Detection of Communities in Large Scale Networks

Baisakhi Chatterjee
Department of Computer Science and Engineering
Institute of Engineering and Management, Kolkata, India
baisakhi.chatterjee95@gmail.com

Himadri Nath Saha
Department of Computer Science and Engineering
Institute of Engineering and Management, Kolkata, India
himadri@iemcal.com

***Abstract*— With the advent of the internet and the boom of information sharing, study of networks has begun to play a central role in social science, mathematics and statistical analysis. A large variety of data can be modeled via complex networks making them increasingly more important for research. However, it is not easy to extract meaningful information from a mesh of interconnected of nodes.. That is why, detection of communities in network has become of utmost importance in recent times. Communities can be said to act as meta-nods. They correspond to functional units of the system and hence, often shed light on the function of the system represented by the network. Detecting an underlying community structure in a network thus allows us to create a map of a network which makes it easier to study. In this paper, we examine a number of research involving detection of communities and summarise them based on their avenues of approach to solving the problem.**

Keywords — Clustering networks, Large scale networks, Community, Community detection, Modularity, Label Propagation, Triadic Closures, Social Networks

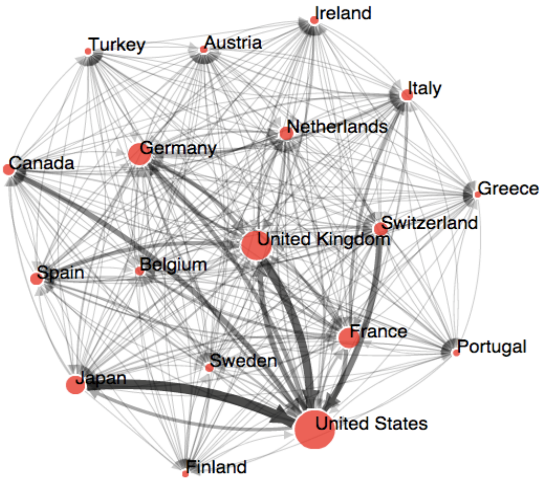

Fig. 1 A complex network

I. Introduction

Networks can be defined as a group or system of interconnected people or things where each link represents a connection. They play a wide role in our everyday lives. From communication networks comprising telephones, mobiles and power grids to biological networks consisting of the genes and proteins in our bodies, networks can come in various forms [1]. In real life, too, we come across several organic networks consisting of people and their lives. Here each person may represent a node while those around them are connected to them via links or edges. Analysing the nature of these relationships is of utmost importance to us.

The empirical study of networks was originally devised in the 1970s as a tool for sociology and social sciences[2] Since then the field has grown exponentially and in recent times, the study of complex networks has gained widespread interest [3][4]. In network theory, complex networks are those which show certain topographical features that are not present in simple networks. Complex networks often feature commonalities like high clustering coefficients, small-world property and community structure. Most social and biological networks fall under this category.

Real networks often show discernible community structures [5][6]. A network is said to have community structure if the nodes present in the network form distinct groups with each other. For distinct communities, this means that, each such set of node has a series dense amongst themselves and sparser connections with nodes outside the community. Groups may be both overlapping or non-overlapping. Mathematically, given a graph G(V,E) where V is the set of vertices and E is the set of edges or links, community detection attempts to partition the nodes into sets of {C1,C2,C3⋯.Cn} where Ci = {vj}, vj ∈ V. Each Ci is a community.

Communities often have very different properties than the average property of the network. For example, in a given social network, both cat lovers and dog lovers might exist simultaneously. However, if we only focus on the network as a whole, we may identify only animal lovers, and miss the nuanced information. This makes communities significant since if we only concentrate on the average properties of a network, we may miss the characteristics of the different groups

978-1-7281-2530-5/19/$31.00 ©2019 IEEE

present in it. But if we can identify each community separately, we may be able to extract more information out of them. Identification of communities therefore allows us to glean vital information from a network and study and detect various functions represented by it.

II. Related Works

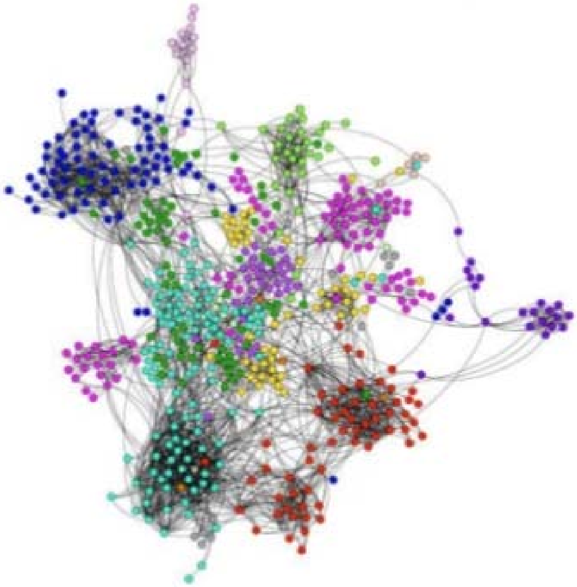

Fig. 2 Community Structure

The topic of community detection has always been ill-defined and precise notations have remained evasive. Thus it has been difficult to present a robust review that considers all aspects of this problem. Most surveys often focus on a certain type of network in order to mitigate this problem. Papadopoulos et al., Wang et al. and Bedi et al. [7][8][9] have focused their review on community detection in the context of social media. Their works focus on algorithms which try to identify characteristics and groups from networks based on social media mining. Jia et al. [10] have presented their review on both social and biological networks, while Tripathy et al. and Barek et al [11][12]. have only studied community detection for biological networks.

