Lin, An efficient heuristic procedure for partitioning graphs, *Bell System Technical Journal*, **49**, 2 (1970) 291–307. ⇒222

[39] A. Lancichinetti, S. Fortunato, F. Radicchi, Benchmark graphs for testing community detection algorithms, *Physical review E*, **78**, 4 (2008) 046110. ⇒225, 226, 227

[40] F. Meng, F. Zhang, M. Zhu, Y. Xing, Z. Wang, J. Shi, Incremental density-based link clustering algorithm for community detection in dynamic networks, *Mathematical Problems in Engineering*, 2016. ⇒221, 234, 235, 243

[41] Z. Mu, M. Fanrong, Z. Yong, Density-based link clustering algorithm for overlapping community detection, *Journal of computer research and development*, **50**, 12 (2013) 2520. ⇒234

[42] K. Nath, S. Roy, Detecting intrinsic communities in evolving networks, *Social Network Analysis and Mining*, **9**,1 (2019)1–15. ⇒223, 239, 240, 243

[43] M. E. J. Newman, M. Girvan, Finding and evaluating community structure in networks, *Physical review E*, **69**, 2 (2004) 026113. ⇒222, 227

[44] M. E. Newman, Spectral methods for community detection and graph partitioning, *Physical Review E*, **88**, 4 (2013) 042822. ⇒222

[45] M. E. Newman, Detecting community structure in networks, *The European Physical Journal B-Condensed Matter and Complex Systems*, **38**, 2 (2004) 321–330. ⇒231

[46] N. P. Nguyen, T. N. Dinh, Y. Xuan, , M. T. Thai, Adaptive algorithms for detecting community structure in dynamic social networks, *Proceedings IEEE INFOCOM*, 2011, pp. 2282–2290. ⇒221, 232, 233, 243

[47] N. P. Nguyen, T. N. Dinh, S. Tokala, M. T. Thai, Overlapping communities in dynamic networks: their detection and mobile applications, *Proceedings of the 17th annual international conference on Mobile computing and networking*, 2011, pp. 85–96. ⇒234, 235, 243

[48] G. Palla, A. L. Barabási, T. Vicsek, Quantifying social group evolution, *Nature*, **446**, 7136 (2007) 664–667. ⇒224

[49] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, D. Parisi, Defining And identifying communities in networks, *Proceedings of the National Academy of Sciences*, **101**, 9 (2004) 2658–2663. ⇒222, 227

[50] U. N. Raghavan, R. Albert, R. Kumara, Near linear time algorithm to detect community structures in large-scale networks, *Physical Review E*, **76**, 3 (2007) 036106. ⇒222, 236

[51] W. M. Rand, Objective criteria for the evaluation of clustering methods, *Journal of the American Statistical association*, **66**, 336 (1971) 846–850. ⇒227

[52] N. Schlitter, T. Falkowski, J. Lässig, DenGraph-HO: a density-based hierarchical graph clustering algorithm, *Expert Systems*, **31**, 5 (2014) 469–479. ⇒234, 235, 243

[53] N. Sengupta, M. Hamann, D. Wagner, Benchmark generator for dynamic overlapping communities in networks, *IEEE International Conference on Data Mining (ICDM)*, 2017, pp. 415–424. ⇒225, 227

[54] J. Shang, L. Liu, F. Xie, Z. Chen, J. Miao, X. Fang, C. Wu, A real-time detecting algorithm for tracking community structure of dynamic networks, *ArXiv preprint arXiv: 1407.2683*, 2014. ⇒238, 239, 243

[55] J. Shang, L. Liu, X. Li, F. Xie, C. Wu, Targeted revision: A learning-based approach for incremental community detection in dynamic networks, *Physica A: Statistical Mechanics and its Applications*, **443**, (2016)70–85. ⇒224, 233, 243

[56] J. Shi, J. Malik, Normalized cuts and image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**, 8, (2000) 888–905. ⇒222

[57] Z. Sun, J. Sheng, B. Wang, A. Ullah, F. Khawaja, Identifying Communities in Dynamic Networks Using Information Dynamics, *Entropy*, **22**, 4(2020) 425. ⇒ 223, 236, 237, 243

[58] M. Takaffoli, F. Sangi, J. Fagnan, O. R. Zäıane, Community evolution mining in dynamic social networks, *Procedia-Social and Behavioral Sciences*, **22**, (2011) 49–58. ⇒224

[59] Y. Van Gennip, B. Hunter, R. Ahn, P. Elliott, K. Luh, M. Halvorson, ... and P. J. Brantingham, Community detection using spectral clustering on sparse geosocial data, *SIAM Journal on Applied Mathematics*, **73**, 1, (2013) 67–83. ⇒ 222

[60] D. R. White, F. Harary, M. Sobel, M. Becker,The cohesiveness of blocks in social networks: node connectivity and conditional density, *Sociological Methodology*, **31**, 1(2001) p. 305–359. ⇒222

[61] J. Xie, B. K. Szymanski, Community detection using a neighborhood strength driven label propagation algorithm, *IEEE Network Science Workshop*, 2011, pp. 188–195. ⇒223

[62] J. Xie, B. K. Szymanski, Towards linear time overlapping community detection in social networks, *Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer*, Berlin, Heidelberg, 2012, pp. 25–36. ⇒223, 236

[63] J. Xie, M. Chen, B. K. Szymanski, LabelrankT: Incremental community detection in dynamic networks via label propagation, *Proceedings of the workshop on dynamic networks management and mining*, 2013, pp. 25–32. ⇒235, 237, 243

[64] J. Xie, B. K. Szymanski, Labelrank: A stabilized label propagation algorithm for community detection in networks, *IEEE 2nd Network Science Workshop (NSW)*, 2013, pp. 138–143. ⇒235

[65] S. Yin, S. Chen, Z. Feng, K. Huang, D. He, P. Zhao, M. Y. Yang, Node-grained incremental community detection for streaming networks, *IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*, 2016, pp. 585–592. ⇒ 221, 238, 239, 243

[66] N. Yuruk, M. Mete, X. Xu, T. A. J. Schweiger, A Divisive Hierarchical Structural Clustering Algorithm for Networks, *Seventh IEEE Int. Conf. on Data Mining Workshops (ICDMW 2007)*, Omaha, NE, 2007, pp. 441-448. ⇒222

[67] N. Yuruk, M. Mete, X. Xu, T. A. J. Schweiger, AHSCAN: Agglomerative Hierarchical Structural Clustering Algorithm for Networks, *International Conference on Advances in Social Network Analysis and Mining*, Athens, 2009, pp. 72–77. ⇒222

[68] N. Zarayeneh, A. Kalyanaraman, A fast and efficient incremental approach toward dynamic community detection, *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2019, pp. 9–16. ⇒238, 239, 243

[69] Z. Zhao, C. Li, X. Zhang, F. Chiclana, E. H. Viedma, An incremental method to detect communities in dynamic evolving social networks, *Knowledge-Based Systems*, **163**, 2019, pp. 404–415. ⇒241

[70] D. Zhuang, M. J. Chang, M. Li, DynaMo: Dynamic community detection by incrementally maximizing modularity, *IEEE Transactions on Knowledge and Data Engineering*, 2019. ⇒221, 233, 243

Received: June 2, 2021 • Revised: October 19, 2021