Even when some surveys do present a wider perspective, often fail to categorize the different metrics used for community detection. Fortunato and Hric [13] have defined what community detection is, mentioned the benchmarks for validation of methods and finally, outlined a critical analysis of different methods. They discuss the advantage of the number of communities beforehand, how to approach finding significant clusters in an evolving network and suggest methods that currently appear to be most promising. Javed et al [15] have put forth a review of prevailing community detection algorithms, ranging from traditional to modern. Their goal is to highlight the strength and weaknesses of each approach while also discussing different aspects of their performance when tested against benchmark algorithms. They also discuss various challenges and open issues faced by the community detection algorithms. Schaub et al. [14], on the other hand, have claimed that the practice of reviewing against benchmark algorithms can be misleading. In their work, they discuss a focused review of the different motivations that underpin community detection. They conclude that no universal community detection algorithm is possible. Instead, it is more efficient to tailor each algorithm to the problem it is trying to solve.

In our survey, we aim to present a comprehensive review of the prevailing community detection techniques, categorized by their approach. Moreover, we have presented a detailed analysis on modern approaches using parallel computing. A key problem of community detection is computational time required. By using distributed systems, researchers have aimed to solve this issue.

III. Community Detection Algorithms

Community structures are quite common in real networks. Social networks, for example consist of common groups based on location, demographics, interests, religion etc. As previously discussed, finding an underlying community structure lets us identify properties which may not be present in the entire network. Unfortunately, there exists no universal method that can be employed to partition nodes into clusters. In this paper we will review different common methods of community detection and attempt to discuss their benefits and shortcomings.

A. Modularity Maximization

Modularity is a measure of the strength of the community structure and helps to choose the number of communities into which a network should be divided. Modularity is defined as the difference between the edges in the given graph and the edges possible in the random graph i.e., the number of edges within a community minus the expected number of edges in an equivalent network with edges placed at random. The goal of modularity is to quantify the strength of the partition. Thus, the higher the modularity, the stronger is the community. Modularity maximization is highly effective at discovering community structure in real-world network data.

Modularity was first proposed by Girvan-Newman in 2004[16] as a metric for community detection. They defined modularity Q as

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(c_v, c_w) \qquad (1)$$

where m is number of edges, A_{vw} is Adjacent Matrix,

$$A_{vw} = \begin{cases} 1, \text{if vertices v and w are connected} \\ 0, \text{otherwise} \end{cases},$$

$$k_v = \sum_w A_{vw}, \text{(degree of node v)} \text{ and}$$

$$\delta(i, j) = \begin{cases} 1, \text{if } i = j \\ 0, \text{otherwise} \end{cases}$$

Originally, Girvan-Newman[17] proposed a divisive hierarchical clustering method that iteratively splits the network into communities iteratively by removing edges from the network.. If any inter-community edges joins two different communities, then all paths through the network from vertices in one community to vertices in the other must pass along one of those edges. If we can count the number of paths that go along each edge in the graph then we expect the number to be the largest for the inter-community edges. This provides a method for identifying communities.

Later refining this work, Girvan-Newman[16] described a new class of algorithms for extracting the natural community structure from networks of vertices and edges. In this method we use a 'divisive' technique which iteratively removes edges from the network, breaking it up in communities. A set of edge betweenness measures identifies the edges to be removed and edge betweenness scores are re-evaluated after the removal of every edge. This iterative step, which is vital to the success of the algorithm, was absent in the authors' previous work. This algorithm, however, might produce some communities that have no meaningful community structure. For example, placing all vertices in a single community would give a good result while giving no relevant information.

Clauset et.al [18] proposed a new algorithm, referred to as CNM, which infers community structure from network topology. CNM is a greedy algorithm which aims to optimize the modularity. It is considerably faster than most previous general algorithms, and allows identify communities in networks which were previously deemed to be too large to be processed. The algorithm discovers clear communities within this network and utilizes lesser memory space than the algorithm proposed by Girvan-Newman[16], especially in case of sparse matrices.

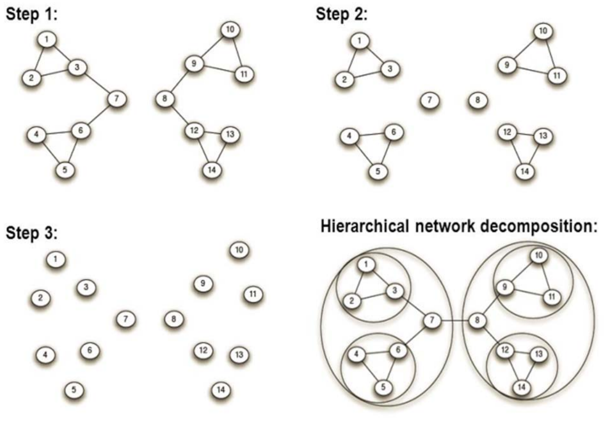

Fig. 3 Concept of Edge Betweenness as proposed by Garvin-Newman-Example

Improving their algorithm even further, Newman[19] stated that the problem of modularity detection can be formulated as a matrix by incorporating the concept of eigenvalues and eigenvectors. The new algorithm so formed, aims exploit this transformation in order to improve on the authors' previous work. When compared, modularity matrix reliable outperformed previous general-purpose algorithms in terms of both speed of execution and quality of results.

Modularity maximization, however, has two opposite yet coexisting problems. In some cases, it tends to split large communities into multiple smaller communities, while, in other cases, it merges communities that are smaller than a certain threshold forming large communities by. The latter problem is also known as the resolution limit problem. As observed by Fortunato and Barthelemy [20], for sufficiently large graphs, these algorithms, failed to detect sufficiently small communities even if they were densely linked. In recent times, a number of algorithms have been proposed which try to solve this problem.

Chen et al. [21] analyzed the modularity metric and proposed an algorithm that would maximize modularity in order to effectively determine network community structure and remove resolution limits. The authors presented an algorithm that iteratively improves the GN quality metric (Q) [16] by splitting and merging the graph. The modularity metric for this algorithm (Qds) was compared with Girvan-Newman (Q) [16] against four real networks, as well as, classical clique and LFR benchmark networks. Results showed that fine-tuned the algorithm dramatically improved performance. However, the metrics used for evaluation were not always consistent for Q.

Xiang et al. [22] proposed a parameterized local modularity using self-loop as a criterion. Each vertex is assigned a self-loop that depends on resolution parameter and community division. The number of links in the community is also considered, as opposed to only considering total links in the entire network. The new metric was used to measure various sized LFR networks. For small graphs, local and global modularity did not show much difference, however, for graphs with nodes numbering over 5000, local modularity clearly outperformed global modularity where mixing parameter was >= 0.3.

Another strategy to mitigate resolution limit is presented by Xiang et al [23] where the authors use edge-reweighting strategy to better cluster communities. The paper lists 12 similarity functions that can calculate similarity nodes and proposes appropriate modifications to optimize them for use. This strategy was utilized for a number of algorithms including GN[16], Newman's greedy algorithm (NF)[24], Blondel (BF)[25], InfoMap[26], COPRA[27], OSLOM[28]. Each similarity measure was computed and the difference in NMI of each reweighted network to that of the previous one was compared for the different algorithms. In GN networks the similarity function PA failed to provide fruitful results. The rest performed well in all cases except BF.

Ghoshal et al. [29] proposed a simplified version of modularity for ease of computation. In their algorithm, they compared the number of neighbours a node had inside a community to the total number of neighbours of the node. This ratio was named node entropy. The algorithm then computed the average of all nodes in a community to find community entropy. A node would only be considered a part of the community if it increased the community modularity. To keep the communities well defined and mitigate the resolution limit, a maximum diameter was set as a parameter and checked on each successive iteration. A sequential version of the algorithm was tested and compared with GN [24] and Louvain [25] networks using well known real world networks. The results showed that the new metric performed similarly or better than the original metric in all cases. A maximum diameter of 4 usually provided best results.

B. Spectral Analysis

Clustering is one of the most widely used techniques for exploratory data analysis. Compared to the “traditional algorithms” such as k-means[30] or single linkage [31], spectral clustering has many fundamental advantages as it is very simple to implement and can be solved efficiently by standard linear algebra methods. Spectral clustering includes transforming the initial set of objects into a set of points in space. The coordinates of the points are elements of eigenvectors. This set of points is then clustered via existing clustering algorithms. Algorithms using spectral clustering very often show better performance than the traditional approaches [32]

Pons e al. [33] have proposed an algorithm, called Walktrap, which efficiently computes the community structure in a network based on the random walk length between graph nodes. A new distance metric 'r' to measure similarity between vertices (and between sets of vertices) is introduced that relates to the spectral properties (eigenvalues) of the transition matrix P of random walk processes. A comparison between the Walktrap algorithm and other previously proposed methods shows that Walktrap can form good quality partitions in large graphs(upto 300,000 vertices) at high speed.

Zhang and You[34] propose a method where the similarity between two points is related not only to the two points , but also to their neighbors thereby generating a matrix that is close to the ideal matrix which helps to obtain accurate clusters. A method to handle noise items which may cause deterioration of clustering performance is also presented. When compared against other standard algorithms, this method outperformed a number of traditional and other improved spectral clustering algorithms.

Chen and Feng [35] propose a Discriminant cut(Dcut) algorithm which normalizes the affinity matrix with the corresponding regularized Laplacian matrix. The method is feasible as it solves a eigenvalue decomposed problem rather than solving intractable NP-hard graph cut problem. The algorithm is tested using Toy-data and real-data experiments and in all context Dcut outperforms the traditional spectral clustering algorithms by generating much precise clusters. This makes it more robust since it is less sensitive to perturbation.

Hardiman and Katzir[36] defines an efficient algorithm for computing the clustering coefficient and size of large-scale social network. These algorithms use the data collected by random walk[37] and does not explore the ego network[38]. It assumes that the nodes of graph is not known and does not require a tailored sampling distribution. The algorithms were tested on publicly available social network datasets. For social-network graphs the algorithms considerably outperform prior works, as well as, any analytic bounds placed on the number of steps required for convergence.

Despite being of considerably high quality, most of the works discussed so far are not suitable for dynamic networks as they are designed for static networks. Qin et al.[39] propose a multi-similarity spectral (MSSC) method as an advancement to the former evolutionary clustering method. To detect the community structure in dynamic networks, the method considers the different similarity metrics of networks. It was seen that the proposed MSSC method outperformed different baseline models on some widely used synthetic and real-world datasets with dynamic ground-truth community structure.

Liu et al. [40] present a new concept of community attractiveness in order to detect communities in large-scale, internet social networks. Attractiveness Based community detection (ABCD) algorithm first merges the communities with larger attractiveness and then the cluster density (node weight) and cluster attractiveness matrix is updated. Both ABCD algorithm and CNM[18] algorithm was tested on data set got from Sina micro-blog, which contains 70 thousand users and 0.6 million bi-connect links. It can be seen that ABCD algorithm generates more distinct communities of smaller size thus making network partitioning much more prominent and clear.

Tsung et al.[41] propose a heuristic community detection algorithm, Vector Partition at Polar Coordinate (VPPC), that partitions the network into non-overlapping communities so as to maximize the modularity density. VPCC first partitions the vertex vectors into groups according to the angle and then moves them to the initial communities based on change of modularity density. Noise reduction techniques and common friends model, strive to solve the over-partitioning of a single community within weaker community structure. VPPC provides higher accuracy than Fine-Tuned Qds. VPCC overcomes the problem of using ratio-cut method to partition the community.

C. *Truiadic Closure*

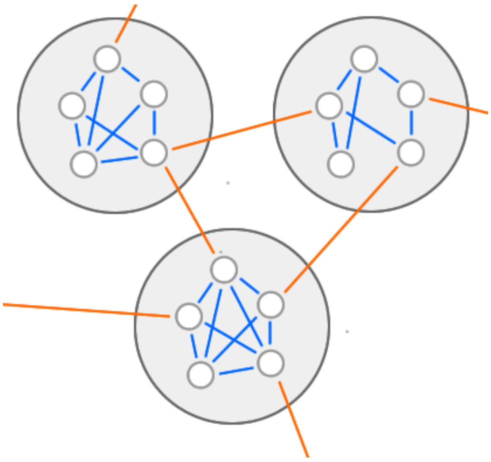

Fig. 4 Strong ties in community and weak ties outside community

The Strength of Weak Ties [42] espoused the importance of triadic closures in networks. The paper claimed that if three nodes A, B,C are such that A has a connection with both B and C, each of B and C should have a connection with each other. This theory, henceforth referred to as Triadic Closure has been extensively for graph structures. Several algorithms have exploited this structural property to cluster communities accurately. General wisdom suggests that the greater the structural density in a group, the stronger is the community. The following section summarizes recent works in this field.

Prat-Perez et al.[43] have proposed a metric that can identify disjoint communities from a given graph by considering their internal structures. The paper summarizes four properties that should be considered for community detection, viz, internal structure, linear increase of connections with increasing size of community, bridge connections and absence of cut vertices. They proposed a metric WCC (Weighted Community Clustering) that considers number of triangles present in a potential community in order to maximize cohesion. They validate their findings using synthetic and real world graphs.

Tsintos and Tsaparas [44] attempt to solve the problem of characterizing edges as strong or weak as originally proposed in The Strength of Weak Ties. The theory states that if a node A has two strong connections to each of B and C, those two must also have a strong connection. To enforce this property, the authors present two algorithms – maximizing the number of strong edges (Max-STC) or minimizing the number of weak edges (MinSTC). It can be proved that MaxSTC is an NP-hard problem, thus, the focus is kept on optimizing MinSTC. The algorithms were applied on five real world datasets. The greedy algorithm was consistently found to label more number of edges as strong. Thus, this metric needs further research

Klymko et al. [45] theorized a method to detect communities in directed graphs. Extending the triangle property for closed connections, the authors detected seven types of triangles that may be formed in directed networks. The paper postulated a metric to transform the graph into an undirected graph by assigning weights based on the type of connection and thus simplify computation. METIS was used for experimental analysis and comparison with other metrics showed negligent decrease in modularity, while there was a significant reduction in percentage of 3-cycles cut. However, this scheme lacks optimization

Liu et al. [46] designed an algorithm that could analyse networks with negative links, e.g., where relationships may be classified into friends or foes. The basic premise of this algorithm is partition nodes such that positive links are present inside communities and negative links are present in between communities. Thus, it is a multi-objective problem. Based on this theory, the signed similarity measure is formulated. A decoder was proposed that could accept the signed network as an indirect representation and partition it into communities. However, this algorithm cannot account for directed networks

Collingsworth and Menezes [47] developed a self-organized algorithm (SOCIAL) for decentralized detection of network communities. They used an extended form of Shannon entropy which could considered the number of triadic closures present in the graph. Entropy was defined as

$$S = -\sum_{i=1}^{n} \frac{p_i \log_2 p_i}{e^{\frac{k}{4}}} \qquad (2)$$

Based on this entropy, each node decided which community it would join in parallel. The algorithm was tested using benchmark graphs against well-known algorithms like Girvan-Newman[17],[24], Blondel et al[25]. and was found to be comparable in speed and accuracy. Moreover, it did not suffer from resolution limit that is a severe disadvantage for the other two methods.

Charalampos E. et al. [48] further studied triangle motifs in connected graphs. The authors extend the idea of conductance in graphs by adding a framework for triangle conductance and thus assigning weights based on number of triangles present. This proves problematic and hence the method is further refined in an algorithm termed TECTONIC such that new weights are assigned based on degree of nodes as well as number of triangles. Edges less than a particular threshold are removed. The algorithm was tested with Amazon, DBLP and YouTube networks and was compared with MCL[49], InfoMap[28] and GN[26] algorithms. TECTONIC proved to be the fastest and had comparable precision in all cases. It was computed on a 1.7 GHz Intel Core i7 processor with 8GB memory. Since the algorithm only uses node degree and number of triangles to make its decisions, it may be optimized to run in parallel.

D. Label Propagation

Label propagation is an algorithm for finding communities, proposed by Raghavan et al. [50]. The advantage of label propagation lies in its computational time and in the fact that it does not require prior information about the network Initially, every node is assigned a unique label representing the community it belongs to. Based on the label of the neighboring node, membership in a community may change. The maximum number of nodes with the same label present within one degree of the nodes determines how this change occurs, The labels thus propagate through the network, giving this algorithm its name. However, this algorithm fails to produce any unique solution. Instead it produces an aggregate of many solutions, which makes it not reliable. In this subsection, we discuss some of the label propagation algorithms

Guendouz et al. [51] uses a variant of the label propagation technique by optimizing the Fireworks Detection Algorithm for community detection. The fireworks detection algorithm has four keys – explosion operator, mapping rule, gaussian operator, selection strategy. The authors present a new discrete version by modifying the initialization phase and mutation operators, and by using modularity density as a criterion for the fitness function, the algorithm can effectively select the final number of sparks in a set. The algorithm was tested on synthetic, as well as, real-world networks, and pitted against well-known algorithms like CNM[18], InfoMap[28], and GA-Net[52]. Results showed DMWFA considerably outperforming the other three algorithms, except in the case of Krebs' Books on US Politics network where the GA-net proved superior. However, this algorithm cannot function with signed networks, which the authors plan to rectify in the future.

Zhang et al [53] proposed a label propagation algorithm that used node information metric to label communities. The Node Importance Label Propagation Algorithm (LPA_NI) calculates the importance of each node in a graph based on user data. Thus, when assigning labels, it uses this value to compute the most important node present in a cluster, thereby eliminating the random selection in LPA. Thus, the algorithm is more stable and consistent. The algorithm was compared with LPA and NIBLPA[54] using metrics like NMI and Modularity. In both cases, LPA_NI performed better, had lesser execution time and was shown to be more stable..

E. Combined Metrics

Modularity and Triadic structures are both important measures for community detection. While Triads posed an interesting metric for judging community strength, modularity still proved to be the benchmark for networks, as well as its general acceptance made it easier to test cases where triads failed. Thus, advances were made using both approaches together. Cohen et al.[55] proposes a method to combine both approaches to better detect communities. Cohen et al. devised a community detection algorithm that generalizes the Louvain method [25] and combines modularity and WCC[43] to accurately map nodes to communities. The algorithm they designed extends both metrics to give WOCC (Weighted Overlapping Community Clustering) and QE (extended modularity), out of which one is chosen dynamically by the algorithm based on number of triangles in graph. A threshold value is set that makes this decision. If the number of triangles are greater, WOCC is computed, otherwise modularity is chosen. The algorithm was analyzed and compared against known methods. It ranked either first or second, losing out by only a small margin in the latter case

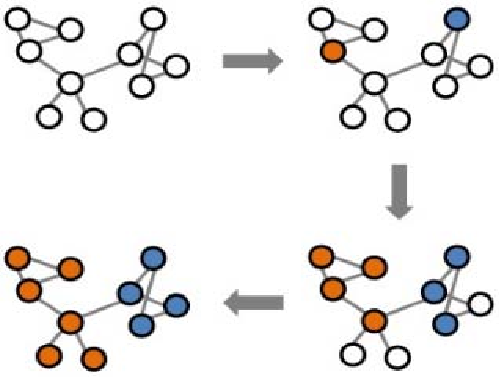

Fig. 5 Label Propagation Algorithm

While Label propagation algorithms are powerful on their own, presence of loops or triangle structures in communities increases the strength of the structure considerably. In the original LPA, all nodes were treated fairly and equally. This caused the algorithm to be somewhat unstable since choice of next node is truly random. Zhang et al.[56] proposed an algorithm that curtails this to a certain degree by prioritizing local cycles and triadic closures when making next choice. This can still be non-deterministic in certain cases, but less so than the original algorithm.

Liu et al [57] present an edge label propagation algorithm (ELPA) that could combine the accuracy presented by link community framework for detecting clusters with the computational efficiency of label propagation algorithms (LPA). identified nodes based on edge clusters and accounted for triadic closures. ELPA can detect overlapping communities. It was compared with well-known algorithms (both overlapping and disjoint), like Linkcomm[58], InfoMap[28], Copra[29], LPA[50], NeTA[59] with various parameters, where applicable. For synthetic networks of disjoint communities, ELPA has comparable accuracy with the other algorithms except Linkcomm which fares poorly. In case of overlapping communities, ELPA again shows comparable accuracy, except in the case of high mixing parameter and large overlap, where Linkcomm outperforms all of the other algorithms.

F. Parallel Computing

A major problem faced during community detection is the speed of computation. As network size continues to increase, large execution time creates a major problem. A new area of research focuses on using parallel algorithms and strategies to speedup computational time. Usage of GPUs aid tremendously in this aspect. As already mentioned, SOCIAL [47] is distributed algorithm which uses triadic closures to determine communities. Apart from this, a number of different algorithms have been proposed.

Yi Song and Bressnan [60] proposed an algorithm for fast community detection by exploiting data parallelism. The algorithm uses degree of neighbours of each vertex and clustering coefficient The authors compared their algorithm to well-known community detection algorithms like InfoMap[28], WalkTrap[61] and Girvan-Newman[25] using synthetic and real world datasets. FCD performed better than InfoMap and GN in all cases, and proved better than WalkTrap in a few cases. It proved to be the faster than all three.

Bu et al. [62] devised a fast algorithm (FPMQA) that works in parallel to optimize detection of communities in networks. The paper identified CNM [18] as the fastest algorithm which optimized modularity and demonstrated how the merging of communities could be accomplished in parallel. Initially every node is assigned a community and at every iteration, a local maximum is computed and if ΔQ is positive, two communities are merged. FPMQA combines the advantages provided by CNM and FUC[27] to categorize nodes faster without compromising quality. The algorithm was analysed on multiple real and ground-truth networks and performed very well in all cases.

Staudt et al. [63] proposed three independent algorithms which could operate in parallel systems. The first was a simple label procedure algorithm which the authors named PLP. A refinement phase was added to the second method and it was named PLMR. And finally, the authors proposed a combination of the two algorithms, called EPP. All the algorithms were reviewed and tested. PLM performed reasonable well in detecting communities, while, PLP proved to be an extremely fast algorithm capable of processing 50 million edges in seconds. EPP however, fared worse than PLM and PLMR in some cases, which makes it irrelevant.

A Shehbab et al. [64] used presented a parallel implementation of FCM[65],[66] originally proposed by Zhang et al [67]. The paper attempts to parallelize the FCM and modularity portion of the algorithm. For this both CPU and GPU have been used. This is called Hybrid CPU-GPU (HCG). Multiple features studied.

Richard Forster[68] developed a parallel version of the Louvain method and implemented it using CUDA programming. The algorithm executes in multiple phases where each phase has multiple iterations. After each phase, the new phase takes data from the previous phase as its input. This continues till modularity gain is above a threshold value. The algorithm had comparable results to the sequential version.

Fender et.al [69] developed a parallel approach for computing the modularity clustering. In their approach, they approximate modularity by looking at the largest eigen pairs of the weighted graph adjacency matrix which has been perturbed by a rank one update. This formula is then generalized to detect multiple clusters at the same time. By taking advantage of Lanczos eigenvalue solver and k-means algorithm, they then implemented a fast parallel version of this algorithm on the GPU. The advantage of this implementation is that the number fixed clusters is arbitrary and does not need to be a power of two. The algorithm successfully processed networks with up to hundred million edges in less than a second. Compared to previous state-of-the-art results, it even showed speedups of up to 8 times was achieved, even when compensating for bandwidth difference of up to 3 times due to lack of hardware resources.

IV. Discussion

Community detection can be viewed through a range of different lenses. The problem we face when it comes this event is how to select the best approach that could singlehandedly detect and identify communities across a variety of network. The problem is thus creating a generic algorithm that can satisfy most, if not all, criteria. Modularity metric tried to solve this problem by quantifying good communities based on graph structures. However, as we have already mentioned resolution limit hindered this. Xiang et al. solved this issue to a certain degree in 2016. Ghoshal et al, presented a simplified modularity definition for ease of computation. Modularity was improved upon using spectral analysis and recent focus on this area developed the method. Tsung et al.[41] refined the work proposed by Chen et al.[21] by the advent of spectral analysis. Dynamic network detection was also made possible by Qin et al. [39]

However, the papers discussed so far failed to account for triangles present in communities. Triangles, or triadic closures are vital to community detections as proposed in Strength of Weak Ties [42]. Long et al. [70], Cohen et al. [55], Collingsworth and Menezes[47] proposed algorithms to utilize this property. Cohen et al. combined his work with dynamic selection of Modularity to further refine the procedure.

Another avenue of difficulty faced in this endeavor is the speed of execution. With the advent of larger and larger graphs, it is imperative to find algorithms that run in short time. Label Propagation Algorithms with their propensity for working with individual nodes have been particularly useful for implementing parallel schemes. Liu et al [57] went so far as to make it deterministic be removing randomness. Parallel execution of algorithms is also a solution. Staudt et al. [63] proposed parallel versions of both LPA and Louvain method. Richard Forster [68], Shehbab et al.[64] used CUDA programming to present parallel versions of

Louvain method and FCM respectively. Fender et al[69] used modularity and spectral partitioning in parallel using CUDA programming. Table 1 summarizes our findings so far.

Table 1 Advantages and Disadvantages of Different Approaches

Sl No	Approach	Advantages	Disadvantages
1.	Modularity	1. Simple to Implement 2. Usually performs well	1. Cannot detect communities for certain types of networks 2. Resolution Limit is a major problem
2.	Spectral Clustering	1. Can be solved with relational algebra methods which makes it simple 2. Often outperforms traditional methods	1. Computationally very expensive 2. Needs approximations
3.	Triadic Closures	Considers the structure of the network in order to determine communities	May fail when clear structures are not defined or in case of overlapping communities
4.	LPA	1. Relatively fast 2. Does not need prior information about the network	1. Does not consider the structure of the network 2. No unique solution
5.	Distributed Algorithms	1. Computationally very fast 2. Often outperforms traditional methods in terms of processing power	1. Difficult to implement 2. May need additional hardware

V. Conclusion

In our work, we have discussed the problem of detecting communities in large scale networks and the different approaches one can take to solve it. We surveyed works by eminent authors and presented their findings. Based on these results, we have categorized the work into five main criteria, viz., Node modularity, Spectral Clustering, Triadic Loops, Label Propagation and Distributed Algorithms. We have highlighted the benefits and drawbacks of each metric and discussed how combining multiple metrics could yield better results.

References

[1] R. Cohen, S. Havlin, Complex Networks: Structure, Robustness and Function, Cambridge University Press, Cambridge, UK, 2010.

[2] M. Newman, Networks: An Introduction, Oxford University Press, Inc., New York, NY, USA, 2010.

[3]Saleh, Mahmoud; Esa, Yusef; Mohamed, Ahmed (2018-05-29). "Applications of Complex Network Analysis in Electric Power Systems". Energies. 11 (6): 1381. doi:10.3390/en11061381.

[4]Stephenson, C.; et., al. (2017). "Topological properties of a self-assembled electrical network via ab initio calculation". Scientific Reports. 7: 41621. Bibcode:2017NatSR...741621S. doi:10.1038/srep41621. PMC 5290745. PMID 28155863.

[5] Fani, Hossein; Bagheri, Ebrahim (2017). "Community detection in social networks". Encyclopedia with Semantic Computing and Robotic Intelligence. 1. pp. 1630001 [8]. doi:10.1142/S2425038416300019.

[6] Hamdaqa, Mohammad; Tahvildari, Ladan; LaChapelle, Neil; Campbell, Brian (2014). "Cultural Scene Detection Using Reverse Louvain Optimization". Science of Computer Programming. 95: 44–72. doi:10.1016/j.scico.2014.01.006.

[7] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos, “Community detection in Social Media,” Data Mining and Knowledge Discovery, vol. 24, no. 3, pp. 515–554, Jun. 2011.

[8] C. Wang, W. Tang, B. Sun, J. Fang, Y. Wang, Review on community detection algorithms in social networks, in: Progress in Informatics and Computing (PIC), 2015 IEEE International Conference on, IEEE, 2015, pp. 551–555.

[9] P. Bedi, C. Sharma, Community detection in social networks, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 6 (3) (2016) 115–135.

[10] G. Jia et al., “Community Detection in Social and Biological Networks Using Differential Evolution,” in Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2012, pp. 71–85.

[11] B. Tripathi, S. Parthasarathy, H. Sinha, K. Raman, and B. Ravindran, “Adapting Community Detection Algorithms for Disease Module Identification in Heterogeneous Biological Networks,” Frontiers in Genetics, vol. 10, Mar. 2019.

[12] M. B. M’Barek, A. Borgi, W. Bedhiafi, and S. B. Hmida, “Genetic Algorithm for Community Detection in Biological Networks,” Procedia Computer Science, vol. 126, pp. 195–204, 2018.

[13] S. Fortunato and D. Hric, “Community detection in networks: A user guide,” Physics Reports, vol. 659, pp. 1–44, Nov. 2016.

[14] M. T. Schaub, J.-C. Delvenne, M. Rosvall, and R. Lambiotte, “The many facets of community detection in complex networks,” Applied Network Science, vol. 2, no. 1, Feb. 2017.

[15]Javed, M. A., Younis, M. S., Latif, S., Qadir, J., & Baig, A. (2018). Community detection in networks: A multidisciplinary review. Journal of Network and Computer Applications, 108, 87–111. doi:10.1016/j.jnca.2018.02.011

[16] M.E.J. Newman, Mark & Girvan, Michelle. (2004). “Finding and Evaluating Community Structure in Networks”. Physical review. E, Statistical, nonlinear, and soft matter physics. 69. 026113. 10.1103/PhysRevE.69.026113.

[17] Girvan, M. & Newman, M. E. J., “Community structure in social and biological networks”, Proc. Natl Acad. Sci. USA 99, 7821-7826, 2002

[18] Aaron Clauset, M. E. J. Newman, and Cristopher Moore, “Finding community structure in very large networks”, Phys. Rev. E 70, 066111, 2004

[19] M. E. J. Newman, “Modularity and community structure in networks”, Proc. Natl. Acad. Sci. USA 103, 8577-8582, doi: 10.1073/pnas.0601602103, 2006

[20] Santo Fortunato, Marc Barthelemy,“Resolution limit in community detection”, Proc. Natl. Acad. Sci. USA 104 (1), 36-41,doi: 10.1073/pnas.0605965104, (2007)

[21] Mingming Chen, Konstantin Kuzmin, Boleslaw K. Szymanski, "Community Detection via Maximization of Modularity and Its Variants", IEEE 2014

[22] Ju Xiang, Tao Hu, Yan Zhang, Ke Hu, Jian-Ming Li, Xiao-Ke Xu, Cui-Cui Liu, Shi Chen, "Local modularity for community detection in complex networks", Physica A: Statistical Mechanics and its Applications Volume 443, 1 February 2016, Pages 451-459

[23] Ju Xiang-Ke Hu-Yan Zhang-Mei-Hua Bao-Liang Tang-Yan-Ni Tang-Yuan-Yuan Gao-Jian-Ming Li-Benyan Chen-Jing-Bo Hu, "Enhancing community detection by using local structural information" - Journal of Statistical Mechanics: Theory and Experiment – 2016

[24] Newman M E J 2004 “Fast algorithm for detecting community structure in networks” Phys. Rev. E 69 066133

[25] Blondel V D, Guillaume J-L, Lambiotte R and Lefebvre E 2008 Fast unfolding of communities in large networks J. Stat. Mech. P10008

[26] Rosvall M and Bergstrom C T 2008 Maps of random walks on complex networks reveal community structure Proc. Natl Acad. Sci. USA 105 1118

[27] Steve G 2010 Finding overlapping communities in networks by label propagation New J. Phys. 12 103018

[28] Lancichinetti A, Radicchi F, Ramasco J J and Fortunato S 2011 Finding statistically significant communities in networks PLoS One 6 e18961

[29] AK Ghoshal, N Das, "On Diameter Based Community Structure Identification in Networks", ICDCN '17 Proceedings of the 18th International Conference on Distributed Computing and Networking Article No. 41, 2017

[30] Lloyd, Stuart P. "Least squares quantization in PCM." Information Theory, IEEE Transactions on 28.2 (1982): 129-137.

[31] F. Murtagh and P. Contreras, “Algorithms for hierarchical clustering: an overview,” Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 2, no. 1, pp. 86–97, Dec. 2011.

[32] Ulrike von Luxburg, “A Tutorial on Spectral Clustering”, Technical Report No. TR-149, 2006

[33] Pascal Pons, Matthieu Latapy,“Computing communities in large networks using random walks”, Proceedings of the 20th international conference on Computer and Information Sciences, ISBN:3-540-29414-7 978-3-540-29414-6,doi: 10.1007/11569596_31, 2005

[34] Xianchao Zhang, Quanzeng You,” An improved spectral clustering algorithm based on random walk”, doi:10.1007/s11704-011-0023-0, 2011

[35] Weifu Chen,Guocan Feng, “Spectral clustering with discriminant cuts”, Journal Knowledge-Based Systems, Volume 28, 2012, ISSN: 0950-7051 doi: 10.1016/j.knosys.2011.11.010

[36] Liran Katzir, Stephen J. Hardiman, “Estimating Clustering Coefficients and Size of Social Networks via Random Walk”, Journal ACM Transactions on the Web (TWEB), Volume 9 Issue 4, 2015, doi: 10.1145/2790304

[37] K. PEARSON, “The Problem of the Random Walk,” Nature, vol. 72, no. 1865, pp. 294–294, Jul. 1905.

[38] C. Jones and E. H. Volpe, “Organizational identification: Extending our understanding of social identities through social networks,” Journal of Organizational Behavior, vol. 32, no. 3, pp. 413–434, Jun. 2010.

[39] Xuanmei Qin, Weidi Dai, Pengfei Jiao, Wenjun Wang, Ning Yuan, “A multi-similarity spectral clustering method for community detection in dynamic networks”, Scientic Reports 6, Article number: 31454,2016, doi:10.1038/srep31454

[40] Ruifang Liu, Shan Feng, Ruisheng Shi, Wenbin Guo, “Weighted Graph Clustering for Community Detection of Large Social Networks”, Procedia Computer Science 31, 2014, 85 – 94, doi:10.1016/j.procs.2014.05.248

[41] Chen-Kun Tsung, HannJang Ho, ShengKai Chou, JanChing Lin, SingLing Lee, “A Spectral Clustering Approach Based on Modularity Maximization for Community Detection Problem”, IEEE 2017, doi:10.1109/ICS.2016.0012

[42] Granovetter, M. S. (1973). "The Strength of Weak Ties" (PDF). The American Journal of Sociology. 78 (6): 1360–1380. JSTOR 2776392. doi:10.1086/225469

[43] Arnau Prat-Pérez, David Dominguez-Sal, Josep M. Brunat, Josep-Lluis Larriba-Pe, "Shaping communities out of triangles", CIKM '12 Proceedings of the 21st ACM international conference on Information and knowledge management, Pages 1677-1681

[44]Stavros Sintos-Panayiotis Tsaparas, "Using Strong Triadic Closure to Characterize Ties in Social Networks" - Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '14 – 2014

[45] Christine Klymko, David Gleich, Tamara G. Kolda, "Using Triangles to Improve Community Detection in Directed Networks", 2014, arXiv:1404.5874

[46] Chenlong Liu, Jing Liu, and Zhongzhou Jiang, "A Multiobjective Evolutionary Algorithm Based on Similarity for Community Detection from Signed Social Networks", IEEE TRANSACTIONS ON CYBERNETICS, VOL. 44, NO. 12, DECEMBER 2014

[47] Ben Collingsworth, Ronaldo Menezes, A Self-organized Approach for Detecting Communities in Networks, R. Soc. Netw. Anal. Min. (2014) 4: 169. https://doi.org/10.1007/s13278-014-0169-5

[48] Charalampos E. Tsourakakis, Jakub Pachocki, Michael Mitzenmacher, "Scalable Motif-aware Graph Clustering", WWW '17 Proceedings of the 26th International Conference on World Wide Web, Pages 1451-1460

[49] S. v. Dongen. Graph clustering by flow simulation. 2000.

[50] U.N.Raghavan – R. Albert – S. Kumara "Near linear time algorithm to detect community structures in large-scale networks", 2007

[51] Guendouz, M., Amine, A. & Hamou, R.M. Appl Intell (2017) 46: 373. ttps://doi.org/10.1007/s10489-016-0840-9

[52] Pizzuti C (2008) Ga-net: A genetic algorithm for community detection in social networks. In: Parallel Problem

[53]Xian-Kun Zhang-Jing Ren-Chen Song-Jia Jia-Qian Zhang, "Label propagation algorithm for community detection based on node importance and label influence" - Physics Letters A Volume 381, Issue 33, 5 September 2017, Pages 2691-2698

[54] Y. Xing, F. Meng, Y. Zhou, M. Zhu, M. Shi, and G. Sun, "A Node Influence Based Label Propagation Algorithm for Community Detection in Networks," The Scientific World Journal, vol. 2014, pp. 1–13, 2014.

[55] Cohen Y., Hendler D., Rubin A. (2017) Node-Centric Detection of Overlapping Communities in Social Networks. In: Shmueli E., Barzel B., Puzis R. (eds) 3rd International Winter School and Conference on Network Science. NetSci-X 2017. Springer Proceedings in Complexity. Springer, Cham

[56] Xian-Kun Zhang, Song Fei, Chen Song, Xue Tian, Yang-Yue Ao, "Label propagation algorithm based on local cycles for community detection" - International Journal of Modern Physics B - 2015

[57] Wei Liu, Xingpeng Jiang, Matteo Pellegrini & Xiaofan Wang,"Discovering communities in complex networks by edge label propagation" - Scientific Reports 6, Article number: 22470 (2016)

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

[59] Wei Liu, Matteo Pellegrini & Xiaofan Wang. Detecting Communities Based on Network Topology. Sci. Rep. 4, 5739 (2014).

[60] Yi Song and S. Bressnan, "Fast Community Detection" - DEXA 2013 Proceedings of the 24th International Conference on Database and Expert Systems Applications - Volume 8055 Pages 404-418

[61] Pons, P., Latapy, M.: Computing communities in large networks using random walks. In: Yolum, p., G¨ung¨or, T., G¨urgen, F., ¨Ozturan, C. (eds.) ISCIS 2005. LNCS, vol. 3733, pp. 284–293. Springer, Heidelberg (2005)

[62] Zhan Bu, Chengcui Zhang, Zhengyou Xia, Jiandong Wang, "A fast parallel modularity optimization algorithm (FPMQA) for community detection in online social network" - Knowledge-Based Systems Volume 50, September 2013, Pages 246-259

[63] Christian L. Staudt, Henning Meyerhenke, "Engineering Parallel Algorithms for Community Detection in Massive Networks" - IEEE Transactions on Parallel and Distributed Systems archive Volume 27 Issue 1, January 2016 Pages 171-184

[64] Mohammed Alandoli, Mohammed Shehab, Mahmoud Al-Ayyoub, Yaser Jararweh, Mohammad Al-Smadi, "Using GPUs to speed-up FCM-based community detection in Social Networks" - Computer Science and Information Technology (CSIT), 2016 7th International Conference on July 2016

[65] J. C. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters," 1973.

[66] J. C. Bezdek, Pattern recognition with fuzzy objective function algorithms.

[67] S. Zhang, R.-S. Wang, and X.-S. Zhang, "Identification of overlapping community structure in complex networks using fuzzy c-means clustering," Physica A: Statistical Mechanics and its Applications, vol. 374, no. 1, pp. 483–490, 2007.Springer Science & Business Media, 2013.

[68] Richard Forster, "Louvain Community Detection With Parallel Heuristics On GPUs" - Intelligent Engineering Systems (INES), 2016 IEEE 20th Jubilee International Conference on July 2016

[69] Alexandre Fender, Nahid Emad, Serge Petition, Maxim Naumov, "Parallel Modularity Clustering", Procedia Computer Science Volume 108,2017, Pages 1793-1802, doi:10.1016/j.procs.2017.05.198

[70] Hua Long, Baoan Li, "Overlapping Community Identification Algorithm in Directed Network" - Procedia Computer Science, Volume 107, 2017, Pages 527-532